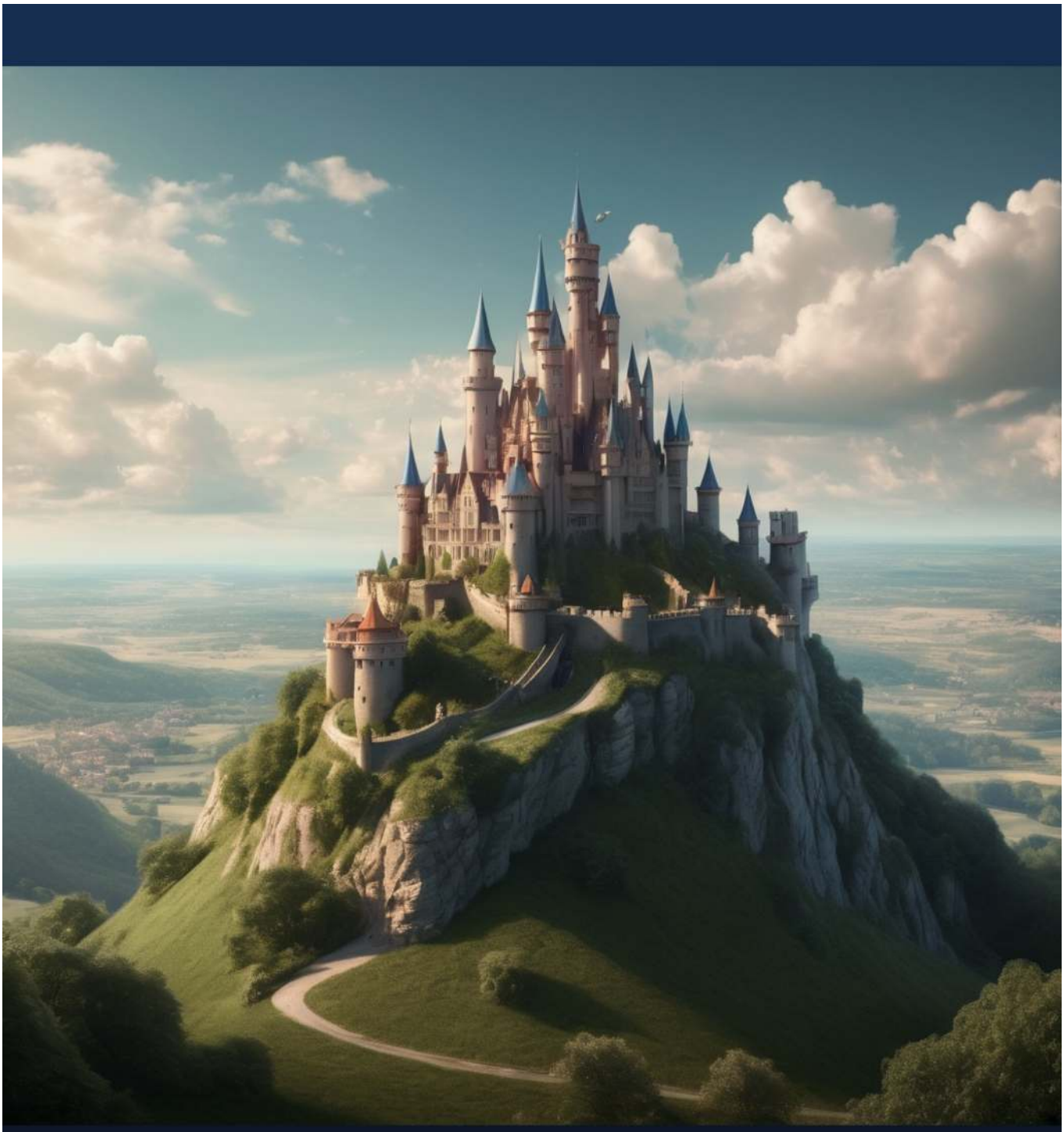
LINEAR MODELS AND RURITÀ KRÀLOVSTVÌ



Ole Forsberg
Knox College

LibreTexts™



Linear Models and Řuritá Kràlovstvì

Using the Kingdom for Greater Insight

Version 0.704442η

This text is disseminated via the Open Education Resource (OER) LibreTexts Project (https://LibreTexts.org) and like the thousands of other texts available within this powerful platform, it is freely available for reading, printing, and "consuming."

The LibreTexts mission is to bring together students, faculty, and scholars in a collaborative effort to provide an accessible, and comprehensive platform that empowers our community to develop, curate, adapt, and adopt openly licensed resources and technologies; through these efforts we can reduce the financial burden born from traditional educational resource costs, ensuring education is more accessible for students and communities worldwide.

Most, but not all, pages in the library have licenses that may allow individuals to make changes, save, and print this book. Carefully consult the applicable license(s) before pursuing such effects. Instructors can adopt existing LibreTexts texts or Remix them to quickly build course-specific resources to meet the needs of their students. Unlike traditional textbooks, LibreTexts' web based origins allow powerful integration of advanced features and new technologies to support learning.



LibreTexts is the adaptable, user-friendly non-profit open education resource platform that educators trust for creating, customizing, and sharing accessible, interactive textbooks, adaptive homework, and ancillary materials. We collaborate with individuals and organizations to champion open education initiatives, support institutional publishing programs, drive curriculum development projects, and more.

The LibreTexts libraries are Powered by NICE CXone Expert and was supported by the Department of Education Open Textbook Pilot Project, the California Education Learning Lab, the UC Davis Office of the Provost, the UC Davis Library, the California State University Affordable Learning Solutions Program, and Merlot. This material is based upon work supported by the National Science Foundation under Grant No. 1246120, 1525057, and 1413739.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation nor the US Department of Education.

Have questions or comments? For information about adoptions or adaptions contact info@LibreTexts.org or visit our main website at https://LibreTexts.org.

This text was compiled on 04/19/2026

TABLE OF CONTENTS

Licensing

Preface

1: An Introduction to Ruruitania

- 1.1: The Background of Ruritania
- 1.2: The Geography
- 1.3: The Government
- 1.4: The Economy
- 1.5: US-Ruritanian Relations
- 1.6: Illustrations of Analyses
  - 1.6.1: Illustration- The Missing Kraj
  - 1.6.2: Illustration- Cows in Děčín
  - 1.6.3: Illustration- Ruritanian Crops
  - 1.6.4: Illustration- Wealth in Ruritania
  - 1.6.5: Illustration- Elections in Ruritania
  - 1.6.6: Illustration- Insurance in Ruritania
  - 1.6.7: Illustration- Warmth for the King
- 1.7: Conclusion

2: Part One- Least Squares

3: Intro to Linear Regression

- 3.1: Scalar Representation
- 3.2: First Results
- 3.3: Our First Assumptions
- 3.4: PRE Measures
- 3.5: Conclusion
- 3.6: End-of-Chapter Materials

4: Matrices and Linear Regression

- 4.1: Matrix Representation
- 4.2: Predictions and the Hat Matrix
- 4.3: The Geometry of OLS
- 4.4: Multicollinearity and Categorical Independent Variables
- 4.5: Conclusion
- 4.6: End-of-Chapter Materials

5: Improved! Now with Probabilities

- 5.1: Probability Distributions
- 5.2: Test Statistics and Hypothesis Testing
- 5.3: Confidence Intervals
- 5.4: The Working-Hotelling Bands
- 5.5: Conclusion

- 5.6: End-of-Chapter Materials

6: Dood! Check the Requirements

- 6.1: Normality
- 6.2: Functional Fit
- 6.3: Homoskedasticity
- 6.4: Multicollinearity
- 6.5: Conclusion
- 6.6: End-of-Chapter Materials

7: Time for Some Examples

- 7.1: Violent Crime
- 7.2: Violent Crime, Wealth, Region
- 7.3: Cows in Děčín
- 7.4: Conclusion
- 7.5: End-of-Chapter Materials

8: Fixing the Violations

- 8.1: The Issue of Boundedness
- 8.2: The South Sudanese Referendum
- 8.3: Adjustments for Heteroskedasticity
- 8.4: Conclusion
- 8.5: End-of-Chapter Materials

9: Beyond the Ordinary

10: Other Least Squares

- 10.1: Ordinary Least Squares
- 10.2: Weighted Least Squares
- 10.3: Generalized Least Squares
- 10.4: May the (Strong) Force be with You
- 10.5: Elections in Ruritania
- 10.6: Conclusion
- 10.7: End-of-Chapter Materials

11: Quantile Regression

- 11.1: Parameter Estimation
- 11.2: Quantile Regression
- 11.3: Conclusion
- 11.4: End-of-Chapter Materials

12: Maximizing the Likelihood

- 12.1: The Likelihood
- 12.2: The MLE and the CLM
- 12.3: Conclusion
- 12.4: End-of-Chapter Materials

13: Beyond the Classical Model

14: Generalized Linear Models

- 14.1: The CLM and the GLM
- 14.2: The Requirements for GLMs
- 14.3: Assumptions of GLMs
- 14.4: The Gaussian Distribution
- 14.5: Generalized Linear Models in R
- 14.6: Conclusion
- 14.7: End-of-Chapter Materials

15: Binary Dependent Variables

- 15.1: Binary Dependent Variables
- 15.2: Latent Variable Modeling
- 15.3: The Mathematics
- 15.4: Modeling with the Logit
- 15.5: Measuring Accuracy
- 15.6: Modeling with Other Links
- 15.7: Model Selection
- 15.8: Conclusion
- 15.9: End-of-Chapter Material

16: Binomial Dependent Variables

- 16.1: Binomial Distribution
- 16.2: The Mathematics
- 16.3: O Canada!
- 16.4: Sri Lanka in 2010
- 16.5: Beta-Binomial Regression
- 16.6: Conclusion
- 16.7: End-of-Chapter Materials

17: Count Dependent Variables

- 17.1: Linear or Poisson Regression?
- 17.2: The Mathematics
- 17.3: Overdispersion
- 17.4: Body Counts
- 17.5: The Bias-Variance Tradeoff
- 17.6: Conclusion
- 17.7: End-of-Chapter Materials

18: Nominal and Ordinal Dependent Variables

- 18.1: Nominal Dependent Variables
- 18.2: Ordinal Dependent Variables
- 18.3: Cattle Feed
- 18.4: The State University of Ruritania
- 18.5: Conclusion
- 18.6: End-of-Chapter Materials

19: Appendices

20: The Appendix of Matrices

- 20.1: Matrix Basics
- 20.2: Addition
- 20.3: Multiplication
- 20.4: Other Terms and Operations
- 20.5: Consequences
- 20.6: Statistics in Matrices
- 20.7: End-of-Appendix Materials

21: The Appendix of R

- 21.1: Installing R
- 21.2: R Packages
- 21.3: R Functions
- 21.4: Programming Practice
  - 21.4.1: Plotting a Distribution
  - 21.4.2: Functions of RVs
  - 21.4.3: Effects of Rounding
  - 21.4.4: Algorithmic Thinking
  - 21.4.5: Distribution of Extreme Values
  - 21.4.6: Non-Parametric Bootstrap
  - 21.4.7: Distribution of Sample Means
  - 21.4.8: Distribution of Sample Sums

22: The Appendix of Statistics

- 22.1: Importantly Confusing Points
- 22.2: Sample Statistics
- 22.3: Population Parameters
- 22.4: Probability Distributions
- 22.5: Distributions of Sample Statistics
- 22.6: Other Topics
- 22.7: End-of-Appendix Materials

Index

Glossary

Detailed Licensing

Detailed Licensing

Licensing

*A detailed breakdown of this resource's licensing can be found in **Back Matter/Detailed Licensing**.*

Preface

Three Vignettes

Vignette 1: Showing Your Work

I sat there, as I was wont to do, in the Spillane Reading Room, drinking coffee in the early morning hour, trying to find my wakefulness. The air was peaceful, with two first-years discussing and sharing insights about international politics and world events with each other and with me. It was life as I expected it to be in academia.

Frequently, more so as the term wore on, "Simon" would rush into the room and explode, "I don't know what this *thing* is telling me!" Then, he would throw down five pages of printout from a well-known statistical package and throw up his hands, as if beseeching the gods of statistics to send down an answer to him.

After allowing the situation to calm, and the first-years to start breathing again, I would ask the question "What did you do to get the printout?" (Show your work.) For some reason, I always expected the answer to differ from all the times previous; I expected him to tell me what he intended to do, what specific actions he performed, what information those commands were supposed to give him, and why he needed that information.

"I clicked on some menu things and this came out." As expected.

He and I would then sit down and go over the fifteen pages of printout, examining what each of the tables and numbers meant in relation to his research. Eventually, after dealing with several "Why is it giving me *that* information?!" questions, Simon would be vaguely satisfied with the printout and could select several statistics from the printout that would provide the information he sought.

However, there were many more questions I wanted to ask him. Most centered on questions about the validity of the tests performed. I knew, however, that such a line of questioning would be moot with the statistical package he (and his class) used. Either the company that owned the package had the test available, or it did not. There was no (easy) way to add tests and procedures. Thus, Bartlett's test of equal variances was not an option, even when the data was such that it should be analyzed using it. Furthermore, the number of available tests was quite limited.

In addition to the extensibility issue, there was the issue of clicking your way to an analysis. If Simon needed to repeat the exact analysis, except for a tweak or two, he would have to start from scratch and repeat it all, clicking all the same menu items, hoping that he did not make a mistake along the way. This repeat analysis happens quite frequently in life.

Vignette 2: Educational Research

One of the many pleasures I have as a statistician is my exposure to many different disciplines. I consulted for a woman doing research in science education. Her specific problem was to determine if a specific Science Teaching Unit (STU) had the students like science more. She performed her experiment on a class of fifth- and sixth-graders in rural North Dakota (hardly representative). She gave them a 50-question pre-test, taught the Unit, then gave the same students a 50-question post-test (what about reliability?!?!).

She then contacted me and sent the data.

On the data she sent to me, I spent about three hours analyzing the data, coming to some interesting (and counter-intuitive) conclusions. In my experience, researchers have a sufficient feel for their discipline that surprising results are frequently a result of analysis error. Thus, I checked my analysis for errors.

I was actually able to check the analysis because I wrote a script --- a series of commands --- detailing every bit of the analysis I performed. Mouse-clicking my way through the analysis would make it all but impossible to check my work (something my math teachers in grade school always emphasized). Thus, I was confident that the analysis I returned to her were correct...

> ...conditional on the data being correct.

The next day, she emailed me and let me know that she found several serious errors in her data. Relying on mouse-clicks would have made those original three hours a waste of time. However, once she sent the corrected version of the data, the analysis took 90 seconds. Clicking my way to re-analysis would have taken the same amount of time as the first analysis --- time we did not have (we were facing a deadline). Re-running the script only took processing time.

While I *am* a fan of mouse-clicks under many circumstances, I am not a fan when it comes to serious statistical analysis. Scripting provides at least three definite advantages over mouse clicks: You get what you request, you can check your work easily, and you can re-run the analysis with little effort.

Vignette 3: Flexibility

One thing I love about being an academic is that I get to travel the world sharing my research and its results. One June found me standing in front of a formidable group in Odense, Denmark, giving such a presentation. After I had discussed the current literature on the causes of terrorism, my statistical model, the results, and my conclusions, I opened it up for questions. After about five minutes of questions on the validity of my statistical model, one professor asked how my analysis would change were I to add an offset to the model.

After a brief panic, I decided to actually run the regression with the offset in front of the audience. I apologized for not knowing the answer to his question, but I would be happy to hypothesize the effect of the variable offset while running the analysis. The professor smiled as I tried to open my analysis script to modify and run it. It turned out that R was not installed on the presentation computer. No worries. I opened the R folder on my USB drive, double-clicked on the R program, and proceeded with the altered analysis --- all the while discussing the theoretical effects of using such an offset under these circumstances. Before I could finish hypothesizing, R gave me the answer (which, thankfully, agreed with my hypotheses).

Now that I had my model laid bare before all, many more questions arose about different alterations I could (or should) make to the model. All of which I was able to perform in front of the now hyper-interested crowd.

Some Benefits of R

These three vignettes illustrate many strengths of a statistical environment like R. First, it encourages one to write out the analysis and "show the work." This makes it easier to see the entire scope and sequence of the analysis. It also makes it easier to check for errors. Second, it is extensible. If there is a cutting-edge test or procedure you wish to run, there is probably a package that contains it. If not, you are quite free to write it yourself. Finally, one can carry R around on a USB drive, allowing anyone to perform analyses whenever there is a computer, like in Denmark.

Oh yeah, that R is free is also a nice feature, especially as statistical packages can run from $600 to $6000 and up, *and* can have licenses requiring annual payments. As budgets get tighter, an ability to work successfully with a free (and powerful) statistical environment is invaluable.

Book Prerequisites

For any book (or course), there is a necessary assumption made about the background of the reader (or student). For the material in this book, I assume that you have had experiences with elementary statistics, matrices, and calculus.

In **differential calculus**, you will need to understand how to optimize (minimize or maximize) functions. In **integral calculus**, you should understand how to calculate areas under a curve (probabilities related to density functions). Beyond that, there is little calculus needed.

The **matrix topics** you need consist of being able to perform algebra on matrices. Beyond that, anything you remember from a typical linear algebra course will help things make a bit more sense. Matrix rank, invertibility, idempotency, projection matrices, orthogonality, etc., are all important in ordinary least squares. So, if you remember those topics, you will be ahead of the curve. If

you do not remember them, then you will need to (re-)learn them in this course. Appendix M will help with remembering and learning the important matrix topics.

Finally, I wrote this book to be a second course in statistics, one that started where a typical **introductory course** ended. Because of this, I also assume you remember many topics from such a course. These topics definitely include the meanings of confidence intervals and p-values. They also include probability distributions, t-tests, issues with multiple testing, and the Central Limit Theorem (CLT). To help refresh your memory, work through Appendix S. Note that Appendix S also introduces you to some (optional) advanced items. These topics were included at the request of past students who wanted to actually see a proof of the Central Limit Theorem. Rest assured that understanding the CLT is more important than being able to prove it. Furthermore, the proof offers little in the way of a deeper understanding.

The Plan

A Note on Notation

Sadly, notation varies across the discipline. This is a result of the history of statistics: Many of the methods came from disciplines that *used* statistics, rather than from statistics itself. Different disciplines use different notation for the same idea. Thus, any discussion of Survival Analysis needs to include Event History Analysis and Reliability Analysis, as they all study the same phenomena but from different disciplines (medicine, social sciences, and engineering, respectively). Furthermore, the term "reliability analysis" means different things in different areas. It could mean studies of how long until a part or a machine breaks. It could also mean how robust conclusions are to changes in model assumptions.

Even within a discipline, there is often a variety of notation used to indicate the same ideas. For instance, probability functions are often parametrized in different ways. The parameter of the Exponential distribution can be the rate λ or the expected value θ; the second parameter of the Normal distribution (Gaussian distribution, Gauss-Laplace distribution) may be the variance σ^2, the standard deviation σ, or the precision as $\tau = 1/\sigma$ or as $\tau = 1/\sigma^2$. The symbol for the average rate in the Poisson distribution can be μ or λ. In this volume, I will (try to) keep consistent with notation, and I will explain the notation before I use it.

To that end, population parameters will be signified using Greek minuscules. Sets from which the population parameters can belong (parameter spaces) will be Greek majuscules. Both are included in the table below. All random variables are Roman majuscules. All realized random variables (data) are Roman minuscules. Violations of these rules will exist, but should be kept to a minimum.

Table 1: The usual Greek alphabet in the canonical order. Being familiar with the letters will make it easier to recognize the implied meaning behind the letter.

miniscule	Majuscule	name
α	A	alpha
β	B	beta
γ	Γ	gamma
δ	Δ	delta
ε	E	epsilon
ζ	Z	zeta
η	H	eta
θ	Θ	theta
ι	I	iota
κ	K	kappa
λ	Λ	lambda
μ	M	mu
ν	N	nu
ξ	Ξ	xi
o	O	omicron
π	Π	pi
ρ	P	rho
σ	Σ	sigma
τ	T	tau
υ	Y	upsilon
φ	Φ	phi
χ	X	chi
ψ	Ψ	psi
ω	Ω	omega

Thus, if we theorize that our measurements come from a population that is Normally distributed, with mean μ and standard deviation σ, we would specify that $\mu \in \mathbf{M}$ and that $\sigma \in \mathbf{\Sigma}$, where $\mathbf{M} = \mathbb{R}$ and $\mathbf{\Sigma} = (0, \infty)$. Be aware of the difference between $\mathbf{\Sigma}$ and $\sum$. The first is the set of possible values of σ. The second is the symbol indicating the sum of what follows. Usually, the difference will be obvious from context.

Now, if we know that the mean is 15 and the standard deviation is 10, I would write this as

$$\boldsymbol{X} \sim \boldsymbol{N}(\mu = 15;\ \sigma = 10) \tag{1}$$

where the mean of the population is denoted by μ, the standard deviation by σ, and the Normal distribution by Normal.

Once we take those measurements, we would call the variable x. The difference between random variables and *realizations* of those random variables is that the random variable has a probability distribution associated with it; the realized data are just numbers.

By the way, if we wish to specify "parameter" *in general*, we use $\boldsymbol{\theta}$. As a result, the symbol for the generic parameter space is $\boldsymbol{\Theta}$.

Matrices (and vectors) will be indicated with bold-faced letters (as bold as possible, but definitely in serif font). Thus, $\mathbf{x}$ is the data matrix (observed values) and $\mathbf{X}$ is the data matrix (theoretical values).

Between these two, it *does* make sense to say

$$\mathbf{X} \sim N(\mu = 15;\ \sigma = 10) \tag{2}$$

It does NOT make sense to say something similar about $\mathbf{x}$ (the data); the observed data are *not* random variables.

Conclusion

And so, with all of this said, turn the page and begin your trek through linear models. The first chapter introduces you to the topics of both linear models and the Kingdom of Ruritania. The former is the purpose of the book. The latter is a common theme and source of examples. Since the Kingdom of Ruritania does not exist, think of it as a generic country with no real information about it beyond what is given.

Had I used a real country, it would be perfectly defensible for the student to bring in real information about that country. This may cloud the intended statistical lesson.

Also, using Ruritania allows me the ability to be creative in my story-telling.

I hope you enjoy the journey.

~ Ole J. Forsberg
December 2024
Knox College of Illinois

CHAPTER OVERVIEW

1: An Introduction to Ruruitania



Behind any good textbook is a good writer. Unfortunately, behind this textbook is me.

Seriously, behind every (hopefully) great textbook, there is not only a foundation of correct mathematics and statistics, but also a compelling narrative that brings the material to life. This book is no different — well, except perhaps in its choice of setting. Instead of abstract scenarios or fragmented real-world data, our journey through linear models will unfold within the borders of a single, coherent, and wonderfully fictional place: **The Kingdom of Ruritania** (Řurità Kràlovstvì).

Why Ruritania? Because learning statistics is more than just mastering formulas — it's about cultivating intuition, understanding context, and asking meaningful questions. Ruritania offers a consistent, conflict-free canvas where we can focus purely on the logic and application of statistical thinking. Here, data tells stories about its farms, universities, businesses, and citizens — stories we will learn to decode together.

This opening chapter serves two important roles. First, it invites you into Ruritania's world, offering a glimpse of its geography, economy, and society — the backdrop against which all our examples will unfold. Second, it provides a gentle foreshadowing of the kinds of questions we'll tackle: predicting crop yields, analyzing educational outcomes, modeling consumer behavior, and more. Consider this your orientation to both the kingdom and the craft of statistical modeling.

So, prepare to embark on an educational journey — one that will transform your mathematical knowledge into statistical insight and equip you to uncover the hidden relationships that shape our world, one Ruritanian dataset at a time. Welcome, and enjoy the adventure ahead.

1.1: The Background of Ruritania
1.2: The Geography
1.3: The Government
1.4: The Economy
1.5: US-Ruritanian Relations
1.6: Illustrations of Analyses
1.6.1: Illustration- The Missing Kraj
1.6.2: Illustration- Cows in Děčín
1.6.3: Illustration- Ruritanian Crops
1.6.4: Illustration- Wealth in Ruritania
1.6.5: Illustration- Elections in Ruritania
1.6.6: Illustration- Insurance in Ruritania
1.6.7: Illustration- Warmth for the King

1.7: Conclusion

This page titled 1: An Introduction to Ruruitania is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.1: The Background of Ruritania

The Kingdom of Ruritania (officially: Ŭrità Kràlovstvì) is a small kingdom (62 mi^2; 161 km^2) surrounded by Germany and the Czech Republic. Its inhabitants are Slavic-speaking Roman Catholics currently under the absolute monarchy of Rudolph II.

Under Rudolph II, ascension talks between Ruritania and the European Union have stalled. The two issues are the deficiency of democratic structures and an excess of controls on the press. Regardless of not being a member of the European Union, or of its monetary union (euro area), Rudolf pegged the Ruritanian Crown (*Koruna Ŭrità*) to the Euro at 2.00Kř to €1.00. This offers greater stability for the koruna in the world currency market. Three decades ago, economic reform allowed the koruna to be entirely convertible on the world market. It also completely eliminated the black market in Ruritania.



King Rudolph II, 2017
(Source: Wikipedia)

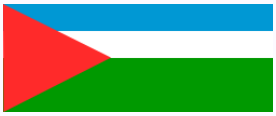
This page titled 1.1: The Background of Ruritania is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.2: The Geography

Ruritania is land-locked. It is located between southern Germany (Saxony) and western Czech Republic (Bohemia). The land is a beautiful combination of high mountains in the west (the Ruritanian Alps) and rolling farmland in the east (the Ruritanian Veldt).

This page titled 1.2: The Geography is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.3: The Government



Vlajka Ruritánie, 2023

Note that Ruritania is an autocratic kingdom, not a constitutional monarchy; the king rules by fiat. While the king is advised by a council of ministers, they are people that he selects. Furthermore, these ministers need not be citizens of Ruritania. The current council is composed of five Ruritanians and one Oregonian: the current Minister of Economics, who is Knox-educated and from Galesburg, Illinois. These councilors also perform the tasks of a "Committee King" when the king is unable to perform his duties as Chief of State and Head of Government.

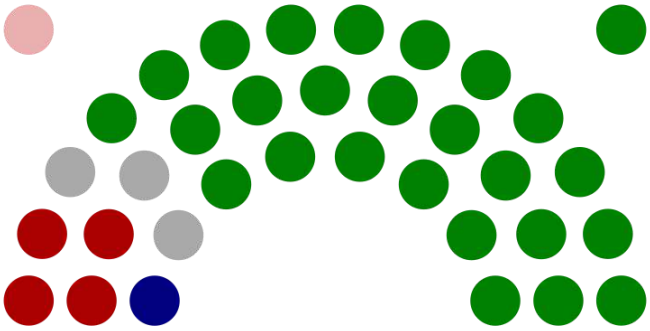
Strešlau, which serves as the official capital city, lies on the rail line between Dresden and Prague and has a population of 24,312. The royal capital, Sčwånstein, has a population of 7478 and lies on a spur (branch) line. According to the 2020 census, the total population of Ruritania is 43,670, with about 90% of the population living in its three main cities: Strešlau, Sčwånstein, and Děčín.

Ruritania is divided into seven states (státy). Each stát is named after its largest city. Thus, the seven státy are Děčín, Hora, Reka, Sčwånstein, Strešlau, Venkovský, and Zámek. Each státy is further divided into five counties (kraj). Officially, the name of each kraj is the stát name followed by a letter between A and E.

Národní Shromáždění

In addition to the Council of Ministers, there is a parliament (Národní Shromáždění; National Assembly) that can also advise the King — when he wishes. Members of parliament are elected by all adults when the King calls for elections. Since World War II, elections have happened every four years or so. The 2024 election saw two parties field candidates in every kraj. In that election, the Král a Země (King and Country) party, led by Vasilij Vasiljevič Kuzněcov, won 26 of the 35 seats in the parliament. The Republikánská Strana (Republican Party), led by Saša Ondřej Ivanović, won the second-most number of seats, four.

The following illustrates the current structure of the parliament.



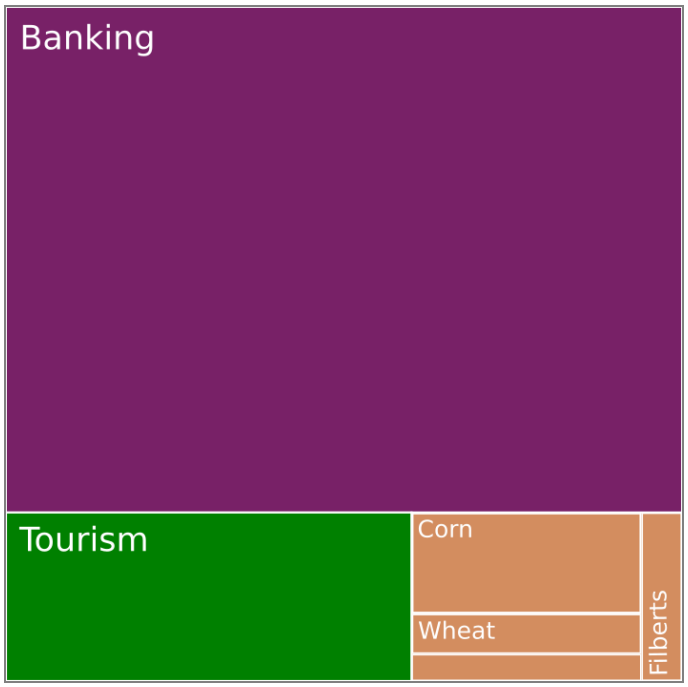
The green dots represent seats won by a Král a Země candidate; grey, independent; red, Republikánská Strana; blue, Liberálně-Demokratická Strana (Liberal-Democratic Party); and pink, Komunistický (Communist).

Note that the Republikánská Strana, Liberálně-Demokratická Strana, and Komunistický parties form the Republikán Bloc — the parties that seek to eliminate the monarchy. The Král a Země and independents support the monarchy, as such, they compose the Monarčista Bloc. There are six seats held by the Republikán Bloc and 29 held by the Monarčista Bloc.

This page titled 1.3: The Government is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.4: The Economy

While Ruritania's size limits its ability to diversify its economy, Rudolf has been an able administrator and businessman. As such, he was able to lift many Ruritanians out of poverty. When he took power in 1940, the GDP per capita (PPP) was approximately USD50, with a poverty rate of 99%. Today, it is approximately USD55,000 (the eighth highest in the world), with a poverty rate of just 50%.



The primary source of revenue for Ruritania is its banking industry — the source of 75% of its GDP (see Figure above). The remainder comes from tourism (15%) and agriculture (10%). Ruritania's tourism industry primarily bases itself on winter recreation in the west. Because of favorable visa requirements, low costs for hotels, and fantastic skiing in its alps, Ruritania is the destination of choice for vacationers from (in descending order of visitors) Czechia, Germany, Turkey, Oman, Morocco, United States, and Palau.

The primary crops are corn (55%), summer wheat (25%), and filbert nuts (15%), with soybeans and hops being secondary. What corn is not eaten is exported to neighbors Germany (75%) and Czechia (25%). Similar export patterns hold for the excess wheat. Filbert exports go to Czechia (40%), Germany (35%), and Switzerland (25%), where they are turned into delicious confections.

Imports to Ruritania come from Russia, Turkey, and Oman (petroleum), and Czechia and Germany (manufactured goods and foodstuffs).

Because of the strength of the monarchy, Ruritania is neither a production point nor a transshipping point for drugs. Illicit drug use is the lowest in Europe, with approximately 2% of the population using marijuana, and none using harder drugs.

This page titled 1.4: The Economy is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.5: US-Ruritanian Relations

The United States and Ruritania share full diplomatic recognition. However, the United States does not have an ambassador to Ruritania. US interests in Ruritania are handled by the US Ambassador to Poland, Mark Brzezinski (since February 22, 2022). This is not an unusual circumstance, as embassies are quite expensive to build and operate. Unfortunately, this reduces the amount of reliable information coming out of Ruritania.

Rudolf was a staunch ally of the United States during the Cold War. However, with the geographic position of his country in the world (entirely surrounded by Soviet satellite states East Germany and Czechoslovakia), he was able to offer the United States little more than intermittent vocal and moral support in the United Nations. For fear of losing sovereignty, Rudolf often kept quiet and followed the lead of the Soviet Union in all but domestic economic matters.

When the Soviet Union fell, Rudolf increased his support of the United States and its efforts to bring peace and prosperity to the world. As Rudolf often pointed out to various US presidents, Ruritania has never seen one of its sons die in battle. With his declining health and advancing age, Rudolf became much more vehement in his support of the United States, especially with respect to the Global War on Terror.



Mark Brzezinski, 2022
(Source: Wikipedia)

This page titled 1.5: US-Ruritanian Relations is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

SECTION OVERVIEW

1.6: Illustrations of Analyses

That concludes the background to Ruritania. The following examples show you what you will be able to do by the time you finish this book. The illustrations provide neither the data nor the code. All they provide are examples of how linear models are helpful to Ruritania... and to the rest of the world. Read through them and become excited about what this course will bring to you!

Analysis Illustrations

1.6.1: Illustration- The Missing Kraj

1.6.2: Illustration- Cows in Děčín

1.6.3: Illustration- Ruritanian Crops

1.6.4: Illustration- Wealth in Ruritania

1.6.5: Illustration- Elections in Ruritania

1.6.6: Illustration- Insurance in Ruritania

1.6.7: Illustration- Warmth for the King

This page titled 1.6: Illustrations of Analyses is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.6.1: Illustration- The Missing Kraj

Every 10 years of a king's reign, Ruritania holds a census. Their last was in 2020, marking King Rudolph's 80th year as monarch. After compiling all information, they discovered that information was missing for one *kraj* (state). This is unfortunate, because King Rudolph needs the information to evaluate his latest five-year plan and determine what he should do to make it better.

To fill in the missing information, we can regress all other variables on the GKP per capita, then predict the GKP per capita based on the known values for the missing kraj.

With this data and model, we just predict the GKP per capita in the kraj. The most-likely value is $2400, with a 95% prediction interval from $2100 to $2650.

From this information, His Majesty concludes that the plan helped the entire country, but did a better job with the rural areas than the urban. As a result of this analysis, he asks his ministers to generate a plan that does a better job of spreading the prosperity to more of the Kingdom.

✦•·····• ✦ •·····•✦

This use of regression has a very important use: Estimating values for missing data in a data set (imputation). Frequently, the amount of missing data will be significant with respect to the amount of complete data. In such a case, the researcher may use multiple imputation to create multiple data sets, estimate the parameters of interest on each, and report them and their standard errors.

This page titled 1.6.1: Illustration- The Missing Kraj is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.6.2: Illustration- Cows in Děčín

The voters of Děčín are being sent to the polls to vote on a city referendum that proposes to limit the number of cows that can be kept inside the city limits. The wording on this referendum is quite similar to ones proposed over the past several years.

Given the information from previous votes, the Director of the Independent Electoral Commission (NVK) estimates that the probability the referendum has of passing is 40%, with an estimated 48% of the voters supporting it.

✦•·····• ✦ •·····•✦

Note that this research question deals with the *probability* of winning, not just the best estimate of the vote in favor. This requires estimating the entire probability distribution of the dependent variable.

While there are some rather sophisticated methods, we will be able to answer a similar question using Monte Carlo simulation. Such simulation consists of drawing large samples from each of the parameter-estimate distributions, calculating a predicted outcome for each of those sets of estimates, and examining the distribution of these predictions.

Monte Carlo methods are a powerful set of techniques that allows you to estimate results when the assumptions of the mathematical model are not fully met by the data.

This page titled 1.6.2: Illustration- Cows in Děčín is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.6.3: Illustration- Ruritanian Crops

As usual, His Majesty Rudolph II would like some input from his Council of Ministers on his next five-year plan. Currently, the primary crop in Ruritania is corn. To help optimize the profits made by farmers, Rudolph wants to know if that crop should be changed to summer wheat or to soybeans.

To help him, let us model the relationship between farmer profit and crop in Ruritania. The dependent variable is the profit per acre, and the independent variable is the crop. Using linear models, we see that it is more profitable at this point to grow wheat. The average profit per acre is $845.75. This is about $150 greater than that of corn and about $300 greater than soybeans.

✦•·····• ✦ •·····•✦

Linear models can also be applied to cases where the independent variable is categorical, as here. This method is actually termed "**analysis of variance**" (ANOVA). The difference between ANOVA and regression is only conceptual, in that each level of the categorical variable is treated as a separate variable.

Using statistics to inform policy decisions is an important use. However, we professionals need to be aware of the limitations of our research --- and let our clients know them as well. Here, we only looked at the current average profit per acre for each crop. Future prices may fluctuate enough to make soybeans more valuable. Furthermore, shifting all Ruritanian crops to wheat puts the entire economy at risk of a drop in wheat prices. Diversification may be the better strategy, with some of the profit shared among all farmers.

This page titled 1.6.3: Illustration- Ruritanian Crops is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.6.4: Illustration- Wealth in Ruritania

The gross domestic product (GDP) per capita is one of many measures of average wealth in countries. If extant theory is correct, then the wealth in the country is directly affected by the level of corruption in the government — countries with higher levels of corruption should be poorer (on average) than those with low levels of corruption. Furthermore, if theory is correct, the level of democracy in a country should *also* influence the country's level of wealth — countries with higher levels of democracy should be wealthier than countries with lower levels of democracy.

His Majesty is curious to see how Ruritania fits in this model. If the actual GDP per capita is greater than what is expected from modeling the rest of the world, then Rudolph is doing a great job as king. Otherwise, he needs to improve the lot of his people.

And so, to help Rudolph, we predict that the GDP per capita for Ruritania, according to the model, is $26,795.64, with a 95% prediction interval from $5232 to $48,360. Since the actual GDP per capita is $55,000, King Rudolph is happy that he is better than average at guiding Ruritania forward towards prosperity.

✦•·····• ✦ •·····•✦

Frequently, we can use our models in novel ways. Usually, we would model the data and calculate predictions and confidence intervals. However, if we have confidence in our model, we can use it to determine which units are under- or over-performing expectations (the line of best fit). In this case, Ruritania's GDP per capita is significantly higher than what the model predicts.

This means either the model needs to take more covariates into consideration or that Ruritania is much more prosperous than one would expect... or both.

This page titled 1.6.4: Illustration- Wealth in Ruritania is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.6.5: Illustration- Elections in Ruritania

Even though it is an absolute monarchy, national elections are held in Ruritania to elect members of the Ruritanian parliament, the Národní Shromáždění (National Assembly). After the most recent election, Ruritanian exiles in Denmark claimed that the ballot boxes were stuffed. That is, the ballot boxes had votes for the government party in them even before voting began. Because guarantees of the "secret ballot" are built into the Ruritanian Constitution, the ballot boxes are opaque. As such, there is no direct evidence of stuffing.

Ordinary least squares regression is not well-suited for this type of data. There is inherent heteroskedasticity in the proposed model. However, we can use Binomial regression to test the election for evidence of ballot box stuffing.

As a result of our analysis, we were able to detect some evidence for stuffing ($\textbf{p-value} = 0.0441$). However, because the p-value is so high with respect to the usual $\alpha = 0.05$ value, do we really have the necessary support in the data to claim there was unfairness in the election?

To claim something so heinous, we should really contemplate the real meaning of the p-value and selecting an appropriate value for α.

✦•·····• ✦ •·····•✦

Here, we had a good understanding of the data-generating process. This allowed us to use that understanding to create a stronger model. Heteroskedasticity can be "adjusted for" in ordinary least squares. However, it is **ALWAYS** better to explicitly model the different aspects of your data instead of merely "adjusting for them."

This page titled 1.6.5: Illustration- Elections in Ruritania is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.6.6: Illustration- Insurance in Ruritania

The decision to buy life insurance is related to several variables, including age and income. We would like to explore this relationship in Ruritania.

Since the dependent variable is **dichotomous** (life insurance purchased *or* life insurance not purchased), we need a new type of regression to ensure that our predictions make sense. One option is called logistic regression. Using this regression, we find significant positive relationships between the person's age and the likelihood to buy life insurance, as well as between the person's income and the likelihood to buy life insurance.

Additionally, we predict that the Knox graduate on the Council of Ministers, a 65-year-old making $125,000 annually, has a 74% chance of having life insurance.

✦•·····• ✦ •·····•✦

In this example, we had to use a different type of regression because the dependent variable was dichotomous, could only take on two values. This type of regression is known collectively as logistic regression, even though the link function can be almost any that map the real line to the interval $(0,1)$. Such functions include the venerable logit function (inverse of the logistic function). It also includes the probit (used in a lot of medical studies) and the cauchit (used in some financial studies to allow for highly variable events).

It was this class of problems that forced Nelder and associates to formulate an over-arching framework for regression. He called it the "generalized linear model" (GLM). While he rues the name to this day, he created it to signify that this class of regression problems is actually just a generalization of the class that can be solved using ordinary least squares regression. In other words, OLS regression is a special case of GLM regression.

This page titled 1.6.6: Illustration- Insurance in Ruritania is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.6.7: Illustration- Warmth for the King

Finally, let us help the King be more beloved of his people — if that is possible. We took a poll and asked Ruritanians their "warmth of feeling" for Rudolph and his political agenda. In addition to this one variable, we also asked several demographic questions, allowing us to provide suggestions to the King. The demographic information includes gender, race, age, and number of years of education. The response variable has four ordered levels: Strongly Disagree, Disagree, Agree, and Strongly Agree.

With this information, we are able to let the King know that he is widely loved, but that women tend to agree with his policies more than do men. Furthermore, the better-educated also tend to support him more. The younger members of Ruritania also feel warmer towards him and his agenda. Finally, there was no relationship between race and support; all races seem to support him equally.

With this information from the poll, what can Rudolph do to help his people? Such is the question royals have asked for generations.

✦•·····• ✦ •·····•✦

Noting that the dependent variable is an ordinal variable, we could not use ordinary least squares regression. We had to use something called "ordinal regression." The concepts behind ordinal regression are quite similar to those behind other types of regression covered in this book. The mathematics are a bit more difficult, however.

Thankfully, statistical programs make using ordinal regression almost as easy as using other types. We just have to know how to get the data in the right form for the program *and* how to test the assumptions made by the technique.

In fact, this seems to be the lesson we need to learn throughout this course. The concepts are quite similar. The mathematics are different. And, to make usage easier, those who write statistical environments try to make the functions as similar as they can.

This page titled 1.6.7: Illustration- Warmth for the King is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

1.7: Conclusion



King Rudolph II, 1952
(Source: Wikipedia)

And that brings us to the end of this introductory chapter. In this chapter, you were introduced to the incredible Kingdom of Ruritania. Ruritania offers us a rich source of examples, which will be exploited throughout the text.

This chapter also offered several examples that foreshadow what you will learn in this course. While you may not be able to do — or understand the underlying theory for — any of them at this point, you will by the end of the course.

So, take a deep breath and turn the page to the book's first part: Ordinary Least Squares (OLS). In this part of the book, we start by asking what we mean by "summarizing the relationship with a line of best fit." From that point, we leverage that definition to inform our mathematics, thus allowing us to create formulas for estimating the population parameters.

After that chapter, we use the mathematics and elementary probability theory to create test statistics and confidence intervals for testing hypotheses of interest.

And after that... the sky is the limit! It is a great journey, and King Rudolph II thanks you for starting it.

This page titled 1.7: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

2: Part One- Least Squares

Welcome to the foundation of it all.

This first part of the book is dedicated to the Classical Linear Model, the workhorse of statistical analysis and the cornerstone from which countless other techniques are derived. You may know it simply as linear regression. Our journey begins here because understanding this model — truly understanding its mechanics, its logic, and its soul — is essential for everything that follows.

We will first build the model from the ground up, deriving the Ordinary Least Squares (OLS) solution with the familiar tools of scalar algebra. This approach provides an intuitive, tangible feel for how the model finds the "best-fitting" line by minimizing error. Then, we will ascend to a more powerful perspective: the matrix derivation. This framework is not merely a mathematical formality; it is the key that unlocks the analysis of complex, multi-dimensional data and paves the way for modern statistical computing. Once the mathematics are out of the way, we introduce probability distributions so we can explore confidence intervals and hypothesis testing.

Crucially, we will also put the model's famous assumptions under the microscope. We will explore the core requirements — linearity, independence, constant variance, and normality — asking not just **what** they are, but **how crucial** each one truly is to a valid analysis. You will learn which assumptions are deal-breakers, which are flexible, and how to diagnose and address problems when they arise.

By the end of this part, you will possess a deep and operational knowledge of OLS regression. You will be able to derive it, fit it using R, interpret its results, and critically assess its validity. The examples from Ruritania will guide our way, transforming abstract equations into meaningful stories about relationships in the world. Let us begin.

This page titled 2: Part One- Least Squares is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

3: Intro to Linear Regression



Regression is a set of methods that seek to learn the specific *relationship* between one or more influenced (dependent, response) variables and one or more influencing (independent, predictor) variables. There are many existing regression methods, each focusing on different ways of determining how best to quantify that relationship.

As is tradition, this chapter starts with our first definition of "best fit" and derives many results from that definition. This chapter is entirely mathematical in that probability distributions are not considered (until Chapter 5).



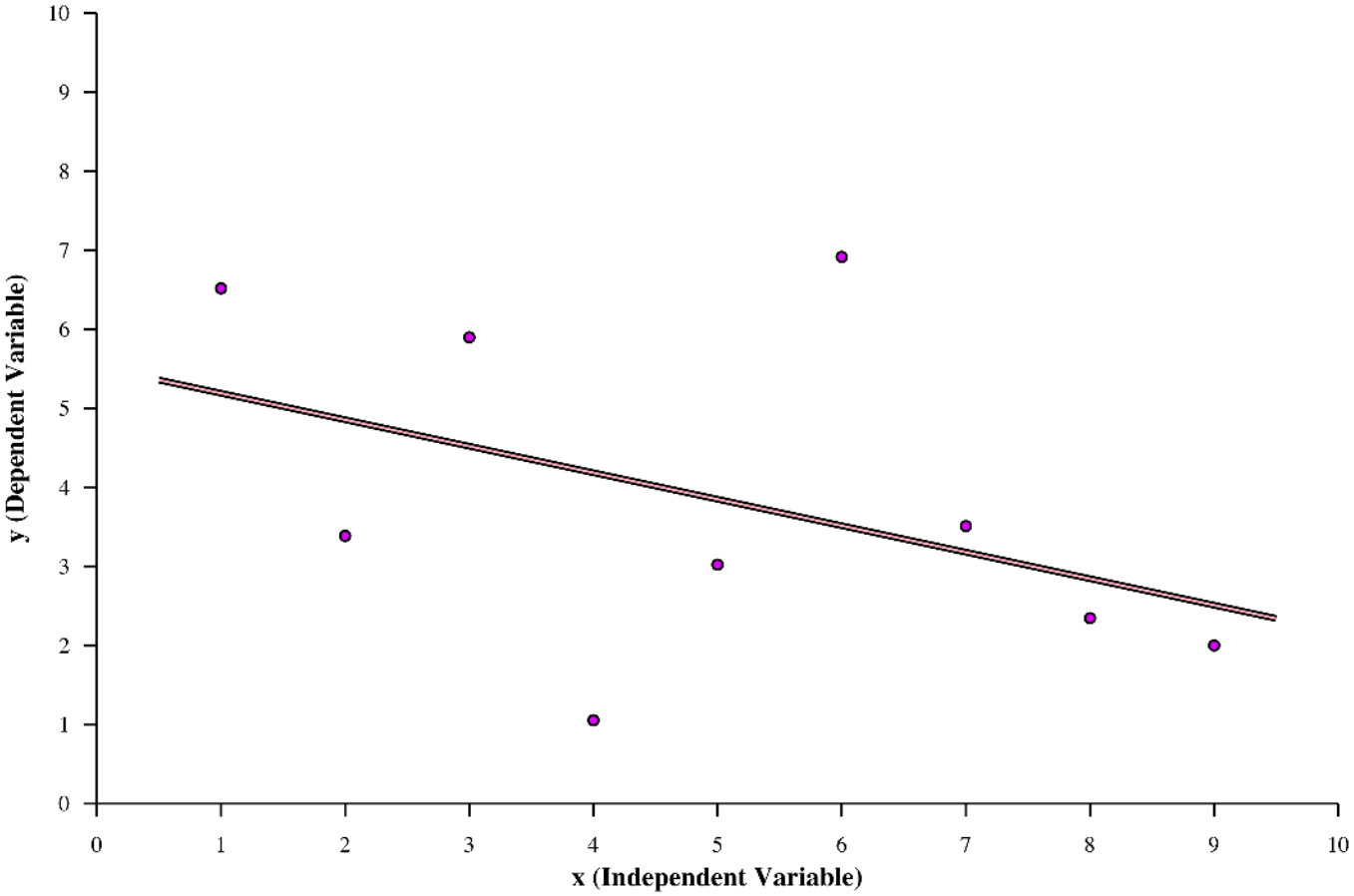


Figure **3.1**: Sample data and a line of best fit for that data. Note that the slope of the line is negative. This indicates that increasing values of x *tend to* correspond to lower values of y. Regression detects for such trends.

Let x and y be numeric variables. The linear relationship between x and y can be summarized by a line that "best" fits the observed data. That is, we can summarize the relationship between x and y using a linear equation:

$$y = \beta_0 + \beta_1 x + \varepsilon \tag{3.1}$$

Here, parameter β_1 represents the *slope* and parameter β_0 represents the **y-intercept** (the value of y on the line when $x = 0$). The slope is usually the only thing in the equation that is interesting; it is the effect of x on y. The ε represents the vertical distance between the observation and the population line of best fit. It contains all of the things that affect y that are not included in x.

We said that the line given in equation 3.1 "best" fits the observed data. What we mean by "best" determines where we go from here. In thinking about "best," it may help to see some sample data and the "line of best fit" for it (Figure **3.1**, above).

A good statistician will ask:

What makes *this* line the "best"?

A good statistician will answer:

It depends.

Note that there are *at least* three definitions of "best" that we can use:

1. Maximize the likelihood that the data were generated
2. Minimize the sum of the absolute value of the residuals
3. Minimize the sum of the square of the residuals

All three definitions are entirely legitimate — as are many other definitions. However, each leads to different estimation methods and estimators.

> **Note**
>
> While different models will usually give different estimates, the substantive conclusions will rarely differ significantly in a well-formed model.

Note that the result will be a line represented by

$$\hat{y} = b_0 + b_1 x \tag{3.2}$$

Using Latin characters indicates that these are based on your particular sample; they are sample estimates. Contrast this with using Greek characters to indicate population parameters. The "hat" on the y indicates that this is an estimate. All together, this is our *model equation*. It is the equation of the line of best fit based on the data you collected.

The first definition leads to "**maximum likelihood estimation**," which will be covered in Chapter 12. It is an excellent technique that can be generalized to many more settings than can ordinary least squares. Its greatest strength is that it makes use of the researcher's greater understanding of the data-generating process (Chapter 14 to Chapter 18). Its greatest weakness is the mathematics involved.

The second definition leads to a type of robust regression frequently termed "**median regression**." This method is helpful for times when there are outliers in the data that you cannot (or *should* not) remove. The drawback to this method is that estimating the two parameters (β_0 and β_1) does not provide a closed-form solution. In other words, it requires a repetitive sequence of steps and can only approximate those estimates. Furthermore, the approximation process is computationally intensive. Because of this, median regression was little used until recently. Because of this, the statistical theory behind it is not as well explored as other types. We will see this in Chapter 11: Quantile Regression.

The most popular definition of "best," and the one that starts our journey, is the final definition. It leads to an estimation method called "**ordinary least squares**" (OLS). It is rather straight-forward to minimize a sum of squared values using differential calculus. One strength is that an equation results from this process — a closed-form solution with no need for iteration. This means that the process returns mathematically exact values. The drawback is that it is limited in the types of processes that can be modeled.

We start exploring ordinary least squares immediately.

· · — · ✶ · — · ·

Chapter Sections

3.1: Scalar Representation
3.2: First Results
3.3: Our First Assumptions
3.4: PRE Measures
3.5: Conclusion
3.6: End-of-Chapter Materials

This page titled 3: Intro to Linear Regression is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

3.1: Scalar Representation

This section and the next show how our definition of "best" mathematically leads to specific results. That leading can be done by representing the regression problem in scalar or in matrix form. At one level, there is no difference in the two representations. At another level, one representation may make proofs — and understandings — easier and much more manifest. And so, let us begin with the scalar representation of the regression problem. From experience, it seems to make more sense than starting with the matrix representation (Chapter 4: Matrices and Linear Regression).

Learning Objectives

By the end of this section, you will be able to:

- Derive and calculate the ordinary least squares (OLS) estimators for the slope (β_1) and intercept (β_0) of a simple linear regression model using the scalar approach, and identify the mathematical condition required for their existence.
- Interpret the estimated coefficients b_1 and b_0 within the context of a problem, including special cases like a dichotomous independent variable, and distinguish between the model's use for estimating a conditional mean (expected value) versus predicting an individual future outcome.
- Explain the concept of extrapolation and its inherent dangers when using a regression model.

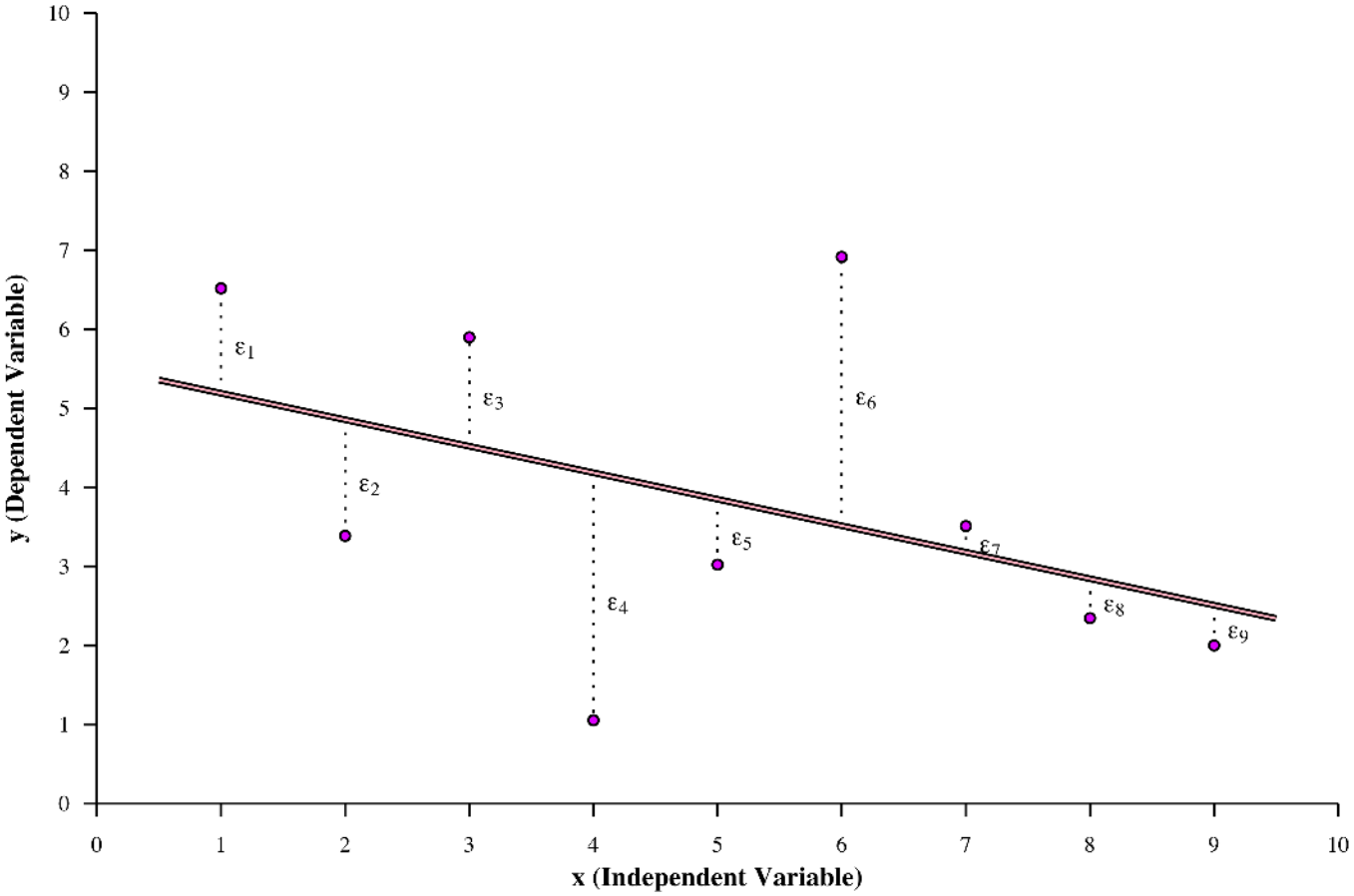


Figure **3.1.1**: Sample data and a line of best fit for that data. Also marked are the residuals, the difference between what was observed (dots) and what is predicted by the model (line). This particular line of best fit minimizes the sum of squared residuals.

Finding the OLS Line

Ordinary least squares estimation defines "best" as "having the lowest sum of squared errors." These errors (or residuals) are the *vertical* distance between the observed point and the corresponding point on the line (see Figure **3.1.1**). With this explanation of what we mean by "error" in this context, we can use our third definition of "best" to obtain the OLS estimators for β_0 and β_1.

Remember that to optimize (maximize or minimize) a function using calculus, one takes its derivative(s) with respect to the parameter(s) of interest, sets the resulting equations equal to 0, then solves the system of equations.

Note

Also, one should perform the second derivative test to determine the type of optimization point found: minimum, maximum, and saddle point (neither).

And so, the first step is to form the objective function that we want to minimize. Since we seek to minimize the sum of squared errors, that Q function is is the sum of squared errors:

$$Q = \sum_{i=1}^{n} \varepsilon_i^2 \tag{3.1.1}$$

$$= \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{3.1.2}$$

$$= \sum_{i=1}^{n} \left(y_i - (\beta_0 + \beta_1 x_i) \right)^2 \tag{3.1.3}$$

$$= \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_i \right)^2 \tag{3.1.4}$$

OLS Estimator of β_0

Now that we have the objective function, Q, we take its derivative with respect to each parameter, set it equal to 0, and solve for that parameter.

Let us start with β_0:

$$\frac{\partial}{\partial \beta_0} Q = \sum_{i=1}^{n} -2 (y_i - \beta_0 - \beta_1 x_i) \tag{3.1.5}$$

$$0 \overset{\text{set}}{=} \sum_{i=1}^{n} -2 (y_i - b_0 - b_1 x_i) \tag{3.1.6}$$

$$= \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} b_0 - b_1 \sum_{i=1}^{n} x_i \tag{3.1.7}$$

$$= n\bar{y} - nb_0 - nb_1 \bar{x} \tag{3.1.8}$$

This immediately leads to

$$b_0 = \bar{y} - b_1 \bar{x} \tag{3.1.9}$$

This is called the OLS estimator of β_0. Note that this formula needs $\bar{x}$ and $\bar{y}$. These are both easily calculated from the data. This formula also need b_1, which is the OLS estimate of β_1. Thus, we will need to determine the value of b_1 to use this formula.

OLS Estimator of β_1

And so, to obtain a formula for b_1, we take the derivative of Q with respect to the second parameter, β_1:

$$\frac{\partial}{\partial \beta_1} Q = \sum_{i=1}^{n} -2x_i \left(y_i - \beta_0 - \beta_1 x_i\right) \tag{3.1.10}$$

$$0 \overset{\text{set}}{=} \sum_{i=1}^{n} -2x_i \left(y_i - b_0 - b_1 x_i\right) \tag{3.1.11}$$

$$= \sum_{i=1}^{n} x_i y_i - b_0 \sum_{i=1}^{n} x_i - b_1 \sum_{i=1}^{n} x_i^2 \tag{3.1.12}$$

$$= \sum_{i=1}^{n} x_i y_i - n b_0 \bar{x} - b_1 \sum_{i=1}^{n} x_i^2 \tag{3.1.13}$$

Substituting our estimator b_0, we have

$$0 = \sum_{i=1}^{n} x_i y_i - \left(\bar{y} - b_1 \bar{x}\right) n \bar{x} - b_1 \sum_{i=1}^{n} x_i^2 \tag{3.1.14}$$

$$= \sum_{i=1}^{n} x_i y_i - n \bar{x}\, \bar{y} + b_1 n \bar{x}^2 - b_1 \sum_{i=1}^{n} x_i^2 \tag{3.1.15}$$

$$b_1 \left(\sum_{i=1}^{n} x_i^2 - n \bar{x}^2 \right) = \sum_{i=1}^{n} x_i y_i - n \bar{x}\, \bar{y} \tag{3.1.16}$$

Finally, we have

$$b_1 = \frac{\sum_{i=1}^{n} x_i y_i - n \bar{x}\, \bar{y}}{\sum_{i=1}^{n} x_i^2 - n \bar{x}^2} \tag{3.1.17}$$

The Two Estimators

Thus, the two OLS estimators of β_0 and β_1 are

$$\begin{cases} b_0 &= \bar{y} - b_1 \bar{x} \\ b_1 &= \dfrac{\sum_{i=1}^{n} x_i y_i - n \bar{x}\, \bar{y}}{\sum_{i=1}^{n} x_i^2 - n \bar{x}^2} \end{cases} \tag{3.1.18}$$

Note that this mathematical process **had but one requirement**:

$$\sum_{i=1}^{n} x_i^2 - n \bar{x}^2 \neq 0 \tag{3.1.19}$$

If that requirement is not met by the data, then the divisor of b_1 is zero in equation 3.1.17, which leads to dividing by zero, Armageddon, and a *really* bad hair day. However, note that

$$\sum_{i=1}^{n} x_i^2 - n \bar{x}^2 = (n-1)\, s_x^2 \tag{3.1.20}$$

As such, this requirement is met when the variance of x is non-zero. In other words, we require that the independent variable varies.

Note

For a mathematician, this is an important observation. For a statistician, it gives insight into how to "break" OLS: Measure all of the observations using the same value of the independent variable. In other words, to a statistician, the steps have meaning *beyond the mathematics*.

We will return to this requirement later, when we discuss multiple regression (more than one independent variable). When we do see this requirement again, it will be as a part of the requirement that the independent variables provide information about the dependent, i.e., that they are actually *independent* of each other.

Also note that some sources will give the formula for b_1 as:

$$b_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{3.1.21}$$

I leave it as an exercise to show these two:

? Exercise 3.1.1

$\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$ is equivalent to $\sum_{i=1}^{n} x_i y_i - n\bar{x}\,\bar{y}$

? Exercise 3.1.2

$\sum_{i=1}^{n}(x_i - \bar{x})^2$ is equivalent to $\sum_{i=1}^{n} x_i^2 - n\bar{x}^2$

Definition: Sum of Squared Deviations, S_{xx}

We will come across the denominator of equation 3.1.21 in many settings. Thus, to save ink, we will symbolize it as S_{xx} and define it as:

$$S_{xx} = \sum_{i=1}^{n}(x_i - \bar{x})^2 \tag{3.1.22}$$

Thus, our OLS line of best fit is the line defined by the set of points $(x, \hat{y})$, where

$$\hat{y} = b_0 + b_1 x \tag{3.1.23}$$

Note that $\hat{y}$ is the expected value of Y (dependent variable), given that value of x (independent variable). In other words,

$$\hat{y}_i = E[Y|x_i] \tag{3.1.24}$$

It is the **conditional mean** of Y given x_i; the expected value of Y, given this value of x_i; the mean of Y when the independent variable has value x_i.

Note

There is a difference between an "expected" and a "predicted" value. The **expected value** is the mean: If you were to rerun the universe a gazillion times, collected the same amount of data, and estimated $\hat{y}$ each time, the expected value is the average of all of those $\hat{y}$s. The **predicted value** is the value of an additional observation measured at the same value of x.

That the two are both on the line is happy happenstance — but happenstance nonetheless. The main difference, as you will discover, is in the intervals that surround that value. Intervals on predictions (**prediction intervals**) are wider than intervals on the mean/expected value (**confidence intervals**). This is because we are more uncertain about a future value than we are about an average.

And this is as far as we can go without making additional assumptions. As such, it marks a great place for a toy example.

A Toy Example

Example 3.1.1

Let us measure two variables on four subjects. Those two variables are x and y. For the first subject, the value of x is -2 and the value of y is 3. For the second subject the x and y values are 0 and 0. For the third subject, the values are 0 and 2. For the fourth subject, they are 2 and -1.

Given this information, let us calculate the ordinary least squares estimators of β_0 and β_1.

Solution:
First, the formulas for b_0 and b_1 require we calculate $\bar{x}$ and $\bar{y}$. To do this, we need the data. Here they are

Sample data

x	y
-2	3
0	0
0	2
2	-1

The means are 0 and 1, respectively. And, with that, we can use the formula for b_1 (Equation 3.1.18b):

$$b_1 = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{x}\,\bar{y}}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2} \tag{3.1.25}$$

$$= \frac{\Big(-2(3) + 0(0) + 0(2) + 2(-1)\Big) - 4(0)1}{\Big((-2)^2 + (0)^2 + (0)^2 + (2)^2\Big) - 4(0)^2} \tag{3.1.26}$$

$$= \frac{-8-0}{8-0} \tag{3.1.27}$$

$$= -1 \tag{3.1.28}$$

For the OLS estimator of the intercept, β_0, we have (Equation 3.1.18a):

$$b_0 = \bar{y} - b_1\bar{x} \tag{3.1.29}$$

$$= 1 - (-1)0 \tag{3.1.30}$$

$$= 1 \tag{3.1.31}$$

Thus, the OLS line of best fit is the line defined by the set of points $(x, \hat{y})$, where

$$\hat{y} = 1 - 1x \tag{3.1.32}$$

Figure **3.1.2** below shows the points *and* the OLS line of best fit.

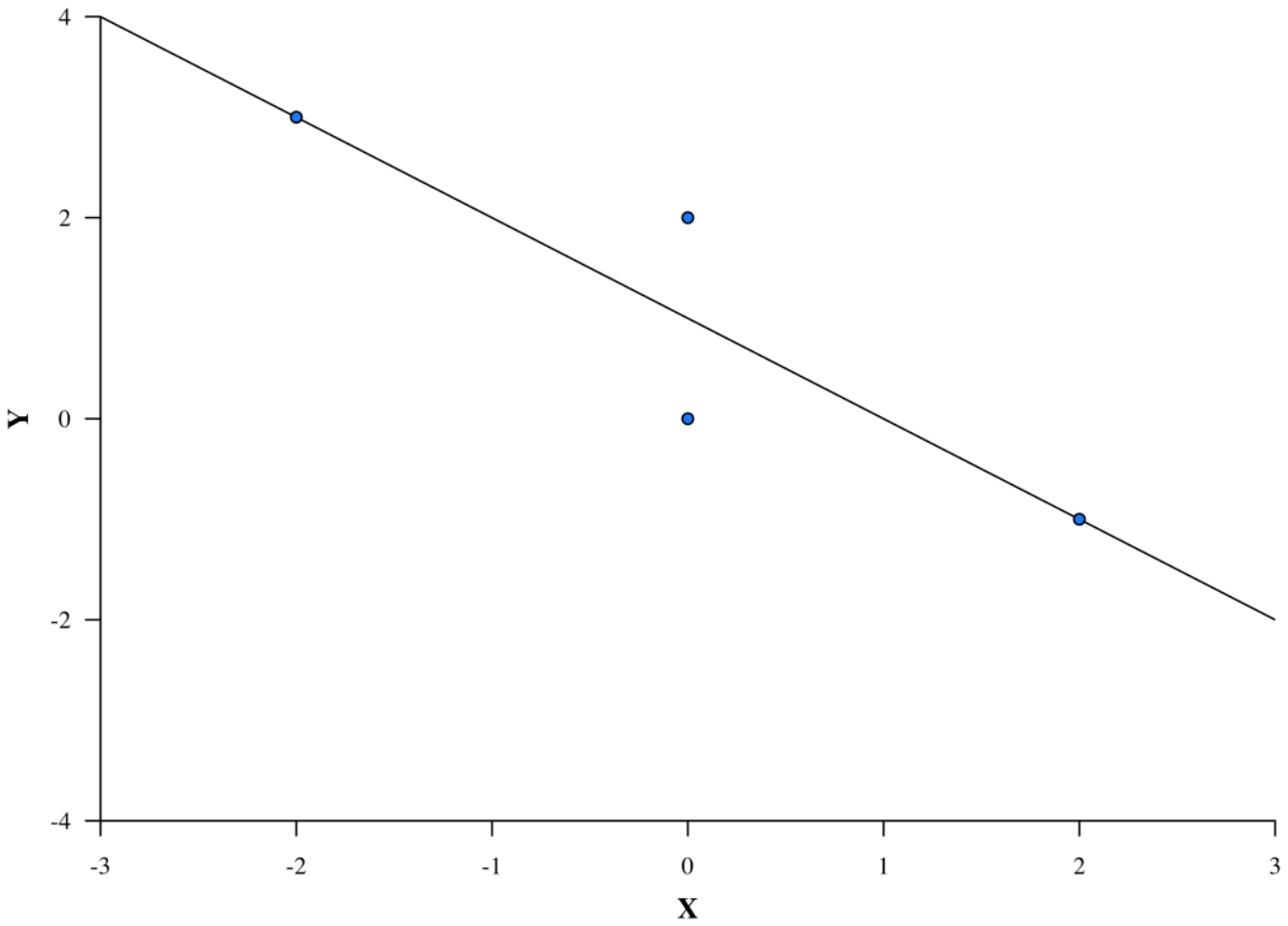


Figure **3.1.2**: Graphic of the data and the OLS line of best fit for the toy data of Example 3.1.1.

So, what does the equation *mean*? It means that the expected value of Y when $x = 0$ is 1, the y-intercept. It also means that for every one increase in the value of x, the expected value of Y increases by -1 (decreases by 1), which is the value of the slope.

To go beyond this rote interpretation, we need to know what the numbers represent, **we need context**. That information was lacking in this toy example.

Note

From a scientific standpoint, it is dangerous to interpret the y-intercept when $x = 0$ is outside the observed range of the data (x-values). Models are best when you are trying to understand the relationship within the observed ranges of the independent variable(s). This is **interpolation** — "inter" from "within."

Trying to use the model to understand relationships outside the observed values of the independent variables is called "**extrapolation**," where "extra" means "outside." Extrapolation is dangerous: all curves look linear at a small enough scale (remember Newton's Method from Calculus). Thus, fitting the data with a line may be a good approximation in one scale, it may not make sense at a wider range, where the non-linearity of the relationship may become more pronounced.

Another Toy Example: Handedness

Example 3.1.2

Let us measure two variables on four Ruritanian subjects. Those two variables are *handedness* and *time* to cut a sheet of paper. Sounds familiar? The difference here is that the independent variable is dichotomous.

Given the information in the table, let us calculate — and interpret — the ordinary least squares estimators of β_0 and β_1.

Solution:
Here are the data:

Handedness data

Handedness	Time
Left	3
Right	1
Right	2
Left	2

The first thing to do is change our independent variable into a numeric variable. When the variable is dichotomous (has only two possible values), this is easy. Set one value to 0 and the other to 1. So, without loss of generality, let us follow the alphabet and replace `Left` with **0** and `Right` with **1**. With this transformation, the values for handedness are now {0, 1, 1, 0} and we can use the same procedure as we used in Example **3.1.1**.

First, the formulas for b_0 and b_1 require we calculate $\bar{x}$ and $\bar{y}$. They are **0.5** and **2**, respectively. With that, we can use the formula for b_1 (Equation 3.1.18b):

$$b_1 = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{x}\,\bar{y}}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2} \tag{3.1.33}$$

$$= \frac{\Big(0(3) + 1(1) + 1(2) + 0(2)\Big) - 4(0.5)2}{\Big((0)^2 + (1)^2 + (1)^2 + (0)^2\Big) - 4(0.5)^2} \tag{3.1.34}$$

$$= \frac{3-4}{2-1} \tag{3.1.35}$$

$$= -1 \tag{3.1.36}$$

For the OLS estimator of the intercept, β_0, we have (Equation 3.1.18a):

$$b_0 = \bar{y} - b_1\bar{x} \tag{3.1.37}$$

$$= 2 - (-1)0.5 \tag{3.1.38}$$

$$= 2.5 \tag{3.1.39}$$

Thus, the OLS line of best fit is the line defined by the set of points $(x, \hat{y})$, where

$$\hat{y} = 2.5 - 1\ x \tag{3.1.40}$$

Now that we have some **context**, what does *this* equation mean?

Remember that an x-value of 0 indicates we are discussing left-handed people. Thus, the expected value of Y for the lefties is $1.5 + (1)0 = 1.5$. The expected value of Y for right-handed people is $1.5 + (1)1 = 2.5$.

Thus, the y-intercept is the predicted value for **base level** (lefties). The "slope" is the "effect of handedness" (moving from left- to right-handed) on that y-intercept.

You have seen an analysis of this type in your past introductory statistics course. This is just the **two-sample t-procedure** under the guise of linear models.

Note

Since we can compare the means of two group in the regression realm (Example **3.1.2**, above), can we compare the means of more than two groups? In other words, can we extend linear models to ANOVA? The answer is Yes! In fact, **ANOVA** is built on a base of linear models, as we will see in the future (4.4: Multicollinearity and Categorical Independent Variables).

Last Thought

Please do *not* forget that **all statistical procedures have requirements that have to be met**. So far, the only requirement is that there is variation in the independent variable. To draw stronger conclusions, perhaps calculate confidence intervals and test hypotheses, we will need to make stronger requirements. We will do this in the future. For now, let's just *require* that there is variation in the independent variable.

This page titled 3.1: Scalar Representation is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

3.2: First Results

So, the previous section laid out the mathematical results based on a rather minor requirement. As mathematicians, the next usual step is to explore to find out what more we can learn. Sometimes, the exploration leads to dead ends or trivial results. However, what keeps the mathematician excited for the material is that they will frequently learn something quite interesting.

In this section, let's see what consequences arise from our assumption.

Learning Objectives

By the end of this section, you will be able to:

1. Prove that the ordinary least squares regression line always passes through the point defined by the means of the independent and dependent variables (the center of gravity).
2. Demonstrate that the OLS slope estimator, b_1, is equivalent to the ratio of the sample covariance between x and y to the sample variance of x.
3. Express the slope estimator b_1 as the product of the correlation coefficient between x and y and the ratio of their sample standard deviations.
4. Explain the condition under which the slope estimator b_1 will equal zero.

✦•·····• ✦ •·····•✦

Now that we have formulas for our estimators, we have a few mathematical results. The first result is that the OLS line of best fit passes through the **center of gravity**.

Theorem 3.2.1: The Center of Gravity

The point $(\bar{x},\ \bar{y})$, the center of gravity, is on the OLS line of best fit.

Proof:
To see this just substitute $\bar{x}$ for x in the prediction equation and show that $\hat{y} = \bar{y}$.

From the formula for the OLS estimator of β_0 given in Section 3.1, we have

$$\hat{y} = b_0 + b_1 x \tag{3.2.1}$$

$$= b_0 + b_1 \bar{x} \tag{3.2.2}$$

$$= (\bar{y} - b_1 \bar{x}) + b_1 \bar{x} \tag{3.2.3}$$

$$= \bar{y} \tag{3.2.4}$$

Thus, we have shown that $\hat{y} = \bar{y}$ when $x = \bar{x}$. In other words, we showed that the OLS line of best fit passes through the center of gravity.

∎

Note

You should also be able to prove that *any* line passing through the center of gravity has the sum of the residuals being zero.

Example 3.2.1

Illustrate this result using the previous example. In other words, show that the point $(\bar{x},\ \bar{y}) = (0.5,\ 2)$ is on the line.

Solution:
In the previous section, we already showed that the line of best fit is $\hat{y} = 1.5 + x$. Substituting $\bar{x} = 0.5$ gives $\hat{y} = 1.5 + 0.5 = 2$. Note that 2 is also the value of $\bar{y}$.

Thus, we have *illustrated* Theorem 3.2.1.

———— ⋆·☆·⋆ ————

A second result is that the slope estimator b_1 is the ratio of the covariance between x and y to the variance of x.

Theorem 3.2.2

An equivalent formula is

$$b_1 = \frac{Cov[x,y]}{V[x]} = \frac{s_{xy}}{s_x^2} \tag{3.2.5}$$

Proof:
To see this we substitute the formulas for the covariance and variance into this equation and quickly simplify:

$$b_1 = \frac{s_{xy}}{s_x^2} \tag{3.2.6}$$

$$= \frac{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{3.2.7}$$

$$= \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{3.2.8}$$

■

Two additional results are

Theorem 3.2.3

The slope estimator can also be represented as

$$b_1 = r_{xy}\frac{s_y}{s_x} \tag{3.2.9}$$

In other words, the slope estimator is the correlation between the two variables times the ratio of their standard deviations.

Theorem 3.2.4

The slope estimator is zero if the y-values do not vary.

I leave these last two theorems as exercises.

This page titled 3.2: First Results is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

3.3: Our First Assumptions

This was fun. We were able to determine the correct formula for the line of best fit — given our particular definition of "best." Those equations lead to other equations. These are the *mathematical* results for our given sample.

Cool mule.

Learning Objectives

By the end of this section, you will be able to:

1. State the three core assumptions made about the residuals in a simple linear regression model and explain why they are necessary for deriving further statistical properties.
2. Prove that the OLS slope estimator b_1 is an unbiased estimator of the true population parameter β_1 using these assumptions.
3. Define the Mean Square Error (MSE) for simple linear regression and explain its role as an unbiased estimator of the residual variance σ^2.

✦•·····• ✦ •·····•✦

Until *this* point, we have only required variation in the independent variable. If we make three additional assumptions, we have additional results.

Note

This is how mathematical statistics progresses. Assumptions are made, then we play with the equations to learn about the consequences of those assumptions. Then, when we have exhausted our efforts, we make additional assumptions... *ad infinitum*

Recall that the data model is

$$y = \beta_0 + \beta_1 x + \varepsilon \tag{3.3.1}$$

With this, here are the three assumptions we will make, all about the residuals:

- The first assumption is that the residuals are realizations of a random variable (ε has a distribution).
- The second is that the expected value of the residuals is zero, $E[\varepsilon] = 0$ (the measurements are not systematically biased).
- The third is that the residuals are independent and have a finite and constant variance, $V[\varepsilon] = \sigma^2 < \infty$ (the residuals are homoskedastic).

The above simple assumptions lead to several additional interesting results. Some are proven here, some are left as exercises.

Theorem 3.3.1

The OLS estimator for β_1 is unbiased. That is,

$$E[b_1] = \beta_1 \tag{3.3.2}$$

Proof:

To prove this, we will start with the formula for b_1 and simplify until we obtain the results.

$$E[b_1] = E\left[\frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}\right] \tag{3.3.3}$$

$$= E\left[\frac{\sum_{i=1}^n (x_i - \bar{x}) y_i}{\sum_{i=1}^n (x_i - \bar{x})^2}\right] \tag{3.3.4}$$

$$= \frac{\sum_{i=1}^n (x_i - \bar{x}) E[y_i]}{\sum_{i=1}^n (x_i - \bar{x})^2} \tag{3.3.5}$$

$$= \frac{\sum_{i=1}^n (x_i - \bar{x}) (\beta_0 + x_i\beta_1 + \varepsilon)}{\sum_{i=1}^n (x_i - \bar{x})^2} \tag{3.3.6}$$

$$= \frac{\sum_{i=1}^n (x_i - \bar{x})\beta_0}{\sum_{i=1}^n (x_i - \bar{x})^2} + \frac{\sum_{i=1}^n (x_i - \bar{x}) x_i \beta_1}{\sum_{i=1}^n (x_i - \bar{x})^2} + \frac{\sum_{i=1}^n (x_i - \bar{x}) E[\varepsilon]}{\sum_{i=1}^n (x_i - \bar{x})^2} \tag{3.3.7}$$

$$= \frac{\beta_0 \sum_{i=1}^n (x_i - \bar{x})}{\sum_{i=1}^n (x_i - \bar{x})^2} + \frac{\beta_1 \sum_{i=1}^n (x_i - \bar{x}) x_i}{\sum_{i=1}^n (x_i - \bar{x})^2} + \frac{E[\varepsilon] \sum_{i=1}^n (x_i - \bar{x})}{\sum_{i=1}^n (x_i - \bar{x})^2} \tag{3.3.8}$$

$$= \beta_0 \frac{0}{\sum_{i=1}^n (x_i - \bar{x})^2} + \beta_1 \frac{\sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})}{\sum_{i=1}^n (x_i - \bar{x})^2} + 0 \frac{0}{\sum_{i=1}^n (x_i - \bar{x})^2} \tag{3.3.9}$$

$$= \beta_1 \tag{3.3.10}$$

Thus, the OLS estimator of β_1 is **unbiased**. This is a nice property. It means $E[b_1] = \beta_1$.

Note

Did you notice where we used these three results?

$$\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) = \sum_{i=1}^n (x_i - \bar{x}) y_i, \tag{3.3.11}$$

$$\sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x}) = \sum_{i=1}^n (x_i - \bar{x}) x_i, \text{ and} \tag{3.3.12}$$

$$\sum_{i=1}^n (x_i - \bar{x}) = 0 \tag{3.3.13}$$

All three are just simple algebra.

———— ⋆·☆·⋆ ————

Here are a few other results for you to prove:

Theorem 3.3.2

$E[b_0] = \beta_0$

Theorem 3.3.3

$V[b_1] = \frac{\sigma^2}{S_{xx}}$

Theorem 3.3.4

$V[b_0] = \sigma^2 \left(\frac{1}{n} + \frac{\bar{x}}{S_{xx}} \right)$

Theorem 3.3.5

$Cov[b_0, b_1] = -\sigma^2 \frac{\bar{x}}{S_{xx}}$

———— ⋆·☆·⋆ ————

Finally, let us define the mean square error (**MSE**) in the case of simple linear regression, SLR (one dependent and one independent variable).

Definition: Mean Square Error (MSE)

$$\text{MSE} = \frac{1}{n-2} \sum_{i=1}^{n} e_i^2$$

Theorem 3.3.6

$E[\text{MSE}] = \sigma^2$

In other words, the definition of the MSE provides an *unbiased* estimator of the variance of the residuals. This is why we define it in this manner.

Note

Be aware that the above definition of the MSE only holds in the case of simple linear regression (SLR); that is, it holds when there is just one dependent and one independent variable. It can be shown that the general definition of the MSE is

$$\text{MSE} = \frac{1}{n-p} \sum_{i=1}^{n} e_i^2$$

Here, p is the number of parameters being estimated, which is usually the number of independent variables plus the constant term.

This page titled 3.3: Our First Assumptions is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

3.4: PRE Measures

We have yet to introduce distribution assumptions on our data. Thus, we can still think about these results as mathematical. In a couple of chapters, we will make assumptions on the distribution of the residuals. However, until we do, we cannot speak about hypothesis testing and confidence intervals.

That will not stop us from continuing our exploration of our mathematical model (theory) and how it relates to the observed data (reality). This section introduces two measures of goodness-of-fit — measures of how well the model describes the reality. In Section 4.3, we will look at another measure and at what is happening in terms of the geometry of the problem.

Learning Objectives

By the end of this section, you will be able to:

1. Explain the purpose of Proportional Reduction in Error (PRE) measures as tools for assessing how well a linear model describes observed data compared to a simpler "null model."
2. Calculate and interpret the two primary PRE measures for simple linear regression: the coefficient of determination (R^2) and the adjusted coefficient of determination ($\bar{R}^2$).
3. Distinguish (R^2) from other potential PRE measures and explain why it is the most frequently reported statistic for assessing goodness-of-fit in linear regression.

✦•·····• ✦ •·····•✦

Now that we have reality (y_i) and our errors (e_i), we can create a measure of how well the model summarizes ("fits") the data. In fact, we will create two of them! Both are termed "**proportional reduction in error**" measures; that is, they both are measures of how well the model reduces the unexplained variation in the dependent variable. The first is the venerable R^2 ("R-squared") measure. The second is the $\bar{R}^2$ ("adjusted R-squared" or "R-bar-squared") measure.

Both measure how much the model reduces the variation in the dependent variable. They differ in how that variation is measured. The R^2 measure uses the sum-of-squares; the $\bar{R}^2$, the variance.

R^2 Measure

The formula for the R^2 measure is

$$R^2 = 1 - \frac{SSE}{TSS} \tag{3.4.1}$$

Here, SSE is the sum of the squared errors *using the model* (the error remaining), and TSS is the sum of squares *without* using the model (or with using the **null model**).

$$SSE = \sum (y_i - \hat{y})^2 \tag{3.4.2}$$
$$TSS = \sum (y_i - \bar{y})^2 \tag{3.4.3}$$

In this formula, $\hat{y}$ is the predicted value of y for each value of x_i in the data according to the model; $\bar{y}$ is the predicted value of y for each value of x_i in the data absent of the model (the mean of the dependent variable).

Thus:

- the SSE is a measure of how much variation remains in the model — the residual (unexplained) variation after applying the *model*.
- the TSS is a measure of the variation in the original data — the residual variation after applying the "*null model*."

Note

The "**null model**" always refers to the model with no independent variable. Thus, it is the model with only the y-intercept (here). The concept of the "null model" is extremely important in statistics, because it allows us to determine how much the model is a improvement over the "lack of" model.

Caution

The R^2 measure only tells us how much of the variation in the dependent variable is described by the model (as compared to how much was there originally). It tells us nothing beyond that.

For instance, an R^2 value of 0.04 tells us that the model explains only 4% of the variation in the dependent variable. There is a lot of variation left unexplained by the model. It does *not* mean that the model is "poor." An R^2 value of 0.98 tells us that the model explains 98% of the variation in the dependent variable. It does *not* tell us that the model is "good."

Note that the formula for R^2 is equivalent to

$$R^2 = 1 - \frac{\frac{1}{n-1} SSE}{\frac{1}{n-1} TSS} \tag{3.4.4}$$

Note that the numerator of the fraction is an estimator for the unexplained *variance* — but using the wrong number of degrees of freedom ($n-1$ instead of $n-p$). Thus, the R^2 measure is how much the model reduces the unexplained variance, when the variance is estimated using a *biased* estimator.

$\bar{R}^2$ Measure

Where R^2 measures how much the model reduces the unexplained variance, when that variance is estimated using a biased estimate of the total variance, $\bar{R}^2$ measures how much the model reduces the unexplained variance, when that variance is estimated using an *unbiased* estimate of the total variance. In other words, the adjusted R^2 measure uses the correct degrees of freedom to describe the proportional reduction in error.

$$\bar{R}^2 = 1 - \frac{\frac{1}{n-p} SSE}{\frac{1}{n-1} TSS} = 1 - \frac{(n-1) SSE}{(n-p) TSS} \tag{3.4.5}$$

Here, p is the number of parameters estimated in the model. For simple linear regression (one independent variable), $p = 2$, because we are calculating b_0 and b_1 from the data (to estimate β_0 and β_1). For the null model, $p = 1$, because we are only calculating $\bar{y}$ from the data (to estimate β_0).

Note

Usually, one reports the R^2 value and uses the $\bar{R}^2$ to help with model selection (select the model with the larger $\bar{R}^2$). This, I believe, needs to change. It is the *adjusted* R-squared value that better estimates the proportion reduction in error. This is because the adjusted R-squared measure uses unbiased estimators of the variances.

One strength of the R^2 measure is that it ranges from 0 to 1. The $\bar{R}^2$ measure *can* be less than 0. However, it is only less than 0 when the model describes very little of the variability in the dependent variable.

Other PREs

These are the two most frequently used PRE measures in linear regression. They are not the only ones, however. There is an entire class of PRE measures called "pseudo-R^2" measures. These are all genuine measures of how well the model helps reduce unexplained variation in the dependent variable. Their formulas tend to follow the structure of

$$\text{PRE} = 1 - \frac{\text{variation in the dependent variable with the model}}{\text{variation in the dependent variable without the model}}$$

Note

In the future, you will be introduced to "pseudo-R^2 measures" for several different modeling schemes. Because these follow the same scheme as above, they have the same interpretation. They are measures of how well the model reduces the uncertainty in the dependent variable.

The differences are in how that uncertainty is measured.

This page titled 3.4: PRE Measures is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

3.5: Conclusion

This chapter started with defining what we can mean by "best." Because we decided to define "best" as "minimizing the sum of squared residuals," we were able to obtain closed-form solutions for the estimates. From those equations, we were able to learn even more about our estimators — things not obvious from the definition.

That was the entire purpose of this chapter: to see that our results arise from applying mathematics to our selected definition of "best." Had we chosen a different meaning, we may have arrived at different results.

In the next chapter, we will change how we represent the data. Using matrices allows us to generalize what we did in this chapter to more than one independent variable. It also allows us to draw some interesting results. For instance, the estimators are only independent in simple linear regression if $\bar{x} = 0$. Showing this requires us to find the covariance between b_0 and b_1. This is much easier when working with matrices.

Realize This

One myth about linear regression is that the data can only be modelled using lines. Linear models are called "linear models" because the estimating equations are **linear *in the coefficients***. If you believe (or your substantive theory believes) that the relationship between x and y is quadratic, then you can define a new variable $x^* = x^2$ and use x^* in the model.

Voilà!

Quadratic modelling.

This page titled 3.5: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

3.6: End-of-Chapter Materials

Here are the expected materials to supplement the chapter.

R Functions

We used no R functions in this chapter, so there are none to cover here.

Exercises

I left many things as exercises for you. Here they are. You should be able to prove any and all of them using your prior knowledge of mathematics (matrices and calculus).

1. Perform the second derivative test on b_0 and b_1 to show that these estimators are really minima.
2. Show that $\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) = \sum_{i=1}^{n} x_i y_i - n\bar{x}\,\bar{y}$
3. Show that $\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{i=1}^{n} x_i^2 - n\bar{x}^2$
4. Prove $b_1 = r_{xy}\frac{s_y}{s_x}$
5. Prove that the slope estimator b_1 is zero if the y-values do not vary. Explain why this makes sense.
6. Using the scalar form, show that $Cov(b_0, b_1) = -\sigma^2 \frac{\bar{x}}{S_{xx}}$

This page titled 3.6: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

4: Matrices and Linear Regression



In the previous chapter, we were introduced to the classical linear model and estimating the parameters using ordinary linear regression. All of the work was done using a scalar representation of the data. When moving beyond simple linear regression, the estimators are more difficult to determine. The calculus remains almost as simple, but solving the system of equations becomes prohibitive.

The usual solution to solving complicated system of equations is to use a matrix representation of the problem. That is what this chapter does. Along the way, we discover more about linear models than we expected.

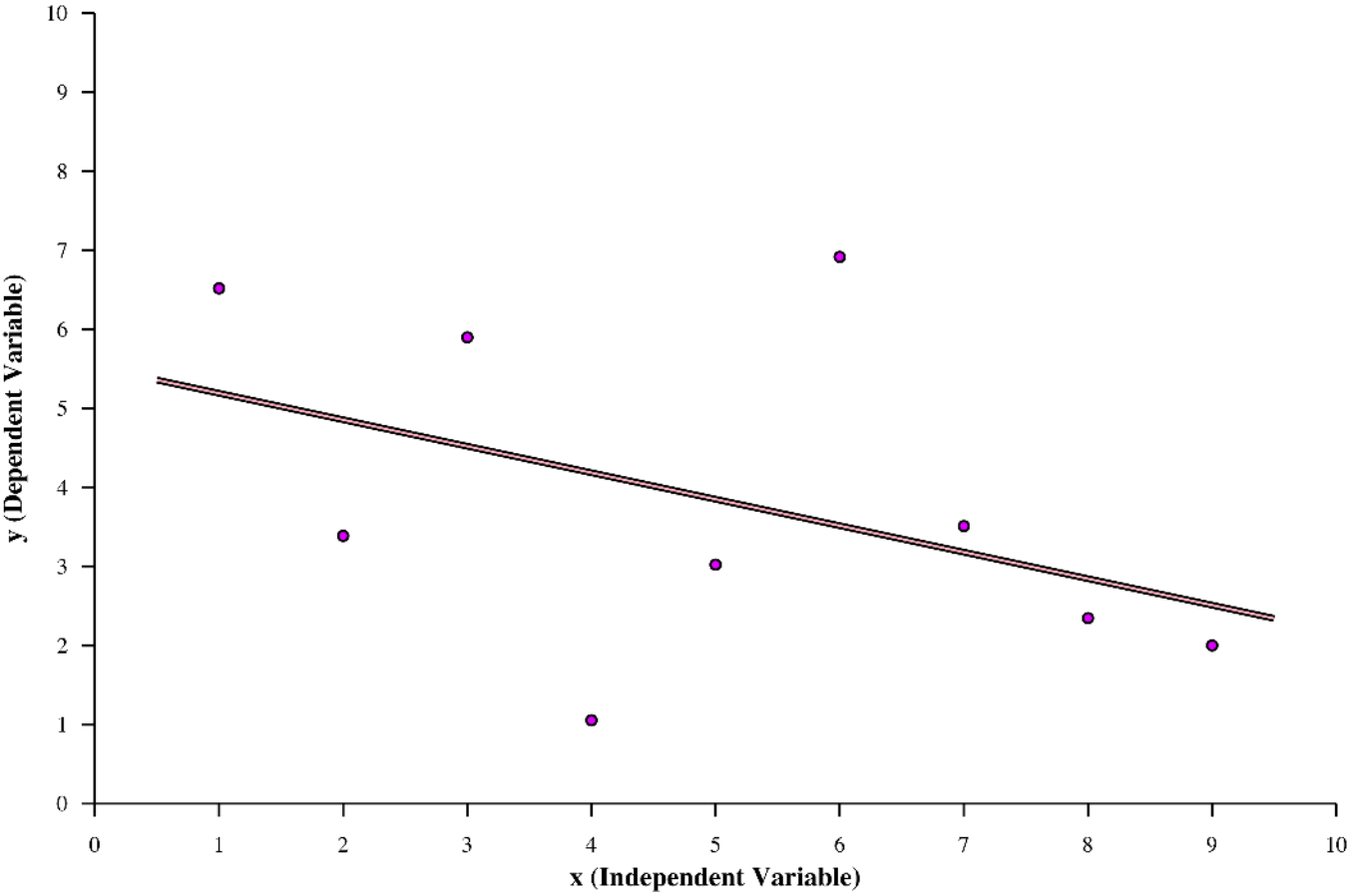


Figure **4.1**: Sample data and a line of best fit for that data. Note that the slope of the line is negative. This indicates that increasing values of x *tend to* correspond to lower values of y. Regression searches for such trends.

As in the previous chapter, let x and y be numeric variables. The linear relationship between x and y can be summarized by a line that "best" fits the observed data. That is, we can (and will) summarize the relationship between x and y using a linear equation:

$$y = \beta_0 + \beta_1 x \tag{4.1}$$

The above holds in the case of simple linear regression (SLR). So, what do we do when there are more independent variables? Here is that representation:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \cdots + \beta_k x_k \tag{4.2}$$

Here, β_0 is the y-intercept (still). And, β_j is the effect of variable x_j on the dependent variable, assuming all the other variables remain constant. By the way, his is called the *ceteris paribus* assumption. If the independent variables are independent of each other, then this requirement it met. However, there is frequently some correlation among the independent variables. Read on to see what to do about this.

Note

We say that the "line" given by equation 4.2 best fits the observed data. However, when dealing with two independent variables, it is not a line but a plane; with three, a space; with four, a hyperplane; etc. Clearly, meaningfully representing an entire four-variable model is quite difficult.

Chapter Sections

- 4.1: Matrix Representation
- 4.2: Predictions and the Hat Matrix
- 4.3: The Geometry of OLS
- 4.4: Multicollinearity and Categorical Independent Variables
- 4.5: Conclusion
- 4.6: End-of-Chapter Materials

This page titled 4: Matrices and Linear Regression is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

4.1: Matrix Representation

In the previous chapter, we were introduced to the classical linear model and estimating the parameters using ordinary linear regression. All of the work was done using a scalar representation of the data. When moving beyond simple linear regression, the estimators are more difficult to determine. The calculus remains almost as simple, but solving the system of equations becomes prohibitive.

The usual solution to solving complicated system of equations is to use a matrix representation of the problem. That is what this chapter does. Along the way, we discover more about linear models than we expected.

Learning Objectives

By the end of this section, you will be able to:

1. Express the multiple linear regression model, the data, and the OLS objective function in matrix notation, correctly defining all component matrices.
2. Derive the matrix formula for the OLS estimator $b = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$ and state the necessary condition for its existence ($X'X$ must be invertible, implying no perfect multicollinearity).
3. State the matrix form of the core assumptions about the residuals ($E[\mathbf{E}] = 0$, $V[\mathbf{E}] = \sigma^2 I$) and use them to prove that $\mathbf{b}$ is an unbiased estimator of $\mathbf{B}$.
4. Derive the variance-covariance matrix of the OLS estimator as $V[\mathbf{b}] = \sigma^2(\mathbf{X}'\mathbf{X})^{-1}$ and use it to show that the covariance between b_0 and b_1 is zero only when the mean of x is zero.

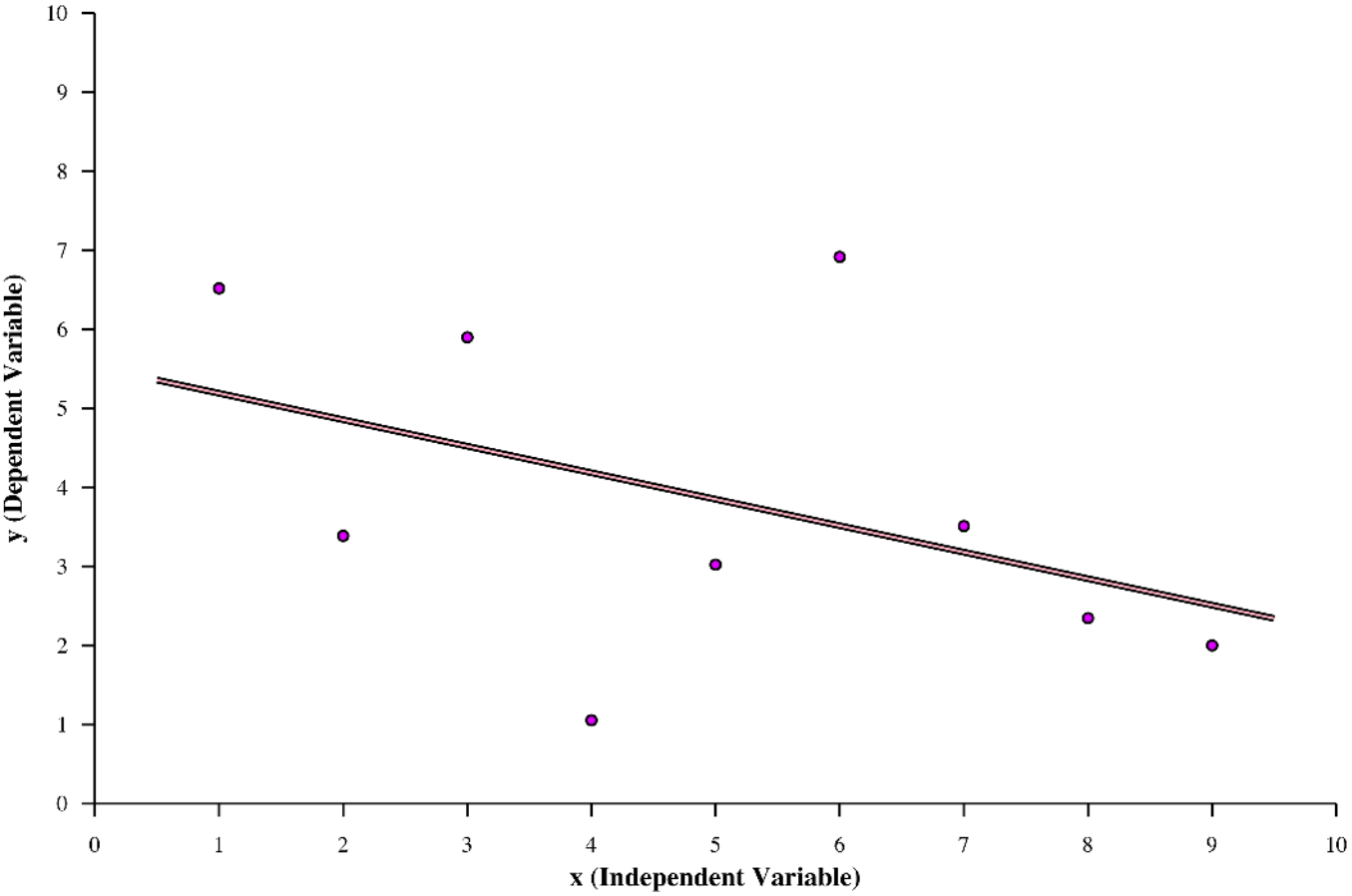


Figure **4.1.1**: Sample data and a line of best fit for that data. Note that the slope of the line is negative. This indicates that increasing values of x **tend to** correspond to lower values of y. Regression searches for such trends.

As in the previous chapter, let x and y be numeric variables. The linear relationship between x and y can be summarized by a line that "best" fits the observed data. That is, we can (and will) summarize the relationship between x and y using a linear equation:

$$y = \beta_0 + \beta_1 x \tag{4.1.1}$$

The above holds in the case of **simple linear regression** (SLR). So, what do we do when there are more independent variables? Here is *that* representation:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \cdots + \beta_k x_k \tag{4.1.2}$$

Here, β_0 is the y-intercept (still). And, β_i is the effect of variable x_i on the dependent variable, assuming all the other variables remain constant.

Note

This is called the *ceteris paribus* assumption. If the independent variables are independent of each other, then this requirement it met. However, there is frequently some correlation among the independent variables. Read on to see what to do about this.

We say that the "line" given by equation 4.1.2 best fits the observed data. However, when dealing with two independent variables, it is not a line but a plane; with three, a space; with four, a hyperplane; etc. Clearly, meaningfully representing an entire four-variable

model is quite difficult.

Matrix Representation

We learned a lot about our solution by exploring the scalar representation of the system of equations in the previous chapter. We may be able to gain some additional insights by exploring its matrix representation.

Caution

It may be helpful to re-familiarize yourself with Appendix M: The Appendix of Matrices at this point.

And so, let us begin with our matrix model.

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E} \tag{4.1.3}$$

In this model, $\mathbf{Y}$ represents the response variable; $\mathbf{X}$, the predictor variable(s) pre-pended with a column of 1s; $\mathbf{B}$, the coefficient vector; and $\mathbf{E}$, the residuals. The dimensions are $n \times 1$ for $\mathbf{Y}$, $n \times p$ for $\mathbf{X}$, $p \times 1$ for $\mathbf{B}$, and $n \times 1$ for $\mathbf{E}$.

Thus, in the case of simple linear regression, the matrices are

$$\mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} \quad \mathbf{X} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \vdots & \\ 1 & x_n \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} \quad \mathbf{E} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

In this formulation, n is the sample size and $p = 2$ is the number of parameters that need to be estimated. Usually, this is one more than the number of independent variables, $k = 1$.

Note that $\mathbf{X}$ is a matrix of "the predictor variable(s) prepended with a column of 1s."

Question:

What does this mean?

Also:

Why are those 1s needed?

Let our independent variable be the same as in the Toy Example in Section 3.1. Since $x = \{-2, 0, 0, 2\}$, the corresponding $\mathbf{X}$ matrix is

$$\mathbf{X} = \begin{bmatrix} 1 & -2 \\ 1 & 0 \\ 1 & 0 \\ 1 & 2 \end{bmatrix} \tag{4.1.4}$$

Again, we want to minimize the **sum of squared errors**. Again, we will create the objective function Q, take its derivative with respect to the parameter vector, $\mathbf{B}$, and solve:

$$Q = \mathbf{E}'\mathbf{E} \tag{4.1.5}$$

$$= (\mathbf{Y} - \mathbf{XB})'(\mathbf{Y} - \mathbf{XB}) \tag{4.1.6}$$

$$= \mathbf{Y}'\mathbf{Y} - \mathbf{B}'\mathbf{X}'\mathbf{Y} - \mathbf{Y}'\mathbf{XB} + \mathbf{B}'\mathbf{X}'\mathbf{XB} \tag{4.1.7}$$

Note that each of these terms is a 1×1 matrix, thus each is equal to its transpose. Using that on the third term and gathering the two like terms together gives our objective function.

$$Q = \mathbf{Y}'\mathbf{Y} - 2\mathbf{B}'\mathbf{X}'\mathbf{Y} + \mathbf{B}'\mathbf{X}'\mathbf{X}\mathbf{B} \tag{4.1.8}$$

Now, taking the derivative with respect to $\mathbf{B}$ and simplifying gives

$$\frac{d}{d\mathbf{B}}Q = -2\mathbf{X}'\mathbf{Y} + 2\mathbf{X}'\mathbf{X}\mathbf{B} \tag{4.1.9}$$

$$\mathbf{0} \overset{\text{set}}{=} -\mathbf{X}'\mathbf{Y} + \mathbf{X}'\mathbf{X}\mathbf{b} \tag{4.1.10}$$

$$\mathbf{X}'\mathbf{Y} = \mathbf{X}'\mathbf{X}\mathbf{b} \tag{4.1.11}$$

$$(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} = \mathbf{b} \tag{4.1.12}$$

This result is so important that I will repeat it here:

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{4.1.13}$$

Note the switch between $\mathbf{B}$ and $\mathbf{b}$. The former concerns the **population**. It is a population parameter that we are trying to estimate.

$$\mathbf{B} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{p-1} \end{bmatrix} \tag{4.1.14}$$

The latter concerns the sample. It is the **estimator** we are using to estimate the population parameter.

$$\mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{p-1} \end{bmatrix} \tag{4.1.15}$$

With this, the equation for our OLS regression line (plane, space, hyperplane, etc.) is

$$\hat{\mathbf{Y}} = \mathbf{X}\mathbf{b} \tag{4.1.16}$$

The Mathematical Requirement

In performing these calculations, we made one assumption: $(\mathbf{X}'\mathbf{X})^{-1}$ exists. If it does not exist, then the last step in the process cannot be done.

So, the first question to ask is:

> When does $(\mathbf{X}'\mathbf{X})^{-1}$ not exist?

Ans. It does not exist when $\det \mathbf{X}'\mathbf{X} = 0$.

> So, when does $\det \mathbf{X}'\mathbf{X} = 0$?

Ans. From linear algebra (and The Appendix of Matrices) we know that this determinant is zero when the $\mathbf{X}$ matrix is not of full (column) rank; that is, when $\text{rank}\mathbf{X} \neq p$. This happens when at least one column of the $\mathbf{X}$ matrix is a linear combination of the other columns. A statistician would say that there is **redundant information** in $\mathbf{X}$; one variable can be determined by the others.

When in the realm of multiple regression (more than one independent variable), this happens when one variable is a linear combination of the others. This condition is called "multicollinearity" or "super multicollinearity," depending on the source.

> Why didn't we have to worry about multicollinearity in the SLR case?

Ans. We did!

When in the realm of simple linear regression (SLR), multicollinearity happens when there is no variation in the x variable (i.e., it is a constant multiple of the columns of 1s).

New Assumption/s

Before we continue, as before, let us make the three assumptions about our residuals. These are just the same non-parametric assumptions we made back in Section 3.3, but in matrix form.

1. The first is that they are realizations of a random variable ($\mathbf{E}$ has a distribution).
2. The second is that the expected value of the residuals is zero, $E[\mathbf{E}] = \mathbf{0}$ (the measurements are not systematically biased).
3. The third is that the residuals are independent and have a finite constant variance, $V[\mathbf{E}] = \sigma^2\mathbf{I}$, with $\sigma^2 < \infty$.

In other words, let us make this assumption:

$$\mathbf{E} \sim N\left(\mathbf{0},\ \sigma^2\mathbf{I}\right) \tag{4.1.17}$$

Here, $\sigma^2 < \infty$.

Results and Consequences

Again, we have several results from this simple assumption.

Theorem 4.1.1

$E[\mathbf{Y}] = \mathbf{XB}$

Proof.
The proof of this proceeds from simple algebra.

$$E[\mathbf{Y}] = E[\mathbf{XB} + \mathbf{E}] \tag{4.1.18}$$

$$= E[\mathbf{XB}] + E[\mathbf{E}] \tag{4.1.19}$$

One **pervasive requirement** is that the values of $\mathbf{X}$ are not random variables. That is, the *researcher* selected those particular x values. Since this is true,

$$E[\mathbf{Y}] = \mathbf{X}\ E[\mathbf{B}] + E[\mathbf{E}] \tag{4.1.20}$$

Also, the values in the $\mathbf{B}$ matrix are population parameters. They, too, are not random variables. In fact, the only random variable on the right-hand side of that matrix equation is the zero-mean $\mathbf{E}$ matrix. Thus, we have

$$E[\mathbf{Y}] = \mathbf{XB} + E[\mathbf{E}] \tag{4.1.21}$$

$$= \mathbf{XB} \tag{4.1.22}$$

■

> **Note**
>
> The requirement that the independent variables are are not random allows us to easily calculate expected values, variances, and covariances. When designing experiments, this assumption is not problematic.
>
> When working with *observational* data, this becomes troublesome in terms of the mathematics. It also becomes troublesome in terms of the variances of $\mathbf{Y}$... and of the $\mathbf{b}$. The confidence intervals for the estimates are wider than estimated here. Also, if the variability in the $\mathbf{X}$ are *not* independent, even more difficulties arise.
>
> If any of this interests you, please look into **errors-in-variables models** (among other topics).

Similarly, it is quite easy to prove $V[\mathbf{Y} \mid \mathbf{XB}] = \sigma^2\mathbf{I}$. I leave that to you as an exercise.

Another result is that the two estimators are unbiased (i.e., their expected values equal the population parameter):

> **Theorem 4.1.2**
>
> The OLS estimator $\mathbf{b}$ is unbiased for $\mathbf{B}$.

Proof.
An estimator is unbiased for the parameter if its expected value equals the parameter. Thus, we need only show $E[\mathbf{b}] = \mathbf{B}$.

$$E[\mathbf{b}] = E[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}] \tag{4.1.23}$$

$$= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'E[\mathbf{Y}] \tag{4.1.24}$$

$$= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{XB} \tag{4.1.25}$$

$$= \mathbf{B} \tag{4.1.26}$$

■

A third result is that the two estimators are not necessarily independent.

> **Theorem 4.1.3**
>
> The OLS estimators b_0 and b_1 are not necessarily independent.

Proof.
To see this, we calculate the covariance matrix of $\mathbf{b}$ and look at the value corresponding to the covariance between b_0 and b_1.

$$V[\mathbf{b}] = V\left[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}\right] \tag{4.1.27}$$

$$= \left((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right) V[\mathbf{Y}] \left((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right)' \tag{4.1.28}$$

$$= \left((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right) \sigma^2\mathbf{I} \left((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right)' \tag{4.1.29}$$

$$= \left((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right) \sigma^2 \left((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right)' \tag{4.1.30}$$

$$= \sigma^2 \left((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right) \left((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right)' \tag{4.1.31}$$

$$= \sigma^2 (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' \mathbf{X} (\mathbf{X}'\mathbf{X})^{-1} \tag{4.1.32}$$

$$= \sigma^2 (\mathbf{X}'\mathbf{X})^{-1} \tag{4.1.33}$$

If *this* matrix is diagonal, then the estimators are independent.

To see that the two estimators are linearly correlated (i.e., not independent), we just need to calculate the matrix $(\mathbf{X}'\mathbf{X})^{-1}$.

Note that in general, this is difficult to do by hand. *However*, if we restrict ourselves to simple linear regression, that inverse is rather straight-forward to calculate. So, let's see the correlation in the SLR case:

$$\mathbf{X} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \tag{4.1.34}$$

With that, we have

$$\mathbf{X}'\mathbf{X} = \begin{bmatrix} n & n\bar{x} \\ n\bar{x} & \sum_{i=1}^n x_i^2 \end{bmatrix} \tag{4.1.35}$$

The determinant of this $\mathbf{X}'\mathbf{X}$ is

$$\det \mathbf{X}'\mathbf{X} = n\sum_{i=1}^n x_i^2 - n^2\bar{x}^2 = n\,S_{xx} \tag{4.1.36}$$

Thus, its inverse is

$$(\mathbf{X}'\mathbf{X})^{-1} = \frac{1}{n\,S_{xx}} \begin{bmatrix} \sum_{i=1}^n x_i^2 & -n\bar{x} \\ -n\bar{x} & n \end{bmatrix} \tag{4.1.37}$$

Finally, we have the covariance matrix:

$$V[\mathbf{b}[= \frac{\sigma^2}{n\,S_{xx}} \begin{bmatrix} \sum_{i=1}^n x_i^2 & -n\bar{x} \\ -n\bar{x} & n \end{bmatrix} \tag{4.1.38}$$

From this matrix, we see that the covariance between b_0 and b_1 is

$$Cov[b_0, b_1] = -n\bar{x}\,\frac{\sigma^2}{n\,S_{xx}} = -\sigma^2\frac{\bar{x}}{S_{xx}} \tag{4.1.39}$$

Thus, the OLS estimators are independent if and only if $\bar{x} = 0$.

■

Note

As an extension, note that the sign of the covariance is the *opposite* that of $\bar{x}$.

Finally, while this last results may seem just slightly interesting, it is the basis of the Working-Hotelling (1929) procedure, which we will see later in Section 5.4.

This last result also suggests why many disciplines tend to **center their x-values** (subtract off $\bar{x}$ from all of the x values) before doing regression. It ensures that the two estimators *are* independent.

Example 4.1.1

Let us revisit the Toy Example and show how to use the matrix representation to answer the same problem.

Solution.
The first step is to create the two matrices. The dependent variable matrix is

$$\mathbf{Y} = \begin{bmatrix} 3 \\ 0 \\ 2 \\ -1 \end{bmatrix} \tag{4.1.40}$$

The independent variable matrix, also called the "data matrix" and the "design matrix," is

$$\mathbf{X} = \begin{bmatrix} 1 & -2 \\ 1 & 0 \\ 1 & 0 \\ 1 & 2 \end{bmatrix} \tag{4.1.41}$$

Where did the column of 1s come from in $\mathbf{X}$? Remember that the matrix equation is

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E} \tag{4.1.42}$$

and that this is equivalent (in simple linear regression) to

$$y_i = \beta_0\, 1 + \beta_1\, x_i + \varepsilon_i \tag{4.1.43}$$

The 1s column in $\mathbf{X}$ is the multiplier of the β_0 in the $\mathbf{B}$ matrix. As long as you have a β_0 in your model, you need that column of 1s.

Now that we have the two matrices, we can calculate $\mathbf{b}$.

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{4.1.44}$$

$$= \left(\begin{bmatrix} 1 & -2 \\ 1 & 0 \\ 1 & 0 \\ 1 & 2 \end{bmatrix}' \begin{bmatrix} 1 & -2 \\ 1 & 0 \\ 1 & 0 \\ 1 & 2 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & -2 \\ 1 & 0 \\ 1 & 0 \\ 1 & 2 \end{bmatrix}' \begin{bmatrix} 3 \\ 0 \\ 2 \\ -1 \end{bmatrix} \tag{4.1.45}$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ -2 & 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} 1 & -2 \\ 1 & 0 \\ 1 & 0 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 4 & 0 \\ 0 & 8 \end{bmatrix} \tag{4.1.46}$$

$$\Rightarrow \qquad (\mathbf{X}'\mathbf{X})^{-1} = \frac{1}{32} \begin{bmatrix} 8 & 0 \\ 0 & 4 \end{bmatrix} \tag{4.1.47}$$

$$\mathbf{X}'\mathbf{Y} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -2 & 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} 3 \\ 0 \\ 2 \\ -1 \end{bmatrix} \tag{4.1.48}$$

$$= \begin{bmatrix} 4 \\ -8 \end{bmatrix} \tag{4.1.49}$$

Thus, we have

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{4.1.50}$$

$$= \frac{1}{32} \begin{bmatrix} 8 & 0 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} 4 \\ -8 \end{bmatrix} \tag{4.1.51}$$

$$\mathbf{b} = \frac{1}{32} \begin{bmatrix} 32 \\ -32 \end{bmatrix} \tag{4.1.52}$$

$$\text{And finally,} \tag{4.1.53}$$

$$\mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \tag{4.1.54}$$

From all of this, we have $b_0 = 1$ and $b_1 = -1$.

The conclusion is exactly the same, $\hat{y} = 1 - x$. The process is different. Here, this process is much easier for computers to perform, as they can do matrix multiplication (and inverting) with little problem. We have to spend a lot of extra effort to perform those operations. Here it is in R:

```
y = matrix( c(3,0,2,-1), ncol=1 )
x = matrix( c(1,1,1,1,-2,0,0,2), ncol=2 )

solve( t(x)%*%x ) %*% t(x) %*% y
```

Also, if we have more than one independent variable, we need to calculate the OLS estimator equations again; the ones from last chapter only hold for one independent variable. Using matrices, however, Formula 4.1.13 holds for **any number of** independent variables.

This page titled 4.1: Matrix Representation is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

4.2: Predictions and the Hat Matrix

In this section, we continue to explore the results of our decision to define "best" in the manner we did... and the additional assumptions we made in 4.1: Matrix Representation. This is how mathematics progresses. Assumptions are made and mathematicians explore the consequences of those assumptions. Here, we will just focus on the **hat matrix** and see what it can tell us about our choices.

Learning Objectives

By the end of this section, you will be able to:

1. Define the hat matrix $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ and explain its role in projecting the observed $\mathbf{Y}$ values onto their fitted values $\hat{\mathbf{Y}}$.
2. Prove that the hat matrix H is symmetric and idempotent, and use this property to demonstrate that the vectors of fitted values ($\hat{\mathbf{Y}}$) and residuals ($\mathbf{E}$) are orthogonal.
3. Derive the decomposition of the total sum of squares ($\mathbf{Y}'\mathbf{Y}$) into the sum of squared fitted values ($\hat{\mathbf{Y}}'\hat{\mathbf{Y}}$) and the sum of squared residuals ($\mathbf{E}'\mathbf{E}$) using matrix algebra.
4. Prove the important result that $\mathbf{X}'\mathbf{E} = \mathbf{0}$, and interpret its meaning: under OLS, the residuals are necessarily uncorrelated with each independent variable and have a mean of zero.

✦•·····• ✦ •·····•✦

Beyond modeling the relationship, one may also want to estimate or predict values of $\mathbf{Y}$ for a given value of $\mathbf{X}$. In matrix terms, this requires solving the equation $\hat{\mathbf{Y}} = \mathbf{Xb}$. But note the following:

$$\hat{\mathbf{Y}} = \mathbf{Xb} \tag{4.2.1}$$

$$= \mathbf{X}\ (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{4.2.2}$$

$$= \left(\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right)\ \mathbf{Y} \tag{4.2.3}$$

Note that the matrix $\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ "puts a hat" on the $\mathbf{Y}$ matrix. As such, it is called the "hat matrix," $\mathbf{H}$.

$$\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' \tag{4.2.4}$$

Thus, we have simple matrix equations for the estimators and the residuals:

$$\hat{\mathbf{Y}} = \mathbf{HY} \tag{4.2.5}$$

$$\mathbf{E} = \mathbf{Y} - \hat{\mathbf{Y}} = (\mathbf{I} - \mathbf{H})\,\mathbf{Y} \tag{4.2.6}$$

Why is this important? It shows that the predictions and the residuals are orthogonal.

Exploring the $\mathbf{H}$ Matrix

There are some surprising results from the above observations. Let us spend the rest of this section exploring the hat matrix, $\mathbf{H}$.

First, let us show that the OLS estimator is optimal in the sense that it is closest to reality of any estimator.

Theorem 4.2.1

The matrices $\mathbf{H}$ and $\mathbf{I}-\mathbf{H}$ are orthogonal.

Proof:
To show orthogonality, we need to show that the inner product is zero:

$$\mathbf{H}'(\mathbf{I}-\mathbf{H}) = \mathbf{H}(\mathbf{I}-\mathbf{H}) \tag{4.2.7}$$
$$= \mathbf{H}-\mathbf{H}\mathbf{H} \tag{4.2.8}$$
$$= \mathbf{H}-\mathbf{H} \tag{4.2.9}$$
$$= \mathbf{0} \tag{4.2.10}$$

In the proof, we used the fact that the hat matrix is **symmetric idempotent**. The next theorem proves this to be the case.

Theorem 4.2.2

The matrix $\mathbf{H}$ is symmetric idempotent.

Proof:
Let us start by showing $\mathbf{H}$ is symmetric.

$$\mathbf{H}' = \left(\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\right)' \tag{4.2.11}$$

Recall from 20.4: Other Terms and Operations that $(\mathbf{AB})' = \mathbf{B}'\mathbf{A}'$. Thus:

$$\left(\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\right)' = \mathbf{X}''\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\right)'\mathbf{X}' \tag{4.2.12}$$
$$= \mathbf{X}\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\right)'\mathbf{X}' \tag{4.2.13}$$

I leave it as an exercise to show that $\mathbf{X}'\mathbf{X}$ is symmetric, and so is its inverse. Thus

$$\left(\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\right)' = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' \tag{4.2.14}$$
$$= \mathbf{H} \tag{4.2.15}$$

Next, let us show that $\mathbf{H}$ is idempotent.

$$\mathbf{H}\mathbf{H} = \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\ \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}' \tag{4.2.16}$$
$$= \mathbf{X}\left[\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\ \mathbf{X}\right]\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}' \tag{4.2.17}$$
$$= \mathbf{X}\ \mathbf{I}\ \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}' \tag{4.2.18}$$
$$= \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}' \tag{4.2.19}$$
$$= \mathbf{H} \tag{4.2.20}$$

■

Since $\mathbf{H}$ is symmetric and idempotent, it is an projection matrix that projects $\mathbf{Y}$-space onto the restricted (smaller) $\hat{\mathbf{Y}}$-space. Because it is an orthogonal projection matrix, $\hat{\mathbf{Y}}$ is as close to $\mathbf{Y}$ as possible in its subspace. That is, the errors are minimized. Figure 4.2.1, below, illustrates this.

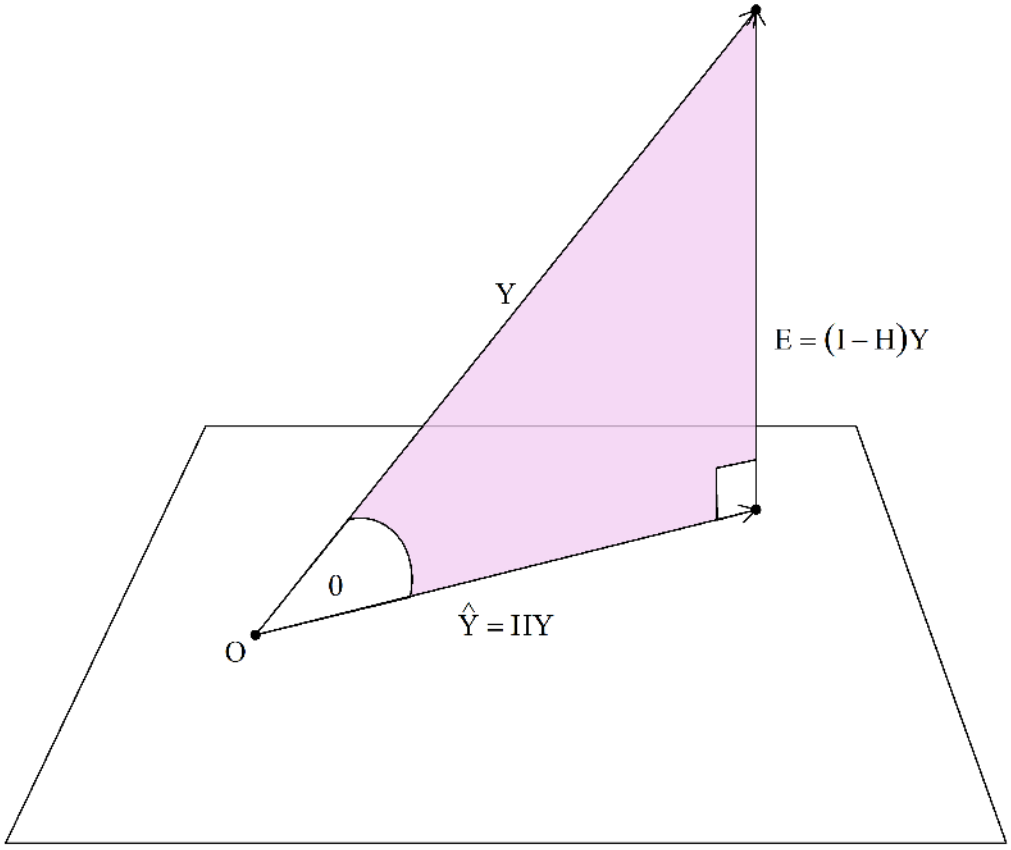


Figure 4.2.1: A schematic illustrating that $\hat{\mathbf{Y}}$ is as close to $\mathbf{Y}$ as possible, while remaining in its subspace (represented by the plane). In other words, the $\mathbf{Y}$ matrix exists in an n-dimensional space. The solution, $\hat{\mathbf{Y}}$, is in a p-dimensional space, with $n > p$. Under the assumptions of ordinary least squares, the distance between $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ (represented as the residuals, $\mathbf{E}$) is as small as possible if you define "distance" in terms of the Euclidean distance, L_2.

Theorem 4.2.3

The vectors $\hat{\mathbf{Y}}$ and $\mathbf{E}$ are orthogonal.

Proof:
I leave this as an exercise.

Since the predictions and residuals are orthogonal, we know the following is true by the **Pythagorean Theorem**:

$$\mathbf{Y}'\mathbf{Y} = \hat{\mathbf{Y}}'\hat{\mathbf{Y}} + \mathbf{E}'\mathbf{E} \tag{4.2.21}$$

Let us also prove this using matrices.

Theorem 4.2.4

$$\mathbf{Y}'\mathbf{Y} = \hat{\mathbf{Y}}'\hat{\mathbf{Y}} + \mathbf{E}'\mathbf{E} \tag{4.2.22}$$

Proof.
Let us prove this without resorting to the Pythagorean Theorem. We know $\mathbf{Y} = \hat{\mathbf{Y}} + \mathbf{E}$. Thus,

$$\mathbf{Y}'\mathbf{Y} = \left(\hat{\mathbf{Y}} + \mathbf{E}\right)' \left(\hat{\mathbf{Y}} + \mathbf{E}\right) \tag{4.2.23}$$

$$= \hat{\mathbf{Y}}'\hat{\mathbf{Y}} + \mathbf{E}'\mathbf{E} + \hat{\mathbf{Y}}'\mathbf{E} + \mathbf{E}'\hat{\mathbf{Y}} \tag{4.2.24}$$

$$= \hat{\mathbf{Y}}'\hat{\mathbf{Y}} + \mathbf{E}'\mathbf{E} + (\mathbf{H}\mathbf{Y})'(\mathbf{I}-\mathbf{H})\mathbf{Y} + ((\mathbf{I}-\mathbf{H})\mathbf{Y})'\mathbf{H}\mathbf{Y} \tag{4.2.25}$$

$$= \hat{\mathbf{Y}}'\hat{\mathbf{Y}} + \mathbf{E}'\mathbf{E} + \mathbf{Y}'\mathbf{H}'(\mathbf{I}-\mathbf{H})\mathbf{Y} + \mathbf{Y}'(\mathbf{I}-\mathbf{H})'\mathbf{H}\mathbf{Y} \tag{4.2.26}$$

Remember that $\mathbf{H}$ and $\mathbf{I}-\mathbf{H}$ are symmetric. That gives us

$$= \hat{\mathbf{Y}}'\hat{\mathbf{Y}} + \mathbf{E}'\mathbf{E} + \mathbf{Y}'\mathbf{H}(\mathbf{I}-\mathbf{H})\mathbf{Y} + \mathbf{Y}'(\mathbf{I}-\mathbf{H})\mathbf{H}\mathbf{Y} \tag{4.2.27}$$

Finally, since $\mathbf{H}(\mathbf{I}-\mathbf{H}) = (\mathbf{I}-\mathbf{H})\mathbf{H} = \mathbf{0}$, we have

$$\mathbf{Y}'\mathbf{Y} = \hat{\mathbf{Y}}'\hat{\mathbf{Y}} + \mathbf{E}'\mathbf{E} + \mathbf{Y}'\mathbf{0}\mathbf{Y} + \mathbf{Y}'\mathbf{0}\mathbf{Y} \tag{4.2.28}$$

$$\mathbf{Y}'\mathbf{Y} = \hat{\mathbf{Y}}'\hat{\mathbf{Y}} + \mathbf{E}'\mathbf{E} \tag{4.2.29}$$

■

This will come in handy when we add probability distributions to our mathematics, thus creating statistics. Independence is important in determining the distributions of test statistics.

By the way, we also can show that the residuals and predicted values are **uncorrelated** by showing their covariance is zero.

Theorem 4.2.5

$$Cov[\hat{\mathbf{Y}}, \mathbf{E}] = 0 \tag{4.2.30}$$

Proof.
I will only give the first step to this proof. The rest will be up to you to figure out.

$$Cov[\hat{\mathbf{Y}}, \mathbf{E}] = Cov[\mathbf{H}\mathbf{Y}, (\mathbf{I}-\mathbf{H})\mathbf{Y}] \tag{4.2.31}$$

So, where to go from this?

> By the way, this result should not be surprising given that the prediction and residual vectors are orthogonal.

Consequences

In this section, we started with the matrix equation $\mathbf{Y} = \mathbf{X}\mathbf{B} + \mathbf{E}$ and obtained the OLS estimator of $\mathbf{B}$. With that solution (and the requirement that $\mathbf{X}$ be full column rank), we have another result.

Theorem 4.2.6

$$\mathbf{X'E} = \mathbf{0} \tag{4.2.32}$$

Proof.
Again, I will just start you off with this proof. Completing it is up to you.

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E} \tag{4.2.33}$$

Where from here?

What does this theorem mean? Recall that $\mathbf{X'E}$ is a $p \times 1$ matrix. The first column of $\mathbf{X}$ is a column of 1s. Thus, the first element of $\mathbf{X'E}$ is just the sum of the residuals.

- That means the residuals must sum to 0 when we use the OLS estimator.

The other elements in the $\mathbf{X'E}$ matrix consist of the sum of the residuals times the values of each independent variable.

- This means that, under OLS, the residuals are necessarily linearly independent of each of the independent variables. It is a consequence of the mathematics used.

To see this in simple linear regression:

$$\mathbf{X'E} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_1 & x_2 & x_3 & \cdots & x_n \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ \vdots \\ e_n \end{bmatrix} \tag{4.2.34}$$

$$= \begin{bmatrix} \sum e_i \\ \sum x_i e_i \end{bmatrix} \tag{4.2.35}$$

This matrix is $\mathbf{0}$ only when all of its elements are also 0. Thus, we have $\sum e_i = 0$; the sum of the residuals in OLS is ***mathematically*** guaranteed to be zero.

We also have $\sum x_i e_i = 0$, which is equivalent to $\sum x_i e_i - n\bar{x}\,\bar{e}$ because $\bar{e} = 0$ and thus to $(n-1)Cov[x, e]$. This covariance is zero if x and e are linearly independent. This means that the residuals arising from OLS estimation are *linearly* uncorrelated with the predictor variables.

Be Aware!

Again, these are mathematical results from applying ordinary least squares. They are *guaranteed* simply because of the estimation method we selected. Had we chosen a different definition of "best fit," then this section may not hold.

Everything follows from our chosen definition of "best fit."

This page titled 4.2: Predictions and the Hat Matrix is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

4.3: The Geometry of OLS

The real purpose of this section is build on the previous section and to illustrate OLS in terms of geometry. This is because we have our model ($\hat{\mathbf{Y}}$) and reality ($\mathbf{Y}$) as in the figure below. The figure below gives a lot of insight into what is happening in OLS. Actually, this figure tells us that the OLS estimator is optimal in the sense of being closest to reality. All it takes is for us to understand geometry and linear algebra.

Learning Objectives

By the end of this section, you will be able to:

1. Visualize the fundamental limitation of statistical modeling: the dependent variable $\mathbf{Y}$ exists in a high-dimensional "reality" space, while any model restricts itself to a lower-dimensional subspace defined by the chosen independent variables.
2. Explain geometrically that the OLS fitted values $\hat{\mathbf{Y}}$ represent the orthogonal projection of the observed $\mathbf{Y}$ vector onto the model's subspace, making it the closest possible approximation within that restricted space.
3. Interpret the residual vector $\mathbf{E}$ as the orthogonal distance between the observed $\mathbf{Y}$ in "reality space" and its projection $\hat{\mathbf{Y}}$ in "model space," visually representing the information not captured by the chosen variables.

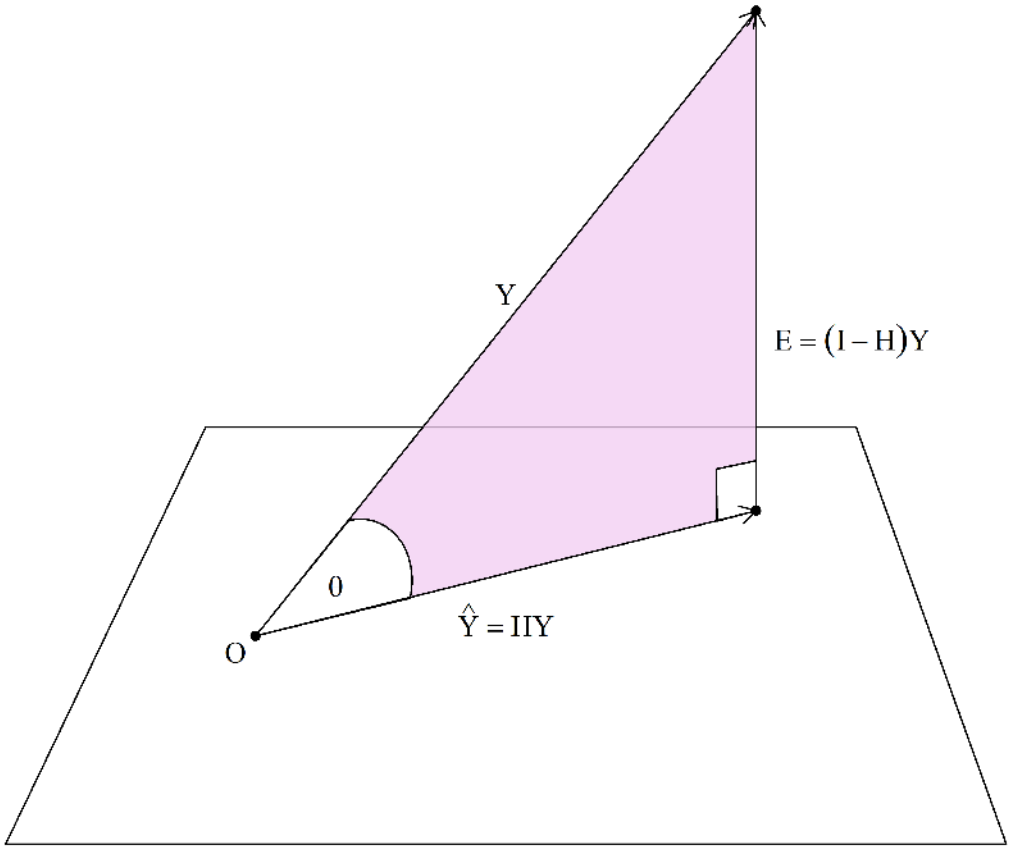


Figure **4.3.1**: Schematic of OLS showing that it is optimal in the sense that the OLS estimators are the closest of all points in the reduced space to the true value.

The Inherent Limitation of Statistical Models

In any real-world scenario, an outcome we wish to understand (the dependent variable, $\mathbf{Y}$) is influenced by a vast and complex web of factors. Think of something like house prices, a patient's blood pressure, or a Ruritania's stock performance. In a perfect world, to fully explain $\mathbf{Y}$, we would need to account for every single one of these influencing factors (the independent variables).

However, this is practically impossible. We can *never* identify or measure every possible influence. Therefore, when we build a statistical model — like an Ordinary Least Squares (OLS) regression — we are making a conscious choice to **simplify reality**. We select a limited set of what we believe are the most important variables to include in our model.

> **Note**
>
> In essence, we are restricting our view from the immense complexity of reality to a more manageable, but incomplete, subset of factors.
>
> This observation gave rise to George E. P. Box's observation that all models are wrong, but some are useful.

A Geometric Interpretation of Our Model

The figure above offers a way to visualize this. Imagine that in "Reality," the dependent variable $\mathbf{Y}$ is truly determined by three key independent variables. We can think of this full reality as existing in a three-dimensional space (3-space). Now, imagine that our statistical model, either by choice or by necessity, only uses only two of these three variables. Geometrically, this means our model is confined to a two-dimensional plane within the larger three-dimensional space.

- $\mathbf{Y}$ represents the actual, observed value, which exists as a point in the full 3D space, *above* (or below) our model's 2-space plane.
- $\hat{\mathbf{Y}}$ is the prediction made by our OLS model. By definition, $\hat{\mathbf{Y}}$ must lie *within* the restricted 2-space defined by the two variables we chose.

The Best Possible Approximation

So, how does the OLS estimator find $\hat{\mathbf{Y}}$? It operates under the constraint of our model. It cannot leave the 2-space (the plane). Its goal is to find the point *within that plane* that is *closest* to the real-world $\mathbf{Y}$. Here, this "closeness" is measured by the vertical distance between the point $\hat{\mathbf{Y}}$ and the plane — this is the **residual** (or error) of our model.

Therefore, the OLS estimate $\hat{\mathbf{Y}}$ is the optimal projection of reality onto our simplified model.

It is the **best** approximation we can get while remaining within the limitations we've imposed on ourselves. It acknowledges that our model is incomplete, but it ensures that, given its limitations, it is as accurate as it can possibly be. The gap between $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ is a visual representation of all the influences we failed to include in our model.

This page titled 4.3: The Geometry of OLS is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

4.4: Multicollinearity and Categorical Independent Variables

So far, our independent variable was either numeric (Toy Example) or dichotomous (Handedness Example). Let us now look at the interesting case of a discrete independent variable with three levels. This example raises questions about how to handle categorical independent variables in a manner that lends itself to using OLS and in a manner that eases interpretation.

Note that we are examining categorical *independent* variables in this chapter. Should we want to examine a categorical *dependent* variable, we will have to move away from OLS. To see why, you will need to continue reading the book.

Learning Objectives

By the end of this section, you will be able to:

1. Explain why including a categorical independent variable with k levels as k binary (0/1) indicator variables in the design matrix $\mathbf{X}$ leads to perfect multicollinearity, making $\mathbf{X'X}$ singular and the OLS estimator $(\mathbf{X'X})^{-1}\mathbf{X'Y}$ impossible to compute.
2. Demonstrate how to resolve this issue by dropping one indicator variable to create a design matrix of full rank, and identify the two standard approaches for interpreting the resulting coefficients: the means model and the effects model.
3. Interpret the estimated coefficients from both a means model (where coefficients directly represent the group means) and an effects model (where coefficients represent deviations from a chosen baseline group mean).

✦•·····• ✦ •·····•✦

Lengthy Section Example: Profiting Farmers

His Majesty Rudolph II would like some input on his next five-year plan. The primary crop in Ruritania is corn. To help optimize the profits made by farmers, Rudolph wants to know if that crop should be changed to summer wheat or to soybeans.

To help him, let us model the relationship between farmer profit and crop in Ruritania.

Table **4.4.1**: Raw data for the example

Crop	Profit per Acre
Wheat	722
Wheat	965
Wheat	940
Wheat	756
Corn	763
Corn	765
Corn	565
Corn	621
Soybean	566
Soybean	658
Soybean	540
Soybean	485

Solution:

Collecting the data is not as difficult as it may seem at first. All three crops are currently grown in Ruritania. All we had to do was obtain a list of all farms and their primary crop and randomly select records from that. The table above provides our raw data. Note that the response variable is numeric, and the predictor variable is categorical. How do we code that variable so that we can use the methods of this chapter (and this class)???

In the Toy Example, it was easy to change our dichotomous variable into a numeric variable by selecting one level as the **base level** and measuring the other level from there. In other words, one level was given the value **0** (absence) and the other was given the value 1 (presence). This is a nice thing about **dichotomous variables**. In this case, we have *three* levels in our independent variable. It does not seem to make sense to select one level to represent with **0** (absence), one to represent with **1** (presence), and one to represent with **2** (huh???? What does "double presence" mean???).

One method that always works is to create a series of dichotomous indicator variables from the one nominal variable. Thus, since there are three levels here, we would create three new dichotomous variables: `corn` , `soybeans` , and `wheat` . This change is presented in the table below. Note that each of the three dichotomous variables is now numeric. Each value indicates absence (0) or presence (1) of that trait (crop). With this change, we can use the methods of this chapter to calculate the values of the OLS estimators β_0, β_1, β_2, and β_3... or can we?

To see why I ended that paragraph in an evil and foreboding voice, let us work through this using matrices.

Table **4.4.2**: Transformed data for the example

Wheat	Corn	Soybean	Profit per Acre
1	0	0	722
1	0	0	965
1	0	0	940
1	0	0	756
0	1	0	763
0	1	0	765
0	1	0	565
0	1	0	621
0	0	1	566
0	0	1	658
0	0	1	540
0	0	1	485

Remember the formula to calculate the OLS estimators:

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{4.4.1}$$

Here, $\mathbf{Y}$ is

$$\mathbf{Y} = \begin{bmatrix} 722 \\ 965 \\ 940 \\ 756 \\ 763 \\ 765 \\ 565 \\ 621 \\ 566 \\ 658 \\ 540 \\ 485 \end{bmatrix} \tag{4.4.2}$$

The design matrix, $\mathbf{X}$ is

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \tag{4.4.3}$$

So far, so good!

> **Note**
>
> At this point, can you see why this matrix is termed the "*design*" matrix? From it, one can deduce the experimental design that gave rise to the data.

Next, let us calculate $\mathbf{X}'\mathbf{X}$:

$$\mathbf{X}'\mathbf{X} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}' \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \tag{4.4.4}$$

$$= \begin{bmatrix} 12 & 4 & 4 & 4 \\ 4 & 4 & 0 & 0 \\ 4 & 0 & 4 & 0 \\ 4 & 0 & 0 & 4 \end{bmatrix} \tag{4.4.5}$$

Fantastický!

That is a rather *interesting matrix*. From it, you can pick out the sample size ($n = 12$) and the sample sizes in each of the three levels ($n_i = 4$ in the diagonals). The next step is to calculate the inverse of this matrix.

At this point, it is *soooooooooo* much easier to use technology to perform this calculation. However, when you do, you will get a notification that the matrix is singular. This means two things:

1. Its inverse does not exist.
2. One column is a linear combination of the others.

Notice that the first column is the sum of the other three columns. Thus, there is redundant information. Thus, the columns are not linearly independent. To prove this last point, use the coefficient vector $a = \{1, -1, -1, -1\}$ in the definition of linearly dependent.

> **Note**
>
> From an information standpoint, if one column is a linear combination of the others, then that column is *redundant*. The model can be repeated without that information.
>
> This is one of the very few places in statistics where throwing away information helps. It is rather ironic that it helps solely in terms of the mathematics.

So, what do we do? We drop one of the redundant columns. The one we drop determines how we interpret the results. The next few sections looks at dropping the different columns in the $\mathbf{X}$ matrix.

Dropping the First Column

This is appropriate. In fact, it leads to this design matrix:

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.4.6}$$

This leads to

$$\mathbf{X}'\mathbf{X} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix} \tag{4.4.7}$$

$$(\mathbf{X}'\mathbf{X})^{-1} = \frac{1}{4}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.4.8}$$

$$\mathbf{X}'\mathbf{Y} = \begin{bmatrix} 3383 \\ 2714 \\ 2249 \end{bmatrix} \tag{4.4.9}$$

Finally, this leads to

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{4.4.10}$$

$$= \begin{bmatrix} 845.75 \\ 678.50 \\ 562.25 \end{bmatrix} \tag{4.4.11}$$

We can **interpret** these results as the average profit for wheat is 845.75; for corn, 678.50; and for soybeans, 562.25.

This is called the "**means model**" because the returned values are the means in each group.

Dropping the Second Column

This is also appropriate (the second column corresponds to the wheat design). When doing so, the design matrix is

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix} \tag{4.4.12}$$

Feel free to work through the calculation to obtain these estimates:

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{4.4.13}$$

$$= \begin{bmatrix} 845.75 \\ -167.25 \\ -283.50 \end{bmatrix} \tag{4.4.14}$$

The **interpretation** here is that the average profit for wheat (the base category/dropped column) is $\$845.75$. The effect of corn over wheat is -167.25, and the effect of soybeans over wheat is -283.50. In other words, the expected corn profit is -167.25 over the wheat profit, and the expected soybean profit is -283.50 over the wheat profit.

Note that we dropped the first data column. Thus, the first result is the expected value of the first variable and the other results are the effects of those levels *as compared to* the base category (wheat). Because the estimate are the effects of the other levels as compared to the selected base level, this is called an "**effects model**" for wheat.

Dropping the Third Column

As you can probably guess, this is appropriate, as well. When doing so, the design matrix is

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix} \tag{4.4.15}$$

Feel free to work through the calculation to obtain these estimates:

$$\mathbf{b} = \begin{bmatrix} 678.50 \\ 167.25 \\ -116.25 \end{bmatrix} \tag{4.4.16}$$

This **interpretation** is similar to the previous. The mean of the *base category* (corn) is 678.50 (first number). The *effect of wheat over corn* is 167.25. The effect of soybean over corn is -116.25.

Corn is the base category because the third column corresponds to the corn design. Note that this is the **effects model** for corn. The estimates are the effects in relation to the base category.

If you look at these three sets of results, you will see a lot of commonalities. The one chosen depends on what you are trying to say about the relationship between the crop and the profit. Here, is it also very easy to move between the means model and the effects model.

> **Note**
>
> We are only investigating expected values (averages) in this analysis. Should we also decide to include the uncertainties in our estimates (as we should), the two models are complementary. It is very difficult to move between the standard error in the means model and the standard error in the effects model. It is so much easier to have the computer perform that computation for you.

The Code

For the record, here is the code I used for fitting the means model:

```
X = matrix( c(1,0,0, 1,0,0, 1,0,0, 1,0,0,
    0,1,0, 0,1,0, 0,1,0, 0,1,0,
    0,0,1, 0,0,1, 0,0,1, 0,0,1 ),
    ncol=3, byrow=TRUE)
Y = matrix( c(722, 965, 940, 756, 763, 765,
    565, 621, 566, 658, 540, 485) )

solve(t(X)%*%X)
t(X)%*%Y

solve(t(X)%*%X) %*% t(X)%*%Y
```

Here is the code I used for the first effects model, in which I dropped the second column:

```
X = matrix( c(1,0,0, 1,0,0, 1,0,0, 1,0,0,
    1,1,0, 1,1,0, 1,1,0, 1,1,0,
    1,0,1, 1,0,1, 1,0,1, 1,0,1),
    ncol=3, byrow=TRUE)
Y = matrix( c(722, 965, 940, 756, 763, 765,
    565, 621, 566, 658, 540, 485) )

solve(t(X)%*%X)
t(X)%*%Y

solve(t(X)%*%X) %*% t(X)%*%Y
```

Note that the only change is in the line that defines the data matrix, $\mathbf{X}$.

Finally, here is the code I used when dropping the third column, the corn column.

```
X = matrix( c(1,1,0, 1,1,0, 1,1,0, 1,1,0,
    1,0,0, 1,0,0, 1,0,0, 1,0,0,
    1,0,1, 1,0,1, 1,0,1, 1,0,1),
    ncol=3, byrow=TRUE)
Y = matrix( c(722, 965, 940, 756, 763, 765,
    565, 621, 566, 658, 540, 485) )

solve(t(X)%*%X)
t(X)%*%Y

solve(t(X)%*%X) %*% t(X)%*%Y
```

Again, the only change is in the line that defines the data matrix, $\mathbf{X}$.

Caution

While I did provide the code at the matrix-multiplication level, I only do that to show you the options when fitting data with a categorical independent variable. This section encourages you to think about what you can get out of the model.

A real statistician figures out the easiest way to get the computer to provide the information. We will see how in the future.

This page titled 4.4: Multicollinearity and Categorical Independent Variables is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

4.5: Conclusion

This chapter has served as a crucial bridge between foundational concepts and advanced analytical tools. We began by deliberately carrying forward our core definition of "best" fitting models — minimizing the sum of squared residuals — from the simple, intuitive world of Simple Linear Regression (SLR) into the more complex, multidimensional terrain of Multiple Regression. This conscious continuity of purpose is vital; it ensures that our expansion in methodology remains anchored to a consistent philosophical principle, even as the mathematical landscape grows considerably richer.

The transition was more than a mere increase in variable count. In moving from SLR (with its single independent variable and two parameters, β_0 and β_1) to a model incorporating k predictors, we confronted a fundamental shift in scope. We are no longer fitting a line to a scatterplot but constructing a **hyperplane** in a $(k+1)$-dimensional space, estimating a parameter vector that encapsulates the partial effect of each predictor, holding the others constant (a.k.a. *ceteris paribus*). This leap required us to harness the power of matrix algebra — a tool that provides not just computational efficiency but also profound conceptual clarity. The elegant compactness of the matrix form, $\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$, belies the sophisticated operation it performs: it is the simultaneous, optimal solution for all parameters dictated by our definition of "best."

This formal mathematical derivation did more than provide formulas; it allowed us to explore the consequences of our decisions. It acted as a powerful lens, revealing properties of our estimators that were entirely non-obvious from the initial definition alone. For instance, through this process, we discovered the specific conditions under which our Ordinary Least Squares (OLS) estimators possess desirable properties. A revealing insight was that in the SLR framework, the estimators for the intercept and slope are statistically independent *only if* the sample mean of the independent variable is zero. This is a subtle, mechanical consequence of the mathematics that informs practical data handling (such as centering variables) and deepens our understanding of the model's inner structure.

And indeed, that revelation underscores the central theme of this chapter:

> Our results are the inevitable logical consequences of applying rigorous mathematics to our chosen definition of "best."

The entire edifice of OLS properties — the unbiasedness, the variances and covariances of the estimators encapsulated in $\sigma^2(\mathbf{X}'\mathbf{X})^{-1}$, and the famed Gauss-Markov Theorem guaranteeing Best Linear Unbiased Estimators (BLUE) — all spring deterministically from the axioms *we* established at the outset. This is a point of both strength and limitation. It is a strength because it provides a rock-solid, deductive foundation. It is a limitation because it forces us to acknowledge that our conclusions are conditional. Had we chosen a *different* criterion — minimizing absolute deviations, for example, or prioritizing robustness to outliers — the entire mathematical cascade would have diverged, leading us to **a different set of estimators with different properties**. The mathematics is a perfect servant to our initial philosophical choice.

Thus far, our journey has been predominantly deductive and mathematical. We have assumed our model form is correct and focused on the mechanics of estimation. However, this sets the stage for the essential next phase of our inquiry. In the forthcoming chapter, we will introduce the critical layer of statistical inference. While the mathematics of OLS tells us about the expected behavior of our estimators given the model assumptions, it is statistics that equips us to navigate the real world of uncertainty, sample variability, and imperfect information.

We will transition from asking "What are the values of our parameter estimates?" to asking "What can these estimates, derived from a finite and often far-too-small sample, tell us about the true, unknown parameters in the broader population?" We will develop tools for hypothesis testing — asking if a relationship we observe is statistically discernible from chance — and construct confidence intervals to express the precision of our estimates. In essence, if this chapter was about using mathematics to *derive* our estimators from a definition, the next chapter is about using statistics to *interrogate* these estimators when applied to the messy reality of empirical data. It is here that the assumptions we have quietly relied upon (linearity, independence, homoscedasticity, and normality of errors) will move to the foreground, as their validity becomes the bedrock upon which reliable inference is built.

This progression — from philosophical definition, through mathematical derivation, to statistical application — mirrors the scientific process itself. We first define what we seek, then use logic to deduce the tools to find it, and finally employ probabilistic reasoning to judge what we have found in light of uncertainty. Having now firmly established the mathematical underpinnings of multiple regression, we are fully prepared to embark on this final, indispensable stage of the journey.

This page titled 4.5: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

4.6: End-of-Chapter Materials

Here are the expected materials to supplement the chapter. Since there is R code in this chapter, I am including an explanation of several helpful R functions.

R Functions

In this chapter, we were introduced to a couple R functions that will be useful in the future. These are listed here.

Mathematics

- `%*%`
  This multiplies two matrices in \R. Thus, running the command `A%*%B` will return the matrix product **AB**. Be careful: `A*B` returns the Hadamard product, which is rarely what is needed.

- `c()`
  This combines the several scalar values into a single vector of values.

- `matrix()`
  This function creates a matrix from the given vector. The first slot belongs to the values in the matrix. After that is the number of rows (or columns) and whether you are entering the number by rows or by columns.

- `solve(m)`
  This calculates the usual inverse of the provided matrix **m**.

- `t(m)`
  This calculates the transpose of the provided matrix **m**.

Exercises

I left many things as exercises for you. Here they are. You should be able to prove any and all of them using your prior knowledge of mathematics (matrices and calculus).

1. Prove that $V[\mathbf{Y}|\mathbf{XB}] = \sigma^2\mathbf{I}$.

2. Let $\mathbf{A}$ be any full column rank matrix. Prove that $\mathbf{A}'\mathbf{A}$ is symmetric. Prove that its inverse is symmetric.

3. Prove that the vectors $\hat{\mathbf{Y}}$ and $\mathbf{E}$ are orthogonal.

This page titled 4.6: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

5: Improved! Now with Probabilities



This chapter extends the mathematics from last chapter by adding a probability distribution to the residuals. This results in the independent variable having a probability distribution. Please keep in mind that the independent variables are *not* random variables. The researcher specifically selects their values. Adhering to this paradigm allows us to more easily determine the resulting distributions. As such, this chapter continues this requirement. Should we not adhere to this requirement, the results of this chapter will technically be wrong, but will be close if the independent variable is statistically independent of the dependent variable.

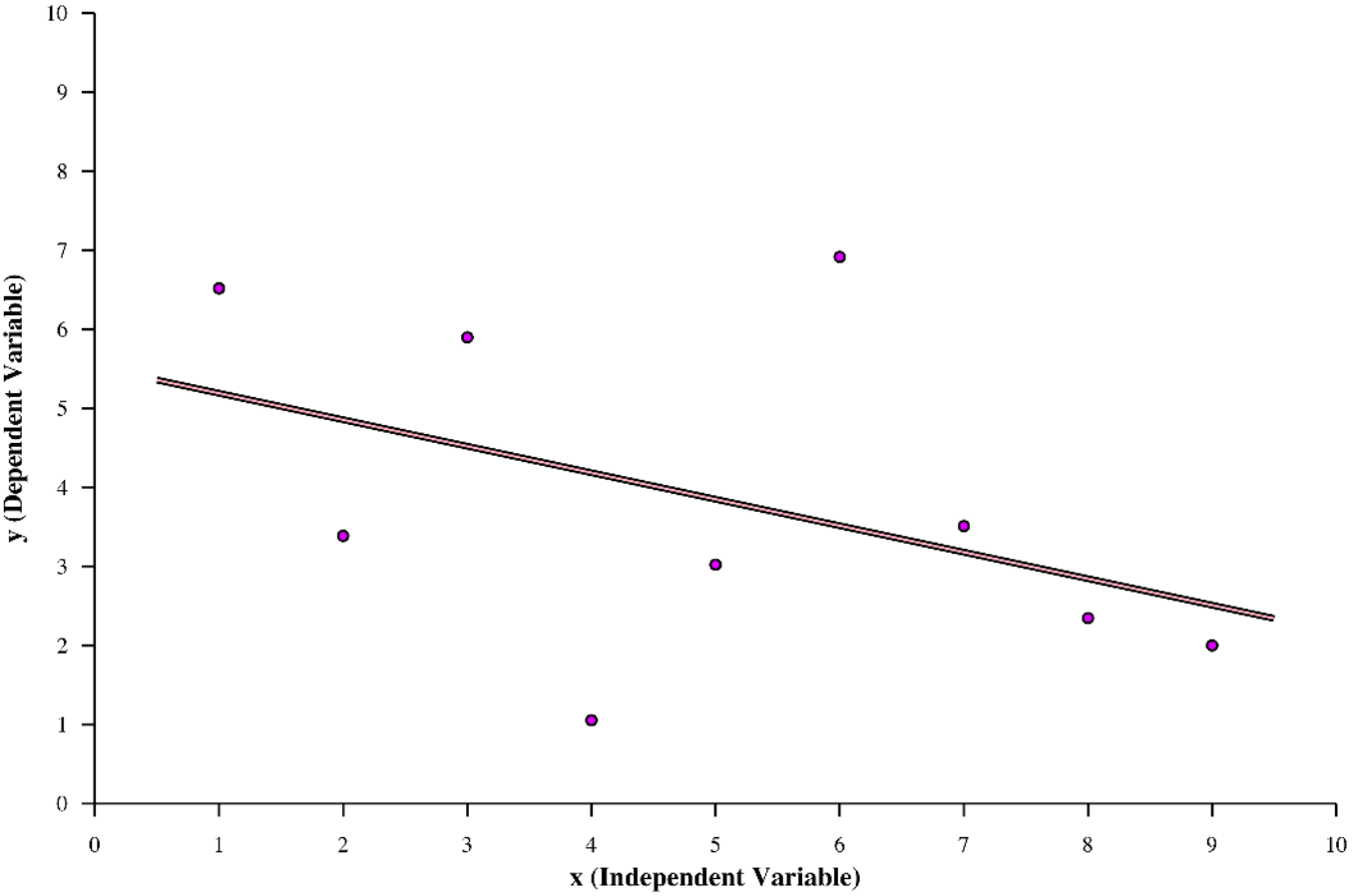


Figure 5.1: The basic scatter plot. This provides the observed values of the data as well as the line of best fit according to the Ordinary Least Squares method. The residuals are also indicated, with the values represented by dotted segments.

In the previous chapter, we explored the mathematical consequences of our choice of definition of "best." In this chapter, we will acknowledge that the residuals are observations from a random variable, specify its distribution, and see where that takes us.

And so, let us return to our scalar model for our data (Figure 5.1), above:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \tag{5.1}$$

and see what we can learn if we make the assumption that the ε_i are generated from a Normal distribution. Specifically, in conjunction with our previous assumptions, let us assume:

$$\varepsilon_i \overset{\text{iid}}{\sim} N\left(0,\ \sigma^2\right) \tag{5.2}$$

That single probability statement actually contains four parts:

1. The residuals follow a Normal distribution. No matter the values of the other variables, the residuals follow a Normal distribution.
2. The expected value of ε_i is a constant 0. No matter the values of the other variables, the expected value of the residual is 0 *at that point*.
3. The variance of the ε_i is a constant σ^2. No matter the values of the other variables, the variance of the residual is σ^2 at that point.
4. The abbreviation "iid" on top of the distribution sign means "independent and identically distributed." It indicates that the ε_i are independent of each other, and that the distribution of each is the same, $N\left(0,\ \sigma^2\right)$.

On the right-hand side (RHS) of Equation 5.1, the ε_i is the only random variable. The β_0 and β_1 are population parameters we are trying to estimate. The x_i are values selected by the experimenter, so they are also *not* random variables. This last sentence is rather

important for a lot of the calculations we make. The values of the independent variable are selected by the researcher, they are *not* realizations of a random variable.

Since the only thing on the right hand side that is a random variable is the ε_i, then it is rather easy to determine the distribution of $\boldsymbol{Y}$. And, with that, we are able to determine the distribution of almost all parameters we find important.

Important Idea

The RHS of Equation 5.1 is actually in two parts. The ε_i part is the source of the randomness, it is the "**stochastic**" part. The rest has no randomness associated with it. It is called the "**systematic**" part:

$$y_i = \underbrace{\beta_0 + \beta_1 x_i}_{systematic} + \underbrace{\varepsilon_i}_{stochastic} \tag{5.3}$$

Note that the systematic part contains the effects of the variables you are measuring and including in your model. The stochastic part contains **everything else**.

Let's continue exploring the consequences of making these additional assumptions.

· · — · ✶ · — · ·

Chapter Sections

5.1: Probability Distributions
5.2: Test Statistics and Hypothesis Testing
5.3: Confidence Intervals
5.4: The Working-Hotelling Bands
5.5: Conclusion
5.6: End-of-Chapter Materials

This page titled 5: Improved! Now with Probabilities is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

5.1: Probability Distributions

Alright, we've set the stage with our model. We made a few assumptions. Now, let's see what we can actually say about the numbers we calculate from our data — the estimators. This section is all about understanding where these numbers "live" and how they behave. Since our model's errors (residuals) are Normally distributed, it turns out most of our key estimators — like the slope b_1, the intercept b_0, and our predictions $\hat{Y}$ — also follow Normal distributions. We'll walk through why that's true and pin down exactly what their means and variances are.

We'll also look at the difference between predicting an average outcome versus a brand new single observation. Finally, we'll check out the distribution of our estimate for the error variance, the MSE. These results might seem a bit math-heavy now, but they're the secret sauce that lets us build confidence intervals and run hypothesis tests later on. So let's dive in and get to know our estimators a little better. King Rudolph would want us to.

Learning Objectives

By the end of this section, you will be able to:

1. State the distributional assumption that the errors are independent and normally distributed with mean zero and constant variance, and use this to derive the conditional distribution of $\mathbf{Y}$ given $\mathbf{x}$.
2. Derive the sampling distributions of the OLS estimators $\mathbf{b_1}$ and $\mathbf{b_0}$, showing that they are normally distributed with their respective means and variances.
3. Derive the sampling distribution of a fitted value $\hat{\mathbf{Y}}_i$ and distinguish it from the distribution of a new observation $\mathbf{Y}_{\text{new}}$, explaining why the prediction interval for a new observation is wider than the confidence interval for the mean.
4. State the distribution of the scaled mean square error, showing that $(n-p)\,\text{MSE}/\sigma^2$ follows a chi-square distribution with $n-p$ degrees of freedom.

✦•·····• ✦ •·····•✦

The Point Estimators

From this one assumption/requirement, and the math from the previous chapter, we have many consequences. This section explores the model and provides some results regarding the distribution of our estimators. The next sections build on this.

Theorem 5.1.1: Distribution of $Y|x$

The distribution of Y, conditional on the value of x, is

$$Y \mid x \overset{\text{ind}}{\sim} N\left(\beta_0 + \beta_1 x,\ \sigma^2\right) \tag{5.1.1}$$

Proof.

We are given $Y = \beta_0 + \beta_1 x + \varepsilon$, with the only random variable on the RHS being ε. Since ε follows a Normal distribution, so too does Y (see Theorem: The Sum of Normals).

Next, since the Normal distribution has two parameters, the mean and the variance, we need to determine those two values:

The **expected value** of $Y \mid x$ is

$$E[Y \mid x] = E[\beta_0 + \beta_1 x + \varepsilon \mid x] \tag{5.1.2}$$

$$= E[\beta_0 \mid x] + E[\beta_1 x \mid x] + E[\varepsilon \mid x] \tag{5.1.3}$$

$$= \beta_0 + \beta_1 x + 0 \tag{5.1.4}$$

$$= \beta_0 + \beta_1 x \tag{5.1.5}$$

The **variance** of Y, conditional on x, is

$$V[Y \mid x] = V[\beta_0 + \beta_1 x + \varepsilon \mid x] \tag{5.1.6}$$

$$= V[\beta_0 \mid x] + V[\beta_1 x \mid x] + V[\varepsilon \mid x] \tag{5.1.7}$$

$$= 0 + 0 + V[\varepsilon] \tag{5.1.8}$$

$$= \sigma^2 \tag{5.1.9}$$

Thus, putting this together, we have our final result

$$Y \mid x \overset{\text{ind}}{\sim} N\left(\beta_0 + \beta_1 x,\ \sigma^2\right) \tag{5.1.10}$$

■

Note

Note that the Y are only "independently distributed" and not "independent and *identically* distributed." This is because the expected value of Y depends on the value of x. Since the Y do not all have the same (identical) distribution, they are only "independently distributed."

As for the results of the theorem above, they may not be too interesting. However, as our **estimators** depend on the Y_i, so too do *their* distributions. And *that* is where the interest arises.

We see this in the next theorem.

Theorem 5.1.2: Distribution of b_1

The distribution of b_1 is $N\left(\beta_1,\ \sigma^2 \frac{1}{S_{xx}}\right)$.

Proof.
Before we start, we need to note that b_1 can be written as a linear combination of the Y_i:

$$b_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})\ Y_i}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \tag{5.1.11}$$

I leave the proof of *this* as an exercise.

Now, since our b_1 is a linear combination of the Y_i, and since the Y_i come from independent Normal distributions, we have that b_1 also follows a Normal distribution (again, see Theorem: The Sum of Normals).

Again, since the Normal distribution has two parameters, the mean and the variance, we need to find those two values, as we do next.

The **expected value** of b_1 is

$$E[b_1] = E\left[\frac{\sum_{i=1}^{n}(x_i - \bar{x})\ Y_i}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right] \tag{5.1.12}$$

$$= \frac{\sum_{i=1}^{n}(x_i - \bar{x})\ E[Y_i]}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{5.1.13}$$

$$= \frac{\sum_{i=1}^{n}(x_i - \bar{x})\ (\beta_0 + \beta_1 x_i)}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{5.1.14}$$

$$= \frac{\sum_{i=1}^{n}(x_i - \bar{x})\ \beta_0}{\sum_{i=1}^{n}(x_i - \bar{x})^2} + \frac{\sum_{i=1}^{n}(x_i - \bar{x})\ \beta_1 x_i}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{5.1.15}$$

$$= \beta_0 \frac{\sum_{i=1}^{n}(x_i - \bar{x})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} + \beta_1 \frac{\sum_{i=1}^{n}(x_i - \bar{x})\ x_i}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{5.1.16}$$

$$= \beta_0 \frac{0}{\sum_{i=1}^{n}(x_i - \bar{x})^2} + \beta_1 \frac{\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{5.1.17}$$

$$= 0 + \beta_1 \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{5.1.18}$$

$$= \beta_1 \tag{5.1.19}$$

In this sequence, note that (*and* be able to prove that):

$$\sum_{i=1}^{n}(x_i - \bar{x}) = 0 \tag{5.1.20}$$

and that

$$\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{i=1}^{n}(x_i - \bar{x})\ x_i \tag{5.1.21}$$

Thus, we know $E[b_1] = \beta_1$; that is, our estimator is unbiased.

The final part is to determine the **variance** of b_1:

$$V[b_1] = V\left[\frac{\sum_{i=1}^{n}(x_i - \bar{x})\ Y_i}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right] \tag{5.1.22}$$

$$= \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2\ V[Y_i]}{\left(\sum_{i=1}^{n}(x_i - \bar{x})^2\right)^2} \tag{5.1.23}$$

$$= \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{\left(\sum_{i=1}^{n}(x_i - \bar{x})^2\right)^2}\ \sigma^2 \tag{5.1.24}$$

$$= \frac{1}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\ \sigma^2 \tag{5.1.25}$$

$$= \sigma^2 \frac{1}{S_{xx}} \tag{5.1.26}$$

Recall that since we will be coming across the quantity $\sum_{i=1}^{n}(x_i - \bar{x})^2$ many, many, many times, we denote it by S_{xx}. And so, putting all of these parts together gives us

$$b_1 \sim N\left(\beta_1,\ \sigma^2 \frac{1}{S_{xx}}\right) \tag{5.1.27}$$

■

Theorem 5.1.3: Covariance of b_1 and $\overline{Y}$

The covariance between our b_1 estimator and $\overline{Y}$ is 0.

Proof.
I leave this as an exercise.

Theorem 5.1.4: Distribution of b_0

The distribution of b_0 is $N\left(\beta_0,\ \sigma^2\left(\frac{1}{n}+\frac{\bar{x}^2}{S_{xx}}\right)\right)$.

Proof.
Remember that our estimator is

$$b_0 = \overline{Y} - b_1\bar{x} \tag{5.1.28}$$

Since we have previously shown $Cov[\overline{Y},\ b_1] = 0$, the proof is straight forward.

First, we note that b_0 is a linear combination of the Y_i. Thus, it follows a Normal distribution. (Again, see Theorem: The Sum of Normals for a proof of this.) Because the Normal distribution has two parameters, we must find formulas for each:

Expected value:

$$E[b_0] = E[\overline{Y} - b_1\bar{x}] \tag{5.1.29}$$

$$= E[\overline{Y}] - E[b_1\bar{x}] \tag{5.1.30}$$

$$= (\beta_0 + \beta_1\bar{x}) - \beta_1\bar{x} \tag{5.1.31}$$

$$= \beta_0 \tag{5.1.32}$$

Variance:

$$V[b_0] = V[\overline{Y} + b_1\bar{x}] \tag{5.1.33}$$

$$= V[\overline{Y}] + V[b_1\bar{x}] + 2\ Cov[\overline{Y},\ b_1\bar{x}] \tag{5.1.34}$$

$$= V[\overline{Y}] + V[b_1]\bar{x}^2 + 2Cov[\overline{Y},\ b_1]\bar{x} \tag{5.1.35}$$

$$= \frac{\sigma^2}{n} + \frac{\sigma^2}{S_{xx}}\bar{x}^2 + 0\bar{x} \tag{5.1.36}$$

$$V[b_0] = \sigma^2\left(\frac{1}{n}+\frac{\bar{x}^2}{S_{xx}}\right) \tag{5.1.37}$$

Finally, putting these three parts together gives us what we want:

$$b_0 \sim N\left(\beta_0,\ \sigma^2\left(\frac{1}{n}+\frac{\bar{x}^2}{S_{xx}}\right)\right) \tag{5.1.38}$$

■

The Estimates and Predictions

The above looked at distributions of the usual parameters of interest. Here, let us look at the distribution of estimates and predictions... hoping to tease out the difference between an estimate and a prediction.

Theorem 5.1.5: Distribution of $\hat{Y}_i$

The distribution of Y for an observed value of x_i, which we will term $\hat{Y}_i$, is

$$\hat{Y}_i \sim N\left(\beta_0 + \beta_1 x_i,\ \sigma^2\left(\frac{1}{n} + \frac{(x_i - \bar{x})^2}{S_{xx}}\right)\right) \tag{5.1.39}$$

Note

What does this actually mean?

If we repeat this experiment (of collecting a sample of size n) an infinite number of times and estimate $\hat{Y}_i$ for each of those experiments using our formulas, then those many $\hat{Y}_i$ would follow the specified distribution.

Proof.

Remember that $\hat{Y}_i = b_0 + b_1 x_i$ and that x is non-stochastic (it is not a random variable). With this, we have that $\hat{Y}_i$ is a linear combination of Normally distributed random variables (b_0 and b_1). As such, the name of the distribution of $\hat{Y}_i$ is "Normal." What remains is to calculate the expected value and variance.

$$E[\hat{Y}_i] = E[b_0 + b_1 x_i] \tag{5.1.40}$$

$$= E[b_0] + E[b_1 x_i] \tag{5.1.41}$$

$$= \beta_0 + \beta_1 x_i \tag{5.1.42}$$

As expected, the estimator is unbiased.

What about the **variance**? That is a bit more difficult, because we must deal with the covariance between b_0 and b_1.

$$V[\hat{Y}_i] = V[b_0 + b_1 x_i] \tag{5.1.43}$$

$$= V[b_0] + V[b_1 x_i] + 2Cov[b_0,\ b_1 x_i] \tag{5.1.44}$$

$$= V[b_0] + V[b_1]x_i^2 + 2Cov[b_0,\ b_1]\ x_i \tag{5.1.45}$$

$$= \sigma^2\left(\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}\right) + \sigma^2\left(\frac{1}{S_{xx}}\right)x_i^2 + 2\frac{-\bar{x}\sigma^2}{S_{xx}}x_i \tag{5.1.46}$$

$$= \frac{\sigma^2}{n} + \frac{\sigma^2}{S_{xx}}\left(\bar{x}^2 + x_i^2 - 2\bar{x}x_i\right) \tag{5.1.47}$$

$$= \frac{\sigma^2}{n} + \frac{\sigma^2}{S_{xx}}(\bar{x} - x_i)^2 \tag{5.1.48}$$

$$= \sigma^2\left(\frac{1}{n} + \frac{(\bar{x} - x_i)^2}{S_{xx}}\right) \tag{5.1.49}$$

And so, putting these three things together gives us our hoped-for result

$$\hat{Y}_i \sim N\left(\beta_0 + \beta_1 x_i,\ \sigma^2\left(\frac{1}{n} + \frac{(x_i - \bar{x})^2}{S_{xx}}\right)\right) \tag{5.1.50}$$

... as we expected.

■

Note

There are a couple of things interesting about this result.

First, the uncertainty in $\hat{Y}_i$ is a function of n, S_{xx}, and $\bar{x} - x_i$. Larger sample sizes (larger n) produce a more precise estimate.

Second, samples with larger values of S_{xx} also produce more precise estimates. To maximize S_{xx}, the researcher must have half of the x_i values at the minimum and half at the maximum. Unfortunately, the drawback to doing this is that one is not able to detect a curvature in the expected values of Y. Thus, we again see that there is a trade off in statistics. The important part is to understand what you are trying to understand... and use your statistical understanding to understand it.

Finally, the precision of the estimate also depends on how far that x value is from the center of gravity, $(\bar{x},\ \bar{y})$. Note that the uncertainty in $\hat{Y}_i$ when $x = \bar{x}$ *only* comes from the uncertainty in the value of $\bar{Y}$.

Convince yourself that all of this makes sense (non-mathematically). Draw graphics to illustrate these results. Trust me, it will help you better connect the mathematics with the reality.

Theorem 5.1.6: Distribution of Y_{new}

The distribution of Y_{new}, a new observation, for a new value of x, is

$$Y_{new} \sim N\left(\beta_0 + \beta_1 x_{new},\ \sigma^2\left(1 + \frac{1}{n} + \frac{(x_{new} - \bar{x})^2}{S_{xx}}\right)\right) \tag{5.1.51}$$

Note

Before we begin this proof, remember that

$$Y_{new} = b_0 + b_1 x_{new} + \varepsilon = \hat{Y}_{new} + \varepsilon \tag{5.1.52}$$

Since we are estimating a new *observation* (as opposed to just an expected value), we need to include ε in our calculations. This is subtle and *very* important. It emphasizes the importance of ε.

Also, before we start the proof, compare and contrast this distribution with the distribution of $\hat{Y}_i$. What is the difference? Where does that difference come from?

Proof.

And now for the expected proof. See that Y_{new} is a linear combination of Normally distributed random variables (b_0, b_1, and ε). Thus, Y_{new} follows a Normal distribution. All that remains is to calculate its expected value and its variance. To do so, we rely on the previous theorem.

$$E[Y_{new}] = E[b_0 + b_1 x_{new} + \varepsilon] \tag{5.1.53}$$

$$= E[b_0 + b_1 x_{new}] + E[\varepsilon] \tag{5.1.54}$$

$$= \beta_0 + \beta_1 x_{new} + 0 \tag{5.1.55}$$

$$= \beta_0 + \beta_1 x_{new} \tag{5.1.56}$$

Next, for the variance:

$$V[Y_{new}] = V[b_0 + b_1 x_{new} + \varepsilon] \tag{5.1.57}$$

$$= V[\hat{Y} + \varepsilon] \tag{5.1.58}$$

$$= V[\hat{Y}] + V[\varepsilon + 2Cov[\hat{Y}, \varepsilon] \tag{5.1.59}$$

$$= \sigma^2 \left(\frac{1}{n} + \frac{(\bar{x} - x_{new})^2}{S_{xx}} \right) + \sigma^2 + 0 \tag{5.1.60}$$

$$= \sigma^2 \left(1 + \frac{1}{n} + \frac{(\bar{x} - x_{new})^2}{S_{xx}} \right) \tag{5.1.61}$$

Putting these parts together gives us the distribution of a new observation (a *prediction*):

$$Y_{new} \sim N\left(\beta_0 + \beta_1 x_{new},\ \sigma^2 \left(1 + \frac{1}{n} + \frac{(x_{new} - \bar{x})^2}{S_{xx}} \right) \right) \tag{5.1.62}$$

■

Note that the only difference in the uncertainties between Y_{new} and $\hat{Y}$ is an additional term of σ^2 due to the inclusion of the residuals. Thus, all of the things that affect the variance of $\hat{Y}$ also affect the variance of Y_{new}, and in the same way.

Note

Also note that the uncertainty in an observation is higher than the uncertainty in the expected value (see Figure **5.1.1**, below).

The important difference between this theorem and the previous is that this theorem models a **new** observation, while the previous models the **expected value** of an observation. The difference is important.

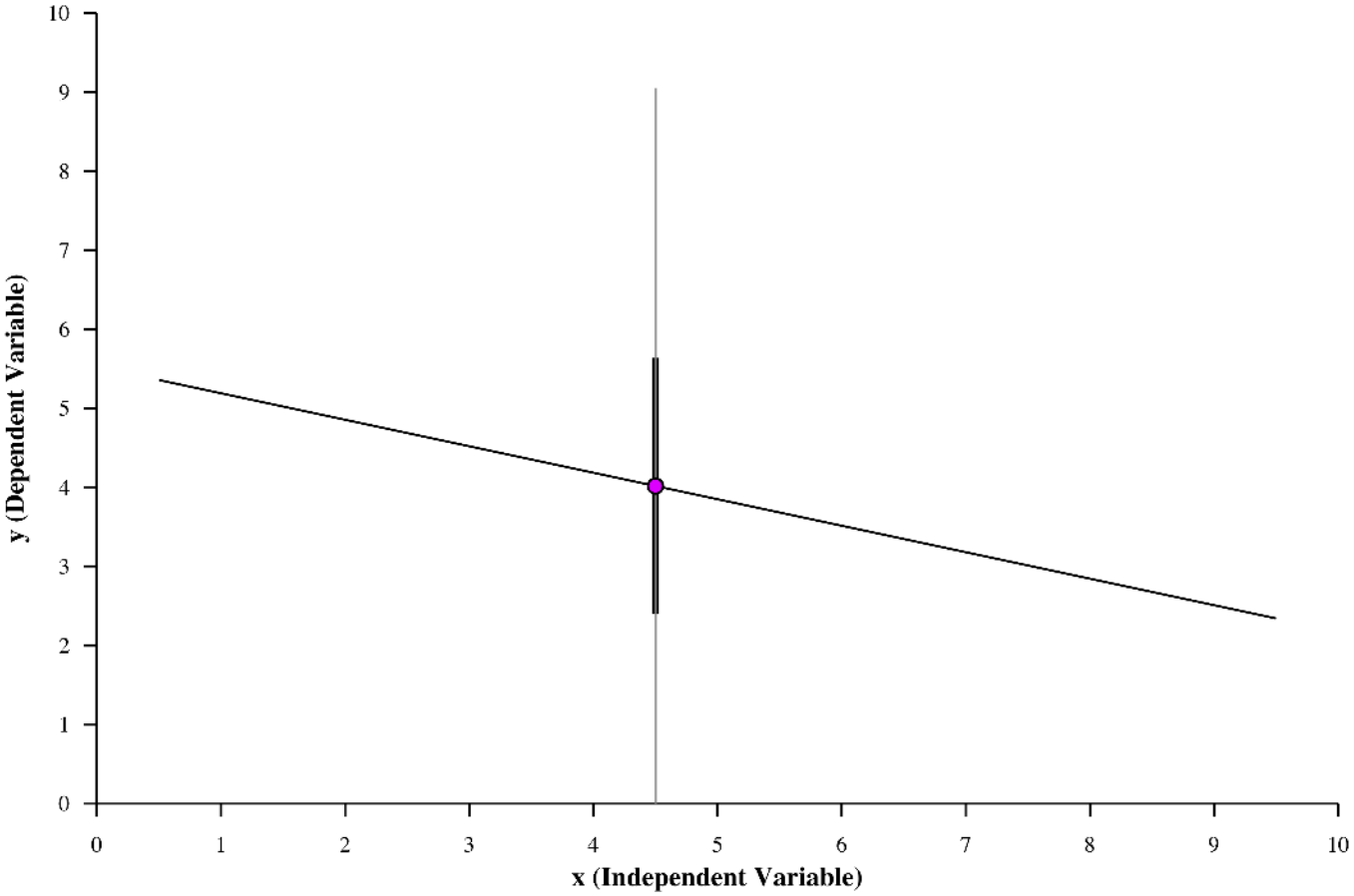


Figure 5.1.1: The basic scatter plot with the confidence and prediction intervals for $x = 4.5$ provided. Note that the prediction interval (thin line) is much wider than the confidence interval (thick line). This is because the prediction interval uncertainty includes *both* the uncertainty in the mean value (confidence interval) and the inherent variation in the residuals (σ^2).

The Mean Square Error

There is another parameter in our model that we may like to estimate. That is the variance of ε. The ordinary least squares estimator of σ^2 is called the mean square error. It is defined as

$$\text{MSE} = \frac{1}{n-p}\sum_{i=1}^{n}\varepsilon_i \tag{5.1.63}$$

Here, p is the number of parameters in the regression. So far, we have dealt with estimating β_0 and β_1. Thus, $p = 2$ in simple linear regression.

Theorem 5.1.7: The Distribution of the MSE

The distribution of the mean square error, **MSE**, can be written as

$$\frac{(n-p)\ \text{MSE}}{\sigma^2} \sim \chi^2_{n-p} \tag{5.1.64}$$

Proof.

The first thing to do is remind ourselves of the definition of a χ^2 random variable. From the definition of the Chi-square distribution, we have that if $Z_i \sim N(0,\ 1)$, then $\sum Z_i^2 \sim \chi^2_\nu$, where ν is the number of those Z_i that are independent (the degrees of freedom).

With this definition, we just need to find a random variable with a Normal distribution and transform it into the proper form. To that end, here is some algebra:

$$\varepsilon_i \sim N\left(0,\ \sigma^2\right) \tag{5.1.65}$$

$$\frac{\varepsilon_i}{\sigma} \sim N(0,\ 1) \tag{5.1.66}$$

$$\frac{\varepsilon_i^2}{\sigma^2} \sim \chi^2_{\nu=1} \tag{5.1.67}$$

$$\frac{\sum \varepsilon_i^2}{\sigma^2} \sim \chi^2_{\nu=n-p} \tag{5.1.68}$$

$$\frac{(n-p)\ \frac{1}{n-p} \sum \varepsilon_i^2}{\sigma^2} \sim \chi^2_{n-p} \tag{5.1.69}$$

$$\frac{(n-p)\ \text{MSE}}{\sigma^2} \sim \chi^2_{n-p} \tag{5.1.70}$$

And this is what we were to prove.

■

As usual, knowing the distribution of a sample statistic like the **MSE** allows us to create confidence intervals and perform hypothesis testing about the variance of the residuals, σ^2.

Note

With that said, the importance of the previous theorem lies more in how we can use it to obtain confidence intervals and test hypotheses about the OLS estimators of the intercept and slope parameters.

By the way, the reason that equation 5.1.68 has $n-p$ degrees of freedom is that there are only $n-p$ independent terms. The other p terms can be determined (to within a constant) from the $n-p$ terms.

This page titled 5.1: Probability Distributions is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

5.2: Test Statistics and Hypothesis Testing

Building on the solid mathematical foundation established throughout this chapter, we now possess the essential tools to move from pure derivation to practical inference. The preceding section formalized the sampling distributions of our key estimators — specifically, showing that under standard assumptions, the estimated regression coefficients follow a multivariate normal distribution, and the error variance estimate, appropriately scaled, follows a chi-square distribution. This critical step transforms our algebraic results into probabilistic ones.

With these exact distributions in hand — and armed with our foundational knowledge of probability theory — we can now construct rigorous statistical procedures to test individual hypotheses. This allows us to interrogate specific, targeted questions about our model, such as whether a particular predictor variable, β_i, has a statistically significant partial relationship with the outcome after accounting for all other variables in the model.

Learning Objectives

By the end of this section, you will be able to:

1. Construct a t-statistic to test hypotheses about an individual regression coefficient, such as the slope β_1, by standardizing the estimator with its standard error and the mean square error.
2. Explain why the resulting test statistic follows a Student's t-distribution with $\nu = n - p$ degrees of freedom under the null hypothesis, based on the known distributions of the estimator and the MSE.
3. Apply this result to conduct hypothesis tests and construct confidence intervals for β_1, and recognize that analogous t-statistics can be formed for β_0 and for fitted values $\hat{Y}_i$.

✦•·····• ✦ •·····•✦

For this section, we rely heavily on the definition of Student's t distribution. Please review it before continuing.

Definition: Student's t Distribution

If we let $Z \sim N(0,\ 1)$ and $V \sim \chi^2_\nu$, with Z and V independent, then

$$T \stackrel{\text{def}}{=} \frac{Z}{\sqrt{V/\nu}} \tag{5.2.1}$$

follows a Student's t distribution with ν degrees of freedom.

You have, most likely, come across this ratio in your elementary statistics course when you were investigating hypotheses about a single population mean (assuming that the data came from a Normal distribution).

Theorem 5.2.1: Distribution of T(b_1)

The quantity

$$T = \frac{b_1 - \beta_1}{\sqrt{\text{MSE}/S_{xx}}} \tag{5.2.2}$$

follows a Student's t distribution with $n - p$ degrees of freedom.

Proof.

To prove this statement, one must show that it can be written in the form of Equation 5.2.1. First, let us look at the numerator.

$$b_1 \sim N\left(\beta_1,\ \sigma^2/S_{xx}\right) \tag{5.2.3}$$

$$\implies \quad \frac{b_1-\beta_1}{\sqrt{\sigma^2/S_{xx}}} \sim N(0,\ 1) \tag{5.2.4}$$

Now for the denominator we use a previous theorem (The MSE Theorem):

$$\frac{(n-p)\ \text{MSE}}{\sigma^2} \sim \chi^2_{n-p} \tag{5.2.5}$$

Next, we put these two together

$$T = \frac{b_1-\beta_1}{\sqrt{\text{MSE}/S_{xx}}} \tag{5.2.6}$$

$$= \frac{\dfrac{b_1-\beta_1}{\sqrt{\sigma^2/S_{xx}}}}{\sqrt{\dfrac{\text{MSE}}{\sigma^2}}} \tag{5.2.7}$$

$$= \frac{\frac{b_1-\beta_1}{\sqrt{\sigma^2/S_{xx}}}}{\sqrt{\frac{(n-p)\ \text{MSE}}{\sigma^2}/(n-p)}} \tag{5.2.8}$$

Note that the numerator of Equation 5.2.8 follows a standard Normal distribution, while the denominator is the square-root of a chi-square distribution divided by its degrees of freedom. Thus, by the definition of Student's t distribution, the quantity T follows a Student's t distribution with $n-p$ degrees of freedom.

$$T \sim t_{n-p} \tag{5.2.9}$$

■

Note

This result is important for two reasons:

First, it allows us to test hypotheses regarding the β_1 parameter.

Second, this result allows us to calculate confidence intervals for β_1 (see Section 5.3: Confidence Intervals).

This parameter is usually of most interest to researchers as it provides "the *effect* of the independent variable on the dependent variable."

Since we know the distribution of this ratio, we can calculate p-values for any hypothesis about β_1 using the same rules as from your elementary statistics course (see the Table below).

Table 5.2.1: How to calculate p-values, given the alternative hypothesis.

Hypothesis Testing		
$H_0 : \beta_1 = \beta_{1,0}$	$H_A : \beta_1 \neq \beta_{1,0}$	p-value = $P[t \leq -\lvert T\rvert] \times 2$
$H_0 : \beta_1 \leq \beta_{1,0}$	$H_A : \beta_1 > \beta_{1,0}$	p-value = $P[t \geq T]$
$H_0 : \beta_1 \geq \beta_{1,0}$	$H_A : \beta_1 < \beta_{1,0}$	p-value = $P[t \leq T]$

Technically, we *do* need to show that b_1 and MSE are independent. If they are not, then Theorem 1 above theorem is not valid. For the proof, you will want to investigate Cochran's Theorem and its uses (Bapat 2000, Cochrane 1934).

Theorem 5.2.2: Distribution of T(b_0)

The ratio

$$T = \frac{b_0 - \beta_0}{\sqrt{\text{MSE}\left(\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}\right)}} \tag{5.2.10}$$

follows a Student's t distribution with $n - p$ degrees of freedom.

Proof. I leave this as an exercise.

This theorem allows us to easily prove the next.

Theorem 5.2.3: Distribution of T($\hat{Y}$)

The ratio

$$T = \frac{\hat{y} - \hat{y}_0}{\sqrt{\text{MSE}\left(\frac{1}{n} + \frac{(x-\bar{x})^2}{S_{xx}}\right)}} \tag{5.2.11}$$

follows a Student's t distribution with $n - p$ degrees of freedom.

Proof. I leave this as an exercise, too.

That sure is a lot of exercise.

How are those abs doing? Sore yet?

This page titled 5.2: Test Statistics and Hypothesis Testing is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

5.3: Confidence Intervals

In the previous section, we examined hypothesis testing. This required that we create a test statistic and determine its distribution. One can think of confidence intervals as the **dual** of test statistics. Test statistics are functions of an unknown population parameter and have a distribution. Confidence intervals are for that unknown population parameter, where a probability is known (assumed). Once a person has the test statistic and its definition, the confidence interval can be determined by inverting the test statistic function (solve for the parameter).

Learning Objectives

By the end of this section, you will be able to:

1. Explain the relationship between a hypothesis test and a confidence interval, recognizing that a confidence interval can be constructed by inverting the corresponding test statistic.
2. Construct and interpret confidence intervals for the population slope β_1, the intercept β_0, and the mean response at a given x ($\hat{Y}$), using the general form: estimate ± (critical t-value) × (standard error of the estimate).
3. Construct and interpret a prediction interval for a new individual observation Y_{new}, and explain why this interval is wider than the confidence interval for the mean response at the same x value.

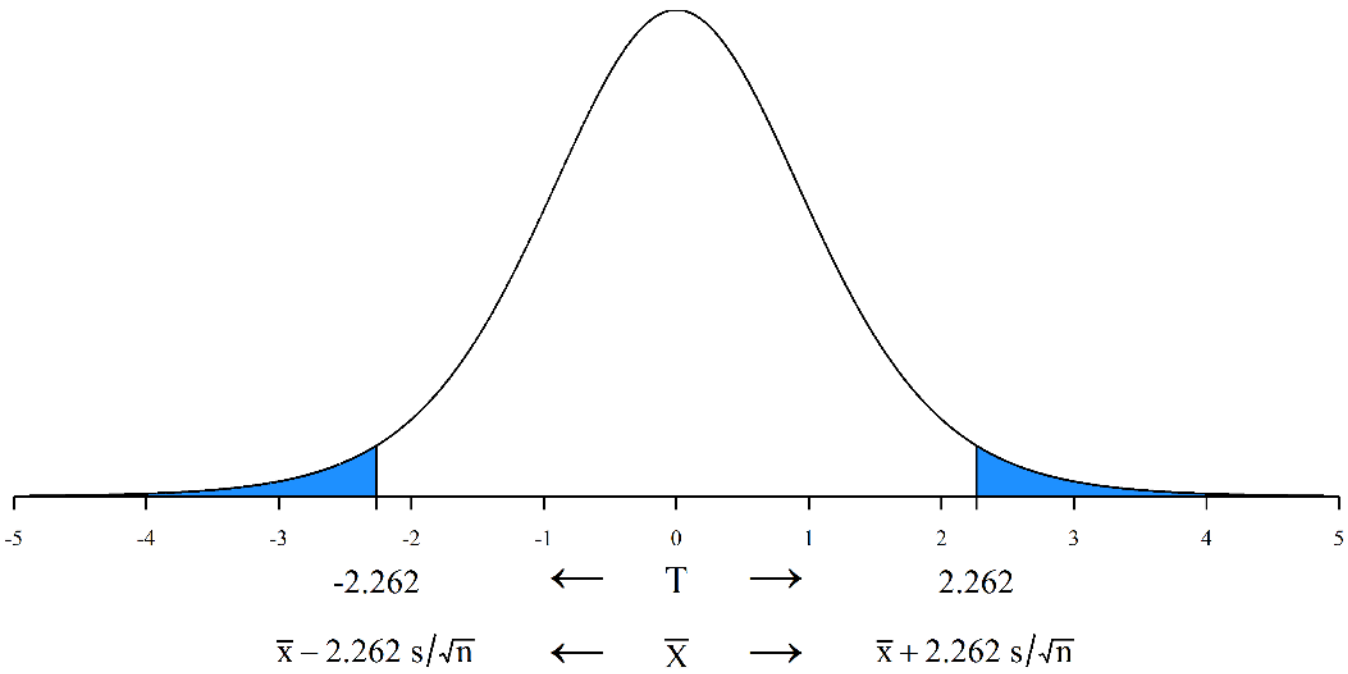


Figure **5.3.1**: An illustration of a confidence interval seen from the standpoint of T or from $\bar{X}$. The unshaded area constitutes 95% of the area under the curve. Thus, the vertical segments delimit the endpoints of a central 95% confidence interval.

From your elementary statistics course, you learned that the distribution of

$$T = \frac{\bar{x} - \mu}{s/\sqrt{n}} \tag{5.3.1}$$

followed a Student's t distribution with $n-1$ degrees of freedom. Solving the formula for the parameter of interest, μ, gives

$$\mu = \bar{x} - T\,\frac{s}{\sqrt{n}} \tag{5.3.2}$$

The interpretation of T here is that it contains the values (quantiles) that correspond to the confidence level claimed (Figure 5.3.1). For instance, if you desire a 95% confidence interval for a sample of size 10, the central T values are ± 2.262 because the probability $P[-2.262 < t < 2.262] = 0.95$.

Thus, the interpretation of μ in Equation 5.3.2 is that it contains the values that correspond to the endpoints of the confidence level claimed for the distribution of the right-hand side of the formula.

> This interpretation holds for *all* confidence intervals.

With this discussion, it is rather straightforward to calculate the endpoints of confidence intervals for *all of the population parameters* we have explored thus far.

> When the distribution of the test statistic is unimodal and symmetric (like the Normal or the Student's t), the central confidence interval is *also* the narrowest. Obtaining the narrowest interval *may* be important if the researcher desires the most precise estimate of the population parameter

Theorem 5.3.1: Confidence Interval for β_1

The endpoints of a central $(1-\alpha)\times 100$ confidence for β_1 are defined by

$$b_1 \pm t_{\alpha/2,n-p}\sqrt{\text{MSE}/S_{xx}} \tag{5.3.3}$$

Proof.

From our theorem on the distribution of β_1, we know $T = \frac{b_1-\beta_1}{\sqrt{\text{MSE}/S_{xx}}}$ follows a t distribution with $n-p$ degrees of freedom. Solving this for β_1 gives

$$\beta_1 = b_1 - T\sqrt{\text{MSE}/S_{xx}} \tag{5.3.4}$$

Because the distribution of T is symmetric unimodal, the endpoints of the minimum-width interval for T correspond to the two quantiles $t_{\alpha/2,n-p}$ and $t_{1-\alpha/2,n-p}$. These two endpoints are equivalent to $\pm t_{\alpha/2,n-p}$.

As such, the endpoints of a minimum-length $(1-\alpha)\times 100$ confidence for β_1 are defined by

$$b_1 \pm t_{\alpha/2,n-p}\sqrt{\text{MSE}/S_{xx}} \tag{5.3.5}$$

■

This is a **typical result** when dealing with the Student's t distribution.

By the way, there is absolutely no reason we *need* a minimum-width confidence interval. It is, however, useful in maximizing the precision of the estimate. But, that's about all it is good for. When the distribution of the test statistic is unimodal and symmetric, the central interval and the minimum-width interval are identical. When the distribution is *not* symmetric, they are not. The following illustrates this.

Theorem 5.3.2: Confidence Interval for the MSE

The endpoints of a central $(1-\alpha)100$ confidence for σ^2 are defined by

$$\frac{(n-p)\ \text{MSE}}{\chi^2_{1-\alpha/2,n-p}} \quad \text{and} \quad \frac{(n-p)\ \text{MSE}}{\chi^2_{\alpha/2,n-p}} \tag{5.3.6}$$

Proof.

From the MSE Theorem, we know

$$\frac{(n-p)\ \text{MSE}}{\sigma^2} \sim \chi^2_{n-p} \rightarrow \quad \sigma^2 \quad = \frac{(n-p)\ \text{MSE}}{\chi^2_{n-p}} \tag{5.3.7}$$

Thus, a central $(1-\alpha)100$ confidence interval (see Figure 5.3.2, below) is defined by the endpoints

$$\frac{(n-p)\ \text{MSE}}{\chi^2_{n-p,1-\alpha/2}} \quad \text{and} \quad \frac{(n-p)\ \text{MSE}}{\chi^2_{n-p,\alpha/2}} \tag{5.3.8}$$

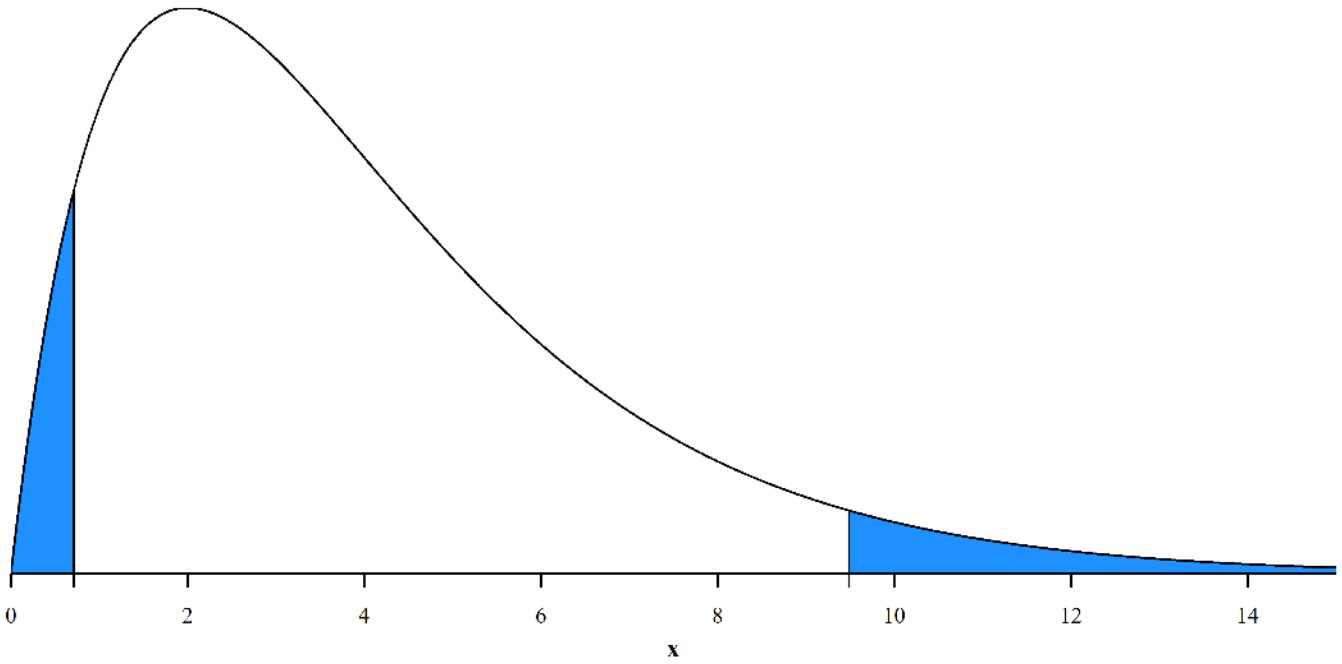


Figure 5.3.2: A plot of the chi-square distribution with 4 degrees of freedom. The unshaded area constitutes 90% of the area under the curve. Thus, the vertical segments delimit the endpoints of a central 90% confidence interval.

Note

This is *not* the minimum-width interval. It is, however, the *usual* confidence interval provided. Calculating the minimum-width interval takes a little calculus that is beyond the scope of this section... and the typical coverage of this topic.

The minimum-width interval is illustrated in Figure **5.3.3**, below. Note that the area in the shaded area to the right is not the same as that to the left. However, the two areas still account for 10% of the area, leaving 90% unshaded in the middle.

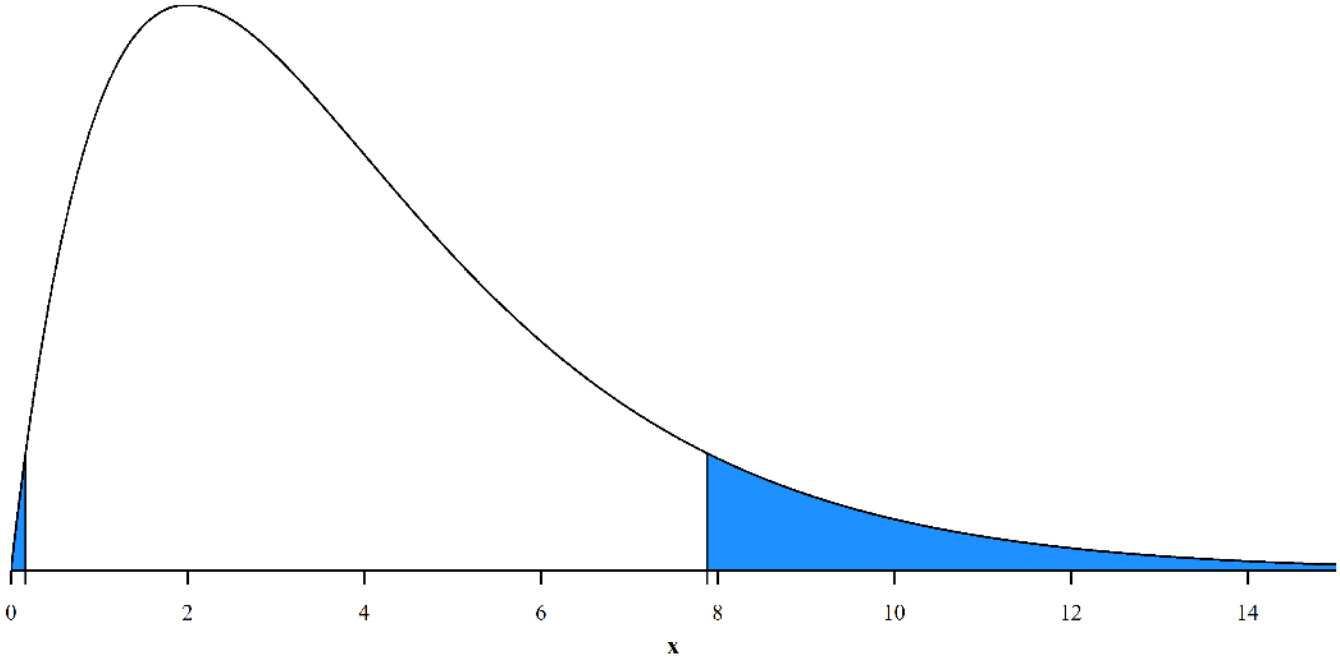


Figure **5.3.3**: A plot of the chi-square distribution with 4 degrees of freedom. The shaded area constitutes 10% of the area under the curve. Thus, the vertical segments delimit the endpoints of a 90% confidence interval. *This* confidence interval, however, is the minimum-width interval.

The width of the central 90% confidence interval shown in Figure **5.3.2** is 8.777. This is wider than the width of the minimum-width confidence interval shown in Figure **5.3.3** , which is 7.714. The minimum-width interval is 12% narrower than the central interval. That is an increase in estimator efficiency.

How do we obtain the minimum-width interval? Let's make that topic beyond the scope of this course. However, as a teaser, notice that the value of the density function for each of the two endpoints is the same in the minimum-width interval. If the distribution is unimodal, then that observation will be true. That's enough of a hint. Feel free to explore this on your own. Calculus will serve you well here.

Theorem 5.3.3: Confidence Interval for β_0

The endpoints of a central (and minimum width) confidence interval for β_0 are defined by

$$b_0 \pm t_{\alpha/2,n-p}\sqrt{\text{MSE}\left(\frac{1}{n}+\frac{\bar{x}^2}{S_{xx}}\right)} \tag{5.3.9}$$

Proof. I leave this as an exercise.

Theorem 5.3.4: Confidence Interval for Y

The endpoints of a confidence interval for $\hat{y}$ are defined by

$$b_0 + b_1 x \pm t_{\alpha/2, n-p} \sqrt{\text{MSE}\left(\frac{1}{n} + \frac{(x-\bar{x})^2}{S_{xx}}\right)} \tag{5.3.10}$$

Proof. I leave this as an exercise.

Theorem 5.3.5: Prediction Interval for Y

The endpoints of a prediction interval for $y(new$ are defined by

$$b_0 + b_1 x \pm t_{\alpha/2, n-p} \sqrt{\text{MSE}\left(1 + \frac{1}{n} + \frac{(x-\bar{x})^2}{S_{xx}}\right)} \tag{5.3.11}$$

Proof. I leave this as an exercise.

Note

This interval (Theorem 5.3.5) is termed a "prediction interval" because it is used to *predict a new observation* of y. It is not used to estimate the *expected* value of y — or trends in y. That would be the purpose of a confidence interval.

This page titled 5.3: Confidence Intervals is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

5.4: The Working-Hotelling Bands

Building upon our exploration of confidence intervals for individual predictions, one last confidence interval we may be interested in is a confidence interval for the regression line itself. This concept extends beyond predicting a single future observation to instead defining a confidence region for the entire conditional mean function — the true average response for all possible values of the predictor. To address this, we employ Working-Hotelling (1929) confidence bands, which construct a set of simultaneous confidence intervals for the expected value of the response across the entire range of X. Unlike pointwise intervals that guarantee coverage at a *single* x-value, this method creates a hyperbolic band around the fitted line that ensures, with a specified confidence level (e.g., 95%), that the *entire true regression line* lies within the band simultaneously.

This broader, more conservative approach is essential when making inferences about the **underlying linear relationship *as a whole***.

Learning Objectives

By the end of this section, you will be able to:

1. Explain the purpose of Working-Hotelling confidence bands as a method for constructing a simultaneous confidence region for the entire true regression line, ensuring that the whole line lies within the bands with a specified confidence level.
2. State the formula for the Working-Hotelling bands and identify how it differs from the pointwise confidence interval for the mean response $\hat{Y}$, specifically noting the use of an F-distribution critical value to account for simultaneous inference.
3. Describe the conceptual basis of the proof, which involves considering the joint distribution of the estimated coefficients b_0 and b_1 and transforming their elliptical confidence region into a hyperbolic band around the regression line in the original $x-y$ space.

✦•·····• ✦ •·····•✦

Note that all of the confidence intervals in this chapter (except for the non-central σ^2) have been of the form

$$\text{point estimate} \pm K \times se \tag{5.4.1}$$

That is because they were confidence intervals for *a measure of center*. The Working-Hotelling (1929) confidence band for the regression line follows this format. It is

$$(b_0 + b_1 x) \pm F_{2,n-2} \times \sqrt{\text{MSE}\left(\frac{1}{n} + \frac{(x-\bar{x})^2}{S_{xx}}\right)} \tag{5.4.2}$$

The proof is very interesting. The key is to focus on the *joint distribution* of b_0 and b_1. This joint distribution is bivariate Normal. Thus, "confidence intervals" take the form of confidence *ellipses* with the same meaning and interpretation. However, as is common for confidence regions, the distribution of interest is the Chi-squared, instead of the Normal. Why?

Ans: Think about the formula for an ellipse.

Finally, the problem is transformed from the β_0-β_1 plane to the x-y plane.

Believe it or not, the hardest part of the proof is the algebra.

Question: So, where does the F distribution come from in the formula?

Ans: The same place as the Student's t distribution in the univariate case: the fact that we do not know the population variances involved.

Note

Technically, Working and Hotelling only worked in the case of knowing the population variances σ_0^2 and σ_1^2 , which led to a Chi-square distribution in the formula. This is because the F distribution had not been invented (or discovered) yet. It was not until George W. Snedecor in 1934 that we were able to take that final step.

The rest of this section proves this result. It progresses in three phases.

The Proof of Working-Hotelling (1929)*

This extra proof follows this structure.

1. First, it determines the joint distribution of the OLS estimators $\mathbf{b}$.
2. Second, it thinks about confidence regions in that coordinate system.
3. Finally, it transforms that distribution of $\mathbf{b}$ into a distribution of ℓ, the lines.

That is all. The first part is rather straightforward and should remind us of what we have already done in this chapter. The second is a simple exercise in algebra. The third is intense algebra and geometry.

The reason I am providing this proof is because this is probably the first time you have been exposed to thinking about a problem by working on a similar problem, then transforming it. Thus, the most important thing about this proof is not the actual proof, but the logic behind it that you can take with you to other problems.

Part 1: The Joint Distribution of b_0, b_1

The first thing to note is that the line of best fit depends on both b_0 and b_1, which are random variables. Thus, to speak of a confidence region for the regression line, we can first look at the joint confidence region of b_0 and b_1. From previous theorems, we determined that the OLS estimator of $\mathbf{B}$ has this distribution:

$$\mathbf{b} \sim N\left(\mathbf{B}; \sigma^2 \left(\mathbf{X}'\mathbf{X}\right)^{-1/2}\right) \tag{5.4.3}$$

That is, their joint distribution is a bivariate Normal distribution. Bivariate Normal distributions have this form

$$f(\mathbf{b}) = \frac{1}{2\pi}(\det \mathbf{\Sigma})^{-1/2} e^{-\frac{1}{2}(\mathbf{b}-\mathbf{B})'\mathbf{\Sigma}^{-1}(\mathbf{b}-\mathbf{B})} \tag{5.4.4}$$

In Equation 5.4.4, $\mathbf{\Sigma}$ is the covariance matrix of $\mathbf{b}$, which we already calculated to be

$$\sigma^2(\mathbf{X}'\mathbf{X})^{-1/2} \tag{5.4.5}$$

Question: What does this distribution *look like*???

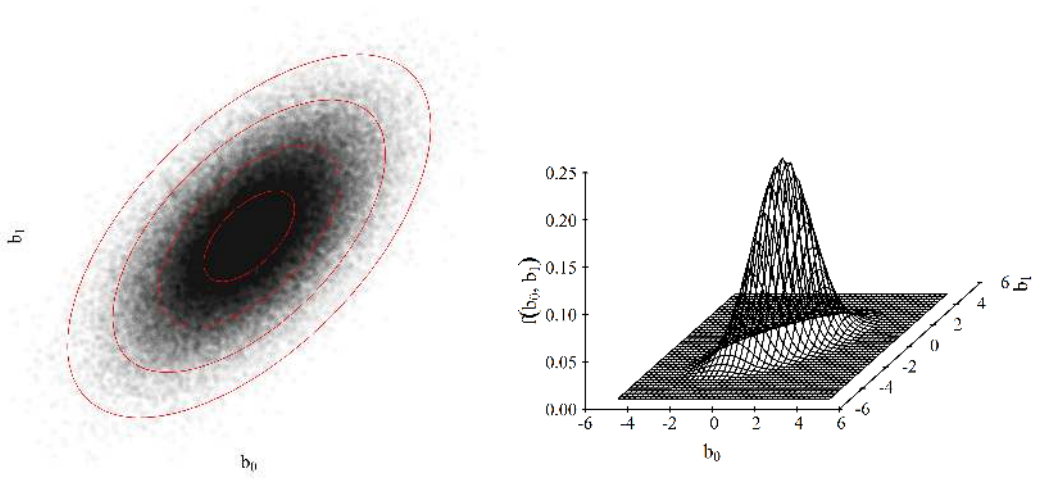


Figure 5.4.1: Two representations of a Bivariate Normal distribution. The left panel is a simple scatter plot with ellipses of equal density marked. The right panel is an attempt at a three-dimensional density mesh plot. You can think of the red ellipses on the left as being horizontal cuts through the graphic on the right.

Figure 5.4.1, above, is a plot of the b_0 and b_1 values generated from the distribution of $\mathbf{b}$. The red curves in the left panel are ellipses of equal density. The smallest contains 50% of the distribution. The outermost contains 99% of the distribution. Thus, a 99% confidence region for $\mathbf{B}$ is inside that outer ellipse.

This point is rather important. We now have a confidence region for the estimators b_0 and b_1. This means, with a little algebra, we have a confidence region for the lines of best fit.

Part 2: The Confidence Region

Let us return to Equation 5.4.4. The only place that the $\mathbf{b}$ estimator occurs is in the exponent. So, it would behoove us to look closely at the exponent $(\mathbf{b}-\mathbf{B})'\mathbf{\Sigma}^{-1}(\mathbf{b}-\mathbf{B})$. If we can get a distribution for it, then we have simplified the problem a bit.

Lemma 5.4.1

The quantity $(\mathbf{b}-\mathbf{B})'\mathbf{\Sigma}^{-1}(\mathbf{b}-\mathbf{B}) \sim \chi^2_2$.

Proof.
Recall that the χ^2 distribution is defined as the sum of squared standard Normal random variables, with ν of them being independent. Yeppers, that is the definition. Only *later* did we find a probability density function for it.

So, we have two things to do for this part of the proof:

1. First, show that the quantity is the sum of squared standard Normal distributions.
2. Second, to determine how many are independent.

Both parts are trivial.

First, the distribution of $\mathbf{b}$ is $N(\mathbf{B}; \mathbf{\Sigma})$. Thus,

$$\mathbf{b}-\mathbf{B}\sim N(\mathbf{0};\boldsymbol{\Sigma}) \tag{5.4.6}$$

$$\boldsymbol{\Sigma}^{-1/2}(\mathbf{b}-\mathbf{B})\sim N(\mathbf{0};\mathbf{I}_2) \tag{5.4.7}$$

$$\left(\boldsymbol{\Sigma}^{-1/2}(\mathbf{b}-\mathbf{B})\right)'\left(\boldsymbol{\Sigma}^{-1/2}(\mathbf{b}-\mathbf{B})\right)\sim\chi^2_{\nu=2} \tag{5.4.8}$$

$$(\mathbf{b}-\mathbf{B})'\boldsymbol{\Sigma}^{-1}(\mathbf{b}-\mathbf{B})\sim\chi^2_{\nu=2} \tag{5.4.9}$$

The **second part** is determining the number of degrees of freedom, ν. In general, the number of degrees of freedom is the rank of the matrix. Here, the rank of $(\mathbf{b}-\mathbf{B})'\boldsymbol{\Sigma}^{-1}(\mathbf{b}-\mathbf{B})$ is 2, the number of parameters in $\mathbf{b}$.

Thus, we have shown that $(\mathbf{b}-\mathbf{B})'\boldsymbol{\Sigma}^{-1}(\mathbf{b}-\mathbf{B})\sim\chi^2_2$.

■

By the way, in the simple linear regression case, what is $(\mathbf{b}-\mathbf{B})'\boldsymbol{\Sigma}^{-1}(\mathbf{b}-\mathbf{B})$? It is

$$\frac{\left(b_0-\beta_0\right)}{\sigma_0^2}+\frac{\left(b_1-\beta_1\right)}{\sigma_1^2}=\chi^2_{\nu=2} \tag{5.4.10}$$

So, we have determined a confidence region for $\mathbf{b}=[b_0,b_1]'$. All we have to do is transform from $\mathbf{b}$ coordinates to (x,y) coordinates. That is done in the next section.

Part 3: The Envelope for ℓ

The previous section gave us a distribution for the quantity $(\mathbf{b}-\mathbf{B})'\boldsymbol{\Sigma}^{-1}(\mathbf{b}-\mathbf{B})$. That is, we now have a confidence region for b_0 and b_1 jointly (see left panel of Figure 5.4.2, below). This final part of the proof finds the equivalent region in terms of x and y (see right panel of Figure 5.4.2, below). In other words, given that the envelope for $\mathbf{b}$ is elliptical, what is the envelope for the equivalent lines?

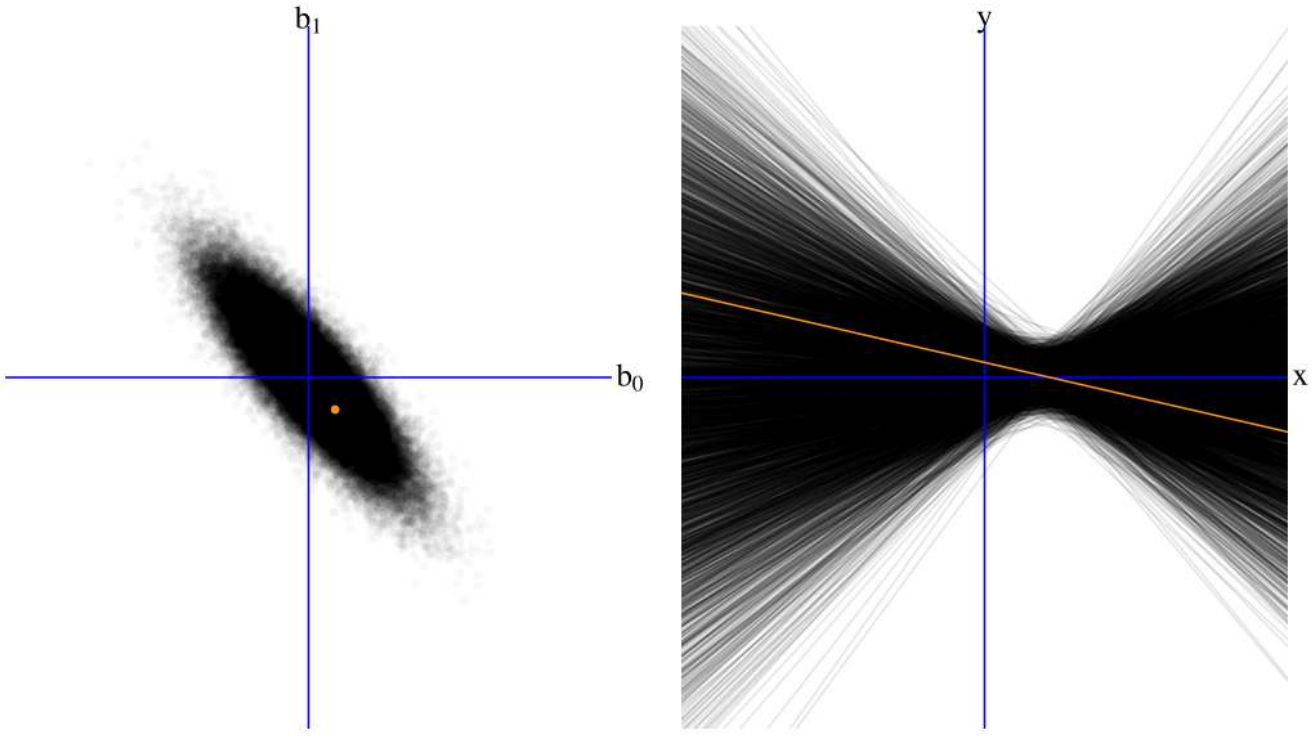


Figure 5.4.2: A graphic showing the distribution of the $\mathbf{b}$ matrix on the left and the corresponding lines on the left. The orange dot identifies the orange line. On the left, the shape is clearly elliptical. What is the shape on the right? Hyperbolic.

It turns out that the envelope for the lines is a hyperbola! This leads to Equation 5.4.2.

Since the confidence region in b_0-b_1-space is an ellipse (see equation 5.4.9), we have this

$$\frac{(b_0-\beta_0)^2}{\sigma_0^2}+\frac{(b_1-\beta_1)^2}{\sigma_1^2}=\chi^2 \tag{5.4.11}$$

Here, "χ^2" is the appropriate quantile for the confidence region.

Note that there are two parameters in this equation, b_0 and b_1. As usual, since we are trying to get a function in terms of x, y, β_0, and β_1, we try to eliminate them. First, we solve $y=b_0+b_1x$ for b_0, substitute into the equation above, and simplify.

This gives

$$\frac{\left(y-\beta_0-b(x-\bar{x})\right)^2}{\sigma_0^2}+\frac{\left(b_1-\beta_1\right)^2}{\sigma_1^2}=\chi^2 \tag{5.4.12}$$

Note that this is a family of lines, parametrized by b_1. Each line is defined by parameters $\beta_0, \beta_1, \sigma_0^2$, and σ_1^2 and the pairs (x, y).

Because we have set the boundary of the ellipse to be constant (we set the confidence level), we take the derivative of the above with respect to b_1 to get

$$\frac{-(x-\bar{x})\left(y-\beta_0-b_1(x-\bar{x})\right)^2}{\sigma_0^2}+\frac{b_1-\beta_1}{\sigma_1^2}=0 \tag{5.4.13}$$

Solving this for b_1, substituting this in to 5.4.12, and simplifying gives us the answer

$$y=\beta_0+\beta_1(x-\bar{x})\pm\sqrt{\chi_2^2\left(\sigma_0^2+(x-\bar{x})^2\sigma_1^2\right)} \tag{5.4.14}$$

Note

This is as far as Working and Hotelling got. They made the assumption that we knew the population variances, σ_0^2 and σ_1^2. In reality, we never know these, and we use s_0^2 and s_1^2 in their stead.

This additional source of uncertainty (variability) is important... and easily handled. When in the univariate cases, it changes our distribution from a Normal to a Student's t distribution. In this bivariate case, it changes the χ^2 into an F distribution. This is what Snedecor (1934) offered to this problem.

Because of this, we have our final result:

$$y=\beta_0+\beta_1(x-\bar{x})\pm\sqrt{2\,F_{2,n-2}\left(\text{MSE}\,\left(\frac{1}{n}+\frac{(x-\bar{x})^2}{S_{xx}}\right)\right)} \tag{5.4.15}$$

Exploring the Results

Alright. So, the confidence region for the regression line is a hyperbola. What do we know about hyperbolas? A couple of things. First, they have a center.

Corollary 5.4.2: Center of Gravity

The center of gravity for the confidence region for the regression line is $y = b_0 + b_1(x - \bar{x})$

Second, they have asymptotes.

Corollary 5.4.3: Asymptotes

The asymptotes for the confidence region are $y = \beta_0 + \beta_1 \pm \chi\sigma_1)(x - \bar{x})$

Using the Bounds

This graphic illustrates the Working-Hotelling bounds. We are 95% confident that the true relationship between the property crime rate in 1990 and the violent crime rate in 2000 fits between the hyperbolas.

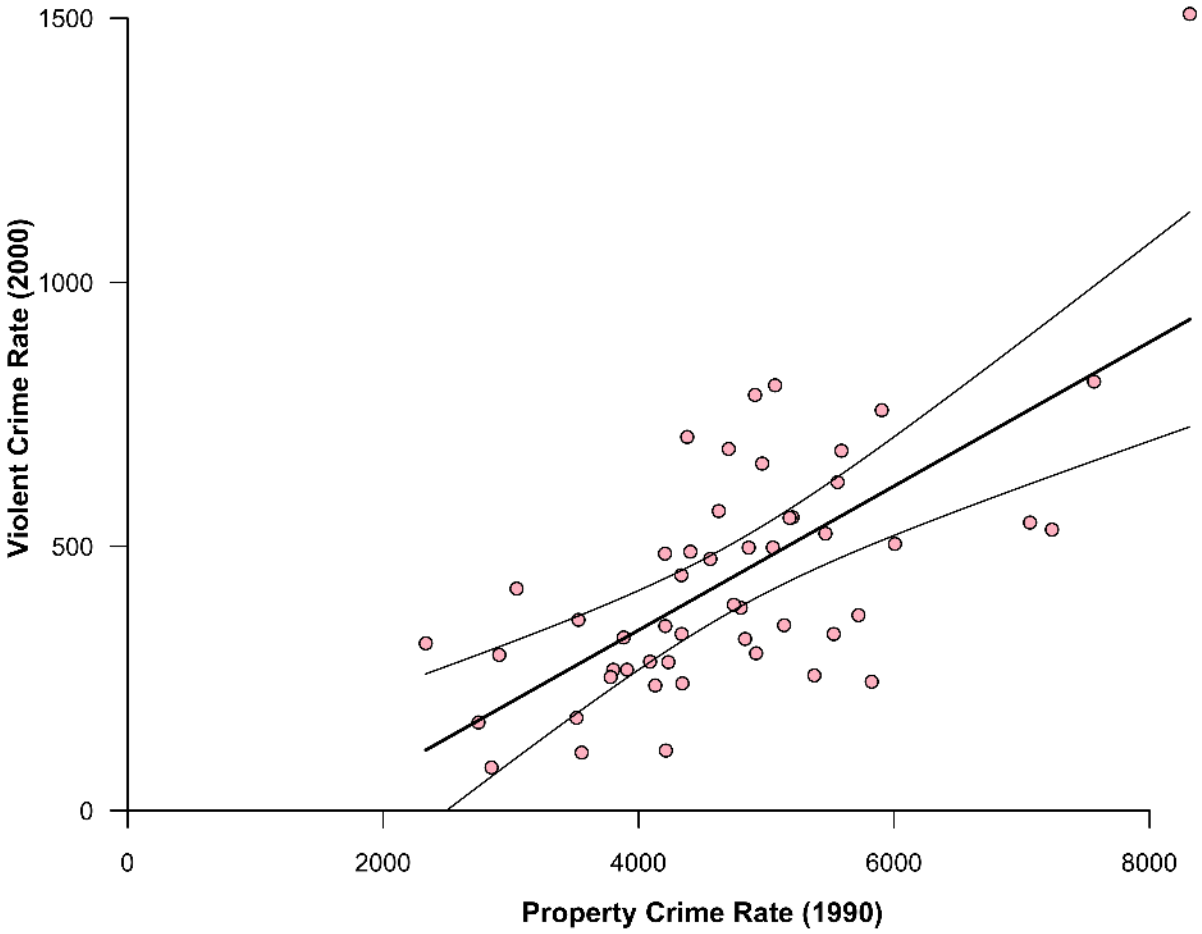


Figure 5.4.2: Graphic showing the relationship between the property crime rate in the United States in 1990 and the violent crime rate in 2000. The thick link is the line of best fit showing the relationship between these variables. The two hyperbolas are the 95% Working-Hotelling bounds for the line.

Here is the code to get the graphic:

```
library(KnoxStats)

dt = read.csv("http://rfs.kvasaheim.com/data/crime.csv")
summary(dt)
attach(dt)

mod = lm(vcrime00 ~ pcrime90)
summary(mod)

newX = seq(min(pcrime90),max(pcrime90))
ybounds = predictWH(mod, newdata=data.frame(pcrime90=newX))
yest = predict(mod, newdata=data.frame(pcrime90=newX))


par(mar=c(4,4,1,2), las=1)
par(font.lab=2, cex.lab=1, cex.axis=0.9)
par(xaxs="i", yaxs="i")

plot.new()
plot.window(xlim=c(0,9000), ylim=c(0,1600))

axis(1); axis(2)
title(xlab="Property Crime Rate (1990)", line=2.5)
title(ylab="Violent Crime Rate (2000)", line=2.75)

points(pcrime90,vcrime00, pch=21, bg="pink")
lines(newX,ybounds$lcb)
lines(newX,ybounds$ucb)
lines(newX,yest, lwd=2)
```

This page titled 5.4: The Working-Hotelling Bands is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

5.5: Conclusion

This chapter has served as a critical bridge, transforming the deterministic framework of Ordinary Least Squares (OLS) into a powerful engine for statistical inference. We began with the solid mathematical foundation established in the previous chapter — a structure built *entirely* upon our chosen definition of "best" as minimizing the sum of squared residuals. To this purely algebraic edifice, we introduced a single, pivotal probabilistic assumption:

$$\varepsilon_i \overset{\text{iid}}{\sim} N(0,\ \sigma^2)$$

This (overly) concise statement — that the unobserved error terms are independently and identically drawn from a Normal distribution with zero mean and constant variance — proved to be a keystone. From the marriage of our least-squares mathematics and this normality assumption, the entire inferential architecture of the chapter arose. The assumption endowed our previously abstract estimators with precise, known probability distributions: the regression coefficients b_i inherited a Normal distribution, while the quadratic form of the residuals led to a Chi-square distribution for the estimator of the error variance, $\hat{\sigma}^2$.

These derived distributions were the raw material from which we forged our tools of inquiry. By constructing what MATH 322 students will call "**pivotal quantities**" (but the rest of us will just call ratios where the parameters of interest were separated from their standard errors), we arrived at test statistics with fully known, tabulated distributions. Specifically, we saw how the t-distribution emerged naturally for testing hypotheses about individual regression parameters. (In the future, we will see something similar about the F-distribution for comparing nested models.) This logical progression from assumption to distribution to test statistic is the very core of classical statistical inference (a.k.a. "Fisherian statistics"). Once a test statistic's distribution is known under a null hypothesis, the gates to ~~hell~~ inference are opened wide: we can calculate p-values to weigh the evidence against the null, and we can construct confidence intervals to express the plausible range of a parameter with a specified degree of certainty.

A particularly elegant application of this framework was the development of the Working-Hotelling confidence bands for the entire regression line. Unlike an interval for a single predicted value, these bands provide a **simultaneous confidence region** for the mean response at *all levels* of the predictor variables (a.k.a. creating bounds on the *entire regression line*), illustrating how our probability statements can be extended to entire functions derived from the model.

The power of this process is undeniable, but its validity rests entirely on the foundation we built. The difficulty in statistics often lies not in the calculations themselves, but in satisfying the conditions required to ensure that a test statistic follows its presumed theoretical distribution. Here, the assumptions of independence, identical distribution (iid), homoskedasticity, and Normality were the indispensable keys that unlocked these known distributions. They allowed us to move from "this is our best estimate" to "this is *how confident* we can be in that estimate, and here is how we can test hypotheses about it."

This dependency, however, directs our attention to a crucial and unavoidable question:

What happens when these foundational assumptions are not met?

The reliability of every p-value and confidence interval we have constructed is conditional upon their truth. In the next chapter, we will therefore turn from application to diagnosis. We will develop methods — both graphical and numerical — to test these very assumptions. Our task will be to learn how to scrutinize the residuals (the proxies for our unobservable errors) to assess whether or not the assumptions are *reasonable*, given the data and our model. This process of model criticism is not merely a technical step; it is a fundamental safeguard in statistical inference, ensuring that the elegant machinery we have just built is applied to a world of data that conforms (at least approximately) to the reality our models assume.

This page titled 5.5: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

5.6: End-of-Chapter Materials

Here are the expected materials to supplement the chapter.

R Functions

We used no new R functions in this chapter, so there are none to cover here.

Exercises

1. Prove that the covariance between our b_1 estimator and $\bar{y}$ is 0.
2. Prove that the OLS estimators b_0 and MSE are independent.
3. Prove that the OLS estimators b_1 and MSE are independent.
4. Prove that the ratio $T = \frac{b_0 - \beta_0}{\sqrt{MSE\left(\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}\right)}}$ follows a Student's t distribution with $n - p$ degrees of freedom.
5. Prove that the ratio $T = \frac{\hat{y} - \hat{y}_0}{\sqrt{MSE\left(\frac{1}{n} + \frac{(x - \bar{x})^2}{S_{xx}}\right)}}$ follows a Student's t distribution with $n - p$ degrees of freedom.
6. Prove that the endpoints of a central confidence interval for β_0 are defined by $b_0 \pm t_{\alpha/2,n-p}\sqrt{MSE\left(\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}\right)}$.
7. Prove that the endpoints of a confidence interval for $\hat{y}$ are defined by $b_0 + b_1 x \pm t_{\alpha/2,n-p}\sqrt{MSE\left(\frac{1}{n} + \frac{(x-\bar{x})^2}{S_{xx}}\right)}$.
8. Prove that the endpoints of a prediction interval for y_{new} are defined by $b_0 + b_1 x \pm t_{\alpha/2,n-p}\sqrt{MSE\left(1 + \frac{1}{n} + \frac{(x-\bar{x})^2}{S_{xx}}\right)}$.

Theory Readings

- R. B. Bapat (2000). *Linear Algebra and Linear Models* (Second ed.). New York: Springer.
- William G. Cochrane (1934). "The distribution of quadratic forms in a normal system, with applications to the analysis of covariance." *Mathematical Proceedings of the Cambridge Philosophical Society*. 30(**2**): 178–191.
- Franklin A. Graybill and David C. Bowden (1968). "Linear Segment Confidence Bands for Simple Linear Models." *Journal of the American Statistical Association*, 62(**318**): 403–408.
- Henry Scheffé (1959). *The Analysis of Variance*. New York: Wiley.
- Holbrook Working and Harold Hotelling (1929). "Applications of the Theory of Error to the Interpretation of Trends." *Journal of the American Statistical Association*, 24(**165A**): 73–85.

This page titled 5.6: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

6: Dood! Check the Requirements



There are several requirements for ordinary least squares regression to be applicable. In this chapter, we cover them, their tests, and their relative importance. Focus on the relationship between the assumption/requirement and the tests used. As these assumptions (requirements) are used for some other fitting methods, not just ordinary least squares, the lessons learned here are helpful in the future.

However, realize that each fitting method has its own set of assumptions/requirements. For instance, median regression requires only the constant expected residual value. Maximum likelihood for Poisson regression requires that, *plus* a specific relationship between the expected value and variance.



In Chapters 3: Intro to Linear Regression and 4: Matrices and Linear Regression, we created the ordinary least squares regression technique. The mathematics of the situation required $\det \mathbf{X}'\mathbf{X} \neq 0$. Because this corresponds to the requirement that the design matrix $\mathbf{X}$ be of full column rank, the requirement is met when the independent variables are not linear combinations of each other.

Chapter 5: Improved! Now with Probabilities added the requirement

$$\varepsilon_i \overset{\text{iid}}{\sim} N\left(0,\ \sigma^2\right) \tag{6.1}$$

This allowed us to move beyond the pure mathematics of Chapters 3: Intro to Linear Regression and 4: Matrices and Linear Regression, and begin making **inferences** about the population based on the sample.

The requirement that $\varepsilon_i \overset{\text{iid}}{\sim} N\left(0,\ \sigma^2\right)$, equivalently $\mathbf{E} \sim N\left(\mathbf{0},\ \sigma^2\mathbf{I}\right)$, has several parts to it: Normality, constant expected value (of zero), constant variance, and independence. This chapter takes these assumptions and explores them. Both graphical and numeric tests are presented. Throughout it all, we will rely heavily on R to generate the residuals, create the graphics, and perform the tests.

Note

Some R functions require an additional package to be loaded. When such is the case, I indicate that in a special font. For example, the runs test (Bradley 1968) is implemented in the `lawstat` package as the function `runs.test`. The `lmtest` package has the `bptest`, which performs the Breusch-Pagan test.

> The `KnoxStats` package extends the `runs.test` to something more useful for us. That package also provides some other useful functions for regression analysis. Review The Appendix of R for how to load a package.

· · — · ✶ · — · ·

Chapter Sections

- 6.1: Normality
- 6.2: Functional Fit
- 6.3: Homoskedasticity
- 6.4: Multicollinearity
- 6.5: Conclusion
- 6.6: End-of-Chapter Materials

This page titled 6: Dood! Check the Requirements is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

6.1: Normality

Let us tackle the important requirement of normality in our linear models. As established in Chapter 5: Improved! Now with Probabilities, the assumption of normally distributed errors is what grants us the powerful ability to derive exact sampling distributions for our test statistics. This foundation is rather important — it allows for the precise calculation of p-values and the construction of reliable confidence intervals, forming the backbone of statistical inference. While the Central Limit Theorem often provides a safety net with large samples, rigorously checking this assumption is essential, especially with smaller datasets or when making precise probabilistic statements.

In this section, we will move from theory to practice, exploring both graphical diagnostics (like Q-Q plots and histograms) and formal statistical tests (Shapiro-Wilk) to assess whether your model's residuals meet this key assumption, ensuring the validity of your conclusions.

Learning Objectives

By the end of this section, you will be able to:

1. Explain why checking the normality assumption is important for the validity of exact inference (p-values and confidence intervals) in linear regression, particularly with smaller sample sizes.
2. Construct and interpret graphical diagnostics for normality, including histograms with overlaid normal curves and quantile-quantile (Q-Q) plots, to visually assess whether the model residuals appear to come from a normal distribution.
3. Conduct a formal Shapiro-Wilk test for normality on the model residuals, correctly interpreting the p-value as evidence against the null hypothesis that the residuals are normally distributed.
4. Describe, based on simulation evidence, how violations of the normality assumption affect the performance of hypothesis tests, noting that ordinary least squares is relatively robust to moderate skewness but fails when the error distribution has infinite variance (e.g., Cauchy).

✦•·····• ✦ •·····•✦

Graphical Checks

From your elementary statistics course, you were most likely introduced to a pair of graphical tests of Normality: Q-Q plots and histograms. In this section, we examine each when the residuals are generated from a Normal process and when they are not. To do this, let's do some experiments using R.

To begin, let us generate 100 residuals from a Normal process, with mean $\mu = 0$ and standard deviation $\sigma = 2$:

```
e = rnorm(100, m=0, s=2)
```

The `rnorm` function generates `n` random values from a Normal distribution with mean `m` and standard deviation `s`. Running this line only stores those 100 random values in the `e` variable.

Caution

Be aware that R, like most statistics programs, parameterizes the Normal distribution using the **standard deviation** instead of the variance.

With that one line, we have some "residuals" to play with that are generated "under the null hypothesis" (i.e., they are generated from the assumed Normal distribution). This will allow us to better understand **what the Normal distribution looks like**.

The Q-Q Plot

So, let us look at how to generate a Normal-based **quantile-quantile** (Q-Q) plot using R.

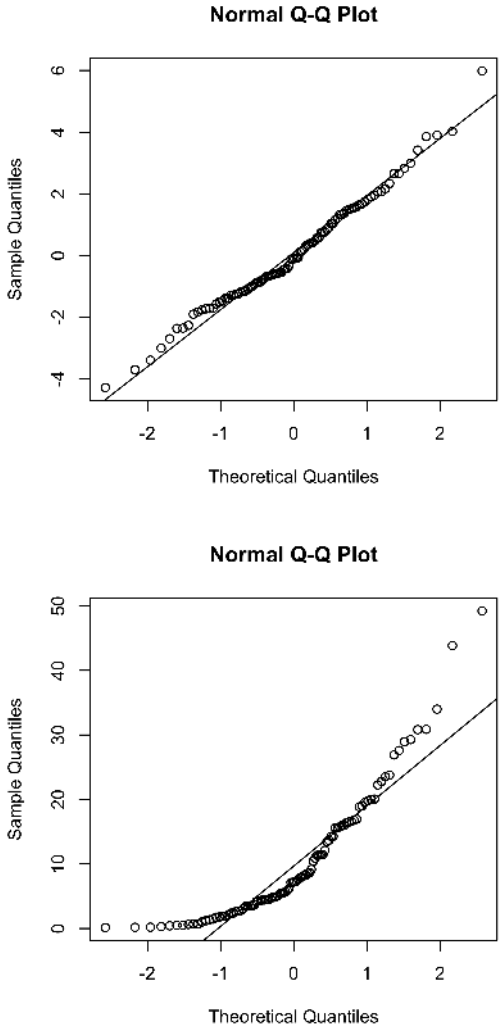


Figure **6.1.1**: Default Normal quantile-quantile plots The top panel is for the randomly-generated Normal data. Note that the circles fall close to the diagonal target line. The bottom panel is for the randomly-generated Exponential data. Note that the circles do *not* tend to fall close to the diagonal target line. There is a distinct bow to them, which signifies the residuals have a right (positive) skew.

```
qqnorm(e)
```

That's it. After running that one line, R creates the usual Q-Q plot for the Normal quantiles. It is a default graphic, so it does not look awesome, but it does get the point across to the statistician. One shortcoming of the default Q-Q plot in R is that it does not provide the diagonal line by default. You can add it by also running the command.

```
qqline(e)
```

Figure **6.1.1**, top panel, is the graphic produced by running these lines. Note that most of the circles cluster around the diagonal target line. Let us compare that graphic to a Q-Q plot from a non-Normal distribution. The non-Normal (and highly skewed)

distribution will be the Exponential($\lambda = 0.10$) distribution.

```
enon = rexp(100, rate=1/10)
qqnorm(enon)
qqline(enon)
```

Compare the two graphics in Figure 6.1.1. Note that the shape of the Q-Q plot from the Normal residuals (top) is very different from the one generated from the skewed distribution (bottom). This particular shape indicates that the distribution of the residuals is positively skewed (right skewed).

Those who are *particularly* skilled are reading Q-Q plots can also determine if the distribution is kurtotic.

Note

The above only looks at a single random sample from a Normal. It is dangerous to draw conclusions from just a single instance. It cannot tell you the inherent randomness of a Q-Q plot when showing a Normal. To see that, you need to generate many, many, many Normal samples and plot them. That will show how precise (or imprecise) the Q-Q plot is.

We do this next.

Example 6.1.1: Q-Q Plot Variability

Just how variable is the Q-Q plot?

Solution.

To figure this out, let us generate multiple samples from the same Normal distribution, then plot all of the samples on a single Q-Q plot. This will show us the expected spread.

```
set.seed(370)

x = rnorm(1e5, m=0, s=1)
y = matrix(x, ncol=100)

qqnorm(y[1,], xlim=c(-4,4), ylim=c(-4,4), pch=16, col=rgb(0,0,0,0.1))

for(i in 2:dim(y)[1]) {
  qq = qqnorm(y[i,], plot=FALSE)
  points(qq$x,qq$y, pch=16, col=rgb(0,0,0,0.1))
}

abline(a=0,b=1, col="white", lwd=3)
abline(a=0,b=1, col="red", lwd=1)
```

Note that I used an " `rgb` " color designation in the code. This rgb designation has four parameters. The first is the amount of red in the color; the second, green; third, blue. The last is the level of opacity. Me specifying an opacity of 0.1 means that it will take 10 dots on top of each other to show black. Why would I do this? It helps to show the density of the dots. Darker areas have a higher number of dots. It helps to see where "most" of the dots are, as opposed to "any" dots.

Run the above code to see all of the Q-Q plots (well, the dots defining the Q-Q plot). Note the amount of variability in the dots. Remember that all of the dots come from a Normal distribution... all of them.

> What lesson can you take away from this experiment?

Histogram

In the previous section, we examined quantile-quantile plots. We saw that a Q-Q plot with the dots aligning closely to the diagonal target line suggests Normality. In this section, we will use histograms to obtain a better view of the distribution of the data.

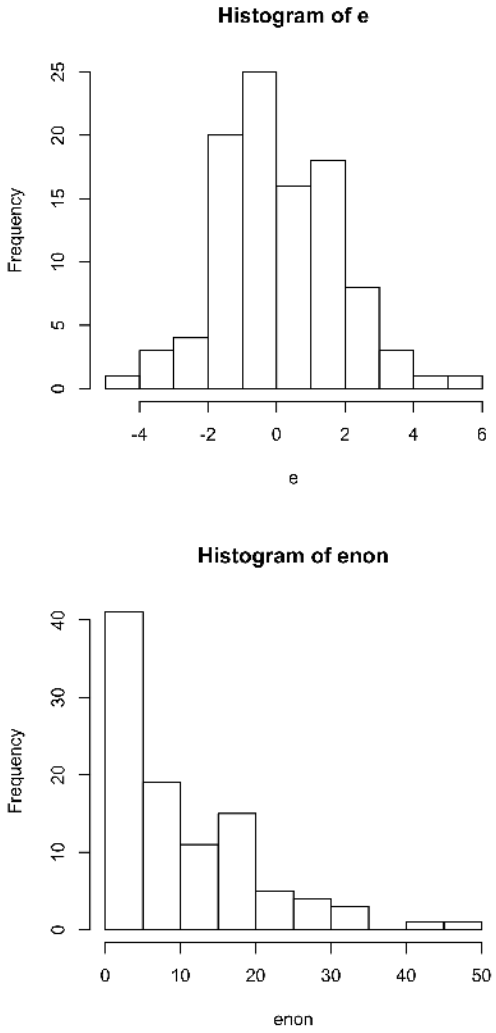


Figure **6.1.2**: Default histograms for the randomly-generated residuals. In the top panel, note the basic bell shape to the histogram. Such a shape suggests that the residuals come from a Normal distribution. In the bottom panel, note the lack of a bell shape to the histogram. Such a shape suggests that the residuals do not come from a Normal distribution.

Again, let us generate Normally-distributed residuals.

```
e = rnorm(100, m=0, s=2)
```

The function to create a basic histogram is just

```
hist(e)
```

or

```
overlay(e)
```

The former is just a basic, utilitarian histogram. The latter, from the `KnoxStats` package, overlays the histogram with a Normal curve.

This basic histogram produced is provided as in the top panel of Figure **6.1.2**. Note that it has the stereotypical "bell shape" to it, thus suggesting the data come from a Normal distribution. This is the shape you are seeking when using a histogram to explore the distribution of the residuals.

What does the histogram look like when the data come from a *highly skewed distribution* like the Exponential($\lambda = 0.10$) distribution above? See Figure **6.1.2**, bottom panel. Note the lack of bell shape in the top histogram (and enhanced in Figure **6.1.2**, below). In fact, one can easily see that the residuals are positively skewed. Recall that the direction of the skew is in the same direction as the long tail.

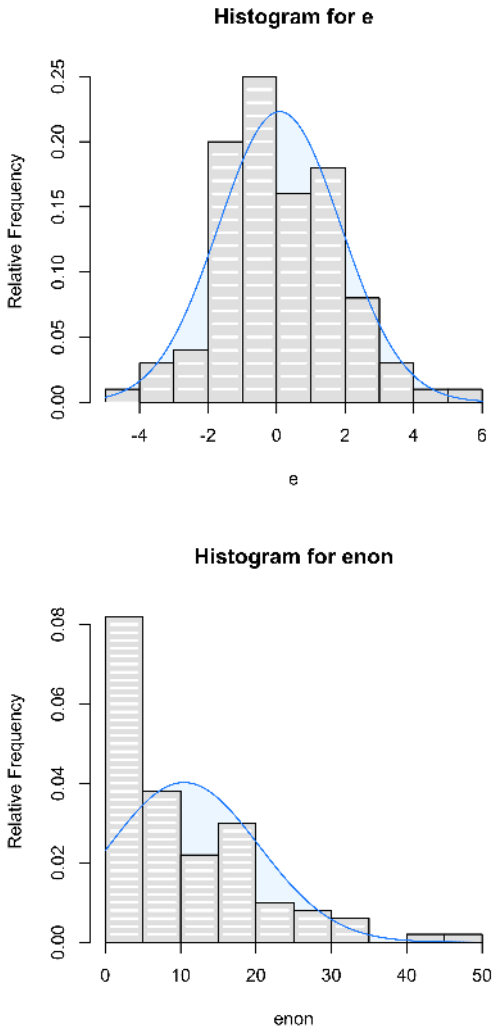


Figure **6.1.3**: Enhanced histograms for the randomly-generated residuals. These are the same data as in Figure **6.1.2**. The difference is that these graphics also overlay the Normal density function to aid in comparison.

Note

I find using the histogram much easier than using the Q-Q plot. I can "see" how the residuals are distributed in the histogram. In the Q-Q plot, I have to interpret much more, remembering what the different shapes indicate.

With that said, however, a Q-Q plot is much more useful than a histogram when there are few data points. So, if your sample size is small, you will want to use a Q-Q plot. Otherwise, a histogram may be best

Numeric Tests

Because of the importance of the Normal distribution in Statistics, there are several Normality tests available, with more being created yearly. I mean, we gotta award those PhD degrees for something, right? The reality is that it matters little which Normality test you use. As you will see below, the Normality assumption in OLS reflects the Normality assumption in t-tests and the like. The

Central Limit Theorem ensures that a sufficiently large sample size makes the distribution of the *data* largely irrelevant; the distribution of sample sums is "close" to Normal.

So, for me, I default to using the **Shapiro-Wilk test** as my Normality test. In base R, this test is implemented as the function `shapiro.test`. In the `KnoxStats` package, it is implemented as `shapiroTest`. I recommend the latter, because it adds some helpful functionality.

So, let us generate some Normally-distributed residuals and see how we can use the Shapiro-Wilk test.

Let us first set the random number seed. Why? Technically, there is no such thing as a truly random number when they are generated by a computer. This is because these pseudo-random numbers are just functions of a number called a **seed**. Think of a seed as being the name of an entire list of "random-looking" numbers. Thus, if we use the same seed, then our "random numbers" will be the same.

To see this, run the following three lines:

```
set.seed(370)
e = rnorm(100, m=0, s=2)
head(e, 5)
```

The numbers you get are

```
-0.5566842 -1.7264618  1.3320962 -0.5392740  0.3477423
```

Setting the random number seed ensured that your values are mine. Even more interesting: if you rerun those three lines, you will get the same "random" numbers.

> In research, why would one *want* to set the random number seed?

The `head` function returns the first six values in the variable, by default. A similar function is the `tail` function, which returns the *last* six values in the variable. I included the number 5 to have R only return five numbers.

To perform the Shapiro-Wilk test, just run the following line

```
shapiroTest(e)
```

The output you get will be

```
    Shapiro-Wilk Normality test

data:  e
W = 0.98554, p-value = 0.3472
```

The p-value is interpreted in the usual way: If the p-value is greater than your pre-determined value of α, then you fail to reject the null hypothesis. If the p-value is less than α, then you reject the null hypothesis. The null hypothesis in this case is that the data are generated from a Normal process (that the data come from a Normal distribution).

Note

The Normal distribution is identical to the Gaussian distribution. The only difference is the discipline. In much of statistics, it is known as the Normal distribution. However, when we get to generalized linear models in Chapter 14: Generalized Linear Models, this same distribution will be called the Gaussian distribution. In the Francophone world, this distribution is routinely called the Laplace-Gauss distribution.

Since the p-value is greater than $\alpha = 0.05$, we fail to reject the null hypothesis. We do not have sufficient evidence that the data are non-Normal.

Caution

We did *not* conclude that the data *do* come from a Normal distribution. We only concluded that there is not enough evidence to the contrary.

This point is **very important**. We first assume that the residuals are from a Normal distribution, then we test to see how reasonable that assumption is, given our data.

To drive home this point, let us generate our data from a *non*-Normal distribution and perform the Shapiro-Wilk test:

```
set.seed(370)
e = rt(100, df=100)
shapiroTest(e)
```

By the way that we generated the data, we *know* that the residuals do not come from a Normal distribution; they come from a Student's t distribution with 100 degrees of freedom ($\nu = 100$). Here is the output:

```
    Shapiro-Wilk Normality test

data:  y
W = 0.98979, p-value = 0.6474
```

Note that the p-value is greater than our α value of 0.05. As such, we again fail to reject the null hypothesis. There is not sufficient evidence that the residuals do not come from a Normal distribution.

However, we absolutely *know* they don't.

The Lesson

Never "accept" the null hypothesis. The p-value depends on how reasonable the null hypothesis is, *as well as* how good the test is and how large the sample is.

If we are good statisticians, then there are two possible reasons that we "fail to reject" the null hypothesis.

- The first is that the null hypothesis is actually true.
- The second is that there is not enough power in our test (the sample size is too small).

With our given set of data, there is *no* way to determine which of the two conditions is true.

Note

Remember that when making decisions, there are really three usual decisions: Yes, No, and Maybe. In statistics, since we are just testing how reasonable the null hypothesis is, we only have two possible decisions: **Reject** and **Don't Reject** the null hypothesis. These correspond to "the null hypothesis is wrong" and "the null hypothesis is right *or* we don't know."

Let us now generate severely non-Normal residuals and use the Shapiro-Wilk test to test if they do come from a Normal distribution:

```
set.seed(370)
e = rexp(100, rate=1/10)
shapiroTest(e)
```

Here are the results:

```
    Shapiro-Wilk Normality test

data:  y
W = 0.68002, p-value = 1.891e-13
```

Note that the p-value of $1.891 \times 10^{-13} = 0.000\ 000\ 000\ 000\ 189\ 1$ is much less than $\alpha = 0.05$. That strongly indicates that the residuals were not generated from a Normal process.

Note

Remember the Central Limit Theorem and the effect of sample size on the Normality of the sample sums. In the next section, we will explore the effects of non-Normality on our estimators and their distributions.

Why should I invoke the holy CLT in this context? Recall that the CLT speaks to the distribution of sums (and means) of independent random variables. As the sample size increases, the distribution of that *sum* approaches Normality.

Look at the formulas for b_0 and b_1. They both include the *sum* of y_i. Thus, it is the distribution of $\sum y_i$ that we really care about. If the y_i are from a Normal distribution, then this requirement is met. **However**, if the sample size is large enough, this condition is *also* met, as long as the data are from a distribution with a finite variance. In other words, the CLT rules the world.

How "large" is "large enough"?

Exploration of the Effects of Non-Normality

Mathematically speaking, if the Normality assumption does not hold, then nothing in Chapter 5: Improved! Now with Probabilities

is *absolutely* true for the model. **However**, the Central Limit Theorem tells us that a large sample size usually mitigates the effects of non-Normality in the residuals.

Let us explore that here...

An Appropriate Test?

Recall from your previous statistics course two main types of errors: Type I and Type II. A Type I Error happens when the null hypothesis is correct, but the test tells you to reject it. For a statistical test to be appropriate, the first requirement is that the Type I Error rate is equal (or sufficiently close to) the claimed α level. Thus, to determine if OLS is an appropriate test, let us examine the Type I Error rate to check that it is (close enough to) α.

This means we generate our data from the null hypothesis *while* meeting all of the requirements of the test.

Here is the script to generate the data once

```
set.seed(30)
beta0 = 3
beta1 = 0

x = 1:20
e = rnorm(20, m=0, s=1)

y = beta0 + beta1*x + e
```

This tests the null hypothesis that $\beta_1 = 0$:

```
model = lm(y~x)
summary(model)
```

The p-value corresponding to the test of our hypothesis is 0.3784. We can have R just give that number to us:

```
summary(model)[2,4]
```

That is *one* p-value.

If the test is entirely appropriate (for all values of α), we would reject our true null hypothesis about α of the time. This statement is equivalent to the statement that the p-values follow a standard Uniform distribution (see the section on multiple comparisons).

$$P \sim Unif(0,1) \tag{6.1.1}$$

Why?

If the test is appropriate, and if we reject when the p-value is less than α, what is the probability of rejecting? It should be α. Why?

> Why should the probability of committing a Type I error be α?

In statistical symbols, this is

$$P[P \leq \alpha] = \alpha \tag{6.1.2}$$

This is just the cumulative distribution function for the standard Uniform distribution. Thus, $P \sim Unif(0,1)$.

The code above gave us just one p-value. To investigate the *distribution* of p-values, you need to obtain thousands (or millions) of p-values. The easiest way to do this is to loop through the above steps and save the p-value from each test. That is what the following code does:

```
set.seed(30)

pval = numeric()

beta0 = 3
beta1 = 0
x = 1:20

for(i in 1:1e4) {
  e = rnorm(20, m=0, s=1)
  y = beta0 + beta1*x + e
  model = lm(y~x)
  pval[i] = summary(model)[2,4]
}
```

At the end of running this code, the variable `pval` contains 10,000 ($\texttt{1e4} = 1 \times 10^4$) observed p-values.

To determine if these p-values follow a standard Uniform distribution, we can create a histogram. When you do so, you note that it appears to closely follow a standard Uniform distribution, although not exactly. We would not expect it to follow the distribution *exactly*; those p-values are based on random samples and are therefore random values and the results will therefore be random.

Understanding the *meaning of the p-value* (above), we can check if the test is appropriate for $\alpha = 0.05$ by checking that the rejection rate is sufficiently close to 0.05. In other words, we can test if the proportion of p-values less than α is close enough to α. The Binomial test can accomplish this.

Note

The Binomial test is the exact test for checking that the rejection rate is equal to the claimed rate, 0.05. In your introductory course, you may have learned either the proportions test or the Wald test. Both are approximate tests that rely on the Normal approximation to the Binomial distribution

Here is the code to run the Binomial test:

```
binom.test( x=sum(pval<0.05), n=length(pval), p=0.05 )
```

The null hypothesis is that the proportion of rejections is equal to 0.05. The p-value of 0.9817 indicates that the claim that the Type I Error rate is 0.05 is *reasonable*; that is, the test seems reasonable. In other words, the test appears to be fine for $\alpha = 0.05$.

To determine if the test is appropriate for *all possible* α-values, we could repeat the above for every possible value of α. That would only take ∞ years. On the other hand, we can test all possible α values at once by recognizing that the p-values should follow a standard Uniform distribution and testing if they do. We can perform a statistical test to determine if the distribution of the observed p-values is sufficiently non-Uniform:

```
ks.test(pval, "punif")
```

This line performs the Kolmogorov-Smirnov test (Massey 1951). Its null hypothesis is that the observed values follow the distribution stated. Because the p-value is 0.4361, we fail to reject the null hypothesis that the p-values follow a standard Uniform distribution.

In other words: the test *appears to be* universally appropriate. That is, it seems to be appropriate for *any* value of α you select. Thus, we know that the test associated with testing the null hypothesis $\beta_1 = 0$ is appropriate for our usual value of α as well as for all values of α.

It is good to know that OLS works when all of the assumptions *are* met. What about when they are *not* met?

Caution

When running tests, you will tend to just look for the p-value and draw conclusions based on that one number. If you do, be very clear what that p-value measures. In the previous example, there were 10,002 different p-values. The first 10,000 were p-values for the test that $\beta_1 = 0$ for 10,000 different sets of data. The $10{,}001^{\text{th}}$ p-value determined if the distribution of those p-value was standard Uniform. The last p-value determined if the proportion of those p-values less than $\alpha = 0.05$ was 0.05.

There were 10,002 different tests in this example.

Non-Normal Residuals I

Now, what about if the Normality assumption is not met? What if the residuals follow an Exponential distribution? The following code generates 10,000 p-values where the residuals follow an Exponential distribution with $\lambda = 1$. Be able to compare it to the previous code listing and find the *one* difference.

```
set.seed(30)

pval = numeric()
beta0 = 3
beta1 = 0
x = 1:20

for(i in 1:1e4) {
  e = rexp(20, rate=1)
  y = beta0 + beta1*x + e
  model = lm(y~x)
  pval[i] = summary(model)[2,4]
}

binom.test( x=sum(pval<0.05), n=length(pval), p=0.05 )

ks.test(pval, "punif")
```

The **Binomial test** returns a p-value of 0.422, which suggests to us that the OLS test is appropriate for $\alpha = 0.05$. Furthermore, the **Kolmogorov-Smirnov test** returns a p-value of 0.2867. Thus, even if the residuals are skewed this much ($\gamma_1 = 2$), the tests arising

from the ordinary least squares estimation method appear universally appropriate.

> **Note**
>
> The sample size in these test is $n = 20$. This is much lower than the usual "rule of thum" of $n = 30$.

Also note that what we actually discovered is that OLS is *very robust* to violations of some of its requirements.

> Not all of them, just some.

Exploration helps us determine which and how violated the requirements must be before the tests give useless results.

Non-Normal Residuals II

Now, let's create a different skewed distribution for the residuals that is even more skewed: a Chi-square($\nu = 1$) distribution ($\gamma_1 = \sqrt{8} \approx 2.8$). Again, be able to compare it to the previous code listing and find the one difference.

```
set.seed(30)

pval = numeric()
beta0 = 3
beta1 = 0
x = 1:20

for(i in 1:1e4) {
  e = rchisq(20, df=1)
  y = beta0 + beta1*x + e
  model = lm(y~x)
  pval[i] = summary(model)[2,4]
}
```

The Kolmogorov-Smirnov test returns a p-value of 0.2667. Thus, even if the residuals are this skewed, the tests arising from OLS appear universally appropriate. The Binomial test returns a p-value of 0.0939, which tells us that the OLS test appears to be appropriate for $\alpha = 0.05$. Thus, even when the residuals follow *this* heavily skewed distribution, the conclusions based on our OLS tests seem to be appropriate for a sample size of $n = 20$.

Non-Normal Residuals III

Now, for fun, let's use a *symmetric* distribution for the residuals, but one where the CLT does not apply: the Cauchy distribution. Again, be able to compare it to the previous code listing and find the one difference.

```
set.seed(30)

pval = numeric()

beta0 = 3
beta1 = 0
```

```
x = 1:20

for(i in 1:1e4) {
  e = rcauchy(20)
  y = beta0 + beta1*x + e
  model = lm(y~x)
  pval[i] = summary(model)[2,4]
}
```

The Kolmogorov-Smirnov test returns a p-value of essentially 0. The Binomial test also returns a p-value of essentially 0. This tells us that the OLS test is not appropriate everywhere or even at $\alpha = 0.05$ when the residuals come from a Cauchy distribution.

Increasing the sample size, which usually fixes the issue, will not fix this issue, either:

```
set.seed(30)

pval = numeric()
beta0 = 3
beta1 = 0
x = 1:5000

for(i in 1:1e4) {
  e = rcauchy(5000)
  y = beta0 + beta1*x + e
  model = lm(y~x)
  pval[i] = summary(model)[2,4]
}
```

In this code, the sample size is 5000. The conclusions are the same.

To see what is happening, create the histogram of the p-values and examine it. You should see that the first bar is much smaller than the others. This means the OLS tests reject at a lower rate than 0.05.

Be Aware!

When the underlying distribution does not have a finite variance, such as the Cauchy distribution, the Central Limit Theorem does *not* apply. That means increasing the sample size has absolutely no effect on the Normality of the sums.

Such sample sums are *never* Normally distributed.

This page titled 6.1: Normality is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

6.2: Functional Fit

A second requirement of ordinary least squares (OLS) is that the expected value of the residuals is constant (*and zero*). From the gut, this means that the residuals randomly bounce above and below our estimates (regression curve). If the residuals are above (or below) our estimates more frequently than expected, then the curve should be moved up (or down) to provide a "better" fit.

We used this requirement in many places in prior chapters. This requirement is entirely equivalent to the assumption that the underlying model (expected/predicted values) consistently fits the data, that there is no systematic error.

Learning Objectives

By the end of this section, you will be able to:

1. Explain that the requirement of a constant expected residual ($E[\varepsilon_i] = 0$) is equivalent to assuming the model's functional form is correctly specified, meaning there is no systematic error or missing pattern in the fit.
2. Construct and interpret a residual plot (residuals versus an independent variable) as the primary graphical tool to check for misspecification, identifying patterns such as curvature that indicate the need for a transformation or additional term (e.g., x^2).
3. Conduct a runs test on the residuals ordered by an independent variable to numerically assess whether the residuals show a systematic pattern, interpreting the p-value as evidence for or against the null hypothesis of correct model specification.
4. Describe the mathematical consequence of violating this assumption: the OLS slope estimator b_1 becomes biased, with the bias term depending on the covariance between the independent variable and the expected value of the errors.

✦•·····• ✦ •·····•✦

Be aware, however, that the mathematics behind OLS will force the average of the residuals to be zero. This means two things. First, the OLS model is "**self-correcting**," in that the "line of best fit" will provide the best linear fit to the data. Second, the OLS model ensures that it is impossible to detect a systematic error in the measurements.

Graphical Check

Graph the residuals against each of the independent variables. Look for non-linear patterns in the plot (parabolas, cubics, etc.). If such exists, your model is misspecified. A fix is to transform the independent variable to eliminate that pattern. This is one place where the graphical "test" is superior to the numeric test. If you can identify the pattern, you have the fix. *For instance*, if the residuals have the pattern in Figure 6.2.1, then the solution may be to use x^2 in place of (or in addition to) x. To see this, run the following code to obtain Figure 6.2.1.

```
set.seed(370)

x = seq(0,3,length=20)
n = length(x)
e = rnorm(n)
y = 4 + 2*x^2 + e

mod = lm(y ~ x)
```

```
E = residuals(mod)

plot(x, E)                ## residuals plot
```

Note the strong quadratic shape to this **residuals plot** (Figure **6.2.1**). This strongly suggests that the model is misspecified.

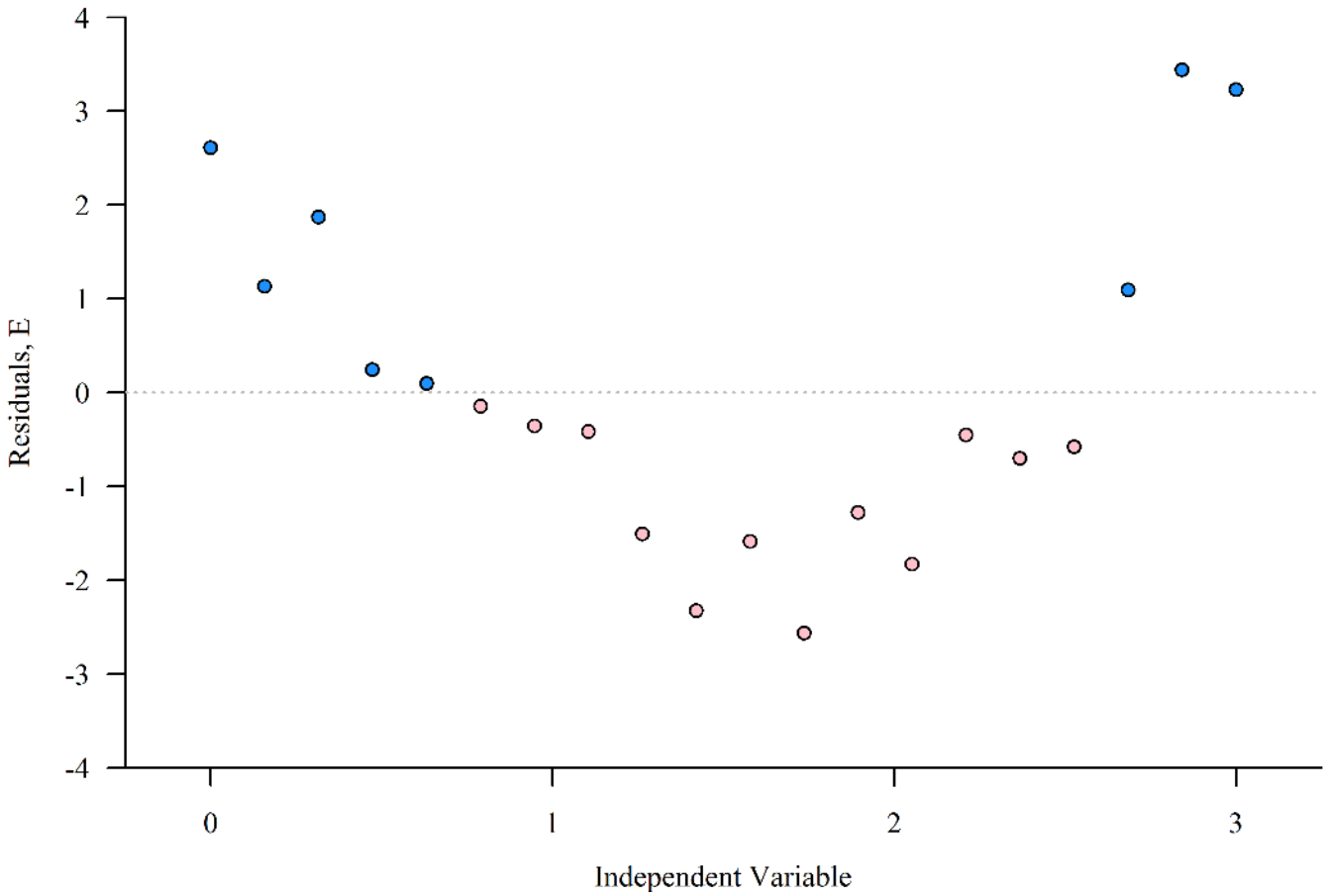


Figure 6.2.1: A **residuals plot** for a misspecified model. Note that the residuals show a definite quadratic form to them. Fixing this issue may be as simple as including x^2 as an additional independent variable.

Note

As an aside, note that there are only three "runs" in the residuals. From the discussion of the runs test, a run is a sequence of values on one side of the prediction line. According to the above figure, the first run consists of the first 5 values; the second, the next 12 values; and the third, the last 3 values.

Since the number of runs is based on the Binomial distribution, we can calculate the probability of observing this number of runs under the null hypothesis. Thus, we can calculate a p-value for the hypothesis that the model is properly specified.

This example clearly shows that the model is misspecified. There is still more information contained in the residuals. It would be wrong to ignore that information.

Since the residuals plot has a prominent *quadratic* shape, a solution could be to include x^2 in the model:

```
x2 = x^2
mod = lm(y ~ x + x2)
E2 = residuals(mod)

plot(x, E2)                ## residuals plot
```

With this change, we re-examine the residuals plot — one residuals plot for each independent variable. Since we have two (one?), we need to examine two (one?) residuals plots (Figure **6.2.2**). Note the transformation was successful. Neither plot shows anything other than what appears to be random bouncing across the line $y = 0$.

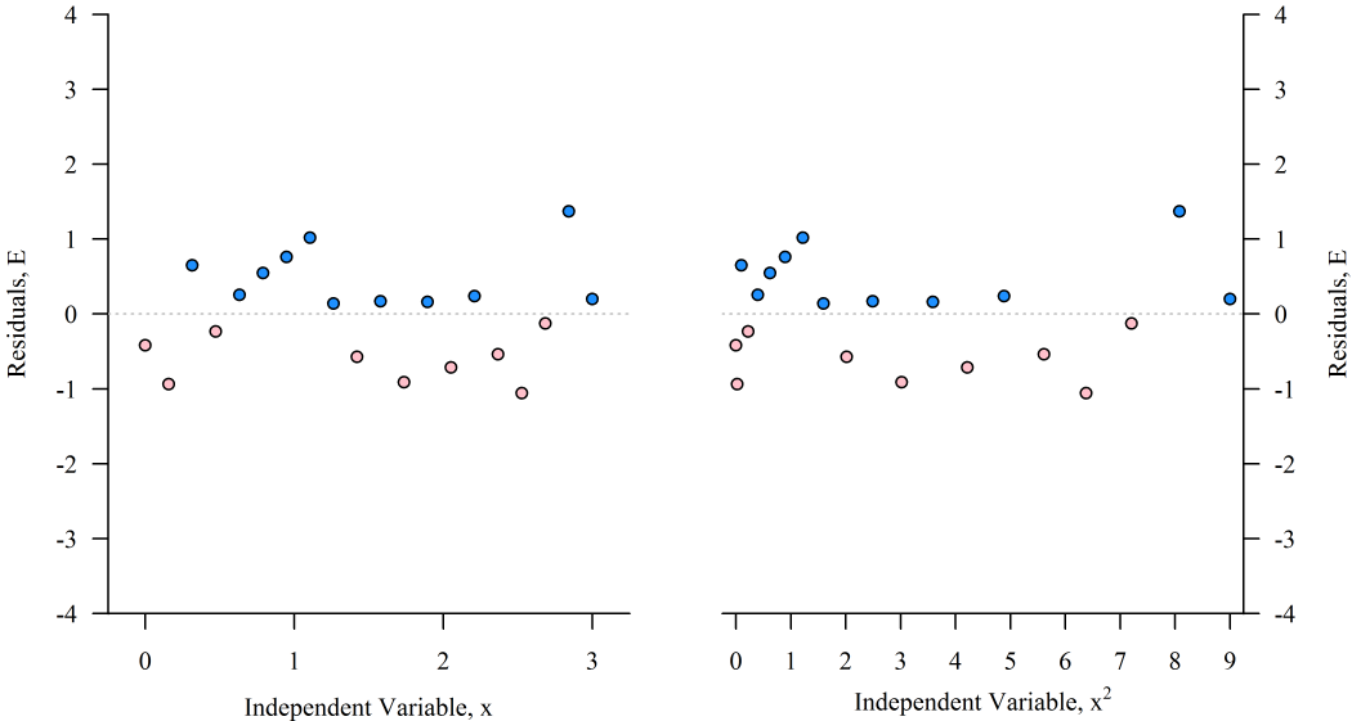


Figure 6.2.2: Residuals plots for the properly specified model. There is one residuals plot per independent variable. Note that the residuals in neither plot suggest anything other than a lack of pattern.

Caution

There is a habit to feel sad that some requirement is not met by the model/data, such as above. However, do not feel sad. Feel happy, because you have learned something new about the relationships in the data! We know more!

Celebrate!

A Numeric Test

That last sentence leads us to a numeric test. Compare the plots of Figure **6.2.1** and **6.2.2**. The residual is colored blue if it is positive; pink, otherwise. In Figure **6.2.1**, there are long unbroken streaks (runs) of blue and pink. In Figure **6.2.1**, on the other hand, the length of those runs is much reduced *and* their number is increased.

The test suggested by the above is not-surprisingly called the **runs test**. It is implemented in the `lawstat` package as the function `runs.test`. It takes just one piece of information: the residuals in the order of the independent variable. This test is also implemented in the `randtests` and `snpar` packages, as well as the `KnoxStats` package, which is what I use in this book.

The **null hypothesis** of the runs test is that the succeeding values are independent and have mean **0**.

Here, I demonstrate the `runs.test` function in the `KnoxStats` package.

```
library(KnoxStats)
set.seed(370)

x = runif(100)
e = rnorm(100)

runs.test(e, order=x)      ## The runs test
```

In this version of the `runs.test` function, the first slot goes to the residuals, and the second slot goes to the independent variable.

The output of this code is

```
        Runs Test - Two sided

data:  e, as ordered by x
Standardized Runs Statistic = 1.6081, p-value = 0.1078
```

As usual, check the p-value. Since the p-value of **0.1078** is greater than the α level of **0.05**, we fail to reject the null hypothesis that the expected value of the residuals is constant and zero. Thus, since the p-value is greater than α, the model passes this test.

Exploration of the Effects of Non-Constant Expected Value

To see the effect of a non-constant expected value *mathematically*, let us revisit one of the proofs from Section 3.3: Our First Assumptions.

What is the expected value of b_1?

Proof.

$$E[b_1] = E\left[\frac{\sum_{i=1}^{n}(x_i - \bar{x})(Y_i - \overline{Y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right] \tag{6.2.1}$$

$$= E\left[\frac{\sum_{i=1}^{n}(x_i - \bar{x})Y_i}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right] \tag{6.2.2}$$

$$= \frac{\sum_{i=1}^{n}(x_i - \bar{x})E[Y_i]}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{6.2.3}$$

$$= \frac{\sum_{i=1}^{n}(x_i - \bar{x})(\beta_0 + \beta_1 x_i + E[\varepsilon_i])}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{6.2.4}$$

$$= \beta_0 \frac{\sum_{i=1}^{n}(x_i - \bar{x})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} + \beta_1 \frac{\sum_{i=1}^{n}(x_i - \bar{x})x_i}{\sum_{i=1}^{n}(x_i - \bar{x})^2} + \frac{\sum_{i=1}^{n}(x_i - \bar{x})E[\varepsilon_i]}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{6.2.5}$$

$$= \beta_0 \frac{0}{\sum_{i=1}^{n}(x_i - \bar{x})^2} + \beta_1 \frac{\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} + \frac{\sum_{i=1}^{n}(x_i - \bar{x})E[\varepsilon_i]}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{6.2.6}$$

$$= \beta_1 + \frac{\sum_{i=1}^{n}(x_i - \bar{x})E[\varepsilon_i]}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{6.2.7}$$

Now, this last line is β_1 ***if*** ε is independent of x. So, if the expected value is constant zero, the OLS estimator of β_1 is clearly unbiased. I leave it as an exercise to show that if it is constant, but *non-zero*, then the OLS estimator of β_1 remains unbiased.

If, however, the assumption of a constant expected residual value is violated, the OLS estimate of β_1 is **biased**. This is not a good thing. It means that your predictions are wrong... even "on average."

But, what about the OLS estimator of β_0, the y-intercept? What effect does a non-zero expected residual value have on it? To see, let us revisit the proof of the unbiasedness of b_0.

$$E[b_0] = E\left[\overline{Y} - \bar{x}b_1\right] \tag{6.2.8}$$

$$= E\left[\overline{Y}\right] - \bar{x}\,E[b_1] \tag{6.2.9}$$

$$= (\beta_0 + \bar{x}\beta_1 + E[\varepsilon_i]) - \bar{x}\left(\beta_1 + \frac{\sum_{i=1}^{n}(x_i - \bar{x})E[\varepsilon_i]}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right) \tag{6.2.10}$$

$$E[b_0] = \beta_0 + E[\varepsilon_i] - \bar{x}\frac{\sum_{i=1}^{n}(x_i - \bar{x})E[\varepsilon_i]}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{6.2.11}$$

Thus, there are *two* places that the non-constant expected value ("poor model fit") affects the OLS estimator of β_0. The result above shows:

- If $E[\varepsilon_i] = 0$, the expected value of the errors is zero, then b_0 is unbiased for β_0.
- *However*, if $E[\varepsilon_i]$ is constant, but non-zero, then $E[b_0] = \beta_0 + E[\varepsilon_i]$.
- *And*, if $E[\varepsilon_i]$ is a function of x, then $E[b_0]$ is $\beta_0 + E[\varepsilon_i]$ plus some function of x.

I leave it as an exercise for you to show that $E[b_0] = \beta_0 + E[\varepsilon_i]$ if $\bar{x} = 0$, *regardless* of whether the residuals are correlated with the independent variable. This is yet *another* reason some disciplines center their data before analyzing it.

Caution

The actual *assumption* is that the expected value of the residual is constantly zero. Again, because of the mathematics of OLS, the mean residual is *guaranteed* to be zero. So, there is no way to test if the expected value of the residuals is constantly *zero*, only that it is constant

Of all assumptions/requirements, this is the **most important** to meet. If your residuals depend on the value of x, then both the b_0 and b_1 estimators are biased. If the expected value of the residuals is not zero, then just the b_0 estimator is biased.

It is even worse. Because OLS mathematically forces $\bar{e} = 0$, one cannot test if the expected value of the residuals *really* is zero. One must rely on the assumption that the data were collected without systematic error. That is, the statistician must trust the scientist to measure things correctly.

Question

When would the residuals be a function of the residuals?

This is an **excellent question** that you need to grapple with. It fundamentally means that you are missing an important variable from your model. Perhaps that variable is not an independent variable. Perhaps it is a *confounding* variable. Perhaps it is another *dependent* variable, which requires multivariate regression (beyond the scope of this book). It definitely means your model is weak and should be rejected as incomplete (if possible).

This page titled 6.2: Functional Fit is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

6.3: Homoskedasticity

Of the core assumptions underpinning classical linear regression, the final one we will examine is the requirement that the model's residuals exhibit homoskedasticity — that is, they possess a constant variance, denoted as σ^2, across all levels of the independent variables. This assumption was foundational to the derivations in Section 3.3: Our First Assumptions. Every instance where the term σ^2 appeared, we implicitly relied on it being a single, stable parameter, not a variable σ_i^2 that changes with each observation. When this assumption is violated — a condition known as heteroskedasticity, where the error variance systematically changes with the predictors — the standard machinery of inference breaks down. The ordinary least squares estimators, while still unbiased, no longer have the minimum variance among linear unbiased estimators.

More critically, the estimated standard errors of the coefficients become *biased*. Consequently, the test statistics (such as t-statistics and F-statistics) lose their validity, and the confidence intervals constructed using these flawed standard errors will be **incorrect**, leading to potentially misleading conclusions about the significance of the predictors. Therefore, diagnosing and addressing heteroskedasticity is not a mere technicality but a crucial step in ensuring the reliability of hypothesis tests and the accuracy of the uncertainty estimates we place on our model parameters.

Learning Objectives

By the end of this section, you will be able to:

1. Explain the assumption of homoskedasticity (constant variance of errors) and why its violation (heteroskedasticity) leads to biased standard errors, invalid t-tests, and incorrect confidence intervals.
2. Construct and interpret a residual plot to visually diagnose heteroskedasticity, identifying common patterns such as a trumpet or funnel shape that indicate non-constant variance.
3. Conduct a formal Breusch-Pagan test to numerically assess the presence of heteroskedasticity, correctly interpreting the p-value as evidence against the null hypothesis of constant variance.
4. Describe, based on simulation evidence, how different patterns of heteroskedasticity affect the coverage rates of confidence intervals for the regression coefficients and for predictions.
5. Explain the purpose of Huber-White heteroskedasticity-consistent (sandwich) estimators as a method for obtaining valid standard errors and inference when heteroskedasticity is present, noting that they adjust the standard errors but not the coefficient estimates.

✦•·····• ✦ •·····•✦

A note on vocabulary. If the variance of the residuals is constant, we claim the residuals are "**homo**skedastic." Otherwise, they are "**hetero**skedastic." Recall that "homo" is a ancient Greek combining form meaning "same." Similarly, "hetero" means different, and "**skedastic**" means scatter.

Graphical Check

The graphical test is very similar to that for checking constant expected value. In *that* assumption, a residual plot was created. The middle of the vertical spread was traced out and checked to see if it was always near the zero line. For testing **constant variance**, a residual plot can also be used. The vertical spread of the data is traced out and checked that it does not vary too much. Note that the "vertical spread" is not the range, but the middle portion that contains about two-thirds of the data.

There are three stereotypical shapes that suggest heteroskedasticity. These three shapes, along with a shape suggesting homoskedasticity, are presented in the figure below.

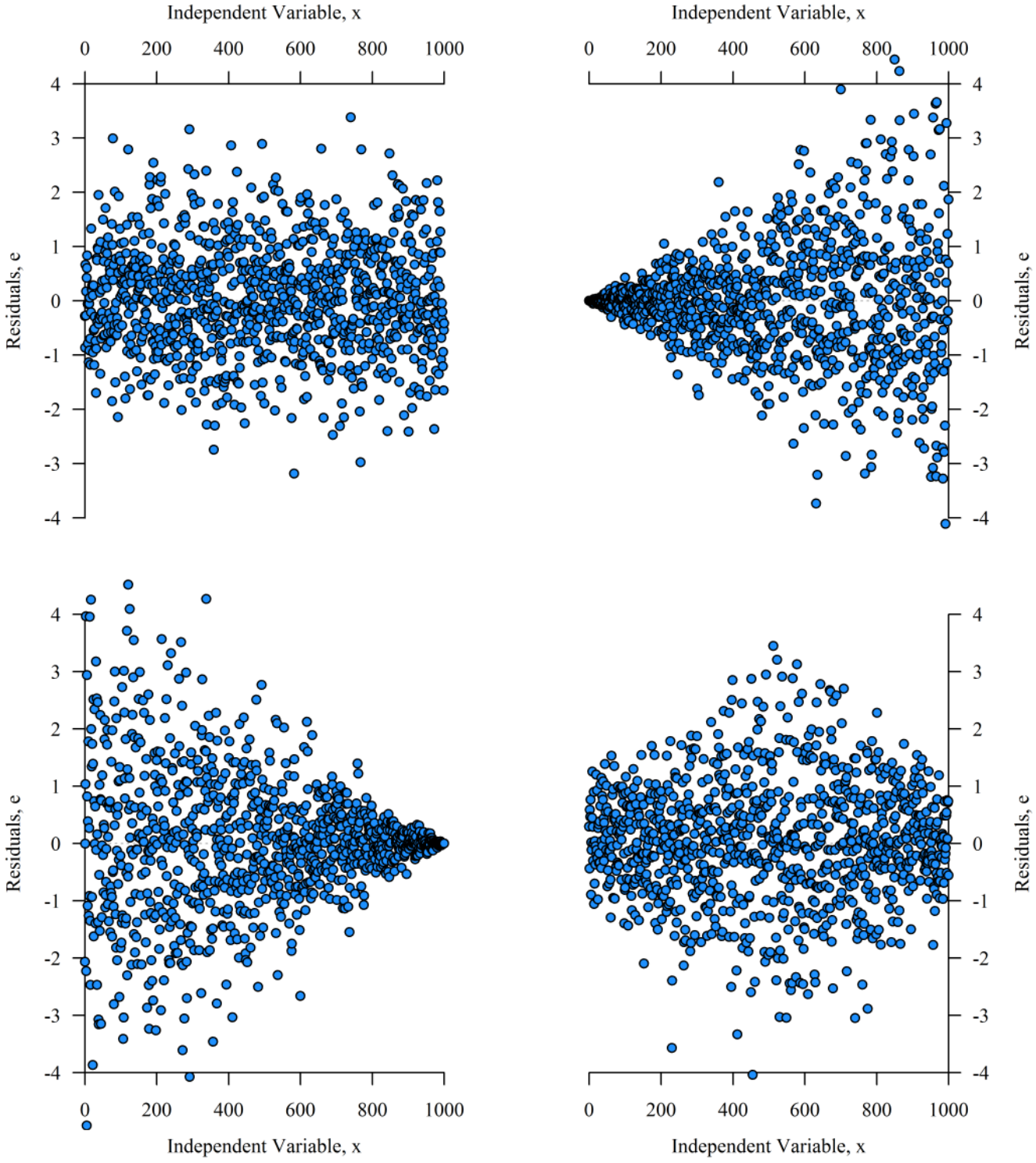


Figure **6.3.1**: Residual plots illustrating homoskedasticity (top-left) and three typical types of heteroskedasticity: **trumpet** (top-right), **funnel** (bottom-left), and **bulge** (bottom-right).

https://stats.libretexts.org/@go/page/57733

Numeric Test

In addition to what your eyes may tell you, it may be better to *also* perform a numeric test of homoskedasticity.

> **Note**
>
> I recommend performing both, when possible. If the p-value on the numeric test is too low, then the graphical test either gives you clues on where the problem is, or that the problem is practically minor and can be ignored.

There are a couple of tests that can be used. However, the usual test used in the regression area is the **Breusch-Pagan test**.

The way that the Breusch-Pagan test works is to refit the model including higher-order terms of the independent variables and compare the *ratio of the two* SSE values to a specific distribution. Since the null hypothesis is homoskedasticity (constant variance), a ratio close to one (large p-value) indicates that the model passes this test (the higher-order terms add little to the predictive ability of the model). A sufficiently *small* p-value indicates the presence of heteroskedasticity (those larger-order terms should be included to the model), which means the model needs improvement.

So, to **examine** the Breusch-Pagan test, let us generate our data, first according to our assumptions, then in violation of them. The Breusch-Pagan test is contained in the `lmtest` package as the function `bptest` .

```
set.seed(30)

x = seq(0,100)
n = length(x)
e = rnorm(n, m=0, s=1)
y = 3 + 2*x + e

plot(x,y, pch=21, bg="dodgerblue")
```

The scatter plot of this data does *not* seem to suggest a changing variation in the residuals. Let us perform the Breusch-Pagan test to see if this numeric test also fails to detect a problem with the constant-variance assumption.

```
mod = lm(y~x)
bptest(mod)
```

If you get

`Error: could not find function "bptest"`

then you need to load (or install and load) the `lmtest` package.

Once you do, your output should look exactly like this

```
        studentized Breusch-Pagan test

data:  mod
BP = 0.23429, df = 1, p-value = 0.6284
```

Because the p-value is *greater than* α, we cannot reject the null hypothesis of homoskedasticity. The model passes the test. ✅

Let us look at what happens when the residuals *are* heteroskedastic.

```
set.seed(30)
x = seq(0,100)
n = length(x)
e = rnorm(n, m=0, s=sqrt(x))
y = 3 + 2*x + e

plot(x,y, pch=21,bg="dodgerblue")
```

The scatter plot of this data definitely suggests increasing variation in the residuals, a trumpet shape that indicates heteroskedasticity. Let us perform the Breusch-Pagan test to see if this numeric test also detects a problem with the constant-variance assumption.

```
mod = lm(y~x)
bptest(mod)
```

Your output should look like this

```
        studentized Breusch-Pagan test
data:  mod
BP = 8.0992, df = 1, p-value = 0.004428
```

Because the p-value is *less than* α, we *reject* the null hypothesis of homoskedasticity. The model does *not* pass the test. ❌

Note

Recall from your previous statistics course that a true null hypothesis will be rejected α proportion of the time and a false null hypothesis will be wrongly accepted β proportion of the time, where α is the Type I error rate and β is the Type II error rate.

Neither of these numbers can be zero without making the other 1. So, **be aware that you may be wrong**.

Exploration of the Effects of Non-Constant Variance (coverage)

Instead of looking at proofs or the distribution of p-values, let us generate data and look at the confidence intervals estimated using OLS. While it is easy to see that there *is* an effect using proofs, this method may make it easier to see *how serious* those effects are.

The steps to estimate coverage are as follows:

1. generate heteroskedastic data
2. calculate the 95% confidence intervals
3. check if the true population parameters β_0 and β_1 are in the intervals
4. repeat this many, many, many times

The first three steps check the coverage of a single random sample. It is step 4 that allows you to better estimate the coverage rate.

If the coverage of the estimated intervals are close to the claimed 95%, then the effect of heteroskedasticity is minor.

Here is the entire code

```
set.seed(30)

b0covered = numeric()
b1covered = numeric()

for(i in 1:1e4) {
  x = seq(0, 10, 0.5)
  n = length(x)
  e = rnorm(n, m=0, s=sqrt(x))
  y = 3 + 2*x + e
  mod = lm(y ~ x)
  ci = confint(mod)
  b0covered[i] = (3>ci[1,1]) && (3<ci[1,2])
  b1covered[i] = (2>ci[2,1]) && (2<ci[2,2])
}
```

Running this may take anywhere from a few seconds to a minute or more, depending on your computer. At the end, you will have two variables, `b0covered` and `b1covered`, each containing 10,000 Boolean values. A `TRUE` indicates the population parameter was covered by the confidence interval for that random sample; a `FALSE` indicates it was *not*. The proportion of `TRUE` values can quickly be calculated using `mean(b0covered)` and `mean(b1covered)`.

In this particular example, the proportion of times the β_0 parameter was **covered** by the confidence interval was **0.996**. Thus, the estimated coverage for β_0 when the data are heteroskedastic *in this manner* is **99.6**. The coverage rate for β_1 is about **95.1**. Both *should* be close to $1-\alpha$, 95%.

From these results, we know a couple of things.

1. First, this amount of heteroskedasticity did not (from a practical standpoint) significantly affect the confidence intervals for the slope parameter, β_1.
2. Second, it *did* (from a practical standpoint) significantly affect it for the intercept parameter.

In fact, because the reported confidence interval is much *wider* that it should have been, a researcher will reject too **in**frequently. This means the power of the test is too low.

Let us try a greater level of heteroskedasticity. And, let us make it *funnel* instead of the above trumpet:

```
set.seed(30)
b0covered = b1covered = numeric()

for(i in 1:1e4) {
  x = seq(0,10,0.5)
  n = length(x)
```

```
  e = rnorm(n, m=0, s=15-x)
  y = 3 + 2*x + e
  mod = lm(y~x)
  ci = confint(mod)
  b0covered[i] = (3>ci[1,1]) && (3<ci[1,2])
  b1covered[i] = (2>ci[2,1]) && (2<ci[2,2])
}
```

Note what changed: the standard deviation of the residuals. It starts high and gets smaller — a funnel shape.

This scheme produces a coverage estimate of 88.8% for the intercept parameter and 93.9% for the slope parameter. Thus, the confidence interval for the intercept is again affected more than that of the slope parameter. In fact, the slope parameter is relatively unchanged.

Let us try once more. This time, let us look at **bulge** heteroskedasticity.

```
set.seed(30)
b0covered = b1covered = numeric()

for(i in 1:1e4) {
  x = seq(0,10,0.5)
  n = length(x)
  e = rnorm(n, m=0, s=11-2*abs(x-5))
  y = 3 + 2*x + e
  mod = lm(y~x)
  ci = confint(mod)
  b0covered[i] = (3>ci[1,1]) && (3<ci[1,2])
  b1covered[i] = (2>ci[2,1]) && (2<ci[2,2])
}
```

Again, note the common code and the **slight changes**. Here, the variance starts low (1), gets higher (11), then decreases again to 1 — a bulge heteroskedasticity.

The coverage for both parameters are far from 95%. For β_0, the coverage is 99.3%; for β_1, 99.8%. Both indicate that the researcher will reject the hypotheses **far too infrequently**.

Type I and Type II Trade-off

The test statistics for β_0 and β_1 are calculated using the MSE. That test statistic, if all of the assumptions are true, follows the Student's t distribution with $n-p$ degrees of freedom. However, if the variance is a function of the independent variable, the distribution of the test statistic is also a function of the independent variable. This is a problem, because the "correct" p-values and confidence intervals would *also* be functions of x

The analysis of the effect of heteroskdasticity can be repeated for the distribution of test statistics. **Or**, we can realize that we reject the null hypothesis when the confidence interval does not contain our hypothesized value. That is, the analysis for the confidence interval is sufficient for our understanding of the p-value:

1. If the confidence interval is too wide, then the p-values will be too high (reject the null hypothesis too infrequently).

2. If the confidence intervals are too narrow, then the p-values are too low (reject too frequently).

Of the two possibilities, the first (which produces a higher Type I error rate) **may be better** from the researcher's point of view in some cases: While the researcher would end up rejecting too infrequently, those rejections are more sure because the true p-value is less than the observed p-value. One may prefer this when it worse to commit a Type I error (rejecting a true null hypothesis).

On the other hand, however, the second *may be better* in certain circumstances. It will have a lower Type II error rate. If it is more important to protect against a Type II error, then this will be the better scenario.

Note

These findings regarding the hypothesis tests about $\mathbf{B}$ *strictly* relate only to these particular three types of heteroskedasticity at these three particular levels. If you find your heteroskedasticity is much higher than those used in these examples, you should run a coverage analysis similar to these to see how bad the heteroskedasticity really affects the confidence intervals.

The above analysis examined the effects of heteroskedasticity on the parameter estimates. It did *not* examine its effect on the prediction of y-values. Those effects are *much more pronounced* and are also a function of the variance *at the x-value*.

To see this in one example, let us generate trumpet-shaped heteroskedastic residuals, predict the value of $\boldsymbol{y}$ at three points along the x-axis, and determine how frequently those points are covered by the calculated confidence intervals.

Before we do this, let us see if we can determine what to expect.

- For low values of x, there is very little true variation in the predicted value. Thus, we would expect the predicted value to fall in the calculated confidence interval very frequently.
- For middling values of x, the true and estimated variances are close to each other. Thus, we would expect coverage to be close to the nominal 95\%.
- For large values of x, the true variation is much higher than the estimated variation. Thus, a lot of the predicted y-values should fall outside the confidence interval. We should expect the coverage to be relatively low.

Let's see if these expectations are met by reality.

Here is the entire code

```
set.seed(30)

y1covered = numeric()
y2covered = numeric()
y3covered = numeric()

for(i in 1:1e4) {
  x = seq(0,10,0.5)
  n = length(x)
  e = rnorm(n, m=0, s=sqrt(x))
  y = 3 + 2*x + e

  mod = lm(y~x)
```

```
  ypr = predict(mod, newdata=data.frame(x=c(0,5,10)), interval="confidence")

  y1covered[i] = ( 3>ypr[1,2]) && ( 3<ypr[1,3])
  y2covered[i] = (13>ypr[2,2]) && (13<ypr[2,3])
  y3covered[i] = (23>ypr[3,2]) && (23<ypr[3,3])
}

mean(y1covered)
mean(y2covered)
mean(y3covered)
```

The coverage for y when x is very low is 99.6%, which is very high. The coverage for y when x is in the middle is 94.5%, which is about what we want. The coverage for y when x is very high is 88.7%, which is relatively low.

These results are what we expected.

Note

Realize again the connection between p-values and confidence intervals. To ensure that our p-values are "protected" — are no more than estimated — we need to limit ourselves to the areas where the true spread is no larger than the average

Huber-White Heteroskedastic Estimators

Because heteroskedasticity is common in several areas of study, it would be helpful to find a method for reducing its effect. This was the task that Huber and White (1980) set for themselves. They created the so-called "sandwich estimators" to reduce (not eliminate) the effect of heteroskedasticity on the estimates. Ultimately, their solution arises from using matrix multiplication to make the residuals homoskedastic.

Remember that in classical linear regression under the Gauss-Markov assumptions, the ordinary least squares (OLS) estimator $\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$ has a covariance matrix estimated in the usual manner, $\widehat{V}[\mathbf{b}] = \text{MSE}\ (\mathbf{X}'\mathbf{X})^{-1}$, where MSE is the mean squared error. This formula (as with most we've covered so far) depends on the assumption of homoskedasticity — that the variance of the residuals is constant across all observations.

When errors are heteroskedastic, this covariance estimator is biased. This leads to incorrect standard errors, invalid hypothesis tests, and unreliable confidence intervals. The Huber-White sandwich estimator (also called the heteroskedasticity-consistent covariance matrix estimator, or HCME) provides a consistent estimate of the covariance matrix of $\mathbf{b}$ even in the presence of heteroskedasticity of unknown form.

Derivation and Matrix Form

Under our usual conditions, the *asymptotic* covariance matrix of $\mathbf{b}$ is:

$$\text{A.Cov}(\mathbf{b}) = (\mathbf{X}'\mathbf{X})^{-1}\,(\mathbf{X}'\mathbf{\Sigma}\mathbf{X})\,(\mathbf{X}'\mathbf{X})^{-1} \tag{6.3.1}$$

where $\mathbf{\Sigma} = \text{diag}(\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2)$ is the $n \times n$ covariance matrix of the errors. The "sandwich" form is evident: $(\mathbf{X}'\mathbf{X})^{-1}$ forms the "bread" and $\mathbf{X}'\mathbf{\Sigma}\mathbf{X}$ is the "meat." Or, as I learned it, the $\mathbf{X}$ is the bread and the $\mathbf{\Sigma}$ is the meat. But, this way, there is a whole lotta bread on that sandwich.

Time for a quick sanity check:

If there is no heteroskedasticity, then $\Sigma = \sigma^2 \mathbf{I}$. Substituting this into equation 6.3.1 returns us to our usual estimator of $V[\mathbf{b}]$.

It is always a good idea to check that we are merely extending what we already know.

Since Σ is unknown (it is the *population* covariance matrix), White (1980) proposed replacing σ_i^2 with the squared OLS residuals $\hat{\epsilon}_i^2 = (y_i - x_i'\hat{\beta})^2$. This gives the "HC0" estimator:

$$\widehat{\text{A.Cov}}_{\text{HC0}}(\hat{\beta}) = (X'X)^{-1} \left(X' \text{diag}(\hat{\epsilon}_i^2) X\right) (X'X)^{-1} \tag{6.3.2}$$

In full matrix notation, if we define an $n \times n$ diagonal matrix $\hat{\Sigma} = \text{diag}(\hat{\epsilon}_1^2, \ldots, \hat{\epsilon}_n^2)$, then:

$$\widehat{\text{A.Cov}}_{\text{HC0}}(\hat{\beta}) = (X'X)^{-1}(X'\hat{\Sigma}X)(X'X)^{-1}. \tag{6.3.3}$$

Small-Sample Corrections

For a first attempt, this was fantastic! However, we later discovered that the HC0 estimator can be biased in small samples. Several refinements adjust the residuals to improve finite-sample performance. These are (thankfully) beyond the scope of this course.

BTW: The function in `KnoxStats` uses HC0 for adjustments.

Properties and Usage

- Consistency:
  The sandwich estimator is consistent for $\text{A.Cov}(\mathbf{b})$ under heteroskedasticity, provided $E[\varepsilon_i | x_i] = 0$ and the other usual conditions.
- Robustness:
  It is "robust" in the sense of providing valid inference without specifying the form of heteroskedasticity.
- Limitation:
  It does *not* change the point estimates $\mathbf{b}$; it only adjusts (corrects?) the estimated covariance matrix. For efficiency gains, weighted least squares or other methods may be needed.

✔ Example 6.3.1: Violent and Property Crime

Is there a significant relationship in the United States between the violent crime rate and the property crime rate a decade earlier?

Solution.
To test this, let us use the years 2000 and 1990 and the `crime` dataset.

The usual initial analysis is done.

```
library(KnoxStats)

dt = read.csv("http://rfs.kvasaheim.com/data/crime.csv")
attach(dt)
```

```
mod = lm(vcrime00 ~ pcrime90)
bptest(mod)
```

The Breusch-Pagan test indicates strong evidence of heteroskedasticity (p-value = 0.0008917). As such, at this point in your understanding, you should use `summaryHCE` instead of `summary` to obtain proper estimates of the standard errors. Here it is:

```
summaryHCE(mod)
```

with output

```
                Estimate HC Std. Error HC t value HC Pr(>|t|)
(Intercept) -203.8202734  153.08525598  -1.331417    0.183052
pcrime90       0.1363034    0.03457255   3.942531    0.000081
```

According to this analysis, there is a significant relationship between the property crime rate in 1990 and the violent crime rate in 2000. For every 1 increase in the property crime rate in 1990, the violent crime rate increases by 0.1363, on average.

If we did *not* use the sandwich correction, we would have seen this table:

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -203.82027  104.24087  -1.955   0.0563 .
pcrime90       0.13630    0.02136   6.381 6.04e-08 ***
```

Note that the estimated effects do not change. Only the standard error (and what depends on it) changes between the two models. This is because the sandwich correction acts **only on $V[\mathbf{b}]$**, not on the parameter estimates, **b**.

This page titled 6.3: Homoskedasticity is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

6.4: Multicollinearity

Recall that there was just one strict, mathematical requirement for ordinary least squares estimation: the independent variables *cannot* be linear combinations of each other. If they are — a condition known as perfect (or super) multicollinearity — the underlying matrix calculations become impossible, and the OLS estimates simply cannot be computed.

However, perfect multicollinearity is a relatively rare and extreme case. A far more common and insidious challenge arises when two or more independent variables are highly, but not perfectly, correlated. This phenomenon, often termed high multicollinearity or simply multicollinearity, allows the mathematics to proceed but introduces significant problems that can distort and undermine the regression analysis.

These problems manifest in two primary domains: the computational and the interpretative. From a computer science or numerical perspective, high multicollinearity inflates the standard errors of the coefficient estimates, making them unstable and highly sensitive to minor changes in the model or data. From the standpoint of experimental logic, it becomes difficult, if not impossible, to isolate the unique effect of each correlated variable on the dependent variable, thereby crippling the goal of causal inference. The following section will delve into these dual consequences, exploring both the diagnostic symptoms and the practical implications for researchers.

Learning Objectives

By the end of this section, you will be able to:

1. Distinguish between perfect multicollinearity, which makes OLS estimation impossible, and high multicollinearity, which inflates standard errors and makes it difficult to isolate the unique effect of each correlated predictor.
2. Explain the logical problem posed by multicollinearity: when independent variables are highly correlated, the model cannot statistically separate their individual influences on the dependent variable, leading to imprecise estimates and ambiguous conclusions.
3. Identify the key statistical indicators of multicollinearity, including large standard errors for coefficients despite a significant overall F-test, and calculate and interpret the Variance Inflation Factor (VIF) as a diagnostic tool.
4. Describe common strategies for addressing multicollinearity, such as removing one of the correlated variables or using dimension-reduction techniques like principal component analysis, while recognizing that the core problem often lies in the underlying experimental logic.

✦•·····• ✦ •·····•✦

The CS of Multicollinearity

Note that we are using a computer to perform our calculations. Because a computer is not able to store most numbers in memory perfectly, rounding errors creep into the calculations. While this will always happen when the number does not have a finite binary representation, it is most important to understand when the decimal is close to zero, when it is less than the "machine epsilon." When this is the case, the number rounds to zero. In other words, if the value is between $-\epsilon$ and $+\epsilon$, the computer treats it as a zero.

On 64-bit computers, the value of epsilon is approximately $2.220446 \times 10^{-16} = 0.000\ 000\ 000\ 000\ 000\ 222\ 044\ 6$. If the determinant of the matrix is this value or less in magnitude, the computer will claim it is singular.

```
solve( matrix( c(1,0, 0,2.220446e-16), ncol=2) )
```

Mathematically, we can calculate the determinant to be 2.220446×10^{-16}, which means the matrix is actually *not* singular. **However**, calculating the inverse returns the following:

```
Error in solve.default(matrix(c(1, 0, 0, 2.220446e-17), ncol = 2)) :
  system is computationally singular: reciprocal condition number = 2.22045e-17
```

The lesson to take beyond this specific case is that the matrix does not have to be singular for the computer to tell you it is. All that is needed is that the determinant of the $\mathbf{X'X}$ matrix be sufficiently close to zero.

Note

I term this a **Computer Science result** because it is based on the vagaries of computers instead of the vagaries of mathematics. The next section looks at what multicollinearity means in terms of the logic of experimental design and interpretation.

The Logic of Multicollinearity

The previous section examined the effects of multicollinearity on calculations, specifically of the inverse of the $\mathbf{X'X}$ matrix. If its determinant is sufficiently close to zero, then the computer will treat is as a zero, meaning the matrix will be effectively singular. However, this is only a CS problem. A good statistician will pay attention to the conditions that *cause* multicollinearity... **even minor amounts of it**.

Recall that multicollinearity occurs when a column in the data matrix is a linear combination of the other columns; that is, it happens when one variable is a linear combination of the others; that is, **it happens when one variable adds no new information beyond what the other variables contain**. For instance, if one variable is a person's height in inches and another variable is a person's height in centimeters, then multicollinearity exists. The first variable offer no information that is not contained in the second.

A statistician cares about the independent variables in the model. They are designed to explain the dependent variable. Each independent variable is supposed to be independent of the others, because each is designed to *explain a different aspect* of the response variable. If two explanatory variables are highly correlated with each other, then it will be logically impossible to determine *which of the two* is causing the change in the dependent variable:

1. Is it the logarithm of a person's height in inches or the logarithm of the *square* of a person's height in inches that can be used to estimate weight?
2. Is it average daily temperature or ice cream consumption that can be used to estimate the violent crime rate?
3. Is it educational attainment or parental income that can be used to estimate a person's future income?

These three exemplify the issue with multicollinearity in practice. The **first example** produces mathematical (a.k.a. "super-") multicollinearity because the logarithm of the square of a variable is exactly twice the logarithm of the variable. The first column is twice the second.

The **second example** does not exemplify mathematical multicollinearity. There is no function of average daily temperature that gives the ice cream consumption. However, there is a very strong linear relationship between the two. Because of this, one cannot *statistically* tell if it is the temperature or the ice cream that is affecting violent crime. With this being said, unless the dairy farmers are attacking the very foundation of Ruritanian society, the substantive scientific theory suggests that the temperature is likely the factor affecting the crime rate, not the frigid dairy.

The **third example** is more subtle. There is also a strong relationship between a person's education attainment and the parent's income (at least in Ruritania). Because of this, we are unable to statistically determine if it is the person's educational attainment or the income of the person's parents that affects the person's future income. Social science theories suggest each. The statistics with each explanatory variable also suggest each. What can we do in this case?

Indications of Multicollinearity

To see some statistical indications of multicollinearity, try the following code.

```
set.seed(30)

b0 = 3
b1 = 2
b2 = 3

x1 = seq(0,10,length=8)
x2 = c(1,2,3,4,6,7,8,9)
e  = rnorm(8)

y = b0 + b1*x1 + b2*x2 + e

mod1 = lm(y~x1)
mod2 = lm(y~x2)
modA = lm(y~x1+x2)
```

Clearly, from how this experiment is set up, we know the following:

1. There is a strong relationship between `x1` and `y` .
2. There is a strong relationship between `x2` and `y` .
3. There is a strong relationship between `x1` and `x2` .

Running `summary(mod1)` shows us that the first statement is true, *if we ignore the effect* of `x2` . Similarly, running `summary(mod2)` shows us that the second statement is true, *if we ignore the effect* of `x1` . Combining the two explanatory variables in `modA` is confusing if we do not think about multicollinearity, because `summary(modA)` gives the following results:

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    2.814      1.898   1.483    0.198
x1             2.145      1.681   1.276    0.258
x2             2.861      2.009   1.424    0.214

Residual standard error: 1.386 on 5 degrees of freedom
Multiple R-squared:  0.9946,    Adjusted R-squared:  0.9924
F-statistic: 458.7 on 2 and 5 DF,  p-value: 2.164e-06
```

Note that this model shows that *neither* independent variable is valuable in modeling the dependent variable. Since a scientist will usually put all the explanatory variables in the model, this is a lesson for us to pay attention to the relationships among the independent variables.

Caution

It is also important to look at **all** of the regression output. Note that neither of the two independent variables have small p-values. However, the R^2 value is *very close* to 1. Thus, this inconsistency also suggests that something is wrong.

A Test of Multicollinearity

So, how do we statistically detect this type of multicollinearity? A simple correlation test will not suffice if we have more than two independent variables because correlation is between only two variables.

The answer comes from the cause of the multicollinearity: **If there is multicollinearity**, then one independent variable should be linearly related to the others. A linear regression will be able to detect this. *Technically*, a linear regression for each independent variable will detect this. To make this process easy, there is the `vif` function in the `car` package. This function calculates the "**variance inflation factor**" for each independent variable. The variance inflation factor for independent variable i is defined as

$$\text{VIF}_i \stackrel{\text{def}}{=} \frac{1}{1-R_i^2} \tag{6.4.1}$$

Here, R_i^2 is the R-squared value for the model regressing the independent variables on independent variable i.

In our example above, one can calculate the VIF by hand:

```
summary( lm(x1~x2) )
1/(1-0.9921)
```

The higher the value of R^2, the *more* independent variable i can be explained by the other independent variables. In other words, the higher the VIF, the less new information that variable adds to the model.

As in much of the field, this description leads to the question

How high is too high?

The "rule of thumb" depends on the discipline. Typical cut-offs are 5, 8, and 10. If the VIF for any of the variables is greater than the cut-off, then there is "too much multicollinearity in the the model."

Fixing Multicollinearity

So, let's say that you have detected multicollinearity in your model. What can you do?

The presence of multicollinearity means that one of your independent variables is highly correlated with a linear function of the others. It adds little to the understanding of the response variable. However, is it variable i that should be examined or the others?

From a statistical standpoint, the model is not too helpful. Multiple variables are trying to explain the same aspects of the dependent variable. In other words,

1. Is it educational attainment or parent's income that affects the respondent's income?
2. Is it race or poverty that affects violent crime?
3. Is it intelligence or birth position (oldest, youngest, middle, etc. child) that affects success?
4. Is it the ranch or the cattle feed that affects the weight?
5. Is it race or income or religion or parental income or home state that affects voting behavior?

6. Is it Nordic ancestry or blood type or Neanderthal genes that affect the severity of CoViD-19?

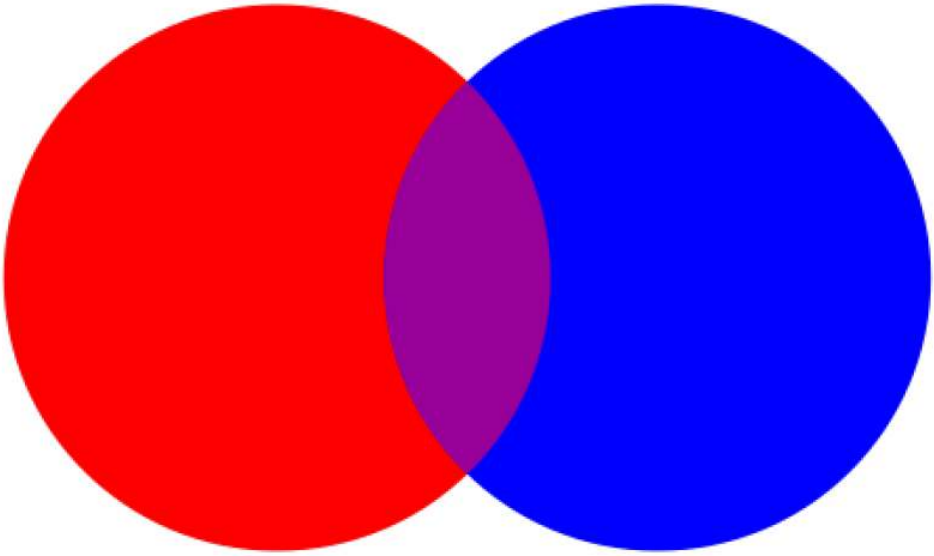


Figure 6.4.1: Diagram to illustrate multicollinearity. The left circle is the effect of the first independent variable on the dependent variable; the right, the second. The purple overlap represents the similarity between the two variables. The rectangle represents *everything* that affects the dependent variable, which means the colored portion represents everything in the *model* that affects the dependent variable.

In each of these, the explanatory variables are highly correlated and have been used to model the response variable. Because of the correlations, conclusions about what *really* affects the dependent variable are unclear. Statistically, the answer is "Yes, each does."

However, since each of the independent variables above are correlated, their effects overlap. This is represented as the purple overlapping area in Figure 6.4.1. While each variable has an effect on the dependent variable (red and blue), that effect is also split with the other variable (or variables). As such, the key is trying to separate the three sections to determine whether it is the red, the blue, or the purple that is affecting the response variable.

Unfortunately, this is beyond the scope of this course. For those who are interested, you may want to investigate factor analysis (FA) and principal component analysis (PCA). These are two methods for dealing with that overlap (purple area). The first focuses on estimating the purple area; the second, on creating two other variables that combine the two independent variables into their independent components (the parts that are purely red and blue). The advantage is that the independent variables become independent ($\text{VIF} = 1$). The disadvantage is that the newly-created independent variables are only related to the original explanatory variables; thus, interpretation is made more complicated.

Note

By the way, Figure 6.4.1 illustrates several things about the model. Everything that affects the dependent variable is represented by the rectangle, whereas the colored part represents **only what the model explains**. Thus, one may think of the ratio of the colored area to the total area as being the R^2 value.

Question:

Given that the size of the circles is fixed, how would you arrange them to cover the most area of the rectangle? What would that mean in terms of the variables?

This page titled 6.4: Multicollinearity is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

6.5: Conclusion

In this chapter, we systematically examined the consequences of violating the core assumptions of ordinary least squares regression — along with the effects of multicollinearity. In each case, we introduced a specific violation and observed its impact on our parameter estimates, test statistics, and confidence intervals.

Our exploration revealed that not all violations are equally severe. For instance, deviations from the Normality assumption generally have minimal effects on inference — unless the error distribution lacks a finite variance, in which case standard tests become nearly useless. A more critical issue is misspecifying the model's expected value, such as omitting a necessary predictor. This bias undermines everything: estimators, tests, and confidence intervals become unreliable, underscoring the necessity of a correctly specified model.

Heteroskedasticity, while leaving coefficient estimates unbiased, distorts inference by invalidating the standard errors used in hypothesis tests and confidence intervals. The issue is not one of bias, but of miscalibrated uncertainty. Finally, multicollinearity — stemming from highly correlated predictors — inflates standard errors, reducing statistical power. Beyond the numerical consequence, it creates an interpretive challenge by making it difficult to disentangle the distinct effects of the correlated variables.

Taken together, these findings reinforce a central principle: **awareness and diagnosis of these conditions are essential for trustworthy regression analysis**.

Caution

Again, be aware of the multiple comparisons issue discussed in The Appendix of Statistics. It explains why you need to either adjust your p-value or your alpha level when performing multiple tests, such as when you are testing **both** $\beta_0 = 4$ *and* $\beta_1 = 0$.

This page titled 6.5: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

6.6: End-of-Chapter Materials

R Functions

In this chapter, we were introduced to many, many, many R functions that will be useful in regression. In fact, this chapter uses more R functions than any other chapter in this book. Here are the many.

Packages



- `car`
  This package provides several statistical tests used in the book "An R Companion to Applied Regression'" by J. Fox and S. Weisberg. It is a great package that provides a lot of additional functionality for R.
- `lawstat`
  This package provides several statistical tests used in law and public policy analysis. It provides the `runs.test` function for us.
- `lmtest`
  This package provides many tests related to linear models. It provides an implementation of the Breusch-Pagan test, `bptest`, which tests for heteroskedasticity in the residuals.
- `KnoxStats`
  This package was designed to make some regression analysis a bit easier. It contains an extension to the `runs.test` to make it easier to implement in the realm of regression diagnostics.



Statistics

- `source(filename)`
  This function runs an R script from a separate file. That file may be local or on the Internet.
- `runs.test(E, order)`
  This `KnoxStats` alteration to the `lawstat` function tests whether the variable E, as ordered by `order` exhibits fit issues.
- `shapiroTest(E)`
  This tests the null hypothesis that the variable E comes from a Normal distribution. It is based on the `shapiro.test()` function in the basic R installation. It adds capabilities to test Normality in multiple groups.
- `lm(formula)`
  This is the function that performs ordinary least squares estimation on linear models.
- `bptest(mod)`
  This function from the `lmtest` package performs the Bresuch-Pagan test for heteroskedasticity.
- `confint(mod)`
  This calculates confidence intervals for the parameters in ordinary least squares regression.
- `mean(x)`
  This calculates the mean of a sample.
- `summary(x)`
  This produces the six-number summary or a frequency table of the provided variable, depending on the type of variable.
- `summary.lm(mod)`
  When applied to a linear model fit using either the `aov` function or the lm function, this provides estimates of the effects of

the numeric variables and the levels of the categorical variables in the model.

- `summary.aov(mod)`
  When applied to a linear model fit using either the `aov` function or the lm function, this provides estimates of the statistical significance of the variables in the model.
- `predict(mod)`
  This predicts the values of the dependent variable at each point in the dataset *or* for the values specified.
- `fligner.test(formula)`
  This tests for heteroskedasticity when the independent variable is categorical.
- `aov(formula)`
  This function performs ordinary least squares estimation on linear models.
- `vif(model)`
  This function calculates the variance inflation factor (VIF) for each of the independent variables in the model.
- `set.base(var,level)`
  This `KnoxStats` package function redefines the base category of the provided level. By default, the base category is the first according to the alphabet.

Probability

- `set.seed(x)`
  This sets the random number seed. It is done to ensure the replicability of experiments.
- `rexp(n, rate)`
  This generates n random values from an Exponential distribution with the specified rate parameter.
- `rnorm(n, mean, sd)`
  This generates n random values from a Normal distribution with specified mean and standard deviation. By default, the mean is 0 and the standard deviation is 1.
- `runif(n, min, max)`
  This generates n random values from a Uniform distribution with specified minimum and maximum values. By default, the minimum is 0 and the maximum is 1.

Mathematics

- `head(x)`
  This returns the first six values in the variable.
- `foot(x)`
  This returns the *last* six values in the variable.
- `seq(from, to, by, length)`
  This returns a vector of sequential values, where **by** indicates the step size and **length** specifies the vector length.
- `length(x)`
  This calculates the length of a vector (variable), which is the sample size, **n**.
- `residuals(mod)`
  This calculates the residuals in the model. Recall that the residuals are the difference between the observed and the predicted.

Graphics

- `qqnorm(x)`
  This creates a Normal quantile-quantile plot for the given values.
- `qqline(x)`
  This adds the diagonal line to the quantile-quantile plot.
- `overlay(x)`
  This, from the `KnoxStats` package, produces a histogram with a Normal curve overlaying it.
- `par(...)`
  This sets the parameters for the next graphic to be started. Look through the help page for this function to see all you can specify.
- `plot(x,y)`
  This produces a scatter plot of the y-values against the x-values.
- `axis(side)`
  When a plot is already drawn, this adds values along axis number **side**.
- `title(...)`
  When a plot is already drawn, this adds the x- and y-labels.
- `lines(x,y)`
  When a plot is already drawn, this draws lines between each subsequent (x, y) pair.
- `points(x,y)`
  When a plot is already drawn, this draws points at each (x, y) pair.

Programming

- `attach(dataframe)`
  This allows you to access the variables in the **dataframe** without having to prefix each with **dataframe$**.
- `library(package)`
  This loads an external package that you have already installed on your computer. It allows access to all functions and data sets in the **package** package.
- `as.character(x)`
  This changes the values in variable **x** to be characters.
- `as.numeric(x)`
  This changes the values in variable **x** to be numbers.

Exercises

1. Show that if the expected value of the residuals is constant, but non-zero, then the OLS estimator of β_1 remains unbiased.
2. Show that $E[b_0] = \beta_0 + E[\varepsilon]$, if $\bar{x} = 0$, regardless of whether the residuals are correlated with the independent variable.

Applied Readings

- Joshua D. Angrist and Alan B. Krueger (1991). "Does Compulsory School Attendance Affect Schooling and Earnings?" *The Quarterly Journal of Economics* 106(**4**): 979–1014.
  doi:10.2307/2937954

- Eugene F. Fama and Kenneth R. French (1992). "The Cross-Section of Expected Stock Returns." *The Journal of Finance* 47(**2**): 427–65.
  doi:10.2307/2329112
- Jerayr Haleblian and Sydney Finkelstein (1999). "The Influence of Organizational Acquisition Experience on Acquisition Performance: A Behavioral Learning Perspective." *Administrative Science Quarterly* 44(**1**), 29–56.
  doi:10.2307/2667030

Theory Readings

- George E. P. Box (1976). "Science and Statistics," *Journal of the American Statistical Association*, 71(**356**): 791–799.
  doi:10.1080/01621459.1976.10480949
- James V. Bradley (1968). *Distribution-Free Statistical Tests*. New York: Prentice-Hall.
- John Fox and Harvey Sanford Weisberg (2019). *An R Companion to Applied Regression*, third edition. Thousand Oaks, CA: SAGE Publications.
  ISBN: 9781544336473
- Frank J. Massey, Jr. (1951). "The Kolmogorov-Smirnov Test for Goodness of Fit." *Journal of the American Statistical Society*. 46(**253**): 68–78.
  JSTOR: 2280095
- Abraham Wald and Jack Wolfowitz (1940). "On a test whether two samples are from the same population." *The Annals of Mathematical Statistics* 11(**2**): 147–162.
  JSTOR: 2235872

This page titled 6.6: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

7: Time for Some Examples



We have covered a lot of theory and mathematics over the past several chapters. Here, we will apply what we have learned to help settle the theory into our minds. In other words, we will perform the analysis process with the information and skills we now have.

This means we will use data to answer our research questions. Of course, we will need to examine the research question to determine the appropriate model, check the assumptions — both statistically and graphically — and properly interpret the results.

That is a lot of summarizing to do!

✦•·····• 👻 •·····•✦

And so, we have completed a majority of the important mathematics underlying **ordinary least squares estimation**. Be aware that OLS is how we estimate the parameters. The models itself is referred to as the classical linear model. It makes the usual four assumptions. The observations follow the equation

$$y_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \cdots + \beta_k x_{k,i} + \varepsilon_i \tag{7.1}$$

and the residuals follow this distribution

$$\varepsilon_i \overset{\text{iid}}{\sim} N\left(0;\ \sigma^2\right) \tag{7.2}$$

From *those* assumptions, we were able to use OLS to calculate formulas for the estimators of $\beta_0, \beta_1, \ldots, \beta_k$. The next chapter used the distribution to determine the distribution of those estimators. This led to confidence intervals for the parameters and test statistics for testing hypotheses about the parameters. It also led to distributions and intervals and test statistics for estimated and predicted values of y.

All of that from four small assumptions.

This chapter will apply these results to different research questions to illustrate the statistical research process. So, turn the page and begin seeing applications of what we have done.

· · — · ✶ · — · ·

Chapter Sections

7.1: Violent Crime

7.2: Violent Crime, Wealth, Region

7.3: Cows in Děčín

7.4: Conclusion

7.5: End-of-Chapter Materials

This page titled 7: Time for Some Examples is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

7.1: Violent Crime

To help settle all of this, let's see a simple extended example of modeling the violent crime rate in 2000 using just the violent crime rate in 1990.

Learning Objectives

By the end of this section, you will be able to:

1. Work through a complete simple linear regression analysis of the relationship between violent crime rates in 1990 and 2000, following a systematic workflow from data loading to final interpretation.
2. Check the key regression assumption s —normality of residuals, correct functional form (constant expected value), and homoskedasticity (constant variance) — using a combination of graphical diagnostics (histograms, residual plots) and formal statistical tests (Shapiro-Wilk, runs test, Breusch-Pagan).
3. Interpret the full regression output, including coefficient estimates, standard errors, t-statistics and p-values, measures of fit (R^2), and overall model F-test.
4. Construct and interpret point estimates, confidence intervals for the mean response, and prediction intervals for a new observation at a specified value of the predictor, correctly distinguishing between the width and interpretation of the intervals.

✦•·····• ✦ •·····•✦

The Example

Determine the relationship between the violent crime rate in 1990 and in 2000.

If you find an appropriate model, *estimate* the expected 2000 violent crime rate for a state with a 1990 violent crime rate of 100. Finally, *predict* the 2000 violent crime rate for a state with a 1990 violent crime rate of 100

The Preamble

The preamble is the part of the code that imports the extra functions, loads the data, and gives us an overview of it. This is a typical preamble

```
### Preamble
library(KnoxStats)
library(lawstat)
library(lmtest)

dt = read.csv("http://rur.kvasaheim.com/data/crime.csv", stringsAsFactors = TRUE)
attach(dt)

summary(dt)
```

Note that there are *many* variables in this particular data set. Since we are modeling the violent crime rate in 2000 using the rate in 1990, we will only use the variables `vcrime00` and `vcrime90` .

R Feature

Back in the early part of this century, R used to read strings in the data as factors, by default. However, a few years back, it changed the default behavior. So, if you are loading categorical variables — and using them in your analysis — you will need to include `stringsAsFactors = TRUE` in your `read.csv` call. Just be aware of this.

The Model

To fit the model and estimate the parameters using ordinary least squares, run this line:

```
crimeMod = lm(vcrime00 ~ vcrime90)
```

Nothing gets outputted by this line. R just echoes it if you typed it correctly. However, *a lot* has happened behind the scenes. Inside R, the model was fit using ordinary least squares (using matrices). The parameters were estimated. All of this was done behind the scenes.

Test Requirements

The next step is to check that the model does not violate any of the assumptions/requirements.

Normality of the Residuals

The first we will check is the Normality of the residuals:

```
e = residuals(crimeMod)

# Normal Residuals?
overlay(e)
shapiroTest(e)
```

The histogram overlaid with the normal curve suggests the residuals are slightly skewed to the right. The Shapiro-Wilk test strongly indicates a lack of Normality (p-value = 0.004424). The sample size of $n = 51$, however, definitely seems large enough to ensure the sums of the residuals closely follows a normal distribution (this is the *actual* requirement, by the way).

If you would like to check this, run the following code (first think what it does and why it answers this problem):

```
et = numeric()
for(i in 1:1e3) {
  x = sample(e, replace=TRUE)
  et[i] = sum( x )
}
shapiroTest(et)
```

Since the reported p-value is much greater than α, we can conclude that the sample sums are sufficiently Normal. And, it is the sample sums that affect the distribution of b_0 and b_1.

Thus, the model passes the normality requirement. ✅

Constant Expected Value (Functional Form)

The second assumption to test is constant expected value (proper model form):

```
# Constant Expected Value of Residuals
plot(vcrime90,e)
runs.test(e, order=vcrime90)
```

The residuals plot seems a bit inconclusive to me. This is mainly due to the single point far to the right (the District of Columbia). The runs test, however, indicates that there is no significant evidence the residuals follow anything other than a horizontal line (p-value = 0.6732).

Thus, the model does not violate the second assumption.

Homoskedasticity (Constant Variance)

The third assumption is that the variance of the residuals is constant:

```
# Constant Variance of Residuals
plot(vcrime90,e)
bptest(crimeMod)
```

For me, the graphic is inconclusive because of DC. The Breusch-Pagan test did not detect significant heteroskedasticity (p-value = 0.1041 > 0.05 = α). Thus, the model passes the third requirement.

Multicollinearity

The last thing to check is multicollinearity. There is no need to check it for this model because there is just the one independent variable that we already see has variation. So, it passes this test, too. ✅

The Results of the Model

Since none of the requirements are violated in this model, it seems appropriate. With that, we can see the estimates:

```
Call:
lm(formula = vcrime00 ~ vcrime90)

Residuals:
    Min      1Q  Median      3Q     Max
-241.32  -42.84  -18.04   40.97  208.41

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept) 109.52716   21.42679   5.112 5.27e-06 ***
vcrime90      0.58065    0.03107  18.689  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.10   1

Residual standard error: 85.55 on 49 degrees of freedom
Multiple R-squared:  0.877, Adjusted R-squared:  0.8745
F-statistic: 349.3 on 1 and 49 DF,  p-value: < 2.2e-16
```

The **first section** of the output reports the model you presented. Use it to double-check you typed things in correctly or to remind you what the model is examining. The **second part** produces the five-number summary of the residuals. Since the mean (0) is greater than the median (-18.04), there is evidence of a positive skew to the residuals. This, we discovered above by looking at the histogram of the residuals.

The **third section** is the "regression table." Each row corresponds to a different independent variable (or the intercept). The columns are the estimates, the standard errors, the test statistic (estimate divided by standard error), and the p-value.

In this example, there is *very strong* evidence that the relationship between the violent crime rate in 1990 and in 2000 is positive with the effect $b_1 = 0.58065$. If State A had a higher violent crime rates in 1990 than State B, then it also tended to have a higher violent crime rates in 2000.

The intercept, $b_0 = 109.52716$ indicates that for a state with 0 violent crime in 1990, the expected violent crime rate in 2000 is 109.52716 crimes per 100,000 people. However, since no state was close to having a violent crime rate in 1990 of 0, so this interpretation does not make statistical sense.

Remember that we should only use our models to predict and estimate for values of the 1990 violent crime rate within the domain of the `vcrime90` variable in our data. Interpolation is appropriate; extrapolation, rarely so.

The **fourth part** contains the statistics that describe the model, as a whole. The estimate for σ is 85.55. [Remember $\varepsilon_i \overset{\text{iid}}{\sim} N(0, \sigma)$.] The model explains 87.45% of the variance contained in the null model. The model does give some information about the dependent variable ($F = 349.3$; $\nu_1 = 1$, $\nu_2 = 49$; p-value $\ll 0.0001$).

The Graphic

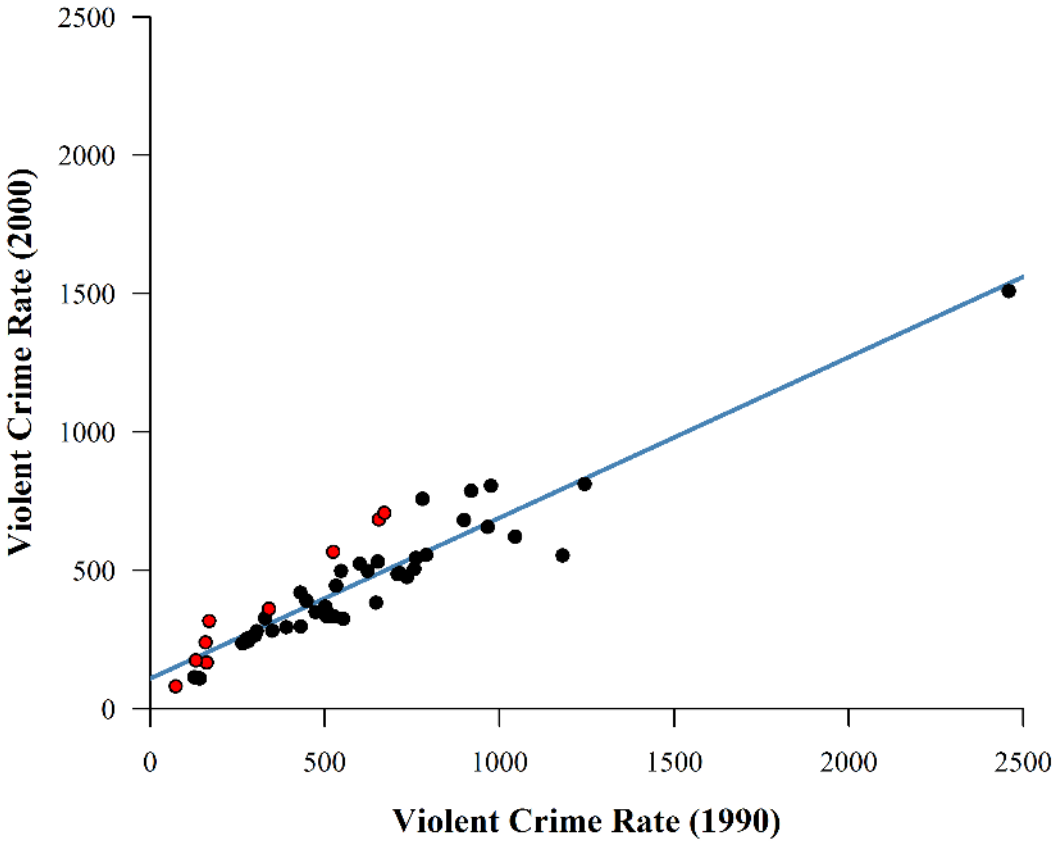


Figure **7.1.1**: Plot of the violent crime rate in 2000 against that in 1990. The ordinary least squares line of best fit is included. Red-colored states are those whose violent crime rate increased from 1990 to 2000.

The following code produces a graphic like that of Figure **7.1.1**:

```
plot.new()
plot.window( xlim=c(0,2500), ylim=c(0,2500) )

axis(1); axis(2)

title(xlab="Violent Crime Rate (1990)", line=2.75)
title(ylab="Violent Crime Rate (2000)", line=3.25)

xx = seq(0,2500)
yy = predict(crimeMod, newdata=data.frame(vcrime90=xx))
lines(xx,yy, col="steelblue", lwd=2)

points(vcrime90,vcrime00, pch=21, bg=1+(vcrime00>vcrime90))
```

Note that the graphic also indicates which states had their violent crime rate increase. It comes from this line:

```
points(vcrime90,vcrime00, pch=21, bg=1+(vcrime00>vcrime90))
```

The first two slots are the x- and y-values. The third slot specifies the plotting character. A `pch` of 21 is a dot with its insides colorable. The fourth slot, `bg`, specifies the color to fill the inside of the dots (`bg` = "background").

The part `(vcrime00>vcrime90)` is a Boolean value that takes on value 1 if the statement is true (violent crime rate increased) and 0 otherwise. Adding 1 to each ensures that the two colors are 1 and 2 — black and red.

Confidence Interval for β_1

In addition to calculating the point estimates of the slope and intercept, we can also calculate confidence intervals:

```
confint(crimeMod)
```

From this output, we are 95% confident that the effect of the violent crime rate in 1990 on 2000 is between **0.518** and **0.643**.

Confidence Interval for Y

We can calculate the expected value of Y for a given value of x:

```
predict(crimeMod, newdata=data.frame(vcrime90=100), interval="confidence")
```

We are 95% confident that the expected value of Y when $x = 100$ is between **129.5** and **205.6**, with a point estimate of **167.6**.

Prediction Interval for Y

Finally, we can **predict the value** of Y for a new value of x:

```
predict(crimeMod, newdata=data.frame(vcrime90=100), interval="prediction")
```

We are 95% sure that the next *observation* of the violent crime rate in 2000 for a state with a violent crime rate in 1990 of 100 is between **−8.5** and **343.7**, with a point prediction of **167.6**.

This page titled 7.1: Violent Crime is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

- **Current page** by Ole Forsberg is licensed CC BY-NC-SA 4.0.
- **5: Improved! Now with Probabilities** by Ole Forsberg is licensed CC BY-NC-SA 4.0.

7.2: Violent Crime, Wealth, Region

That was fun!

Let's now try this with *two* independent variables. We will model the violent crime rate in 2000 using the GSP per capita in 1990 *and* the region of the state. This will give us the opportunity to reiterate and emphasize that these methods are not constrained to numeric independent variables. As in the second Toy Example in Chapter 3, we can represent categorical independent variables appropriately and model using ordinary least squares estimation.

Learning Objectives

By the end of this section, you will be able to:

1. Construct and compare interaction and additive multiple regression models to analyze the relationship between a continuous response variable (violent crime rate), a continuous predictor (wealth/GSP per capita), and a categorical predictor (census region).
2. Test the key regression assumptions—normality, correct functional form, homoskedasticity, and multicollinearity—in a model with both numeric and categorical predictors, using appropriate graphical methods and statistical tests (e.g., ANOVA for constant expected value across groups, Fligner-Killeen test for constant variance, VIF for multicollinearity).
3. Interpret the output from both the analysis of variance table (summary.aov) for overall variable significance and the regression coefficient table (summary.lm) for comparisons between specific levels of a categorical variable, understanding the role of the baseline category.
4. Change the baseline category for a categorical predictor to test specific hypotheses about differences between groups, and explain the multiple comparisons issue that arises when conducting many such tests.
5. Construct and interpret point estimates, confidence intervals for the mean response, and prediction intervals for new observations at specified values of the predictors.

✦•·····• ✦ •·····•✦

The Example

Model the 2000 violent crime using the 1990 GSP per capita and determine if the relationship differ across the *nine* census regions.

If you find an appropriate model, estimate the expected 2000 violent crime rate for a state in the Pacific region with a GSP per capita of $50,000 in 1990. Also, predict the 2000 violent crime rate for that same state.

The Preamble

This preamble is mostly the same as the one in last section, so only run these lines if you are starting a new session in R. However, there is a new line. Because R (by default) loads string data as strings,

```
### Preamble
library(KnoxStats)
library(lawstat)
library(lmtest)
```

```
dt = read.csv("http://rur.kvasaheim.com/data/crime.csv", stringsAsFactors = TRUE)
attach(dt)

summary(dt)
```

Note that there are *many* variables in this particular data set. Since we are modeling the violent crime rate in 2000 using the rate in 1990 and the state's region, we will only use the variables `vcrime00`, `gspcap90`, and `census9`.

Caution

Again, remember that back in the early part of this century, R used to read strings in the data as factors, by default. However, a few years back, it changed the default behavior. So, if you are loading categorical variables — and using them in your analysis — you should include `stringsAsFactors = TRUE` in your `read.csv` call.

Just be aware of this.

Finding the Model

The following creates the interaction model between a numeric and a categorical variable. This particular type of interaction analysis is referred to as the Analysis of Covariance, ANCOVA:

```
modEd1 = lm(vcrime00 ~ gspcap90 * census9)
summary.aov(modEd1)
```

The interaction model allows for the effect to vary between the levels. In terms of this problem, the interaction model allows the effect of the 1990 violent crime rate on the 2000 to be different for the Midwest, the Northeast, the South, and the West.

It does not *force* it to be different. It only *allows it* to be different.

· · — · ✶ · — · ·

Because of the writings of a 14th-Century monk by the name of William of Ockham, there is a bias in science to create models that are as simple as possible, without being too simple (his doctrine of efficient reasoning).

Non sunt multiplicanda entia sine necessitate.

The usual translation is "Things are not to be multiplied without necessity." In other words, simpler models tend to be more helpful than complicated ones. Realize that they are more "helpful" and not more "correct." To drive this point home, allow me to quote George E. P. Box (1976):

> Since all models are wrong the scientist cannot obtain a "correct" one by excessive elaboration. On the contrary following William of Occam he should seek an economical description of natural phenomena. Just as the ability to devise simple but evocative models is the signature of the great scientist so overelaboration and overparameterization is often **the mark of mediocrity**. [emphasis mine]



George E. P. Box (2011)
(Source: Wikipedia)

> Note, however, that this doctrine/belief did not originate with Billy. It goes back to — at least — Aristotle in his *Posterior Analytics*:
>
> > "We may assume the superiority *ceteris paribus* of the demonstration which derives from fewer postulates or hypotheses."



The results from the above code are given here

```
                  Df  Sum Sq Mean Sq F value   Pr(>F)
gspcap90           1  816163  816163  26.149 1.32e-05 ***
census9            8  794820   99353   3.183   0.0087 **
gspcap90:census9   8  273868   34233   1.097   0.3901
Residuals         33 1029986   31212
```

Note that the p-value (last column) for the interaction term (`gspcap90:census9`) is greater than our usual $\alpha = 0.05$ ($\mathbf{p-value = 0.3901}$). This tells us that the **interaction term** is *not* statistically significant. In other words, we can remove it from our model **without adversely affecting our understanding of the dependent variable**.

When the model has no interaction terms, it is called an **additive model**. The following code fits the additive model.

```
modEd2 = lm(vcrime00 ~ gspcap90 + census9)
```

Testing the Requirements

As usual, the next step is to test the assumptions. Note that this process is the same, even if the particulars differ. The `census9` variable is categorical, not numeric. That requires we think a bit more about how to perform the assumption testing.

Normality

Again, the first assumption to test is the Normality of the residuals:

```
e = residuals(modEd2)

# Normality checking
overlay(e)
shapiroTest(e)
```

According to the Shapiro-Wilk test there is no significant evidence that the residuals come from a non-Normal distribution ($\mathbf{p-value = 0.1732}$). Thus, the model passes this test. ✅

Constant Expected Value (Functional Form)

The second requirement we test is that the expected value of the residuals is constant against *each of* the independent variables.

```
# Expected Value
plot(gspcap90, e)
runs.test(e, order=gspcap90)
```

The runs test indicates that there is no evidence the expected values are not constant ($\mathbf{p-value = 0.2038}$). Thus, this test is passed, too. ✅

> Yippee!

But wait! This only tested for constant expected value of the residuals against one of the two independent variables. It is required that the expected value is constant against **all of them**.

Here is the problem with just using the runs test when the independent variable is categorical: The ordering within each level is not uniquely defined. For a hint, always graph and think about what the graphic is telling you. Look at the residuals plot.

```
plot(census9,e)
```

Holy side-by-side box-and-whiskers plot, Batman! This makes sense, because we are plotting a numeric variable (`e`) across nine levels. We need to test that the expected value (mean) is the same in each group.

From your prior statistics class, this screams ANOVA!

```
summary(aov(e~census9))
```

The p-value returned is 1, which is greater than α. So we fail to reject the null hypothesis of equal expected values.

Thus, this requirement is *entirely passed*. (sigh of relief) ✅

Caution

Well, this result should not be surprising. Because of the mathematics of OLS, the means in each group will be centered at zero. Thus, you should expect p-values of 1 whenever doing this test.

Constant Variance

The last requirement is that the residuals have a constant variance against each of the independent variables. For the numeric variable, this is not a problem:

```
# Heteroskedasticity
plot(gspcap90,e)
hetero.test(e,gspcap90)
```

With a p-value of 0.7008 for the heteroskedasticity test, this test is passed for the numeric independent variable.

For the categorical variable, remember that we need to test equality of variance *across several groups*. From your previous statistics course, you may recall that the Fligner-Killeen test does just this:

```
plot(census9,e)
fligner.test(e~census9)
```

According the the Fligner-Killeen test, there is no evidence of heteroskedasticity ($\mathbf{p-value = 0.7869}$).

Thus, this model passes the homoskedastic requirement. Vahooo! ✅

Note

We could have taken care of both tests for heteroskedasticity using the Breusch-Pagan test. However, if the model fails that test, we would have no clue as to how to fix it. Breaking it into two parts allows us the additional information of which variable caused the issues.

Personally, I run the Breusch-Pagan test to determine *if* there is a violation, then the separate tests to get information on *where* the issue lies.

Multicollinearity

The last thing to check is multicollinearity. Let us use the `vif` function in the `car` package to see how that function is used.

```
vif(modEd2)
```

This gives output

```
             GVIF Df GVIF^(1/(2*Df))
gspcap90 1.226757  1        1.107591
census9  1.226757  8        1.012855
```

So, what does this output mean?

Focus on the `GVIF^(1/(2×df))` column. This holds the VIF values, adjusted for dimensionality. Since both are below our usual cut-off of 5, there is no multicollinearity concern. ✅

If we had used the interaction model, we would have run

```
vif(modEd1, type="predictor")
```

That version also adjusts for interaction terms.

Note

In reality, it **never hurts** to run the version that includes `type="predictor"` . If there are no interaction terms, then it returns exactly the same thing as the default `type="terms"` .

The Final Model

This model is appropriate, and we can now see the estimates. However, if we are in the "model creation" or "model selection" mode, we need to determine if both variables are statistically significant. If either is not, then that variable needs to be dropped and the new model tested.

To get p-values for the *variables*, just run

```
summary.aov(modEd2)
```

Yeppers, that is `summary.aov` that you are using. It provides statistical significance of the **variables**, while `summary.lm` provides the statistical significance of the levels of the categorical variables.

The output from the `summary.aov(modEd2)` command is

```
             Df  Sum Sq Mean Sq F value   Pr(>F)
gspcap90      1  816163  816163  25.664 9.07e-06 ***
census9       8  794820   99353   3.124  0.00753 **
Residuals    41 1303854   31801
```

The p-value for the `gspcap90` variable is less than alpha, so that variable has a significant effect on the violent crime rate. The p-value for the `census9` variable is also less than alpha, so it too needs to be included in the final model.

In short, this is the model we can use. The abbreviated regression table from this model is

```
Coefficients:
                             Estimate        Pr(>|t|)
(Intercept)                 1.125e+02           0.308
gspcap90                    1.550e-02         6.7e-05 ***
census9East South Central   7.518e+01           0.535
census9Middle Atlantic     -6.777e+01           0.608
census9Mountain            -1.707e+02           0.101
census9New England         -1.708e+02           0.122
census9Pacific             -1.316e+02           0.263
census9South Atlantic       1.590e+02           0.125
census9West North Central -4.953e+01           0.638
census9West South Central   1.197e+02           0.323
```

From this, we can conclude that there is a statistically significant, and positive(!), effect of average state wealth on the violent crime rate. We get that conclusion from the `gspcap90` line in the table.

The rest of the table compares the effect of *each level* of the `census9` variable to the **base category**, East North Central. As no p-values is less than alpha, we cannot conclude that any of the regions is statistically **different in its effect from the East North Central region**.

What about when compared to the Mountain region?

Changing the Base Category

In other words, how do we change the base level? **First**, we have to specify that we want the Mountain region to be the base category against which everything else is calculated. Then, we need to re-fit the model with the new base.

```
census9 = set.base(census9, "Mountain")
modEd3 = lm(vcrime00 ~ gspcap90 + census9)
summary(modEd3)
```

The regression table now indicates that the violent crime rate in the Mountain region is significantly lower than that in the East South Central, South Atlantic, and West South Central regions. Here is the regression table

```
Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
(Intercept)                -58.209772  96.710953  -0.602 0.550559
gspcap90                     0.015502   0.003493   4.438  6.7e-05 ***
census9East North Central 170.671828 101.668158   1.679 0.100815
census9East South Central 245.850643 109.804042   2.239 0.030645 *
census9Middle Atlantic    102.903945 121.830985   0.845 0.403211
census9New England         -0.152624  96.646604  -0.002 0.998748
census9Pacific             39.030098 105.450428   0.370 0.713193
census9South Atlantic     329.709936  89.146396   3.699 0.000637 ***
census9West North Central 121.136876  92.324591   1.312 0.196793
census9West South Central 290.395590 109.258299   2.658 0.011159 *
```

Question: How does the violent crime rate in the different regions compare to the *Pacific* region?

```
census9 = set.base(census9, "Pacific")
modEd3 = lm(vcrime00 ~ gspcap90 + census9)
summary(modEd3)
```

The violent crime rate in the Pacific region is significantly lower than that in the South Atlantic and the West South Central regions.

Question: How do the regions compare to the South Atlantic region?

```
census9 = set.base(census9, "South Atlantic")
modEd3 = lm(vcrime00 ~ gspcap90 + census9)
summary(modEd3)
```

The violent crime rate in the South Atlantic region is significantly higher than in the Mountain, New England, Pacific, and West South Central regions.

> Please remember the multiple comparisons issue. That these individual analyses only work if you perform only one of them. Multiple comparisons require adjustment of the alpha-level.

Creating the Graphic

The following code generates the graphic in Figure 7.2.1, below.

```
par(mar=c(4,4,0,1)+0.5, family="serif", las=1)
par(xaxs="i", yaxs="i")
par(cex.lab=1.2, font.lab=2)

plot.new()
plot.window( xlim=c(0,75), ylim=c(0,2500) )

axis(1); axis(2)
title(xlab="GSP per Capita (1990) [$000]", line=2.75)
title(ylab="Violent Crime Rate (2000)", line=3.5)

xx = seq(15,70)*1000
yyPac = predict(modEd2, newdata=data.frame(gspcap90=xx, census9="Pacific"))
yyMtn = predict(modEd2, newdata=data.frame(gspcap90=xx, census9="Mountain"))
yySAt = predict(modEd2, newdata=data.frame(gspcap90=xx, census9="South Atlantic"))

lines(xx/1000,yyPac, col="steelblue", lwd=2)
lines(xx/1000,yyMtn, col="gold", lwd=2)
lines(xx/1000,yySAt, col="pink", lwd=2)

points(gspcap90/1000,vcrime00, pch=21, bg="lightsteelblue")
```

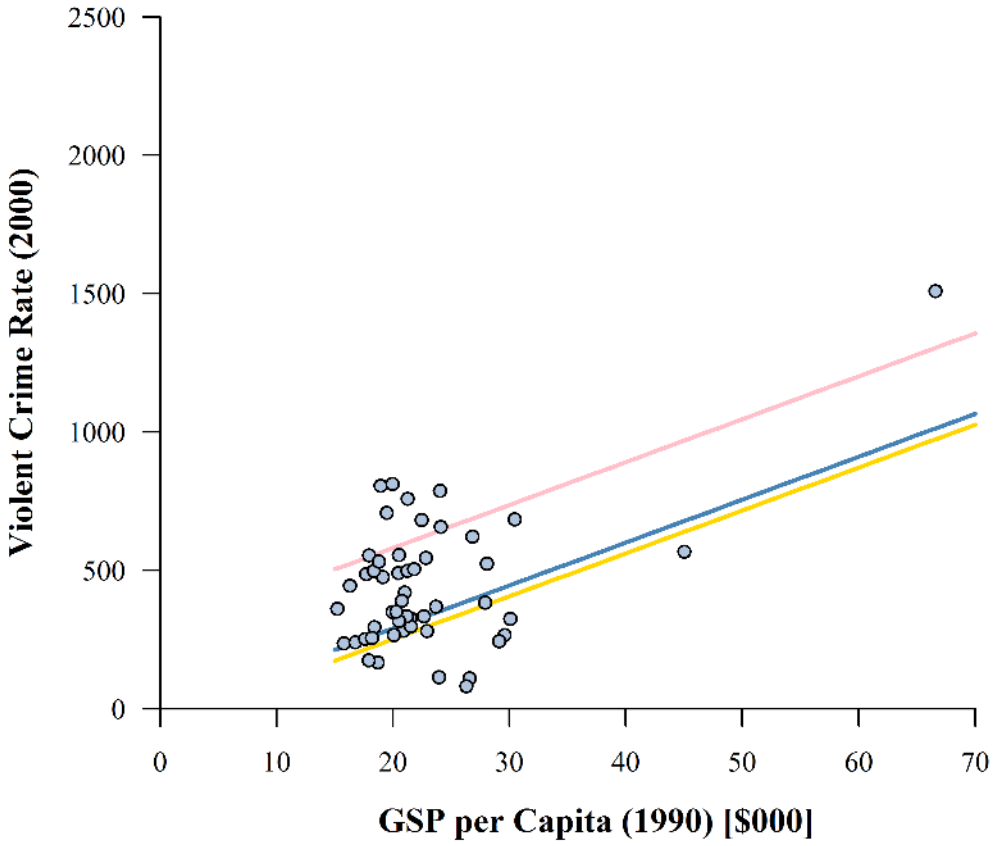


Figure **7.2.1**: Plot of the violent crime rate in 2000 against the GSP per capita in 1990 (in thousands of dollars). The ordinary least squares line of best fit is included. The red-colored line is the estimate for the South Atlantic states; blue, Pacific states; and gold, Mountain states.

It would be helpful to have a legend, but let us leave that for another day!

Also, you need to be able to determine what each line of this script does.

Confidence Interval for Slope (Effect)

We can obtain confidence intervals for the effects using the same method as before. The interpretation follows the same rules, but the table is *much* bigger:

```
confint(modEd2)
```

We are 95% confident that the effect of the GSP per capita on the violent crime rate is between 8 and 23 additional violent crimes (per 100,000 population) for every $10,000 increase in GSP per capita.

The confidence intervals for the effects of each of the levels is *as compared to* the base level. Thus, we are 95% confident that the average violent crime rate in the West North Central region is between 21 and 396 *lower* than that in the South Atlantic region (the base level).

Confidence Interval for Y

Again, we can estimate the expected value of a violent crime rate, given the GSP per capita and the region.

```
predict(modEd2, newdata=data.frame(gspcap90=50000, census9="Pacific"), interval="confi
```

We are 95% confident that the expected violent crime rate in a Pacific-region state with a GSP per capita of $50,000 is between 537 and 975, with a point estimate of 756 violent crimes per 100,000 people.

Prediction Interval for Y

Finally, we can also calculate a prediction interval for a new observation:

```
predict(modEd2, newdata=data.frame(gspcap90=50000, census9="Pacific"), interval="predi
```

We are 95% sure that the violent crime rate for a new observation of a Pacific-region state with a GSP per capita of $50,000 is between 334 and 1177, with a best guess of 756 violent crimes per 100,000 people.

Note

As is always the case, the width of the prediction interval is larger than the width of the confidence interval. Remember why this is the case.

This page titled 7.2: Violent Crime, Wealth, Region is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

7.3: Cows in Děčín

To illustrate the process of **model selection**, let us examine the Děčín ballot measure of 2009. That ballot measure sought to constitutionally restrict the number of cows that can be housed within the city limits. While this extended example seems rather dated, it does cover some interesting issues in statistical modeling and questions we can answer with our model.

Learning Objectives

By the end of this section, you will be able to:

1. Conduct a complete multiple regression analysis, from preliminary data exploration and model selection to checking all key assumptions (normality, constant variance, functional form, multicollinearity).
2. Use R's formula grammar to specify and compare interaction and additive models, selecting a final model based on statistical significance and scientific theory.
3. Interpret the final model's coefficients, create graphs to illustrate the effects of predictors, and construct point estimates, confidence intervals, and prediction intervals for the response.
4. Use a parametric bootstrap (Monte Carlo simulation) to estimate the probability of a specific outcome (e.g., a ballot measure passing) by accounting for the uncertainty in the coefficient estimates.

✦•·····• ✦ •·····•✦

The Example

The voters of Děčín are being sent to the polls to vote on a constitutional referendum that proposes to limit the number of cows that could be housed within the city limits. This was not the first time that Ruritanians were sent to the polls to vote on this or a closely related issue. Given the information from previous votes, and the demographics of Děčín voters, what is the probability that this ballot measure will pass?

Before attempting any analysis, there needs to be a search of the literature to inform us as to which variables should be present, and which directions those variables should affect the dependent variable. From that literature review, we hypothesize that the vote in favor of such ballot measures depends on three variables: age of the population, religiosity of the population, and whether the ballot measure also restricts chickens. The effect direction for each is that kraj that are more religious should vote against cow-housing at a higher rate; Measures that also ban chickens should have a harder time passing; Measures passed more recently should have a more difficult chance of passing, as the young tend to support cows, and the elderly tend to oppose them (wanting quiet, dung-free neighborhoods).

With this theory and the resulting hypotheses, we can take our next step: **Getting to know the data**.

Get to Know the Data (The Preamble)

Before we begin trying to answer this question, we must get to know our data. There are several functions available to us to visualize the data: histogram, scatter plots, and quantile-quantile plots. In addition to visualizing the data, we should calculate several of the descriptive statistics for the variables of interest.

```
library(KnoxStats)
```

```
cows = read.csv("http://rur.kvasaheim.com/data/cows.csv")
summary(cows)
```

Here are some descriptive statistics on the variables in the `cows` dataset.

```
           Year Passed    Chicken Ban    Religious Percent
Minimum       1998           0                51.00
Maximum       2008           1                85.00
Median        2004           1                67.50
Mean          2004           0.5938           66.75
Variance      6.07           0.2490           88.19
Cv            0.58           0.8404            0.14
```

Variability

Since we have multiple independent variables, we should calculate both univariate and bivariate descriptive statistics. The table above provides the univariate descriptive statistics. The primary univariate question to ask about the independent variables here is whether there is sufficient variation. The two measures we need to examine are the variance and the coefficient of variation. If both of these numbers are small, then there may be an issue.

In this data, the variance of the Chicken Ban variable is small and potentially worrisome; however, its coefficient of variation (a scaled standard deviation),

$$c_v = \left| \frac{s}{\bar{x}} \right| \tag{7.3.1}$$

indicates that there is no serious issue (the value is close to 1).

Note

As this is a dichotomous variable, the mean is the percent of the values equal to 1. Thus, there are about 60% of the values 1 and 40% of the values 0 --- more than sufficient variation.

None of the three variables have small enough variation to cause us concern.

Relationships

After getting to know the variables individually, it is important to get to know the relationships between the variables. This can be done through correlation tests and bivariate scatter plots. Independent variables with strong correlations with the dependent variable should be considered for inclusion in the model. Independent variables with strong correlations with other independent variables should be of concern. Remember that one of the assumptions of OLS regression is that the independent variables are statistically independent of each other. If independent variables are highly correlated, the statistical properties of the method weaken.

Question:

> If independent variables are highly correlated, the statistical properties of the method weaken. Why?

The pairwise correlations are provided in the table below. Of the three independent variables, only Chicken Ban and Religious Percent have a statistically significant correlation ($t = 3.2869; \nu = 30;$ p-value $= 0.0026$). Should the level of correlation be a concern?

Perhaps. While their correlation is only $r = 0.5146$, this corresponds to an R^2 value of just 0.2648. As such, the correlation may not be large enough to severely affect our coefficient estimates. Let us just remember this relationship for the future.

Table 7.3.1: Covariance table

	Year Passed	Chicken Ban	Religious Percent
Year Passed		0.1903	0.2399
Chicken Ban	0.1903		0.5146
Religious Percent	0.2399	0.5146	

This table provides the correlations between the variables in the `cows` data. The correlation between Chicken Ban and Percent Religious is statistically significant ($t = 3.2869; \nu = 30; \text{p-value} = 0.0026$). This is the sole statistically significant correlation.

Note

The issue is actually more than a statistics issue. If two independent variables are highly correlated with each other, it is logically impossible to determine *which* affects the dependent variable or how much of the effect to partition to each independent variable. Statistics is, however, able to tease out the independent relationships better than not.

As a rule of thumb, if the correlation is greater than $r = 0.90$, there may be a **serious logical issue**. If two variables are so highly correlated, which of the two is the "correct" independent variable? How can one tell? Can both be good? Is the commonality between them the real independent variable?

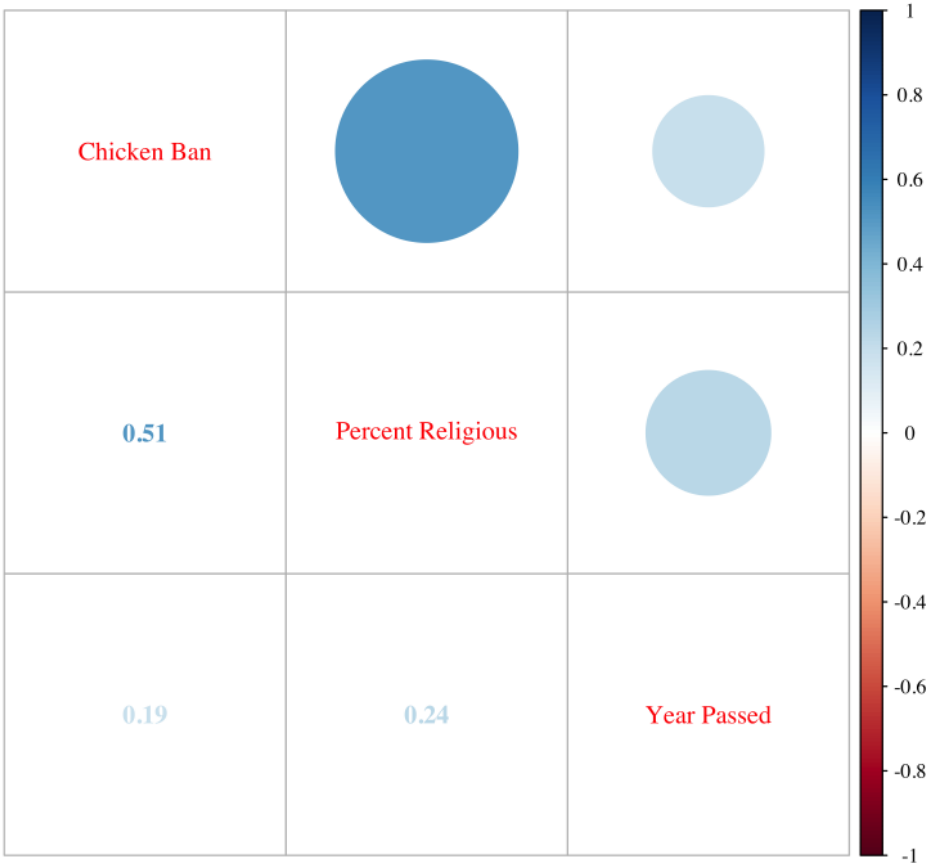


Figure **7.3.1**: Correlation plots between the three independent variables. The correlation between `Civil Ban` and `Percent Religious` was statistically significant according to the Pearson product-moment correlation test. This is evident in this graph, as well.

Variance Inflation Factor

This problem is a bit more extensive than suggested above. Recall that one of the mathematical requirements is that the rank of the design matrix equals p (the number of parameters to be estimated). This can happen if one variable is perfectly correlated with (linear function of) another variable. It can also happen if a variable is a linear function of the other variables.

Thus, while checking the bivariate correlations is helpful, it is not the answer we need. We need to check if the independent variable is a linear function of (or close to) a combination of the others.

To determine the level of multicollinearity we can use the "variance inflation factor" (VIF). Recall how to calculate the VIF for a given independent variable: Regress all other independent variables on it, calculate the R_i^2 and calculate the VIF from

$$\text{VIF}_i = \frac{1}{1 - R_i^2} \tag{7.3.2}$$

Once you have calculated the VIF for each independent variable, compare the values to the "usual" Rule of Thumb that depends on your discipline. The three typical boundaries are 5, 8, and 10. If all of your VIF scores are greater than 10, then there is an issue with multicollinearity. If all are less than 5, then there is no issue. If it is between those extremes, then you should think about the

effect of multicollinearity on your estimators. What do those VIF values correspond to? A VIF of 5 means the R^2 value is 0.80. A VIF of 10 means the R^2 is 0.90. Keep that in mind. In other words, the *other* independent variables explain 90% of the variation in this independent variable.

Of course, if we think about the effects of multicollinearity, rather than just detecting "severe levels," then we may wish to eschew such tests, assume multicollinearity is an issue, and adjust for it. On the other hand, if there is no reason to think that the model should suffer from multicollinearity, we will want to avoid such adjustments.

These are questions of science, not of statistics.

Know the science behind your theory.

Model the Data

The example asked us to determine the probability that the ballot measure will pass. Before we can answer that question, we need to model the proportion of the vote in favor of the ballot measure using our independent variables; that is, we need to be able to predict the proportion of the vote in favor of the ballot measure with the information we have.

Thus, the dependent variable will be `propWin` and the independent variables will be `yearPassed`, `chickens`, and `religPct`. For now, let us assume a linear relationship between the independent variables and the dependent variable.

Model Selection

Unless you have a lot of independent variables, I recommend you start with the interaction model.

Caution

Some will disagree and recommend starting with the simplest model and building complexity from that. There tends to be little difference between the two model-building methods. On either case, one has to worry about the multiple comparisons issue. How we should address it in the realm of model building is still unknown. We are certain of two things, however. First, the Bonferroni procedure is far too conservative. Second, doing nothing is not an acceptable option.

The interaction model includes the effects of each independent variable singly (main effects) as well as all possible combinations of those variables (interaction effects).

Table 7.3.2: The Grammar of Formulas

~	y ~ x1	Separates the dependent variable (left-hand side) and the independent variables (right-hand side).
+	y ~ x1 + x2	Indicates the following variable is added to the formula
-	y ~ x1 + x2 - 1	Indicates the following variable is removed from the formula. In the example, -1 removed the constant term.
:	y ~ x1:x2	Indicates the following and the preceding variable are multiplied in the formula
*	y ~ x1 * x2	Indicates the following and the preceding variable are crossed in the formula
^	y ~ (x1+x2+x3)^2	Includes the specified level of interactions.
I()	y ~ I(x1 + x2)	Replaces the formula grammar of what is in the parentheses with algebraic grammar. In this example, the sole independent variable is the sum of the two variables.

Table 7.3.1: The symbols and their meanings in the grammar of formulas. I sure wish I could locate the book that created these, but I cannot find it anymore. It is in the Oklahoma State University library... somewhere. It's a blue book, if that helps.

R uses the usual formula grammar (the table above). Its use takes a little practice to get the hang of, but it is entirely logical (for the most part). For instance, if you wish to fit the model $y = \beta_0 + \beta_1 x + \varepsilon$, you would use `y ~ x` .

However, if you wish to force the y-intercept to be 0, that is to fit the model $y = \beta_1 x + \varepsilon$, then you have a choice:

- `y ~ x - 1`
- `y ~ x + 0`

The two are identical.

The first is logically interpreted as the usual model, "less the intercept term" (hence the `-1`). This is the usual method I use. The second exists for backward compatibility. **I strongly encourage you to use the first method**.

Some other examples of this grammar include:

Table 7.3.3: Formula Examples

Algebraic Form	Formula Form
$y = \beta_0 + \beta_1 x1 + \beta_2 x2 + \beta_3 x1 x2$	`y ~ x1*x2`

$y = \beta_0 + \beta_1 x1 + \beta_2 x2 + \beta_3 x1 x2$	`y ~ x1 + x2 + x1:x2`
	
	`y ~ (x1 + x2)^2`

$y = \beta_0 + \beta_1 x1 x2$	`y ~ x1:x2`
$y = \beta_0 + \beta_1 x1^3 + \beta_2 \sin(x2)$	`y ~ I(x1^3) + I(sin(x2))`
$y = \beta_0 + \beta_1 x1 + \beta_2 x2 + \beta_3 x3 + \beta_4 x1 x2 + \beta_5 x1 x3 + \beta_6 x2 x3 + \beta_7 x1 x2 x3$	`y ~ x1*x2*x3`
$y = \beta_0 + \beta_1 x1 + \beta_2 x2 + \beta_3 x3 + \beta_4 x1 x2 + beta_5 x1 x3 + \beta_6 x2 x3$	`y ~ (x1 + x2 + x3)^2` `y ~ (x1 + x2 + x3)^2 - x1:*x2:x3`

Note that the first two examples above are the same model. These illustrate what comprises a "crossed" model. Also note that the "wedge" operator, `^`, indicates how many variables are multiplied in the terms. So `^2` indicates you want all one-way and two-way interactions for the model, and `^3` means you want all one-, two-, *and* three-way interactions.

Thus, we can see the following pairs are equivalent:

- `y ~ (x1 + x2 + x3)^3` and `y ~ (x1 * x2 * x3)`
- `y ~ (x1 + x2 + x3)^2` and `y ~ (x1 * x2 * x3) - x1:x2:x3`

With this brief introduction to the grammar of formulas, we can return to our example. We have three independent variables; the formula to give a full three-way interaction model is

```
propWin ~ yearPassed * chickens * religPct
```

As we will use this model a bit, we save the linear regression results into a variable. Thus, the two lines to run are

```
mod1 = lm(propWin ~ yearPassed * chickens * religPct)
summary(mod1)
```

These lines give the following output (well the first, fourth, and fifth column of that output):

```
                             t value   Pr(>|t|)
(Intercept)                    1.148      0.262
yearPassed                    -0.901      0.377
chickens                      -1.084      0.289
religPct                       1.557      0.133
yearPassed:chickens            0.950      0.352
yearPassed:religPct            0.510      0.615
chickens:religPct              0.979      0.338
yearPassed:chickens:religPct  -0.895      0.379
```

The line starting `yearPassed:chickens:religPct` is the three-way interaction term. As it is the highest interaction, it is the **only one** we can interpret here. Note that it is not statistically significant (**p-value = 0.379**). Thus, removing that term will do

two things. First, it will simplify the model. Second, it will not significantly harm the model's descriptive (or predictive) ability.

That second model can be written as either

```
mod2 = lm(propWin ~ yearPassed * chickens * religPct - yearPassed:chickens:religPct)
```

or as

```
mod2 = lm( propWin ~ (yearPassed + chickens + religPct)^2 )
```

The two formulas are equivalent.

Note that the `summary.aov(mod2)` command indicates that *none* of the three two-way interactions are statistically significant. Thus, these two-way interactions should be removed from the model.

Caution

Again, some would alternatively advocate removing just the least significant effect, then refit the new model. Others would suggest refitting with three different models, one for each combination of interaction. There is no "always best" answer, other than the one that your science suggests.

This leaves a model with no interactions — an additive model. Fitting the additive model and checking the statistical significance of the variables is as above.

```
mod3 = lm(propWin ~ yearPassed + chickens + religPct)
summary.aov(mod3)
```

Note that all three variables are significant according to this output (the chicken variable is statistically significant because we specified an effect direction). Thus, this is our **provisional model**.

Question:

Given that the additive model is the appropriate model, does the effect of religiosity `religPct`) change over time?

The Additive Model

That is, the equation we will use to fit the data is

$$\texttt{propWin} = \beta_0 + \beta_1(\texttt{yearPassed}) + \beta_2(\texttt{chickens}) + \beta_3(\texttt{religPct}) + \varepsilon \tag{7.3.3}$$

If $\varepsilon \sim N(0, \sigma)$, then we know

$$E[\texttt{propWin}] = \beta_0 + \beta_1(\texttt{yearPassed}) + \beta_2(\texttt{chickens}) + \beta_3(\texttt{religPct}) \tag{7.3.4}$$

Check the Assumptions

But, does this model violate any of the assumptions of OLS regression? All of the usual tests (Shapiro-Wilk, Breusch-Pagan, and runs) pass.

What about multicollinearity? Remember that the effect of multicollinearity is to inflate the standard errors (reduce the t-value, increase the p-value). Thus, if multicollinearity exists, fixing it will make the variables even more statistically significant.

How do we test it? We can do it the hard way or the easy way. The hard way is to estimate three regression equations, calculate the individual R^2 values, and calculate the resulting VIF values. So, by hand, this would be:

```
vif1 = lm(yearPassed ~ chickens + religPct)
vif2 = lm(chickens ~ yearPassed + religPct)
vif3 = lm(religPct ~ yearPassed + chickens)

1/(1-summary(vif1)$r.squared)
1/(1-summary(vif2)$r.squared)
1/(1-summary(vif3)$r.squared)
```

Or, you can use the `car` package:

```
library(car)
vif(mod3)
```

The results of these VIF checks are

```
yearPassed   chickens   religPct
  1.067965   1.368963   1.399954
```

None of these three is even close to the lowest "Rule of Thumb." As such, multicollinearity is not an issue in this model.

The Results

The regression table for model `mod3` is given below. Notice that all three variables of interest are statistically significant at the $\alpha = 0.05$ level. Additionally, the model has an $\overline{R}^2$ of 0.7565, which is a great fit in most of the social sciences. The direction of the coefficients also agrees with theory.

Caution

You may claim that the Chicken variable is not statistically significant at the $\alpha = 0.05$ level. However, the provided p-values are two-tailed (non-directional) p-values. Our hypotheses were all directional hypotheses (one-tailed). Thus, to get the one-tailed p-values you just halve the two-tailed p-values. With that, all three independent variables are statistically significant.

Thus, the equation for the line of best fit is approximately

$$\begin{aligned} E[\texttt{propWin}] = 0.1512 \\ &-0.0201(\texttt{yearPassed}) \\ &-0.0373(\texttt{chickens}) \\ &+0.0095(\texttt{religPct}) \end{aligned}$$

	Estimate	Std. Error	t-value	p-value
Constant Term	0.1512	0.0659	2.293	0.0295
Year Passed (post 2000)	-0.0201	0.0036	-5.618	<< 0.0001
Banned Chickens	-0.0373	0.0200	-1.868	0.0723
Percent Religious	0.0095	0.0011	8.801	<< 0.0001

Table 7.3.2: Results table for the regression of proportion support of a generic ballot limiting the number of cows housed in the city against the three included variables. The R^2 for the model is 0.7801; the $\overline{R}^2$, 0.7565. The p-values calculated are based on two-tailed test. The hypotheses were one-tailed hypotheses. As such, all three explanatory variables are statistically significant at the standard level of significance ($\alpha = 0.05$).

From this, we can make the following observations. First, the expected vote proportion is declining as time passes by about 2 percentage points per year. Second, ballot measures that also ban chickens are less likely to pass. Finally, the more religious a kraj, the more likely a cow ban is to pass.

Question:

What does the 0.1512 represent in this context (in real terms)?

Predicting Děčín

According to this model, what is the expected vote in Děčín? To answer this, we need information about Děčín and its ballot measure, specifically the value of the independent variables:

`yearPassed` = 9
`chickens` = 0
`religPct` = 48

With this information, and under the assumption that the model is correct, we have our prediction that 42% of the Děčín voters will vote in favor of this ballot measure.

Thankfully, R does not require us to do this calculation by hand. The R code for predicting the percent of Děčín voters voting in favor of this ballot measure can be

```
DECIN = data.frame(yearPassed=9, chickens=0, religPct=48)
predict(mod3, newdata=DECIN)
```

The first line was used to make the code more readable. It is also helpful to first define the variable `DECIN` if you are going to make predictions for Děčín using several models.

If neither of these methods appeals to you and you wish to do this in one line, that line would be

```
predict(mod3, newdata=data.frame(yearPassed=9, chickens=0, religPct=48))
```

Note the inclusion of the `predict` function, which predicts the dependent variable value given values for each of the independent variables (read the help file on `predict` ; we will use this function frequently).

Question:

How does the estimate change if you specify more variables than are in the model for Děčín? For instance:

```
DECIN = data.frame(yearPassed=9, chickens=0, religPct=48, age=45.2, channels=12,
turnout=0.72)
```

Graphing the Results

Now that we have confidence in our model, we can use it to predict the effects of each of the three independent variables on the vote in favor of these ballot measures. There are three independent variables, so we cannot create a single graph that displays the results. However, as one of the variables is dichotomous, we can show the results in just two graphs (the number of continuous independent variables).

Both of these graphs will have the vote in favor as the dependent variable (vertical axis). One of the two graphs will have percent religious as the primary independent variable, whereas the other will have the year passed as the primary independent variable. The chicken variable will be present in both graphs, signified by two separate curves, one where the ballot measure banned chickens and one where it did not:

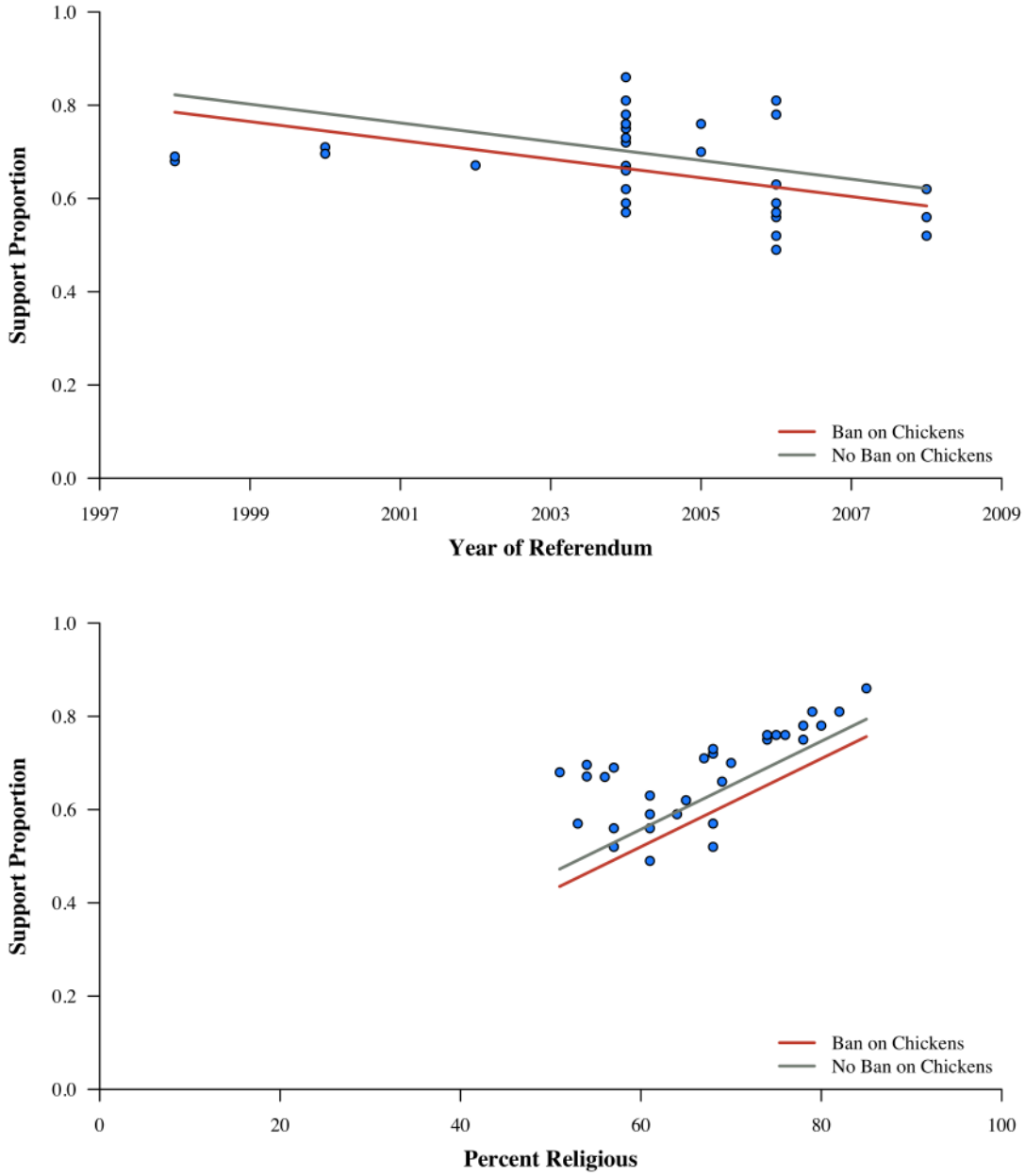


Figure **7.3.2**: Prediction graphs of our `COWS` model. These graphs contain two independent variables plotted against the dependent variable, with the dichotomous independent variable included as separate lines. Note that the effect of each of the three independent variables is made manifest by these two graphs.

The graphs illustrate the results of the model — this is their purpose. Although the graphs "illustrate the story," we must still "tell the story" of the graphics, including numbers from the prediction table. The following paragraphs explain the graphics.

> *Both graphics show that the effect of adding a chicken ban to the referendum tends to reduce the vote in favor of the referendum. All things being equal, a ballot measure banning chickens will have 3.7% fewer people vote for it than a like measure not banning chickens* ($s = 1.9988, t = -1.87$, p-value $= 0.0723$).
>
> *The top graphic illustrates the effect of passing time on the proportion of the vote in favor of these referenda: As the year increases by one, the proportion voting in favor of the*

> *referendum decreases by 2% on average* ($t = -5.62$, p-value $\ll 0.0001$).
>
> *The bottom graphic shows the effect of religiosity on the ballot outcome: those kraj with higher levels of religiosity tend to vote in favor of these measures at a higher level than kraj with lower levels of religiosity. In fact, increasing the level of religiosity in the kraj by 1% will tend to increase the vote in favor of the ballot measure by 0.95%* ($t = 8.80$, p-value $\ll 0.0001$).

Note the interweaving of the graphic discussion with concrete, numerical effects (and statistical significance in parentheses) from the prediction table. This combination aids the reader in interpreting the graphic(s) in terms of statistical language.

Answering the Question with the Parametric Bootstrap

Thus, we have a prediction of 42% of the voters will support the ballot measure. However, this is ***not*** an answer to the original question, which asked about the *probability* of the ballot measure passing. From a modeling standpoint, this probability depends on the coefficient estimates, which are just estimates of the true population value, and the standard errors, which are measures of our certainty in those estimates.

In the ordinary least squares method, those parameter estimates are random variables, since they are *functions of the data*. In other words, if we re-ran human history, the estimated effect would be different, since reality would be different. Furthermore, as these are random variables, they have an associated distribution --- the Normal distribution. In fact, the distribution of each parameter estimate is Normal, with expected value equal to the estimate and standard deviation equal to the standard error. Thus, for example, the effect of `yearPassed` is $b_1 \sim N(\mu = -0.0201, \sigma = 0.0036)$; of `chicken`, $b_2 \sim N(\mu = -0.0373, \sigma = 0.0200)$; and of `pctRelig`, $b_3 \sim N(\mu = 0.0095, \sigma = 0.0011)$.

Let us leverage these facts to (virtually) re-run human history many, many, many times, get the parameter estimates for each history, and predict the outcome of the ballot measure in Děčín. In other words, let us perform a Monte Carlo (simulation) analysis. The steps are the same as with any Monte Carlo analysis we have done (Kennedy 2008). The only difference is what we do within the loop. Here, we draw random numbers from the appropriate distribution and calculate the predicted vote.

Before you look at the following algorithm, write your own in your notes and compare it to the one below:

1. Initialize variables
2. Perform loop
    1. Draw from the four distributions
    2. Predict the Děčín outcome
3. Calculate the proportion of times the ballot measure garnered more than 50% of the vote

One can also store the random numbers inside the loop and predict outside the loop. Also, if the statistical program allows it, you can avoid the loop and just draw all the numbers at once. This last has the advantage of being *very* fast. It is also the method I use here, in the R script:

```
# Initialize variables
  outcome <- numeric()
  trials  <- 1e6
```

```
# Coefficient estimates
b.intc  <-  0.151221
b.year  <- -0.020095
b.cban  <- -0.037331
b.rpct  <-  0.009452

# Coefficient standard errors
s.intc  <-  0.065938
s.year  <-  0.003577
s.cban  <-  0.019988
s.rpct  <-  0.001074

# Distributions (the "loop")
e.intc  <- rnorm(trials, m=b.intc, s=s.intc)
e.year  <- rnorm(trials, m=b.year, s=s.year)
e.cban  <- rnorm(trials, m=b.cban, s=s.cban)
e.rpct  <- rnorm(trials, m=b.rpct, s=s.rpct)

outcome <- e.intc + e.year*9 + e.cban*0 + e.rpct*48
```

At this point, the variable `outcome` holds the proportion of people voting in favor of the ballot measure in one million simulated elections. To answer the question, we just need to determine the proportion of those elections in which the `outcome` is greater than 0.50: `mean(outcome>0.50)` will work.

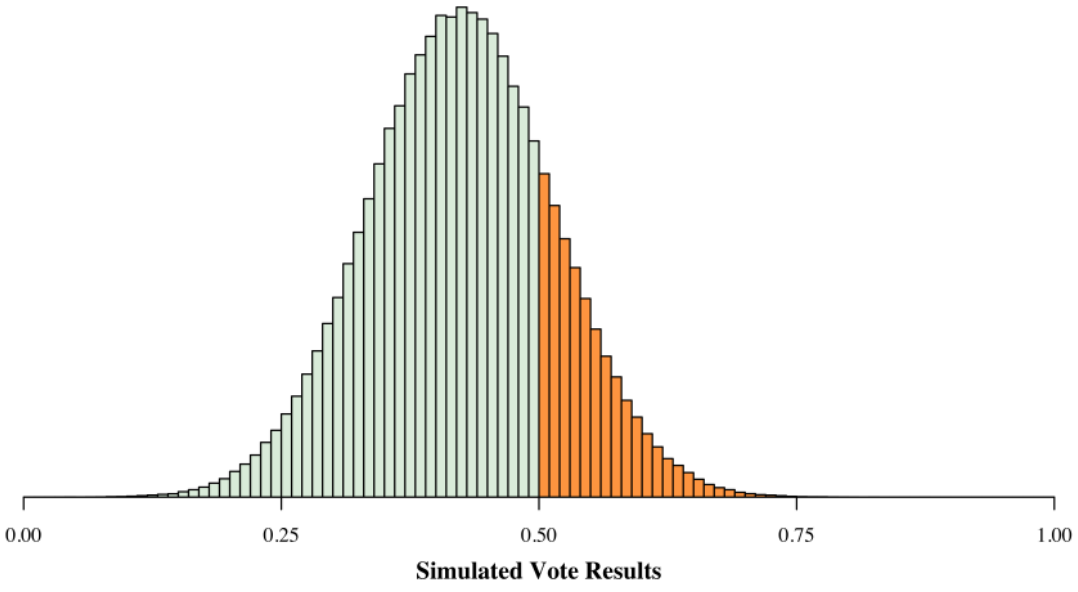


Figure 7.3.3: Plot of the predicted vote outcomes from the Monte Carlo experiment described in the text. Note that, while the expected proportion of the vote in favor of the ballot measure is 42%, there is still a 20% chance of the ballot measure passing, given that our model is correct.

Of course, the numbers are nice, but a histogram may tell a better story. The following code will produce a histogram similar to that above.

```
hist(outcome, main="", xlab="Proportion Vote for Ballot Measure", breaks=-1:99/100)
hist(outcome[outcome>0.50], main="", yaxt="n", breaks=-1:99/100, col=2, add=TRUE)
axis(1, at=0.50, labels="50%")
```

The histogram of the Děčín predictions is presented above (Figure **7.3.3**. Note that the expected outcome is still 42%, which we found above, but that there is a spread to that prediction the histogram makes manifest, which the single prediction did not. In fact, prior to this analysis, we may have concluded that there was no possibility that the ballot measure would pass in Děčín based on our model; now, we see that there is a 20% chance of the ballot measure passing.

Thus, we have an estimated answer to our original question. Given that our model is correct, there is approximately a 20% chance that the ballot measure to limit the number of cows in the city will pass in Děčín, with a point prediction of 42% in favor of the bill.

The actual results of the 2009 ballot measure in Děčín were that the ballot measure passed with 53% of the vote. This result is well within the 95% prediction interval suggested by Figure **7.3.3**. Also, the fact that the ballot measure passed should not be too surprising, since this model gave it a 20% probability of passing, and 20% is not a rare event by *any* stretch of the imagination.

A Fundamental Problem

There is a really big problem with these results, however. Run the following code and interpret it.

```
mean(outcome>1) + mean(outcome<0)
```

Always

This is an important exercise: **Always check that predictions make sense.**

In a future chapter, we will revisit this issue and propose a solution.

This page titled 7.3: Cows in Děčín is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

7.4: Conclusion

This chapter presented three comprehensive analytical examples demonstrating the practical application of linear regression techniques to real-world criminological data.

The chapter began with an analysis using simple linear regression to model a state's violent crime rate. This section showed the process of specifying a single-predictor model, examining the relationship between the violent crime rate in 2000 and in 1990. It covered model formulation, parameter estimation, interpretation of the slope and intercept, and assessment of model fit using the R^2.

The second analysis expanded the framework to ANCOVA, which combines regression and categorical factors. This section explored how to test for group differences (here, across the nine census regions) while statistically controlling for one or more continuous confounding variables (here, the covariate was the GSP per capita).

The final analysis introduced a robust, computer-intensive method by applying non-parametric bootstrap to a linear regression context. This technique was used to answer the actual research question about the likelihood of the ballot measure passing. Regression was only able to estimate the proportion of those votes in favor.

Collectively, these analyses illustrated a progression in analytical complexity — from a basic associative model to a controlled group comparison, and finally to a robust, resampling-based inference. The chapter emphasized not only the mechanics of each method but also their appropriate application, assumption checking, and interpretation of results within a coherent research narrative focused on understanding crime-related phenomena.

Caution

Again, be aware of the multiple comparisons issue discussed in The Appendix of Statistics. It explains why you need to either adjust your p-value or your alpha level when performing multiple tests, such as when you are testing *both* $\beta_0 = 0$ *and* $\beta_1 = 0$.

This page titled 7.4: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

7.5: End-of-Chapter Materials

Here are the expected materials to supplement the chapter.

R Functions

In this chapter, we were introduced to many, many, many R functions that will be useful in regression. In fact, this chapter uses more R functions than any other chapter in this book. Here are the many.

Packages

- `car`
  This package provides several statistical tests used in the book *An R Companion to Applied Regression* by J. Fox and S. Weisberg. It is a great package that provides a lot of additional functionality for R.
- `lawstat`
  This package provides several statistical tests used in law and public policy analysis. It provides the basic `runs.test` function for us.
- `lmtest`
  This package provides many tests related to linear models. It provides an implementation of the Breusch-Pagan test, `bptest`, which tests for heteroskedasticity in the residuals.
- `KnoxStats`
  This package adds much general functionality to R. Specifically, it improves upon the runs test in the `lawstat` package.

Statistics

- `source(filename)`
  This function runs an R script from a separate file. That file may be local or on the Internet.
- `lm(formula)`
  This is the function that performs ordinary least squares estimation on linear models.
- `aov(formula)`
  This function performs ordinary least squares estimation on linear models.
- `summary(x)`
  This produces the six-number summary or a frequency table of the provided variable, depending on the type of variable.
- `summary.lm(mod)`
  When applied to a linear model fit using either the `aov` function or the `lm` function, provides estimates of the *effects* of the numeric variables and the levels of the categorical variables in the model.
- `summary.aov(mod)`
  When applied to a linear model fit using either the `aov` function or the `lm` function, provides estimates of the statistical significance of the *variables* in the model.
- `shapiroTest(E)`
  This tests the null hypothesis that the variable `E` comes from a Normal distribution. It is based on the `shapiro.test` function in base R installation. It adds capabilities to test Normality in several groups.
- `fligner.test(formula)`
  This tests for heteroskedasticity when the independent variable is categorical.

- `bptest(mod)`
  This function from the `lmtest` package performs the Bresuch-Pagan test for heteroskedasticity.
- `runs.test(E, order)`
  This alteration to the `lawstat` function tests whether the variable `E`, as ordered by `order` exhibits fit issues.
- `vif(model)`
  This function calculates the variance inflation factor (VIF) for each of the independent variables in the model.
- `predict(mod)`
  This predicts the values of the dependent variable at each point in the dataset *or* for the values specified.
- `confint(mod)`
  This calculates confidence intervals for the parameters in ordinary least squares regression.
- `set.base(var,level)`
  This function redefines the base category in the provided level. By default, the base category is the first according to the alphabet.

Probability

- `set.seed(x)`
  This sets the random number seed. Doing so makes replication possible.
- `rexp(n, rate)`
  This generates n random values from an Exponential distribution with the specified rate parameter.
- `rnorm(n, mean, sd)`
  This generates n random values from a Normal distribution with specified mean and standard deviation. By default the mean is 0 and the standard deviation is 1.
- `runif(n, min, max)`
  This generates n random values from a Uniform distribution with specified minimum and maximum values. By default, the minimum is 0 and the maximum is 1.

Mathematics

- `head(x)`
  This returns the first six values in the variable.
- `foot(x)`
  This returns the last six values in the variable.
- `seq(from, to, by, length)`
  This returns a vector of sequential values, where `by` indicates the step size and `length` specifies the vector length. Only one of these two should be provided. If neither is provided, then `by` defaults to 1.
- `length(x)`
  This calculates the length of a vector (variable), which is the sample size, n.
- `residuals(mod)`
  This calculates the residuals in the model, which is the difference between the observed and the predicted.

Graphics

- `qqnorm(x)`
  This creates a Normal quantile-quantile plot for the given values.
- `qqline(x)`
  This adds the diagonal line to the quantile-quantile plot.
- `plot(x,y)`
  This produces a scatter plot of the y-values against the x-values.
- `overlay(x)`
  This produces a histogram with a Normal curve overlaying it. Technically, there are several possible overlays, but the Normal curve is the default.
- `par(...)`
  This sets parameters for the next graphic started. Look through the help page for this function to see all you can specify.
- `plot.new()`
  Creates a blank, new plot.
- `plot.window(xlim, ylim)`
  Specifies the limits for the x- and y-axes.
- `axis(side)`
  When a plot is already drawn, this adds values along axis number `side` .
- `title(...)`
  When a plot is already drawn, this adds the x- and y-labels.
- `lines(x,y)`
  When a plot is already drawn, this draws lines between each subsequent (x, y) pair.
- `points(x,y)`
  When a plot is already drawn, this draws points at each (x, y) pair.

Programming

- `library(package)`
  This loads an external package that you have already installed on your computer. It allows access to all functions and data sets in the `package` package.
- `attach(dataframe)`
  This allows you to access the variables in the `dataframe` without having to prefix each with `dataframe$` .
- `as.character(x)`
  This changes the values in variable `x` to be characters.
- `as.numeric(x)`
  This changes the values in variable `x` to be numbers.

Exercises

1. In the two panels in Figure 7.3.2, the lines of best fit do not go beyond the data. Why?

2. Section 7.3 ended by stating that there was a really big problem with those results. Run the following code.

```
mean(outcome>1) + mean(outcome<0)
```

What value is given, what does it mean, and why does it imply there is something fundamentally wrong with the analysis?

This page titled 7.5: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

8: Fixing the Violations



In Chapter 6: Dood! Check the Requirements, we examined the assumptions of ordinary least squares and how to check that they are not violated by your model. The requirements (assumptions) have different importance to our estimation method. The most important requirement is that the model uniformly fits the data (constant expected value of the residuals). In this chapter, we see some ways to fix those violations.

Much of this chapter will deal with transforming the dependent variable, because misidentified models is the greatest problem in modeling. Frequently, fixing this problem also fixes other problems with assumption violations.

In the previous chapters, we introduced the ordinary least squares (OLS) estimation method for the classical linear model (CLM) — and its assumptions (requirements). In the previous chapter, we looked at how to test that the requirements are sufficiently met in our data and model. We also looked at the importance of the assumptions. In this chapter, we determine some methods for dealing with *some* violations of those requirements. Hopefully, this extends the usefulness of this simple and straightforward estimation method.

Recall that the ordinary least squares estimation method (OLS) requires that the error terms have a constant expected value, have a constant variance, and are generated from a Normal (Gaussian) process. But, what happens when these requirements are *not* met?

There are essentially three ways of handling violations depending on the type and the severity: **First**, you can ignore it. Ignoring the violations is usually not *too* bad when you are dealing with predicting within the domain of the observed data, as the increase in bias and the loss of efficiency are usually minor. However, if it is important to estimate parameters, you definitely should not ignore this violation. Furthermore, if the assumption of a constant expected value is practically violated, you *need* to fix it.

Second, we can use other methods (and modeling paradigms) to perform regression. Two popular alternatives to the Classical Linear Model paradigm are the Generalized Linear Model (GLM) and the Generalized Additive Model (GAM). The former paradigm will be covered in Chapter 14: Generalized Linear Models and beyond. The latter is well examined in Wood (2006). The strength of these models (and estimation methods) is that they extend the CLM to include (for instance) discrete dependent variables and non-linear relationships (Nelder and Wedderburn 1972; Wood 2006). These unified paradigms allow the computer to estimate the effect coefficients using a very powerful method (called Maximum Likelihood Estimation). The drawback is that not all problems lend themselves to fitting using Maximum Likelihood Estimation. Luckily, most do. Even more luckily, new estimation methods are developed frequently.

However, if we desire to stay *within* the realm of the classical linear model, estimating the parameters using ordinary least squares, we can adjust for many violations simply by transforming the dependent variable — especially if the violations are minor.

These transformations are very flexible. Once you get used to working in two different systems of units, you can easily use transformation methods to "Normalize" many restricted dependent variables. Unfortunately, one cannot transform an arbitrary dependent variable; there are types that cannot be fit using this technique, such as categorical. To handle these types of dependent variables, we will need to introduce a new modeling paradigm.

Finally, you can make adjustments to the estimates and their standard errors to "fix" or "adjust for" the violation. This is a common practice in the presence of heteroskedasticity and of multicollinearity.

Unfortunately, these do *not* work for violations of model fit (non-constant expected residuals)... which is the most important of the requirements.

· · — · ✶ · — · ·

Chapter Sections

8.1: The Issue of Boundedness
8.2: The South Sudanese Referendum
8.3: Adjustments for Heteroskedasticity
8.4: Conclusion
8.5: End-of-Chapter Materials

This page titled 8: Fixing the Violations is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

8.1: The Issue of Boundedness

The ordinary least squares (OLS) model requires that the range of the dependent variable $\boldsymbol{Y}$ is from $-\infty$ to ∞. Yet in the social and health sciences, analysts routinely encounter outcomes that are logically or practically constrained — from proportions and percentages (bounded at 0 and 1) to Likert rating scales (e.g., 1–7) or counts with a clear upper limit (e.g., number of correct items on a 10-question test) to counts with no upper limit (e.g., dents in a car or mistakes in a book). Applying standard OLS to such "bounded" data raises a significant concern: the model, being linear and unbounded, can produce predicted values that lie outside the possible range of the actual data, rendering them nonsensical and compromising inference. We saw this at the end of Section 7.3: Cows in Děčín.

This section addresses this common dilemma. When is it permissible to use the simplicity and interpretability of OLS when Y is bounded, and what adaptations or diagnostics are required? We will focus on practical strategies for once-bounded (e.g., $\boldsymbol{Y} \geq 0$) and twice-bounded (e.g., $0 \leq \boldsymbol{Y} \leq 1$) outcomes. While specialized alternatives like Tobit, beta, or fractional regression exist, the techniques covered here provide a principled approach for determining when OLS remains a robust and informative tool, and how to implement it correctly when it does. The key lies not in abandoning OLS outright (at least not yet), but in understanding its limitations, checking for critical pathologies, and knowing when a transformation or adjusted interpretation can salvage its utility.

Note that the techniques covered in this chapter are useful beyond ordinary least squares. Frequently, you will have data that you want to model using Technique X, but that technique requires an unbounded dependent variable. (OLS is not the only one!) You can use these techniques to adjust your data, while still using Technique X.

Learning Objectives

By the end of this section, you will be able to:

1. Identify the problem of bounded dependent variables (e.g., proportions between 0 and 1, or non-negative values like income) in linear regression, recognizing that unbounded model predictions can fall outside the physically possible range.
2. Apply appropriate transformations to map bounded variables to an unbounded scale—using the logit transformation for variables bounded by 0 and 1, and the logarithmic transformation for variables bounded below by 0.
3. Perform a complete regression analysis on the transformed variable, including model fitting, assumption checking, and interpretation of coefficients on the transformed scale.
4. Back-transform predictions, confidence intervals, and simulation results (e.g., from a parametric bootstrap) to the original units of measurement to answer the original research question.

✦•·····• ✦ •·····•✦

We finished Chapter 7 with a model of vote proportions for ballot measures concerning keeping cows in the city. We applied that model to an upcoming vote in Děčín to predict the outcome. Finally, we used Monte Carlo methods to estimate the probability that the ballot measure would pass. In the end, we predicted that the ballot measure had a 20% chance of passing, with a point-prediction of 42% of the voters in favor of the bill.

Results, however, suggest that there may be something gravely wrong with this model. To see this more clearly, let us predict the proportion of voters in support of a hypothetical 1994 ballot measure in Venkovský (religious percent = 85) that also banned chickens.

From the results summarized in the table, the point-prediction for this 1994 Venkovský ballot measure is

$$\begin{aligned}\hat{p} &= 0.1512 + -0.0201(\texttt{yearPassed}) + -0.0373(\texttt{chicken}) + 0.0095(\texttt{religPct}) \\ &= 0.1512 + -0.0201(-6) + -0.0373(1) + 0.0095(85) \\ &= 1.0379\end{aligned}$$

Thus, this model predicts that the ballot measure will pass with over 103% of the vote — a physically impossible outcome. What went wrong? How can we fix this model so that this cannot happen?

First, nothing "went wrong," *per se*. The model did *exactly* what it was supposed to do. The prediction, however, is based on assuming the effect (slope) is *constant*. If the slope is constant, one can find large enough (or small enough) values for the independent variables to make the prediction arbitrarily large or small. When we are predicting a bounded dependent variable, this will necessarily lead to an impossible prediction, such as a 103.79% support rate.

Thus, the issue is either with the linear (constant slope) aspect of the prediction equation *or* with the bounded nature of the dependent variable (bounded below by 0 and above by 1).

So, to improve the model, we can either model using non-linear coefficient functions or eliminate the boundedness. At this point, the easier of the two is to eliminate this boundedness; that is, we need to change the dependent variable so that *all* values make physical sense. This is done through the process of **variable transformation**. There are three steps:

1. First, transform the dependent variable from a restricted range to an unrestricted range.
2. Second, perform the analysis on this transformed variable.
3. Finally, back-transform the estimated values (not estimated effects) into the original units.

The key is the transformation. It must change the range of Y from its current limited version to an unlimited version, denoted $\widetilde{Y}$. Luckily, there are two transformations that take care of most of our needs, in general: the logit (*LOH-jit*) and the logarithm transformations.

Data Bounded by 0 and 1

One type of data you may come across in your research is proportion data, data where the values are bounded below and above (by 0 and 1, respectively); that is, if Y is the dependent variable, then $0 \le Y \le 1$). One appropriate function that transforms this bounded domain into an unbounded range is the **logit function**:

$$\tilde{y} = \text{logit}(y) \stackrel{\text{def}}{=} \log\left(\frac{y}{1-y}\right) \tag{8.1.1}$$

The logit function transforms (maps) variables bounded by 0 and 1 into unbounded variables; in symbols,

$$\text{logit} : (0,1) \mapsto \mathbb{R}$$

The logit's inverse, which maps it from logit units back into level units is called the logistic function:

$$y = \text{logistic}(\tilde{y}) \stackrel{\text{def}}{=} \frac{1}{1+\exp(-\tilde{y})} = \frac{\exp(\tilde{y})}{1+\exp(\tilde{y})} \tag{8.1.2}$$

The logistic function transforms unbounded variables into variables bounded by 0 and 1:

$$\text{logistic} : \mathbb{R} \mapsto (0,1)$$

The figure below shows a graphic of the logistic function. The logit is the inverse.

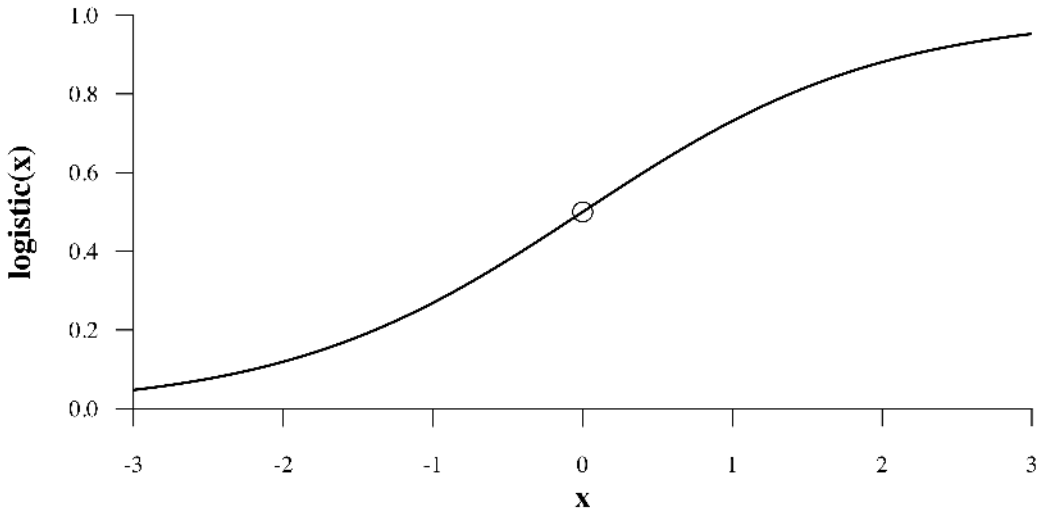


Figure **8.1.1**: Graphic of the logistic function. The logit function is the inverse of the logistic. Note that the graph is symmetric about the point (0, 0.5).

While other transforms are available (and will be covered when we get to GLMs), the logit is frequently used for the following three reasons:

1. The transformation *and* its inverse are both functions (the transform is a bijective function). This means that the results are always commensurate to the original problem.
2. The transformation is symmetric. This means that stretching above 0 is the same as below.
3. The function is exact, as opposed to the probit transform that requires numerical approximations. This increases the speed and accuracy of your predictions.

A careful reader will note that the domain of $\boldsymbol{Y}$ includes *neither* 0 nor 1. This is because there is no way of transforming a closed (or a half-closed) interval into an open interval such as $\mathbb{R}$ while ensuring that the inverse is also a function. This is a provable fact of mathematics (Strichartz 2000).

Data that are Either 0 or 1

But, what do we do if there are y-values that *are* zero or one? One solution is to add (subtract) an extremely small number, δ, to the zero (one). A second solution is to completely drop those data from the analysis.

Caution

Neither of these solutions is perfect. *If* you insist on using linear regression, then you should use both methods and see how much your answer changes.

A general rule of thumb is that if your underlying research model is correct, then the results should not vary wildly based on similar models. That is, if we know $\boldsymbol{Y}$ depends on $\boldsymbol{X1}$ and $\boldsymbol{X2}$, then all appropriate modeling techniques should give *approximately* the same results. If they do not, then there is something seriously wrong with our assumptions about the underlying relationships — the model.

A third solution is to change the proportion into a bounded count and use a different paradigm (Chapter 15: Binary Dependent Variables). While this is the best option, it requires a bit more background before we can cover it.

Example 8.1.1: Voting in Děčín

Let us return to the `cows` data file and the example of Section 7.3: Cows in Děčín. The voters of Děčín are being sent to the polls to vote on a constitutional referendum that proposes to limit the number of cows kept in the city. This was not the first time that Ruritanians were sent to the polls to vote on this or a closely related issue. Given the information from previous votes, what is the *probability* that this ballot measure will pass in Děčín?

Solution.

Let us now answer this question more correctly. Recall that without performing a transformation of the dependent variable, there existed predictions that fell outside possible reality. To fix this, let us transform the dependent variable using the logit function, repeat the analysis, back-transform these transformed results to the original units, and compare the results.

The first step is to transform the dependent variable. As the dependent variable is a proportion, let us use the `logit` transform (from the `KnoxStats` package). If we decide to call the new variable `logitWin`, then the command will be

```
logitWin = logit(propWin)
```

Now, this is our new dependent variable. As such, we perform the same analysis as in Section 7.3: Cows in Děčín:

```
modLgt = lm(logitWin ~ yearPassed + chickens + religPct)
```

The `summary(modLgt)` command provides the results summarized in the table below.

```
                          Estimate   Std. Error   t-value      p-value
Constant Term              -1.8909       0.2898     -6.53    << 0.0001
Year Passed (after 2000)   -0.0885       0.0157     -5.64    << 0.0001
Contains a Chicken Ban     -0.2318       0.0878     -2.64       0.0134
Percent Religious in Kraj   0.4750       0.0047     10.06    << 0.0001
```

■

Interpreting the Coefficients ✨

How shall we interpret the results? There are a few ways. As always, a good graphic is the best.

However, an older manner relies on the "**log odds ratio**." The odds ratio is frequently used to illustrate the strength of the association between two variables. For every increase of 1 in the "percent religious in the kraj," the log of the odds of the vote passing increases by 0.4750. Said another way, the *odds* of the ballot measure passing increases by approximately 60.80% for each increase of 1 percentage point in religiosity. [Note: $\exp[0.4750] = 1.6080$.]

An increase of 2pp in religiosity increases the odds by about 157%. [Again, note: $\exp[2 \times 0.4750] = 2.5857$.] Thus, if the original odds were 3-to-1 against, increasing the religiosity by 2pp means that the odds are now about 7.75-to-1 against.

Note

Beyond this, one *cannot* directly compare the magnitudes of these coefficients with the magnitudes of the previous coefficients; these effect estimates are in different units. The coefficients seen in the original model predict in the original units (proportions). The coefficients in the logit model predict in logit (of proportions) units.

> Furthermore, merely taking the logistic of the coefficients will not put them in level units; the transform is non-linear, as we designed, thus the effect of *any* variable depends on the values of *all* variables. To compare the two models, we need to perform predictions (remembering to back-transform them).

Predicting the proportion of the vote for the Děčín ballot measure is almost as easy as it was before. The only additional step is that we need to **back-transform** the prediction to get it in proportion units.

So, according to this transformed model, what is the expected vote in Děčín? To answer this, we again need the Děčín information:

```
yearPassed = 9
chickens   = 0
religPct   = 48
```

With this information, and under the usual assumption that the model is correct, we have our prediction of -0.4091 logits. Back-transforming this value gives a prediction of logistic(-0.4091)=40% of the population will vote in favor of this ballot measure — just slightly different from our original prediction of 42%.

```
DECIN = data.frame(yearPassed=9, chickens=0, religPct=48)
voteLgt = predict(modLgt, newdata=DECIN)
voteEst = logistic(voteLgt)
```

However, remember that the original question was not this point estimate, it was a *probability* of the ballot measure passing. To determine this probability, we just need to repeat the same steps as we did answering this question before, but remembering to back-transform the results.

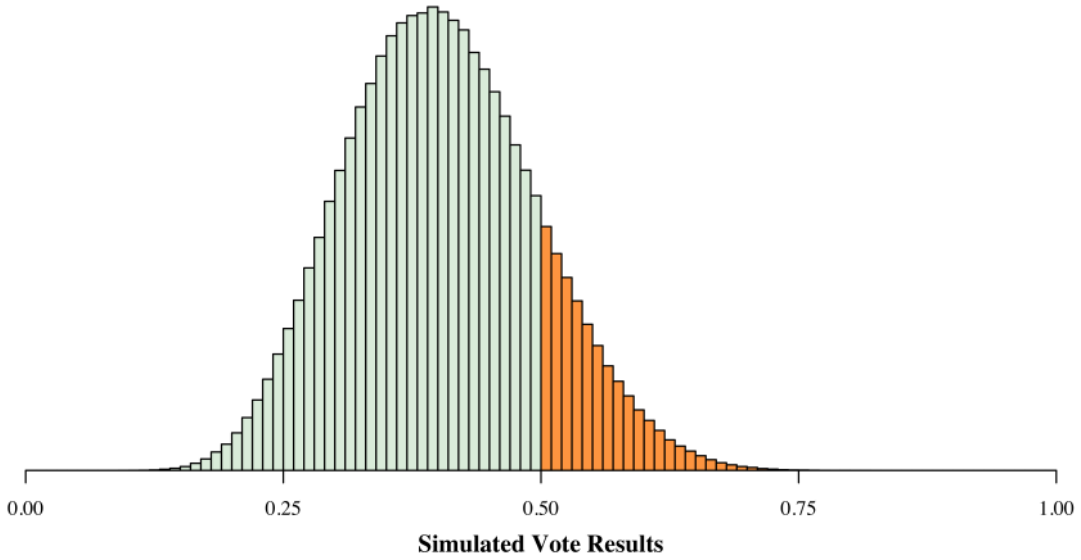


Figure 8.1.2: Histogram of the results of the Monte Carlo experiment described in the text. Note that the distribution has a slight right-skew as a result of the transformation process. Also note that there are no predicted vote outcomes less than 0 or greater than 1, as compared to the original untransformed model of Section 7.3: Cows in Děčín. In fact, the lowest prediction is 9.0%, while the largest is 81.6%.

The Monte Carlo results of the transformed model indicate that there is a 15% chance that the ballot measure will pass in Děčín. The histogram of a million predictions is presented in Figure 8.1.2. From this information, we can conclude that there is a definite possibility that the cow ballot measure will pass in Děčín (15%), with a predicted 40% vote in favor.

If we were into betting, we could also conclude that this model predicts that the **odds** of this ballot measure passing are

$$\frac{1-p}{p} = \frac{1-0.15}{0.15} \tag{8.1.3}$$

Which is about 5.67-to-1 *against*. Thus, a "fair" bet would pay $5.67 for every $1.00 bet in favor of the ballot measure and $1/5.67 = $0.176 for every dollar bet against the ballot measure passing.

Regardless, since the probability of the measure passing is 15%, a pass would not be wholly unexpected. Its passing is more likely than flipping a fair coin three times and having it come up heads all three times (15% vs. 12.5%) — definitely not unheard of.

The 95% prediction interval for the Děčín referendum outcome, according to our model, is from 23.5% to 59.0%. The observed value of 53% is within that interval.

Note

From this past discussion, we were able to estimate success probabilities and fair betting odds. This is yet another use of statistical modeling.

Note that we are estimating the probability of an event. Unless that probability is 0 or 1, there is always a chance the event will (or will not) happen. Thus, the passing of the Děčín referendum in 2009 does not directly detract from our model. There was a 15% chance it would pass, according to our model.

Caution

Stay aware of what your statistical model says **and does *not* say** — the choice is humility or humiliation.

Data Bounded Below by 0

When the dependent variable represents a proportion (bounded by 0 and 1), we can use the logit function to transform it into an unbounded variable, perform the usual analysis, and back-transform those results into level units (the previous section). However, not all bounded variables fit this bounding, e.g., age, height, and income. These variables are bounded below by 0 and have no theoretical upper bound. For such variables, we may want to use the **logarithm transform**.

The logarithm function transforms variables bounded below by 0 into unbounded variables; in symbols,

$$\log : (0, \infty) \mapsto \mathbb{R}$$

Its inverse is the exponential function,

$$\exp : \mathbb{R} \mapsto (0, \infty)$$

Both functions are bijections and strictly increasing and so are appropriate functions for transforming our variables.

Data that *are* 0

Note that values of 0 are problematic for the logarithm in much the same way that values of 0 and 1 were problematic for the logistic function. The solutions are similar.

Example 8.1.2: Wealth and Corruption

The gross domestic product (GDP) per capita is one of many measures of average wealth in countries. If extant theory is correct, then the wealth in the country is directly affected by the level of honesty in the government — countries with high levels of honesty (low levels of corruption) should be wealthier than those with low levels of honesty (high levels of corruption). Furthermore, if theory is correct, the level of democracy in a country should *also* influence the country's level of wealth — countries with higher levels of democracy should be wealthier than countries with lower levels of democracy.

Let us determine if reality (in the form of the data in the `gdp` data file) supports the current theory or if current theory needs to explain the severe discrepancies. Furthermore, let us estimate the GDP per capita for Ruritania and provide a 95% confidence interval for that estimate.

Solution.

For this section, recall that the level of honesty in government for Ruritania is 5.1 and the level of democracy is -7. With that information, I leave it as an exercise for you to model the data *without* transforming the dependent variable and discovering the predicted GDP per capita for Ruritania is $26,795.64. This seems awesome for Ruritania. The 95% prediction interval is from $5232 to $48,360. That's rather wide. It is a function of the high level of variation in the data.

However, to see **a problem with the model**, let us estimate the GDP per capita for Papua New Guinea (democracy=10, hig=2.1). According to the model, the predicted GDP per capita is -$2337, which is not physically possible. If nothing else, this prediction should suggest to you that the data needs transformation before being modeled.

The process to estimate the GDP per capita in Ruritania using a **transformed model** is formulaic for us by now: transform the dependent variable by applying the logarithm function, model the transformed variable, estimate in the transformed units, back-transformed into level units — here, dollars.

One feature of R that is shared by few other statistical packages is that you do not *have to* actually create a new variable; you *can* perform the transformation within the modeling command:

```
modLog = lm(log(gdpcap) ~ democracy + hig)
```

I'm not sure this is a strength. I prefer to clearly define my variables elsewhere.

The results table for this model is provided here.

```
                        Estimate   StdErr   t-value     p-value
Constant term             6.9333   0.1479     46.89   << 0.0001
Level of Democracy       -0.0028   0.0113     -0.25      0.8055
Honesty in Government     0.4702   0.0359     13.11   << 0.0001
```

Again, as we have transformed the dependent variable, the coefficients are *not* in units of dollars. As such, their magnitudes cannot be directly compared to those in the untransformed model. Their *directions*, however, can be compared because the transformation we used was strictly increasing. Thus, this model tells us that higher levels of honesty in government correspond to countries with higher GDPs per capita (in this sample). Additionally, countries with higher democracy scores correspond to countries with *lower* GDPs per capita (in *this* sample).

The first finding is so strong in this sample that we can conclude that there is evidence of this relationship *in the population*. This second finding, which conflicts with current theory, is not statistically significant at the usual $\alpha = 0.05$ level. Thus, we cannot

conclude that the effect in the population is negative, positive, or null (zero). All we can conclude is that we did not detect an effect *with this data*. Whether this is due to a lack of effect in the population, the sample selected, the sample size, no one can tell.

With this model, we can estimate the GDP per capita in Ruritania using the standard method, but remembering that we must **back-transform** the final estimate. That is, if we used the commands

```
RUR = data.frame(hig=5.1, democracy=-7)
estLog = predict(modLog, newdata=RUR)
```

then we would report Ruritania's GDP per capita as an estimated value of $11,508 (using `exp(estLog)`).

■

Interpreting the Coefficients ✨

From your mathematics course, you may recall that $\log(1+x) \approx x$ for small values of x. This means we can interpret the coefficients in the log-model as percent increases/decreases. For instance, the coefficient for the level of democracy in the country is -0.0028. We can interpret this as "one increase in the level of democracy decreases the GDP per capita by about 0.28%, on average." The coefficient of the level of honesty in government is 0.4702. We could interpret this as "one increase in the level of honesty in the government increases the GDP per capita by approximately 47%, on average."

What do we mean by "small values of x"?

Anything less than 0.20 is usually fine. As such, our interpretation of the honesty-in-government coefficient probably should not have been done. A log-coefficient value of 0.4702 really corresponds to a percent increase of only 38.5%. It is more accurate, but less easily calculated.

Here is my code to explore the relationship $\log(1+x) \approx x$:

```
x = seq(0,1, length=1e4)
y = log(1+x)
plot(x,y, col="blue1")
abline(0,1, col="orange")
```

▪——— ⚔ ———▪

The question asked us to calculate the estimate, but to also provide a 95% confidence interval. One way of doing this is to use Monte Carlo methods. The steps are all the same, with the additional step of back-transforming the estimates (last line).

Here is the code for parametric bootstrapping:

```
b.int   =  6.933298
b.dem   = -0.002776
b.hig   =  0.470225

s.int   =  0.147873
s.dem   =  0.011253
s.hig   =  0.035855
```

```
e.int   = rnorm(trials, m=b.int, s=s.int)
e.dem   = rnorm(trials, m=b.dem, s=s.dem)
e.hig   = rnorm(trials, m=b.hig, s=s.hig)

outcome = e.int + e.dem*-7 + e.hig*5.1
est     = exp(outcome)
```

The assignments in the second and third groups are the coefficient estimates and standard errors from the model. The histogram of these results is provided in Figure **8.1.3**. To calculate a 95% confidence interval, we merely find the values of `est` for which 2.5% and 97.5% of the data are less.

```
quantile(est, c(0.025,0.975))
```

From this, we can conclude that our model estimates the GDP per capita for Ruritania is $11,508, with a 95% confidence interval being from $7075 to $18,733. It is interesting to note that the actual GDP per capita in Ruritania is $55,000, which is well above our confidence interval. Thus, our question is this: Is our model that weak, or is Ruritania doing that well?

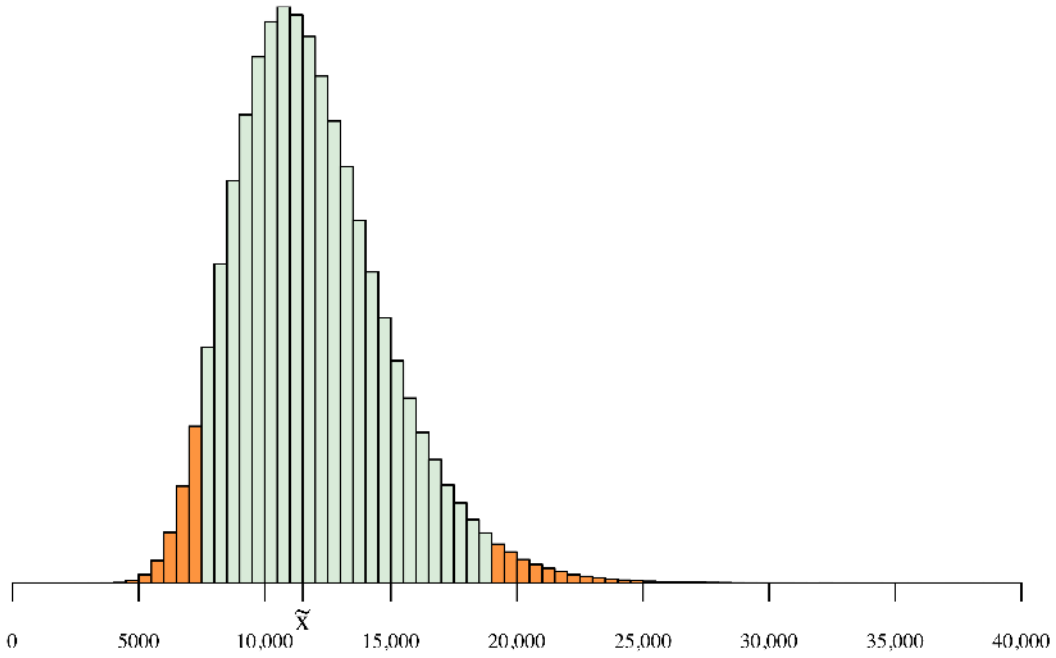


Figure **8.1.3**: Results of the Monte Carlo experiment estimating the GDP per capita for Ruritania and its 95% confidence interval. Note that 5% of the estimates fall in the rejection (orange) region, 2.5% above and 2.5% below. The median of this distribution is designated by $\tilde{X}$.

Note

Here, I use the original estimate as the point estimate for the GDP per capita of Ruritania ($11,508). It would have also been appropriate to use the mean of the Monte Carlo trials ($11,870) *or* the median of the Monte Carlo trials ($11,510). All three are acceptable measures of the center. It is usual, however, to use the original prediction.

Here is an interesting **question**:

> In the previous example, we estimated a confidence interval. How could we estimate a prediction interval?

To answer this, we need to remember the *only* difference between confidence and prediction intervals. In a confidence interval, we are estimating an expected value. In a prediction interval, we are predicting a *new* outcome. That new outcome is a combination of the expected value *and* the σ^2 from the ε term.

And so, to get a prediction interval, we use the following. Check to see the difference between this and the previous script.

```
b.int   =  6.933298
b.dem   = -0.002776
b.hig   =  0.470225
b.err   =  0

s.int   =  0.147873
s.dem   =  0.011253
s.hig   =  0.035855
s.err   =  0.8841

e.int   = rnorm(trials, m=b.int, s=s.int)
e.dem   = rnorm(trials, m=b.dem, s=s.dem)
e.hig   = rnorm(trials, m=b.hig, s=s.hig)
e.err   = rnorm(trials, m=b.err, s=s.err)

outcome = e.int + e.dem*-7 + e.hig*5.1 + e.err
est     = exp(outcome)
```

From this, the 95% prediction interval is from \$1907 to \$69,345. Note that it is *much* wider than the confidence interval. Also, note that this should not surprise us at all. Prediction intervals are always wider than the corresponding confidence interval.

Additional Bounds

Thus far, we have looked at transforming the dependent variable when it is bounded above and below by 0 and 1 (two bounds), and when it is only bounded below by 0 (one bound). Other bounds are possible. In this section, we figure out how to handle all types of bounds. The basic steps are to determine if the variable is bounded on one side or two. If one, then perform an **algebraic transformation** so that the new variable is bounded below by 0, then use the log transform. If two, then perform an algebraic transformation so that the new variable is bounded by 0 and 1, then use the logit transform. In either case, you will need to remember to back-transform the predictions with this algebraic transformation.

> **Note**
>
> The only bounds I frequently come across in my own research are those bounded by 0 and 1, bounded by 0 and 100 (percentages), bounded by 0 and 4 (GPAs), and bounded below by 0. The quick solution for percentages is to divide them by 100 to make them proportions, then multiply the predictions by 100 to turn the predictions back into percentages.

Bounded by L and U

What if our data has a theoretic lower bound L and a theoretic upper bound U? As it is bounded above *and* below, we will change it into a proportion and use the logit transform as above. The algebraic transformation is

$$a(y) = p = \frac{y - L}{U - L} \tag{8.1.4}$$

The back-transform is

$$a^{-1}(p) = y = p(U - L) + L \tag{8.1.5}$$

Example 8.1.3: GRE Quant Score

The scores on the quantitative portion of the Graduate Record Examination (GRE) range from $L = 130$ to $U = 170$.

If we wished to properly model a person's GRE quantitative score, we would first subtract 130 from each score, then divide by $170 - 130 = 40$. The new variable would range from 0 to 1, a proportion.

Example 8.1.4: GPAs

The grade point averages (GPAs) at Knox College are bounded below by $L = 0$ and above by $U = 4$.

To appropriately model GPAs, we would have to subtract 0, then divide by 4. This new variable would now be a proportion.

Bounded Below by L

It may be that your dependent variable is bounded below by a specific value, L, but not bounded above. As it is bounded on only *one* side, we will transform it into a variable **bounded below by 0** and then apply the logarithm transform as above, remembering to back-transform with the additional transformation. The algebraic transformation is

$$a(y) = p = y - L \tag{8.1.6}$$

The back-transform is

$$a^{-1}(p) = y = p + L \tag{8.1.7}$$

Example 8.1.5: Excess Wage

Hourly workers make at least $7.25 per hour.

To model the excess hourly wage, we would subtract $L = 7.25$ from each hourly wage. This new variable is bounded below by 0, so we can apply the log transformation to it.

Bounded Above by U

It may be that your dependent variable is theoretically bounded *above* by U. As there is only *one* bound, we will perform an algebraic transformation so that it is bounded **below by 0** and then apply the log transform as above, remembering to back-transform with the additional transformation. The algebraic transformation is

$$a(y) = p = U - y \tag{8.1.8}$$

The back-transform is

$$a^{-1}(p) = y = U - p \tag{8.1.9}$$

Example 8.1.6: Ocean Depths

In the ocean, different species live at different depths. In fact, we can predict the depth based solely on the species observed. Ocean depth is bounded above by 0 and has no *theoretic* lower bound (although it certainly has a genuine lower bound at the Challenger Deep in the Mariana Trench, which has a depth of -35,994 ft). To transform the depths into a variable upon which we can perform a log transform, we subtract each value from $U = 0$. After we predict, we will have to back-transform by again subtracting each prediction from $U = 0$.

Of course, the transformation in this last example is equivalent to measuring depth in terms of "distance below the surface," which is a positive number requiring no additional transformation.

This page titled 8.1: The Issue of Boundedness is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

8.2: The South Sudanese Referendum

This page presents a case study in electoral forensics, applying linear regression to a pressing political question: Was the 2011 South Sudanese independence referendum conducted fairly?

We begin with a fundamental statistical test for electoral fairness: the independence between a voter's intent and its probability of being invalidated. You will transform vote-count data, perform a linear regression on logit-transformed proportions, and interpret the results as evidence for or against electoral fairness. Crucially, you will also learn how to visualize this analysis effectively by plotting data points, a prediction curve, and confidence bands in the original proportion units — revealing the story behind the numbers with compelling clarity. The workflow demonstrates how statistical modeling, when paired with thoughtful visualization, can provide powerful insights into complex real-world events.

Learning Objectives

By the end of this section, you will be able to:

1. Apply electoral forensics principles to test for fairness in an election by examining the statistical relationship between the proportion of votes for a particular outcome (e.g., independence) and the proportion of invalidated votes across electoral divisions.
2. Perform a complete regression analysis on logit-transformed proportions to model this relationship, correctly handling data points with zero counts that fall outside the domain of the logit function.
3. Interpret the regression results, specifically a statistically significant negative relationship, as evidence against the assumption that a voter's probability of having their vote counted is independent of their voting choice.
4. Create a compelling visualization by plotting the original data points and overlaying the back-transformed prediction curve with Working-Hotelling confidence bands, effectively communicating the statistical findings in the original units of measurement.

✦•·····• ✦ •·····•✦

Electoral Forensics Background



Republic of South Sudan
(Source: Wikipedia)

Free and fair elections are one of the requirements for a legitimate democratic system; furthermore, being a legitimate democratic State is necessary for some forms of external assistance. As such, many not-so-democratic States wish to appear democratic. They hold elections, but the elections are either fraudulent or the electoral system (rules governing the elections) is unfair.

There are many definitions for fairness in an election, but they all contain the same requirement that a person's vote has the same probability of being counted as anyone else's. In other words, the probability of a vote being invalidated is **independent** of the characteristics of the person casting the vote — including who the vote was for. *This aspect of fairness* can actually be tested in elections where the number of invalidated votes is counted: If the proportion of the vote for a specific candidate or position is not independent of the proportion of the vote invalidated in the electoral division, then there is evidence against this assumption of fairness.

The Research Question

And so, with this background in elections and democracy, ...

Research Question:

> Does the 2011 independence referendum in southern Sudan indicate an issue with fairness?

Narrative Solution

As one of the conditions of the 2005 Naivasha Agreement, which ended the civil war in Sudan, the South was allowed to vote on independence from the North. That referendum was held January 9–15, 2011. Official results stated that 98.83% of the South Sudanese voted against unity and in favor of independence.

The `xsd2011referendum` data set contains the number of votes in favor of independence (`Secession`), the number of votes declared invalid (`Invalid`), and the total number of votes cast (`Votes`). Load it and save it into the `xsd` variable *without attaching the data*.

```
### Preamble

library(KnoxStats)
library(car)
library(lmtest)

xsd = read.csv("https://rur.kvasaheim.com/data/xsd2011referendum.csv")
```

Because we need to determine if there is a relationship between the proportion of the vote for a specific side and the proportion of the vote invalidated in the electoral division, and because we just have vote counts, we need to *create those proportions*. The proportion of the vote for the candidate is the number of votes for the candidate divided by the number of valid votes. The invalidation rate is the number of invalid ballots divided by the number of cast ballots.

```
xsd$Valid = xsd$Votes - xsd$Invalid
xsd$pSec = xsd$Secession/xsd$Valid
xsd$pInv = xsd$Invalid/xsd$Votes
```

Once that is done, we need to transform these proportions using the logit transformation (why?), perform linear regression, and check for a (linear) relationship. If a relationship exists in the transformed variables, then a relationship exists in the untransformed variables. **First**, however, it is **always a good idea** to plot the variables to see if there is an obvious answer to the question. Figure **8.2.1** gives a plot of the proportion of the vote invalidated against the proportion of the vote in favor of independence.

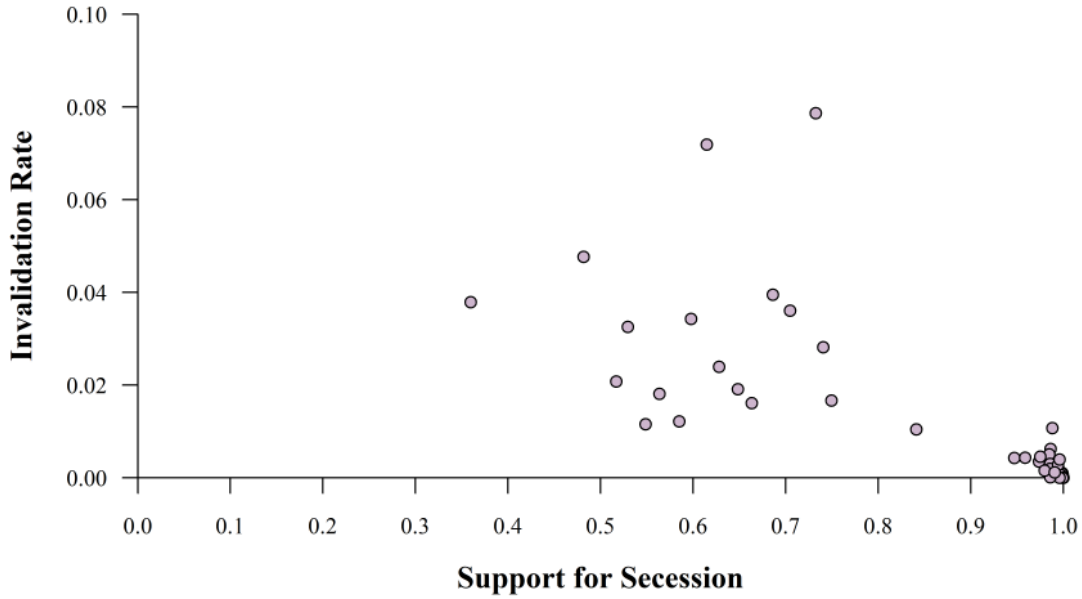


Figure 8.2.1: A scatterplot of the results of the 2011 referendum on independence for South Sudan. Note the apparent presence of a relationship between these two variables. As such, there appears to be evidence that the election was not fair for those voting against independence.

Suggested by the plot, there appears to be a strong relationship between the two variables, evidence of an election that is not fair. Because of the direction of the slope, it appears as though those areas voting most strongly in favor of independence had a *much* lower probability of having their votes rejected.

> As we are using the logit transform, we must drop any electoral division (here, county) that has zero invalid votes or zero votes in favor of secession. We need to do this because the domain of the logit function is $p \in (0,1)$.

Question:

> How will removing these counties affect the conclusions drawn?

To easily do this in R, we can use the `which` function, which returns which entries have the provided condition. Thus,

```
dr = which(xsd$Invalid==0)
```

returns a vector of values $\{15, 19, 23, 24, 28, 46, 47, 49, 50, 57, 72, 73\}$.

These numbers correspond to the counties that had zero invalid votes cast. Storing this vector in the variable `dr` (for "drop") allows us to remove those counties from any subsequent calculations. As such, our proportion calculations are:

```
p.ind = xsd$Secession[-dr]/xsd$Votes[-dr]
p.inv = xsd$Invalid[-dr]/xsd$Votes[-dr]
```

The negative signs tells R to return values in the vector **other than** these entries (to drop these entries).

And so, the two lines to transform the dependent variable and fit the OLS model are

```
l.inv = logit(p.inv)
model.xsd = lm(l.inv ~ p.ind)
```

The results of the linear regression on the transformed dependent variable are given in the table below. There is a very strong relationship between the proportion of the vote invalidated in the county and the proportion of the vote in favor of secession: Those counties with a greater proportion of people voting for independence also had a lower proportion of the vote invalidated. That there is a strong relationship between these two variables is troubling.

	Estimate	StdErr	t-value	p-value
Constant term	1.8978	0.7690	2.468	0.0155
Proportion of Vote for Independence	-9.3991	0.8287	-11.342	<< 0.0001

Table: Results table for the South Sudan referendum. The results are in logit units. Note the high level of statistical significance in the effect of the proportion of the vote in favor of independence. This very strongly suggests a lack of fairness in the election.

To make this relationship more obvious, and to make our point stronger, we can plot the data, the prediction curve, and the 95% Working-Hotelling confidence bands on the same plot.

Recall that what confidence intervals are to univariate data, confidence bands are to bivariate data.

The Graphing Philosophy of R

In base R, the philosophy behind graphing using base graphics is to start with a fresh plot and paint successive layers on top of it. This allows us to create graphs that tell the story and to do so easily. To make the graph described above, we need to

1. Plot the points (displayed in proportion units),
2. Plot the prediction curve (displayed in proportion units, but calculated in logit units),
3. Plot the 95% confidence bands (displayed in proportion units, but calculated in logit units).

The **first step** has been done already (Figure **8.2.1**).

The **second step** requires the repeated use of the `predict` function. First, to make things easier, let us define `newX` as a series of "proportion of vote in favor of independence" values for which we want to make predictions:

```
newX = seq(0, 1, length=1e4)
```

This creates a vector containing 10,000 values equally spaced between 0 and 1.

With this, our predict statement will be

```
l.pred = predict(model.xsd, newdata=data.frame(p.ind=newX), se.fit=TRUE)
```

Note

The `se.fit=TRUE` parameter, which calculates the standard error of the fit at that x-value, will be important for calculating the confidence bands. This is just a courtesy from R, as we know how to calculate this value from earlier in this book.

Remember that these predictions are in logit space. To get them into level units, we just apply the logistic function to these point predictions:

```
p.pred = logistic(l.pred$fit)
```

The `$fit` selects only the fitted predictions from the `l.pred` variable. This is necessary as we are also using the `se.fit=TRUE` parameter.

Now that we have the predictions in the original units, we merely paint it on the current plot (from Step 1):

```
lines(newX, p.pred)
```

The **third step** requires us to calculate the 95% confidence bands and paint them on the plot as well. For want of better estimates, let us use the Working-Hotelling bands. The formula to calculate the 95% confidence bands is

```
prwh = predictWH(model.xsd)
```

Once again, we must back-transform these two variables using the logistic function. So, our final confidence bands are

```
ucb = logistic(prwh$ucb)
lcb = logistic(prwh$lcb)
```

Finally, we paint this on the current plot with

```
lines(newX, ucb, col="grey")
lines(newX, lcb, col="grey")
```

This gives the Figure 8.2.2, below.

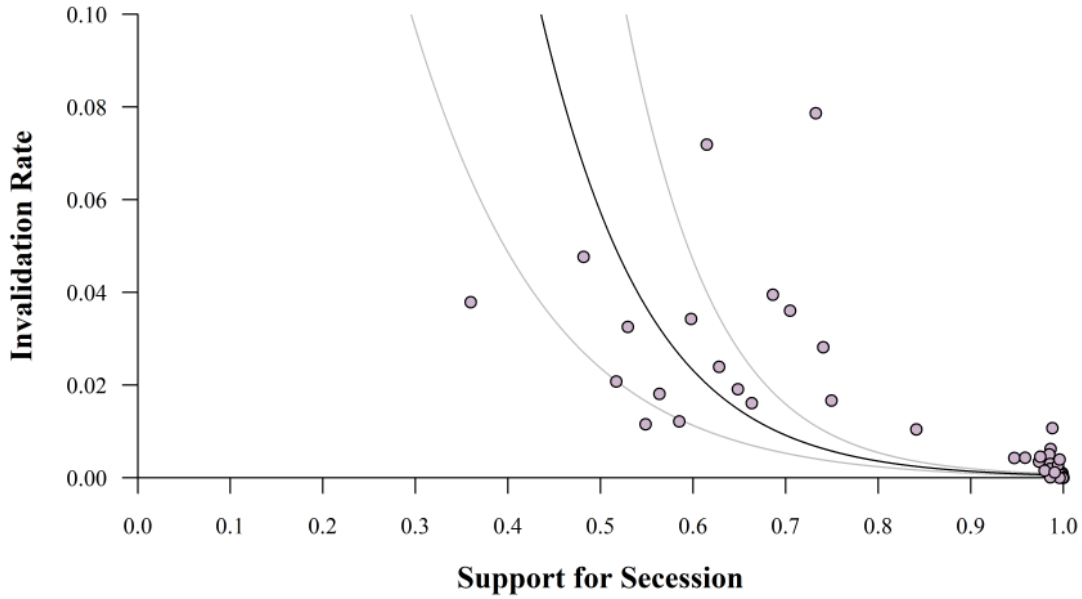


Figure **8.2.2**: A plot of the results of the South Sudan referendum. Included are the prediction line (in black) and the 95\% confidence bands (in grey). Note that a horizontal line cannot fit between the confidence bands. This indicates a statistically significant relationship between the proportion of the votes invalidated and the proportion of the votes in favor of independence. This, in turn, supports the conclusion of an unfair election.

Note that the predictions are curved in these units; they are straight in logit units. Also note the confidence bands are wider where the value of x is farther from $\bar{x}$. Lastly, note that no horizontal line can fit between the two confidence bands. This illustrates that there is a statistically significant relationship between the two variables at the $\alpha = 0.05$ level.

Question:

This illustrates that there is a statistically significant relationship between the two variables at the $\alpha = 0.05$ level. (Why?)

Note that the figure gives the same information as the regression table. The difference is that the graphic tells a clear story.

Caution

Graphs usually make the points more manifest.

The Graphic's Code

Here is the code I used to obtain Figure **8.2.2**. There are some interesting things in it. Please pore through the code to know what each line does.

```
par(mar=c(4,4,0,0)+0.5)
par(xaxs="i", yaxs="i")
par(family="serif", las=1)
par(font.lab=2, cex.lab=1.2)
par(xpd=NA)
```

```
plot.new()
plot.window(xlim=c(0,1), ylim=c(0,0.1))

axis(1, at=0:10/10); axis(2)
title(xlab="Support for Succession", line=2.5)
title(ylab="Invalidation Rate", line=2.75)

points(p.ind,p.inv, pch=21, bg="lavender")

lines(newX, p.pred)
lines(newX, ucb, col="grey")
lines(newX, lcb, col="grey")
```

This page titled 8.2: The South Sudanese Referendum is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

8.3: Adjustments for Heteroskedasticity

Linear regression assumes constant error variance, i.e., homoskedasticity. When this assumption is violated, as often happens with real-world data like electoral results, our standard errors become unreliable, leading to invalid hypothesis tests and confidence intervals. While transformations can sometimes remedy this, they may introduce other model problems.

This page re-introduces a powerful alternative: the Huber-White sandwich estimator. Instead of altering the model itself, this method directly adjusts the estimated standard errors to account for observed heteroskedasticity, providing "heteroskedasticity-consistent" inference. You will see the mathematical foundation of this adjustment and learn to implement it in R with a single command. Using the South Sudan referendum case, we'll compare original and adjusted results, highlighting how this correction can change — or affirm — your conclusions while leaving the original coefficient estimates unchanged. This technique is a crucial tool for robust statistical inference when model assumptions are violated.

Learning Objectives

By the end of this section, you will be able to:

1. Explain why heteroskedasticity violates the ordinary least squares assumption of constant error variance, leading to unbiased coefficient estimates but incorrect standard errors and invalid inference.
2. State the matrix form of the Huber-White heteroskedasticity-consistent estimator for the variance of $V[\mathbf{b}]$, and describe how it estimates $V[\mathbf{Y}|\mathbf{x}]$ from the observed residuals.
3. Apply the Huber-White correction in R using the `summaryHCE` function to obtain heteroskedasticity-consistent standard errors, t-statistics, and p-values for a fitted regression model.
4. Interpret the adjusted results, recognizing that the coefficient estimates remain unchanged while the standard errors are corrected to account for the observed heteroskedasticity.

✦•·····• ✦ •·····•✦

The previous transformations also work well on fixing problems with heteroskedasticity and non-Normality. Unfortunately, if you perform an appropriate transformation to fix the problem with model fit, further transformations to fix heteroskedasticity may end up creating a new problem with model fit. Thus, it may happen that you *cannot* find a way of fixing the heteroskedasticity without breaking something else. In such cases, we can adjust the standard errors using a technique introduced by White in 1980.

Recall from ordinary least squares estimation that our estimator for $\mathbf{B}$ is

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{8.3.1}$$

From this we showed that this estimator was unbiased; that is, $E[\mathbf{b}] = \mathbf{B}$. We also calculated the variance of the estimator as

$$V[\mathbf{b}] = \sigma^2 (\mathbf{X}'\mathbf{X})^{-1} \tag{8.3.2}$$

Homoskedasticity

That result, however, **required** that $V[\mathbf{Y}] = \sigma^2\mathbf{I}$. This is the assumption of homoskedasticity. Under heteroskedasticity, $V[\mathbf{Y}]$ cannot be reduced. This leaves the variance of our OLS estimators as

$$V[\mathbf{b}] = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' \;\; V[\mathbf{Y} \mid \mathbf{X}] \;\; \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1} \tag{8.3.3}$$

So, to better estimate $V[\mathbf{b}]$, we need to estimate $V[\mathbf{Y} \mid \mathbf{X}]$ from the data. (Everything else in Equation 8.3.3 is known.)

- How do we estimate $V[\mathbf{Y} \mid \mathbf{X}]$ from the data?

We recall that $V[\mathbf{Y} \mid \mathbf{X}] = V[\mathbf{E}]$. Thus, we can estimate $V[\mathbf{Y}]$ **from the residuals**, specifically from how large each residual is. If the ith residual is large, then the value of $V[Y_i]$ will be large; if e_i is small, then $V[Y_i]$ will be small.

Note

The Huber-White Sandwich Estimator makes *adjustments* for heteroskedasticity. These adjustments, however, are based on the data, which are random. Including this new source of randomness affects all confidence intervals and needs to be acknowledged.

It seems to me that it would be easier to create an entirely new estimation method than to just pile on adjustment after adjustment on OLS. This is, in fact, where we are headed in this book.

This was all covered in greater detail in Section 6.3: Homoskedasticity. I encourage you to review that section before continuing.

Having R Do This for Us

Instead of performing the above calculations by hand, we can have R do the adjustments for us. That helps with the accuracy and precision. The `summaryHCE` function in the `KnoxStats` package provides the adjustment and presents it in the form of our usual regression table.

Example 8.3.1: South Sudan Again

To illustrate the operation of the `summaryHCE` function, let us calculate the White-adjusted standard errors for the South Sudanese model of Section 8.2: The South Sudanese Referendum.

Solution.
We have already calculated the regression table for the logit model. From looking at the graphic, it seems as though there may be heteroskedasticity. There appears to be a lot more variation in the invalidation rate for smaller values of secession support than for larger values of secession support.

Running the following adjusts the standard errors to reflect the observed heteroskedasticity.

```
summaryHCE(model.xsd)
```

The heteroskedasticity-adjusted regression table is given below. Note that the estimates remain the same. That is because heteroskedasticity does not affect the estimates. The only changes are in the standard errors (and the test statistics and the p-values).

Original Regression Table

```
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   1.8978     0.7690   2.468   0.0155 *
p.ind        -9.3991     0.8287 -11.342   <2e-16 ***
```

Heteroskedasticity-Adjusted Regression Table

```
              Estimate HC Std. Error HC t value HC Pr(>|t|)
(Intercept)  1.897846     0.6704021   2.830907    0.004642
p.ind       -9.399121     0.7420048 -12.667196    0.000000
```

Notice that adjusting the standard errors is rather easy using R. It is just a single line. Also, notice that we did not model the heteroskedasticity, **we merely adjusted for it**.

Think About

At some level, it is unsettling to me to merely adjust for model weaknesses. It is a strong model that does not need fixes. Thus, if you can avoid using these Huber-White standard errors, I recommend it strongly. Heteroskedasticity is an important part of the data/model. It seems sinful to ignore it.

This page titled 8.3: Adjustments for Heteroskedasticity is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

8.4: Conclusion

In this chapter, we systematically addressed the challenge of applying ordinary least squares (OLS) regression to dependent variables with bounds. The core strategy centered on variable transformation, a process designed to remap bounded data into an unbounded space where the assumptions of OLS — particularly linearity and unbounded error — are more tenable. We began by classifying continuous variables into three fundamental types: unbounded, once-bounded (e.g., variables with a minimum of zero), and twice-bounded (e.g., proportions between 0 and 1). For unbounded variables, transformation is often unnecessary for compliance with OLS, though it may still be employed to improve residual normality and homoskedasticity. For once-bounded variables, the approach required a two-step procedure: first, an algebraic shift to ensure a lower bound of zero, followed by a log transformation to stretch the scale to negative infinity. For twice-bounded variables, the procedure similarly involved an algebraic transformation to standardize the bounds to 0 and 1, succeeded by a logit transformation, which maps the unit interval to the entire real line.

Crucially, the interpretative power of any model lies in its connection to the original, substantive scale of the data. Therefore, this chapter emphasized that the transformation process is not complete without the correct back-transformation. After fitting the model and generating predictions on the transformed scale, one must first apply the appropriate inverse function — the exponential for log-transformed variables or the logistic for logit-transformed variables. This step returns values to the intermediate, algebraically shifted scale. The final, essential step is to apply the inverse of the initial algebraic transformation to restore the predictions to the original units of the dependent variable. This sequence — inverse function first, then inverse algebraic shift — is paramount; reversing the order produces nonsensical results. Through this disciplined framework, we can harness the simplicity and interpretability of linear regression while responsibly modeling the complex, bounded realities present in much social, biological, and economic data.

This page titled 8.4: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

8.5: End-of-Chapter Materials

Here are the expected materials to supplement the chapter.

R Functions

In this chapter, we used several R functions that will be useful in the future. These are listed here.

Packages

- `KnoxStats`
  This package adds much general functionality to R.

Statistics

- `lm(formula)`
  This function performs linear modeling on the data, with the supplied formula. As there is much information contained in this function, you will want to save the results in a variable, to retrieve the information through the \function{summary} and \function{names} functions.

- `predict(model)`
  The `predict` function calculates the value of the dependent variable in the `model` given the independent variables used to create the model. If new predictions are required, the `newdata=` parameter must be used. This parameter takes a new set of data as its argument. Make sure that all independent variables used in the model are defined in the `newdata=` parameter. If not, an error message will result. Finally, specifying the `se.fit=TRUE` option calculates the standard error at each prediction point.

- `summaryHCE(model)`
  This function, a part of the `KnoxStats` package, allows us to easily calculate the heteroskedastic-consistent standard errors (White 1980).

Probability

- `pnorm(x, m, s)`
  This function is the cumulative distribution function (CDF) for the Normal distribution. It returns a **p**robability that a Normally-distributed variable will be less than or equal to `x` . This function has two additional parameters that remove the requirement that `x` has undergone the z-transformation, `m` and `s` .

- `rnorm(n, m, s)`
  This function returns `n` draws from a Normal distribution centered at `m` and with a standard deviation `s` . This function is the cornerstone of much Monte Carlo analysis.

Graphics

- `lines(x,y)`
  This is an extremely handy line-generating function, painting a line on the current plot (or returns an error if no plot exists). It first invisibly plots the pairs of points (x, y) then connects the points with drawn line segments. If the `col` parameter is not set, then the line will be black. Otherwise, the line will be the color specified. There are at least three ways of stating the color: using the Windows 1-16 values, using names, and using the rgb values. The following all refer to "red": `col=2` , `col="red"` , and `col="#ff0000"` .

- `plot()`
  This function produces a scatterplot of the two-dimensional data. The call can be either `plot(x,y)` or `plot(y~x)` ; both give identical results. This function can produce graphs that are very customized. The R help file for `par` is **invaluable**.

Some important parameters include `xlab` (label for the x-axis), `ylab` (label for the y-axis), `xlim` and `ylim` (axis limits, min and max, for the x- and y-axis), and `las=1` (makes axis values painted horizontal).

Mathematics

- `log(x, b)`
  This returns the logarithm of `x`, with a base of `b`. If you omit the `b`, this function returns the **natural logarithm** of `x`. To calculate the *common* logarithm, set `b=10`. The logarithm function is used to transform variables bounded on one side into variables bounded on neither side.

- `exp(x)`
  This function returns the exponential of the argument, `x`; that is, it returns e^x. The exponential function is the inverse of the logarithm function.

- `logit(x)`
  This function returns the logit of the provided number. This number must be between 0 and 1, not including either 0 or 1. The logit function is frequently used to transform proportions into unbounded data.

- `logistic(x)`
  This function returns the logistic of a given number. The range of the logistic function is 0 to 1, exclusive. it is the *inverse* of the logit function. As such, it is often used to transform predictions from logit units to proportion units.

- `cloglog(x)`
  The complementary log-log function is a second appropriate transformation for proportion data. It is, however, not a symmetric function.

- `cloglog.inv(x)`
  This function is the *inverse* of the complementary log-log function.

Programming

- `which(condition)`
  This function returns a vector of indices corresponding to the original vector's values meeting the criteria. Thus, `which(x==4)` returns the indices of all elements in vector `x` that equal 4. Note that equality is checked with a *double* equals, `==`. Other comparisons include: `>`, `<`, `>=`, `<=`, `!=`, `&`, `|`, and `!`. The last four are "not equal to," "and," "or," and "not."

Exercises

This section offers suggestions on things you can practice from this chapter.

1. Predict the Venkovský 1994 cow ballot measure vote using the transformed vote model. Is *this* prediction physically possible?
2. Determine a 95% confidence interval, with the *un*transformed cow vote model, for predicting Děčín's vote. Is the actual outcome within the 95% confidence interval?
3. Determine a 95% confidence interval, with the *transformed* cow vote model, for predicting Děčín's vote. Is the actual outcome within the 95% confidence interval?
4. Determine if the assumptions of OLS are violated in the transformed cow vote model.
5. The actual vote share for Děčín was 52.8%. Explain why both models failed in predicting the actual vote outcome. How bad was the error? What can be done to improve the predictions?

6. The logit transformation is not the only possible choice. There is also the asymmetric complementary log-log transformation (`cloglog`). Use this function as the transformation to predict Děčín's vote, its 95% confidence interval, and the probability of the cow ballot measure passing. The inverse of the complementary log-log transform has no name, but the R function is `cloglog.inv` .

7. Estimate the GDP per capita for Papua New Guinea using the untransformed model, as well as the 95% confidence interval. How close is this estimate to the real answer, and is the real answer within the predicted confidence interval?

Applied Readings

- James M. Avery (2009). "Political Mistrust among African Americans and Support for the Political System." *Political Research Quarterly*. 62(**1**): 132–45.
- Mark Andreas Kayser (2009). "Partisan Waves: International Business Cycles and Electoral Choice." *American Journal of Political Science*. 53(**4**): 950–70.
- Pamela A. Morris (2008). "Welfare Program Implementation and Parents' Depression." *The Social Service Review*. 82(**4**): 579–614.
- Kar Tean Tan, Christopher C. White, and Donald L. Hunston (2011). "An adhesion test method for spray-applied fire-resistive materials." *Fire and Materials*. 35(**4**): 245–59.

Theory Readings

- George Casella and Roger L. Berger (2001). *Statistical Inference*. New York: Duxbury Press.
- Annette J. Dobson and Adrian Barnett (2008). *An Introduction to Generalized Linear Models*, Third Edition. New York: Chapman & Hall.
- Friedhelm Eicker (1967). "Limit Theorems for Regression with Unequal and Dependent Errors." *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*: 59–82.
- Julian J. Faraway (2004). *Linear Models with R*. New York: Chapman & Hall.
- Julian J. Faraway (2005). *Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models*. New York: Chapman & Hall.
- Peter J. Huber (1967). "The Behavior of Maximum Likelihood Estimates under Nonstandard Conditions." *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*: 221–233.
- John A. Nelder and Robert W. M. Wedderburn (1972). "Generalized Linear Models." *Journal of the Royal Statistical Society*. Series A (General) 135(**3**): 370–84.
- Henry Scheffé (1959). *The Analysis of Variance*. New York: Wiley.
- Shayle R. Searle (1997). *Linear Models*. New York: Wiley-Interscience.
- James H. Stapleton (2009). *Linear Statistical Models*. New York: John Wiley and Sons.
- Robert S. Stritchartz (2000). *The Way of Analysis*, Revised Edition. Boston: Jones and Bartlett Mathematics.
- Halbert White (1980). "A Heteroskedasticity-Consistent Covariance Matrix Estimator and a Direct Test for Heteroskedasticity." *Econometrica*. 48(**4**): 817–838.
- Simon N. Wood (2006). *Generalized Additive Models: An Introduction with R*. New York: Chapman & Hall.

- Holbrook Working and Harold Hotelling (1929). "Applications of the Theory of Error to the Interpretation of Trends." *Journal of the American Statistical Association* 24(**1**): 73–85.

This page titled 8.5: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

9: Beyond the Ordinary

The methods of ordinary least squares (OLS) provide a powerful and intuitive foundation for statistical modeling. However, its core assumptions — linearity, homoskedasticity, and normality of errors — are often not met by complicated, real-world data. The transformations covered in the previous section are one key strategy for adapting data to meet these requirements. But what do we do when transformations are insufficient, when the violation of assumptions is fundamental to the process we are studying, or when our research question demands a different lens?

This part of this book moves beyond the foundational framework of OLS to equip you with a more advanced and flexible statistical toolkit. We transition from adapting the data to fit the model, to *tailoring the model itself* to fit the data and answer more nuanced questions. We will explore a family of estimators designed for specific, commonly encountered challenges: Weighted Least Squares (WLS) and Generalized Least Squares (GLS) formally model heteroskedasticity and correlated errors, respectively, to regain efficiency and valid inference. Quantile Regression abandons the exclusive focus on the conditional mean to model the entire conditional distribution, offering robust insights into effects at the median, tails, or any other quantile of interest — crucial for understanding inequality or extreme outcomes. Finally, we will lay the groundwork for a unified theoretical approach through Maximum Likelihood Estimation (MLE), the engine behind many of the most powerful modern models.

Together, these methods represent a paradigm shift. They acknowledge that the "best" model is not always a straight line fitted by minimizing squared errors, but rather the estimator whose logic best aligns with the structure of your data and the specificity of your inquiry. This section will provide you with the conceptual understanding and practical skills to select, implement, and interpret these advanced techniques, dramatically expanding the range of phenomena you can analyze with rigor and confidence.

This page titled 9: Beyond the Ordinary is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

10: Other Least Squares



In Chapter 6: Dood! Check the Requirements, we examined the three assumptions and how to check that they are not violated by your model. In the previous chapter, we saw how to fix our model to handle some of those violations. We were left, however, with some violations we could not fix. In this chapter, we jettison *ordinary* least squares regression and examine other types of regressions that rely on minimizing the sum of squared residuals. Each of these techniques allows you to specify a different covariance matrix. The requirement is that you actually know its structure without having to estimate it.

In the past several chapters, we have examined the classical linear model (CLM) and how to estimate the parameters using ordinary least squares (OLS). That introduction came in these chapters:

- Chapter 3: Intro to Linear Regression
- Chapter 4: Matrices and Linear Regression
- Chapter 5: Improved! Now with Probabilities

In Chapter 6: Dood! Check the Requirements, we discovered how to check the requirements (assumptions) of the ordinary least squares method. Chapter 8: Fixing the Violations gave us some options for dealing with violations of the requirements.

However, it may be that those fixes do not fully succeed — or **cannot** fully succeed. This chapter provides two estimation methods that offer advantages over ordinary least squares, as long as you have sufficient knowledge (science) of the structure of the problem — also known as the data-generation process.

This chapter reintroduces ordinary least squares. It then focuses on the covariance matrix of the residuals. As we reduce requirements on that matrix, we move from ordinary least squares to weighted least squares to generalized least squares.

10.1: Ordinary Least Squares
10.2: Weighted Least Squares
10.3: Generalized Least Squares
10.4: May the (Strong) Force be with You
10.5: Elections in Ruritania
10.6: Conclusion
10.7: End-of-Chapter Materials

This page titled 10: Other Least Squares is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

10.1: Ordinary Least Squares

This page serves as a (very) concise review of the core assumptions underlying ordinary least squares (OLS) regression. It revisits the fundamental premise that the model's errors are independent and identically distributed (i.i.d.) normal random variables with a constant variance. The section succinctly restates this assumption in both symbolic forms and interprets its key implication: the covariance matrix of the errors is a scalar matrix, where the diagonal represents the constant variance and the off-diagonal zeros confirm the errors are uncorrelated. This recapitulation sets the stage for the chapter's exploration of alternative least squares methods that relax one or more of these classic OLS conditions.

Learning Objectives

By the end of this section, you will be able to:

1. State the key ordinary least squares (OLS) assumption that the errors are independent and identically distributed (i.i.d.) with a mean of zero and constant variance, often expressed as $\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \sigma^2\mathbf{I})$.
2. Interpret the covariance matrix of the errors, $\sigma^2\mathbf{I}$, understanding that the diagonal entries represent the constant error variance (σ^2) and the off-diagonal entries of zero represent the assumption of no correlation between errors.
3. Explain that this assumption is fundamental for the standard OLS results regarding the distribution of estimators and the validity of hypothesis tests and confidence intervals.

✦•·····• ✦ •·····•✦

First, let us review ordinary least squares (OLS). When formulating OLS estimation of the classical linear model (CLM), we made the assumption that the residuals are independent and identically distributed Normal with constant zero expected value and variance.

In symbols, this is written as either

$$\varepsilon_i \overset{\text{iid}}{\sim} N\left(0; \sigma^2\right) \tag{10.1.1}$$

or as

$$\mathbf{E} \sim N_n\left(\mathbf{0}; \sigma^2\mathbf{I}\right) \tag{10.1.2}$$

The two statements are different ways of saying the exact same thing.

Note that the covariance matrix of $\mathbf{E}$ is $\sigma^2\mathbf{I}$:

$$V[\mathbf{E}] = \sigma^2 \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} = \begin{bmatrix} \sigma^2 & 0 & 0 & \cdots & 0 \\ 0 & \sigma^2 & 0 & \cdots & 0 \\ 0 & 0 & \sigma^2 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \sigma^2 \end{bmatrix} \tag{10.1.3}$$

The values along the diagonal represent the variances of each residual *in the population*. That they are the same value, σ^2, indicates that the variance of the residuals is constant.

The values off the diagonal represent the covariance between the residuals. For instance, the value at position 1,2 is the covariance between ε_1 and ε_2, which we symbolized as $\sigma_{1,2}$ in The Appendix of Statistics. Since that value is $\mathbf{0}$, we are specifying that the two are linearly uncorrelated (a.k.a. independent). Thus, the covariance matrix above specifies that the variances of the residuals are

constant and that the residuals are independent of each other. If this requirement is met, then we should use ordinary least squares regression. However, not always is this requirement met.

This page titled 10.1: Ordinary Least Squares is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

10.2: Weighted Least Squares

This page extends the ordinary least squares (OLS) framework by introducing weighted least squares (WLS), a method designed to handle a key violation of the OLS assumptions: non-constant error variance, or heteroskedasticity. While OLS assumes the errors have a constant variance ($\sigma^2\mathbf{I}$), WLS relaxes this by allowing each error to have its own variance, structured by a known diagonal matrix $\mathbf{D}$. The core innovation is a mathematical transformation that pre-multiplies the model by a weighting matrix, converting the heteroskedastic problem back into a familiar OLS form. This approach preserves the desirable properties of unbiasedness and allows for valid inference, provided the structure of the variances is correctly specified, either through substantive theory or approximated from the data.

Learning Objectives

By the end of this section, you will be able to:

1. State the weighted least squares (WLS) assumption that the errors are independent but have non-constant variance, expressed as $\varepsilon \sim N((0, \sigma^2\mathbf{D})$, where $\mathbf{D}$ is a known diagonal matrix of weights.
2. Explain how WLS reduces to an ordinary least squares problem through a transformation using the weighting matrix $\mathbf{W} = \mathbf{D}^{1/2}$, which standardizes the errors to have constant variance.
3. Derive the WLS estimator $\mathbf{b}_{\mathbf{WLS}} = (\mathbf{X}'\mathbf{D}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{D}^{-1}\mathbf{Y}$ and state its properties, including that it is unbiased for $\mathbf{B}$ and has variance $\sigma^2(\mathbf{X}'\mathbf{D}^{-1}\mathbf{X})^{-1}$.
4. Identify two approaches for obtaining the weights: using substantive scientific theory (e.g., known variance structures like $\sigma_i^2 = \pi(1-\pi)/n_i$ for Binomial proportions) or approximating them from the data using the diagonal elements of the hat matrix.

✦•·····• ✦ •·····•✦

It may be that the residuals are independent, but that their variance is known to not be constant. That is, we may have a model that leads to this assumption:

$$\varepsilon_i \overset{\text{ind}}{\sim} N\left(0; \sigma_i^2\right) \tag{10.2.1}$$

or as

$$\mathbf{E} \sim N_n\left(\mathbf{0}; \sigma^2\mathbf{D}\right) \tag{10.2.2}$$

Here, $\mathbf{D}$ is a diagonal matrix. Again, the two statements are different ways of saying the same thing.

Note that the covariance matrix of *this* $\mathbf{E}$ is

$$V[\mathbf{E}] = \sigma^2 \begin{bmatrix} d_1 & 0 & 0 & \cdots & 0 \\ 0 & d_2 & 0 & \cdots & 0 \\ 0 & 0 & d_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & d_n \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & 0 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & 0 & \cdots & 0 \\ 0 & 0 & \sigma_3^2 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \sigma_n^2 \end{bmatrix} \tag{10.2.3}$$

The values along the diagonal represent the variances of each residual (in the population). That they are not necessarily the same value indicates that the variance of the residuals can differ from observation to observation. The values off the diagonal represent the covariance between the values of the residuals. So, the value at position 1,2 is the covariance between ε_1 and ε_2, which we symbolized as $\sigma_{1,2}$ in The Appendix of Statistics (note that these are population values). Since $\sigma_{1,2} = 0$, we are specifying that the two residuals are linearly uncorrelated in the population.

> Remember that Greek letters refer to the population, while Latin refer to the sample (usually).

Thus, the covariance matrix above specifies that the variances of the residuals are allowed to be different and that the residuals are independent of each other.

This assumption is **the only difference** between *weighted* least squares and *ordinary* least squares. But, it is a rather significant difference.

> Remember that the value of σ^2 can indicate the variance of the population residuals *or* our uncertainty in the value of that residual.

To use weighted least squares (WLS), we need to know the structure of the $\mathbf{D}$ matrix. We do not need to know the exact values, but we need to know them up to a constant multiplier. That is, we need to know the *structure* of that heteroskedasticity. This usually comes from understanding the data-generating process. Frequently, this is *not* known (thus, WLS should probably *not* be used). However, there are some cases when we would} know this structure.

For instance, if we are working with a response variable that is a proportion arising from a binomially-distributed variable, we know that the variance is

$$\sigma_i^2 = \frac{\pi(1-\pi)}{n_i} = \pi(1-\pi)\frac{1}{n_i} \tag{10.2.4}$$

Thus, the diagonal elements will be $d_i = 1/n_i$ and the multiplier (constant part) will be $\pi(1-\pi)$.

Fitting WLS: The Mathematics

Assuming we know the structure of the $\mathbf{D}$ matrix, we can determine all we need to about the WLS estimators and estimates. We just reduce this problem to a previous problem. To clarify the similarities and differences between ordinary and weighted least squares, here is the classical linear model (CLM) for ordinary least squares:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E} \tag{10.2.5}$$

and for weighted least squares:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E} \tag{10.2.6}$$

Those are the same, whether one does OLS or WLS, because both come from the fact you are using the classical linear model. The difference comes in the assumption. Here is the assumption for OLS:

$$\mathbf{E} \sim N_n\left(0; \sigma^2\mathbf{I}\right) \tag{10.2.7}$$

Here is the assumption for WLS:

$$\mathbf{E} \sim N_n\left(0; \sigma^2\mathbf{D}\right) \tag{10.2.8}$$

Remember that $\mathbf{D}$ is a diagonal matrix.

The All-Important Transformation

There is a joke about how a mathematician solved the problem of a hose connected to a fire hydrant:

> A mathematician and a physicist were asked the following question:
> Suppose you walked by a burning house and saw a hydrant and a hose not connected to the hydrant. What would you do?
> **P**: I would attach the hose to the hydrant, turn on the water, and put out the fire.
> **M**: I would attach the hose to the hydrant, turn on the water, and put out the fire.
>
> Then they were asked this question:
> Suppose you walked by a house and saw a hose connected to a hydrant. What would you do?
> **P**: I would keep walking, as there is no problem to solve.
> **M**: I would disconnect the hose from the hydrant and set the house on fire, reducing the problem to a previously solved form.

And so, in the spirit of mathematicians, let us reduce the weighted least squares problem to that of ordinary least squares. If we can do this via a bijective transformation, then we have our confidence intervals and test statistics.

If we define our weighting matrix $\mathbf{W} = \mathbf{D}^{-1/2}$, then our problem is solved, *sans* burning down the house.

Theorem 10.2.1: Distribution of Weighted Residuals

Let $\mathbf{E} \sim N_n\left(0,\, \sigma^2\mathbf{D}\right)$. If we define $\mathbf{W} = \mathbf{D}^{-1/2}$, then

$$\mathbf{WE} \sim N_n\left(0,\, \sigma^2\mathbf{I}\right) \tag{10.2.9}$$

Proof.

Since $\mathbf{W}$ is a diagonal matrix and $\mathbf{E}$ has a Normal distribution, $\mathbf{WE}$ will *also* follow a Normal distribution. Thus, we need to calculate $E[\mathbf{WE}]$ and $V[\mathbf{WE}]$. In doing this, note that $\mathbf{W}$ is not a random matrix; it is known.

The expected value of $\mathbf{WE}$ is

$$\begin{aligned} E[\mathbf{WE}] &= \mathbf{W}\, E[\mathbf{E}] && (10.2.10)\\ &= \mathbf{W}\, \mathbf{0} && (10.2.11)\\ &= \mathbf{0} && (10.2.12) \end{aligned}$$

For the variance we have

$$\begin{aligned} V[\mathbf{WE}] &= \mathbf{W}\, V[\mathbf{E}]\, \mathbf{W}' && (10.2.13)\\ &= \mathbf{W}\, \sigma^2\mathbf{D}\, \mathbf{W}' && (10.2.14)\\ &= \sigma^2\, \mathbf{D}\, \mathbf{W}\, \mathbf{W}' && (10.2.15)\\ &= \sigma^2\, \mathbf{D}\, \mathbf{D}^{-1/2}\, \left(\mathbf{D}^{-1/2}\right)' && (10.2.16)\\ &= \sigma^2\, \mathbf{D}\, \mathbf{D}^{-1/2}\, \mathbf{D}^{-1/2} && (10.2.17)\\ &= \sigma^2\, \mathbf{D}\, \mathbf{D}^{-1} && (10.2.18)\\ &= \sigma^2\, \mathbf{I} && (10.2.19) \end{aligned}$$

In these steps, remember that matrix multiplication is commutative *if* the matrices are diagonal.

Thus, putting these three parts together gives our conclusion.

■

How do we actually *use* this theorem? We pre-multiply the model equation by the matrix $\mathbf{W}$ to obtain the following:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E} \tag{10.2.20}$$

$$\mathbf{WY} = \mathbf{WXB} + \mathbf{WE} \tag{10.2.21}$$

$$\mathbf{Y}^* = \mathbf{X}^*\mathbf{B} + \mathbf{E}^* \tag{10.2.22}$$

with $\mathbf{E}^* \sim N_n(\mathbf{0}, \sigma^2\mathbf{I})$.

Thus, we can apply all of our OLS results to WLS, as long as we speak to the transformed response variable $\mathbf{WY}$ and the transformed independent variable(s) $\mathbf{WX}$.

This quickly leads to our weighted least squares estimators of $\mathbf{B}$.

To prove this, we could proceed as we did back in Section 4.1: Matrix Representation. Or, since we have reduced the WLS problem to an OLS problem, we can just write out the results and simplify:

$$\mathbf{b}_{WLS} = (\mathbf{X}^{*\prime}\mathbf{X}^*)^{-1}\mathbf{X}^{*\prime}\mathbf{Y}^* \tag{10.2.23}$$

$$= ((\mathbf{WX})'\mathbf{WX})^{-1}\mathbf{WX}'\mathbf{WY} \tag{10.2.24}$$

$$= (\mathbf{X}'\mathbf{W}'\mathbf{WX})^{-1}\mathbf{X}'\mathbf{W}'\mathbf{WY} \tag{10.2.25}$$

$$= (\mathbf{X}'\mathbf{D}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{D}^{-1}\mathbf{Y} \tag{10.2.26}$$

It also quickly leads to showing that the WLS estimator is unbiased for $\mathbf{B}$:

Theorem 10.2.2: b_{WLS} is Unbiased for B

Under the assumptions of weighted least squares, the WLS estimator for $\mathbf{B}$ is unbiased.

Proof.
I am tempted to give this to you as an exercise, but let's see how to prove it.

$$E[\mathbf{b}_{WLS}] = E\left[(\mathbf{X}'\mathbf{D}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{D}^{-1}\mathbf{Y}\right] \tag{10.2.27}$$

Remember that the $\mathbf{D}$ matrix is *known*, is **not** a random variable.

$$E[\mathbf{b}_{WLS}] = (\mathbf{X}'\mathbf{D}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{D}^{-1}E[\mathbf{Y}] \tag{10.2.28}$$

Since $\mathbf{WY} = \mathbf{WXB} + \mathbf{WE}$, and since $\mathbf{W}$ and $\mathbf{D}$ are invertible (why?), we have

$$E[\mathbf{b}_{WLS}] = (\mathbf{X}'\mathbf{D}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{D}^{-1}\mathbf{XB} \tag{10.2.29}$$

$$= \mathbf{B} \tag{10.2.30}$$

Thus, the WLS estimator is unbiased if $\mathbf{D}$ is invertible and if $\mathbf{D}$ is known (non-stochastic). I will leave it as an exercise for you to prove this theorem if $\mathbf{D}$ is a random variable independent of $\mathbf{X}$.

■

Theorem 10.2.3: Variance of b_{WLS}

Under the assumptions of weighted least squares, the *variance* of the WLS estimator for $\mathbf{B}$ is

$$V[\mathbf{b}_{WLS}] = \sigma^2(\mathbf{X}'\mathbf{D}^{-1}\mathbf{X})^{-1} \tag{10.2.31}$$

Proof.
There should be no surprises with this proof. All you have to do is figure out what is a random variable and what is not. As such, I leave it as an exercise for you.

So very generous of me. =)

■

Note that the WLS estimator of $\mathbf{B}$ is a linear combination of independent Normal random variables. With that final observation, we have the distribution of the WLS estimator of $\mathbf{B}$:

Theorem 10.2.4: Distribution of b_{WLS}

The distribution of the WLS estimator of $\mathbf{B}$ is

$$\mathbf{b}_{WLS} \sim N\left(\mathbf{B},\ \sigma^2\left(\mathbf{X}'\mathbf{D}^{-1}\mathbf{X}\right)^{-1}\right) \tag{10.2.32}$$

Note

We again note that the individual estimators are not independent of each other under typical circumstances. We also note that the confidence intervals for the estimators, estimates of $\boldsymbol{Y}$, etc. can easily be determined in the WLS realm. Nothing new is here, only the mathematics is a bit more involved.

By the way, the `lm` function in R actually does weighted least squares estimation. If you do *not* specify the weights, then it defaults to all being 1... which is just ordinary least squares. If you specify the weights, they need to be entered as a **vector of inverse variances**.

R Feature

If you specify the weights, they need to be entered as a **vector of inverse variances**.

The Real Question

Weighted least squares takes care of the problem of heteroskedasticity in our data without introducing any major change in our modeling process or understanding. It just requires that we determine $\mathbf{W}$ and transform our dependent and independent variables by pre-multiplying by that weighting matrix.

Question:

> How do we obtain that weighting matrix?

The best way of obtaining it is through your substantive, scientific **theory**. The second best way is to utilize the hat matrix, $\mathbf{H}$.

... from Theory

Sometimes, knowledge of the problem suggests the weighting matrix. Recall that the $\boldsymbol{V}[\mathbf{E}]$ covariance matrix measures our uncertainty in the residuals. If that uncertainty is known by the way the experiment is constructed, then $\mathbf{W}$ can be determined. For

instance, if the dependent variable is the result of a Binomial experiment, perhaps it is the number of successes out of a given number of trials (which may change), then the weighting matrix is just a diagonal matrix of the square root of trial sizes.

Why? Recall that the variance of a binomially-distributed random variable is $\sigma^2 = \frac{\pi(1-\pi)}{N}$. The π are the unknown (constant) population proportion. The N_i are the (known) size within group i. The population parameter is assumed constant. The sample size is measurable. This leads to the $\mathbf{D}$ matrix being of the form

$$\mathbf{D} = \begin{bmatrix} 1/N_1 & 0 & 0 & \cdots & 0 \\ 0 & 1/N_2 & 0 & \cdots & 0 \\ 0 & 0 & 1/N_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1/N_n \end{bmatrix} \tag{10.2.33}$$

...from the Hat Matrix

When we do not have the theory to know the structure of the $\mathbf{D}$ matrix, one may want to use the hat matrix to give us a hint about its structure.

Caution: This is an Approximation

Make no mistake. This process is not mathematically exact... but what statistics procedure is? Statistics stands astride the real and the ideal, trying to get as much information about the real while acknowledging its limitations.

Remembering Chapter 6: Dood! Check the Requirements, not all violations affect inferences the same. Perhaps a good thing for you to do is to use the processes of that chapter to see how much using the hat matrix in lieu of a theoretically-driven $\mathbf{D}$ matrix affects the estimates, confidence intervals, and p-values.

Let us use the symbol $\mathbf{e}$ to represent the *observed* residuals. Until this point, we have only been working with the *theoretical* residuals, $\mathbf{E}$. The conceptual difference between the two is really just the difference between the population (theoretical) and the sample (observations). In effect, the difference is in terms of the variances.

Question:

What is the variance of $\mathbf{e}$?

Theorem 10.2.5: The variance of e

The variance of $\mathbf{e}$ is $V[\mathbf{e}] = \sigma^2(\mathbf{I} - \mathbf{H})$.

Proof.

$$V[\mathbf{e}] = V[(\mathbf{I} - \mathbf{H})\mathbf{Y}] \tag{10.2.34}$$

$$= (\mathbf{I} - \mathbf{H})\ V[\mathbf{Y}]\ (\mathbf{I} - \mathbf{H})' \tag{10.2.35}$$

$$= (\mathbf{I} - \mathbf{H})\ \sigma^2\mathbf{I}\ (\mathbf{I} - \mathbf{H}) \tag{10.2.36}$$

$$= (\mathbf{I} - \mathbf{H})\ \sigma^2\ (\mathbf{I} - \mathbf{H}) \tag{10.2.37}$$

$$= \sigma^2(\mathbf{I} - \mathbf{H}) \tag{10.2.38}$$

■

😺 So, that was totes cool. What was its purpose? What does it mean?

Remember how we can interpret that variance. It is either the variance of a gazillion observed residuals, *or* we can see it as the uncertainty inherent in the measured residual. For example, the inherent uncertainty in the first residual can be estimated as

$$s_{1,1} = \text{MSE}\,(1 - h_{1,1}) \tag{10.2.39}$$

Here, $h_{1,1}$ is the first element of the diagonal of the hat matrix.

That means, those diagonal elements of $\mathbf{I} - \mathbf{H}$ indicate (or are estimates of) the precision of the y estimate for a given value of x. An estimate of the structure of the $\mathbf{D}$ matrix is just the diagonal of the $\mathbf{I} - \mathbf{H}$ matrix.

Note

The problem is that weighted least squares requires us to *know* the structure of the $\mathbf{D}$ matrix, not that we estimate it from the data. This explains why the hat matrix technique is used only until something better comes along.

It does work nicely, but we statisticians like to "see the math" — sometimes. Also, if we are trying to draw important conclusions, using approximate methods tends to undercut the conclusions for many, especially for those who do not really understand statistics.

Question:

What do the off-diagonal elements of $\mathbf{I} - \mathbf{H}$ estimate?

This page titled 10.2: Weighted Least Squares is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

10.3: Generalized Least Squares

Both ordinary least squares and weighted least squares requires the errors be independent. Reality does not always meet this requirement. If the dependent variable consists of repeated measures on one unit over time, such as in modeling stock prices, it is quite likely that the residuals will be correlated. Also, if the dependent variable is measured on geographic structure, such as states in a country or trees in a forest, it is also likely that errors of near units are correlated.

In such examples, the covariance matrix of $\mathbf{E}$ will *not* be diagonal. Thankfully, it is a covariance matrix, and therefore positive definite under the usual assumption of no multicolinearity (The Appendix of Matrices, Section 20.4: Other Terms and Operations). Since it is positive definite, it is invertible. Thus, we may be able to do a trick not unlike what we did for weighted least squares.

Learning Objectives

By the end of this section, you will be able to:

1. State the generalized least squares (GLS) assumption that the errors are not necessarily independent or homoskedastic, expressed as $\varepsilon \sim N(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$) is a known, symmetric, positive-definite covariance matrix.
2. Explain how GLS extends both ordinary and weighted least squares by allowing for a general covariance structure among the errors, such as autocorrelation in time series data or spatial correlation in geographic data.
3. Describe how to determine or specify the structure of the covariance matrix $\mathbf{\Sigma}$) based on the problem context, using examples like an autoregressive AR(1) process for temporal data or an adjacency matrix for spatial data.
4. Derive the GLS estimator $\mathbf{b}_{\text{GLS}} = (\mathbf{X}'\mathbf{\Sigma}^{-1/2}\mathbf{X})^{-1}\mathbf{X}'\mathbf{\Sigma}^{-1/2}\mathbf{Y}$ by applying a transformation that pre-multiplies the model by $\mathbf{\Sigma}^{-1/2}$), which decorrelates and standardizes the errors, reducing the problem to an ordinary least squares form.

✦•·····• ✦ •·····•✦

The Mathematics

For a reminder, here are the model equations for ordinary, weighted, and general least squares:

```
Ordinary Least Squares     Y = XB + E
Weighted Least Squares     Y = XB + E
General Least Squares      Y = XB + E
```

They sure do look mighty similar. That's because this is still the classical linear model (CLM). The requirements on the residuals differs, however:

```
Ordinary Least Squares     E ~ N(0; σ² I)
Weighted Least Squares     E ~ N(0; σ² D)
General Least Squares      E ~ N(0; Σ)
```

For ordinary least squares, the covariance matrix of the residuals is a constant multiple of the identity matrix, $\mathbf{I}$. This indicates the residuals are independent and have the same variance (uncertainty). For weighted least squares, the covariance matrix of the residuals is a constant multiple of a diagonal matrix, $\mathbf{D}$. This indicates the residuals are independent, but possibly with unequal variances.

For **generalized least squares** (GLS), the covariance matrix is (a constant multiple of) a symmetric, positive definite matrix, Σ. This indicates the residuals are possibly correlated and with possibly unequal variances.

As with weighted least squares, you *do* need to know the structure of the covariance matrix. This requirement is sometimes met by the structure of the problem. The following are two examples showing how one *can* determine the Σ matrix. An understanding of these examples is not needed. They are here *only to illustrate* that there are times Σ can be determined from the problem.

GLS for Time Series Issues

When data are collected on a single unit *over time*, the measurements will tend to be correlated... called "temporal correlation" (or "serial correlation" or "autocorrelation"). For instance, the unemployment rate in Ruritania over the past 20 years is 11.35, 11.41, 11.12, 11.08, 10.93, 10.86, 10.96, 11.05, 11.10, 10.87, 10.79, 10.76, 10.94, 10.94, 10.92, 11.01, 11.04, 11.16, 11.13, and 11.14.

Modeling.
Let us fit this using ordinary least squares regression, then examine the residuals for autocorrelation (correlation between subsequent values).

```
unemp = c(11.35, 11.41, 11.12, 11.08, 10.93, 10.86, 10.96,
    11.05, 11.10, 10.87, 10.79, 10.76, 10.94, 10.94, 10.92,
    11.01, 11.04, 11.16, 11.13, 11.14)
year = 1:20

mod = lm(unemp~year)
E = residuals(mod)

autocor.test(E)
```

Note the sample autocorrelation is 0.719 with a p-value of 0.0005 and a 95% confidence interval from 0.393 to 0.884. The p-value indicates the autocorrelation is not 0. The confidence interval indicates that the residuals are moderately-to-highly correlated. In other words, adjacent observations are not independent, as both ordinary and weighted least squares require. Really, this makes sense because next year's unemployment rate will be heavily influenced by this year's rate.

Example 10.3.1: Time Series Example

Given that the correlation between adjacent measurements is 0.500, determine the covariance matrix.

Solution.
There are many ways of modeling such a situation. One is called "Autoregressive-1" or AR(1) or ARIMA(1,0,0). This model assumes that the primary correlation is only *directly* between adjacent years. The covariance matrix, Σ, would have this structure if the correlation between those adjacent years is $\rho = 0.500$:

$$\Sigma = \sigma^2 \begin{bmatrix} 1 & 0.5 & 0.25 & 0.125 & \cdots \\ 0.5 & 1 & 0.5 & 0.25 & \cdots \\ 0.25 & 0.5 & 1 & 0.5 & \cdots \\ 0.125 & 0.25 & 0.5 & 1 & \\ \vdots & \vdots & \vdots & \vdots & \ddots \\ & & & & 1 \end{bmatrix} \tag{10.3.1}$$

You can get this particular matrix using this R code:

```
Sigma = diag(20)
Sigma = 0.5^abs(row(Sigma)-col(Sigma))
```

Note that the matrix has 1s along the diagonal and higher powers of 0.5 farther from the diagonal. The zeroes arise from the fact that the matrix is 20×20; that is, e.g., the entry in cell $(1, 20)$ is actually $0.5^{19} \approx 0$.

■

Note

Again, this was just an example to show that the structure of Σ can be determined from some problems. There are entire sub-disciplines of statistics that examine such serial correlation. This sub-discipline is called time series analysis.

GLS for Geographic Issues

When data are collected from geographical units, such as neighborhoods, counties, or states, the residuals may be spatially correlated. This is a violation of the independence assumption of ordinary least squares. How that geographic correlation is modeled is up to the expert (researcher). The subject of spatial modeling is extensive and quite interesting... and important. It can, with appropriate matrices, be extended to modeling three-dimensional spatial correlation over time. If you have the opportunity, I suggest studying this topic (Bivand, Pebesma, and Gómez-Rubio 2013, Blangiardo and Cameletti 2013, Sen 2016). If nothing else, it leads to fun maps!



Figure **10.3.1**: A map showing the administrative divisions (kraj) or the Kingdom of Ruritania. For this example, note that no kraj abuts all other kraj. **Yes, I know that this is a map of San Marino**. I am working on putting together a map of Ruritania.

Figure **10.3.1** is a map of Ruritania showing the nine kraj. Note that some kraj abut some kraj but not others. For instance, region CS does not touch region CC. If we are trying to model the spread of something (disease, unemployment, wealth), we *may* decide to take into consideration the fact that some units neighbor others. Thus, from the map above, we know there is a first-level transmission between CS and CD but not between CS and CF.

✔ Example **10.3.2**: Geographic Adjacency Example

Let us determine a matrix describing the adjacencies for the nine kraj.

Solution.

Check that the following is the adjacency matrix for Ruritania

$$\Sigma = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{10.3.2}$$

It is important to ensure the kraj order for the columns is the same as for the rows. The kraj ordering is: CA, CB, CC, CD, CF, CI, CM, CS, CSM.

■

Note that the adjacency matrix is symmetric. Why will this matrix always be symmetric, regardless of the map? Now, that we know Σ is symmetric, we can use the discussion following this Lemma to conclude that Σ is positive definite, which indicates it is allowable as a covariance matrix.

Note on Adjacency

One area of **geographical analysis** tries to decide what adjacency rules are appropriate for a given research question. This example uses a simple 0-1 scheme. Other schemes include distances (measured in some manner).

We do not know the constant multiplier, σ^2. No probs. We only need to know the structure of the covariance matrix. We use the data to estimate the constant multiplier σ^2.

Also, note that the analysis based on this covariance structure is only as good as our assumption that the contagion spreads through touch. If it spreads based on some sort of distance, then the Σ is not correct and we will need to create an appropriate covariance matrix *given our scientific understanding*... if such exists.

Finally, let me reiterate a point I made above. The purpose of this example is *only* to illustrate that these covariance structures *can* be determined from the problem without resorting to estimating them from the data.

Caution

However, there is a lesson for all of us here. If we do not know the correct structure of the correlation matrix, then we should use several and see how sensitive our estimates, confidence intervals, and p-values are to that matrix. The results may be very sensitive, which is not a good position to be in, especially if we do not know the right mode of transmission.

If the estimates, etc. are not sensitive to our choice of covariance matrix, then we need not be as concerned.

The rule is to explore all models that make sense and see how important our assumptions are to our results.

One More Note

Above, we have only focused on being able to determine the structure of the covariance matrix (at least to a scalar multiple). There is one more thing that we need to pay attention to:

What is the square root of $\boldsymbol{\Sigma}$?

When we introduced $\mathbf{D}^{1/2}$ in the previous section, we knew we could calculate it. After all, one square root for a diagonal matrix would be

$$\left[\mathbf{D}^{1/2}\right]_{i,j} = d_{i,j}^{1/2} \tag{10.3.3}$$

That is, the elements of the square root matrix are the square root of the entries of the matrix. This shortcut works because $\mathbf{D}$ is diagonal. Such is not the case in the GLS setting, however. So, how do we calculate the square root of a positive-definite matrix? We use a finding from linear algebra.

For a general positive definite matrix, the most common method is to use an eigen-decomposition (also called spectral decomposition). This involves decomposing the original matrix $\boldsymbol{\Sigma}$ into the product $\boldsymbol{\Sigma} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}'$, where $\mathbf{Q}$ is a matrix whose columns are the eigenvectors of $\boldsymbol{\Sigma}$ and $\boldsymbol{\Lambda}$ is a diagonal matrix containing the corresponding eigenvalues. Because $\boldsymbol{\Sigma}$ is positive definite, all of its eigenvalues (the diagonal entries of $\boldsymbol{\Lambda}$) are positive. The square root of the matrix is then computed as $\boldsymbol{\Sigma}^{-1/2} = \mathbf{Q}\boldsymbol{\Lambda}^{-1/2}\mathbf{Q}'$. This resulting matrix $\boldsymbol{\Sigma}^{-1/2}$ is itself symmetric and positive definite, and it has the key property that when multiplied by itself ($\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\Sigma}^{-1/2}$), it returns the original matrix $\boldsymbol{\Sigma}$.

The GLS Results

So, now that we know what we are doing, let's follow the math through to get the GLS estimator. Recall that the structure of the model remains the same:

$$\mathbf{Y} = \mathbf{X}\mathbf{B} + \mathbf{E} \tag{10.3.4}$$

However, in GLS, the residuals may be correlated and heteroskedastic. That is,

$$\mathbf{E} \sim N(\mathbf{0},\ \boldsymbol{\Sigma}) \tag{10.3.5}$$

So, as we did in the weighted least squares section, we "divide" by the square root of the covariance matrix. That leaves us with this distribution:

$$\boldsymbol{\Sigma}^{-1/2}\mathbf{E} \sim N\left(\mathbf{0},\ \sigma^2\mathbf{I}\right) \tag{10.3.6}$$

So, if we pre-multiply the structural model by $\boldsymbol{\Sigma}^{-1/2}$, we are back to OLS:

$$\begin{cases} \boldsymbol{\Sigma}^{-1/2}\mathbf{Y} = \boldsymbol{\Sigma}^{-1/2}\mathbf{X}\mathbf{B} + \boldsymbol{\Sigma}^{-1/2}\mathbf{E} \\ \boldsymbol{\Sigma}^{-1/2}\mathbf{E} \sim N\left(\mathbf{0},\ \sigma^2\mathbf{I}\right) \end{cases} \tag{10.3.7}$$

Thus, the GLS estimator for $\mathbf{B}$ is

$$\mathbf{b}_{\text{GLS}} = \left((\boldsymbol{\Sigma}^{-1/2}\mathbf{X})'\boldsymbol{\Sigma}^{-1/2}\mathbf{X}\right)^{-1}(\boldsymbol{\Sigma}^{-1/2}\mathbf{X})'\boldsymbol{\Sigma}^{-1/2}\mathbf{Y} \tag{10.3.8}$$

$$= \left(\mathbf{X}'\boldsymbol{\Sigma}^{-1}\mathbf{X}\right)^{-1}\mathbf{X}'\boldsymbol{\Sigma}^{-1}\mathbf{Y} \tag{10.3.9}$$

? Exercise 10.3.3

Find $E[\mathbf{b}_{\text{GLS}}]$, the GLS estimator of $\mathbf{B}$.

? Exercise 10.3.4

Find $V[\mathbf{b}_{GLS}]$, the GLS estimator of $\mathbf{B}$.

This page titled 10.3: Generalized Least Squares is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

10.4: May the (Strong) Force be with You

This page provides a practical, side-by-side comparison of ordinary least squares (OLS) and weighted least squares (WLS) regression using a dataset from high-energy physics. It introduces a scenario where the relationship between variables is theoretically linear, but the measurements of the response variable come with different, known levels of uncertainty (standard deviations). By first applying OLS, which ignores this heteroskedasticity, and then WLS, which incorporates the uncertainty through weights, the example demonstrates how accounting for varying precision can affect parameter estimates and their confidence intervals. The section also guides you through verifying R's implementation of WLS by replicating the matrix calculation manually, reinforcing the conceptual understanding of how the weighting matrix operates.

Learning Objectives

By the end of this section, you will be able to:

1. Fit an ordinary least squares regression model to data with a known linear relationship, interpreting the coefficient estimates, confidence intervals, and measures of fit.
2. Recognize a situation where weighted least squares is appropriate, specifically when the uncertainty (variance) of the observations is not constant and is known for each data point.
3. Implement weighted least squares regression in R by providing the correct weights (inverse variances) in the lm function, and compare the resulting coefficient estimates and confidence intervals to those from ordinary least squares.
4. Verify that the weights are being applied correctly by manually performing the matrix calculations for the weighted least squares estimator and comparing the results to the output from the `lm` function.

✦•·····• ✦ •·····•✦

Ruritania is a patron of high-energy physics — and of Star Wars enthusiasts. King Rudolph donated several million crowns to Switzerland to aid in researching the strong force. That money was used at CERN (the *Conseil Européen pour la Recherche Nucléaire*) for several experiments. Each experiment consisted of a beam of protons crashing into a target. That beam had a constant energy level. What changed were the target sizes and the energy level of the proton after the collision. Many experiments were run at each energy level, and the standard deviation of the energies was measured.

In a theory proposed by Ruritanian scientists that is not entirely clear to His Majesty (or to your author), there should be a linear relationship between the cross sectional area and the *inverse* of the energy. The data are given in the table below.

Table **10.4.1**: The data table for this example.

Cross Section [b]	Energy [MeV]	StDev [MeV]
1	848.9	7.8
2	476.9	9.2
3	350.9	9.4
4	289.2	10.2
5	251.7	7.4
6	225.8	9.3
7	209.79	7.2
8	193.9	5.3

Table 10.4.1: Data for the example regarding the strong nuclear force. Units are given in brackets.

The first column is the value of the independent variable. The second column is the mean of the energy level of the photon after the collision. The third column is the standard deviation in those energy levels. Note that the variability at each cross-section differs. This is based on both the number of experiments *and* the inherent variability at that area.

Ordinary Least Squares

Let us ignore the different uncertainties in each energy level (the standard deviations). That is, let us just fit this as an OLS model.

Here is the code:

```
barns  = c(1,2,3,4,5,6,7,8)
energy = c(848.9,476.9,350.9,289.2,251.7,225.8,209.7,193.9)
Ibarns = 1/barns

modOLS = lm(energy~Ibarns)

summary(modOLS)
confint(modOLS)
```

The output suggests that the relationship between the cross sectional area of the target and the inverse of the resulting energy of the photon is statistically significant.

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 101.9148     0.5464   186.5 1.60e-12 ***
Ibarns      747.5306     1.2505   597.8 1.48e-15 ***
---

Residual standard error: 0.9719 on 6 degrees of freedom
Multiple R-squared:      1,    Adjusted R-squared:      1
F-statistic: 3.574e+05 on 1 and 6 DF,  p-value: 1.479e-15
```

A 95% confidence interval for the relationship is from 744.5 to 750.6 (with units of MeV·barn).

Weighted Least Squares

Note, however, that the uncertainty in the measurements varies. We are more uncertain with some of our estimated energy than with others. If we do not take this uncertainty into consideration, we may be biasing our results. To include this information, we can use weighted least squares regression. The code to fit with weighted least regression is as follows:

```
barns  = c(1,2,3,4,5,6,7,8)
energy = c(848.9,476.9,350.9,289.2,251.7,225.8,209.7,193.9)
stdev  = c(7.8,9.2,9.4,10.2,7.4,9.3,7.2,5.3)
v      = stdev^2

Ibarns = 1/barns

modWLS = lm(energy~Ibarns, weight=1/v)
summary(modWLS)
confint(modWLS)
```

Note that we are 95% confident the effect of the target's cross section on the resulting energy is from 744.5 to 751.2 MeV·barn.

Important R Feature

In R, as in many statistical programs, the weights you provide in the function call are inversely proportional to the variances. This is why we used `weight=1/v` in the function call.

With that being said, **always check the documentation** to make doubly sure. Frequently, this information is difficult to find and an error will not be thrown to let you know.

How do we know this?

Or, an even better **question**:

> How do we check that the weights really are proportional to the inverse of the variance?

The short answer is to **check your work "by hand."** As we once did OLS by hand using the matrix functions in R, we can also do WLS by hand. As always, make sure you know what each line does and why they are put together as they are:

```
## The data
barns  = c(1,2,3,4,5,6,7,8)
energy = c(848.9,476.9,350.9,289.2,251.7,225.8,209.7,193.9)
stdev  = c(7.8,9.2,9.4,10.2,7.4,9.3,7.2,5.3)

## A few minor calculations
v = stdev^2
Ibarns = 1/barns
```

```
n = length(Ibarns)

## The needed matrices
Y = matrix(energy, ncol=1)
X = matrix( c(rep(1,n), Ibarns), ncol=2 )
colnames(X) = c("b0","b1")
D = diag(v)

## The estimate vector
solve(t(X) %*% solve(D) %*% X) %*% t(X) %*% solve(D) %*% Y
```

When I run this, I get the following output

```
       [,1]
b0 101.6074
b1 747.8364
```

These are the matrix calculations according to the WLS estimation formula. Thus, this is the *correct* answer. Double-check that this known correct answer matches the answer given in `modWLS`. If it does not, then you will need to change the expression in the `weights` part of the function call. In R, it will match. In other pieces of software, it may not.

Be aware!

Caution

The difference between the effects estimated from using ordinary least squares and using weighted least squares is rather minor in this example. It need not be, as the next example shows.

This page titled 10.4: May the (Strong) Force be with You is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

10.5: Elections in Ruritania

This lengthy example presents a case study in electoral forensics, applying both ordinary least squares (OLS) and weighted least squares (WLS) regression to investigate allegations of ballot box stuffing in a Ruritanian parliamentary election. The analysis tests the theoretical expectation that systematic fraud would create a detectable negative relationship between a precinct's invalidation rate and its level of support for the ruling party. By first using OLS and then demonstrating why WLS is the more appropriate method due to the inherent heteroskedasticity of proportion data, the section illustrates how the choice of statistical model can fundamentally alter a study's conclusions. It also emphasizes the careful interpretation of results, reminding readers that statistical evidence does not constitute proof, only evidence.

Learning Objectives

By the end of this section, you will be able to:

1. Apply the theory of electoral forensics to a real-world dataset, investigating whether a statistical relationship between a party's vote share and the ballot invalidation rate can provide evidence of potential ballot box stuffing.
2. Fit an ordinary least squares regression model to test for a relationship between the invalidation rate and support for the ruling party, correctly interpreting the results and emphasizing that a non-significant finding does not prove the absence of unfairness.
3. Recognize why ordinary least squares is inappropriate for modeling proportion data like invalidation rates, specifically because the variance is not constant and depends on the total number of votes cast in each division.
4. Implement a weighted least squares regression that correctly accounts for the known heteroskedasticity by using the total votes as weights, and interpret the resulting coefficient estimates and p-values as evidence in the context of the research question.

✦•·····• ✦ •·····•✦

Even though it is an absolute monarchy, national elections are held in Ruritania to elect members of the Ruritanian parliament, the Národní Shromáždění (National Assembly). There are many parties represented in the parliament, but the party that consistently receives a majority of the seats and votes is the monarchist Pohyb pro Ruritánii (PR; Movement for Ruritania). The **main opposition party** is the Demokratické Hnutí (DH; Democratic Movement) party, but votes are also usually received by the Socialistická (SP; Socialist Party), Křstanská Demokratická (KD, Christian Democratic), and Republikánská (RS, Republican) parties.

It is fortuitous that Ruritania does not use computerized ballots. They use ballot papers for the parliamentary election that consist of the party names, symbols, and abbreviations... and a box for the voter to place their inked fingerprint next to the party. After voting, the ballot is placed in a ballot box to await counting.

At the end of the evening, the ballot papers at each precinct are securely transported to the division headquarters, where they are counted by electoral officials. Each ballot is checked by that official to ensure that it was lawfully cast and that the "will of the voter" can be discerned.

When the division is finished counting the ballots, the totals are then telephoned to the Independent Electoral Commission (*Nezávislá Volební Komise*, NVK) in the capital. With much pomp and circumstance, and not a little fanfare, the division totals are added and reported to the people.

After the last election, the exiles in Denmark claimed that the ballot boxes were stuffed. That is, the ballot boxes had votes for the PR party in them before voting began. Because guarantees of the secret ballot are built into the Ruritanian Constitution, the ballot boxes are opaque. In other words, direct evidence of ballot box stuffing does not exist, only claims by those who live in exile in

another country (Denmark). However, if ballot box stuffing existed in this election to any great extent, it would leave evidence. Why/how?

Question:

> What do stuffing ballots have that naturally cast ballots do not?

Easy: The stuffing ballots are all for the ruling PR party *and* they are all completed (filled in) correctly. The naturally cast ballots will consist of votes for all parties and will include ballots not filled-in correctly.

And so, in the presence of systematic and significant ballot box stuffing, there will be a **relationship between the invalidation rate and the level of support for the ruling party**.

That is the theory. The exiles are paying for this analysis. We like the money, so we need to be confident — and clear — in our conclusions. The NVK is providing the official counts in the `rur2013parl` data file, so we need to ensure that the statistical analysis is clean. That is, it is up to us to do the analysis correctly, neither concluding too much nor too little.

And, as always, being clear in our reasoning.

Ordinary Least Squares

The first analysis we will do is ordinary least squares. The dependent variable is the invalidation rate; the independent variable is the support for the Movement for Ruritania (PR) party. Why them? Since they were in charge before the election, they were in position to stuff the ballot boxes.

Here is the code to load the data, create the variables, fit the model, and determine if a relationship between the invalidation rate and PR support rate can be detected.

```
votes = read.csv("http://rur.kvasaheim.com/data/rur2013parl.csv")
attach(votes)

Valid  = Total-Invalid
pPR    = PR/Valid
pInv   = Invalid/Total

modOLS = lm(pInv~pPR)

summary(modOLS)
```

These results indicate that we did not detect a relationship... at the $\alpha = 0.05$ level ($\texst{p-value} = 0.0668$). Thus, ordinary least squares **did not detect** unfairness in the vote.

> It is important to emphasize here that the correct terminology is that we *did not detect* unfairness. We cannot say there was no fairness. We can only say we didn't detect it.

Remember to check the assumptions. This point cannot be over-emphasized. If the assumptions are not met, then the model is not correct. Well, not *perfectly* correct. See Chapter 6: Dood! Check the Requirements for a discussion of this point.

Weighted Least Squares

Note that ordinary least squares is *not* be the best option here. The invalidation rate has greater inherent variability in smaller divisions than in larger. We know this because of the distribution of the invalidation rate. Invalidation counts follow something akin to a Binomial distribution. Its two parameters are sample size (number of votes cast) and success probability (invalidation rate). The variance of a Binomial random variable is $n\pi(1-\pi)$.

Dividing the invalidation count by the number of votes cast gives the invalidation rate. The distribution of the invalidation rate can be approximated with a Normal distribution (see the Central Limit Theorem). The expected value of the observed invalidation rate is π, the inherent invalidation rate. The variance is $\pi(1-\pi)/n \propto 1/n$. Because the data are heteroskedastic in nature, and because the structure of the heteroskedasticity is known, weighted least squares will be more appropriate here.

Here is the code. Compare it to the ordinary least squares code from above.

```
Valid = Total-Invalid
pPR   = PR/Valid
pInv  = Invalid/Total

modWLS = lm(pInv~pPR, weights=Total)

summary(modWLS)
```

This produces the following (abbreviated) output:

```
Coefficients:
            Estimate  Std. Error  t value  Pr(>|t|)
(Intercept)   2.0726      0.7109    2.915    0.0225 *
pPR          -2.8808      1.0124   -2.845    0.0249 *
```

Note that this method *did* detect a relationship between the invalidation rate and the PR party support rate ($\text{p-value} < 0.05 = \alpha$). Furthermore, that differential invalidation helped the ruling party. The negative coefficient indicates that those *kraj* with higher PR support also tended to count more of the votes (reject fewer). Thus, we can conclude that the data are consistent with the exile claim of ballot box stuffing. Figure 10.5.1 illustrates this.

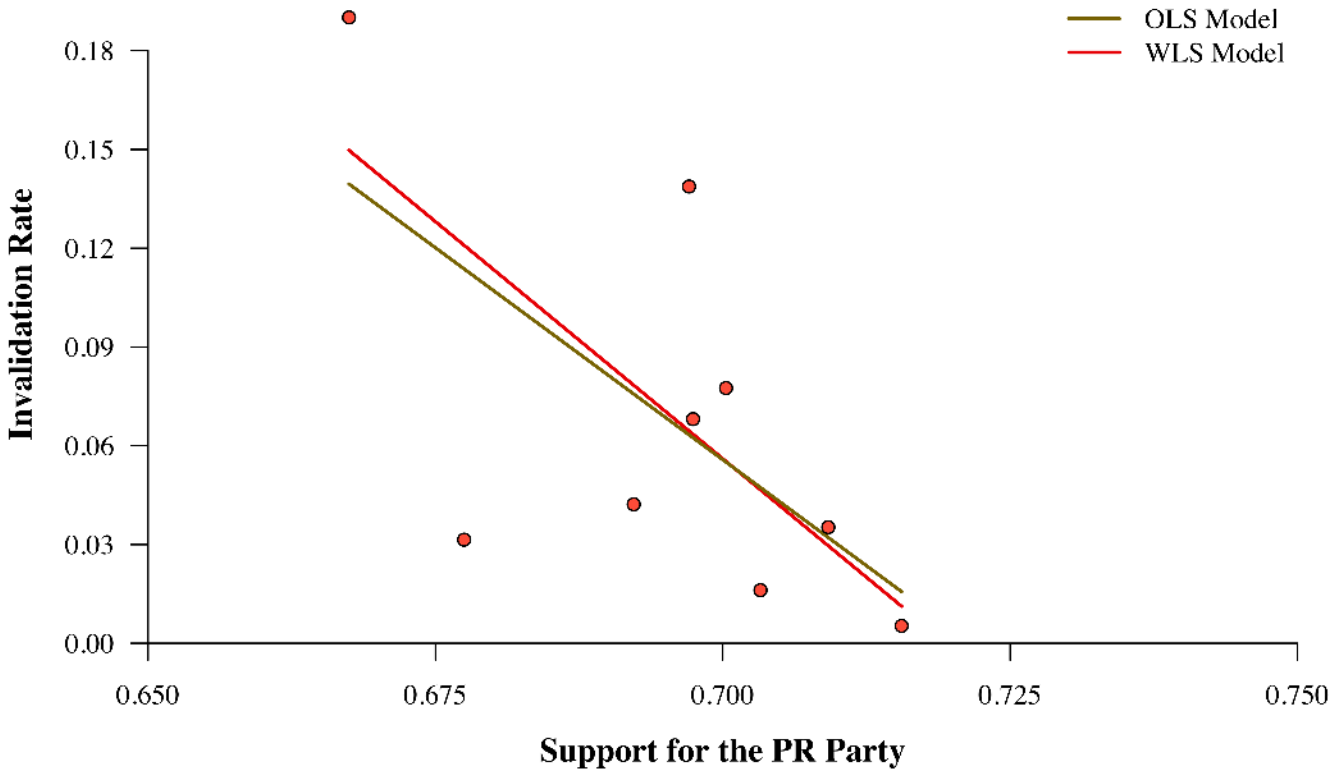


Figure 10.5.1: An invalidation plot for the 2013 Ruritanian parliamentary election. The lines of best fit are provided. The OLS fit is in brown and the WLS is in red.

Question:

How much is the differential invalidation?

Well, from the regression output, we know that when the support for PR increases by 10 percentage points (from 65 to 75%, for instance), the invalidation drops by an average of 27pp. That seems rather substantial to me.

Caution

Nothing in statistics ever constitutes **proof**. Nothing. Ever. Period. End of thought. *Žádné další!*

Statistics only provides evidence in favor of — or against — the null hypothesis. In this case, the p-value is 0.0249. If the null hypothesis is correct, then we would observed results this extreme or more so 2.49% of the time. This is not incredibly rare, especially when you realize you are claiming the government cheated. Cheating is a more serious claim than just that someone was mistaken.

It is always better to report the results, interpret the results, be explicit that there is no proof and that the null hypothesis has a non-zero probability of being reality.

Do not live your statistical life ruled by $\alpha = 0.05$. Realize — and accept — that the p-value is a measure of how well the data support the null hypothesis, the hypothesis of no relationship/difference/effect/evidence.

This page titled 10.5: Elections in Ruritania is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

10.6: Conclusion

In the previous chapters, we established the ordinary least squares (OLS) framework as a foundational tool for linear modeling. Its power derives from a core set of assumptions, most notably that the model residuals are independent and identically distributed (iid). When this condition holds, OLS yields not only unbiased estimates but also the Best Linear Unbiased Estimators (BLUE), allowing for straightforward hypothesis testing and the construction of reliable confidence intervals from a single, elegant formula.

However, the reality of empirical data often violates this tidy assumption. Residuals are frequently neither identically distributed (exhibiting heteroskedasticity) nor independent (exhibiting serial or spatial correlation). While we previously introduced post-hoc adjustments, like the Huber-White sandwich estimator, to "fix" the standard errors after an OLS fit, this chapter has argued for a more principled approach: directly modeling the violation to gain efficiency and insight. Weighted Least Squares (WLS) embodies this philosophy for heteroskedasticity. Rather than treating unequal variance as a nuisance to be corrected, WLS formally incorporates it into the estimation procedure. By weighting each observation inversely to its variance, WLS gives less influence to noisier data points, yielding more precise estimates — *provided* we have a theoretical or empirical basis for specifying those variances. Without such knowledge, we are indeed relegated to the corrective measures discussed earlier.

Extending this logic to the problem of dependence, Generalized Least Squares (GLS) offers a comprehensive framework. When observations are correlated — as in time series, spatial data, or clustered samples — GLS explicitly models the structure of that dependence through a known covariance matrix. By transforming the data using the inverse square root of this matrix, GLS effectively "whitens" the errors, restoring the iid condition and allowing us to apply the inferential machinery of OLS to the transformed model. The critical caveat, and a key limitation, is that this powerful technique rests on the feasibility of that transformation; the covariance matrix must be invertible and its structure must be correctly specified. When these conditions are met, GLS moves us beyond merely patching violations to fundamentally improving our model by embracing the true, complex structure of our data.

Building upon this progression from patching violations to explicitly modeling them, we now turn to a method that challenges a more fundamental OLS assumption: that our primary interest lies in the conditional mean. The techniques of WLS and GLS refine our estimate of the mean relationship by accounting for variance and dependence, but they remain anchored to that central tendency. In the next chapter, we will explore quantile regression, a paradigm shift that liberates us from this focus. Rather than modeling how predictors influence the average outcome, quantile regression allows us to examine how they shape the entire conditional distribution — be it the median, the 90th percentile, or someplace in the tails. This is particularly powerful when heteroskedasticity is not just a violation to be corrected, but a core feature of the phenomenon, indicating that a variable's effect differs across the distribution of the outcome. Thus, our journey from OLS to WLS and GLS, which taught us to model the structure of errors, now leads us to a method that models *the structure of the outcome itself*, offering a richer, more complete portrait of the relationships within our data.

This page titled 10.6: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

10.7: End-of-Chapter Materials

R Functions

In this chapter, we were introduced to a few R functions that will be useful in the future. These are listed here.

Packages

- `nlme`
  This package gives R the functionality to fit generalized least squares using the `gls` function. It actually has many other useful functions that allow us to fit non-linear models and random-effects models. Those are beyond the scope of this book, however.

- `KnoxStats`
  This package adds much general functionality to R.

Statistics

- `autocor.test(e)`
  This function calculates the auto-correlation, which is just the correlation between sequential values in the vector.

- `lm(formula)`
  This function performs linear regression on the data, with the supplied formula. If you specify the `weights`, then they are applied and you are fitting the model using weighted least squares. As there is much information contained in this function, you will want to save the results in a variable.

- `residuals(mod)`
  This calculates the simple residuals in a model, the observed values minus the predicted values.

- `gls(formula)`
  This function performs generalized least squares regression. It even allows you to specify the correlation structure via the `correlation` parameter.

Mathematics

This is just a reminder of some of the matrix functions available in R.

- `\%*\%`
  This multiplies two matrices in R. Thus, running the command `A\%*\%B` will return the matrix product $\mathbf{AB}$.

- `abs(x)`
  This returns the absolute value of the real number `x`, a.k.a. $|x|$.

- `column(A)`
  This returns the column number of the matrix $\mathbf{A}$.

- `diag(n)`
  If n is an integer, then this returns the $\mathbf{I}_n$ identity matrix.

- `diag(v)`
  This returns a diagonal matrix with the elements of the vector $\mathbf{v}$ along the diagonal.

- `diag(A)`
  This returns the diagonal entries of the matrix $\mathbf{A}$.

- `rep(n,x)`
  This returns a vector of the number \(x) repeated \(n) times.
- `row(A)`
  This returns the row number of the matrix $\mathbf{A}$.
- `solve(A)`
  This returns the inverse of the matrix $\mathbf{A}$.
- `t(A)`
  This returns the transpose of the matrix $\mathbf{A}$.

Exercises

1. Let $\mathbf{E} \sim N\left(\mathbf{0};\ \sigma^2\mathbf{D}\right)$ be the residuals. Prove that if $\mathbf{D}$ is a diagonal covariance matrix, then it is invertible.
2. Let $\mathbf{E} \sim N\left(\mathbf{0};\ \sigma^2\mathbf{D}\right)$ be the residuals. Here, $\mathbf{D}$ is a diagonal covariance matrix. Determine a matrix $\mathbf{W}$ such that $\mathbf{WW} = \mathbf{D}$.
3. Prove Theorem 10.2.2.
4. Under the assumptions of weighted least squares, determine the formula for a confidence interval for β_1.
5. What is the difference between $\mathbf{e}$ and $\mathbf{E}$?
6. Under the assumptions of generalized least squares, determine the formula for the estimator of $\mathbf{B}$.
7. Under the assumptions of generalized least squares, determine the formula for a confidence interval for $\mathbf{b}$.
8. Determine if Theorem 10.2.2 holds if the weights matrix $\mathbf{D}$ is a random matrix independent of $\mathbf{X}$. If it does not, what is the distribution of $\mathbf{WE}$?
9. Prove that the WLS estimator is unbiased for B if $\mathbf{D}$ is independent of $\mathbf{X}$.
10. The theorem concerning the variance of the WLS estimator requires $\mathbf{D}$ is non-random. Determine the variance of $\mathbf{b}_{\mathbf{wls}}$ if $\mathbf{D}$ is random, but independent of $\mathbf{X}$.
11. In the second example in Section 10.3, I state that the adjacency matrix is symmetric. Explain why this is so.

Theory Readings

- Adrian Baddeley, Ege Rubak, and Rolf Turner (2015). *Spatial Point Patterns: Methodology and Applications with R*. Boca Raton, FL: Chapman & Hall/CRC.
- Roger S. Bivand, Edzer Pebesma, and Virgilio Gómez-Rubio (2013). *Applied Spatial Data Analysis with R*. New York: Springer-Verlag.
- Marta Blangiardo and Michela Cameletti (2015). *Spatial and Spatio-temporal Bayesian Models with R*. Hoboken, NJ: John Wiley & Sons.
- Chris Brunsdon and Lex Comber (2015). *An Introduction to R for Spatial Analysis and Mapping*. Thousand Oaks, CA: SAGE Publications.
- Robert Haining (2003). *Spatial Data Analysis: Theory and Practice*. Cambridge, UK: Cambridge University Press.
- Tonny J. Oyana and Florence Margai (2013). *Spatial Analysis: Statistics, Visualization, and Computational Methods*. Boca Raton, FL: Chapman & Hall/CRC.

- Zekai Sen (2016). *Spatial Modeling Principles in Earth Sciences*. New York: Springer-Verlag.
- Thorsten Wiegand and Kirk A. Moloney (2013). *Handbook of Spatial Point-Pattern Analysis in Ecology*. Boca Raton, FL: Chapman & Hall/CRC.

This page titled 10.7: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

11: Quantile Regression



In the previous chapters, we defined "best" by minimizing the sum of the squared residuals. An even function was required to ensure that the values in the target function were positive. In this chapter, we use the absolute value function.

While this may sound like a trivial move, it is not. Since the magnitude of the slope of the absolute value is constant, optimization is difficult. Furthermore, since it is undefined at its minimum, we can only use an iterative technique to estimate (approximate) our parameters of interest. However, in building this structure, we are able to go beyond just estimating the median, we can estimate any quantile. This allows us to model the tails of the data-generating process.

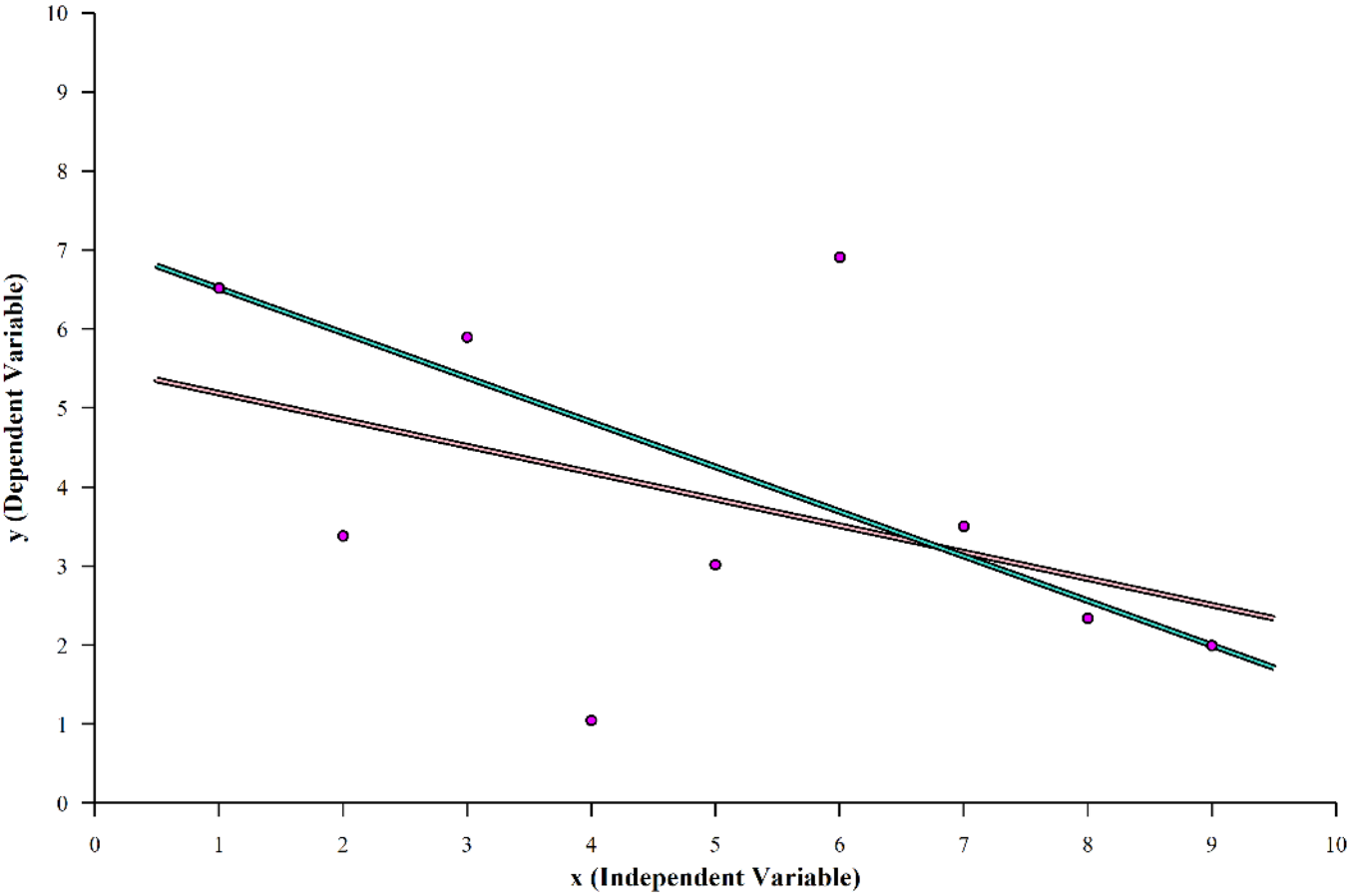


Figure **11.1**: A return to the line of best fit from Chapter 3: Intro to Linear Regression. The pink line above minimizes the sum of squared residuals. The turquoise line minimizes the sum of absolute residuals. Quite the difference!

In the previous sections we examined three types of least squares regressions — ordinary, weighted, and general. These three estimation methods have one thing in common: The estimates were obtained by minimizing the sum of squared residuals (properly weighted). We used the squaring function for two reasons. First, it is everywhere differentiable, especially at its minimum. Second, squaring the residuals ensures that you are adding non-negative values. All even functions attain the second goal. The class of functions that meet the first requirement is more restrictive.

The higher the even power, the more outliers affect the estimates; that is, the outliers will tend to have an increased effect on the estimator when the power is larger. One option to reduce the effect of these outliers is to use a different even function. The absolute value function has been used quite successfully in the past.

Unfortunately, the absolute value function is not everywhere differentiable. Even worse: it is not differentiable at its minimum — the point of interest. This means we cannot obtain a simple set of equations for our estimators. We can still, however, obtain estimators to an arbitrary degree of precision by using a set of equations that get us closer and closer to the true value of the estimate.

Chapter Sections

11.1: Parameter Estimation
11.2: Quantile Regression

11.3: Conclusion
11.4: End-of-Chapter Materials

This page titled 11: Quantile Regression is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

11.1: Parameter Estimation

The purpose of this section is to give some insight into the difficulties created by minimizing the sum of the absolute errors. Remember that the absolute value function has no derivative defined at its minimum and that the derivative has the same magnitude elsewhere. So, there is no way to create a formula for the minimum. So, an iterative procedure must be created to obtain the parameter estimates.

This section looks at a possible way to estimate those parameters using a rather rudimentary iterative procedure. It is not the actual method used; it just illustrates some of the inherent issues with an iterative technique... including the fact that there are several options to choose from, including one that you may created as your PhD dissertation.

Learning Objectives

By the end of this section, you will be able to:

1. Explain why estimating parameters for a model that minimizes the sum of absolute errors (L_1 regression) requires an iterative procedure, in contrast to the closed-form solution available for ordinary least squares.
2. Describe a simple iterative approach for finding L_1 regression estimates by proposing candidate lines, calculating the sum of absolute residuals, and adjusting parameters to reduce this value.
3. Identify that multiple optimization algorithms exist for quantile regression and that they can lead to slightly different solutions, particularly with integer data.
4. Fit a median regression model ($\tau = 0.500$) in R using the `rq` function from the `quantreg` package, and interpret the coefficient estimates and their confidence intervals, comparing them to OLS results.

✦•·····• ✦ •·····•✦

A Ham-Fisted Try Estimating the Parameters

In least squares, we used the L_2 norm to measure how close the model is to reality. We were able to use calculus to get closed-form equations for the estimators. Here, since such solutions do not exist and since we need to use an iterative technique, I think working through a toy example may help understanding. And so, let us start with the data in the left two columns of the table below.

Remember that we want to minimize the sum of the absolute values (the L_1 Norm) of the residuals. *In other words*, we want to minimize the L_1 distance between the n-dimensional data vector and the p-dimensional parameter space. Recall the figure in Section 4.3 where we illustrated this with least squares.

Thus, the **first step** is to obtain residuals. This means we need to somehow obtain our first estimated regression line. Any will work as a starting point. So, let's say our first estimate of the line-of-best-fit is $\ell_1 : y = 3$, which is just the horizontal line at the median.

The **next step** is to calculate the residuals. This is the e_1 column in the table below. The the target function is

$$Q_1 = \sum_{i=1}^{n} |e_i| \tag{11.1.1}$$

$$= \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{11.1.2}$$

$$= \sum_{i=1}^{n} |y_i - 3| \tag{11.1.3}$$

$$= |y_1 - 3| + |y_2 - 3| + |y_3 - 3| + |y_4 - 3| \tag{11.1.4}$$

$$= |1 - 3| + |3 - 3| + |3 - 3| + |9 - 3| \tag{11.1.5}$$

$$= 2 + 0 + 0 + 6 \tag{11.1.6}$$

Thus, for line ℓ_1, the value of the target function is 8.

The next step is to change the regression line. How? Well, that is the important question. Different methods may ultimately lead to slightly different answers. As this section only seeks to illustrate a method --- and not even a good method --- let's use logic to see what would be next. Note that the lower values have estimates that are too low, and the higher values have estimates that are too high. So, it makes sense to increase the slope. So, let us increase the slope to 1. *If* we force the line to pass through the dimension-wise median $(\tilde{x},\ \tilde{y}) = (1.5,\ 3.0)$, the linear equation will be $\ell_2 : y = 1.5 + 1x$. This produces the estimates and residuals in the next two columns of the table below.

x	y	y-hat_1	e_1	y-hat_2	e_2	y-hat_3	e_3
0	1	3	-2.0	1.5	-0.5	0.0	1.0
1	3	3	0.0	2.5	0.5	2.0	1.0
2	3	3	0.0	3.5	-0.5	4.0	-1.0
3	9	3	6.0	4.5	3.5	6.0	3.0

The value of the target function is

$$Q_2 = \sum_{i=1}^{n} |e_i| = \sum_{i=1}^{n} |y_i - \hat{y}_i| = \sum_{i=1}^{n} |y_i - (1.5 + x_i)| \tag{11.1.7}$$

Note that this value is 5. As this is lower than the previous value, we headed in the right direction; we are closer to the estimates because we have reduced the sum of the absolute errors.

We got **closer**. Note that the error for higher x-values is greater than for lower x-values. This suggests we should increase the slope yet again.

So, let us select our third line as $\ell_3 : y = 0 + 2x$. Again, we are forcing the line to pass through the dimension-wise median. Do we need to do this? No. There are algorithms that do not force this restriction. Again, the actual mathematics cannot reasonably be done by hand. I write this part so that you can get a feel for what the computer is doing. The last pair of columns in the table above provide the predictions and residuals for this third line.

The value of the target function for this third line is $Q_3 = 6$. This value is *not* lower than Q_2. Thus, this line is a worse fit than line ℓ_2. The next line, ℓ_4, needs to take this into consideration.

This process would continue until the change in target function values is "small enough." Usually, we define "small enough" as being less than some tolerance, like $\tau = 0.000\,001$.

The Big Question

The big question is how we get from one "line-of-best-fit" to the next, from ℓ_i to ℓ_{i+1}. Unfortunately, there is no "best method" to minimize the L_1 norm when there are more points than dimensions. It is even worse: We were able to find closed-form solutions to the unique estimators for the L_2 norm (squaring). That cannot be done when using the L_1 norm (absolute values). There are multiple appropriate algorithms. The estimators may not be unique. Those are just a *few* problems working with the L_1 norm.

For those interested, here are some methods:

- Barrodale and Roberts (1973),
- Koenker and Bassett (1978),
- Koenker and d'Orey (1987, 1994),
- Li and Arce (2004), and
- Shu-guang and Jian-wen (1992).

> These algorithms make use of different paradigms, different ways of seeing the problems. That is what makes studying statistics fun and interesting. Looking at a problem differently may be the key to its solution.

The R Function

In R, a function to perform median regression is `rq` in the package `quantreg`, which does not come with the default R installation; thus, you will need to install it from CRAN. Its use is very similar to what we are used to. While the `rq` function allows you to select different optimization methods, the default is the Barrodale and Roberts (1974) method. From my experience the optimization algorithm matters little for real data. If the data are all integers, there may be issues with non-unique solutions or non-convergent algorithms. The cause in these cases is the non-uniqueness of the median.

Example 11.1.1: Modeling Violent Crime Rate, I

Using median regression, what is the relationship between the violent crime rates in 2000 and 1990 in the `crime` data?

Solution.
The following code estimates the median regression line for the relationship between the violent crime rates in 2000 and 1990 in the \data{crime} data:

```
library(quantreg)

dt = read.csv("http://rfs.kvasaheim.com/data/crime.csv")
attach(dt)

mod1 = rq(vcrime00~vcrime90)

summary(mod1)
```

The following is the output:

```
Call: rq(formula = vcrime00 ~ vcrime90)
tau: [1] 0.50

Coefficients:
            coefficients lower bd  upper bd
(Intercept)  93.24525     72.83955 102.31731
vcrime90      0.57764      0.57518   0.62676
```

The output is the usual output. The value of `tau` is **0.500**, because we are examining the regression line for the median, the 50th percentile.

The coefficients are the estimates for the intercept and slope. The lower and upper bounds are the 95% confidence interval for those parameters. There are no p-values, because the distribution of the estimators does not follow a nice test distribution. However, because we have an approximate confidence interval, we have even more information than what a simple p-value would give. We are 95% confident that the relationship between the violent crime rate in 1990 and 2000 is between 0.575 and 0.627. Since this does not include the value 0, we can conclude that there is a significant relationship between the two variables.

■

I leave it as an exercise for you to see that the OLS estimator for that effect is 0.581, with a 95% confidence interval from 0.518 to 0.643.

Clearly, there is a difference between the two estimation methods. That difference is in how the method is affected by influential points like the District of Columbia. Median regression reduces the influence of DC, while ordinary least squares does not. The absolute value function increases linearly as the residual increases. The squaring function increases quadratically as the residual increases. Thus, ordinary least squares will work harder to avoid making the DC residual too big. Median regression will not weight it as heavily.

Also, ordinary least squares models the conditional mean (expected value of Y given the value of x). Median regression models the conditional median (expected median of Y given the value of x).

✔ Example 11.1.2: Modeling Property Crime Rate, I

Here is another example of using median regression. What is the relationship between the property crime rates in 1990 and 2000?

Solution.
The code is quite similar to that above:

```
mod3 = rq(pcrime00~pcrime90)
summary(mod3)
```

The following is the partial output:

```
Coefficients:
            coefficients lower bd   upper bd
(Intercept)  730.46936    349.56585 1093.31979
pcrime90       0.60584      0.50893    0.77457
```

Again, the relationship is positive. A point estimate for that relationship is $\tilde{\beta}_1 = 0.606$, with a 95% confidence interval from 0.509 to 0.775. I again leave it as an exercise for you to show that the OLS estimator is 0.582 with a 95% confidence interval from 0.458 to 0.707.

■

When the data are "well behaved" without influential points, there tends to be little difference in the estimators. Figure **11.1.1**, below, illustrates this.



Figure **11.1.1**: A graphic comparing the estimated lines from ordinary least squares (gold) and median (red) regression.

What does the above graphic tell us?

This page titled 11.1: Parameter Estimation is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

11.2: Quantile Regression

This section extends the concept of median regression introduced in the previous section to the broader framework of quantile regression. While ordinary least squares models the conditional mean of the response variable and median regression models the conditional median (the 50th percentile), quantile regression allows estimation of any desired percentile, such as the 10th or 90th. This flexibility is particularly valuable in fields like sociology or education, where researchers may be interested in the lower tail (e.g., poverty line incomes) or the upper tail (e.g., high academic achievers). Using the familiar crime dataset examples, the page demonstrates how to specify different quantiles in the `rq` function and interpret the results, showing how relationships between variables can differ across the distribution.

Learning Objectives

By the end of this section, you will be able to:

1. Distinguish between ordinary least squares, which models the conditional mean of the response variable, and quantile regression, which can model any conditional quantile (such as the median, 10th, or 90th percentile).
2. Fit quantile regression models for specified percentiles (e.g., tau = 0.10, 0.50, 0.90) in R using the `rq` function, and interpret the coefficient estimates and their confidence intervals for each quantile.
3. Compare and contrast the results from different quantiles and from OLS to gain a more complete understanding of the relationship between variables, particularly when the relationship may differ across the distribution of the response.

✦•·····• ✦ •·····•✦

The previous section covered median regression. There, we motivated the method by focusing on minimizing the sum of the absolute value of the residuals. It turns out that this is equivalent to estimating the **conditional median** of the dependent variable (hence its name). In other words, the line of best fit is the line that best goes through the medians at each x-value.

Compare this to how we motivated ordinary least squares back in Chapter 3: by minimizing the sum of the squared errors. This is equivalent to estimating the **conditional mean** of the dependent variable.

In other words, OLS estimates $E[Y \mid x]$ while median regression estimates $\mathbf{Med}[Y \mid x]$, for want of better notation. (Perhaps $Q_2[Y \mid x]$ or $P_{50}[Y \mid x]$ would be better notation.)

There is absolutely no reason we need to focus *only* on the conditional median of the dependent variable (conditional on the independent variable). We may want to focus on other percentiles, like the 10th percentile or 90th. This happens a lot in sociology when studying poverty (10th percentile of income) or education (90th percentile of academic achievement). The idea behind the fitting is the same (Koenker and Hallock, 2001). The R function is also the same. The only difference is that you need to specify the quantile. To see this, let us see a couple familiar examples.

Modeling the Violent Crime Rate

Example 11.2.1: Violent Crime Rate, II

What is the relationship between the violent crime rates in 2000 and 1990 in the `crime` data at the 10th percentile?

Solution:
Here is the code the perform this estimation:

```
mod5 = rq(vcrime00~vcrime90, tau=0.10)
summary(mod5)
```

The following is the output:

```
Call: rq(formula = vcrime00 ~ vcrime90, tau = 0.1)
tau: [1] 0.1

Coefficients:
            coefficients lower bd  upper bd
(Intercept)  40.29964    -14.80397 100.38757
vcrime90      0.55616      0.38948   0.60422
```

Thus, for those states near the 10th percentile ($\tau = 0.10$), the effect of the 1990 violent crime rate on the 2000 is between 0.389 and 0.604, with a point estimate of 0.556. This is only a *little* different from the median results, which suggests those states that are less crime-ridden (at the 10th percentile) still followed the same "rule" with respect to violent crime rate changes between 1990 and 2000.

■

Modeling the Property Crime Rate

Example 11.2.2: Property Crime Rate, II

What is the relationship between the property crime rates in 2000 and 1990 in the `crime` data at the 90th percentile?

Solution:
Here is the code to perform this estimation:

```
mod6 = rq(pcrime00~pcrime90, tau=0.90)
summary(mod6)
```

The following is the output:

```
Call: rq(formula = pcrime00 ~ pcrime90, tau = 0.9)
tau: [1] 0.9

Coefficients:
            coefficients lower bd   upper bd
(Intercept) 1761.72465    327.54503 2436.15997
pcrime90       0.53326      0.40262    0.84489
```

Thus, for those near the 90th percentile ($\tau = 0.90$), the effect of the 1990 violent crime rate on the 2000 is between 0.403 and 0.845, with a point estimate of 0.533. This differs a little from the median results, which suggests those states that are more (property) crime-ridden (at the 90th percentile) followed a similar "rule" with respect to violent crime rate changes between 1990 and 2000. Their rates dropped slightly more than did the typical (median) state.

■

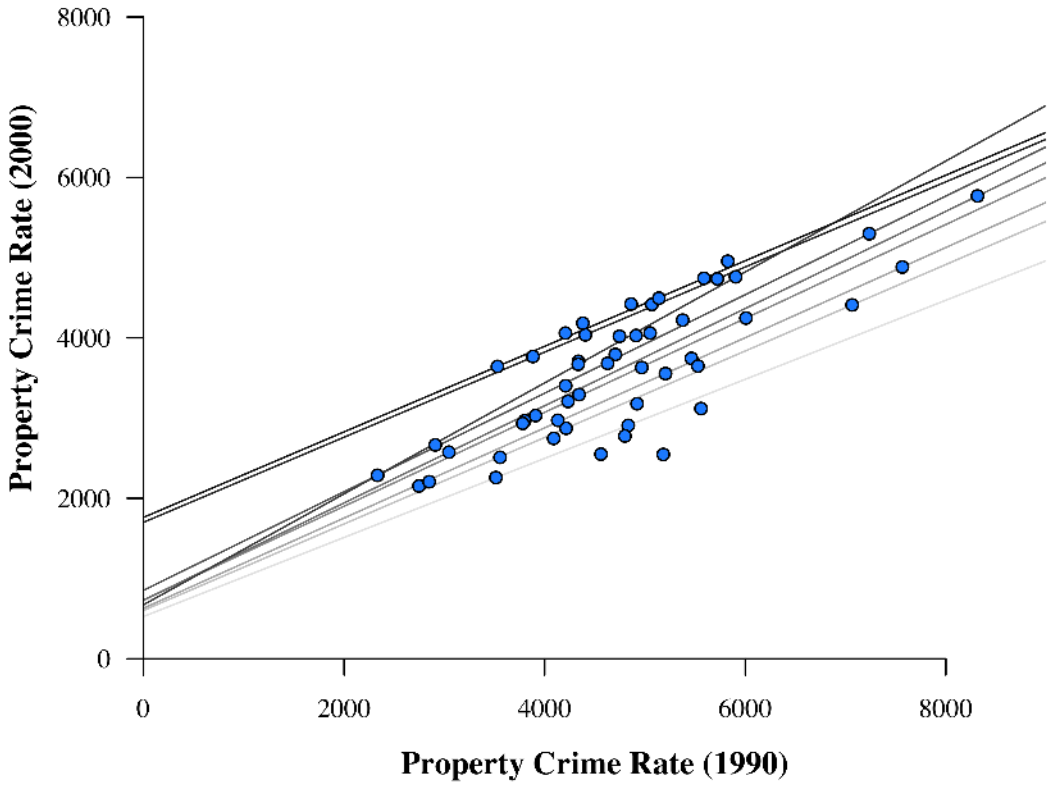


Figure 11.2.1: Graphic illustrating the changing effect based on the quantile examined. The nine lines are regression lines for the deciles 0.10 through 0.90, with darker lines corresponding to higher percentiles.

By the way, Figure 11.2.1 is a graphic of the deciles from 10 to 90% for the relationship between property crime rates in 1990 and 2000. Note that the effect does appear to change as one looks at middle-rate states. The highest levels, quantiles 80 and 90, are very similar in effect to the lower levels, quantile 10 and 20. However, those states near quantile 50 seem to have a greater slope. If we had only looked at the median, we would have only reported these steeper effects. This may have overstated the effect.

Modeling the Property Crime Rate using Wealth

Example 11.2.3: Wealth and Property Crime

What is the relationship between the state's wealth in 1990 and the property crime rate in 2000? Show the effects at the first, second, and third quartiles.

Solution.

We will use the GSP per capita as a proxy measure of wealth in the state. I leave the coding to you. Here is the appropriate output for the median:

```
Call: rq(formula = pcrime00 ~ gspcap90)
tau: [1] 0.5
```

```
Coefficients:
            coefficients lower bd   upper bd
(Intercept) 3061.50674   2383.13082 4989.55600
gspcap90       0.02403     -0.08109    0.04996
```

Interpreting the table indicates that there is no significant evidence that there is a relationship between the average wealth in 1990 and the property crime rate in 2000 for the median state (the confidence interval contains \$0\$).

For the first and third quartile, the conclusions will be the same. As both confidence interval contain both positive and negative numbers, we are unsure of the relationship between these two variables.

■

I leave it as an exercise for you to show that ordinary least squares indicates a statistical significant relationship ($\mathbf{p-value = 0.0475}$). It also provides a point estimate of that relationship of $\boldsymbol{b_1 = 0.03025}$).

Figure **11.2.2** provides the results for all deciles. Note that the slopes also seem to vary according to the quantile examined. Thus, the effect of wealth on property crime rates seem to be a function of those property crime rates. The lowest quantiles suggest the steepest effect. However, performing the analysis shows that the relationship is not statistically significant at the $\boldsymbol{\alpha = 0.05}$ level. In other words, we were unable to detect a relationship.

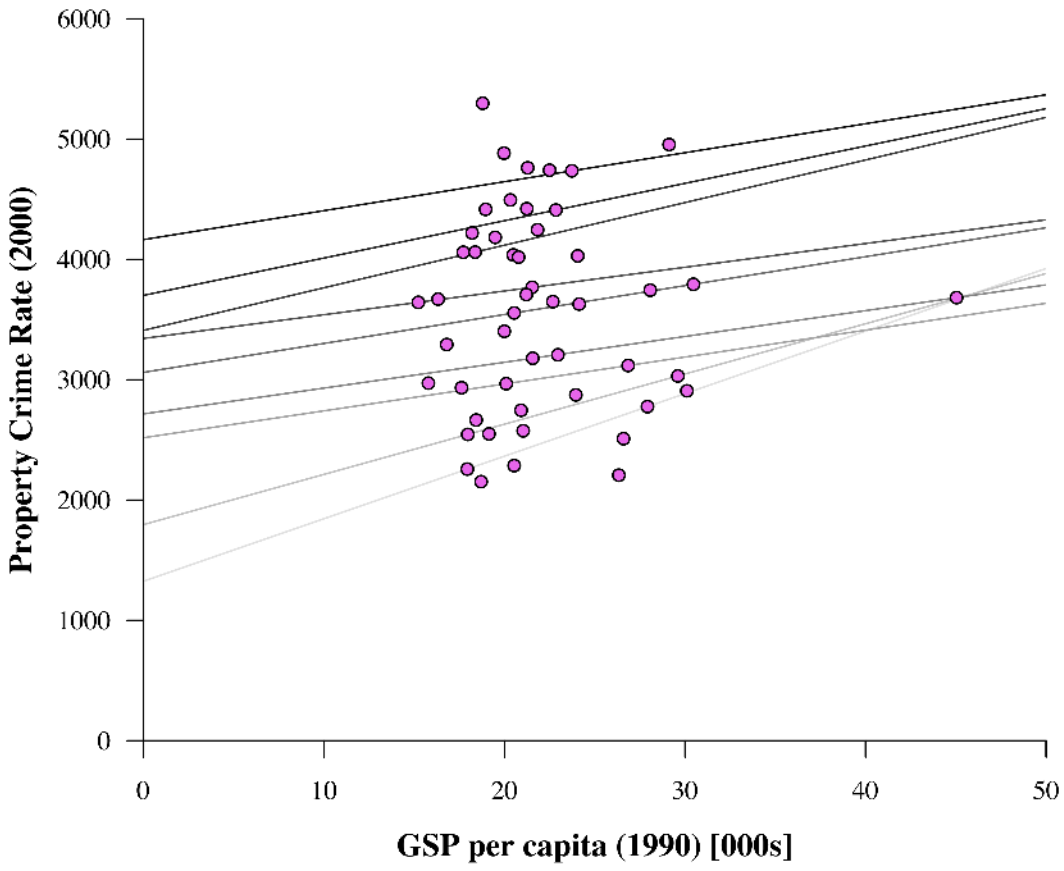


Figure **11.2.2**: Graphic illustrating the changing effect based on the quantile examined. The nine lines are regression lines for the quantiles 0.10 through 0.90, with darker lines corresponding to higher quantiles.

The Ultimate Question

So, is there a relationship between average wealth in 1990 and the property crime rate in 2000? One thing we know is that if there is a relationship, then it is minor. It is not surprising that median regression does not detect a relationship while ordinary least squares does. Median regression, like *all statistics* based on the median (ranks), has a lower power than ordinary least squares (these statistics require Normality).

> So, the answer to the ultimate question is "I'm not sure."
>
> Ultimately, this is unsatisfying. **It is also reality.** By using both OLS and median regression, we have a better understanding of the relationship between average wealth and property crime rates. That is the goal of statistics, not coming up with binary answers.

This page titled 11.2: Quantile Regression is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

11.3: Conclusion

This chapter introduced quantile regression, a powerful expansion of the regression toolkit. Our journey began by fundamentally redefining what constitutes a "best fit." By shifting our focus from minimizing the squared residuals — the cornerstone of OLS — to minimizing their absolute values, we laid the groundwork for estimating the conditional median. This core insight was then generalized through an iterative computational process, allowing us to model not just the center, but any quantile of the conditional distribution, from the 10th percentile to the 90th, with arbitrary precision.

We established that median regression is merely the special case where the quantile is 0.500, placing it within a broader, more flexible framework. The true value of this framework lies in its ability to illuminate dynamics that mean-based models obscure. While OLS, WLS, and GLS tell us how predictors influence the average (expected) outcome, quantile regression reveals how these effects vary across the entire distribution. This is indispensable for research focused on the "wings" — understanding not just typical cases, but also those at the extremes of advantage or disadvantage.

A crucial conceptual point is that quantile regression utilizes the *entire* dataset to estimate these relationships. It does not naively subset the data to observations at a specific quantile, an approach that would be statistically fragile and often impossible, as such exact data points may not exist. Instead, it estimates the *functional relationship* that defines, for any given value of the independent variable, where the qth quantile of the response variable is predicted to lie. In doing so, quantile regression provides a comprehensive, nuanced map of how covariates shape the entire landscape of possible outcomes, making it an essential method for analyzing inequality, assessing risk, and uncovering heterogeneous effects that a single average would inevitably hide.

This page titled 11.3: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

11.4: End-of-Chapter Materials

R Functions

In this chapter, we were introduced to several R functions that will be useful in quantile regression. These are listed here.

Packages

- `quantreg`
  This package contains many functions associated with quantile regression. This chapter just skimmed the surface of what can be done and what should be checked. As this package is not a part of the base R, you will need to install it before loading it.

Statistics

- `rq(formula)`
  This is the function that performs quantile regression. The formula is required. By default, the quantile examined is $\tau = 0.500$, but that can be changed by specifying the value of that τ, "tau."
- `summary(model)`
  This is the familiar command that allows us to see the regression table produced by the regression method. Here, it provides the effect estimates (coefficients) and the central 95% confidence interval for that effect.

Exercises

1. In the setting of Example 11.1.1, perform ordinary least squares regression to calculate the effect estimate and its 95% confidence interval.

Applied Readings

- Lingxin Hao and Daniel Q. Naiman (2008). *Assessing Inequality*. Sage Publishing (Quantitative Applications in the Social Sciences; 166).

Theory Readings

- Ian Barrodale and F. D. K. Roberts (1974). "Solution of an overdetermined system of equations." *Communications of the ACM*. 17(**6**), 319–320.
  doi: 10.1145/355616.361024
- Lingxin Hao and Daniel Q. Naiman (2007). *Quantile Regression*. Sage Publishing (Quantitative Applications in the Social Sciences; 149).
- Roger W. Koenker (2005). *Quantile Regression*. Cambridge University Press.
- Roger W. Koenker and Gilbert W. Bassett, Jr. (1978). "Regression quantiles." *Econometrica*. 46, 33–50.
- Roger W. Koenker and Kevin F. Hallock (2001). "Quantile Regression." *The Journal of Economic Perspectives*. 15: 143–156.
- Roger W. Koenker and Vasco D'Orey (1987). "Algorithm 229: Computing regression quantiles." *Journal of the Royal Statistical Society*. Series C (Applied Statistics). 36: 383–393.
- Roger W. Koenker and Vasco D'Orey (1994). "Remark AS R92: A Remark on Algorithm AS 229: Computing Dual Regression Quantiles and Regression Rank Scores." *Journal of the Royal Statistical Society*. Series C (Applied Statistics). 43: 410–414.
- Yinbo Li and Gonzalo R. Arce (2004). "A Maximum Likelihood Approach to Least Absolute Deviation Regression." *EURASIP Journal on Applied Signal Processing*. 12: 1762–1769.

- Yang Shu-guang and Liao Jian-wen (1992). "Solution of an Overdetermined System of Linear Equations in L_2, L_∞, L_p Norm Using L.S. Techniques." *Journal of Computational Mathematics*. 10: 29–38.

This page titled 11.4: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

12: Maximizing the Likelihood



In the previous chapters, we defined "best" by how the sum of the squared residuals were minimized. We made bad things small. Another way of viewing the "best" line is to maximize good things. That is the idea behind maximum likelihood estimation. While the concepts seem a bit different from ordinary least squares, this method actually leads to the same estimators of β_0 and β_1. It also leads to a *biased* estimator of σ^2. So, why do we look at it here? First, the bias is relatively minor and disappears as the sample size increases.

More importantly, the method is extremely flexible. OLS requires Normality in the residuals. MLE can be used with *any* distribution.

In the previous chapters, we have progressed from our desire to minimize some function of the residuals. This led to several related techniques:

1. ordinary least squares
2. weighted least squares
3. generalized least squares
4. ordinary least absolutes

All of these techniques sought to make the "bad" things as small as possible, to produce a model that minimizes these residuals. However, our first definition of "best" from way back in Chapter 3 was based on making good things as large as possible, where that "good thing" is the likelihood of observing this particular data.

Caution

Technically, it is the "likelihood of the data *given* our parameter estimates."

The theory is that the estimate most likely to have produced the observed data is the "best" estimate. Note that this differs from previous estimation methods in both the objective function and the size we desire. Bigger is better... bigger in terms of the "likelihood."

The fundamental purpose of this chapter is to introduce you to the likelihood and the methods to maximize it. In trying to accomplish this, we will start with the simple and proceed to the less-simple.

· · — · ✶ · — · ·

Chapter Sections

12.1: The Likelihood

12.2: The MLE and the CLM

12.3: Conclusion

12.4: End-of-Chapter Materials

This page titled 12: Maximizing the Likelihood is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

12.1: The Likelihood

This section of this awesome book introduces the foundational concept of likelihood, which serves as the basis for an alternative approach to estimating model parameters known as maximum likelihood estimation (MLE). It clarifies the subtle but important distinction between a probability, which asks about the likelihood of observing certain data given fixed parameter values, and a likelihood, which asks about the plausibility of different parameter values given that specific data have been observed. Through worked examples using the Binomial, Poisson, and Exponential distributions, the section demonstrates how to construct a likelihood function and then maximize it (often using the log-likelihood) to find the parameter value that makes the observed data most probable. These examples reveal that the resulting maximum likelihood estimators are frequently intuitive sample statistics, such as the sample proportion or sample mean.

Learning Objectives

By the end of this section, you will be able to:

1. Distinguish between a probability, which is a function of possible data given fixed parameters, and a likelihood, which is a function of possible parameter values given observed data.
2. Formulate the likelihood function for a given statistical model and dataset, and use it to find the maximum likelihood estimate (MLE) of a parameter by maximizing the likelihood (or log-likelihood) with respect to that parameter.
3. Derive the maximum likelihood estimator for common distributions, such as the Binomial probability of success π, the Poisson rate λ, and the Exponential rate (\lambda), recognizing that the MLE is often a familiar sample statistic (e.g., sample proportion or sample mean).
4. Explain key properties of maximum likelihood estimators, including that they are functions of the data, are asymptotically unbiased, and that the MLE of a function of a parameter is that function of the MLE.

✦•·····• ✦ •·····•✦

From a theoretical standpoint, the likelihood is just a generalization of probability. Where probability is bounded by both 0 and 1, the likelihood is only bounded below by 0. Values with higher probabilities are more likely to be observed. The same is true of likelihood: Values with higher likelihoods are more likely to be observed.

In the discrete case, the likelihood and the probability (mass) are the same. In the continuous case, the likelihood is the probability density. In other words, you really *have* come across the likelihood before. In your previous statistics course, the likelihood was called the "probability density" for continuous random variables and the "probability mass" for discrete random variables.

The difference between the likelihood and the probability mass or density is only one of emphasis. The probability mass (or density) is a function of observable values given the parameters of the distribution.

The Probability and the Likelihood

The likelihood is a function of the parameters, given the observed values (data). That difference is illustrated in the next two examples.

Example 12.1.1a: Probability and the Binomial

Given that the success probability of a binomial random variable is $\pi = 0.25$, what is the probability of observing exactly one success out of two trials?

Solution.
The probability mass function of the binomial distribution is

$$f(x;\ \pi, n) = \binom{n}{x}\ \pi^x\ (1-\pi)^{n-x} \tag{12.1.1}$$

In this particular instance, the probability mass function is

$$f(x;\ \pi = 0.25, n = 2) = \binom{2}{x}\ 0.25^x\ (1-0.25)^{2-x} \tag{12.1.2}$$

And now calculating the probability gives

$$\begin{aligned} f(1;\ \pi = 0.25, n = 2) &= \binom{2}{1}\ 0.25^1\ (1-0.25)^{2-1} && (12.1.3)\\ &= 2\ 0.25^1\ (0.75)^1 && (12.1.4)\\ &= 2\ (0.25)\ (0.75) && (12.1.5)\\ &= 0.375 && (12.1.6) \end{aligned}$$

Thus, the **probability** I observe exactly one success in two trials, given the success probability is 0.25 is just 0.375, which is a probability of 3 in 8.

■

✔ Example 12.1.1*b*: Likelihood and the Binomial

Given that I observed exactly one success in two trials, what is the likelihood that the success probability is $\pi = 0.25$?

Notice that this question is very similar to the previous. The difference is subtle. The previous question asked about the probability of an observation. This one asks about the **likelihood of the parameter**.

Solution.
The likelihood for a binomial random variable is

$$f(\pi;\ x, n) = \binom{n}{x}\ \pi^x\ (1-\pi)^{n-x} \tag{12.1.7}$$

In this particular instance, the likelihood is

$$f(\pi;\ x = 1, n = 2) = \binom{2}{1}\ \pi^1\ (1-\pi)^{2-1} \tag{12.1.8}$$

Thus, the value of the likelihood for $\pi = 0.25$ is

$$\begin{aligned} f(0.25;\ x = 1, n = 2) &= \binom{2}{1}\ 0.25^1\ (1-0.25)^{2-1} && (12.1.9)\\ &= 2\ (0.25)^1\ (0.75)^1 && (12.1.10)\\ &= 2\ (0.25)\ (0.75) && (12.1.11)\\ &= 0.375 && (12.1.12) \end{aligned}$$

Thus, we have calculated the likelihood that $\pi = 0.25$ is 0.375. Is this a lot? It actually depends heavily on the number of data points. In general, the larger your sample size, the smaller the likelihood (of observing that particular data). Thus, the likelihood can only *meaningfully* be interpreted when in relation to other likelihoods **based on the same data**.

■

This last part deserves to be repeated.

Why? It is because we will come across statistics like the AIC and the BIC that can be used for model selection. However, since both are dependent on the likelihood, the values of the dependent variable need to be the same.

> **Caution**
>
> The likelihood can only *meaningfully* be interpreted when in relation to other likelihoods based on the same data.

The Differences?

The probability and the likelihood are *numerically* the same. The use and interpretation, however, are different. With probability, we are looking at a function of possible *outcomes*. With likelihood, we are looking at a function of possible values of the *parameters*. Thus, in the first case, we could ask questions about which value of x is most likely. In the second case, we would ask questions about which value of the parameter (\pi\) is most likely.

Maximizing the Likelihood

Likelihood cries out to be maximized. Is $\pi = 0.25$ the maximum likelihood in the previous example? No. Calculate the likelihood of $\pi = 0.40$ to see that 0.25 is not the maximum (the value of π that produces the largest likelihood value). If you calculated $f(0.40;\ x = 1, n = 2) = 0.48$, then you did the calculations correctly.

Note that $f(0.40;\ x = 1, n = 2) > f(0.25;\ x = 1, n = 2)$. Thus, $\pi = 0.25$ is *not* the maximum likelihood estimate of π in this case (for this data). What is? Such optimization requires using calculus. From the above, you should be able to see that the objective function is

$$Q(\pi) = \binom{2}{1} \pi^1 (1-\pi)^{2-1} = 2\ \pi\ (1-\pi) \tag{12.1.13}$$

This is a function of the parameter, since we are trying to determine the value of π that is *most* likely, given the data. The optimization proceeds as expected:

$$\frac{\mathrm{d}}{\mathrm{d}\pi} Q(\pi) = 2(1-2\pi) \tag{12.1.14}$$

$$0 \stackrel{\text{set}}{=} 2(1-2\hat{\pi}) \tag{12.1.15}$$

$$0 = 1 - 2\hat{\pi} \tag{12.1.16}$$

$$1 = 2\hat{\pi} \tag{12.1.17}$$

$$\frac{1}{2} = \hat{\pi} \tag{12.1.18}$$

Thus, *given* that we observed 1 success in 2 trials (the data), the maximum likelihood estimator of π is $\hat{\pi} = 0.500$.

For some reason, I am not surprised at this outcome. Are you?

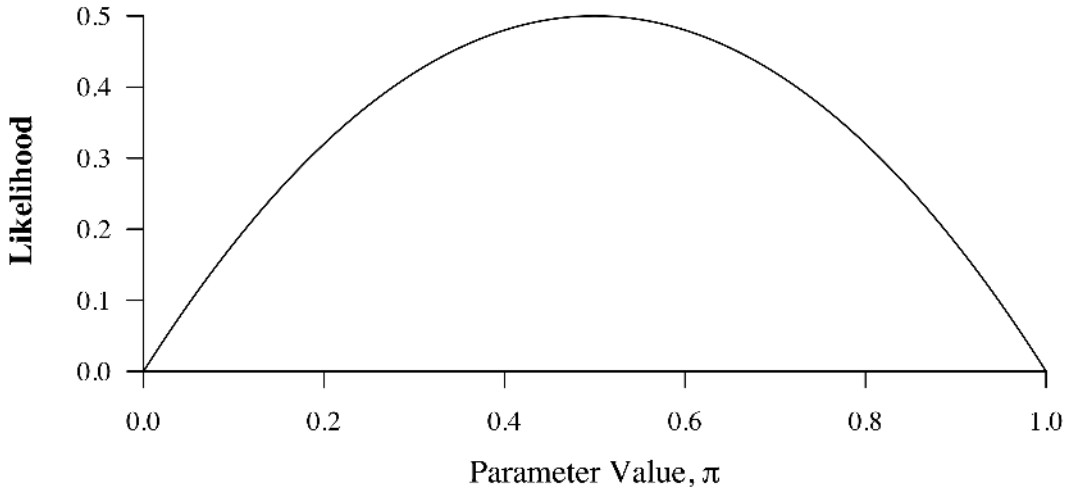


Figure 12.1.1: A graphic showing how the likelihood varies as the parameter changes. Unsurprisingly, the maximum likelihood occurs at $\hat{\pi} = 0.500$.

Figure 12.1.1 shows how the value of the likelihood changes as the parameter value changes. It can be shown that the likelihood for this problem ranges from 0 to 0.5.

In general, one can show that the maximum likelihood estimator of π is $\hat{\pi} = x/n$, where x is the number of successes and n is the number of trials. I will leave that as an exercise.

To prove this, you would perform the same steps, but leave the x and n in the likelihood. If you do the calculations and the calculus correctly, you will end up with

$$\hat{\pi} = \frac{x}{n} \tag{12.1.19}$$

The Poisson Examples

Example 12.1.2a: Valné Shromáždění

Let Y be the number of Ruritanians walking through the door of the Valné Shromáždění, the general assembly building of Ruritania. The King would like to estimate the average number of people entering between noon and 1pm. To do this, the King had his Secretary of the Interior count the number of people entering the building during that hour on Monday.

On Monday, the Secretary counted $y = 17$. With this information, calculate the estimate of λ using maximum likelihood estimation.

A second distribution that you probably saw in your previous statistics class is the Poisson distribution. It has just one parameter, λ, the average rate. In this example, we will determine the maximum likelihood estimator of λ.

Solution.
The likelihood for a discrete distribution, like the Poisson, is just the probability mass function:

$$\mathcal{L}(\lambda;\ y) = \frac{e^{-\lambda}\ \lambda^{y}}{y!} \tag{12.1.20}$$

That is the likelihood of each observation. Here, we only took one measurement (on Monday). Thus this is also the entire likelihood.

The next step is to maximize the likelihood with respect to the parameter, λ:

$$\frac{d}{d\lambda}\mathcal{L}(\lambda;\ y=17) = \frac{d}{d\lambda}\left(\frac{e^{-\lambda}\ \lambda^{17}}{17!}\right) \tag{12.1.21}$$

$$= \frac{-e^{-\lambda}\ 17(\lambda^{17-1}) + e^{-\lambda}\lambda^{17}}{17!} \tag{12.1.22}$$

$$0 \overset{\text{set}}{=} -e^{-\hat{\lambda}}\ 17(\hat{\lambda}^{16}) + e^{-\hat{\lambda}}\hat{\lambda}^{17} \tag{12.1.23}$$

$$0 = -17(\hat{\lambda}^{16}) + \hat{\lambda}^{17} \tag{12.1.24}$$

$$0 = -17 + \hat{\lambda} \tag{12.1.25}$$

Thus, the maximum likelihood estimate of λ is $\hat{\lambda} = 17$.

■

And so, we report to His Majesty that our estimate of the average number of people passing through the doors of the Valné Shromáždění is 17 per hour.

By the way, a graphic of the likelihood function for varying values of λ is given in Figure 12.1.2. Note that the function achieves its maximum when $\lambda = 17$. Thus, the MLE for λ is $\hat{\lambda} = 17$.

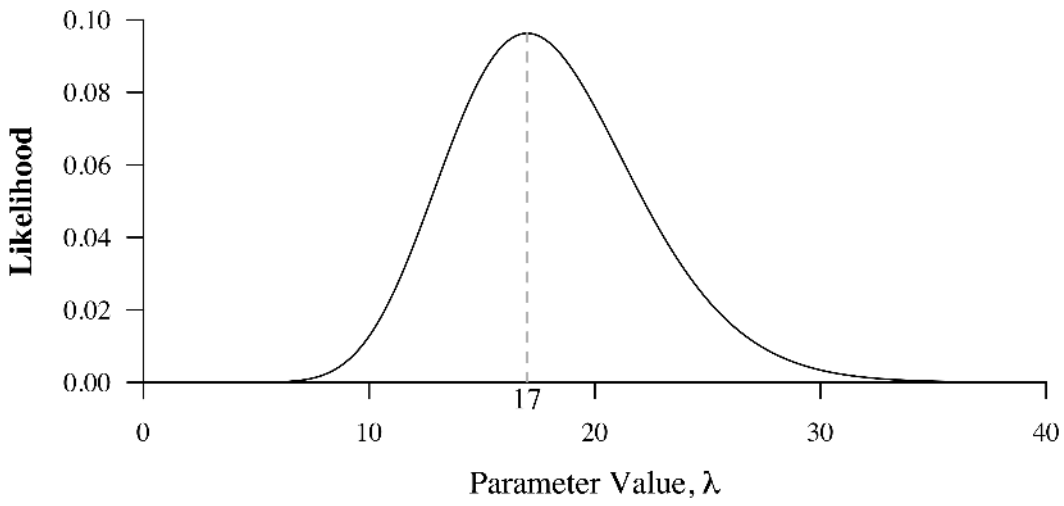


Figure 12.1.2: A graphic showing how the likelihood varies as the parameter changes. This is based on the information in Example 12.1.2*a*.

Example 12.1.2*b*: Valné Shromáždění, Again

His Majesty liked the report, especially the font (he likes serifs). However, he asked an excellent question: "Bylo by lepší měřit více než jednou?"

To address his point, the Secretary of the Interior decided to take multiple measurements over several days. So, for the next week, he measured the number of people entering the Valné Shromáždění an hour at a time, randomly selecting the time of day each time. Here is that data: $15, 20, 23, 34, 23$.

With that new data what is the maximum likelihood estimator of λ, given these $n = 5$ measurements?

Solution.
From the previous example, we know that the likelihood of a single observation is

$$\mathcal{L}(\lambda;\ y) = \frac{e^{-\lambda}\ \lambda^{y}}{y!} \tag{12.1.26}$$

Thus, the likelihood of n independent observations is

$$\mathcal{L}(\lambda;\ y, n) = \prod_{i=1}^{n} \frac{e^{-\lambda}\ \lambda^{y_i}}{y_i!} \tag{12.1.27}$$

How do we know this? Remember from your introductory statistics class the product rule for independent events.

Lemma 12.1.1: Independent Events

Let A and B be two independent events. The probability of both events happening is the product of the individual events. That is

$$P[A \cap B] = P[A] \cdot P[B] \tag{12.1.28}$$

Note

This theorem can easily be extended to any finite number of events. The requirement is that the events are independent. The result is that the probability of all occurring is the product of the probability of each occurring.

Next, since there is a product involved, it will usually be easier to maximize the *logarithm of the likelihood*,

$$\ell(\lambda;\ y, n) = \sum_{i=1}^{n} (-\lambda + y_i\ \log\lambda - \log y_i!) \tag{12.1.29}$$

And so, we maximize *this* target function with respect to λ to obtain our estimator:

$$\frac{\mathrm{d}}{\mathrm{d}\lambda}\ell(\lambda;\ y, n) = \frac{\mathrm{d}}{\mathrm{d}\lambda}\sum_{i=1}^{n}\ (-\lambda + y_i\ \log\lambda - \log y_i!) \tag{12.1.30}$$

$$= \sum_{i=1}^{n}\ -1 + \sum_{i=1}^{n}\frac{y_i}{\lambda} \tag{12.1.31}$$

$$= -n + \frac{n\bar{y}}{\lambda} \tag{12.1.32}$$

$$0 \overset{\text{set}}{=} -n + \frac{n\bar{y}}{\hat{\lambda}} \tag{12.1.33}$$

$$n = \frac{n\bar{y}}{\hat{\lambda}} \tag{12.1.34}$$

Thus, with multiple measurement, the maximum likelihood estimator of λ is

$$\hat{\lambda} = \bar{y} \tag{12.1.35}$$

Before moving on, think about the result to ensure that it makes sense. This is always an important step!

■

Caution

At the end of every result, you should think about its consequences. Make sure the results make sense. If they do not, then double-check your work *or* see the world in a more subtle light.

The Exponential Example

Another important distribution is the Exponential distribution. It is used to model the time until some event occurs. Actuaries may use it to model (estimate) the time until a person dies or gets into an automobile accident or gets sued or some other wonderful event.

It has a single parameter, λ, which is the rate. If you are having *déjà vu* again, do not worry. There is an intimate connection between the Poisson and Exponential distributions. If the time between arrivals follows an exponential distribution, then the number of arrivals follows a Poisson distribution. This means that the average will be $1/\lambda$. Double-check that this actually makes sense.

The following examples deals with this distribution.

Example 12.1.3: Ruritanian Lifetimes

His Majesty has some additional work for us. He would like to estimate the average lifetime of Ruritanians.

Let us use maximum likelihood estimation to provide an estimator for λ, the average *rate* of a person dying (NOT the average time until death).

Solution.
The probability density function for the exponential distribution, when parameterized on its rate, is

$$f(x;\ \lambda) = \lambda\ e^{-\lambda x} \tag{12.1.36}$$

Thus, the likelihood function for a single observation is

$$\mathcal{L}(\lambda;\ x) = \lambda\ e^{-\lambda x} \tag{12.1.37}$$

And, the likelihood function for n independent observations is

$$\mathcal{L}(\lambda;\ x, n) = \prod_{i=1}^{n} \lambda\ e^{-\lambda x_i} \tag{12.1.38}$$

As this is a product, the log-likelihood will be easier to differentiate. It is

$$\ell(\lambda;\ x, n) = \sum_{i=1}^{n} (\log \lambda - \lambda x_i) \tag{12.1.39}$$

Now, we *maximize* it.

$$\frac{\mathrm{d}}{\mathrm{d}\lambda}\ell(\lambda;\ x,n) = \frac{\mathrm{d}}{\mathrm{d}\lambda}\sum_{i=1}^{n}\ (\log\lambda - \lambda x_i) \tag{12.1.40}$$

$$= \sum_{i=1}^{n}\ \frac{1}{\lambda} - \sum_{i=1}^{n}\ x_i \tag{12.1.41}$$

$$= \frac{n}{\lambda} - n\bar{x} \tag{12.1.42}$$

$$0 \overset{\text{set}}{=} \frac{n}{\hat{\lambda}} - n\bar{x} \tag{12.1.43}$$

$$0 = \frac{1}{\hat{\lambda}} - \bar{x} \tag{12.1.44}$$

$$\hat{\lambda} = \frac{1}{\bar{x}} \tag{12.1.45}$$

■

From this, it can be shown that the maximum likelihood estimator of the *mean* of an exponential distribution is

$$\hat{\mu} = \bar{x} \tag{12.1.46}$$

All it takes is knowing that the expected value of an exponential distribution is $\mu = 1/\lambda$.

———— ⋆·☆·⋆ ————

Since the original question dealt with the average age, we would want to calculate $\hat{\mu}$, not $\hat{\lambda}$. I leave it as an exercise to show that a maximum likelihood estimator of μ for the following parametrization of the exponential distribution

$$f(x;\mu) = \frac{1}{\mu}\ e^{-x/\mu} \tag{12.1.47}$$

is $\hat{\mu} = \bar{x}$.

Note

It should be noted that the maximum likelihood estimator is awesome in that functions "pass through." In other words, it can be shown that

$$\widehat{f(x)}_{\text{MLE}} = f(\hat{x}_{\text{MLE}}) \tag{12.1.48}$$

In words, the maximum likelihood estimator of a function of a parameter is that function of the maximum likelihood estimator of the parameter.

The proof is straightforward, but beyond the scope of this textbook.

This is as good a time as any. There are two "**drawbacks**" to using maximum likelihood to estimate parameters. The first is that there is no guarantee that the estimator is **unique**. The second is that there is no guarantee that the estimator is **unbiased**. While these seem bad, there is a nifty theorem that states the MLE is asymptotically unbiased; that is, as the sample size increases, its bias goes to zero.

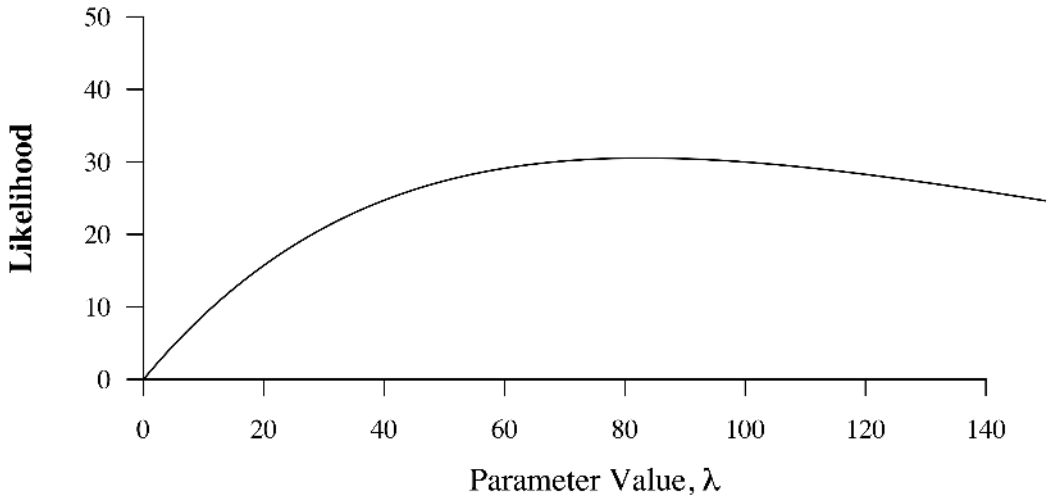


Figure **12.1.3**: A graphic showing how the likelihood varies as the parameter changes.

Figure **12.1.3** shows the likelihood graph for λ of an Exponential distribution. From this graphic, you should be able to estimate the value of μ, the average age of death in Ruritania.

Caution

In a future course, you may be dealing with maximum likelihood estimators frequently. Note that the graphic above tells a story beyond the estimate. It also gives insight into how precise the estimate is. The flatter a graphic around the estimate, the greater the uncertainty.

To see this, note that you probably had more difficulty estimating the maximum in Figure **12.1.3** because of the curve's flatness. The likelihood in Figure **12.1.2** is more sharp, thus it is much easier to determine the maximum value. If you want to explore this, please check out **Fisher Information**, which is defined as

$$\mathcal{I}(\theta) = E\left[\left(\frac{\partial}{\partial\theta}\log f(x;\theta)\right)^2 \;\middle|\; \theta\right] \tag{12.1.49}$$

This page titled 12.1: The Likelihood is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

12.2: The MLE and the CLM

This section demonstrates that maximum likelihood estimation (MLE) produces the same estimators for the regression coefficients β_0 and β_1 in the classical linear model as the method of ordinary least squares. It begins by constructing the likelihood function for the model under the assumption of independent, normally distributed errors. By taking the logarithm of this likelihood and maximizing it with respect to the parameters, the section shows step-by-step that the resulting estimators are algebraically identical to the familiar OLS formulas. The section also derives the MLE for the error variance σ^2, noting that it differs slightly from the unbiased estimator used in OLS and is therefore biased, though asymptotically unbiased. This equivalence confirms that all inferential procedures developed in earlier chapters remain valid when viewed through the lens of maximum likelihood.

Learning Objectives

By the end of this section, you will be able to:

1. Construct the likelihood function for the classical linear model under the assumption of independent, normally distributed errors, and derive the log-likelihood function.
2. Derive the maximum likelihood estimators for the regression coefficients β_0 and β_1 by maximizing the log-likelihood, showing that they are identical to the ordinary least squares estimators.
3. Derive the maximum likelihood estimator for the error variance σ^2 and explain why it is biased, unlike the unbiased estimator MSE used in OLS.
4. Recognize that because the MLEs for the coefficients are identical to the OLS estimators, all inferential results from previous chapters (confidence intervals, hypothesis tests) remain valid when using maximum likelihood estimation.

✦•·····• ✦ •·····•✦

Recall that the simple regression in the classical linear model (CLM) assumes

$$Y_i = \beta_0 + \beta_1\, x + \varepsilon_i \tag{12.2.1}$$

with $\varepsilon_i \overset{\text{iid}}{\sim} N(0;\ \sigma^2)$. When we fit this model using ordinary least squares (OLS), we obtained the following estimators:

$$\begin{cases} b_0 = \bar{y} - b_1\bar{x} \\ b_1 = \dfrac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})^2} \end{cases} \tag{12.2.2}$$

Let us see what we get when we fit this model using **maximum likelihood methods**.

Theorem 12.2.1: MLE of β_0

The maximum likelihood estimator of β_0 is

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1\bar{x} \tag{12.2.3}$$

This is equivalent to the OLS estimator of the y-intercept.

Proof.

The first step is to determine the likelihood function. The second step is to maximize that likelihood with respect to the parameter. As is usual, one maximizes the logarithm of the likelihood instead of the likelihood itself. It is generally easier.

Remember the conditional distribution of y. With that in mind, here is the likelihood for \emph{one} observation:

$$\mathcal{L}(\mu, \sigma^2;\ x, y) = \frac{1}{\sqrt{2\pi\sigma^2}}\ \exp\left[-\frac{1}{2}\ \frac{(y-\mu)^2}{\sigma^2}\right] \tag{12.2.4}$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}}\ \exp\left[-\frac{1}{2}\ \frac{(y-\hat{y})^2}{\sigma^2}\right] \tag{12.2.5}$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}}\ \exp\left[-\frac{1}{2}\ \frac{\left(y-(\beta_0+\beta_1 x)\right)^2}{\sigma^2}\right] \tag{12.2.6}$$

Jeeee-willikers! That is just the probability density function for the normal distribution, where μ (as always) represents an expected value.

That was for a *single* observation. However, we rarely deal with just one data point. We deal with n of them. We remember from our introductory statistics course that **if** the data are independent, then $P[A \cap B] = P[A]P[B]$. That means that the likelihood of observing all of our data is just the product of the individual likelihoods (also see Lemma: Independent Events).

With that, we have

$$\mathcal{L}\left(\beta_0, \beta_1, \sigma^2;\ \mathbf{x}, \mathbf{Y}\right) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\frac{\left(y_i-(\beta_0+\beta_1 x_i)\right)^2}{\sigma^2}\right] \tag{12.2.7}$$

> Since you may not have seen the notation before, $\prod$ is the **p**roduct symbol just like $\sum$ is the **s**ummation symbol.

The next step is to *maximize* this likelihood. From calculus, we recall the product formula for derivatives. Just try applying the product formula here. You will shortly go bald from pulling out your hair. There is no easy way to maximize this likelihood function directly. *Qué lástima!*

However, if we apply an increasing bijection to this likelihood, then maximizing *that* function is equivalent to maximizing the original likelihood... equivalent in terms of the value that produces the maximum (the "argmax").

Because the likelihood has a lot of products, and because it is easier to maximize a sum, we use the logarithm function. The log-likelihood function of the above function is just

$$\ell\left(\beta_0, \beta_1, \sigma^2;\ \mathbf{x}, \mathbf{Y}\right) = \sum_{i=1}^{n} \left(-\frac{1}{2}\log\left(-2\pi\sigma^2\right)\ -\frac{1}{2}\frac{\left(y_i-(\beta_0+\beta_1 x_i)\right)^2}{\sigma^2}\right) \tag{12.2.8}$$

Taking the derivative of a summation is so much easier than taking the derivative of a product... so much easier!

And, now that we have a practically differentiable function, we use calculus to maximize it with respect to β_0:

$$\frac{\partial}{\partial \beta_0}\ell(\beta_0,\beta_1,\sigma^2;\ \mathbf{x},\mathbf{Y}) = \frac{\partial}{\partial \beta_0}\sum_{i=1}^{n} \left(-\frac{1}{2}\log\left(-2\pi\sigma^2\right)\ -\frac{1}{2}\frac{\left(y_i-(\beta_0+\beta_1 x_i)\right)^2}{\sigma^2}\right)$$

$$= \sum_{i=1}^{n}\ -\frac{1}{2}\ \frac{2\left(y_i-(\beta_0+\beta_1 x_i)\right)(-1)}{\sigma^2} \tag{12.2.9}$$

$$= \sum_{i=1}^{n}\ \frac{y_i-\beta_0-\beta_1 x_i}{\sigma^2} \tag{12.2.10}$$

$$0 \overset{\text{set}}{=} \sum_{i=1}^{n}\ \frac{y_i-\hat{\beta}_0-\hat{\beta}_1 x_i}{\sigma^2} \tag{12.2.11}$$

$$= \sum_{i=1}^{n} y_i - \sum_{i=1}^{n}\ \hat{\beta}_0 - \sum_{i=1}^{n}\ \hat{\beta}_1 x_i \tag{12.2.12}$$

$$\sum_{i=1}^{n}\ \hat{\beta}_0 = \sum_{i=1}^{n} y_i - \sum_{i=1}^{n}\ \hat{\beta}_1 x_i \tag{12.2.13}$$

$$n\hat{\beta}_0 = n\bar{y} - n\hat{\beta}_1\bar{x} \tag{12.2.14}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1\bar{x} \tag{12.2.15}$$

Thus, we have shown that $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1\bar{x}$, as we desired. Note that this is also the OLS estimator of the y-intercept. Very interesting!

■

Theorem 12.2.2: MLE of β_1

The maximum likelihood estimator of β_1 is

$$\hat{\beta}_1 = \frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{\sum(x_i-\bar{x})^2} \tag{12.2.16}$$

Proof.

From our proof of the estimator of β_0, we have the following as our log-likelihood function:

$$\ell\left(\beta_0,\beta_1,\sigma^2;\ \mathbf{x},\mathbf{Y}\right) = \sum_{i=1}^{n}\ \left(-\frac{1}{2}\log\left(-2\pi\sigma^2\right)\ -\frac{1}{2}\frac{\left(y_i-(\beta_0+\beta_1 x_i)\right)^2}{\sigma^2}\right) \tag{12.2.17}$$

And so, the proof proceeds by taking the derivative with respect to β_1 and solving for $\hat{\beta}_1$.

$$\frac{\partial}{\partial \beta_1} \ell(\beta_0, \beta_1, \sigma^2;\ \mathbf{x}, \mathbf{Y}) = \frac{\partial}{\partial \beta_1} \sum_{i=1}^{n} \left(-\frac{1}{2} \log\left(-2\pi\sigma^2\right) \ -\frac{1}{2} \frac{\left(y_i - (\beta_0 + \beta_1 x_i)\right)^2}{\sigma^2} \right)$$

$$= -\frac{1}{2} \sum_{i=1}^{n} \frac{2\left(y_i - (\beta_0 + \beta_1 x_i)\right)(-x_i)}{\sigma^2} \tag{12.2.18}$$

$$= \sum_{i=1}^{n} \frac{x_i y_i - \beta_0 x_i - \beta_1 x_i^2}{\sigma^2} \tag{12.2.19}$$

$$0 \stackrel{\text{set}}{=} \sum_{i=1}^{n} \frac{x_i y_i - \hat{\beta}_0 x_i - \hat{\beta}_1 x_i^2}{\sigma^2} \tag{12.2.20}$$

$$= \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} \hat{\beta}_0 x_i - \sum_{i=1}^{n} \hat{\beta}_1 x_i^2 \tag{12.2.21}$$

$$= \sum_{i=1}^{n} x_i y_i - n\bar{x}\hat{\beta}_0 - \sum_{i=1}^{n} \hat{\beta}_1 x_i^2 \tag{12.2.22}$$

$$= \sum_{i=1}^{n} x_i y_i - n\bar{x}\left(\bar{y} - \hat{\beta}_1 \bar{x}\right) - \sum_{i=1}^{n} \hat{\beta}_1 x_i^2 \tag{12.2.23}$$

$$= \sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y} + n\hat{\beta}_1 \bar{x}^2 - \sum_{i=1}^{n} \hat{\beta}_1 x_i^2 \tag{12.2.24}$$

$$\sum_{i=1}^{n} \hat{\beta}_1 x_i^2 - n\hat{\beta}_1 \bar{x}^2 = \sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y} \tag{12.2.25}$$

$$\hat{\beta}_1 \left(\sum_{i=1}^{n} x_i^2 - n\bar{x}^2 \right) = \sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y} \tag{12.2.26}$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2} \tag{12.2.27}$$

We have seen this before. It is the OLS estimator of the slope parameter. No surprise.

To *finish the proof*, use algebra to show that the final equation above is equivalent to

$$\hat{\beta}_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} \tag{12.2.28}$$

■

Theorem 12.2.3: MLE of σ^2

The maximum likelihood estimator of σ^2 is

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i \right)^2 \tag{12.2.29}$$

Proof.
It has been a while, so I will leave this as an **exercise** for you to prove this. I have already shown the log-likelihood function. All you have to do is differentiate with respect to σ^2, solve for $\hat{\sigma}^2$, and use algebra to move things into the right form.

■

Note that the above formula (Eqn 12.2.29) is equivalent to

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} e_i^2 \tag{12.2.30}$$

This *should* raise a red (maybe only light yellow or a nice fuschia) flag, as this is a biased estimator of σ^2.

Note, however, that asymptotically (as $n \to \infty$), the estimator becomes unbiased. This can be proven — more easily than you may imagine.

Actually, this is a guarantee of all maximum likelihood estimators: They may not be unbiased, but they are *asymptotically unbiased*.

Consequences

I leave it as an exercise to see the following consequences:

- $\hat{\beta}_0$ is unbiased for β_0.
- $\hat{\beta}_1$ is unbiased for β_1.
- $\hat{\sigma}^2$ is biased for σ^2.

Because the maximum likelihood estimators are identical to the ordinary least square estimators, and because we have not altered the Normality assumption of the classical linear model, everything from Chapters 3 through 5 hold.

Well, that is not *entirely* true. Remember that the MLE estimator of σ^2 is not the same as the OLS estimator. Thus, the test statistic and confidence interval will need to be altered a bit. However, the differences are minor for large samples.

Multivariate Distributions*

There is one prerequisite to this textbook that would make things a *little* easier: an introduction to multivariate distribution. Thus far, I have "hand-waved" over the topic. Here, I will *briefly* discuss the topic. Feel free to treat this section only as a passing interest.

The following is a **uni**variate distribution:

$$f(x;\ \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\ \exp\left[-\frac{1}{2}\ \frac{(x-\mu)^2}{\sigma^2}\right] \tag{12.2.31}$$

This is the (in)famous probability density function for the normal distribution. It is a function of just one variable (value), x. This is what makes it univariate. The prefix *uni* stands for neither the University of Northern Iowa nor edible sea urchin gonads. It is a Latin combining form for "one." Thus, "univariate" indicates "one variable."

The following is one example of a **bi**variate distribution:

$$f(x,y) = \frac{\exp\left\{-\frac{1}{2(1-\rho^2)}\left[\left(\frac{x-\mu_x}{\sigma_x}\right)^2 - 2\rho\left(\frac{x-\mu_x}{\sigma_x}\right)\left(\frac{y-\mu_y}{\sigma_y}\right) + \left(\frac{y-\mu_y}{\sigma_y}\right)^2\right]\right\}}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \tag{12.2.32}$$

This is a distribution where X and Y are distributed *jointly* normal, and where they have correlation ρ between them. There are a lot of symbols there because it is written in scalar form. Were we to write it in matrix form, we could generalize all of these "—variate" distributions into one form.

If the random vector $\mathbf{Y}$ follows a multivariate normal distribution such that $\mathbf{Y} \sim N_n(\mu;\ \Sigma)$, then

$$f(\mathbf{Y};\ \mu, \Sigma) = (2\pi)^{-n/2}\ (\det \Sigma)^{-1/2}\ \exp\left[-\frac{1}{2}\ (\mathbf{x}-\mu)'\Sigma^{-1}(\mathbf{x}-\mu)\right] \tag{12.2.33}$$

Here, n is the number of *variables* that are jointly normal. This means that each random variable follows a normal distribution, given the values of the others. The vector μ is a column vector of expected values for each X_i. Finally, Σ is the correlation matrix between the n random variables. If the x values are independent, then $\Sigma \in \mathcal{D}_n$ (diagonal). If the x values are independent and identically distributed, then $\Sigma = \sigma^2 \mathbf{I}_n$.

If $n = 1$, then the multivariate normal reduces to the univariate normal. If $n = 2$, then it reduces to the bivariate normal, where the off-diagonal entries in Σ are equal to $\rho\sigma_1\sigma_2$ and the diagonal entries are σ_1^2 and σ_2^2.

That is, if $n = 2$, then

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} \tag{12.2.34}$$

Still, not *much* in this subsection is important — if the observations are independent. If the observations are independent, then the n -variate normal is just the product of the n univariate normals.

HOWEVER, if the observations are *not* independent, however, then the joint distribution — **the actual distribution we care about** — is not so simple. In many cases, it has not been entirely formulated. For instance, what is the multivariate Binomial distribution? That is, what is the distribution of $\{y_1, y_2, y_3, \dots, y_n\}$, given correlation amongst those n measurements?

Even better: How could we measure such correlation?

> In such cases, Dai, Ding, and Wahba (2013) may give you some insight into the difficulty of these questions — and their answers!
>
> This really makes you appreciate random sampling, where both independence and identical distribution hold.

This page titled 12.2: The MLE and the CLM is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

12.3: Conclusion

This chapter has outlined the fundamental process of maximum likelihood estimation (MLE) for linear models, following the established steps of formulating a likelihood function and optimizing it using calculus. Notably, for the classical linear model under the normality assumption, the MLEs for the regression coefficients coincide with the ordinary least squares (OLS) estimators. Consequently, these coefficient estimators share the desirable property of unbiasedness.

However, a key distinction arises in the estimation of the error variance: the MLE of this parameter is biased, which we can see from its divergence from the unbiased mean squared error (MSE). This may initially suggest that MLE offers no advantage over the simpler OLS approach. Yet, such a conclusion would be premature.

MLE provides significant theoretical and practical benefits. First, it is supported by a powerful large-sample (asymptotic) theory, which guarantees that, under general regularity conditions, maximum likelihood estimators are consistent and asymptotically unbiased. This means that for any model where we derive an MLE, its desirable large-sample properties are automatically established, whereas estimators like OLS often require separate, case-by-case proofs.

Second, and more importantly, the MLE framework is remarkably flexible. While the specific results in this chapter relied on the classical assumption of normally distributed errors, the MLE principle itself imposes no such restriction. This allows us to extend our modeling far beyond the confines of the classical linear model. We can specify likelihood functions based on a wide variety of distributions (e.g., binary, count, or duration data), thereby unlocking a vast toolkit for empirical analysis. It is precisely this flexibility that we will leverage in the next segment of the course, as we explore applications of MLE to models that violate the strict assumptions of the classical linear framework.

This page titled 12.3: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

12.4: End-of-Chapter Materials

R Functions

This chapter had no R functions. It was all mathematics and concepts. Yay!!

Exercises

1. Prove that the maximum likelihood estimator of π is x/n in a Binomial experiment.
2. Prove Theorem 12.2.3: The MLE of σ^2.
3. Prove $\hat{\beta}_0$ is unbiased for β_0.
4. Prove $\hat{\beta}_1$ is unbiased for β_1.
5. Prove $\hat{\sigma}^2$ is biased for σ^2 and that the bias is exactly $\frac{n-1}{n}\sigma^2$.

Theory Readings

- Bin Dai, Shilin Ding, and Grace Wahba (2012). "Multivariate Bernoulli Distribution." *Bernoulli*. 19(**4**): 1465--1483. doi: 10.3150/12-BEJSP10
- Dmitry Panchenko (2006). "Lecture 3: Properties of MLE: consistency, asymptotic normality. Fisher information." *Open Courseware/MIT*.
- Aaron Thode, Michele Zanolin, Eran Naftali, Ian Ingram, Purnima Ratilal, and Nicholas C. Makris (2002). "Necessary conditions for a maximum likelihood estimate to become asymptotically unbiased and attain the Cramer-Rao lower bound. II. Range and depth localization of a sound source in an ocean waveguide." *The Journal of the Acoustical Society of America*. 112(**5**): 1890--1910. doi: 10.1121/1.1496765

This page titled 12.4: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

13: Beyond the Classical Model

In this part of the book, perhaps the final part, we expand the framework of the classical linear model (CLM) to address a wider, more realistic range of data types commonly encountered in research. The standard regression model, with its assumption of a continuous, normally distributed dependent variable, is often inadequate. Many outcomes of interest are inherently limited — they may be binary (e.g., success/failure), represent counts (e.g., number of events), or fall into ordered or unordered categories. Generalized Linear Models (GLMs) provide a unified and flexible methodology for these situations by relaxing the assumptions of ordinary least squares regression.

The core innovation of GLMs is their three-component structure:

1. a stochastic component specifying a probability distribution from the exponential family (e.g., Bernoulli, Binomial, Poisson),
2. a linear predictor ($\eta = \mathbf{Xb}$) that functions as in linear regression, and
3. a link function ($g(\cdot)$) that connects the mean of the dependent variable to this linear predictor.

This allows us to model the expected value of non-normal outcomes through a transformation, ensuring predictions remain within meaningful bounds (e.g., between 0 and 1 for probabilities).

We begin with an overview of GLMs in which the conditional distribution is Gaussian (Normal). This will explicitly show that GLMs are just an extension of the CLM. After that bridge, we begin with models for binary dependent variables (logistic regression), where the logit link models the log-odds of an event. This extends to binomial dependent variables for proportions based on a known number of trials. For count data, we employ Poisson regression (and its over-dispersed extensions), using a log link to model rates. Finally, we generalize further to outcomes with multiple categories, exploring multinomial (nominal) regression for unordered choices and ordinal regression for ranked categories, where the link function preserves the inherent order of the response. Together, these tools empower you to model diverse, real-world phenomena with appropriate statistical rigor.

This page titled 13: Beyond the Classical Model is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

14: Generalized Linear Models



Until this point, we have been applying the classical linear model (CLM) to our problems of modeling a dependent variable. It is the model $\mathbf{Y} = \mathbf{XB} + \mathbf{E}$, with Normal errors. While this model is quite prevalent in the literature, it does not always do a good job of approximating reality.

In this chapter, we introduce the generalized linear model (GLM) paradigm and start to show its versatility. We also repeat much of the previous chapter, but from a different perspective, one of paying attention to the data-generating process.

In our regression examples thus far, we have been dealing with continuous dependent variables. The classical linear model (CLM) requires this because the dependent variable needs to be conditionally distributed according to the Normal (a.k.a. Gaussian) distribution. Chapters 3 through 5 discussed this in detail.

Chapter 8: Fixing the Violations examined how we can handle a couple of types of violations of these assumptions, focusing on the case where the dependent variable is bounded. When the dependent variable is bounded, it can*not* be distributed Normal. (Why? What is the support of the Normal distribution?) As such, if your dependent variable *is* bounded, you will have to transform that variable into an unbounded analogue. Once this is done, one might be able to use the methods of the usual CLM paradigm.

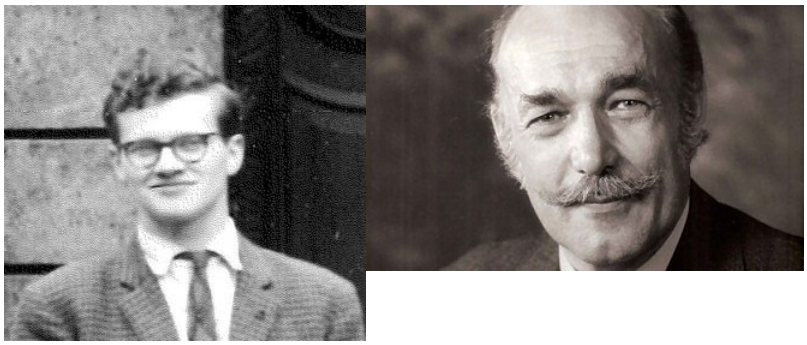
We have, however, encountered some difficulties with this transformation method. In each of our examples from Chapter 8, the dependent variable was bounded — but was *never* equal to its bound. This was necessary. If the dependent variable ever is equal to its bound, then the transformation function you use will return an infinite value (either $-\infty$ or $+\infty$).

In this part of the book, we will extend the classical linear model (CLM) to be more general, and we will introduce a **unifying framework** allowing us to fit many different types of dependent variables — both continuous and discrete.

Note that before the publication of the seminal

- John A. Nelder and Robert W. M. Wedderburn (1972). "Generalized Linear Models." *Journal of the Royal Statistical Society*: Series A (General). 135(**3**), 370–384.
  doi: 10.2307/2344614

each different type of problem tended to have a different type of solution. Logistic regression was a nice procedure, but so was probit regression, and solving one used a totally different technique than the other. This is what John Nelder and Robert William Maclagan Wedderburn added to statistics with their GLM paradigm: not that these problems can be solved, but that these problems can be solved using the same process.



Images of Robert William Maclagan Wedderburn (left, source) and John Nelder (right, source).

· · — · ✶ · — · ·

Chapter Sections

14.1: The CLM and the GLM
14.2: The Requirements for GLMs
14.3: Assumptions of GLMs
14.4: The Gaussian Distribution
14.5: Generalized Linear Models in R
14.6: Conclusion
14.7: End-of-Chapter Materials

This page titled 14: Generalized Linear Models is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

14.1: The CLM and the GLM

This section serves as a bridge between the Classical Linear Model (CLM) and the more flexible framework of Generalized Linear Models (GLM). It begins by reviewing the core assumptions of the CLM — that the response variable is continuous, unbounded, and has constant variance — and acknowledges that real-world data often violate these conditions. While the previous chapters have demonstrated various patches and corrections (such as transformations, weighted least squares, and robust standard errors) to make the CLM work for non-ideal scenarios, this approach becomes increasingly cumbersome and ultimately inadequate for certain types of discrete outcomes like binary or count data. The section therefore introduces the GLM as a new, unified paradigm that relaxes these restrictive assumptions and provides a coherent statistical foundation for modeling a much wider range of response variable types.

Learning Objectives

By the end of this section, you will be able to:

1. Identify the limitations of the Classical Linear Model (CLM), particularly its assumption that the response variable is continuous and unbounded, which makes it poorly suited for discrete outcomes like binary or count data.
2. Recognize that while the CLM can be patched with various corrections (e.g., transformations, weighted least squares, robust standard errors) to handle some violations, these approaches become increasingly complicated and inadequate for certain types of dependent variables.
3. Explain that the Generalized Linear Model (GLM) provides a new, unified framework that relaxes the key assumptions of the CLM, allowing for the modeling of different types of response variables (including binary, count, and nominal data) while maintaining a coherent statistical theory.

✦•·····• ✦ •·····•✦

The Classical Linear Model (CLM) assumes that the relationship between the dependent and the independent variables is linear and that the response variable can take on all possible values; i.e., $\mathbf{Y} \in \mathbb{R}$. Furthermore, to come to statistical conclusions, least squares methods assume that the errors are normally distributed.

Unfortunately, we learned that not all relationships fit this model. Statisticians who realized this, modified the CLM to handle many different types of relationships. Thus, if the dependent variable is continuous and bounded, we modify the dependent variable. If there is heteroskedasticity in the model, we pre- and post-multiply the variance-covariance matrix to better approximate the true standard errors. If you need to weight the data based on some information (such as reliability), you multiply by the weight matrix. And so forth.

However, there are certain types of dependent variables that **can**not be fit using this model (or fit optimally). These are the models with discrete dependent variables. If we want to hold on to the CLM paradigm, we will have to pretend such variables are continuous. Often, this assumption is not a good one. When variables are binary, continuous approximations result in predictions that do not reflect reality. When variables are counts, the variances are functions of the expected value and are heteroskedastic. When the dependent variable is nominal, there is little we can do using the classical linear model.

The Classical Linear Model can *usually* be altered to create good predictions. However, the further your variable is from being continuous and unbounded, the more corrections you will have to make, and the more complicated the process of estimation and prediction becomes — if even possible.

This chapter serves to bridge the gap between the classical linear model (CLM) and the generalized linear model (GLM). In this chapter, we will regenerate the results from the previous chapters, but use a different paradigm. This new paradigm will help us understand the assumptions underlying ordinary least squares regression. It will also serve as a basis for understanding the assumptions of this new modeling paradigm.

This page titled 14.1: The CLM and the GLM is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

14.2: The Requirements for GLMs

The *Generalized* Linear Model (GLM) is a paradigm that extends the CLM and many adjustments to it. To accomplish this feat, the model parts are named and examined. Those three parts are the linear predictor, the conditional distribution of the dependent variable, and the link function. While we have already mentioned all three of these concepts, let us explore them in greater detail before we derive the mathematical results.

Learning Objectives

By the end of this section, you will be able to:

1. Identify and describe the three essential components of a Generalized Linear Model (GLM): the linear predictor ($\eta = \mathbf{XB}$), the conditional distribution of the response variable (a member of the exponential family), and the link function ($g(\mu) = \nu$) which connects them.
2. Select an appropriate distribution for the response variable based on its domain and characteristics (e.g., Gaussian for continuous unbounded, Bernoulli for binary, Poisson for counts), using the provided table as a guide.
3. Explain the role of the link function in mapping the bounded expected value (μ) of the chosen distribution to the unbounded linear predictor (η), and recognize the canonical link associated with each distribution.
4. Recognize that the exponential family of distributions provides the mathematical foundation for GLMs, and understand the general form of its probability density function, $f(y) = \exp\left[(y\theta - b(\theta))/a(\phi) + c(y,\phi)\right]$, and the roles of its more-important components (θ, $b(\theta)$, and $a(\phi)$).

✦•·····• ✦ •·····•✦

Caution

There is a modeling paradigm termed *General* Linear Models, which merely allows for multiple independent variables to the CLM; technically, the CLM uses only one independent variable. General Linear Models are rarely discussed separately from the CLM, as such there is standardized no abbreviation for them. However, authors that do discuss General Linear Models frequently abbreviate them by GLM. These same authors will abbreviate Generalized Linear Models by GLZ.

Upshot: When searching for information on GLMs, make sure you are reading about Generalized Linear Models and not General Linear Models.

The Linear Predictor

Of the three knowledge requirements for using generalized linear models (GLMs), the linear predictor is the most familiar. It is merely the **weighted sum** of your chosen explanatory variables that you used throughout the classical linear model chapters:

$$\eta \overset{\text{def}}{=} \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k \tag{14.2.1}$$

$$= \mathbf{XB} \tag{14.2.2}$$

The *only* difference is that we are providing a name for the weighted sum (η, the Greek letter "eta") and we are calling it the "linear predictor." It is a "linear" predictor because the expression is linear in each of the coefficients (β_i). It is a predictor because it is used to predict the expected value of the dependent variable from the independent variables.

Note that the values produced by the linear predictor are unbounded. That is, note that $\eta \in \mathbb{R}$. This is very important to realize, especially when we get to the third requirement: **the link function**.

The Conditional Distribution

The first "new" addition is the conditional distribution of the dependent variable (its distribution, conditional on the values of the independent variable). Naming it is usually not as difficult as it may seem — a few rules of thumb are very helpful. The distribution chosen reflects your knowledge of the domain of the dependent variable. If the dependent variable can take on all Real values (as before), then an appropriate distribution is the Gaussian distribution (as before). If the dependent variable can take on only values of 0 and 1, then an appropriate distribution is the Bernoulli distribution. And so forth. Table **14.2.1** provides appropriate distributions for several different types of dependent variables (and the chapter in which we discuss them). This is not an exhaustive list, nor are the listed distributions always correct. They are just a good place to start.

> The Gaussian distribution is the eponymous distribution named for Johann Carl Friedrich Gauss (1777–1855). We already know it as the normal distribution. That we are using the name Gaussian reflects standard terminology in GLMs and a desire to give credit where it is due. Well, in Francophone areas, the distribution is known as the Gauss-Laplace distribution to give appropriate credit to Pierre-Simon, Marquis de Laplace (1749–1827). However, Laplace also has his own distribution.
>
> Both the Gaussian and the Laplace distribution were created to describe errors in measurement.

Table **14.2.1**: Listing of some GLM Models

Dependent Variable is...	Default Distribution	Canonical Link	Treated In
Continuous, Unbounded	Gaussian	Identity	14: Generalized Linear Models
Continuous, Bounded by 0	Gamma	Inverse	Not in this book
Discrete, Dichotomous	Bernoulli	Logit	15: Binary Dependent Variables
Discrete, Bounded Count	Binomial	Logit	16: Binomial Dependent Variables
Discrete, Unbounded Count	Poisson	Log	17: Count Dependent Variables
Discrete, Unordered	Multinomial	Logit	18.1: Nominal Dependent Variables
Discrete, Ordered (not interval)	Multinomial	Logit	18.2: Ordinal Dependent Variables

> All of these distributions have something in common: They are members of the **exponential family of distributions** (or exponential *class* of distributions). Section 14.2: The Mathematics (below) discusses why this family of distributions was selected and which distributions belong to it.

The distribution is important in that its expected value automatically restricts the outcome to appropriate values of the dependent variable. Note that we are **explicitly modeling the expected value** of the dependent variable (given the values of the dependent variables). That we are modeling the expected value may sound odd, but we did this previously with the linear models: Our

prediction line was a line of the conditional expectation of the dependent variable, $E[Y \mid x]$. The same is true for GLMs: The fitting routine predicts the expected value of the distribution, $E[Y \mid x]$, *not the observed value*.

The Link Function

The third aspect you need to know in order to use the GLM framework is the link function, which links the linear predictor and the expected value of the distribution. If we symbolize the expected value of the distribution as μ and the linear predictor as η, then the link function is $g(\cdot)$, such that

$$g(\mu) = \eta$$

The most important requirement for the link function is that it maps the bounded domain of the expected value of Y to the unbounded domain of the linear predictor η. An additional requirement is that it is a bijection; that is, the link and its inverse are both functions. It is also usual to make the link a strictly increasing function. This forces the direction of the effect of your variable to be in the same direction as the sign of the estimated coefficient: if the coefficient estimate is positive, then the variable has a positive effect on the dependent variable.

The table of distributions, above, lists the canonical link functions for each of the provided distributions. One can use links that are *not* canonical — and often should — but the canonical link is the default link function used. In subsequent chapters, when an alternate link function is appropriate, we will discuss why.

The Mathematics

Nelder and Wedderburn (1972) formulated the GLM paradigm to unify modeling techniques for several different classes of problems, including logistic regression, count regression, and linear regression. Starting with a member of the exponential family of distributions, Nelder and Wedderburn created an estimation method called iteratively re-weighted least squares (IRLS). This method uses maximum likelihood estimation to estimate the parameter effects using an iterative procedure. MLE remains the primary method of fitting GLMs, but other approaches are used, including maximum quasi-likelihood estimation (which we'll see later), Bayesian estimation, and several variance stabilization methods.

Their choice of MLE was simply one of computing ease. Remember that the early 1970s were a time of sartorial splendor, stagflation, and ABBA — not of cheap computing power. However, even though MLE was chosen for ease, these estimates have some helpful properties. As such, this is still the most widely used method for fitting GLMs, just as OLS has been the preferred method for fitting CLMs for many decades.

Exponential Class of Distributions

The one and only requirement on the distribution is that it belongs to the exponential class of distributions (Nelder and Wedderburn 1974; Wood 2006). Many of the distributions we experience belong to this class, so it is not an issue. Examples of distributions in this class are

- Beta
- Chi-squared
- Exponential
- Gamma
- Gaussian (Normal)
- Geometric
- Poisson
- standard Uniform

Specifically, to be a member of this family, the probability density function (or probability mass function, if discrete) must be expressible in the following form:

$$f(y) = \exp\left[\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\phi)\right] \tag{14.2.3}$$

Let's look at a few features of this form to better understand what each of the parts indicates.

The Mean

The expected value of the distribution is just

$$E[Y] = b'(\theta) \tag{14.2.4}$$

Recall that the expected value is important, as it is what we actually model in GLMs.

The Variance

The variance is

$$V[Y] = b''(\theta) \cdot a(\phi) \tag{14.2.5}$$

The $a(\phi)$ is called the "**dispersion parameter**." Infrequently, the chosen distribution forces this to be a specific value. Usually, however, this variable is free to reflect the data (be estimated from). For those distributions that force this to be a specific number (Binomial and Poisson), we either need to use quasi-likelihood to fit the model *or* we need to test this assumption.

Canonical Link

Next, the θ is the **canonical link function**, $g^{-1}(\cdot)$. It is a function of the parameters of the distribution selected. In the Gaussian case, the canonical link is the identity function, $\mu = \eta$. In the Bernoulli (and Binomial when n is known) case, the canonical link is the logit function, $\text{logit}(\mu) = \eta$, where the logit function is defined as

$$\text{logit}(\mu) \stackrel{\text{def}}{=} \log\left[\frac{\mu}{1-\mu}\right] \tag{14.2.6}$$

In the Poisson case, the canonical link is the logarithm.

Nuisance Parameters

Finally, $c(y,\phi)$ is a term that allows some flexibility to the exponential family of distributions. Without the $c(\cdot)$ function, far fewer distributions would belong to this family. Further, note that the $c(\cdot)$ function affects neither the expected value nor the variance. For all intents and purposes, the $c(\cdot)$ function is ignored and treated as a nuisance parameter.

However, it would be interesting to see what $c(\cdot)$ can tell us about the distribution.

This page titled 14.2: The Requirements for GLMs is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

14.3: Assumptions of GLMs

This section distills the assumptions of Generalized Linear Models (GLMs) into three clear requirements. While the Classical Linear Model rested primarily on the assumption of Normally-distributed errors, a GLM makes assumptions about each of its three core components: the linear predictor must be correctly specified, the conditional distribution of the response variable must be appropriate for the data, and the chosen link function must correctly relate the linear predictor to the expected value. The section emphasizes that violations of any of these assumptions mean the model is missing systematic information present in the data. It also notes that testing these assumptions is more complex than in OLS, as the correctness of the distribution involves understanding its theoretical properties (such as its range and mean-variance relationship) and because heteroskedasticity may be an inherent feature of certain distributions rather than a violation.

Learning Objectives

By the end of this section, you will be able to:

1. State the three core assumptions of a Generalized Linear Model: that the linear predictor ($\eta = \mathbf{XB}$) is correctly specified, that the conditional distribution of the response variable is correctly chosen, and that the link function ($g(\mu) = \eta$) is correct.
2. Recognize that if any of these assumptions are violated, the model is misspecified and fails to capture all the information in the data, potentially leading to biased or inefficient estimates.
3. Explain why testing the assumptions in a GLM is more complex than in OLS, particularly because the correctness of the conditional distribution involves understanding its theoretical properties (range, mean-variance relationship) and because heteroskedasticity may be an inherent feature of the distribution rather than a violation.

✦•·····• ✦ •·····•✦

When we were creating ordinary least squares (OLS) regression, we made one assumption: $\varepsilon \overset{\text{iid}}{\sim} N(0, \sigma^2)$. After learning the mathematics of fitting the models, we went back and figured out how to test these assumptions. The same will be true here.

When performing generalized linear modeling, you make at least three assumptions:

1. you assume the linear predictor is correct;
2. you assume the conditional distribution of the dependent variable is correct; and
3. you assume the link function is correct.

If these assumptions are not met by the data and model, then there is information in the data that you are ignoring.

Testing these is usually not as easy as in the case of OLS regression. The linear predictor and the link function, together, determine the functional form. It can sometimes be tested using a runs test. That is the easy part. Testing the correctness of the conditional distribution is much more involved. It requires that one understands the hypothesized distribution, especially in terms of range, expected values, and variances. Note that tests of heteroskedasticity may not be useful here; many distributions are heteroskedastic by definition.

The testing must be done, however.

Caution

As you read through this part of the book, always keep in mind what we are assuming. That will help you determine the requirements and how to test them.

This page titled 14.3: Assumptions of GLMs is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

14.4: The Gaussian Distribution

To illustrate what we did in the previous sections, let us apply what we know to the Gaussian distribution, and show that it is exponential class. This will allow us to determine the canonical link, the expected value, and the variance. Hopefully, the results will not surprise us.

Learning Objectives

By the end of this section, you will be able to:

1. Demonstrate that the Gaussian (normal) distribution is a member of the exponential family by rewriting its probability density function in the standard GLM form.
2. Identify the components of the Gaussian distribution within this framework: the natural parameter, the dispersion parameter, the cumulant function, and the term.
3. Derive the canonical link function for the Gaussian distribution as the identity link ($g(\mu) = \mu$), and confirm that the expected value and the variance match the known properties of the Normal distribution.
4. Recognize that the GLM framework with the Gaussian distribution and identity link reproduces the Classical Linear Model, illustrating how GLM generalizes the CLM to handle other types of response variables.

✦•·····• ✦ •·····•✦

The Gaussian is Exponential Class

We start with the probability density function (pdf) of the Gaussian.

$$f(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(y-\mu)^2}{2\sigma^2}\right] \tag{14.4.1}$$

Now, to write this in standard form, we use algebra and some logarithm rules.

$$f(y) = \exp\left[-\frac{(y-\mu)^2}{2\sigma^2} + \log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)\right] \tag{14.4.2}$$

$$= \exp\left[-\frac{y^2 - 2y\mu + \mu^2}{2\sigma^2} + \log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)\right] \tag{14.4.3}$$

$$= \exp\left[-\frac{y^2}{2\sigma^2} + \frac{y\mu}{\sigma^2} - \frac{\mu^2}{2\sigma^2} + \log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)\right] \tag{14.4.4}$$

$$= \exp\left[\frac{y\mu - \frac{1}{2}\mu^2}{\sigma^2} + \log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) - \frac{y^2}{2\sigma^2}\right] \tag{14.4.5}$$

Recall from the previous section that the "standard form" is

$$f(y) = \exp\left[\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\phi)\right] \tag{14.4.6}$$

Comparing the two distributions gives us the following:

➡️ $y = y$

$\theta = \mu$
$a(\phi) = \sigma^2$
$b(\theta) = \frac{1}{2}\mu^2 = \frac{1}{2}\theta^2$
$c(y, \phi) = \log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) - \frac{y^2}{2\sigma^2}$

Thus, according to this list, the **canonical link** is $g(\mu) = \mu$, also known as the identity function.

The **dispersion parameter** is $a(\phi) = \sigma^2$.

Also note that the **expected value** is

$$E[Y] = b'(\theta) \tag{14.4.7}$$
$$= \frac{\mathrm{d}}{\mathrm{d}\theta}\left(\frac{1}{2}\theta^2\right) \tag{14.4.8}$$
$$= \theta \tag{14.4.9}$$
$$= \mu \tag{14.4.10}$$

Hopefully, this is as we expect.

Finally, note that the **variance** is

$$V[Y] = b''(\theta)a(\phi) \tag{14.4.11}$$
$$= \frac{\mathrm{d}^2}{\mathrm{d}\theta^2}\left(\frac{1}{2}\theta^2\sigma^2\right) \tag{14.4.12}$$
$$= \frac{\mathrm{d}}{\mathrm{d}\theta}\left(\theta\sigma^2\right) \tag{14.4.13}$$
$$= \sigma^2 \tag{14.4.14}$$

Also as we expect, hopefully.

Other Link Functions

While the canonical link is the identity function ($\eta = \mu$), it is *not* the only allowable link function. In Section 8.1: The Issue of Boundedness, we transformed the continuous dependent variable because it was bounded below by (but never equaled) zero. In such a case, the logarithm is an appropriate link function: The dependent variable has a restricted range. The link function converts that range to an unbounded range. The same is true under the GLM framework. Similarly, the logit function is frequently an appropriate link function, as it was in Section 8.1: The Issue of Boundedness.

With that, we start to see that for continuous dependent variables, what we did under the CLM paradigm we can do under the GLM paradigm. This is *always* true; the GLM paradigm extends the CLM paradigm to handle different classes of dependent variables.

This page titled 14.4: The Gaussian Distribution is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

14.5: Generalized Linear Models in R

This section provides a practical introduction to implementing Generalized Linear Models (GLMs) in R using the `glm` function. It demonstrates how the GLM framework with the Gaussian distribution and identity link reproduces the results of ordinary least squares regression exactly, serving as a bridge between familiar `lm` modeling and the broader `glm` paradigm. The section then explores the key distinction between the Classical Linear Model approach of transforming the dependent variable and the GLM approach of using a non-canonical link function (such as the logit or log link) to handle bounded data. Oh yeah, we also work through two examples: the `cows` and the `gdpcap` datasets.

Learning Objectives

By the end of this section, you will be able to:

1. Use the `glm` function in R to fit a Generalized Linear Model, recognizing that with the Gaussian distribution and identity link, it produces results identical to the lm function for ordinary least squares.
2. Distinguish between transforming the dependent variable in a Classical Linear Model and using a link function (such as logit or log) in a GLM with the Gaussian distribution to handle bounded response variables.
3. Fit a GLM with a non-canonical link in R by specifying the appropriate family and link arguments (e.g., `family=gaussian(link="log")`), and interpret the resulting coefficient estimates.
4. Generate predictions from a fitted GLM and, when necessary, apply the inverse link function (either manually or by using functions like `exp()`) to obtain predictions on the original scale of the response variable.

✦•·····• ✦ •·····•✦

In previous chapters, we performed linear modeling using the `lm` function. To perform *generalized* linear modeling, we use the `glm` function. When one uses the Gaussian distribution and its canonical link, results between the two methods will be *identical*. That is, we could have replaced all of our previous uses of `lm` with `glm` and not change a thing.

Be Aware!

If one uses the Gaussian distribution and a non-canonical link, the predictions will be very close, but not identical. The reason is that the transformation is performed on different quantities between the two methods... as the following shows.

To see this, we can look at two things. The **first** is focusing on what the alteration applies to (the residuals????).

When transforming the dependent variable:

$$g^{-1}(\mathbf{Y}) = \mathbf{XB} + \mathbf{E} \tag{14.5.1}$$

$$\Rightarrow \quad \mathbf{Y} = g(\mathbf{XB} + \mathbf{E}) \tag{14.5.2}$$

When using the link function:

$$g^{-1}(E[\mathbf{Y} \mid \mathbf{X}]) = \mathbf{XB} \tag{14.5.3}$$

$$E[\mathbf{Y} \mid \mathbf{X}] = g(\mathbf{XB}) \tag{14.5.4}$$

$$\mathbf{Y} = g(\mathbf{XB}) + \mathbf{E} \tag{14.5.5}$$

So, the only difference is in whether the function applies to the residuals. In the CLM, it does; in the GLM, it does not. This is why there will be (usually slight) differences between the CLM transformed and the GLM with a link function.

Second, we can see this ourselves in an old example, use the GLM paradigm to find the answers, and see that the results are slightly different from when the model fit when transforming the dependent variable.

Example 14.5.1: Voting in Děčín

Let us return to the `cows` data file. The voters of Děčín are being sent to the polls to vote on a constitutional referendum (ballot measure) that proposes to limit the number of cows that can be housed within the city limits. This was not the first time that Ruritanians were sent to the polls to vote on this or a closely related issue. Given the information from previous votes, what is the estimated proportion of voters who will vote in favor of the ballot measure in Děčín?

Solution.
The example asks us to estimate the proportion of voters who will vote in favor of the ballot measure in Děčín. The dependent variable will be `propWin` and the independent variables will be `yearPassed`, `chickens`, and `religPct`. For now, let us assume a linear relationship between the independent variables and the dependent variable; that is, the equation we will use to fit the data is

$$\mathbf{propWin} = \beta_0 + \beta_1(\mathbf{yearPassed}) + \beta_2(\mathbf{chickens}) + \beta_3(\mathbf{religPct}) + \varepsilon \tag{14.5.6}$$

This is equivalent to

$$E[\mathbf{propWin}] = \beta_0 + \beta_1(\mathbf{yearPassed}) + \beta_2(\mathbf{chickens}) + \beta_3(\mathbf{religPct}) \tag{14.5.7}$$

which is more clearly connected to the GLM paradigm than before.

Performing Generalized Linear Modeling in R is straight-forward (as it is in all modern statistical packages). The function to use is `glm` (for "General*ized* Linear Modeling"):

```
glm(propWin ~ yearPassed + chickens + religPct)
```

As `glm` returns a lot of information, we should store its results in a variable, which I will call `mod1`. Once the computer computes the regression (and all associated information), we can summarize the results in the standard results table (below) using the command

```
summary(model.1)
```

The resulting output is

```
                        Estimate   Std.Err   t-value    p-value
Constant Term             0.1512    0.0659     2.293     0.0295
Year Passed (post 2000)  -0.0201    0.0036    -5.618  << 0.0001
Banned Chickens          -0.0373    0.0200    -1.868     0.0723
Percent Religious         0.0095    0.0011     8.801  << 0.0001
```

Notice that all three variables of interest are statistically significant at the $\alpha = 0.05$ level. Additionally, the model has a residual deviance of 0.063072 (as compared to the null deviance of 0.286802). This indicates that the model reduced the deviance by a factor of

$$1 - \frac{0.063072}{0.286802} = 0.7801 \tag{14.5.8}$$

And this agrees with the R^2 from Section 7.3: Cows in Děčín.

Thus, the equation for the line of best fit (also known as the prediction line) is approximately

E[propWin] = 0.1512 - 0.0201 (yearPassed) - 0.0373 (chickens) + 0.0095 (religPct)

According to this model, what is the expected vote in Děčín? To answer this, we need this information about the Děčín ballot measure: yearPassed = 9, chickens = 0, religPct = 48. With this information, and under the assumption that the model is correct, we have our prediction that 42% of the Děčín voters will vote in favor of this ballot measure.

■

There is **absolutely nothing in the previous paragraphs** that differs from the analysis results from when we answered the same question using OLS. This is because the Generalized Linear Model paradigm *extends* the Classical Linear Model paradigm and is *equivalent to it* when the dependent variable is Gaussian distributed and the link is the identity function. We can even use something like the goodness-of-fit measure we developed in Section 3.4: PRE Measures, the R^2 measure. Here, however, we calculate it based on the **null** and **residual deviances**. The null deviance is the deviance inherent in the data (akin to the variance of the data, TSS). The residual deviance is the deviance in the data unexplained by the model (akin to the SSE).

If we wish to predict the results of a Venkovský ballot measure from 1994, which also restricted chickens, we would still get an impossible prediction --- one that is outside logical limits. In Section 8.1: The Issue of Boundedness, we corrected this error by transforming the data, modeling, then back-transforming the results. Instead of transforming the dependent variable, let us merely change the link function. Here is how that is done in R and with glm :

The command to use is

```
mod3 = glm(propWin ~ yearPassed + chickens + religPct,
           family=gaussian(link=make.link("logit")))
```

The results of this new analysis are

```
                         Estimate   Std.Err   t-value     p-value
 Constant Term            -1.8909    0.2898     -6.53   << 0.0001
 Year Passed (post 2000)  -0.0886    0.0157     -5.63   << 0.0001
 Banned Chickens          -0.2318    0.0878     -2.64      0.0134
 Percent Religious         0.0475    0.0047     10.06   << 0.0001
```

... and the resulting graphic comparing the CLM and the GLM methods is

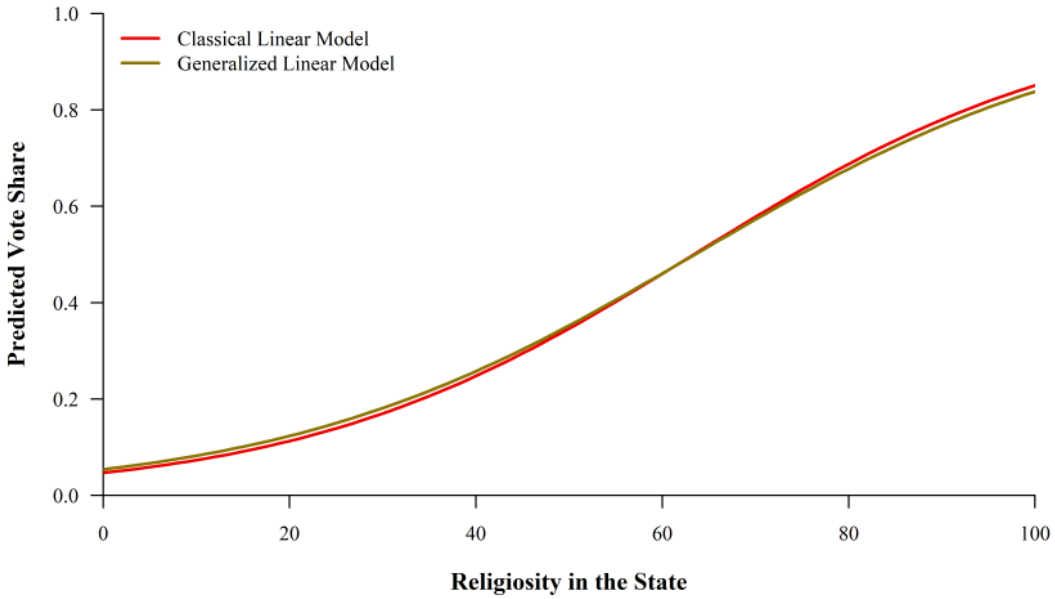


Figure **14.5.1**: A plot of the predictions across various values of religiosity comparing the two models: CLM and GLM. Note that while the two results tables provided different results, the prediction plots are quite close together. The curves would have been equal only if we were to use the canonical link and the Gaussian distribution. For the predictions, the year was 2010 and the ballot measure also banned chickens.

Now, `summary(mod3)` provides many results. Note that all three independent variables are more statistically significant than in the non-transformed model. Also note that the effect directions are the same as before.

Finally, note that these parameter estimates are *not* the same as those where we used the Classical Linear Model with a logit transformation to fit the data in Section 8.1: The Issue of Boundedness. However, if we make predictions, we see that the results are *very* close (Figure **14.5.1**). CLMs and GLMs give identical results **only with** the Gaussian distribution and its canonical link. Here, we used the logit link.

Let us now re-examine Example 8.1: The Issue of Boundedness from Chapter 8: Fixing the Violations. Recall that in *that* example, we were modeling a variable that was bounded below, but not above. This led us to transform the dependent variable using the logarithm function. Here, we fit the model with the Gaussian distribution and another non-canonical link.

✔ Example 14.5.2: Wealth and Corruption

The gross domestic product (GDP) per capita is one of many measures of average wealth in countries. If extant theory is correct, then the wealth in the country is directly affected by the level of honesty in the government --- countries with high levels of honesty (low levels of corruption) should be wealthier than those with low levels of honesty (high levels of corruption).

Furthermore, if theory is correct, the level of democracy in a country should *also* influence the country's level of wealth --- countries with higher levels of democracy should be wealthier than countries with low levels of democracy.

Let us determine if reality (using the data in the `gdpcap` data file) supports the current theory or if current theory needs to explain the severe discrepancies.

Solution.

The process of fitting this model with a GLM should be getting rote by now as it is so similar to fitting with a CLM. The R

command is

```
m2 = glm(gdpcap ~ democracy + hig,
         family=gaussian(link = make.link("log")))
```

To see the results, we perform a `summary` call. The results of that call are provided in the regression table below. Note that both independent variables are highly significant at the usual level of significance. Furthermore, the effect directions are the same as in the CLM model. Here is the regression table.

```
                        Estimate    Std.Err    t-value     p-value
Constant Term             8.1595     0.1546      52.77   << 0.0001
Level of Democracy       -0.0452     0.0061      -7.44   << 0.0001
Honesty in Government     0.3335     0.0219      15.20   << 0.0001
```

The graphic is

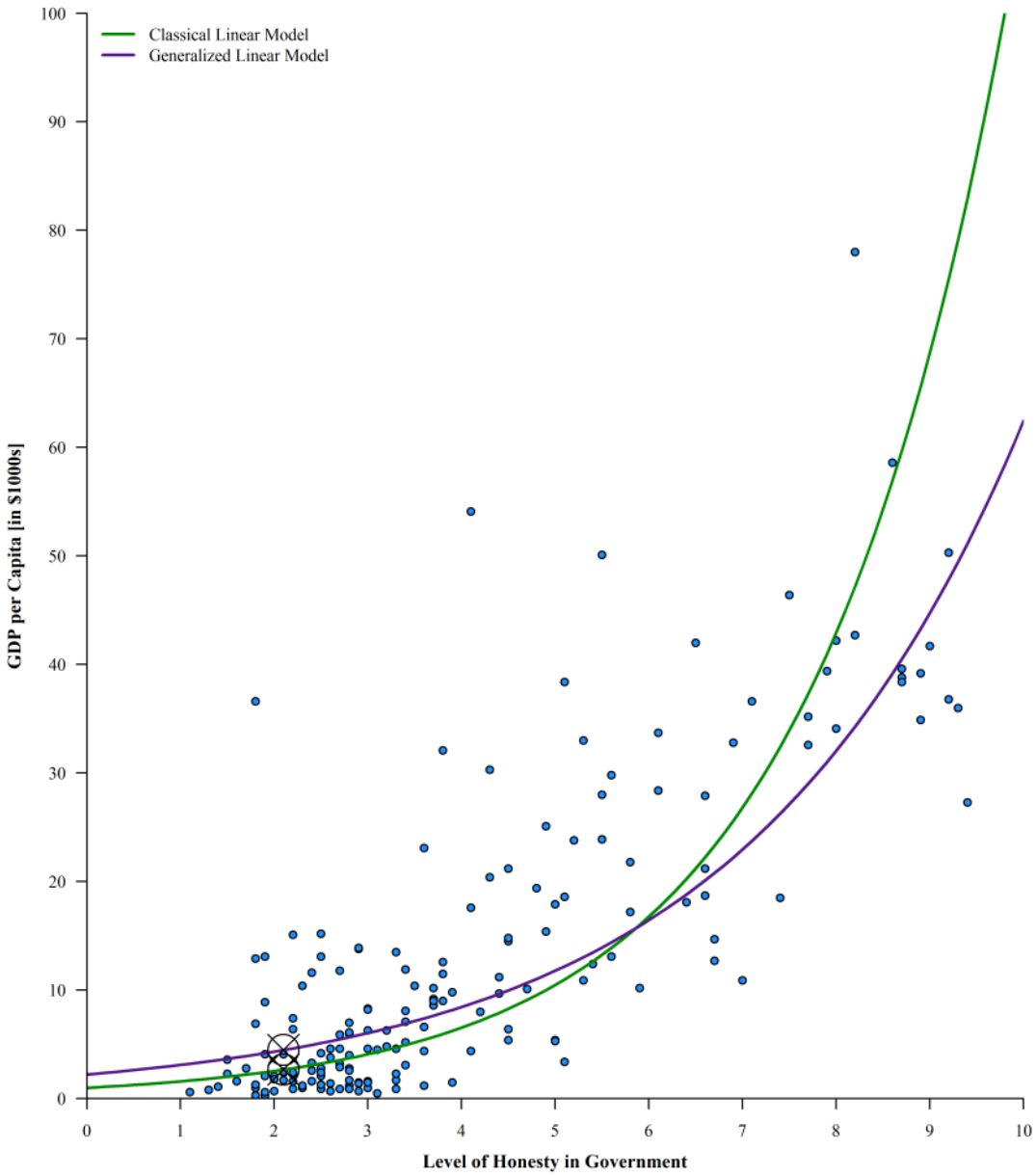


Figure **14.5.2**: A plot of the two prediction curves, corresponding to the model fit using the Classical Linear Model and the Generalized Linear Model. Note that the two prediction curves are similar, but not really that close for large values of honesty in government. Estimates for Papua New Guinea are shown with the two circle-x symbols.

For some link functions, R allows you to skip the `make.link` portion. The `log` link is one of those for the Gaussian. Thus, the following command would *also* work:

```
glm(gdpcap ~ democracy + hig, family=gaussian(link="log"))
```

I recommend writing it out. That helps those who follow you to interpret what you are doing.

To predict the GDP per capita for Papua New Guinea, we repeat the same steps as when we were fitting CLMs: predict, then back-transform. Thus, a prediction statement will be

```
PNG = data.frame(hig=2.1, democracy=10)
exp(predict(m2, newdata=PNG))
```

The predicted GDP per capita for Papua New Guinea was $2678 when fitted with the CLM. For this GLM model, the prediction is $4481. Thus, the prediction for Papua New Guinea is higher using GLMs than when using CLMs. Looking at the prediction graph (Figure **14.5.2**), we see that GLM predictions are lower than CLM predictions for certain values of the dependent variable (and larger for others).

This page titled 14.5: Generalized Linear Models in R is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

14.6: Conclusion

This chapter introduced the Generalized Linear Model paradigm, which is an extension of the Classical Linear Model paradigm from the previous two chapters. The advantage of the GLM paradigm is that more classes of dependent variables can be fit. The disadvantage (*if* we can call it that) is that we need to understand our data and model better. The three things we need to know are the linear predictor, the distribution of the dependent variable, and the function that links the expected value of the distribution with the linear predictor.

We tied this chapter to the previous chapters by showing that a GLM model using the Gaussian distribution (and the identity link) is *equivalent* to using the CLM. Three examples showed that the steps in modeling using the Generalized Linear Model paradigm are very similar to the steps used in modeling using the Classical Linear Model paradigm.

This chapter actually marked a major departure in how we see our data. Before, whenever a datum was different from our prediction, we viewed it as an error. Now, we realize that this variation is simply due to random fluctuations. We know this because we realize that our dependent variable is a random variable.

In the next chapters in this part of the book, we will examine more classes of dependent variables: binary, limited discrete (both nominal and ordinal), count, and non-negative continuous. As we examine these classes, pay attention to the selected distribution and the possible link functions. **This table** provides several of the distributions and their canonical link functions.

Before you move on to the next chapter, ask yourself one thing:

> **What requirements do we need to check for our models in this chapter?**

Make sure you can explain why they are requirements, too.

This page titled 14.6: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

14.7: End-of-Chapter Materials

R Functions

In this chapter, we were introduced to several R functions that will be useful in the future. These are listed here.

Packages

There were no packages used in this chapter.

Statistics

- `lm(formula)`
  This function performs linear regression on the data using ordinary least squares to estimate the parameters. As there is much information contained in this function, you will want to save the results in a variable.

- `glm(formula)`
  This function performs *generalized* linear model estimation on the given formula using maximum likelihood estimation. There are three additional parameters that can (and often should) be specified.
  - The `family` parameter specifies the distributional family of the dependent variable, options include `gaussian` (this chapter), `binomial`, `quasibinomial`, `poisson`, `quasipoisson`, and `gamma`. If this parameter is not specified, R defaults to `gaussian`.
  - The `link` parameter specifies the link function for the distribution. If none is specified, the canonical link is assumed.
  - Finally, the `data` parameter specifies the data from which the formula variables come. This is the same parameter as in the `lm` function. This is important if we are working with several data sets and need to specify which set of variables is to be used.

- `predict(model, newdata)`
  As with almost all statistical packages, R has a predict function. It takes two parameters, the model, and a dataframe of the independent values from which you want to predict. If you omit `newdata`, then it will predict based on the independent variables of the data itself, which can be used to calculate residuals. The dataframe must list all independent variables with their associate new values. You can specify multiple new values for a single independent variable.

Exercises

This section offers suggestions on things you can practice from just the information in this chapter. As the purpose of this chapter was to introduce Generalized Linear Models and emphasize that everything we have done thus far can be done with GLMs, all of the extension questions are from previous chapters. For each of these, use the Generalized Linear Model paradigm (and the `glm` function).

Summary

1. What are the three aspects of your model that must be known before using generalized linear models?
2. When doing ordinary least squares regression, what were these three aspects?
3. How does the canonical link function differ from a link function?
4. What is $a(\phi)$ for the Gaussian distribution?

Data

5. Now, note that the value for the Reka *stat* is 46% weekly church attendance. If, in the year 2012, the voters of Reka were faced with a ballot measure limiting the number of cows in the city limits, but not restricting chickens, what is the probability that it will pass?

6. Calculate a 95% confidence interval, with the *transformed* Cow Vote model, for predicting Děčín's vote. Is the actual outcome within the 95% confidence interval?

7. The logit transformation is not the only possible choice as a link for proportion data, there is also the asymmetric "complementary log-log" transformation (`cloglog`). Use this function as the link function to predict Děčín's vote, its 95% confidence interval, and the probability of the Cow ballot measure passing. The inverse of the complementary log-log transform has no name, but the R function is `cloglog.inv` .

8. Estimate the GDP per capita for Papua New Guinea. For this problem, use the untransformed model. Also, calculate a 95% confidence interval for this estimate. How close is this estimate to the real answer, and it the real answer within the predicted confidence interval?

9. Estimate the GDP per capita for Papua New Guinea. For *this* problem, use the *transformed* model. Also, calculate a 95% confidence interval for this estimate. How close is this estimate to the real answer, and it the real answer within the predicted confidence interval?

10. Compare and contrast the results of your two Papua New Guinea estimates above. Which model works best for Papua New Guinea? Which model works best overall?

Applied Readings

- Denise Gammonley, Ning Jackie Zhang, Kathryn Frahm, and Seung Chun Paek (2009). "Social Service Staffing in U.S. Nursing Homes." *Social Service Review*. 83(**4**): 633–50.
- Katarina A. McDonnell and Neil J. Holbrook (2004). "A Poisson Regression Model of Tropical Cyclogenesis for the Australian–Southwest Pacific Ocean Region." *Weather & Forecasting*. 19(**2**): 440–455.
- Michael A. Neblo (2009). "Meaning and Measurement: Reorienting the Race Politics Debate." *Political Research Quarterly*. 62(**3**): 474–84.
- Weiren Wang and Felix Famoye (1997). "Modeling Household Fertility Decisions with Generalized Poisson Regression." *Journal of Population Economics*. 10(**3**): 273–83.

Theory Readings

- Hirotugu Akaike (1974). "A New Look at Statistical Identification Model." *IEEE Transactions on Automatic Control*. 19(**6**): 716–23.
- Hirotugu Akaike (1977). "On Entropy Maximization Principle." In: P. R. Krishnaiah (Editor). *Applications of Statistics: Proceedings of the Symposium Held at Wright State University, Dayton, Ohio, 14-18 June 1976*. New York: North Holland Publishing, 27–41.
- George Casella and Roger L. Berger (2002). *Statistical Inference*, Second edition. New York: Duxbury.
- Carl F. Gauss (1809). "*Theoria motus corporum coellestium*." Werke, 7, Göttingen: K. Gesellschaft Wissenschaft.
- Peter J. Huber (1967). "The Behavior of Maximum Likelihood Estimates under Nonstandard Conditions." *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. 221–233.
- Pierre-Simon Laplace (1812). "*Théorie analytique des probabilités*." Paris.
- Peter McCullagh and John A. Nelder (1989). *Generalized Linear Models*. London: Chapman and Hall.
- John A. Nelder and Robert W. Wedderburn (1972). "Generalized Linear Models." *Journal of the Royal Statistical Society* Series A (General). 135(**3**): 370–84.

- Samuel S. Wilks (1938). "The Large-Sample Distribution of the Likelihood Ratio for Testing Composite Hypotheses." *The Annals of Mathematical Statistics*. 9(**1**): 60–62.
- Simon N. Wood (2006). *Generalized Additive Models: An introduction with R*. New York: Chapman & Hall.
- Halbert White (1980). "A Heteroskedasticity-Consistent Covariance Matrix Estimator and a Direct Test for Heteroskedasticity." *Econometrica*. 48(**4**): 817–838.

This page titled 14.7: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

15: Binary Dependent Variables



Thus far, we have examined linear regression where the dependent variable is continuous — either unbounded or bounded. These cases cover a wide variety of instances — but not all. Examples of dependent variables we can now use include heights, incomes, vote proportions, distances, and so forth.

However, we do not yet have the ability to handle dependent variables which are discrete. Such variables include dichotomous variables (presence of a characteristic), count variables (ages, deaths, numbers of fires), ordinal variables (importance level), and nominal variables (different outcomes). These types of variables are all limited in that there are *adjacent* outcomes. This chapter deals with modeling dichotomous (binary) random variables.}

The previous chapter introduced the generalized linear model paradigm (GLM). Modeling with GLMs requires that we specify three things:

1. the conditional distribution of the dependent variable (thus a formula for the expected value of the dependent variable), μ;
2. the linear predictor, $\eta = \mathbf{XB}$; and
3. a (bijective) function linking the two, $g(\mu) = \eta$.

In *that* chapter, we showed that the Classical Linear Model is just a special case of the Generalized Linear Model. Specifically, the CLM is just a GLM using the Gaussian (Normal) distribution and the identity link. In this chapter, we cover the case of **dichotomous** (binary) dependent variables. In the following pages, we determine the appropriate distribution and the canonical link function.

Chapter Sections

15.1: Binary Dependent Variables
15.2: Latent Variable Modeling
15.3: The Mathematics
15.4: Modeling with the Logit
15.5: Measuring Accuracy

15.6: Modeling with Other Links
15.7: Model Selection
15.8: Conclusion
15.9: End-of-Chapter Material

This page titled 15: Binary Dependent Variables is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

15.1: Binary Dependent Variables

This section introduces the challenges of modeling a binary dependent variable — a variable that can take only one of two values, such as yes/no or success/failure — within the Classical Linear Model (OLS) framework. Using a concrete example predicting life insurance purchase from age and income, you see three fundamental ways OLS fails for such data: it produces predictions outside the logical 0-1 range, it violates the assumption of homoskedasticity, and its residuals are not normally distributed. The section frames these violations not as dead ends, but as valuable signals that the data contain more information than the model is capturing, motivating the need for a different modeling approach specifically designed for binary outcomes.

Learning Objectives

By the end of this section, you will be able to:

1. Identify a binary dependent variable and explain why the Classical Linear Model (OLS) is generally inappropriate for such data, as it leads to predictions outside the 0-1 range and violates key assumptions like constant variance and normality of residuals.
2. Diagnose the failure of OLS for a binary outcome by examining residual plots (e.g., residuals vs. fitted values) which reveal patterns of heteroskedasticity, indicating that the model is misspecified and ignoring systematic information in the data.
3. Recognize that while ad-hoc solutions like creating a decision rule to clamp predictions to 0 or 1 can address illogical predictions, they do not resolve the underlying violations of model assumptions, which affect uncertainty estimates and inference.

✦•·····• ✦ •·····•✦

A dichotomous variable is one that can take one of two values: 1 or 0, True or False, Yes or No, success or failure. In research, these variables include the *occurrence* of terrorism, the *election* of a specific party to power, the *existence* of a fire, and the *failure* of a plane. In each of these cases, there are only two possible values, which we will refer to as success and failure. This is the hallmark of dichotomous variables. Before Nelder and Wedderburn (1972) created the GLM framework, statisticians created special models for binary dependent variable problems (with different transformations).

They did so because the classical linear model invariably makes predictions outside the logical range, demonstrates heteroskedasticity, and has residuals that are not Normally distributed — all violations of the OLS assumptions. To illustrate this, let us model the decision to purchase life insurance using age and income and the classical linear model (fit using OLS). The next example illustrates these issues.

Example 15.1.1: Life Insurance

In Ruritania, the decision to buy life insurance is related to several variables, including age and income. The table below includes records of several individuals. Fit this data with a linear model using OLS:

$$\texttt{insurance} = \beta_0 + \beta_1 \texttt{age} + \beta_2 \texttt{income} \tag{15.1.1}$$

Next, predict whether Václav will buy life insurance, given that his age is 65 and his income is $125,000. Finally, determine if the assumptions of ordinary least squares are violated with this model and data.

Table 15.1.1: Insurance synthetic data, in which we predict a person purchasing life insurance based on the

person's age and income.

Individual	Insurance	Age	Income ($000)
1	0	25	20
2	0	30	30
3	0	21	30
4	0	35	25
5	0	28	27
6	1	80	90
7	1	55	25
8	1	40	60
9	1	40	65
10	1	25	125

Solution.

Using our statistical program, we get the following as our linear regression equation, when fit using OLS:

$$\texttt{insurance} = -0.4277 + 0.0130 \times \texttt{age} + 0.0088 \times \texttt{income} \tag{15.1.2}$$

Using the provided information, we predict Václav will buy life insurance at:

$$\begin{aligned} \texttt{insurance} &= -0.4277 + 0.0130 \times \texttt{age} + 0.0088 \times \texttt{income} \\ &= -0.4277 + 0.0130 \times 65 + 0.0088 \times 125 \\ &= +1.5121 \end{aligned}$$

What does this value of 1.5121 actually mean?

 I don't know, either.

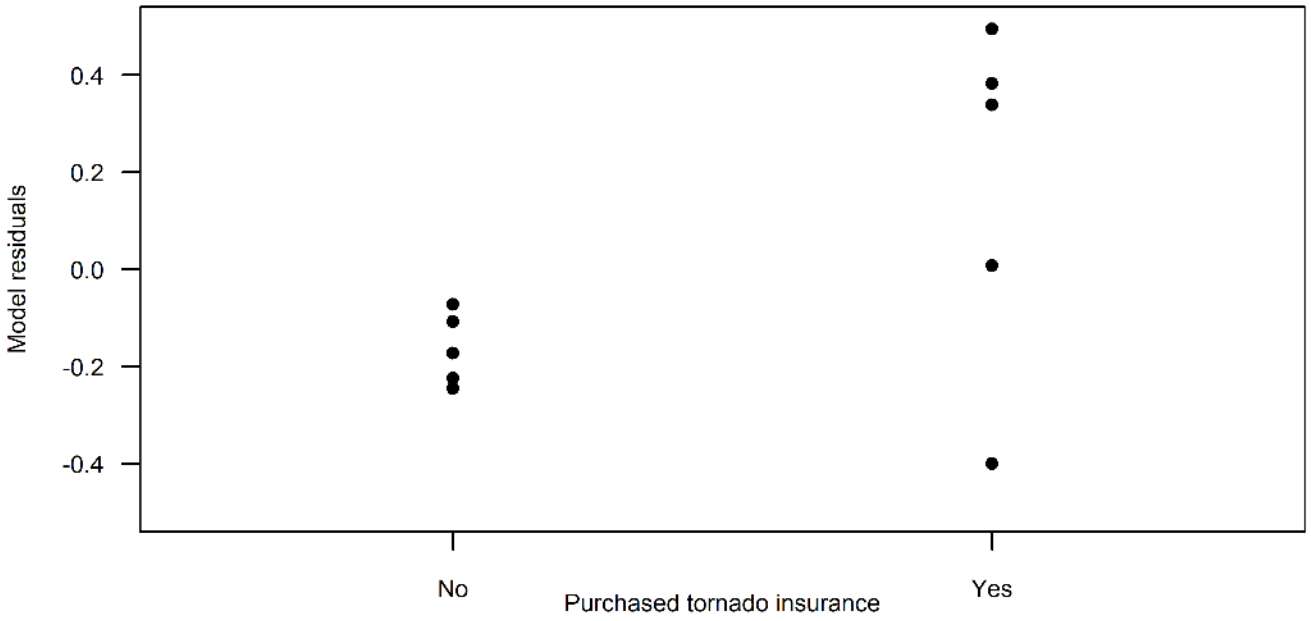


Figure 15.1.1: Scatter plot of the residuals against the values of the dependent variable. Note the different variances for the two groups. As such, the linear model is not appropriate in this case.

Next, to check the assumptions of OLS, let us merely check the assumption of homoskedasticity (constant variance). To do this, we plot the residuals against the values of the dependent variable. Figure 15.1.1 above shows that the variation in the residuals significantly differs across the two groups in this model — a violation of our assumptions. In fact, calculations show that the variance for those who bought insurance is about **24 times higher** than for those who did not (0.1325 vs. 0.0055). This is an example of non-constant variance. Performing the usual F-test for comparing two variances, we also see that this difference is statistically significant ($F = 0.0416, \nu_n = 4, \nu_d = 4, p = 0.0093$). Therefore, we conclude that our model is *not* appropriate for this data.

There were two problems with this analysis. **First**, the model predicted an outcome that did not make sense. Second, the model violated at least one assumption of ordinary least squares (it actually violates all three). To solve the first problem, we *could* create a decision rule that any predicted value above the threshold $\tau = 0.500$ will be treated as a "Buy" prediction, and any predicted value less than $\tau = 0.500$ will be treated as a "Not Buy" prediction.

The **second** problem is more serious and not so easily solved, especially if we care about our estimate's uncertainty (i.e., create confidence intervals). One may consider performing a transformation on the dependent variable to make it unbounded. A logit transformation would be a natural transformation for this; however, all of the dependent variables are either 1 or 0, which means the transformed values will be either $+\infty$ or $-\infty$. Furthermore, this transformation would not take care of the relationship between the residuals and the (transformed) dependent variables.

Be Not Afraid!

There is a tendency to feel disappointed when our model violates assumptions, such as here. However, *instead* of seeing the existence of a relationship between the residuals and the dependent variable as a problem, let us realize such a relationship tells us that there is *more information* in the data than we are modeling at this point.

As an interested researcher, we want to use that information to get more from our data. Thus, violations are not steps backwards; they are a path towards a deeper understanding of the data generating process.

This page titled 15.1: Binary Dependent Variables is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

15.2: Latent Variable Modeling

This section introduces the concept of a latent variable as a solution to the challenges of modeling binary outcomes. Rather than attempting to predict the observed 0/1 outcome directly, the focus shifts to modeling an unobserved (latent) continuous variable: the probability that the event occurs. This probability is naturally bounded between 0 and 1, making it suitable for transformation via the logit function into an unbounded linear predictor. The section establishes the foundational equation for logistic regression, where the logit of the probability is modeled as a linear function of the predictors, and shows how predictions are made by first calculating the linear predictor and then applying the inverse logit (logistic) transformation to obtain an estimated probability.

Learning Objectives

By the end of this section, you will be able to:

1. Explain the concept of a latent variable in the context of binary outcomes, recognizing that we model the underlying probability of success (a continuous variable bounded by 0 and 1) rather than the observed 0/1 outcome itself.
2. State the fundamental logistic regression model as: $\text{logit}(\pi) = \mathbf{XB}$, where π is the probability of success, and the logit link function transforms this bounded probability into an unbounded linear predictor.
3. Describe the two-step process for making predictions from a logistic regression model: first calculate the value of the linear predictor ($\pi = \mathbf{XB}$), then apply the inverse logit (logistic) transformation to obtain an estimated probability: $\pi = 1/(1 + \exp(-\eta))$.

✦•·····• ✦ •·····•✦

In the previous example, we discovered that Václav has a *something* of 1.5121 to buy insurance. What is the *something*? Our gut really wants us to say that it is the *probability* that he buys insurance. In fact, it would be *very* helpful if we could predict Václav's probability of buying life insurance. Unfortunately, what we estimated cannot be a probability, as the value is greater than 1.

Notice, however, that we have just made an unconscious step in our minds: We are no longer thinking in terms of modeling the actual outcome (1 or 0); we are once again thinking in terms of modeling the *expected value of the outcome*, $E[Y \mid x]$; here, that is the *probability* of a success, π.

In other words, we are now modeling a variable we cannot measure — a **latent variable**. Instead of modeling an actual outcome, we now think in terms of modeling the underlying probability that the person will purchase life insurance. This has the dual advantage of being a continuous variable *and* of being bounded by 0 and 1 — *exclusive*.

As such, we can model it using previous techniques. Remember that the predicted value will be a *probability*, not an actual outcome we can measure. To **predict the outcome**, there is an additional step: selecting a threshold value, τ, above which we predict the individual bought insurance; below which, not. The traditional threshold value is $\tau = 0.500$; however, there is **no reason** we cannot alter it *to better fit the data* (see the section on maximum accuracy).

Thus, our research model in the life insurance example becomes

$$\text{logit}\Big(P[\texttt{insurance}]\Big) = \beta_0 + \beta_1 \texttt{age} + \beta_2 \texttt{income}$$

We use the logit function for the same reason we used it before: to transform the bounded variable into an unbounded variable. The right hand side of the equation is η ("eta"), a linear function that can take on all real values — **the linear predictor**. Figure **15.2.1** below shows a schematic of what we are actually modeling. The diagonal line in the top figure is the line of best fit for the linear predictor. The horizontal line is the threshold value we chose to distinguish between "Success" predictions and "Failure"

predictions, which corresponds to $\text{logit}(\tau)$ in this top graph, τ in the bottom. The bottom figure is the linear predictor back-transformed into "probability" units. The horizontal line is the actual τ chosen, here $\tau = 0.500$.

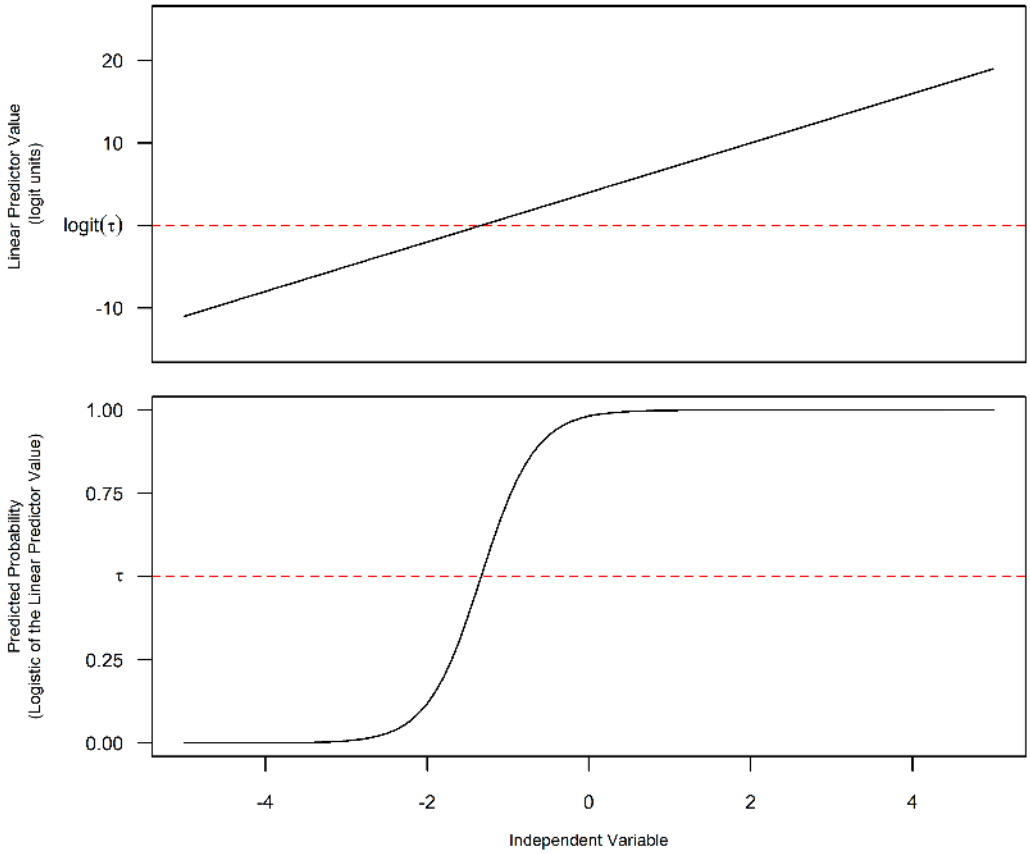


Figure 15.2.1: Plot of the linear predictor and a possible threshold for a typical latent binary dependent variable model. The logit of the Linear Predictor is in level units (proportion units).

If we need to actually calculate the probability that Václav will purchase life insurance, we can calculate it from the linear predictor:

$$\text{logit}\Big(\Pr[\text{insurance}]\Big) = \eta$$

This is equivalent to: $P[\text{insurance}] = \text{logistic}\big(\eta\big)$.

This section examined the relationship between the line of best fit for the linear predictor, η, and the predicted probability of a success. However, we did not discuss *how* that line of best fit was determined. The next section does just that.

This page titled 15.2: Latent Variable Modeling is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

15.3: The Mathematics

This section develops the full mathematical specification of a logistic regression model for a binary dependent variable within the Generalized Linear Model (GLM) framework. It formally defines each of the three required components: the linear predictor, the conditional distribution of the response, and the link function. The section demonstrates that the Bernoulli distribution is the natural choice for binary data, derives its key property that the expected value $E[Y] = \pi$ (the probability of success), and shows that its variance is inherently non-constant ($V[Y] = \pi(1-\pi)$), explaining the heteroskedasticity observed when such data are modeled with OLS. It then derives the logit function as the canonical link for the Bernoulli distribution by writing the probability mass function in the exponential family form. Finally, it introduces alternative link functions (probit, cauchit, log-log, complementary log-log) that can also map the probability π to the unbounded linear predictor, noting that the choice among them is often guided by disciplinary tradition and can affect predictions, though appropriate models should yield broadly similar conclusions.

Learning Objectives

By the end of this section, you will be able to:

1. Specify the three mathematical components of a logistic regression model for a binary outcome: the linear predictor $\eta = \beta_o + \beta_1 x_1 + \cdots + \beta_k x_k$, the Bernoulli distribution for the response (conditional distribution), and the logit link function $\text{logit}\pi = \eta$.
2. Derive the expected value and variance of a Bernoulli random variable, and explain why this variance structure inherently leads to heteroskedasticity when a binary outcome is modeled with ordinary least squares.
3. Demonstrate that the logit function is the canonical link for the Bernoulli distribution by rewriting its probability mass function in the standard exponential family form.
4. Identify several alternative link functions for binary data (e.g., probit, cauchit, log-log, complementary log-log) and recognize that while the choice of link may affect predictions, appropriate models should lead to broadly similar conclusions about the significance and direction of effects.

✦•·····• ✦ •·····•✦

When we model using the Classical Linear Model, we actually model/predict the *expected value* of the dependent variable, the mean. In the previous insurance example, we modeled/predicted the probability of a person purchasing life insurance. What is the connection? It is that $E[Y \mid x] = \pi$.

Remember from the chapter on GLMs, performing GLM estimation requires that we know **three things** about our data and our model:

1. the linear predictor,
2. the conditional distribution of the dependent variable, and
3. the function that links the two domains.

The previous section discussed the linear predictor ($\eta = \beta_0 + \beta_1 \text{age} + \beta_2 \text{income}$) and a link function, $\text{logit}(\mu)$, for our example. That only leaves the **conditional distribution** of the dependent variable.

Conditional Distribution

What are the possible values of the dependent variable? They are $\{0, 1\}$. What distribution has only these two outcomes? It is the Bernoulli distribution. For the Bernoulli distribution, the probability of getting a "1" (success) is π and the probability of getting a "0" (failure) is $1 - \pi$. Mathematically, this means the full probability mass function (pmf) is:

$$f(y) = \begin{cases} \pi^y (1-\pi)^{1-y} & y \in \{0,1\} \\ 0 & \text{otherwise} \end{cases} \tag{15.3.1}$$

Strictly speaking, the probability mass function is *not as important* as the expected value of this distribution. Why? Remember that the Generalized Linear Model paradigm models the *expected value*, $\mathbf{E}[Y \mid x]$, of the distribution of the dependent variable... just like in ordinary least squares.

Calculating the expected value of the Bernoulli distribution is easy using the definition of expected value:

$$E[Y] \stackrel{\text{def}}{=} \sum_i y_i \, f(y_i) \tag{15.3.2}$$

$$= 0 \cdot f(0) + 1 \cdot f(1) \tag{15.3.3}$$

$$= 0 \cdot (1-\pi) + 1 \cdot (\pi) \tag{15.3.4}$$

$$= \pi \tag{15.3.5}$$

Thus, the expected value of a Bernoulli random variable is π, the success probability.

This fact makes the results of modeling more apparent: As the GLM paradigm models the expected value, when we use the Bernoulli distribution, we end up modeling the probability of a success, which is what we want.

Note

Recall that one of the assumptions of Ordinary Least Squares is that the variance is constant with respect to the independent variable. When the outcomes are Bernoulli random variables, we can easily prove that the variance is not constant with respect to the expected values. If there is a relationship between the independent variable and the probabilities (which will be true if X affects Y), then a relationship between variance and expected value indicates heteroskedasticity.

To see this, let $Y \sim Bern(\pi)$. With this, and with the pmf above, we use the definition of variance to calculate $V[Y]$:

$$V[Y] \stackrel{\text{def}}{=} \sum_i (y_i - \mu)^2 f(y_i)$$

$$= (0-\mu)^2 f(0) + (1-\mu)^2 f(1)$$

$$= (0-\pi)^2 f(0) + (1-\pi)^2 f(1)$$

$$= \pi^2 (1-\pi) + (1-\pi)^2 \pi$$

$$= \pi(1-\pi)\left[\pi + (1-\pi)\right]$$

This last line simplifies to $V[Y] = \pi(1-\pi)$, as $\pi + (1-\pi) = 1$, which means $V[Y]$ is a function of π, the expected value. It is not a constant with respect to the expected value, π. Binary dependent variables violate the assumption of homoskedasticity — by definition.

As an aside: The variance is a quadratic function of the success probability, $V[Y] = \pi(1-\pi)$. From this formula, we see that we are most unsure (the variance is highest) when the probability of a success is $\pi = 0.500$. Check that this makes sense: Which has a more uncertain outcome, a fair coin ($\pi = 0.500$) or a two-headed coin ($\pi = 1.000$)?

Now that we understand our choice of distribution a bit better, and the resulting expected value, let us examine the third facet: the link function. First, note that π is bounded: $\pi \in (0,1)$. Thus we need a function that takes a doubly-bounded variable and transforms it into an unbounded variable. We have already met a link function that can handle this — the logit function.

Note

Here, I must mention that the logit is *not* the only appropriate link function. *Any* monotonic function that maps $(0,1) \mapsto \mathbb{R}$ is appropriate. This includes the entire class of quantile functions, of which the probit is a member.

The choice of the link function often reduces to *tradition within your field*. However, social science theory is getting advanced enough to suggest link functions that are more appropriate than others.

And so, we have the three necessary components to use generalized linear models in this example:

- the linear predictor,

$$\eta = \beta_0 + \beta_1 \text{age} + \beta_2 \text{income} \tag{15.3.6}$$

- the distribution of the dependent variable,

$$\text{insurance} \sim \text{Bernoulli}(\pi) \tag{15.3.7}$$

  with the formula for the expected value $\mu = \pi$.

- and the link function,

$$\text{logit}(\mu) = \eta \tag{15.3.8}$$

Here is what you need to take away from this section: The distribution must fit the possible outcomes. The link must translate the bounds on the parameter to the linear predictor. Both require you to know some distributions.

Deriving the Canonical Link

In the chapter on GLMs, we mentioned that each distribution has a canonical link. Let us derive the canonical link for the Bernoulli distribution. As a side note, one does not have to understand this section to use Generalized Linear Models.

The steps to determine the canonical link are the same for the Binomial as it was for the Gaussian:

1. Write the probability mass function (pmf).
2. Write the probability mass function in the required form.
3. Read off the canonical link.

For this distribution, this results in:

$$\text{pmf}: \quad \pi^y(1-\pi)^{1-y} \tag{15.3.9}$$

$$= \exp\left[\log\left(\pi^y(1-\pi)^{1-y}\right)\right] \tag{15.3.10}$$

$$= \exp\left[\log\left(\pi^y\right) + \log\left((1-\pi)^{1-y}\right)\right] \tag{15.3.11}$$

$$= \exp\left[y\ \log\left(\pi\right) + (1-y)\log\left(1-\pi\right)\right] \tag{15.3.12}$$

$$= \exp\left[y\ \log\left(\pi\right) + \log\left(1-\pi\right) - y\log\left(1-\pi\right)\right] \tag{15.3.13}$$

$$= \exp\left[y\ \left(\log(\pi) - \log(1-\pi)\right) + \log(1-\pi)\right] \tag{15.3.14}$$

$$= \exp\left[y\ \log\left(\frac{\pi}{1-\pi}\right) + \log(1-\pi)\right] \tag{15.3.15}$$

$$= \exp\left[y\ \text{logit}(\pi) + \log(1-\pi)\right] \tag{15.3.16}$$

$$= \exp\left[\frac{y\ \text{logit}(\pi) + \log(1-\pi)}{1} + 0\right] \tag{15.3.17}$$

This is in the required form: $\exp\left[\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\phi)\right]$.

Thus, reading off the standard form, we have the following:

➡️ $y = y$
$\theta = \text{logit}(\pi)$
$a(\phi) = 1$
$b(\theta) = \log(1-\pi) = -\log(1+e^{\theta})$
$c(y,\phi) = 0$

As such, the canonical link is the logit function, $g(\pi) = \text{logit}(\pi) = \eta$.

Other Links

As mentioned in the GLM chapter, we do not *have* to use the canonical link. *Any* monotonic, increasing function that maps the restricted domain to the unrestricted domain works. Thus, there are several options for the link function. The table above gives some options.[3]

The logit link is the canonical link. The probit link is frequently used in biostatistics. Its advantage is that it is based on the Normal distribution, with which we are intimately familiar. There is usually little difference between predictions made with the logit link and those made by the probit link. The coefficient estimates will usually differ by a factor of approximately 3.7, and the levels of significance will usually be close. The cauchit link is a symmetric link with heavy tails, as compared to the logit and the probit links (see the figure below).

The log-log link and the complementary-log-log link are asymmetric links. The log-log link has a heavy right tail; the complementary-log-log link, a heavy left tail (see the second figure below). Most science theory is only now beginning to be able to state which of the three types of link functions will be most appropriate for the given model (symmetric, heavy left, heavy right).

R has the built-in ability to model using the following link functions for the binomial (Bernoulli) distribution: logit, probit, cauchit, log, and complementary-log-log. The `KnoxStats` package adds the log-log link function:

```
glm( y ~ x, family=binomial(link=make.link("loglog")) )
```

Table 15.3.1: A list of several possible link functions (not all) to use for binary dependent variables. For the case of the Bernoulli distribution, remember that $\mu = \pi$.

Link	Inverse Link
Logit: $\log(\mu/(1-\mu))$	Logistic: $(1+\exp(-\eta))^{-1}$
Probit: $\Phi^{-1}(\mu)$	Normal CDF: $\Phi(\eta)$
Cauchit: $\tan\left(\pi(\mu-\frac{1}{2})\right)$	Cauchy CDF: $\arctan(\eta)/\pi+\frac{1}{2}$
log-log: $-\log(-\log(\mu))$	$\exp(-\exp(-\eta))$
Complementary log-log: $\log(-\log(1-\mu))$	$1-\exp(-\exp(\eta))$

Graphics of Other Links

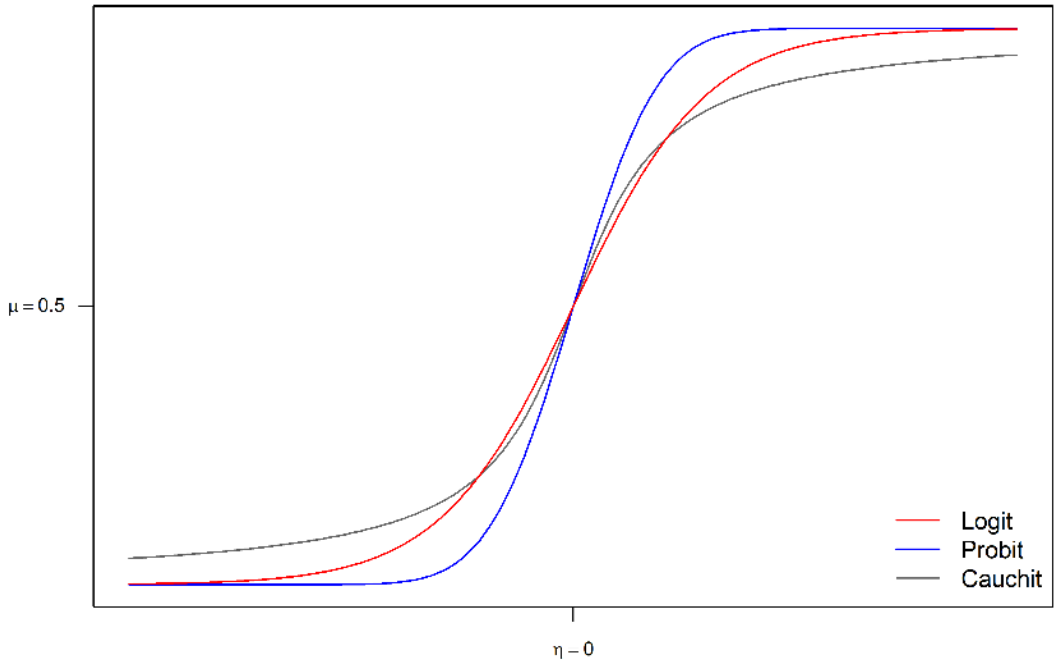


Figure 15.3.1: A graph of three **symmetric** links (logit, probit, and cauchit). Note that they all cross when the linear predictor $\eta = 0$ and that they cross at $\mu = 0.5$.

https://stats.libretexts.org/@go/page/57778

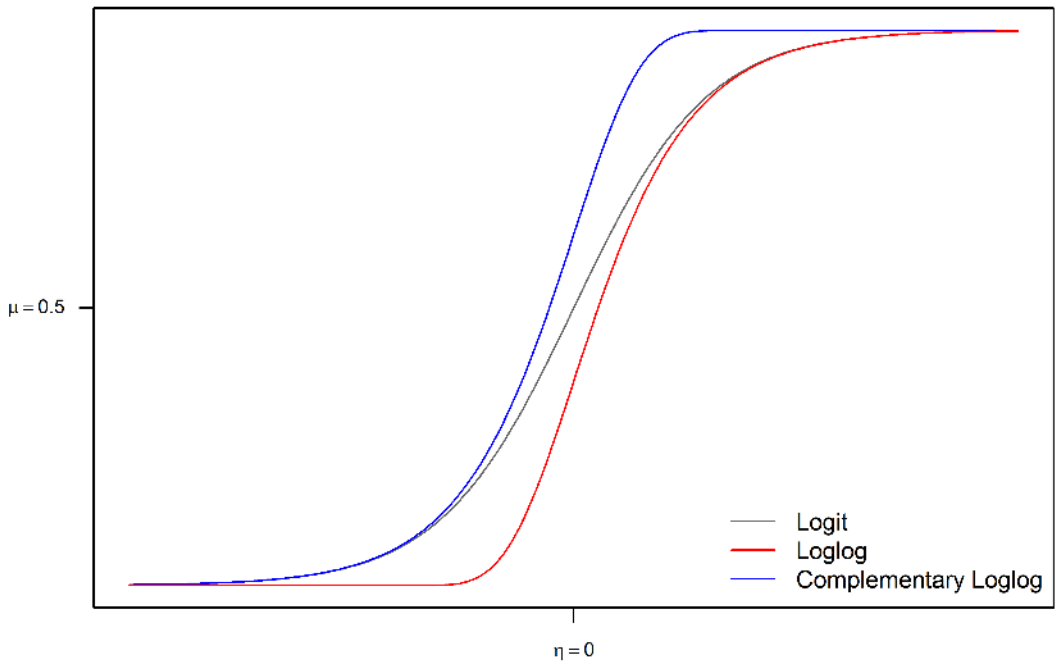


Figure **15.3.2**: A graph of two **asymmetric** links (complementary-log-log and log-log functions) with the symmetric logit link for comparison.

So, ... when to use an asymmetric link ... and which one???

Choosing between the two primary asymmetric links (and the symmetric links) is difficult. It depends on the asymmetry of the underlying probability process you are modeling. Both are used for binary or proportion data, but they handle asymmetry differently.

The complementary log-log link, $g(\pi) = \log(-\log(1-\pi))$, is appropriate when the probability of an event (a "success") is asymmetric, with the probability increasing slowly from zero but then approaching one sharply. This link is commonly used in areas like:

- Mortality studies: When death arises from the failure of any one of a number of independent organs or components.
- Modeling non-zero counts: When modeling the probability of a non-zero count in Poisson-distributed data.
- Constrained ordination: In ecology, it is sometimes preferred over the logit link for certain species distribution models.

The log-log link, $g(\pi) = -\log(-\log(\pi))$, is essentially the mirror image of the cloglog link. Fitting a model with a cloglog link to the probability of a failure $(1-\pi)$ is equivalent to fitting a log-log link to the probability of a success (π) . This means the log-log link is appropriate when the probability of success is asymmetric in the opposite direction: it increases sharply from zero but then approaches one slowly. It is sometimes used for parameters constrained to be greater than 1, though this application is more niche.

In summary, use a complementary log-log link when the increase in probability is slow at the beginning but accelerates near the end. Use a log-log link when the increase is rapid at the beginning but slows down near the end.

Caution

Again, the link choice is usually a matter of tradition, rarely of theory. Statistical significance of the variables should be similar across the several link functions. So, from a theory-testing standpoint, the link functions are rather interchangeable. With that said, *predictions* will vary depending on the link function chosen. Thus, if prediction is important then you will want to investigate the effect of different link functions on your predictions (and confidence bounds).

While the actual predictions will differ, they should only do so slightly. The rule is that all models that are "appropriate" should provide similar conclusions and predictions. If they do not, then your model is too fragile… a bad thing. Build for model **robustness**.

This page titled 15.3: The Mathematics is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

15.4: Modeling with the Logit

This section provides a practical guide to implementing logistic regression for a binary response variable using the canonical logit link function. It emphasizes the importance of the link function choice, noting that while the logit is the default for social sciences due to its symmetry and interpretability, researchers should consider testing multiple link functions to ensure their model predictions are robust. The section walks through a complete example using simulated coin flip data, demonstrating the key steps: specifying the model with the `glm` function in R (using the Binomial family and logit link), interpreting the resulting coefficient estimates (which are in log-odds units), and correctly back-transforming predictions to probabilities using the inverse logit (logistic) function. It reinforces that while the relationship is modeled as linear on the logit scale, the fitted probability curve is S-shaped on the original scale.

Learning Objectives

By the end of this section, you will be able to:

1. Fit a logistic regression model for a binary outcome in R using the `glm` function with the Binomial family and the logit link `family=binomial(link="logit")`, correctly specifying the model formula.
2. Interpret the coefficient estimates from a logistic regression model, understanding that they represent changes in the log-odds of the outcome for a one-unit increase in the predictor variable.
3. Generate predictions from a fitted logistic regression model and apply the inverse logit (logistic) transformation to convert predictions from the log-odds scale back to the original probability scale.
4. Explain that while the relationship is modeled as linear on the logit (log-odds) scale, the fitted probability curve is nonlinear (S-shaped) on the original scale, and recognize the importance of checking model robustness by potentially comparing results from different link functions.

✦•·····• ✦ •·····•✦

For a binary response variable, the canonical link function is the logit link. This link is characterized by being symmetric and having relatively thin tails (see the earlier figure on symmetric link functions). This symmetry may be important when you are dealing with events that are balanced — neither rare nor frequent. The tail thickness may be important when you think there is a sharp transition between success and failure in your data. In reality, current social science theory is rarely so clear as to give you guidance in which link function you should use. As such, try several and see which one gives the best fit.

> There is another reason to try several link functions. Since the "population" link function is not known, the predictions of the model should be robust to the choice of link function: Test several and see if the predictions are stable. If not, then the quality of your model depends heavily on something you cannot measure.

Of course, if there is a traditional link function used in your field, you should use it as the default. Thus, social scientists should start with the logit, while the health science researchers should start with the probit.

Example: 100 Coins

Since binary dependent variable regression is so very important to understand, let us look at it from a different direction:

Let us imagine an experiment where we have a series of 100 coins. Were these coins all fair, then the probability of getting a Head on any throw would be $\pi = \frac{1}{2}$. However, let us assume these coins are not necessarily fair, and that they are weighted in a very specific manner: Coin i has a probability of flipping a Head of π_i, which increases as i increases. That is,

$$\pi_{i+1} > \pi_i, \quad \forall i \tag{15.4.1}$$

Now, if we were allowed to flip each coin *only once*, how can we estimate π_1 from the data?

Solution.
As we have no evidence to the contrary, let us use the canonical link function, the logit. Our steps are quite similar to the steps we performed when we had to transform the dependent variable:

1. Read in the data
2. Model the dependent variable using the GLM paradigm (specify the distribution, the linear estimator, and the link function)
3. Predict outcomes using your model
4. Back-transform the predictions using the inverse of your chosen link function

Be Aware

There is a step missing from when we previously transformed our dependent variable: We do not have to transform the dependent variable. Generalized linear modeling does that for us in R. We do, however, have to *back-transform* the predictions. Be aware of this!

In R, the general form of the command is, showing the most important parameters,

```
glm(formula, family(link), data)
```

Only `formula` is required. If `family` is missing, the Gaussian (or Normal) distribution will be assumed. If `link` is missing, the canonical link for that family will be assumed. If `data` is missing, the current data frame will be assumed.

For binary response variables, the family will need to be the Binomial distribution. Thus, for the example using the `coinflips` data file, the command will be

```
m1 = glm(head~trial,
         family=binomial(link="logit"),
         data=coin)
```

I used the `data` parameter, as I did not attach the data earlier. If you attached, then you do not need to include this parameter. I also included `link="logit"` even though this is the default setting for the Binomial family in order to remind myself of the link function I used in this analysis.

```
                 Estimate     Std. Err     z-value     p-value
Constant Term     -2.2929       0.5384       -4.26   << 0.0001
Trial Number       0.0345       0.0087        3.97      0.0001
```

The results from this command are summarized in the table above. Again, note that the parameter estimates (and predictions) will be in "logit units." You will have to use the logistic function (the inverse of the logit) to get the predictions in units of probability. Recall that the interpretation here would be

"For every increase of 1 in the coin number, the *odds* of a head increases by about 3.45 percent."

Recall that the original question asked us to determine π_1, the probability of getting a Head on the first coin. There are a couple ways of doing that. The best will depend on the numbers involved. Since we want π_1, we know it is equal to the logistic of the intercept plus *one* times the coefficient:

$$\pi_x = \text{logistic}\big(-2.2929 + 0.0345\,x\big) \tag{15.4.2}$$

$$\pi_1 = \text{logistic}\big(-2.2929 + 0.0345\,(1)\big) \tag{15.4.3}$$

$$= 0.0946 \tag{15.4.4}$$

The other way is to use the `predict` function and take the logistic of that value. You will get the same answer (within rounding error). The function call used is

```
predict(m1, newdata=data.frame(trial=1))
```

This gives an answer of -2.2584. The logistic of -2.2584 is our estimate of π_1, which is $\pi_1 = 0.0946$.

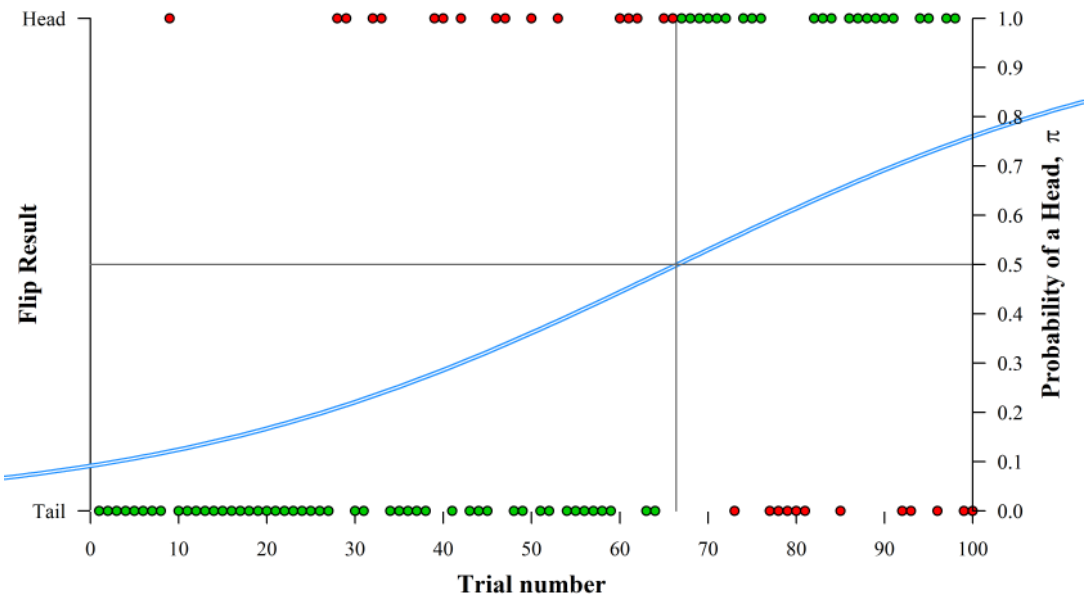


Figure 15.4.1: Overlayed plot of the outcome of the experiment with the estimated probabilities superimposed. The horizontal line is the $\tau = 0.500$ threshold. The vertical line corresponds to a trial number corresponding to that threshold ($\tau = 66.4$). Thus, this model predicts that all coins above number 66.4 have a probability of greater than half of coming up Heads. Red dots are misclassified, green dots are properly classified.

If we so desire, we can also plot the probability curve on a graph of the outcomes (see Figure **15.4.1** above). With such a graph, we could estimate which coin is most fair. With the graph, we could also get a feel for how well the model represents the data.

Note

The linear predictor is represented in the curve graphed in the figure above. Note, however, that the curve is *not* linear. This is because the curve is actually the logistic of the linear predictor.

With that said, the curve *is* linear in the transform-space. If you graph the coin number against the logit of the head probability, the line of best fit is, indeed, a line (see the earlier schematic figure on latent variable modeling).

This page titled 15.4: Modeling with the Logit is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

15.5: Measuring Accuracy

This section addresses the essential question of how to evaluate the quality of a logistic regression model. Moving beyond the pseudo-R^2 measures introduced for GLMs, it introduces a suite of practical metrics for assessing how well a binary classification model predicts actual outcomes. The section begins with overall accuracy and relative accuracy, then explores the crucial trade-off between different types of correct and incorrect predictions through the concepts of sensitivity (true positive rate) and specificity (true negative rate). It introduces the Receiver Operating Characteristic (ROC) curve as a tool for visualizing this trade-off across all possible decision thresholds and explains the Area Under the Curve (AUC) as a summary measure of model discriminative ability. Finally, it covers the F1 score, which balances precision and recall, and discusses how to incorporate the differential costs of false positive and false negative errors when selecting an optimal prediction threshold.

Learning Objectives

By the end of this section, you will be able to:

1. Calculate and interpret overall accuracy and relative accuracy for a logistic regression model, understanding that relative accuracy compares the model's performance to a naive baseline (e.g., always predicting the majority class).
2. Construct a confusion matrix from predicted and actual binary outcomes and use it to calculate sensitivity (true positive rate) and specificity (true negative rate), explaining the inherent trade-off between these two measures as the prediction threshold changes.
3. Generate and interpret a Receiver Operating Characteristic (ROC) curve, and explain that the Area Under the Curve (AUC) represents the probability that the model will rank a randomly chosen positive instance higher than a randomly chosen negative one.
4. Calculate and interpret the F1 score as the harmonic mean of precision and recall (sensitivity), recognizing its value when class distributions are imbalanced.
5. Describe how differential error costs can be used to select an optimal threshold that minimizes the total expected cost.

✦•·····• ✦ •·····•✦

Naturally, the next questions concern issues of goodness-of-fit: How good is the model? This question can be answered in many ways using many related accuracy measures.

Recall that in linear regression, we used R^2 to help us determine how well the model fit the data — an R^2 value close to 1.00 indicated good fit, while an R^2 value close to 0.00 indicated a poor fit. If we recall, the R^2 value — a PRE measure — was calculated using a ratio of the original variability in the data and the variability explained by the model. The R^2 value was not the only PRE we have covered. Many others exist. Similar processes can be used in *this* context to create a pseudo-R^2 measure.

Note

This measure is a pseudo-R^2 measure primarily because it shares some of the characteristics of the true R^2 measure, namely that it measures the decrease in prediction variability due to the model. It is a PRE measure.

The reason it is not called the R^2 measure is only because that name is taken elsewhere.

Accuracy Rate

Let us define the accuracy rate to be the number of correct predictions divided by the total number of predictions. This makes inherent sense as a measure of goodness of fit since it reads as the proportion of correct predictions.

There is no native accuracy function in R (this should raise a red flag 🚩). However, the `KnoxStats` package provides one. The `accuracy` function takes four parameters: `data` (the data variable), `y` (the binary dependent variable), `model` (the model you fit with the data), and `t` (the threshold). The optional parameter, `rate`, tells the function to return the accuracy rate (default) or the number of accurate predictions (`rate=FALSE`).

Thus, to determine the accuracy of this model for this data using the usual threshold value of $\tau = 0.500$, we would use

```
accuracy(data=coin, y=coin$head, model=m1, t=0.500)
```

The result of this command is 0.710, which agrees with our by-hand calculations. Thus, we conclude that this model correctly predicts 71% of the time for *this* data.

Relative Accuracy

Of course, having an accuracy rate of 0.710 does not tell us the entire story. Just as the R^2 was based on a ratio of the model variance to the data (null) variance, a better accuracy number would be the accuracy of the model relative to the accuracy of the null model. The accuracy of the null model refers to merely selecting the modal category as our prediction. In this example, the modal category is Tails, as there were 61 Tails in the data. Thus, the accuracy of merely selecting the modal category is $61 \div 100 = 0.610$. So, the *relative* accuracy is

$$A_R = \frac{0.710}{0.610} = 1.164 \tag{15.5.1}$$

Thus, the model does a 16.4% better job of prediction than does just predicting "Tail" all of the time.

There actually is a proportional reduction in error (PRE) measurement associated with the relative accuracy. Recall that the R^2 value was valuable because it measured the proportion of error explained by the model. For binary dependent variable regression, we can calculate something similar.

$$\text{PRE} = 1 - \frac{\text{error with model}}{\text{error without model}} \tag{15.5.2}$$

Here, we can see that a pseudo-R^2 measure for this data and this model (and this threshold) is

$$1 - \frac{1 - 0.710}{1 - 0.610} \approx 0.2564 \tag{15.5.3}$$

Thus, we can state that this model (and this threshold) reduced the error by 25.64%. Note that there is a bad quality of this measure: while it can never be greater than 1.0, it *can* be less than zero. However, it will only be less than zero when *your* model is worse than no model at all.

Note

There are *many* different ways of calculating pseudo-R^2 measures. Each of the measures are based on different definitions of "error" or of "variability," just as the R^2 and the adjusted R^2 are both based on different definitions of variability. Researchers do not agree on much about pseudo-R^2 measures except that they are not useful *in vacuo*, and rarely useful in concert with other measures.

This is why I am offering it here, alongside many other measures of fit. Getting to know your model results is just as important as getting to know your data.

Maximum Accuracy

In each of the above measures, we assumed our threshold was $\tau = 0.500$. In some cases, this is a logical threshold. In some cases, it is chosen arbitrarily. If we treat τ as a parameter, we may be able to get a better prediction model.

The plan is straight forward: Calculate the accuracy for various values of the threshold. The threshold that gives us the best accuracy will be our optimal threshold. Doing this by hand is prohibitive. Using a script to loop through all threshold values is much easier:

```
a = numeric()
for(i in 1:100) {
  t = i/100
  a[i] = accuracy(coin, coin$head, m1, t=t)
}
```

Graphing these results gives the following figure.

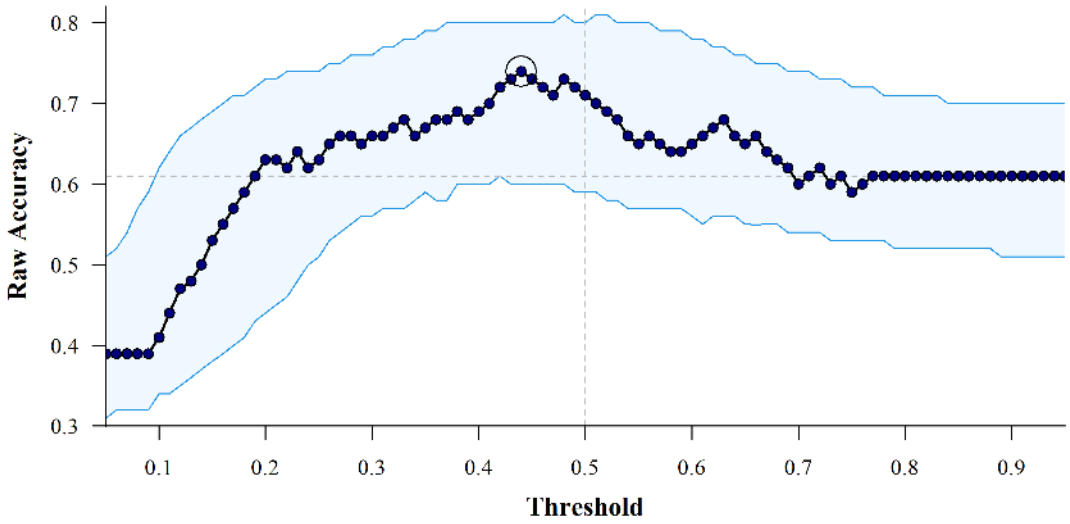


Figure 15.5.1: A plot of the accuracy of the model against various thresholds. The horizontal line corresponds to the accuracy of selecting the modal category (the base accuracy). The vertical line corresponds to the threshold $\tau = 0.50$. The circled point represents the maximal threshold, $\tau = 0.44$ and accuracy $= 0.74$. The light blue envelope consists of a 95% confidence interval for coin accuracy, based on Monte Carlo simulation.

Figure 15.5.1 above is a plot of the calculated accuracy for various thresholds. Note that the "optimal" threshold is *not* $\tau = 0.50$, but $\tau = 0.44$, and the maximal accuracy is 0.74 for that threshold. With that being said, however, that there is little difference in accuracies between this optimal threshold ($\tau = 0.44, A = 0.73$) and the traditional threshold ($\tau = 0.50, A = 0.71$).

Note

Recall that the standard deviation for (variability of) a binomial random variable is $\sigma_x = \sqrt{n\pi(1-\pi)}$. This takes on a maximum value at $\pi = 0.500$… the success probability for a fair coin. This means that we are *least sure* of our answer nearest $\pi = 0.500$.

The blue envelope of the figure above contains 95% of the calculated accuracies based on the true population; that is, 95% of the accuracy curves are contained in that envelope. It is very wide. It supports the contention that accuracy (relative or otherwise) matters little in the estimation of an optimal threshold τ.

> By the way, since this is all based on generated data, we know the true threshold for a fair coin: $\tau = 0.500$. Binomial random variables contain **small amounts of information**.

Finally, here is the script to estimate the confidence bounds above using Monte Carlo simulation:

```
B = 1e4
acc = matrix(NA, ncol=100, nrow=B)

for( j in 1:B ) {
  thisSample = sample(100, replace=TRUE)
  mod = glm(head[thisSample]~trial[thisSample], family=binomial)
  a = numeric()

  for(tau in 1:100) {
    a[tau] = accuracy( trial, head, mod, t=tau/100 )
  }
  acc[j, ] = a
}
```

From reading through this script, you should be able to tell what the variables `head` and `trial` represent. You should also be able to explain the purpose of each line. In fact, you should be able to use this code as the basis of **future accuracy investigations**.

The ROC Curve

There are other types of errors, more-specific types, that are definitely **more useful and generalizable**. If we look back to the coin logistic figure, we see that the threshold line (horizontal) and the corresponding trial line (vertical) divide the dataset into four parts. The lower-left quadrant are those Tails that are correctly predicted by the model and the threshold value to be Tails. The upper-right quadrant are those Heads that are correctly predicted to be Heads. The lower-right quadrant are Tails incorrectly predicted to be Heads. The upper-left quadrant are Heads incorrectly predicted to be Tails. These four types of errors are also referred to, respectively, as

- True Negatives
- True Positives
- False Positives
- False Negatives

For our coin flipping example (and with $\tau = 0.500$), we can write out a confusion matrix to show all four of these, both in magnitude and in rates:

$$\begin{bmatrix} TP = 22 & FP = 12 \\ FN = 17 & TN = 49 \end{bmatrix} \Leftrightarrow \begin{bmatrix} TPR = \frac{22}{17+22} = 0.5641 & FPR = \frac{12}{49+12} = 0.1967 \\ FNR = \frac{17}{17+22} = 0.4359 & TNR = \frac{49}{49+12} = 0.8033 \end{bmatrix}$$

The true negative rate (TNR) is also called *specificity*, and the true positive rate (TPR) is called the *sensitivity*. You will come across these two terms in the field of biostatistics and clinical trials because they mirror what physicians and biomedical researchers want out of their diagnostic tests.

The **receiver operating characteristic** (ROC) curve is a graphical representation of the **true positive rate** against the **false positive rate** (sensitivity against $1 - \text{specificity}$) as the threshold is changed. Thus, to plot a ROC curve, one would calculate the true positive and the false positive rates for various values of the threshold, then plot the first against the second. The figure below shows the ROC curve for our coin model.

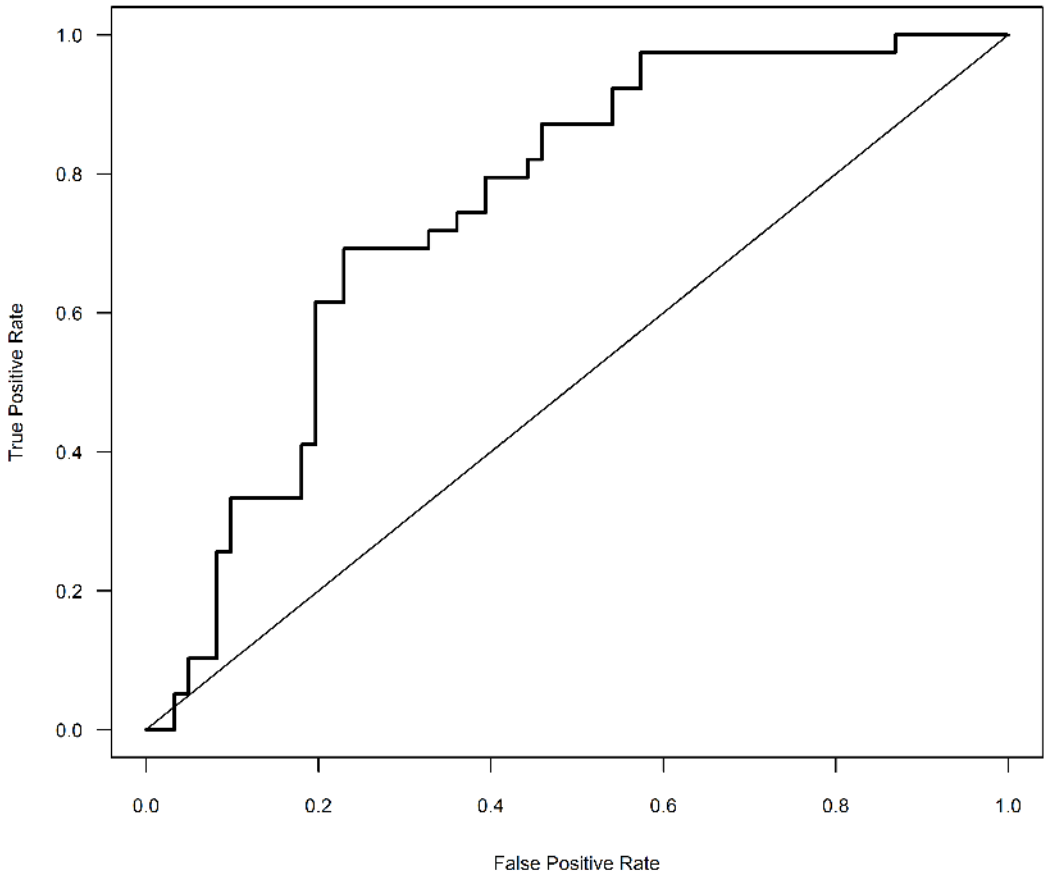


Figure 15.5.2: A receiver operating characteristic curve for the coin flipping model. The diagonal line represents a random model. The thicker line represents our model. The farther the ROC curve is above the random line, the better the model is at distinguishing between the two cases (Head and Tail, here). The area under the ROC curve is a measure of the goodness of the model. Here, $A' = 0.7516$.

In general, a model whose ROC curve is closer to the left and upper axes is the better model. As such, we can define a single number that tells us how good our model is — the **area under the ROC curve** (AUC, A').

The area under the ROC curve is a useful number in that it equals the **probability that a model will classify a positive instance higher than a negative one**. In other words, A' is the probability that the model scores a true Head (success) higher than a true Tail (failure). Calculating the area is very straight forward, in a geometry/Riemann Sum manner.

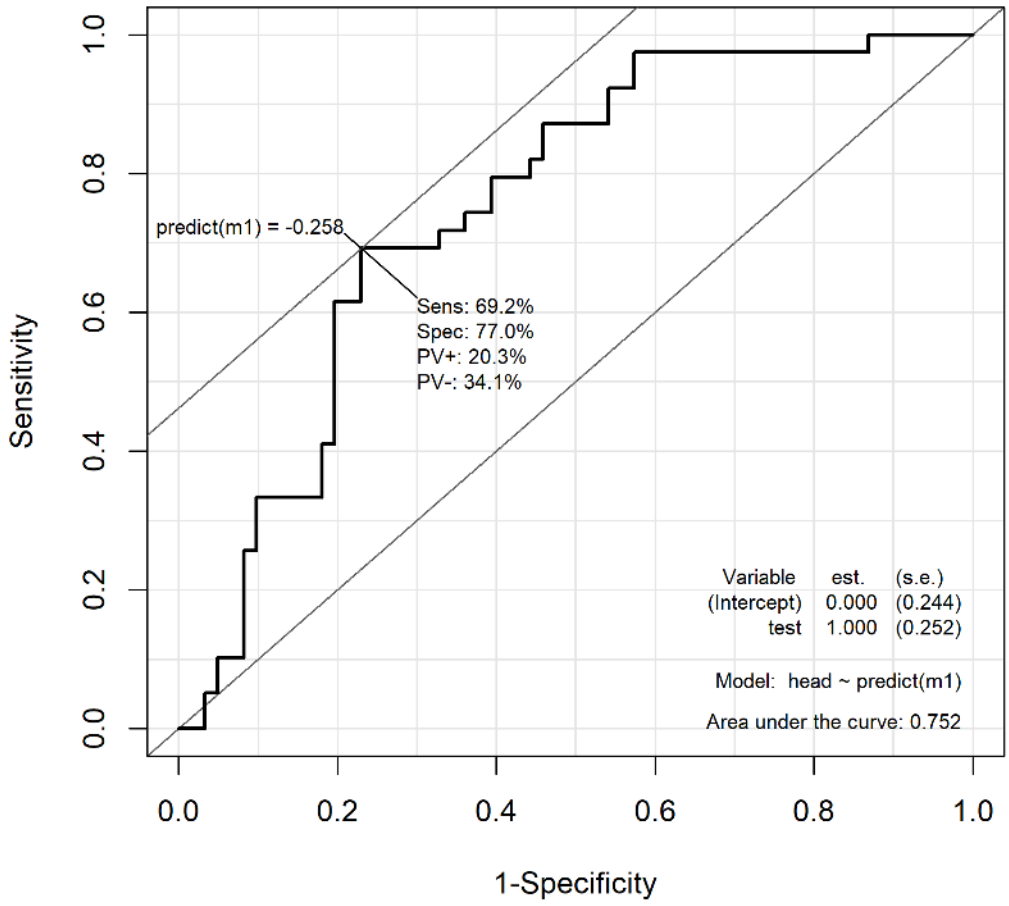


Figure **15.5.3**: The receiver operating characteristic curve for the coin flipping model using the `ROC` command from the **Epi** package.

R Feature

There is an entire R package dedicated to ROC curves, Epi. To create ROC graphs and to calculate the area under the curve in that package, first load it using `library(Epi)`, then use the command

```
ROC(test, stat, plot="ROC")
```

Here, test is the predicted probability of success for each datum from model (a continuous variable bounded by 0 and 1), stat is the binary dependent variable, and `plot="ROC"` produces a ROC plot. This graph (see the figure above) is a bit more useful than the simple graph in the previous figure, as it contains some useful statistics, including the AUC and the optimal threshold, τ, which is the threshold value closest to the upper-left corner.

Sensitivity and Specificity

When a binary classifier, such as a logistic regression model with a chosen probability cutoff, makes predictions, its performance is judged by two fundamental and often competing properties: sensitivity and specificity. These metrics move us beyond simple accuracy to understand precisely how a model is succeeding or failing.

Sensitivity, also called the True Positive Rate or Recall, answers the question: Of all the actual positive cases, what proportion did the model correctly identify? It is calculated as True Positives divided by the sum of True Positives and False Negatives. A test or model with 95% sensitivity is excellent at catching positives; it misses only 5% of them. This is the critical measure when the cost of missing a positive is high, such as failing to diagnose a serious disease or failing to detect a security threat. In contrast, Specificity, the True Negative Rate, answers: Of all the actual negative cases, what proportion did the model correctly rule out? It is calculated as True Negatives divided by the sum of True Negatives and False Positives. A highly specific model, say 98%, is very reliable when it gives a positive prediction because it rarely falsely alarms on negatives.

Crucially, sensitivity and specificity exist in a trade-off, governed by the decision threshold of the classifier. Setting a very low threshold for declaring a positive will catch nearly all true positives, resulting in high sensitivity, but will also falsely label many negatives as positives, resulting in low specificity. Conversely, a very high threshold will correctly rule out nearly all negatives, achieving high specificity, but will miss many true positives, yielding low sensitivity. This trade-off is formally visualized by the Receiver Operating Characteristic, or ROC, curve, which plots the sensitivity against one minus specificity across all possible thresholds. The choice of the optimal operating point on this curve is not a statistical decision, but a practical and ethical one, determined by the relative costs of false positives and false negatives in the specific application domain.

Sensitivity, or the True Positive Rate (TPR), is defined as:

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{15.5.4}$$

where TP represents True Positives and FN represents False Negatives.

Specificity, or the True Negative Rate (TNR), is defined as:

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{15.5.5}$$

where TN represents True Negatives and FP represents False Positives.

These are implemented in KnoxStats as such:

```
sensitivity(confMatrix)

specificity(confMatrix)
```

F and F1 Scores

In many classification scenarios, a single metric that balances the competing priorities of precision and recall is desirable. The F-score, specifically the F1 score, provides this balance. It is the **harmonic mean** of precision and recall (sensitivity). Precision answers the question: "Of all instances the model labeled as positive, what proportion were actually positive?" It is calculated as $\text{Precision} = TP/(TP + FP)$. Recall, as previously defined, is $\text{Recall} = TP/(TP + FN)$. The harmonic mean, unlike a simple arithmetic mean, is particularly stringent; it yields a low score if *either* precision or recall is poor.

The general formula for the F-score is:

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{(\beta^2 \cdot \text{Precision}) + \text{Recall}} \tag{15.5.6}$$

The β parameter allows the user to **weight the importance** of recall relative to precision. A $\beta > 1$ weights recall more heavily, while a $\beta < 1$ weights precision more heavily.

The most common variant is the **F1 score**, where $\beta = 1$, giving equal weight to precision and recall. Its formula simplifies to:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{15.5.7}$$

The F1 score is particularly useful when you have an unbalanced class distribution and a simple measure of accuracy would be misleading. For example, in fraud detection where 99% of transactions are legitimate, a model that simply predicts "not fraud" for every transaction would be 99% accurate but useless. The F1 score, by focusing on the correct identification of the rare positive class (fraud), provides a much more meaningful assessment of the model's practical utility in such contexts.

The F and F1 scores are implemented in KnoxStats as such:

```
Fscore(confMatrix, beta=3)

F1score(confMatrix)
```

For all of the accuracy measures that take the confusion matrix as the input, make sure that your confusion matrices are of the right format:

```
                  Reality
                  FALSE  TRUE
Prediction  FALSE    53    26
            TRUE      8    13
```

This can easily be done using R's built-in `table` function, as such:

```
confMatrix = table(Prediction, Reality)
```

Error Costs

Note that this optimal value is only optimal if the costs of making each type of error is **the same**. If the cost of a Type I Error is greater than that of a Type II Error (or vice-versa), then one should take those costs into consideration when determining the "optimal" threshold τ.

Not all errors hurt the same.

For instance, if one is modeling fraudulent credit card transactions, then a false positive would happen if the model flagged the transaction as being fraudulent (but it isn't). A false negative would occur if the model did not flag a transaction as fraudulent (but it is). A high false positive rate would inconvenience the credit card holder. It would reduce their ability to use it. A high false negative rate would inconvenience the credit card bank by forcing them to pay for fraudulent uses of the card.

Not all errors hurt the same (or even the same people).

For instance, given the following table, which of the two models should be used?

Table **15.5.1**: Example of different costs requiring us to select a different model

Error Type	Error Cost	Model 1	Model 2
FPR	$50	0.10	0.15
FNR	$10	0.15	0.10

The first model costs $6.50; the second, $8.50. Thus, we should select the first model (or threshold).

The Page's Code

Here is the code for the analysis on this page:

```
library(KnoxStats)
library(Epi)

coin = read.csv("http://rfs.kvasaheim.com/data/coinflips.csv")
attach(coin)
summary(coin)


m1 = glm(head~trial, family=binomial(link="logit"), data=coin)
summary(m1)


# Accuracy
accuracy(data=coin, y=coin$head, model=m1, t=0.500)

# Relative Accuracy
accuracy(data=coin, y=coin$head, model=m1, t=0.500)/0.610

# Maximum Accuracy
a = numeric()
for(i in 1:100) {
  t = i/100
  a[i] = accuracy(coin, coin$head, m1, t=t)
}
max(a)
which.max(a)


# MC Confidence Bounds
B = 1e3
acc = matrix(NA, ncol=100, nrow=B)
for( j in 1:B ) {
  thisSample = sample(100, replace=TRUE)
```

```
  thisData = coin[thisSample,]
  mod = glm(head~trial, family=binomial(link="logit"), data=thisData)
  a2 = numeric()

  for(tau in 1:100) {
    a2[tau] = accuracy( data=thisData, y=thisData$head, model=mod, t=tau/100 )
  }
  acc[j, ] = a2
}


# The Bounds
lcl=ucl=numeric()
for(i in 1:100) {
  lcl[i] =quantile(acc[,i],0.025)
  ucl[i] =quantile(acc[,i],0.975)
}


# The Graphic

par(xaxs="i", yaxs="i")
par(family="serif", las=1)
par(cex.lab=1.2, font.lab=2)
par(mar=c(3,3,0,0)+0.75)

plot.new()
plot.window( xlim=c(0.05,0.95), ylim=c(0.3,0.82))

title(xlab="Threshold", line=2.5)
title(ylab="Raw Accuracy", line=2.75)

polygon(x=c(1:100/100,rev(1:100/100)),y=c(ucl,rev(lcl)) , col="AliceBlue", border=4 )

abline(v=0.50, lty=2, col="grey")
abline(h=0.61, lty=2, col="grey")

lines(1:100/100, a, lwd=2)
points(1:100/100, a, pch=21, bg="navy")

axis(1, at=0:10/10)
axis(2, at=0:10/10)

points(which.max(a)/100, max(a), cex=3, lwd=1)
```

```
# ROC Curve
ROC(test=predict(m1), stat=head, data=coin)



# More Accuracy measures
confMatrix = table(predict(m1)>which.max(a)/100, head)

sensitivity(confMatrix)
specificity(confMatrix)

Fscore(confMatrix, beta=3)
F1score(confMatrix)
```

This page titled 15.5: Measuring Accuracy is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

15.6: Modeling with Other Links

This section emphasizes the importance of exploring alternative link functions when modeling binary data, rather than defaulting to the canonical logit link without consideration. Here, I argue that while the logit link is sufficient for basic model fitting, a deeper understanding of the data-generating process can be gained by comparing results across different links. The section introduces two *asymmetric* link functions — the complementary log-log (a.k.a. cloglog) and the log-log — and explains how their distinct shapes (one with a steep approach to its maximum, the other from its minimum) can capture different types of underlying probability processes.

Learning Objectives

By the end of this section, you will be able to:

1. Explain the rationale for using multiple link functions (beyond the canonical logit) when modeling binary data, noting that consistent conclusions across different links strengthen confidence in the model, while discrepancies may indicate model misspecification.
2. Fit logistic regression models in R using asymmetric link functions, specifically the complementary log-log link (`family=binomial(link="cloglog")`) and the log-log link (`family=binomial(link=make.link("loglog"))`).
3. Recognize that coefficient estimates from models with different link functions are not directly comparable because they are on different scales, and that model comparisons should instead focus on the stability of predictions and overall conclusions about the direction and significance of effects.

✦•·····• ✦ •·····•✦

The logistic regression (or "logit" or "logistic" regression) we did above is quite sufficient if all you want to do is fit the data using logistic regression. If, on the other hand, you want to better understand the process that gave you the data, you will want to try different link functions to determine if any of the alternative links do an appreciably better job of fitting your data. The logit link is symmetric. You should also use the probit link as a check on your model: If the results are comparable, then the conclusions are strengthened; if not, there is something wrong with your model.

In addition to using a second symmetric link function, you should use the two main asymmetric link functions: the complementary log-log and the log-log link function.

The Complementary Log-Log Link (cloglog)

As mentioned earlier, there are several other available links functions beyond the logit link (see the earlier table of link functions). Actually, for binary response variables, all that is required of the link function is for it to be increasing, to smoothly map $g\colon (0,1) \mapsto \mathbb{R}$, and to have an inverse that smoothly maps $g^{-1}\colon \mathbb{R} \mapsto (0,1)$. As mentioned earlier, the logit link is symmetric. If you are dealing with rare-events data, you may not want to use a symmetric link function. The complementary log-log link is asymmetric and is often useful (see Figure **15.6.1** below).

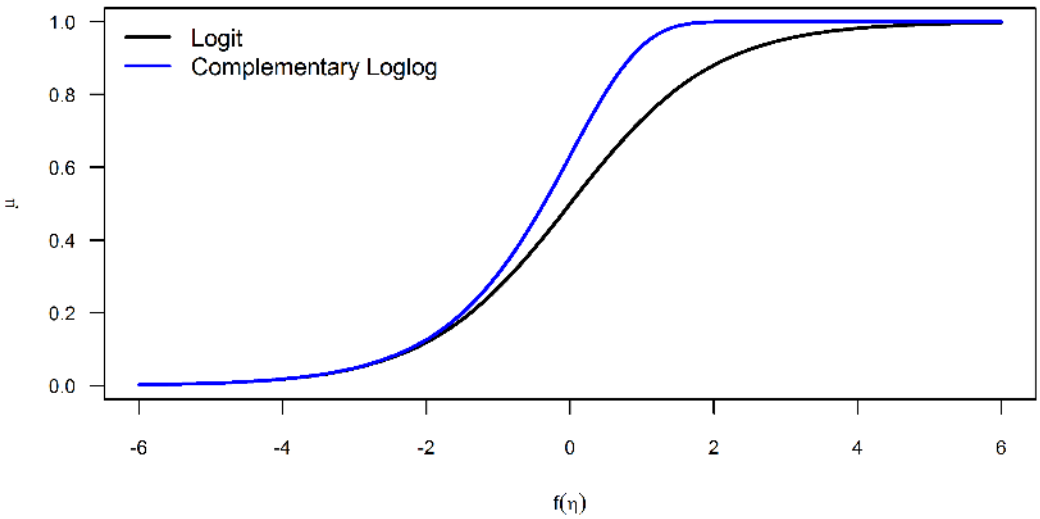


Figure 15.6.1: Plot of the complementary log-log function (upper curve) on top of the logit. Note the difference in shapes between the two curves. The asymmetric complementary log-log function approaches its maximum value much faster than does the symmetric logit.

The formula for the complementary log-log is

$$g(\pi) \stackrel{\text{def}}{=} \log\Big(-\log(1-\pi)\Big) \tag{15.6.1}$$

Its inverse is

$$g^{-1}(\eta) = 1 - \exp\Big(-\exp(\eta)\Big) \tag{15.6.2}$$

The plot of the complementary log-log function is seen in the figure above, overlaid with the same plot for the logit link. Note the difference in shapes. Recall that the logit link is symmetric. The complementary log-log is not; it approaches its maximum value more steeply than the logit.

Because of this asymmetry, it will fit models differently. Let us fit the coin data with a complementary log-log link. The command is

```
glm(head~trial, family=binomial(link="cloglog"), data=coin)
```

Note that the only change is in the link clause. The results of this new model are provided in the table below. Note that the direction of effect is the same in both models. Unfortunately, as the first model is in logit units and the second model is in complementary log-log units, comparing the magnitude of the coefficients tells us nothing. Comparing predictions tells us much more.

Using the logit model, the prediction for π_1 was 0.095. Using the complementary log-log model, the prediction is $\pi_1 = 0.122$, which is closer to the true value of $\pi_1 = 0.150$.

```
                 Estimate     Std. Err     z value     Pr(> |z|)
Constant term     -2.0651       0.4353       -4.74     << 0.0001
Trial number       0.0244       0.0063        3.86        0.0001
```

The Log-Log Link

A second useful asymmetrical link function is the log-log link (see Figure 15.6.2 below). Note that the asymmetric log-log link rises to its maximum much slower than either the symmetric logit link or the asymmetric complementary log-log link. Because of this functional shape, it will be better at fitting certain data sets better than the other link functions discussed.

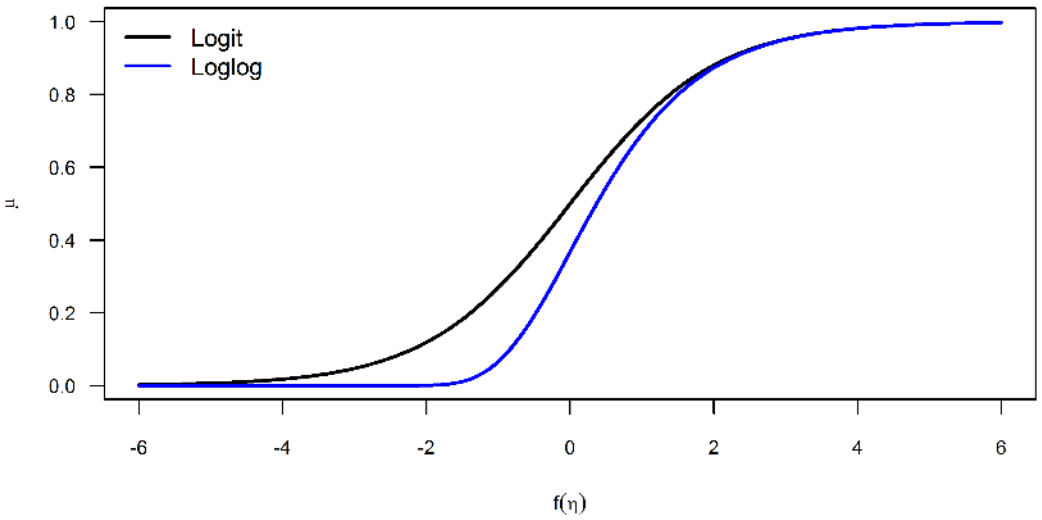


Figure 15.6.2: Plot of the log-log function (upper curve) on top of the logit. Note the difference in shapes between the two curves. The asymmetric log-log function approaches its maximum value much slower than does the symmetric logit.

In reality, there is a functional relationship between the complementary log-log and the log-log link functions. They are 180° rotations of each other. Thus, statistical programs either have no support for either or have support only one. Like most statistics packages, R has native support for only one of the two. For R, it is the complementary log-log link.

> This is actually a decision of history. From how I (and most) have presented the binary dependent variable models, it seems as though we statisticians started with the logit. The first use of this type of regression, however, used the complementary log-log function (Fisher, 1922). It was not pretty, but it was a fantastic step in the right direction!

The command to perform the log-log regression on this data is the same as before, except for the link parameter, which is now

```
link=make.link("loglog")
```

With this, I leave it as an exercise for you to show that the effect of trials in the log-log model is 0.0233 and that the predicted probability of a head for Coin 1 using this model is 0.0498.

So, which link should I use?

Now that you know about these five link functions, which should you use? An answer other than "All of them!" requires subject matter knowledge that scientists rarely have. However, here are some comments on the five links. Take them with a grain of salt.

The logit link is the canonical link function for the Bernoulli distribution. It is symmetric and has relatively thin tails. Because it is the canonical link, its mathematical properties are attractive, and its coefficients can be interpreted in terms of changes in the log-odds of the outcome. The logit link is the default choice in many fields, particularly the social sciences, and is most appropriate when the outcomes are balanced (neither extremely rare nor extremely frequent).

The probit link is also symmetric and is based on the inverse of the standard normal cumulative distribution function (a.k.a. the quantile function). Because the Normal distribution is so familiar, the probit link is frequently used in biostatistics and fields with deep roots in the Normal distribution. In practice, predictions made with the probit link are usually very close to those made with the logit link. The coefficient estimates will typically differ by a constant factor of approximately 1.67, and the levels of statistical significance will generally be similar.

The cauchit link is a symmetric link based on the Cauchy distribution, which has *much* heavier tails than the logistic or Normal distributions. This means the cauchit link should be considered when the underlying probability process may involve extreme values or when the transition from failure to success is *very abrupt*. Because of its heavy tails, the cauchit link can produce model fits that are noticeably different from the others, and it can be a useful diagnostic tool for checking the **robustness of your conclusions**.

The complementary log-log link is an asymmetric link. It is the inverse of the Gumbel distribution. This link function is characterized by rising very slowly from zero for low values of the linear predictor, but then approaching one very sharply. It is most appropriate when the probability of a success is asymmetric in this direction, meaning that the event of interest is initially *very unlikely* but, once conditions become favorable, its probability increases *rapidly*.

The log-log link is also asymmetric and is essentially the mirror image of the complementary log-log link. Fitting a model with a cloglog link to the probability of a failure $(1-\pi)$ is equivalent to fitting a log-log link to the probability of a success (π). This means the log-log link is appropriate when the probability of success is asymmetric in the opposite direction: it increases very sharply from zero at first, but then approaches one very slowly as the linear predictor increases.

This page titled 15.6: Modeling with Other Links is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

15.7: Model Selection

This section addresses the practical challenge of model selection when working with binary dependent variables and Generalized Linear Models. Unlike in ordinary least squares, where an adjusted $\mathbf{R^2}$ provides a straightforward penalty for adding variables, no single universally accepted measure exists for models like logistic regression. The section introduces several tools designed to balance model fit against model complexity. It begins with a pseudo-$\mathbf{R^2}$ measure based on deviance, then moves to information criteria (AIC and BIC) which explicitly penalize the inclusion of additional parameters. Finally, it presents the Likelihood Ratio Test as a formal statistical method for comparing nested models, allowing researchers to determine whether a set of predictors is jointly significant.

Learning Objectives

By the end of this section, you will be able to:

1. Calculate and interpret a pseudo-$\mathbf{R^2}$ for a GLM using the formula (Null Deviance – Residual Deviance) / Null Deviance, understanding it as an analog to the $\mathbf{R^2}$ in OLS.
2. Apply the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) to compare non-nested models fitted to the same data, selecting the model with the lower criterion value as the preferred model.
3. Conduct a Likelihood Ratio Test to compare two nested models, calculating the test statistic as the difference in their deviances and evaluating it against a chi-squared distribution with degrees of freedom equal to the difference in the number of parameters.
4. Recognize the critical requirement that all model comparison methods based on likelihood (pseudo-$\mathbf{R^2}$, AIC, BIC, Likelihood Ratio Test) are only valid when the models are fit to exactly the same set of observations and the same dependent variable.

✦•·····• ✦ •·····•✦

Which of the models is best? That is a model selection question. Model selection *procedures* (not tests) attempt to balance competing desires — accuracy and parsimony — to create the "best" model, by some standard. For linear models, we discussed the R^2 value as a measure of accuracy. However, we noted that adding variables to the model can never decrease the R^2 value, and will usually increase it. Thus, there is a pressure to increase the number of variables. However, science is guided by the philosophy of William of Occam and his Razor:

Numquam ponenda est pluralitas sine necessitate.

Plurality must never be posited without necessity. That is, models should be as simple as possible, but no simpler. In other words, as scientists, we should only include variables if the theory warrants it.

Caution

Make no mistake, all models are wrong. As scientists, we are merely searching for the useful ones.

In linear regression, we corrected for the pressure to keep adding variables by using the adjusted R^2 as a guide. This value penalizes the model for the number of variables it has. Thus, unless the variable is statistically significant, there is no benefit to adding it to the model. This is why many scientists use the adjusted R^2 measure to help them model select.

There is neither a true R^2 nor a true $\bar{R}^2$ value for discrete dependent variable models. Thus, there has been much work in creating an appropriate measure to use for **model selection**. Three different measures are frequently used in the literature: Akaike's Information Criterion (Akaike 1974), Bayesian Information Criterion (Schwarz 1978), and Likelihood Ratio Test (Wilks 1938). Each of these three penalizes additional variables in a different manner and to a different degree. The one you select depends on the one available to you and the relationship between the two models.

Pseudo-R^2

For those who are "addicted" to the R^2 measure, there is a pseudo-R^2 measure that may satisfy the craving. Remember that the R^2 measure is a proportional reduction in error (PRE) measure. Thus, if we can find some measure of variation or error from the output, we will be able to create a PRE for models of dichotomous variables.

Here it is:

$$\text{pseudo-}R^2 = 1 - \frac{\text{residual deviance}}{\text{null deviance}} \tag{15.7.1}$$

Both of these deviances are given in the usual GLM output. The "residual deviance" is the deviance that remains after applying the model. The "null deviance" is the deviance of the null model, the model that has no independent variables.

It is nice that the interpretation of the pseudo-R^2 follows that of the R^2 value from ordinary least squares.

AIC: Akaike Information Criterion

One of the first attempts to explicitly penalize for additional parameters (variables) was done by Hirotugu Akaike (1974). In his paper, he developed (albeit without much mathematical rigor) a comparative measure of "model goodness" that can be used to select the better of two models based on the log-likelihood measure. The Akaike Information Criterion (AIC) score can be calculated whenever Maximum Likelihood Estimation is used to estimate the model parameters. The formula for the AIC is

$$\text{AIC} \stackrel{\text{def}}{=} -2\ln(\mathcal{L}) + 2k \tag{15.7.2}$$

Here, k is the number of parameters being estimated in the model and $\mathcal{L}$ is the likelihood of the data with the model.

> The quantity $-2\ln(\mathcal{L})$ is often called the **deviance** of the model, which will be used in the section on the Likelihood Ratio Test.

The procedure to determine if one model is better than the other is straight-forward:

1. Calculate the AIC for Model A.
2. Calculate the AIC for Model B.
3. The model with the ***lower*** AIC score is the preferred model.

Its simplicity is its strength. Its weakness is that this measure, called the minimum information theoretical criterion (MAICE) in the paper, has no known probability distribution. As such, there is no way to determine whether the model with the lower AIC is *enough* better to justify eliminating the other from the discussion: If the AIC of Model 1 is 3 less than the AIC of Model 2, do we completely ignore Model 2?

This question actually leads to several "rules of thumb" that determine when that difference is "large enough." The usual rules of thumb are to drop the model with the higher AIC if the difference is at least 5 (or 8 or 10).

That there is no *a priori* statistical distribution to the AIC score only means the test is not optimal. In his paper, Akaike concurs (1974: 722):

Although the present author has no proof of optimality of MAICE it is at present the only procedure applicable to every situation where the likelihood can be properly defined and it is actually producing very reasonable results without very much amount of help of subjective judgment.

The R function that calculates the Akaike Information Criterion is `AIC`. Using this function, the AIC for each of the three coin models are $\text{AIC}_{\text{logit}} = 118.48$, $\text{AIC}_{\text{cloglog}} = 119.74$, and $\text{AIC}_{\text{loglog}} = 117.05$. Thus, while the log-log is the "best" model from the AIC standpoint, it is not sufficiently better to completely ignore the other two models (the AIC improvement is not greater than 5). As such, this procedure is inconclusive with respect to the single model we should choose.

Be Aware!

Please keep in mind that for the AIC to be valid in comparing models, the dependent variable values must be the *exactly* same across the models. If not, then this process cannot be used (nor any of these methods). Thus, transformations of the dependent variable mean the AIC cannot be used. Similarly, if data points are removed between two models, the AIC **cannot be used**.

This warning holds for **all** measures based on the likelihood. So, the BIC and the Likelihood Ratio test cannot be used, either.

Keeping multiple appropriate models is a good idea. Since sufficient science theory does not exist to determine the "right" link, we should keep as many as possible. This will allow us to better understand how much our conclusions depend on our choice of the link function.

BIC: Bayesian Information Criterion

Akaike's paper did not give a mathematically solid reason why there should be a 2 point penalty for each additional estimated parameter (the $2k$ factor). This created an opening for other researchers to improve upon Akaike's proof and to create different penalty factors. Schwarz (1978) took Akaike's idea and put it on a more solid foundation. He humbly called his measure the Bayesian Information Criterion (BIC), others may refer to it as the Schwartz Information Criterion (SIC) or the Schwarz Bayesian Criterion (SBC).

Its formula is quite similar to the AIC:

$$\text{BIC} \stackrel{\text{def}}{=} -2\ln(\mathcal{L}) + k\log(n) \tag{15.7.3}$$

Here, k is the number of parameters being estimated, n is the number of data points, and $\mathcal{L}$ is the likelihood of the model. Thus, the difference between the AIC and the BIC is the effect of the additional parameter. In the AIC, each additional parameter penalizes the score by 2 points; in the BIC, $\log(n)$ points — usually a much greater penalty.

The process to select the better of two models is the same as for the AIC: Select the model with the lower BIC score (including the rules of thumb). Furthermore, the requirement that the dependent variable values are the same between the models remains.

Likelihood Ratio Test

Frequently, we wish to determine if a group of variables are **jointly significant** in the model. To do this, we compare the two nested models. We say that Model B is *nested* in Model A if Model A contains all the same variables as does Model B, plus at least one other. For instance, let Model A contain the variables `X1`, `X2`, `X3`, $\texttt{X3}^2$, and `X4`. Let Model B contain variables `X1`, `X2`, and `X3`. Here, Model B is nested within Model A. Now, if we want to determine if variables $\texttt{X3}^2$ and `X4` are jointly significant, then we merely compare Models A and B. To do this, we can use the AIC or the BIC, but the Likelihood Ratio test is more statistically clean.

The Likelihood Ratio test is superior to the AIC and BIC — when it can be used — because there is a known asymptotic probability distribution for the test statistic. As such, we can determine whether Model A is *significantly* better than Model B — i.e., whether variables `X4` and `X5` are jointly significant.

The procedure is also straight forward:

1. Calculate the deviance for Model A.
2. Calculate the deviance for Model B (the nested model).
3. The difference between the two deviances is distributed asymptotically as a chi-squared random variable with degrees of freedom equal to the parameter (variable) difference in the two models.

The deviance of a model is defined as

$$D \stackrel{\text{def}}{=} -2\ln(\mathcal{L}) \tag{15.7.4}$$

Thus, if Model B is nested in Model A, the test statistic is equal to

$$\text{TS} \stackrel{\text{def}}{=} D_B - D_A \sim \chi^2_{v_A - v_B} \tag{15.7.5}$$

Here, v_A is the number of parameters in Model A; v_B, in Model B.

Example 15.7.1

Let us assume that Model A uses three variables, $X1$, $X2$, and $X3$, and has a log-likelihood of -20, and Model B uses one variable, $X1$, and has a log-likelihood of -22. Are variables $X2$ and $X3$ jointly significant?

Solution.
This is a direct application of the Likelihood Ratio test. The test statistic is

$$\begin{aligned} \text{TS} &\stackrel{\text{def}}{=} D_B - D_A && (15.7.6)\\ &= \big(-2\ln(\mathcal{L}_B)\big) - \big(-2\ln(\mathcal{L}_A)\big) && (15.7.7)\\ &= \big(-2(-22)\big) - \big(-2(-20)\big) && (15.7.8)\\ &= 44 - 40 = 4 && (15.7.9) \end{aligned}$$

This test statistic is approximately distributed as a chi-squared random variable with $3 - 1 = 2$ degrees of freedom; that is, $\text{TS} \stackrel{a}{\sim} \chi^2_2$.

A chi-squared table gives us a p-value of approximately $p = 0.15$. This is close to what R gives us:

```
pchisq(4, df=2, lower.tail=FALSE) = 0.135
```

Thus, we conclude at the $\alpha = 0.05$ level that we cannot reject the null hypothesis and we conclude that the restricted model is not significantly different from the full model.

That is, we conclude that the two variables are **not jointly significant** and we can use Model B in lieu of Model A with little loss of precision.

This page titled 15.7: Model Selection is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

15.8: Conclusion

This chapter covered **a lot of material**. First, we examined how to fit binary dependent variable models. The GLM paradigm allows us to easily fit such models. As in all uses of the GLM paradigm, we need to know three things: the conditional distribution of the dependent variable, the linear predictor, and the link function that connects the two.

For binary dependent variables, the dependent variable is distributed Bernoulli (Binomial, with n=1). The linear predictor is the usual combination of our independent variables. The canonical link is the logit link. Additional link functions include the probit, log-log, and complementary log-log functions.

The chapter proceeded to examine issues of determining how well a model fits the data. Accuracy, relative accuracy, and maximum accuracy measures were examined. Additionally, we examined the ROC curve and how it gives us additional information about our model. Believe it or not, this part of the book will be useful for the rest of your data-analyzing life! Know what the various measures of accuracy mean, so you can select the one that best matches your problem

Finally, we examined general techniques to select between two models. Three methods were examined. The first two did not require that the two models be nested. Both the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) only required that the dependent variables be the same. Also in both cases, the model with the lower score was the preferred model, although when that difference was less than 8 there was no reason to jettison the higher-scoring model.

The Likelihood Ratio Test was superior to the two Information Criterion tests as the test statistic has a known asymptotic distribution (χ^2). Thus, we could test the statistical significance of multiple variables at once. The drawback to using the Likelihood Ratio test is that the compared models needed to be nested.

The next chapter continues our examination of discrete dependent variables. Frequently, our outcome variable is a *count* of events. In such a case, we cannot use the techniques discussed in this chapter as the dependent variable takes on more than just two values. We also cannot apply the techniques of Chapter 14: Generalized Linear Models, as the dependent variable is not continuous.

Staying in the realm of GLMs allows us to fit such variables easily. All we need to do is determine the appropriate distribution, the linear predictor, and the link function.

Caution

The section on accuracy measures is, perhaps, the most important section in the book because it applies to all models you will ever fit.

This page titled 15.8: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

15.9: End-of-Chapter Material

R Functions

In this chapter, we were introduced to several R functions that will be useful in the future. These are listed here.

Packages

- `KnoxStats`
  This package adds much general functionality to R, especially in terms of accuracy measures.
- `Epi`
  

  This package adds several functions and procedures related to epidemiology. As it is not a part of the base installation for R, you will need to install it before you can load it with `library(Epi)`. It is a support package for the book, *Epidemiology with R*.

Statistics

- `lm(formula)`
  This function performs linear regression on the data, with the supplied formula. As there is much information contained in this function, you will want to save the results in a variable.
- `glm(formula)`
  This function performs generalized linear model estimation on the given formula. There are three additional parameters that can (and often should) be specified.
  - The `family` parameter specifies the distributional family of the dependent variable, options include `gaussian`, `binomial`, `poisson`, `gamma`, `quasibinomial`, and `quasipoisson`. If this parameter is not specified, R assumes `gaussian`. For this chapter, we relied on the `binomial` family.
  - The `link` parameter specifies the link function for the distribution. If none is specified, the canonical link is assumed. The canonical link for the Binomial distribution is the logit function.
  - Finally, the `data` parameter specifies the data from which the formula variables come. This is the same parameter as in the `lm` function.
- `predict(model, newdata)`
  As with almost all statistical packages, R has a predict function. It takes two parameters, the model, and a dataframe of the independent values from which you want to predict. If you omit `newdata`, then it will predict based on the independent variables of the data itself, which can be used to calculate residuals. The dataframe must list all independent variables with their associate new values. You can specify multiple new values for a single independent variable.
- `AIC(model)`
  This function calculates the Akaike Informations Criterion score for the provided model. The model needs to have been fit using Maximum Likelihood Estimation.
- `BIC(model)`
  This function calculates Schwarz's Bayesian Information Criterion (BIC) for the provided model. The model needs to have been fit using Maximum Likelihood Estimation.
- `deviance(model)`
  This function returns the deviance in the model. This value is useful in the Likelihood Ratio Test. The model needs to have been fit using Maximum Likelihood Estimation.

- `pchisq(x)`
  This gives the value of the cumulative distribution function (CDF) under the Chi-squared distribution. The necessary parameter is the number of degrees of freedom, `df=`. By default, it returns the lower-tail probability. Usually, we will want to have the upper-tail probability, thus we will use the `lower.tail=FALSE` parameter.

- `var.test(x,y)`
  This function performs an F test, which compares the variances of two samples (x and y) from Normal populations. It can only compare two samples. If you need to compare more than two samples for equality of variance, you will need to perform either a Bartlett test or a Fligner-Killeen test.

Accuracy

- `accuracy(model)`
  This function determines the predictive accuracy of a provided model. It takes three necessary parameters: `data`, `truth`, `model`, and `threshold`. It has the optional parameter of returning the *number* of correct classifications (`rate=FALSE`). This actually should not be used, unless your sample is representative of the population in terms of positive and negative realities... which is **never** true if you try to balance your data.

- Here is a list of other accuracy measures you may want to use. Make sure you know what each is measuring and what it means to use that particular measure of accuracy
  - `Fscore` : F-score
  - `F1score` : F1 score (balanced F-score)
  - `MCC` : Matthews correlation coefficient (the phi coefficient)
  - `phiCoef` : Phi coefficient (mean square contingency coefficient)
  - `precision` : Precision rate (positive predictive value)
  - `recall` : Recall rate (sensitivity)
  - `fnr` : False Negative rate (miss rate)
  - `fpr` : False Positive rate (Type I error rate)

Graphics

- `ROC(formula)`
  This function in the `Epi` package performs ROC analysis on the data. It provides a ROC graph as well as some statistical values. This function also provides the `AUC` measure of accuracy.

Programming

- `for`
  This command is one of the basic control-constructs in the R language (as in most programming languages). The usual use is `for(var in seq) expr`, where `var` is the looping variable (the variable that equals the current loop number). The parameter `seq` is a vector of values. Usually, `seq` is something like `1:100`, which is a vector of values from 1 to 100. Finally, `expr` is the expression (or series of expressions) that are performed for each value in the `seq` vector.

Exercises

This section offers suggestions on things you can practice from this chapter.

1. In Example ???, we suggested that you fit the provided pseudo data with linear regression and the OLS method. Please do so now.

2. From Section ???, please fit the \var{coin} data with the formula \var{head$\dist$trial} and the log-log link. What is the predicted probability of getting a Head on Coin 15?

3. Use the coinflip data (coinflips.csv) to estimate the coin that is closest to being fair (a probability of producing a head is closest to 0.500). Use multiple link functions and select which you think is the best.

4. Let us revisit the \var{cows} data. One of the variables is \var{passed}, which is a binary variable indicating whether the ballot measure passed. Your job is to predict the proportion of voters in D\v{e}\v{c}\'{i}n who will vote in favor of the bill to limit cows. Do not use the \var{pctFavor} variable. Decide which model you are supposed to use. Prove that your model is the best model available. Make your prediction of the vote share. Include graphs if you would like, but only if the graph helps to illustrate your point.

Applied Readings

- Judi Bartfeld and Myoung Kim (2010). "Participation in the School Breakfast Program: New Evidence from the ECLS-K." *Social Service Review*. 84(**4**): 541–62.
- Regina P. Branton (2009). "The Importance of Race and Ethnicity in Congressional Primary Elections." *Political Research Quarterly*. 62(**3**): 459–73.
- Denise Gammonley, Ning Jackie Zhang, Kathryn Frahm, and Seung Chun Paek. (2009) "Social Service Staffing in U.S. Nursing Homes." *Social Service Review*. 83(**4**): 633–50.
- Michael A. Neblo (2009). "Meaning and Measurement: Reorienting the Race Politics Debate." *Political Research Quarterly*. 62(**3**): 474–84.
- Lenna Nepomnyaschy and Irwin Garfinkel (2011). "Fathers' Involvement with Their Nonresident Children and Material Hardship." *Social Service Review*. 85(**1**): 3–38.
- Joseph G. Pickard, Megumi Inoue, Letha A. Chadiha, and Sharon Johnson (2011). "The Relationship of Social Support to African American Caregivers' Help-Seeking for Emotional Problems." *Social Service Review*. 85(**2**): 247–66.
- Brian Kelleher Richter, Krislert Samphantharak, and Jeffrey F. Timmons (2009). "Lobbying and Taxes." *American Journal of Political Science*. 53(**4**): 893–909.
- Lori E. Ross, Rachel Epstein, Corrie Goldfinger, and Christina Yager (2009). "Policy and Practice regarding Adoption by Sexual and Gender Minority People in Ontario." *Canadian Public Policy / Analyse de Politiques*. 35(**4**): 451–67.

Theory Readings

- Hirotugu Akaike (1974). "A New Look at Statistical Identification Model." *IEEE Transactions on Automatic Control*. 19(**6**): 716–723.
- Hirotugu Akaike (1977). "On Entropy Maximization Principle." In P. R. Krishnaiah (Editor). *Applications of Statistics: Proceedings of the Symposium Held at Wright State University, Dayton, Ohio, 14-18 June 1976*. New York: North Holland Publishing, 27–41.
- George Casella and Roger L. Berger (2002). *Statistical Inference*. Second edition. New York: Duxbury.
- Peter McCullagh and John A. Nelder (1989). *Generalized Linear Models*. London: Chapman and Hall.
- Gideon E. Schwarz (1978). "Estimating the dimension of a model." *Annals of Statistics*. 6(**2**): 461–64.

- Samuel S. Wilks (1938) "The Large-Sample Distribution of the Likelihood Ratio for Testing Composite Hypotheses." *The Annals of Mathematical Statistics*. 9(**1**): 60–62.

This page titled 15.9: End-of-Chapter Material is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

16: Binomial Dependent Variables



In the previous chapter, we examined what we can do if the dependent variable is dichotomous, has only two possible outcomes. The response variable followed a Bernoulli distribution. In this chapter, we extend this idea to where the outcome variable follows a Binomial distribution --- conditional on the values of the independent variable(s).

In this chapter, we first define a Binomial random variable, we then show that the Binomial distribution is a member of the Exponential Class of distributions (EC). With that, we can look at the assumptions and a couple extended examples.

In the previous chapter, we explored a situation where the classical linear model utterly failed. That failure has been known since the early parts of the 20th century. As a result, statisticians created several specialized fitting techniques for binary dependent variables. The importance of the generalized linear model is not that it can fit models with a binary dependent variable, but that it is a technique that can fit those models *and* many others. This allows the researcher to pay closer attention to the dependent variable and use more information.

In *this* chapter, we cover the case where the dependent variable represents a count of successes out of a known number of attempts, n.

While least squares can be used in such cases, even without clearly violating its assumptions, it is *inefficient*. You will be throwing out important information, namely the distribution of the dependent variable. Statisticians hate to throw out information.

This chapter starts with exploring the most common type of variable representing a count of successes out of a known number of attempts. After working with the Binomial, it looks into what happens when the data show the wrong level of variability... and what to do about it.

Chapter Sections

16.1: Binomial Distribution

16.2: The Mathematics

16.3: O Canada!

16.4: Sri Lanka in 2010
16.5: Beta-Binomial Regression
16.6: Conclusion
16.7: End-of-Chapter Materials

This page titled 16: Binomial Dependent Variables is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

16.1: Binomial Distribution

This section introduces the binomial distribution as the foundational probability model for dependent variables that represent counts of successes out of a fixed number of trials. It begins by defining the binomial distribution as the sum of independent and identically distributed Bernoulli random variables and lists the five specific requirements that must be met for a variable to be properly modeled as Binomial. The section then emphasizes the critical importance of the sample space, particularly the fact that a binomial count has both a natural lower bound of zero and a natural upper bound equal to the number of trials. Through a series of examples, it illustrates which types of count variables are appropriately modeled with the binomial distribution and which are not, drawing a key distinction between bounded counts (Binomial) and unbounded counts over time or space (Poisson).

Learning Objectives

By the end of this section, you will be able to:

1. State the five requirements for a random variable to follow a Binomial distribution: a known number of trials n, two possible outcomes per trial (success/failure), constant success probability π, independent trials, and the variable counts the number of successes.
2. Identify the sample space of a Binomial random variable as the set of integers from 0 to n, recognizing that a Binomial count is naturally bounded both below (by 0) and above (by n).
3. Distinguish between variables that can plausibly follow a Binomial distribution (e.g., number of students passing a class, number of invalid ballots in a division) and those that cannot (e.g., number of crimes in a year, number of errors in a book), based on the presence or absence of a fixed upper bound.
4. Explain that if a variable representing a count can be meaningfully expressed as a proportion (with a known denominator), it suggests a Binomial distribution, whereas if it is more naturally expressed as a rate (with no fixed upper bound), a Poisson distribution may be more appropriate.

✦•·····• ✦ •·····•✦

The Binomial distribution is defined as the sum of independent and identically distributed Bernoulli random variables. Subsequently, this results in five requirements for a random variable to follow a Binomial distribution:

1. the number of trials, n, is known;
2. each trial has two possible outcomes: success and failure;
3. the success probability for each trial, π, is constant;
4. each trial is independent from the others; and
5. the random variable is the number of successes in those n trials.

One can also think of the Binomial distribution as a generalization of the Bernoulli distribution to $n > 1$. Similarly, one can think of the Bernoulli distribution as a special case of the Binomial, where $n = 1$. Whichever way you look at it, there are a number of similarities between the two distributions.

The sample space for the Binomial distribution is key to understanding when it can be used, $\mathcal{S} = \{0, 1, 2, \ldots, n\}$. Thus, the Binomial distribution can be used to model counts of successes when the number of attempts (trials) is known.

The following are variables that *could* follow a Binomial distribution:

- number of students passing a class
- number of Euchre games a person wins in a tournament finals
- number of college students in a class who can locate Ruritania on a map

- number of football games the SUR Lions win in a year
- number of pages in a book that have a typographical error
- number of fireworks in a shipment of 144 that are duds
- number of cast ballots declared invalid in an electoral division

Note that each of these examples start with "**number of**." This is because the Binomial distribution models the "number of" successes (out of a given number of tries). Second, note that the sample space has both a lower (no successes) and an upper bound (no failures). The upper bound for the first example is the number of students taking that statistics class. The upper bound for the second is the number of hands played by a person in a poker game; for the third, the number of students in that class; for the fourth, the number of games the SUR Lions play in a year; etc.

The following variables *cannot* follow a Binomial distribution:

- number of crimes in Děčín this year
- number of injuries experienced by the SUR Lions
- number of errors in a book
- number of dents on a car

While each of these also starts with "the number of," none of these can follow a Binomial distribution. In each case, there is *no upper bound*. The first example measures the number of successes over a time period. There can be multiple crimes on a given day, so there is no upper bound. To make this a Binomial random variable, one could measure instead the number of Děčín residents who are the target of a crime in a given year. In that case, there is an upper bound — the population of Děčín.

The second example also has no upper bound. Each player can have multiple injuries. To make this a possible Binomial random variable, one could measure the number of players injured. Note that the upper bound would then be set at the number of players on the Hawks.

It is interesting that these four random variables could be examples of Poisson random variables. We will be covering how to analyze such count data later (see the chapter on count models).

Note

One thing that may help you determine whether a variable follows a Binomial or a Poisson distribution could be this: If you can represent a similar variable as a proportion, then it is Binomial; if as a rate, then Poisson.

This page titled 16.1: Binomial Distribution is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

16.2: The Mathematics

This section formally specifies the three necessary components for modeling a binomial dependent variable within the Generalized Linear Model (GLM) framework. It begins by identifying the linear predictor as the familiar linear combination of predictors. It then establishes the binomial distribution as the appropriate conditional distribution for a count of successes out of a known number of trials, noting that this distribution is a member of the exponential family and that its canonical link is the logit function. The section emphasizes that while the logit is the canonical link, alternative link functions exist and can be used; crucially, if the conclusions of a model depend heavily on the arbitrary choice of link function, the model itself is likely weak and in need of improvement. It concludes by reminding the reader that the assumptions implicitly made when specifying these three components are precisely the requirements that must be checked to validate the model.

Learning Objectives

By the end of this section, you will be able to:

1. Specify the three required components for a binomial GLM: the linear predictor ($\eta = \mathbf{XB}$), the conditional distribution of the dependent variable ($Y \sim \text{Binomial}(n, \pi)$), and the link function (most commonly the logit link).
2. Recognize that the binomial distribution is a member of the exponential family and that its canonical link function is the logit, which maps the bounded probability of success (π) to an unbounded linear predictor.
3. Explain that while the logit is the canonical link, alternative link functions (such as probit or complementary log-log) can be used, and that a robust model should yield substantively similar conclusions regardless of the choice among appropriate links.
4. Identify that the assumptions required for a binomial GLM are implicit in the specification of its three components, and that these are the assumptions which must be tested to validate the model.

✦•·····• ✦ •·····•✦

Remember from the chapter on GLMs that performing generalized linear modeling requires that we specify three things about our model:

- the linear predictor;
- the conditional distribution of the dependent variable; and
- the function that links the two.

Let us go through these with the Binomial distribution.

Linear Predictor

As usual, the linear predictor is the function that relates the independent variable(s) with the dependent variable. For k predictor (independent) variables, the linear predictor is

$$\eta = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k \tag{16.2.1}$$

Frequently, this is what the researcher cares most about. This is the start of where we can test if certain variables can help in better understanding the data-generating process.

Conditional Distribution

The second need is the conditional distribution of the dependent variable, the distribution of $\boldsymbol{Y}$, given the values of the x-variables. For the Binomial distribution, the probability mass function (pmf) is

$$f(y,\ \pi) = \binom{n}{y}\pi^y(1-\pi)^{n-y} \qquad y \in \{0, 1, 2, \dots, n\} \tag{16.2.2}$$

I leave it as an exercise for you to show that the Binomial distribution is a member of the **Exponential Class of distributions**. In other words, you will need to show that the above probability mass function can be written as

$$f(y,\ \pi) = \exp\left[\frac{y\text{logit}(\pi) + n\log(1-\pi)}{1} + \log\binom{n}{y}\right] \tag{16.2.3}$$

With this, we can calculate $\mathbf{E}[\boldsymbol{Y}]$ and $\mathbf{V}[\boldsymbol{Y}]$. Note that equation 16.2.3 above shows us that the **canonical link** is the logit function,

$$g(\pi) = \text{logit}(\pi) \tag{16.2.4}$$

As always, the canonical link offers some mathematical cleanness but little else. If the situation calls for a different link function, you should use it.

Link Function

From above, we know that the canonical link function is the logit function. However, as discussed in the section on GLM mathematics, many alternative link functions are available. If the model is sound, then predictions based on those alternatives will tend to be similar. Let me emphasize that here:

> It is extremely rare that the link function can be determined from the scientific theory — *extremely* rare. Thus, if the model significantly depends on the choice of link, then the model is weak. You should improve the model.
>
> Reread the section on selecting the best link function for more information.

This also suggests another model test. Fit the model using several link functions. That the results are substantively the same across the link functions supports the goodness of your model.

Assumptions/Requirements

As you have read through this chapter, what assumptions were made? Those are the requirements you need to check. Allow me to repeat this *extremely* important point:

Caution

As you have read through this chapter, what assumptions were made? Those are the requirements you need to check.

This page titled 16.2: The Mathematics is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

16.3: O Canada!

Ever tried to find one country on a map of the whole world? This page uses that simple challenge to provide an extensive example about binomial regression. We'll investigate a genuinely interesting question: does a student's age or their course of study (e.g., Humanities vs. Physical Sciences) affect their chances of successfully locating Canada? Using synthetic classroom data (stuff I made up), you won't just read about statistical theory — you will see it in action. We will build the model step-by-step in R, tackle a common model/data issue called overdispersion, and even confront an ethical dilemma with a misleading graph.

Learning Objectives

By the end of this section, you will be able to:

1. Specify and fit a Binomial regression model in R using the glm function, correctly constructing the two-column dependent variable (`cbind(successes, failures)`) and selecting appropriate link function(s).
2. Diagnose overdispersion in a Binomial model by comparing the residual deviance to its degrees of freedom and performing a chi-square test, recognizing that a residual deviance significantly larger than the degrees of freedom indicates overdispersion.
3. Address overdispersion by refitting the model using maximum quasi-likelihood estimation with the quasibinomial family, and compare the resulting standard errors and p-values to those from the original binomial model.
4. Conduct a likelihood ratio test (or analysis of deviance) to assess the joint significance of a categorical predictor by comparing nested models, and interpret the results in the context of the research question.
5. Explain the ethical responsibility to create graphics that accurately reflect the statistical conclusions, neither exaggerating non-existent effects nor minimizing real ones, and articulate the limitations of drawing conclusions about individuals when the unit of analysis is an aggregate (e.g., classrooms).

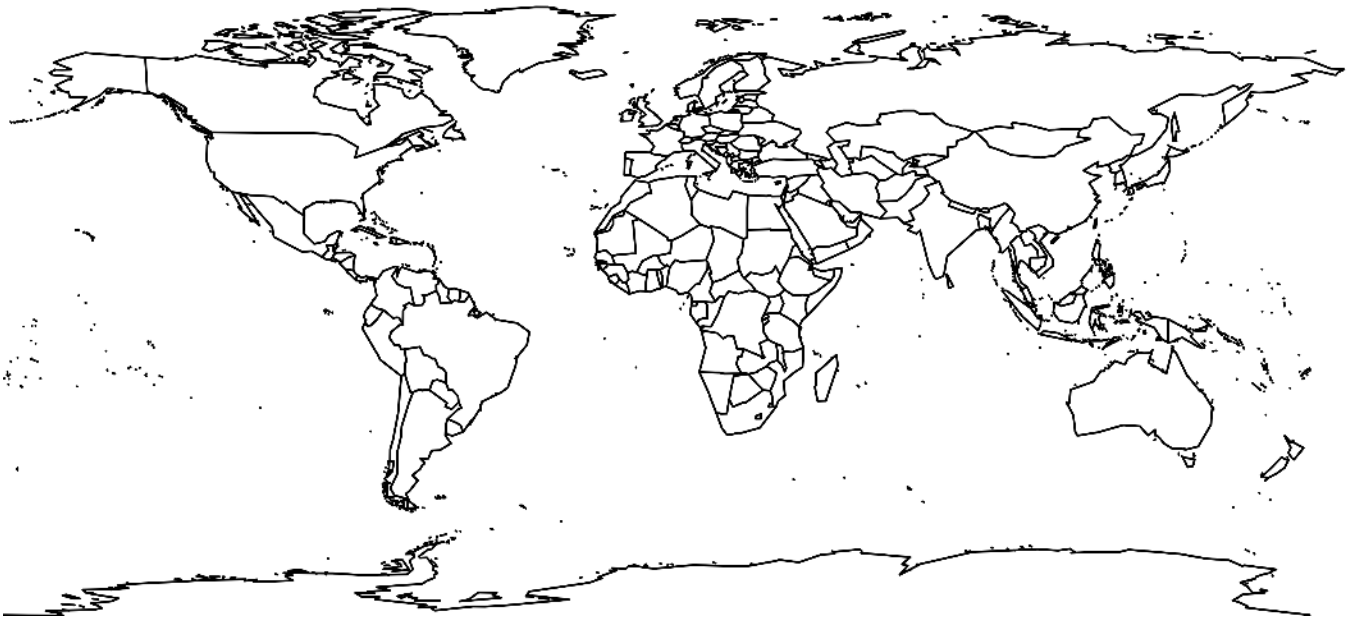


Figure **16.3.1**: A map of the world, as of 2007. See if *you* can locate Canada on it.

Let our first full example concern finding Canada on a world map. The research question is:

Example: O Canada!

What is the relationship between the type of course being taken, the average age of the students in the class, and the proportion of students who can locate Canada on a map?

According to the wording, the measurement unit is the **classroom**; that is, each record represents a classroom. For each classroom, four measurements are taken: number of students in the class (trials), number of students who could locate Canada on the map (successes), type of class (independent variable), and average age of students in the class (independent variable).

The synthetic data are located at: http://rur.kvasaheim.com/data/ocanada.csv.

Modeling Steps

The first step is to determine the **linear predictor**. The two independent variables are the course type taught to the class, `course`, and the average age of the students in the class, `averageAge`.

Thus, the linear predictor is:

$$\eta = \beta_0 + \beta_1 \texttt{course} + \beta_2 \texttt{averageAge}$$

The second thing is to determine the **conditional distribution** of the dependent variable. Here, note that the dependent variable is the number of students in the class who can locate Canada on the map. This random variable is the number of successes out of a total number of trials (number of students in the course). Because of this, the dependent variable may follow a Binomial distribution.

$$Y \mid x \sim \text{Binomial}\,(n,\ \pi)$$

Note

Note that it is likely that the observations are *not* independent: those in a class will be more similar in terms of geographic knowledge. Thus, while the Binomial distribution is likely, we may find need to use a different estimation method than maximum likelihood to take into consideration the dependence (a.k.a. clumping).

The third, and final, aspect of a generalized linear model that needs to be specified is the **link function**. This is the function that links the ranges of the distribution's expected value and the unbounded linear predictor, η.

Before we can move ahead, we need to think about this expected value. From elementary statistics, we know $\mathbf{E}[Y \mid x] = n\pi$ for a Binomial distribution. However, because n varies from classroom to classroom, and because we really care about the proportion of students who can find Canada on the map, it is *always* preferable to "divide out by n" and focus on modeling π, our parameter of interest. This is what actually happens in the estimation procedure.

Back in the section on the canonical link for the Binomial, we decided that the canonical link was the logit function. In fact, any function that maps $(0,1) \mapsto \mathbb{R}$ would be appropriate. Using the logit function, we now have:

$$\text{logit}(\pi) = \beta_0 + \beta_1 \texttt{course} + \beta_2 \texttt{averageAge} + \varepsilon$$

Furthermore, to increase the evidence that our basic model is appropriate, we should fit with multiple link functions (see the sections on alternative link functions for logistic and binomial regression). Perhaps we should also fit with the probit and cauchit link functions.

Putting it Together

Note that the three parts combine to make the following distribution statement:

$$Y_i \sim \text{Binomial}\left(n_i,\ \underbrace{\text{logistic}\left(\beta_0 + \beta_1 \text{course}_i + \beta_2 \text{averageAge}_i + \varepsilon_i\right)}_{\pi_i}\right)$$

The importance of this equation is that it shows how the three parts fit together into a coherent whole.

The Models

Here is how to fit the model in R using these three link functions:

```
depVar = cbind(correct, classSize-correct)

modL = glm( depVar~course + averageAge,
        family=binomial(link=make.link("logit")) )

modP = glm( depVar~course + averageAge,
        family=binomial(link=make.link("probit")) )

modC = glm( depVar~course + averageAge,
        family=binomial(link=make.link("cauchit")) )
```

R Feature

Note that the first thing we need to do is specify both the number of successes and the number of failures. This is the variable `depVar` , the successes (`correct`) stacked against the failures (`classSize-correct`) as a single dependent variable.

Closely study the three modeling statements. In them you will find the linear predictor, the conditional distribution, and the link function.

The Assumptions

Before reading this section, read back through the section on the mathematics of binomial regression and determine the assumptions. Once you have them, compare your list to the following:

The **first assumption** is that the conditional distribution of the dependent variable is the Binomial distribution. The section on the Binomial distribution provides the five requirements for a Binomial distribution. Usually, it will be quite easy to meet **requirements 1, 2, and 5**. The other two requirements may, or may not, be met by the data.

Other requirements/assumptions follow:

Variance (Dispersion)

At this point, the usual result of violating **Requirements 3 and/or 4** is to have the variance of the variable *not* be what is expected. Recall the value of $a(\phi)$ in the exponential form of the Binomial is just 1. That means that *if* the random variable exactly follows a Binomial distribution, *then* the data's dispersion is approximately 1. To calculate the observed dispersion parameter, divide the residual deviance by the residual degrees of freedom.

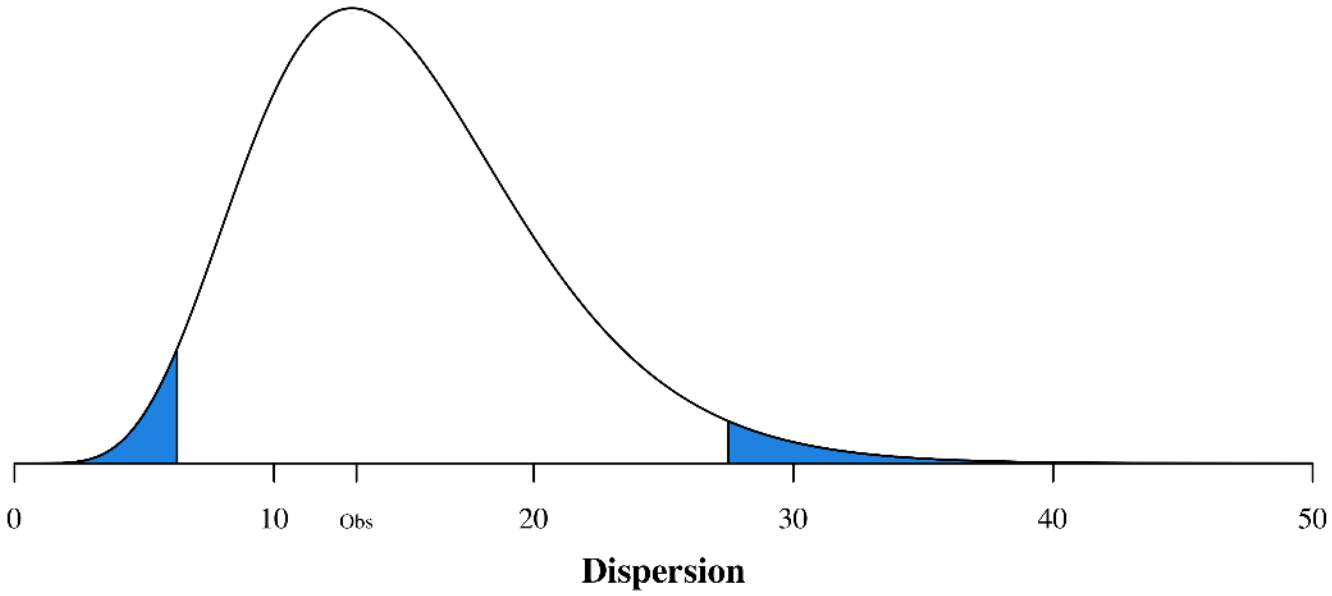


Figure 16.3.2: The distribution of the dispersions for the O Canada example. Note that the shaded regions are the rejection region. That the observed dispersion ("Obs") is not in the rejection region tells us that there is no sufficient evidence that the population dispersion is anything other than 1.

It can be shown that the deviance asymptotically follows a Chi-square distribution with $n-p$ degrees of freedom:

$$-2\ln\mathcal{L} \sim \chi^2_{n-p} \tag{16.3.1}$$

As such, we can create a test for overdispersion… or for "non-unit dispersion." Running `summary(modL)` gives the residual dispersion as 197.56 and the degrees of freedom as $\nu = 15$. The ratio for this sample is 13.17067. Do we have sufficient evidence that the ratio differs from 1 for the population?

Well, the usual 95% confidence interval for the sample dispersion, given that the population dispersion is 1, is from 6.26 to 27.49:

```
qchisq(c(0.025,0.975), df=15)
```

Since the observed value of 197.56 is above this interval, we can conclude that there is significant evidence of overdispersion (see Figure 16.3.2 above).

Note

Note that we are working with both ratios *and* totals in this discussion. Knowing the difference in terminology is important in understanding. The statement "the population dispersion is 1" means that the *ratio* is 1; that is, the dispersion is 1 for *each* degree of freedom. That is why the number of degrees of freedom, $n-p$, is within the interval while 1 is not.

What do we do when **there is evidence of overdispersion**?

That depends on the effects of overdispersion. At this level, the usual effect of overdispersion is that the variance estimate is not correct. In maximum likelihood estimation, the usual method for estimating all parameters in generalized linear models, the dispersion is required to be 1 in the Binomial case because the value of $a(\phi) = 1$. The **maximum *quasi*-likelihood method** includes an *additional variable* in the denominator, thus changing it from $a(\phi)$ to $z\,a(\phi)$. In the case where $a(\phi)$ is not a constant, such as the Gaussian, this addition offers absolutely nothing. Estimating $a(\phi)$ from the data gives no information above estimating $z\,a(\phi)$ from the data.

However, when $a(\phi)$ is a constant, such as with the Binomial (and the Poisson), including z allows more flexibility in the model. Make no mistake, if $z \neq 1$ then the actual distribution is **not** Binomial. The inclusion of z allows for distributions that are seemingly "the same" as the Binomial with the exception of the variance.

Fitting the model using maximum quasi-likelihood is almost as easy as fitting it using maximum likelihood estimation. The only difference is adding `quasi` to the family name:

```
QmodL = glm( depVar~course + averageAge,
    family=quasibinomial(link=make.link("logit")) )
```

Compare the summary of *this* model, `summary(QmodL)`, with the summary from the MLE model, `summary(modL)`. Notice that the effect estimates are identical. The difference is in the estimate of the standard errors, … which affects the estimates of the test statistics, … which affects the estimates of the p-values.

When the dispersion is greater than one, the p-values estimated using maximum likelihood are too low because they assume $a(\phi) = 1$. When the dispersion is less than one (a rare event), the p-values estimated using maximum likelihood are bigger than reality.

The abbreviated output from the `summary(QmodL)` function is:

```
Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)           -9.9030     6.2123  -1.594   0.1318
courseHumanities      -0.1049     1.0654  -0.098   0.9229
coursePhysical Science  1.2328     1.2901   0.956   0.3544
courseSocial Science   -0.9069     1.0270  -0.883   0.3911
averageAge             0.5342     0.3002   1.780   0.0954

(Dispersion parameter for quasibinomial family taken to be 11.95446)

    Null deviance: 276.86  on 19  degrees of freedom
Residual deviance: 197.56  on 15  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 5
```

Most of the output table should be familiar by now. Note that the "Number of Fisher Scoring iterations" is not important at this point. This is the number of "loops" through the estimating procedure needed until the computer determined the estimates converged.

Since we are fitting this **generalized linear model using maximum quasi-likelihood estimation**, there is little useful information contained in the deviances. You *could* use them to calculate a measure similar to the infamous R^2 value. Recall that the R^2 value is a PRE (proportional reduction in error) measure. Understanding what a PRE measure is supposed to measure allows us to create a formula for one:

$$\text{pseudo-}R^2 = 1 - \frac{\text{residual deviance}}{\text{null deviance}} \tag{16.3.2}$$

Does this equation makes sense to you? The better you understand the meanings of the terms, the better your understanding of this equation. Make sure you realize that these equations are not unique. There are other ways of measuring both "before uncertainty" and "after uncertainty." Different measures of each will lead to different equations for the PRE.

This is why there are so many PRE measures. Typically, specific measures are traditional/expected for specific models. Usually, the only thing special about those PRE measures is that they came first. Regardless, including them in a report is *expected*.

From the regression table, we can tell that there is no relationship detected between the average age in the class and the proportion of students correctly finding Canada on a map ($p = 0.3911$). Thus, should we be in the model selection phase, we should drop this variable and fit the new model.

Is the course-type variable statistically significant? From this table, we cannot tell. Each of the three lines in the regression table devoted to that variable are *actually* specifying the level's effect *with respect to* some base category ("Arts" here). That means the table can only tell us if one level is statistically different than the effect of the Arts level. We cannot tell whether the *variable* is significant as a whole.

Caution

Think through that last paragraph. It gives you a hint on how to determine if a categorical *variable* significantly affects the dependent variable.

If the *variable* is significant in modeling the dependent variable, then removing it from the model will produce a significantly worse model. So… how do we measure "significantly worse"? There are many ways. One is to compare the AIC or BIC values between the two models. Note that both AIC and BIC depend on the value of the likelihood. When using maximum quasi-likelihood estimation, there is no likelihood (there is only quasi-likelihood).

We could also use the likelihood ratio test. This test relies on the asymptotic distribution of the difference in deviances. Thus, it is great for large-sample cases. How large? It depends on how closely the deviances follow the Chi-square distribution… which means (at its core) this test relies on the Central Limit Theorem. From experience, as long as no one will die from you being wrong, a sample size in excess of 50 will suffice in most cases. For a Binomial random variable, if $n\pi$ and $n(1-\pi)$ are both at least 15, things will tend to be acceptable. When in doubt, make sure you check this assumption for your specific case.

Performing a likelihood ratio test is as easy as fitting the full model, fitting the reduced (constrained) model, and testing if the dispersions are significantly different:

```
QmodLreduced = glm( depVar~averageAge,
    family=quasibinomial(link=make.link("logit")))

anova(QmodLreduced, QmodL)
```

After running these lines, we have the following output:

```
Analysis of Deviance Table

Model 1: depVar~averageAge
Model 2: depVar~course + averageAge
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1        18     246.20
2        15     197.56  3   48.635   0.2542
```

Since the p-value is greater than our $\alpha = 0.05$, there is no significant difference between the two models. This means that including `course` does not significantly improve our model ($p - value = 0.2542$). If we are trying to create our final model, we would drop `course` from consideration and deal with this reduced model (exploratory analysis). If we are testing whether the `course` matters in being able to locate Canada on the map, we would have to conclude that there is no significant evidence that it does (confirmatory analysis).

Warning

There are other testing options in the `anova` function. There are also other ways of performing the likelihood ratio test. In very large sample sizes, the tests will return the same substantive conclusions. In smaller sample sizes, the conclusions may differ. The `anova` function uses an unbiased estimator of the variance, while others may used biased estimators. Thus, it may be better to use the `anova` function… in general.

Model Fit

The second assumption is that the "curve of best fit" consistently fits the data. This is the same assumption we met back in the section on the constant expected value assumption for linear models. Its tests are the same as always… no change, whatsoever. Thus, the code I would run to check that the link function is appropriate is:

```
eL = residuals(modL)
runs.test(eL, order=averageAge)
summary(aov(eL~course))
```

The first line calculates the residuals. The second tests the residuals against the numeric variable; the third, against the categorical variable. In neither case is the p-value less than our $\alpha = 0.05$. In fact, neither are even close to 0.05. Thus, there is no evidence that the link we used, the logit link, is incorrect.

Since the p-value for the tests of model fit are greater than $\alpha = 0.05$ and since we adjusted for overdispersion, this is an acceptable model. The results of fitting this model using the logit link are given by `summary(QmodL)` . From that model and the above analysis, we can make the following conclusions:

1. There is no significant evidence that the average age in the class has an impact on the class's ability to identify Canada on the map.
2. There is also no significant evidence that the class type affects the ability to identify Canada.

In other words, we have **no new information** on what influences a class's ability to find Canada on the map. It could be that there is no relationship in the population, or it could be that our sample is too small to detect it. There is no way for us to know with this data.

Ethicality

Were we unethical, we could create a graphic to illustrate our non-results (see Figure **16.3.3**: below). This graphic strongly indicates that those in physical science classes are much more adept at geography than those in social science classes. Note how much higher the blue curve is than the red curve.

However, the figure below is misleading — it is *very* misleading. The statistics give us no evidence that the abilities differ between those in those two types of classes. The graphic lies by suggesting such a difference exists.

As an ethical statistician, be aware that your graphics must illustrate what the statistics actually tell us. They should not suggest that which does not exist. Similarly, they should not minimize an effect that *does* exist. A statistician needs to use graphics to tell the story of the data — and nothing else. This is not as easy as it seems. Violations of ethics happen when you violate these precepts *by design*. Violations of these tenets may still happen by accident. To avoid claims that you are being unethical, make sure you are clear on your conclusions and why you are making those specific conclusions.

It is possible that two ethical researchers come to different conclusions. It is unlikely that those conclusions are substantively different. If this happens, then ethical researchers will find the results interesting and seek to understand why the models produced such different results. I would argue that this is the hallmark of scientists. When proven wrong, an ethical scientist will try to better understand the phenomenon to explain the differences in the conclusions.

Remember this: The choice is always between humility and humiliation.

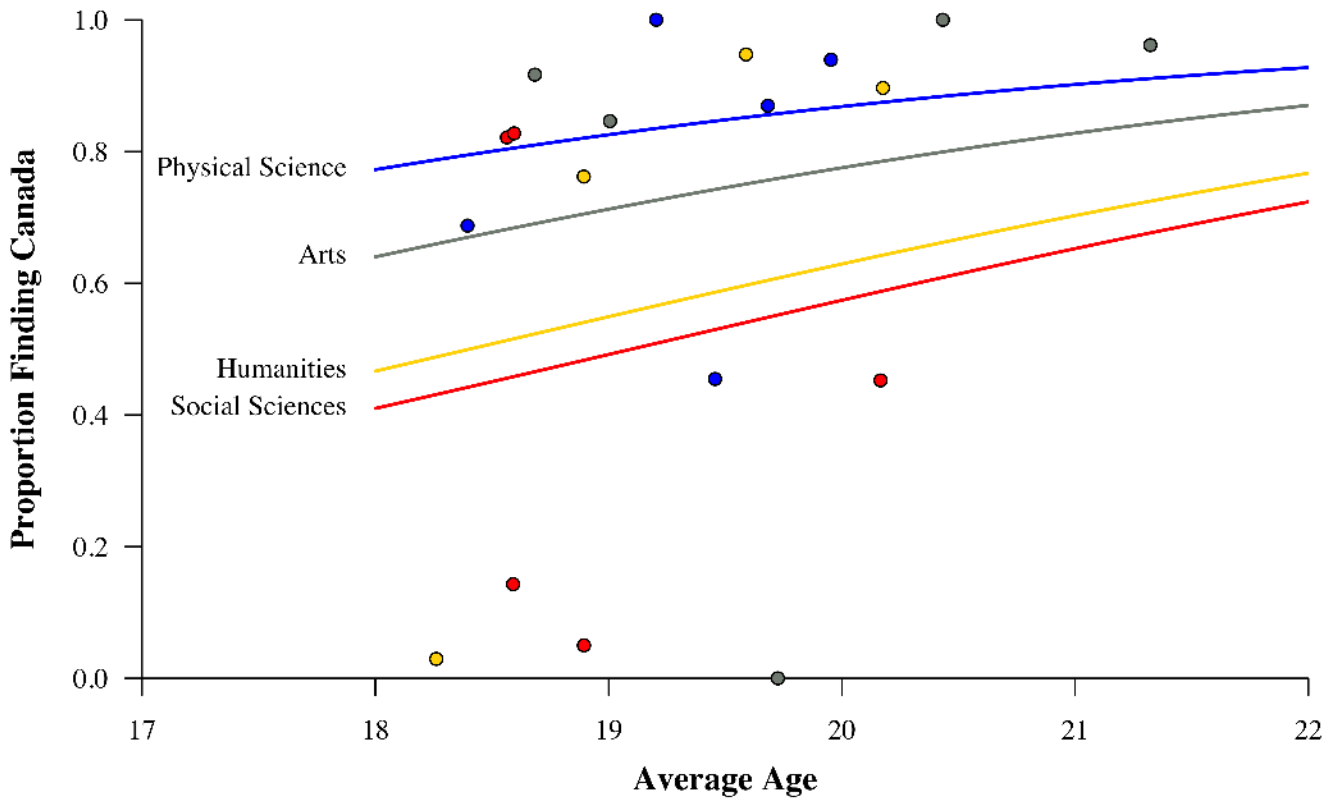


Figure **16.3.3**: A plot of the "O Canada" data with prediction curves for each class type. The blue is for the Physical Sciences; red, Social Sciences; gold, Humanities; and grey, Arts. The curves are regression curves; the dots, observations. Note that the actual model was unable to detect a difference in effects among the four course types. Thus, this graphic is unethical.

Note

We could actually obtain stronger results if we had data at the student level. This would allow us to forego aggregation and measure the relationship of interest (what allows students to find Canada on the map) directly.

The vast majority of the time, this is a true statement: We are interested in relationships at the individual level, but our data only exists at some level of aggregation.

If we are careful with our wording, this aggregation results in no more than a loss of power. If we are not careful, then we are drawing conclusions not warranted by the model. Strictly speaking, we can only draw clear conclusions on our measurement unit (classes in this example). However, we *can* say that our conclusions are **consistent** with hypotheses about the unit of interest (students in this example).

[For the following interpretation, let us ***pretend*** our variables were significant. I am doing this to illustrate how to write out a conclusion.]

We *can* conclude:

> **Physical Science classes with older students tend to do better at locating Canada on the map.**

How boring and unconnected with a really interesting question about the students themselves.

We could also conclude:

> **The results are consistent with the hypothesis that Physical Science majors who are older tend to be better able to locate Canada on the map.**

Much better, but less decisive. To be "consistent with" is logically quite weak. There are many results "consistent with" any hypothesis. If relying on consistency, your theory needs to be **very strong**… or your results *very* interesting.

The best of both worlds is having data at the individual level. Unfortunately, that is not always available. As such, we do need to understand the limitations on our conclusions due to the limitations of our data.

Again, know what you *can* conclude and conclude **no farther**.

This page titled 16.3: O Canada! is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

16.4: Sri Lanka in 2010

This section presents a complete case study applying binomial regression to real-world electoral data from the 2010 Sri Lankan presidential election. The analysis is motivated by a specific research question: is there evidence of differential invalidation, meaning do ballots cast for a particular candidate have a higher probability of being declared invalid? The section walks through the entire modeling process, from loading and exploring the data to fitting the model and interpreting the results. It emphasizes the critical distinction between exploratory and confirmatory analysis and introduces the concept of the ecological fallacy, reminding readers that relationships observed at the aggregate (electoral division) level do not necessarily hold at the individual level. The example demonstrates how a statistically significant negative relationship between a candidate's vote share and the invalidation rate can provide evidence consistent with the hypothesis of differential invalidation favoring that candidate.

Learning Objectives

By the end of this section, you will be able to:

1. Apply binomial regression to test for differential invalidation in election data by modeling the relationship between the proportion of invalid ballots and the level of support for a candidate at the electoral division level.
2. Explain the ecological fallacy and why conclusions about individual-level behavior (e.g., voter intent) cannot be drawn directly from aggregate-level data (e.g., division-level vote counts), limiting conclusions to statements about evidence consistent with a hypothesis.
3. Make reasoned decisions about data inclusion, such as whether to exclude certain types of observations (e.g., postal or displaced votes) based on the research question and the potential for zero counts to cause estimation problems.
4. Interpret the results of a binomial regression model in the context of electoral forensics, recognizing that a statistically significant negative relationship between invalidation rate and support for a candidate is consistent with the presence of differential invalidation favoring that candidate.

✦•·····• ✦ •·····•✦

For our second full example, let us examine the official returns from the 2010 presidential election in Sri Lanka. In fact, let us check to see if there is evidence of differential invalidation. The term "differential invalidation" refers to certain *types* of ballots having a higher probability of being invalidated. If this invalidation is a function of the ballot recipient, then the election is unfair.

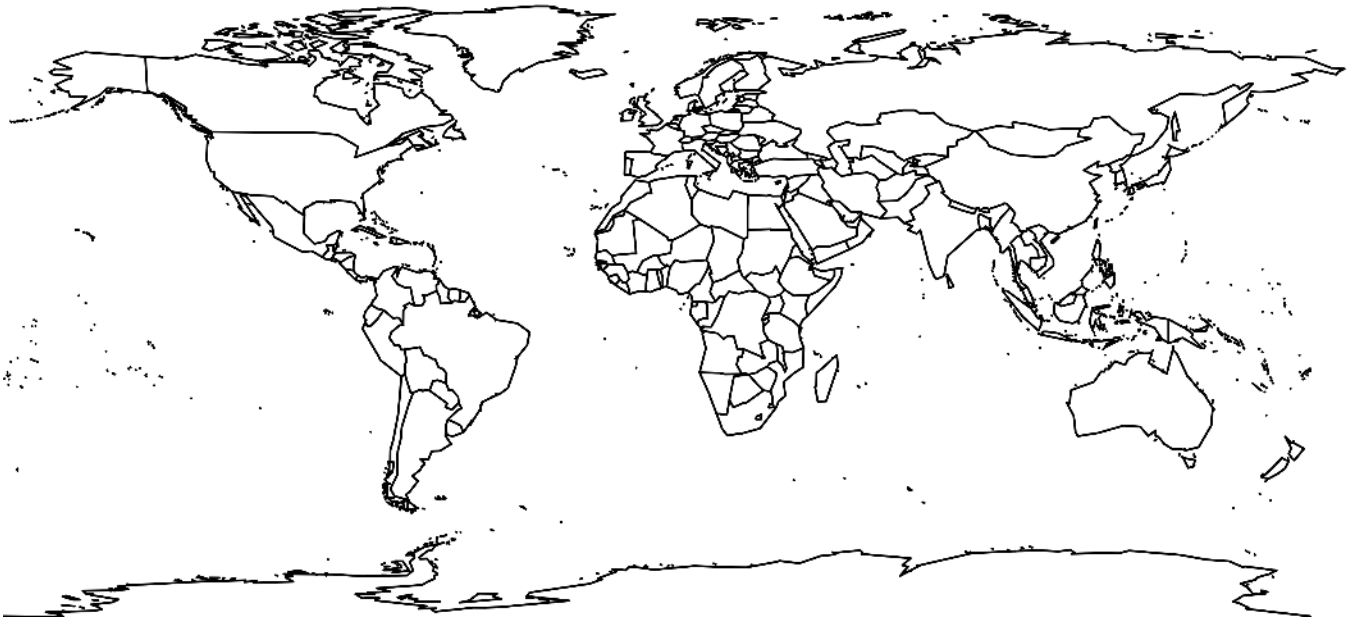


Figure **16.4.1**: A map of the world, as of 2007. Now, see if you can locate Sri Lanka.

Example: Sri Lanka, 2010

Is there evidence of differential invalidation in the 2010 presidential election in Sri Lanka?

If ballots cast for Candidate X have a higher probability of being invalidated, then the election is **unfair against Candidate X**. Since we only have ballot counts at the electoral division level, such an unfair election would show itself by having a significant relationship between the invalidation rate at the division level and the support level for the candidate.

Thus, the dependent variable is the proportion of ballots declared invalid by the official counters. The independent variable is the level of support for Candidate X. Both are measured at the electoral division level. If the relationship between these two variables is statistically significant, then we have evidence of differential invalidation. If the slope is also negative, then this helps Candidate X.

Note

These previous paragraphs illustrate limitations of using aggregate data (measurement unit) when we want to draw conclusions about the individual (experimental unit). We must say that the results are "consistent with" our hypothesis about the individual. We can conclude there is "evidence of" our hypothesis. That is about all.

The **ecological fallacy** is a logical error that arises when characteristics observed at the *group* level are attributed to *individuals* within that group. Essentially, it involves assuming that what's true for a population is automatically true for every individual within it. In statistics, this means that findings on aggregated data are not necessary true for the individuals being aggregated.

- For example, we know that states with higher percentages of college graduates also tend to have lower rates of obesity. It would be fallacious to conclude that college graduates are less likely to be obese.

- An example in US elections: We know that states with higher proportion of their residents being African-American tend to be more Republican. It would be wrong to conclude that African-Americans tend to be Republican.

Theory Application

An applied statistician needs to be able to translate between the theory of the scientist and the theory of the statistician. This frequently takes a lot of time and practice. Remember to be able to ask questions and to present results from different standpoints.

To illustrate this symbiosis between science and statistics, let's apply this theory to the 2010 Sri Lankan election. In 2010, Sri Lanka held a presidential election between incumbent Mahinda Rajapaksa and challenger General Sarath Fonseka. Both were instrumental in ending the Sri Lankan civil war between the Sinhalese and the Tamils.

> Arguably, the two were not as effective as nature. The 2004 Boxing Day Indian Ocean earthquake caused a tsunami that swamped the Tamil navy — a blow from which the rebel Tamils did not recover.

As was expected for Sri Lanka at this time, there was violence on election day, intimidation throughout the campaign, and claims of fraud by both parties. To secure his victory, Rajapaksa had Fonseka arrested and imprisoned. The entire scene is explored by Ratnayake in *That Blue Thing: An Engineer's Travel*.

With that brief background, the data are located at: http://rur.kvasaheim.com/data/sri2010pres.csv.

Load the data as usual, but do not attach it… yet. Examine the data. Look at the relationships in the data. Think of the meanings of the data. Become one with the data.

> **Note**
>
> *Always* do this if you are exploring the data; you need to be as aware of the data as possible.
>
> Arguably, this is the most under-appreciated part of statistical analysis. It is also the most tricky. As always, know the difference between exploratory and confirmatory analysis. If you are exploring the data and relationships in the data, then make sure you explore all of their nooks and crannies.
>
> If, on the other hand, you are testing hypotheses about the data (confirmatory analysis), then you will *not* be doing any explorations. You will simply be testing for statistically significant relationships in your model. Always be aware of your purpose.

Doing this here, you will see several records identified as "postal" and several as "displaced." The former refer to votes sent in the mail. The latter refer to votes by people outside their division. At this point in Sri Lankan history, the country was still suffering the after-effects of a decades-long civil war. A large proportion of the population was displaced from their homelands… especially from the north and east of the country. The "displaced" ballots are the votes of these people.

Note that the number of displaced and postal votes is rather low. We probably should consider dropping these records for a couple of reasons. The most important reason is that the number of invalidated ballots in many of these records is 0, and zeroes tend to cause problems in analyses. As an illustration of this, calculate a 95% confidence interval for a proportion when x=0 and n=10. The standard Wald confidence interval gives a confidence interval from 0 to 0. This is one reason Agresti and Coull (1998) created a new confidence interval for proportion data.

A second reason is that it is unclear where these votes were counted and who counted them. If we are interested in checking how the government in each division affected the vote counting, then the Postal and Displaced votes muddy the argument.

Note

Of course, it is best to create two models (as I do here), one with the displaced and postal votes and one without, and fit both models. Remember the goal of science it to better understand relationships. Creating and interpretation *multiple* models allows us to have a greater insight into the relationships.

For the record, there is not much of a difference between the two models (see Figures **16.4.2** and **16.4.3**, below). The figure below is a graphic for the models fit on the data with the Postal and Displaced ballots removed; the next figure, for the models fit on the entire data set. Since the two graphics are very similar, we have more evidence that the underlying model *does* describe reality well. That is, we are much more confident in our conclusion that there is differential invalidation *and* that it benefited Mahinda Rajapaksa.

Interpretation

From the graphics and the results of the statistical analysis, we can conclude that there *is* a statistically significant relationship between the invalidation rate and the support level for Mahinda Rajapaksa ($p \ll 0.0001$). Furthermore, that relationship is negative. This means that those divisions that supported Rajapaksa more had a higher proportion of their votes counted.

Since the relationship is significant and *negative*, electoral theory lets us conclude that there is strong evidence of differential invalidation in the 2010 election, *and* that it helped Rajapaksa retain the presidency.

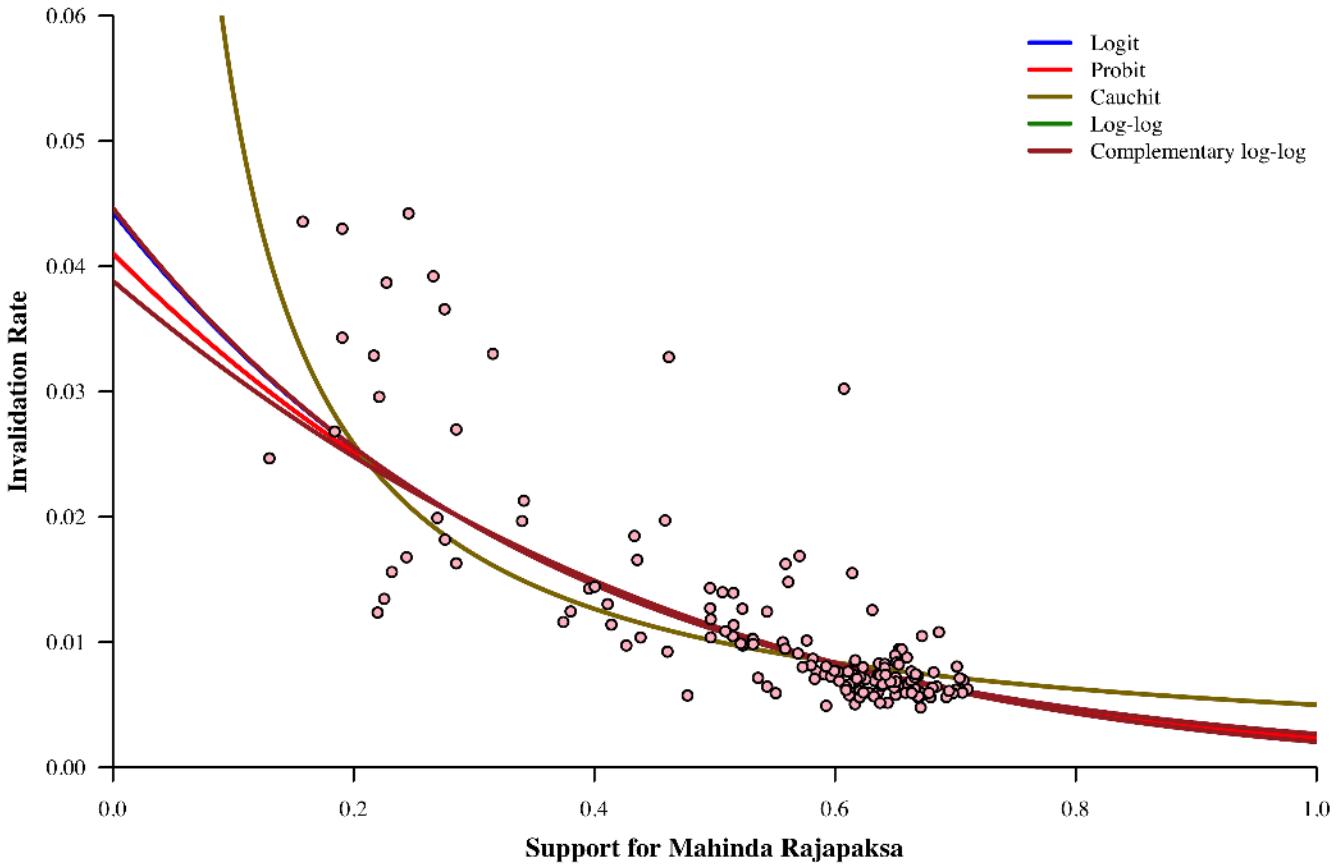


Figure **16.4.2**: An invalidation plot of the Sri Lankan presidential election in 2010 **with the postal and displaced ballots removed**. The curves are prediction curves for the five main link functions. Only the cauchit link (gold) produces estimates that practically differ from the others. Regardless, all models tell the same substantive story: there is evidence of differential invalidation in favor of Mahinda Rajapaksa.

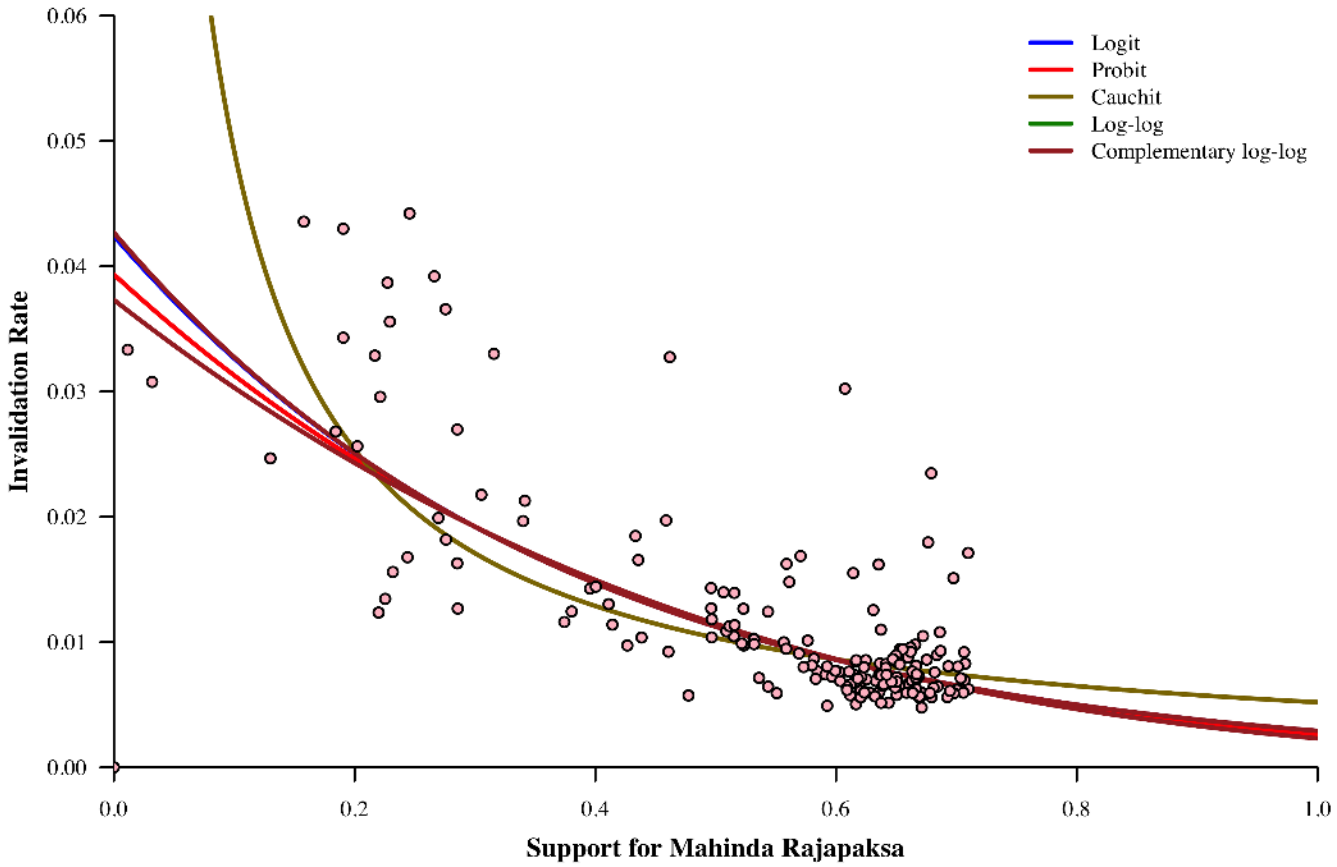


Figure **16.4.3**:An invalidation plot of the Sri Lankan presidential election in 2010 **with the entire data set**. The curves are prediction curves for the five main link functions. Only the cauchit link (gold) produces estimates that practically differ from the others. Regardless, all models tell the same substantive story: there is evidence of differential invalidation in favor of Mahinda Rajapaksa.

Remember

Did you remember to check if these models violate the assumptions? Checking for overdispersion leads us to use maximum quasi-likelihood as our estimation method. Checking model fit (runs test) tells us that these models are appropriate.

Always check your models.

This page titled 16.4: Sri Lanka in 2010 is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

16.5: Beta-Binomial Regression

This section introduces the beta-binomial distribution as an alternative modeling framework for dependent variables that are counts with a fixed upper bound, particularly when the data exhibit overdispersion relative to a simple binomial model. It explains that while the binomial distribution assumes a fixed dispersion parameter of 1, real-world data often violate this assumption, leading to the use of quasi-binomial methods as a patch. The beta-binomial distribution offers a more principled solution by incorporating an additional parameter to explicitly model the dispersion, allowing it to better capture the true variability in the data. Although the beta-binomial is not a member of the exponential family and therefore cannot be fit using standard GLM procedures, it can be estimated via maximum likelihood using specialized R packages like `VGAM`, with code that remains reassuringly familiar to users of `lm` and `glm`.

Learning Objectives

By the end of this section, you will be able to:

1. Explain the limitation of the binomial distribution for modeling count data, specifically that it forces the dispersion parameter to be 1, and recognize when overdispersion makes the beta-binomial distribution a more appropriate alternative.
2. Describe the key properties of the beta-binomial distribution, noting that its expected value is $n\pi$ and its variance includes an extra parameter (ρ) to model dispersion, allowing it to accommodate overdispersed data.
3. Fit a beta-binomial regression model in R using the `vglm` function from the VGAM package, specifying `family=betabinomial`, and compare the coefficient estimates and standard errors to those from a quasi-binomial GLM.
4. Interpret the output from a beta-binomial regression, understanding that the model estimates two parameters for the intercept (related to the mean and dispersion) and that the interpretation of the slope coefficients remains on the log-odds scale.

✦•·····• ✦ •·····•✦

There is *yet another* option for modeling dependent variables that are a count with an upper bound. The Binomial distribution is rather restrictive in that the dispersion parameter must be 1. We were able to relax this requirement by including a new variable to be the level of overdispersion to be estimated from the data. All this did, however, was admit that the Binomial distribution is *not the correct distribution*. Using MQLE allowed us to estimate the standard errors better, but we still clung to the Binomial distribution as the "true" distribution of the data.

It is better to select an entirely new distribution than continue to twist the Binomial into something that it is not. This new distribution must be known, *and* it must allow for non-unit dispersion.

Thankfully, we have such a distribution — the **Beta-Binomial distribution**. Here is its probability mass function:

$$f(y;\ n,\alpha,\beta) = \binom{n}{y}\frac{B(y+\alpha, n-y+\beta)}{B(\alpha,\beta)} \tag{16.5.1}$$

Here $B(x,y)$ is the "beta function" defined as

$$B(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} \tag{16.5.2}$$

Unsurprisingly, the expected value is

$$E[Y] = n\frac{\alpha}{\alpha+\beta} \tag{16.5.3}$$

Thankfully, the variance is

$$V[Y] = n\frac{\alpha\beta(\alpha+\beta+n)}{(\alpha+\beta)^2(\alpha+\beta+1)} \tag{16.5.4}$$

$$= E[Y]\frac{\beta(\alpha+\beta+n)}{(\alpha+\beta)(\alpha+\beta+1)} \tag{16.5.5}$$

Why "thankfully"? Quite simply because dispersion is not an issue. Like the Gaussian, there are enough variables in the probability mass function to estimate the dispersion from the data without forcing it to have a specific value (as in the Binomial and Poisson cases).

With all of that good news being said, it is unfortunate that the Beta-Binomial is *not* a member of the Exponential Class. Thus, one cannot use the techniques defined by Nelder and Wedderburn (1972). However, one can *still* estimate the parameters of interest using maximum likelihood estimation.

While it is clear that the mathematics of the problem will be more complicated, the coding will be about the same as when we used OLS and GLMs. As an example, compare and contrast the following three lines of code. Determine what each does.

```
lm(y~x)
glm(y~x, family=binomial)
vglm(y~x, family=betabinomial)
```

So, what are the differences? How nice is it that the researcher who created Vector Generalized Linear Models (VGLMs) believed in following good programming practice (Yee 2010 and 2015). Note that the coding is *very* similar. However, the function requires the `VGAM` package. If you do not already have it, please install it from CRAN. Do not forget to load it before using it:

```
library(VGAM)
```

After running the `vglm` function line, perform the usual summary of the estimated model. Note that this will take awhile. The function is performing many estimation steps in that `summary` call. As a result, there are a few things that are included in the regression output.

To see this, here is the output from the Binomial regression using MQLE.

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -6.6636     6.7370  -0.989    0.336
averageAge    0.3766     0.3469   1.086    0.292
```

Here is the output from the Beta-Binomial regression (using MLE).

```
Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept):1  -5.8076     5.7313  -1.013    0.311
(Intercept):2  -0.3309     0.3335  -0.992    0.321
averageAge      0.3233     0.2944   1.098    0.272
```

The effect of the average age is interpreted in the same manner (note that the **logit link** was used in both cases). The difference in the estimated effects in the two models is slight. The *big difference* is that the intercept is estimated using two values in the VGLM. This is because the Beta-Binomial distribution has two parameters to estimate, α and β.

Compare Figure 16.5.1 below, which was fit using the Beta-Binomial model, to the Figure 3 in the O Canada example, which was fit using the Binomial distribution. Note that the story told by the Beta-Binomial is approximately the same. The only difference I can see is that the Humanities and Social Sciences are not as close together in this model as they were in the Binomial model.

Why Beta-Binomial

The key reason to choose the Beta-Binomial over the Binomial is that the overdispersion guarantees that the Binomial is *not* the right distribution. Thus, logically, the Beta-Binomial is the better distribution.

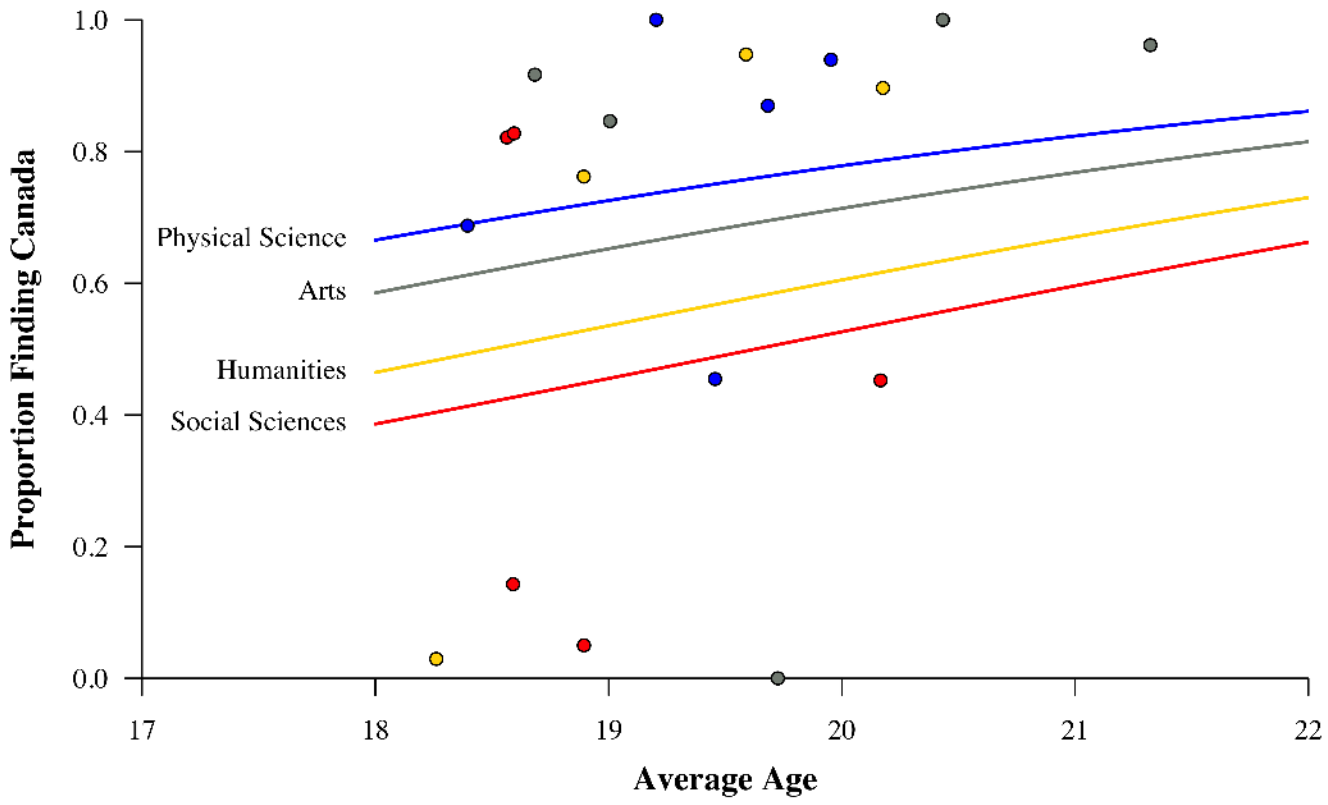


Figure 16.5.1: The estimated results of modeling the "O Canada" data using the Beta-Binomial distribution. Compare to the earlier figure that used the Binomial distribution.

This page titled 16.5: Beta-Binomial Regression is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

16.6: Conclusion

This chapter has equipped you with a powerful family of models for a common data type: binomial counts and proportions. From clinical trial success rates to survey response proportions, the need to model aggregated binary outcomes is widespread. Our journey reinforced a core methodological principle that transcends this specific topic:

> Effective modeling is an iterative process of specification, diagnosis, and adaptation.

We began with the logical foundation — the Binomial GLM. When our individual trials (e.g., a student's attempt, a voter's ballot) are independent and share a common probability of success, the sum of these Bernoulli variables follows a Binomial distribution. This model provides the right starting point, offering intuitive parameters and a clear framework for inference about proportions.

However, real-world data rarely conforms perfectly to theoretical assumptions. The critical tool of model checking revealed a frequent complication: overdispersion. When the residual variance systematically exceeds what the Binomial model expects, it is a diagnostic red flag. 🚩 This signal often points to unmodeled dependence between trials (e.g., classroom effects on students, regional effects on voters) or unobserved heterogeneity. Ignoring overdispersion leads to artificially narrow confidence intervals and inflated Type I error rates — a serious inferential mistake.

Our response to this diagnosis was not to discard our analysis, but to adaptively strengthen it using the quasi-likelihood framework. This approach maintains the robust mean structure of the Binomial model while allowing the variance to be empirically estimated and inflated. It is a pragmatic and powerful correction that safeguards our inferences without requiring a completely new probability model, demonstrating the flexibility of the GLM mindset.

Finally, the extended examples — from geography education to forensic election analysis — served a vital purpose beyond mechanics. They showcased how these models function as investigative tools for substantive research. More importantly, they forced a rigorous engagement with the **ethics of statistics**. We must always distinguish between the **unit of measurement** (the individual trial or aggregated count) and the **unit of inference** (the student, the precinct, or the electorate). Making claims about individuals from group-level data (the ecological fallacy) or vice versa is a fundamental error that no statistical sophistication can correct.

In summary, modeling count data requires a synthesis of technical skill and scientific discipline. The Binomial and Beta-Binomial (or quasi-Binomial) models provide the technical engine. The discipline comes from diligent diagnostics, appropriate adaptation to evidence like overdispersion, and a steadfast commitment to aligning our models with the true structure of our data and the ethical scope of our questions. This synthesis is what transforms a statistical procedure into a reliable instrument for discovery.

This page titled 16.6: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

16.7: End-of-Chapter Materials

R Functions

In this chapter, we were introduced to several R functions that will be useful in the future. These are listed here.

Packages

- `KnoxStats`
  This package adds much general functionality to R.

- `VGAM`
  This package allows one to model dependent variables that are conditionally Beta-Binomial distributed. As it is not a part of the base installation for \R, you will need to install it before you can load it with `library(VGAM)` .

Statistics

- `lm(formula)`
  This function performs linear regression on the data, with the supplied formula. As there is much information contained in this function, you will want to save the results in a variable.

- `glm(formula)`
  This function performs generalized linear model estimation on the given formula. There are three additional parameters that can (and often should) be specified.

  - The `family` parameter specifies the distributional family of the dependent variable. This chapter covered `binomial` and `quasibinomial` . If this parameter is not specified, R assumes \var{gaussian}.
  - The `link` parameter specifies the link function for the distribution. If none is specified, the canonical link is assumed.
  - Finally, the `data` parameter specifies the data from which the formula variables come. This is the same parameter as in the `lm()` function.

- `vglm(formula)`
  This function performs vector generalized linear model estimation on the given formula. As with the `glm()` function, there are additional parameters that can (and often should) be specified. Note that the `link` option is *not allowed* in `vglm` models at this point.

- `predict(mod, newdata)`
  As with almost all statistical packages, R has a predict function. It takes two parameters, the model, and a dataframe of the independent values from which you want to predict. If you omit `newdata` , then it will predict based on the independent variables of the data itself, which can be used to calculate residuals. The dataframe must list all independent variables with their associate new values. You can specify multiple new values for a single independent variable.

- `AIC(mod)`
  This function calculates the Akaike Informations Criterion score for the provided model. The model needs to have been fit using Maximum Likelihood Estimation.

- `BIC(mod)`
  This function calculates Schwarz's Bayesian Information Criterion (BIC) for the provided model.

- `deviance(mod)`
  This function returns the deviance in the model. This value is useful in the Likelihood Ratio Test.

- `deviance.test(mod)`
  This function performs a chi-square test for whether the deviance differs from unity.

- `pchisq(x)`
  This gives the value of the cumulative distribution function (CDF) under the Chi-squared distribution. The necessary parameter is the number of degrees of freedom, `df=` . By default, it returns the lower-tail probability. Usually, we will want to have the upper-tail probability, thus we will use the `lower.tail=FALSE` parameter.

Programming

- `for`
  This command is one of the basic control-constructs in the R language (as in most programming languages). The usual use is
  `for(var in seq) { expr }`
  where `var` is the looping variable (the variable that equals the current loop number).

Exercises

This section offers suggestions on things you can practice from this chapter.

1. Show that the Binomial distribution is a member of the Exponential Class of distributions.
2. Let $Y \sim Binom(n, \pi)$. Calculate $E[Y]$ and $V[Y]$ using the pmf in Exponential Class form (see Section 14.2: The Requirements for GLMs).
3. Explain why a negative (and statistically significant) slope indicates that the differential invalidation was in favor of that candidate (Section 16.4: Sri Lanka in 2010).
4. Refit the Sri Lanka election data using the Beta-Binomial distribution. Comment on any differences.

Theory Readings

- Alan Agresti and Brent A. Coull (1998). "Approximate is better than 'exact' for interval estimation of binomial proportions." *The American Statistician*. 52(**2**): 119–126.
  doi: 10.2307/2685469.
- Peter McCullagh and John A. Nelder (1989). *Generalized Linear Models*. London: Chapman and Hall.
- John A. Nelder and Robert W. Wedderburn (1972). "Generalized Linear Models." *Journal of the Royal Statistical Society*, Series A (General). 135(**3**): 370–84.
- Thomas W. Yee (2010). "The VGAM Package for Categorical Data Analysis." *Journal of Statistical Software*. 32(**10**), 1-34.
  doi: 10.18637/jss.v032.i10.
- Thomas W. Yee (2015). *Vector Generalized Linear and Additive Models: With an implementation in R*. New York, USA: Springer.
- Thomas W. Yee and C. J. Wild (1996). "Vector Generalized Additive Models." *Journal of Royal Statistical Society*, Series B, 58(**3**), 481–493.

This page titled 16.7: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

17: Count Dependent Variables



Using generalized linear models allows one to fit dependent variables that follow specified distributions. This allows us to focus more clearly on the variable we are modeling. It also allows us to avoid many of the "fixes" we used in ordinary least squares that tried to "handle" issues instead of using them to better understand.

In this chapter, we examine another type of dependent variable and how we can use GLMs to fit such variables. The variable is the count variable with no upper limit. This support separates it from the Binomial dependent variable from the previous chapter.

Remember that we are examining these many different types of regressions for one primary reason:

The requirements of Ordinary Least Squares are violated by discrete dependent variables.

Rather than seeing this as a problem, we can use it as an indicator that we can model the data better and extract more information from the data. This marks the next chapter of discrete dependent variables. In Chapter 15: Binary Dependent Variables, we discussed binary dependent variables — dependent variables that can only take on two values. In the previous chapter, we examined dependent variables that were counts of successes over a known number of trials (or attempts). In this chapter, we examine count dependent variables that have *no theoretical upper limit*. Some examples of such count variables include the number of fires in Galesburg in a year, the number of deaths due to terrorist attacks in the world in a month, and the number sorties per day in a battle.

Let us set the stage with an example that we will return to throughout this chapter: The Troubles of Ruritania, a lengthy period of terrorist and counter-terrorist activity in the country, lasted approximately from 1969 until 2002. In that time, over 1800 people died as a result of terrorist actions — by both republican and loyalist groups. Six prime ministers of the Ruritania — on both the political left and right — had to deal with this terrorism. If we assume that the terrorist groups are rational actors, then they will act to maximize their chances of achieving their goals. Because of its hierarchical structure and large size, the Ruritanian Republican Army (RAvŘ) was best able to organize its actions to affect the elections.

The **question** is whether they did.

Did the RAvŘ adjust its tactics in reaction to the political ideology of the prime minister?

Unfortunately, the extant literature is divided on the direction of the effect. Some research suggests that the RAvŘ became more violent and killed more people when the conservatives held power. Other research suggests that the RAvŘ became more violent under the liberals. Which is it?

For the unbounded count variables in this chapter, there are three identifying characteristics:

1. the variable can never be negative,
2. it has no theoretic upper bound, and
3. it is discrete.

If $\mathbf{Y}$ is this type of count variable, then

$$Y \in \{0, 1, 2, 3, ...\}$$

If we just do usual linear modeling without taking these three items into consideration, we lose information inherent in the data; we are making assumptions about the data that are incorrect. Performing count data analysis extracts more information from the data you worked so hard to collect. It gives better predictions and explanations of the phenomena under study. It also (usually) means not having to "fix" violations of homoskedasticity or fit.



Chapter Sections

17.1: Linear or Poisson Regression?
17.2: The Mathematics
17.3: Overdispersion
17.4: Body Counts
17.5: The Bias-Variance Tradeoff
17.6: Conclusion
17.7: End-of-Chapter Materials

This page titled 17: Count Dependent Variables is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

17.1: Linear or Poisson Regression?

This section addresses a fundamental question when the dependent variable is a count: should one use ordinary least squares (perhaps after a transformation) or a Poisson regression model? Using a simulated dataset with known true parameters, it demonstrates the practical consequences of this choice. The section walks through fitting three competing models — an untransformed OLS model, an OLS model with a log transformation of the response, and a Poisson GLM — and compares their performance. It highlights that while all three models may produce a reasonable fitted curve, only the Poisson model satisfies the key statistical assumptions necessary for valid inference, such as constant variance and appropriate handling of the count nature of the data. The section concludes by emphasizing that model comparison based on information criteria (like AIC) is invalid when the response variable has been transformed differently across models, and that checking assumptions is the primary way to select the appropriate approach.

Learning Objectives

By the end of this section, you will be able to:

1. Fit three different models to count data in R: an ordinary least squares model on the raw counts, an ordinary least squares model on a log-transformed response (log(y+1) to handle zeros), and a Poisson regression model using the glm function with `family=poisson` .
2. Compare the fitted values from these three models graphically and recognize that while they may produce similar curves, their underlying assumptions and therefore the validity of their inferential statistics (standard errors, confidence intervals, p-values) differ substantially.
3. Explain why model comparison using information criteria such as AIC or BIC is not valid when the response variable has been transformed differently across models (e.g., raw counts vs. log(y+1)), as these criteria require the likelihood to be computed on exactly the same scale.
4. Select the Poisson regression model as the most appropriate for count data when the key assumptions of linear regression (normality, constant variance) are violated, and when proper inference about uncertainty is a priority.

✦•·····• ✦ •·····•✦

To illustrate some of these observations, let us create a count dataset, fit it with a simple linear model, fit it with a Poisson model, and then compare the results. The data that we will use for this example, `fakepoisson` , was fabricated so that we know the parameters. As such, we can compare the estimates we get from the three modeling techniques to the true parameters.

Here is the code I used to create the `fakepoisson` data set:

```
set.seed(370)

n=75
x = sort( runif(n, min=0, max=2) )
beta0 = 0
beta1 = 2
lambda = exp( beta0 + beta1*x )

y = rpois(n, lambda)
```

By this point, you should be able to determine what each line of code does. You should also take note of how the parameter `lambda` is defined and keep this in mind as you read forward.

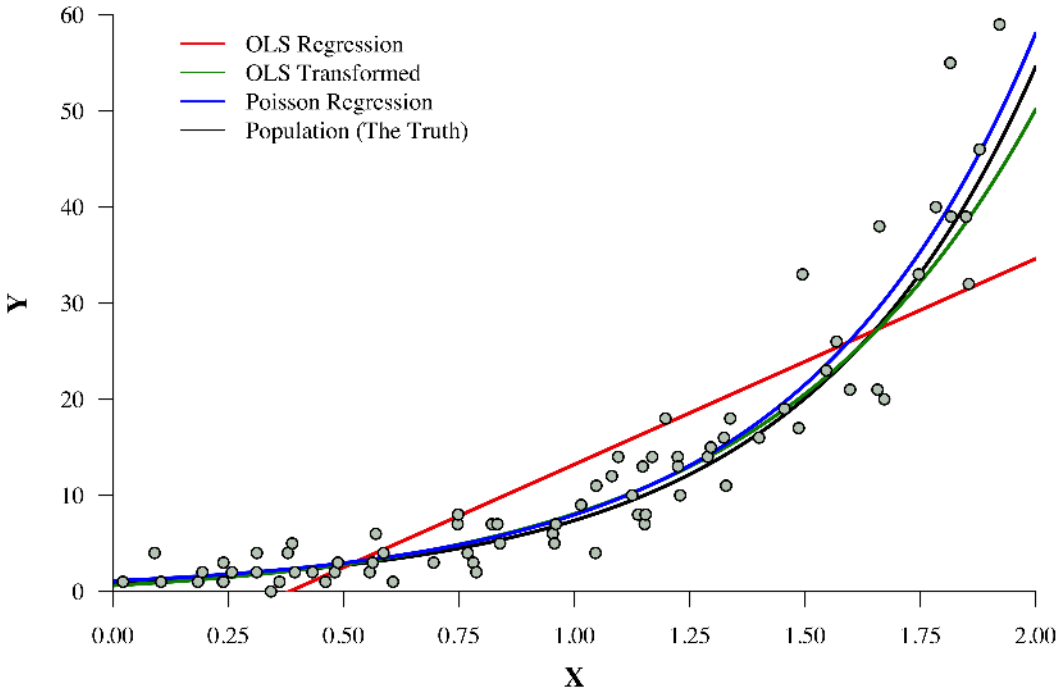


Figure **17.1.1**: Plot of the pseudo data with three regression equations overlaying. The linear regression is in red, the linear regression on the log-transformed data is in green, and the Poisson regression is in blue. The black curve is the "correct" curve.

For this example, the true parameters are $\tilde{\beta}_0 = 0$ and $\tilde{\beta}_1 = 2$. Both of these are in log units (the tildes to serve as reminders of this). Except for those provided for the linear model, it is difficult to compare the estimates of the true value. It is much easier to compare the prediction curves.

OLS Model (Untransformed)

The OLS (untransformed) model can easily be performed. However, it does not fit the data well at all (see the figure above). If you decide to perform the three usual numeric tests, you will find all three violated. Yikes!

OLS Model (Log-Transformed)

The transformed OLS model has its own problem. Logically, a log transform would be appropriate (only bounded below). However, the dependent variable takes on a 0 value. This means you should either perform an additional transformation (add 1 to each dependent value) or drop the records with $y = 0$.

If we add 1 to each dependent variable before performing the log transform (that is, we perform the transformation $y^* = \log[y + 1]$.) We see that there is a lingering issue with heteroskedasticity. Is it ignorable? Perhaps, but let us try Poisson regression.

Poisson Model

Here, we easily perform Poisson regression and check the requirements. What was the code to run the analysis?

```
model.c <- glm(y~x, family=poisson)
summary(model.c)
```

That's all.

At this point, you should be able to guess what the lines do and what information in the first line is missing because it is the default link… as we see shortly.

Which is the best model of the three?

It may not be clear that these three **cannot be directly compared**. The linear model makes no adjustments, the log-transformed model does, as does the Poisson regression model. None of the transformations are the same. This means that we cannot use information criteria to compare the models.

So, how can we determine which of these models is best? A first step is to determine which is appropriate by checking the assumptions. The untransformed OLS model violates all three requirements. The transformed OLS model is heteroskedastic. The Poisson model, however, violates none of the requirements. Thus, on the basis of meeting assumptions, the third model is the best of these three.

If all we care about is estimates (and not confidence intervals), we could look at the graphic comparing the data and the estimations from the model (see the figure above). Numerically, we could also check how much the uncertainty in y has changed. The uncertainty using the null model (predicting $y = \bar{y}$) is 184.96. The uncertainty with the linear model is 43.6 — a reduction of 76%. The log-transformed linear model has an uncertainty of 8.1, which is a reduction of 96%. This is quite much different from the pure linear model. The Poisson model also has an uncertainty of 8.1 and a total reduction of 96%.

Thus, if all you care about is the estimate (which scientists should *not*), finding some adjustment so that the curve fits the data works. If you are a true scientist, then the confidence interval (and p-values) are important. This means assumptions about homoskedasticity are important — if they exist. Some modeling requires homoskedasticity others do not. Poisson regression does not (at least, not *really*).

Remember

Remember that we can use AIC, BIC, etc. *only when* the y-values are the same. This is not true here, as the y-values are all transformed differently.

This page titled 17.1: Linear or Poisson Regression? is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

17.2: The Mathematics

This section develops the full mathematical specification of a Poisson regression model for count data within the Generalized Linear Model (GLM) framework. It formally defines each of the three required components: the linear predictor, the conditional distribution of the response, and the link function. The section demonstrates that the Poisson distribution is the natural choice for unbounded count data, derives its key property that the expected value, and shows that its variance is inherently non-constant, explaining why ordinary least squares is inappropriate for such data. It then derives the log function as the canonical link for the Poisson distribution by writing the probability mass function in the exponential family form. Finally, it applies this framework to a real-world example analyzing the relationship between state population and the use of citizen initiatives, introducing the concept of influential points and foreshadowing the issue of overdispersion.

Learning Objectives

By the end of this section, you will be able to:

1. Specify the three mathematical components of a Poisson regression model for count data: the linear predictor $\eta = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$, the Poisson distribution for the response $Y \sim \text{Poisson}(\lambda)$, and the log link function $\log(\lambda) = \eta$.
2. Derive the expected value and variance of a Poisson random variable, and explain why this mean-variance relationship inherently violates the ordinary least squares assumption of constant variance.
3. Demonstrate that the log function is the canonical link for the Poisson distribution by rewriting its probability mass function in the standard exponential family form.
4. Fit a Poisson regression model in R using the `glm` function with `family=poisson(link="log")`, interpret the coefficient estimates, generate predictions by applying the inverse link (exponential function), and recognize when a data point may be influential.

✦•·····• ✦ •·····•✦

Count models have dependent variables that can take on only non-negative integers. Back in the time of OLS, we handled the non-negative aspect by taking the logarithm of the dependent variable (perhaps by adding 1 before taking the logarithm if there are values of 0). However, OLS does not allow for discrete dependent variables. The discrete aspect must be handled through Generalized Linear Models (GLMs).

Recall that using GLMs requires that we explicitly specify three things.

1. First, we need to know the linear predictor, η.
2. Second, we need to know the distribution of the dependent variable, conditioned on the independent variables.
3. Finally, we need to know the link function that appropriately connects the two of them.

The linear predictor is the same as always: the weighted sum of our independent variables. The canonical link function is the logarithm function. Finally, the distribution we will use is the Poisson distribution. Note that the Poisson is *not* the only option for such count dependent variables. The Negative Binomial distribution can also be used, but as the Negative Binomial distribution is a bit more complicated than the Poisson, we will motivate this chapter with the Poisson and save the Negative Binomial for later.

The Poisson Distribution

The Poisson distribution has the following probability mass function (pmf):

$$f(y;\ \lambda) = \frac{e^{-\lambda}\lambda^{y}}{y!} \qquad y \in \{0, 1, 2, 3, ...\} \tag{17.2.1}$$

Again, the probability mass function (pmf) is not as important as the expected value of this distribution. Why? Remember that the Generalized Linear Model paradigm models the *expected value*, $\mathbf{E}[Y \mid X]$, not the actual outcomes.

Calculating the expected value of the Poisson distribution is not as easy as it was for the Binomial; it requires a trick:

$$E[Y] \stackrel{\text{def}}{=} \sum_{y=0}^{\infty} y\, f(y) \tag{17.2.2}$$

$$= \sum_{y=0}^{\infty} y \frac{e^{-\lambda}\lambda^{y}}{y!} \tag{17.2.3}$$

$$= \sum_{y=1}^{\infty} y \frac{e^{-\lambda}\lambda^{y}}{y!} \tag{17.2.4}$$

$$= \sum_{y=1}^{\infty} \frac{e^{-\lambda}\lambda^{y}}{(y-1)!} \tag{17.2.5}$$

$$= \lambda \sum_{y=1}^{\infty} \frac{e^{-\lambda}\lambda^{(y-1)}}{(y-1)!} \tag{17.2.6}$$

Let us define $z \stackrel{\text{def}}{=} y - 1$. This leads to the next step:

$$E[Y] = \lambda \sum_{z=0}^{\infty} \frac{e^{-\lambda}\lambda^{z}}{z!} \tag{17.2.7}$$

Thus, the expected value of a Poisson random variable is $E[Y] = \lambda$.

Note

Recall that one of the assumptions of Ordinary Least Squares is that the variance is constant with respect to the (expected value of the) dependent variable. When the outcomes are distributed as Poisson random variables, we can actually prove that the variance is *not* constant with respect to the predicted outcomes. To see this, let $Y \sim Pois(\lambda)$. With this, and with the probability mass function above, we can use the definition to calculate the variance of Y. Without proof, the variance of Y is $V[Y] = \lambda$. Yes, the variance is *the same as* the expected value.

Thus, the variance is a function of the expected value, and the homoskedasticity requirement of OLS is violated.

That the variance is a function of the expected value also creates a problem. Quite often, we will be dealing with data in which the variance is *not* equal to, but is greater than, the expected value. Such data is termed **overdispersed**. When we encounter it, we will discuss what it means and what we should do.

By the way, this is the same overdispersion as we discussed in **the section on binomial overdispersion**. It is caused by the same structures and is tested in the same manner. I encourage you to revisit that section.

Now that we understand our choice of distribution a bit better, and the resulting expected value, let us examine the third facet: the link function. First, note that λ is bounded; $\lambda \in (0, \infty)$. Thus, we need a function that takes a bounded variable and transforms it into an unbounded variable. We have already met a link function that can handle this --- the logarithm function.

Note

Again, note that we are modeling $\lambda = E[Y]$, not the observed count. As λ is continuous and bounded below by zero (but never equal to zero), we can use the logarithm function as our transformation link.

And so, we have the three necessary components to use Generalized Linear Models for count data:

- the linear predictor,

$$\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \tag{17.2.8}$$

- the conditional distribution of the dependent variable,

$$Y \mid x \sim Pois(\lambda) \tag{17.2.9}$$

  with the formula for the expected value,

$$\mu = \lambda \tag{17.2.10}$$

- and the link function,

$$\log(\mu) = \eta \tag{17.2.11}$$

Here is what you need to take away from this section: The distribution must fit the possible outcomes. The link must translate the bounds on the parameter to the lack of bounds on the linear predictor. Both require you to know some distributions.

Deriving the Canonical Link

In the chapter on GLMs, we mentioned that each distribution has a canonical link. Let us derive the canonical link for the Poisson distribution. The steps to determine the canonical link are the same for the Poisson as it was for the Gaussian, Bernoulli, and Binomial:

1. Write the probability mass function (pmf).
2. Write the probability mass function in the required form.
3. Read off the canonical link.

For this distribution, this results in:

$$\text{pmf}: \quad f(y \mid \lambda) = \frac{e^{-\lambda}\lambda^y}{y!} \tag{17.2.12}$$

$$= \exp\left[\log\left(e^{-\lambda}\right) + \log\left(\lambda^y\right) - \log\left(y!\right)\right] \tag{17.2.13}$$

$$= \exp\left[-\lambda + y\log(\lambda) - \log(y!)\right] \tag{17.2.14}$$

$$= \exp\left[\frac{y\log(\lambda) - \lambda}{1} + -\log(y!)\right] \tag{17.2.15}$$

This is in the required form:

$$f(y) = \exp\left[\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\theta)\right] \tag{17.2.16}$$

Thus, reading off the standard form, we have the following:

- $y = y$
- $\theta = \log(\lambda)$
- $a(\phi) = 1$
- $b(\theta) = \lambda = \exp(\theta)$
- $c(y,\theta) = -\log(y!)$

As such, the canonical link is the log function. I leave it as an exercise to show $\mathbf{E}[Y] = \lambda$ and $\mathbf{V}[Y] = \lambda$ using the methods of the GLM mathematics section.

Example 17.2.1: Citizen's Initiative

The people in many US states have the ability to formulate binding laws by placing them before the people for a vote. This process is called the Citizens' Initiative. Extant theory suggests that states with a higher population will also use the initiative process more often than states with a lower population. Let us test this hypothesis with data (`crime` data file).

Solution.
As we are performing GLM modeling, we need to determine the three needed components. First, since the dependent variable is a *count* of the number of initiatives placed before the voters, we will assume that the dependent variable has a Poisson distribution:

$$\text{inituse} \mid \lambda \sim \text{Poisson}(\lambda) \tag{17.2.17}$$

The linear predictor will use our explanatory variable:

$$\eta = \beta_0 + \beta_1 \text{pop90} \tag{17.2.18}$$

The link function will be the logarithm function:

$$\log(\lambda) = \eta \tag{17.2.19}$$

With this, we use these commands to load and analyze the data

```
vcr = read.csv("http://rur.kvasaheim.com/data/crime.csv")

m2 = glm( inituse~pop90, family=poisson(link=log),
    data=vcr, subset=(ccode!=93) )
```

Now, `summary(m2)` tells us that there is a statistically significant relationship between the state's population in 1990 and its use of initiatives in the 1990s. Unfortunately, the relationship is negative ($\hat{\beta}_1 = -7.433 \times 10^{-8}$), which is definitely *inconsistent* with the original hypothesis. We have shown that the original hypothesis does not agree with this reality.

Let us now predict the number of initiatives that Utah would have had during the 1990s using the fact that the population of Utah is 1,722,850. We can do this by hand or we can use the `predict` function. In either case, we must remember to back-transform using the inverse of the logarithm function, the exponential function. Using the latter method gives me an *un-transformed* prediction of 2.0, which means the model predicts 7.44 initiatives for Utah in the 1990s. The real value is 3.

```
UTAH = data.frame(pop90=1722850)
```

```
prL = predict(m2, newdata=UTAH)
exp(prL)
```

The `glm` function used here includes an additional parameter that we have not discussed: `subset`. This parameter allows us to explicitly specify which data to include in the analysis. Here, I removed the state with `ccode` equal to 93 (California) from the analysis. The reason I did this is that a plot of the entire data suggested that California was an influential point.

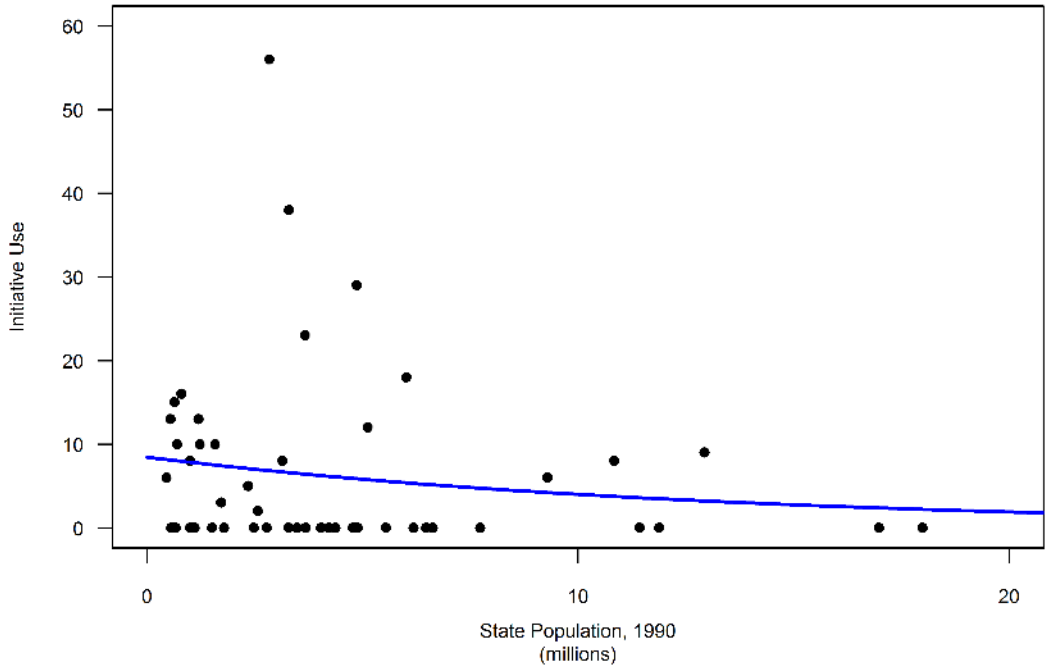


Figure **17.2.1**: A plot of initiative use against the population of the state in 1990, with the Poisson regression curve superimposed.

The interesting thing is that the graph visually calls into question the results of the GLM regression above. While the effect direction does definitely appear to be negative, it is hard to believe that this effect has such a high level of significance ($p \ll 0.0001$). There is a lot of variance in the data. What is happening?

The problem is that the model/data are *overdispersed*.

This page titled 17.2: The Mathematics is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

17.3: Overdispersion

This section addresses the common and important problem of overdispersion in count data regression. Overdispersion occurs when the variance of the observed counts exceeds the mean, violating a core assumption of the Poisson distribution (that the mean equals the variance). The section explains that ignoring overdispersion leads to underestimated standard errors and consequently inflated test statistics and misleading p-values. It then presents three practical solutions for handling overdispersion, ranging from a simple manual adjustment of standard errors to more statistically rigorous approaches: Maximum Quasi-Likelihood Estimation (MQLE) which estimates a dispersion parameter from the data, and the Negative Binomial regression model which explicitly models the extra variance through an additional parameter. The section compares these methods using a real-world example of initiative use across US states, demonstrating how each approach corrects the standard errors and affects conclusions about statistical significance.

Learning Objectives

By the end of this section, you will be able to:

1. Define overdispersion in the context of Poisson regression, recognizing that it occurs when the residual variance exceeds the mean and leads to underestimated standard errors, inflated test statistics, and invalid inference.
2. Calculate or estimate a dispersion factor from a fitted Poisson model and apply a manual adjustment to the standard errors to obtain more accurate p-values and confidence intervals.
3. Fit a quasi-Poisson model in R using the `quasipoisson` "family" in the `glm` function, understanding that this method estimates an additional dispersion parameter from the data while leaving the coefficient estimates unchanged.
4. Fit a Negative Binomial regression model using the `glm.nb` function from the MASS package, explaining that this distribution explicitly models the extra variance with a parameter (θ, theta) and is appropriate when overdispersion is present.

✦•·····• ✦ •·····•✦

Recall that one result of using the Poisson as our distribution of choice is that the residual variance and the expected value are assumed equal, because the probability mass function has $a(\phi) = 1$. As in the case of the Binomial models, overdispersion means that $a(\phi) > 1$.

In essence, this means $\mathbf{V}[Y \mid X] > \mathbf{E}[Y \mid X]$. For a Poisson model (and for a Binomial model), the overdispersion measure equals the ratio of the residual deviance to the residual degrees of freedom. We use the "residual," since overdispersion is a function of the model, as well as the data. For the Initiative Use model, the overdispersion factor is $681.68/48 = 14.2$. In other words, the level of unexplained variance is 14.2 times too high for this model.

Note

Please revisit the section on binomial overdispersion for one way of *testing* whether 681.68 really is evidence of significant overdispersion. If you create the right test, you will get a critical value of just 69. Since $681.86 \gg 69$, there is a lot of overdispersion in this data.

The code I used was `qchisq( c(0.025,0.975), df=48 )` .

Since overdispersion is a function of the model you are fitting to your data, the first solution is to determine if you are missing some important variables (or powers of variables). Frequently, modifying your linear predictor by adding appropriate variables will

reduce the overdispersion.

Even though this is the most appropriate method in many ways, there is an extreme danger to using this method: you may need to include *too many* variables and combinations of variables to eliminate the overdispersion. This results in over-fitting the data; that is, you are fitting the data and not the data-generating process in which we are actually interested.

Thus, if you end up including too many variables before the overdispersion is treated, you may want to consider other options.

The first is to adjust the standard errors by hand. This frequently works acceptably, as the primary effect of overdispersion is to underestimate the standard errors. The second option is to fit the model using a different fitting technique, one that allows you to use the Poisson but also allows you to have a different relationship between the mean and variance. Maximum quasi-likelihood estimation (MQLE) is a common alternative to the usual maximum likelihood methods (Wedderburn, 1974). Finally, you can fit the model using a different distribution, one that does not require the mean to equal the variance. The Negative Binomial is a common alternative to the Poisson.

Adjusting the Standard Errors

This first option adjusts the estimated standard errors to try to compensate for the overdispersion. Recall that the dispersion factor is the ratio of the residual variance to the expected variance. As the standard error is the square root of a variance, it would make sense that we could "fix" the overdispersion by *multiplying* the standard errors by the square root of the dispersion factor.

The table below presents the original standard error estimate along with the adjusted standard errors, z-values, and p-values. Note that the 1990 population was highly significant in the unadjusted model, but is not significant in the adjusted model ($p = 0.2578$).

Parameter	Estimate	Original StdErr	Adjusted StdErr	z-value
Intercept	2.136	0.0814	0.307	6.960
1990 Population	-0.00000007433	0.00000001743	0.00000006568	-1.131

The strength of this method is that it is easily performed. The drawback is that the correction is only an approximate estimate. In the era of expensive computational times, this method was commonly used; in the modern era of cheap computing, not-so-much. The next two methods are more appropriate in that their results are more statistically sound than this approximation.

Maximum Quasi-Likelihood Estimation

The maximum likelihood estimation method makes assumptions about the relationship between the mean and variance of the underlying distribution. For the Poisson distribution, that relationship is the identity function; that is, $E[Y \mid X] = V[Y \mid X]$. The presence of overdispersion indicates that this relationship — or this value of $a(\phi)$ — is incorrect. A different way of estimating the parameters is to use Maximum Quasi-Likelihood Estimation (a.k.a. quasi-maximum likelihood estimation). This method allows for modeling different relationships between the expected value and variance for the distribution. It effectively includes an additional parameter for $a(\phi)$.

The strength of using MQLE is that you can use the same distributions with which we are familiar, and the interpretation is identical. The weakness is that some statistical programs are not able to model using this method. Thankfully, R can. To model using MQLE in R, we prefix the distribution with the word `quasi` . Thus, we would use

```
glm(y~x, family=quasipoisson(link=log))
```

to fit this model. This command produces the results in the table above. Note that the coefficient estimates are the same as for the Poisson model. The difference is in the standard errors — they are increased. This reduction causes our z-values to decrease, resulting in increased p-values.

Parameter	Estimate	Original StdErr	Adjusted StdErr	z-value
Intercept	2.136	0.0814	0.345	6.103
1990 Population	-0.00000007433	0.00000001743	0.00000007492	-0.992

Caution

The only two distributions that have the MQLE option in R are the Poisson (`quasipoisson`) and the Binomial (`quasiBinomial`). These are the only two popular distributions that have a specific numeric value for $a(\phi) = 1$. The rest have a value for $a(\phi)$ that can be estimated from the data; for instance, the Gaussian has $a(\phi) = \sigma^2$.

The Negative Binomial Family

In the Generalized Linear Model framework, you need to select an appropriate distribution that matches your dependent variables. If that variable is a count, then the sole requirements for that distribution are that the outcomes can only be discrete and non-negative. The Poisson is the usual distribution, but it is not the only one. An alternative distribution is the Negative Binomial. The Negative Binomial family allows for both over- and under-dispersion in the model. It does this by assuming the rate parameter λ in the Poisson is distributed as a Gamma random variable. Specifically, it assumes

$$Y \mid \mu, \theta \sim NegBin(\mu, \theta) \tag{17.3.1}$$

$$\text{where} \quad Y \mid W \sim Pois(\mu W) \tag{17.3.2}$$

$$\text{with} \quad W \sim \frac{1}{\theta} Gamma(\theta) \tag{17.3.3}$$

This results in $E[W] = 1$ and $V[W] = 1/\theta$.

With this formulation, it can be shown that $E[Y] = \mu$, $V[Y] = \mu + \mu^2/\theta$, and that the probability mass function for Y is

$$f(y; \theta) = \frac{\Gamma(\theta + y)}{\Gamma(\theta)\ y!} \frac{\mu^y \theta^\theta}{(\mu + \theta)^{\theta + y}} \tag{17.3.4}$$

The strength of this formulation is that a greater number of variations are able to be fit.

The drawback is that interpreting the results is a bit more difficult. However, since we make the computer do all the heavy lifting, this drawback is minor. It does, however, introduce a new set of possible error messages and parameters that you may have to interpret.

The other drawback is that the Negative Binomial distribution is *not* a member of the exponential family (unless θ is known, which it is not). As such, it cannot be used within the GLM paradigm (strictly speaking). With that said, fitting a model using the Negative Binomial distribution is just as easy as it is for any of the previous distributions.

In R, you will have to load the `MASS` package to use the Negative Binomial family, since it has its own regression function: `glm.nb` . The options for `glm.nb` are similar to those for `glm` .

Thus, the command

```
m2n = glm.nb(inituse~pop90, data=vcr, subset=(ccode!=93))
```

will perform Negative Binomial regression similar to the regression performed in the MQLE section. The first thing to notice is that the overdispersion is no longer relevant. With this, we can have more confidence in the parameter estimates (provided in the table above). The second thing to notice is that the effect of population is still no longer statistically significant. This agrees with our observation in the adjusted standard errors and MQLE sections. Finally, we notice that there are additional parameters estimated (at the bottom). The Theta is the estimated value of θ in the Gamma distribution above.

	Estimate	StdErr	z-value	p-value
Constant Term	2.2376	0.4835	4.63	<< 0.0001
Population in 1990	-0.00000010091	0.000000081903	-1.23	0.2179

This model estimates that Utah will have had approximately 7.9 initiatives during the 1990s. I leave it as an exercise to determine this.

Note

The *direction* of the coefficient estimate is still directly comparable to the other coefficients estimates we have examined. The magnitudes are also comparable, but only to the other log-linked models. Thus, this model tells us that there is a negative relationship between the state's population and the level of initiative use (although it is not statistically significant).

That the direction is comparable is due to choosing a link function that is strictly *increasing*. That the estimates are comparable is due to having the same link function or same transform function.

This page titled 17.3: Overdispersion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

17.4: Body Counts

This extended case study applies count regression models to a substantive research question: whether terrorist violence by the Ruritanian Republican Army (RAvŘ) during the Troubles was influenced by the political ideology of the sitting prime minister. Using data on annual deaths, days each prime minister was in power, and a measure of conservatism, the section illustrates the practical considerations of modeling count data. It introduces the concept of an offset variable — a term included in the model with a fixed coefficient of 1 — to control for varying exposure (here, the number of days in power) rather than treating it as a standard predictor. The analysis compares Poisson, quasi-Poisson, and negative binomial models, highlighting the critical issue of overdispersion and its impact on statistical significance. The section demonstrates how the choice between these models can affect conclusions about the effect of conservatism and shows how to explore more complex functional forms (polynomials and interactions) while remaining mindful of issues like multicollinearity.

Learning Objectives

By the end of this section, you will be able to:

1. Explain the role of an offset variable in a count regression model, recognizing that it is included with a fixed coefficient (typically 1) to account for varying exposure (e.g., time, area, population) and should be specified using the `offset()` function in R.
2. Fit and compare Poisson, quasi-Poisson, and negative binomial regression models to count data, using measures like overdispersion, AIC, and the statistical significance of key predictors to evaluate model appropriateness.
3. Interpret the results of these models, understanding that the choice of model family (especially when overdispersion is present) can substantively affect conclusions about the significance and direction of predictor effects.
4. Construct and interpret polynomial and interaction terms in a count model to explore more complex relationships, and create prediction plots to visualize the estimated effects of predictors across different model specifications.

✦•·····• ✦ •·····•✦

Using the above information, let us examine the problem of understanding terrorism. This extended example will also allow us to discuss a few things that are becoming important to our analyses, namely the bias-variance trade-off.

Example: The Ruritanian Republican Army (RAvŘ)

The Troubles in Ruritania lasted from 1969 until 2002. In that time, over 1800 people died as a result of terrorist actions — both republican and loyalist groups. Six prime ministers — both left- and right-leaning — had to deal with the terrorism. If we assume that the terrorist groups are rational actors, then they will act to maximize their chances of achieving their goals. Because of its hierarchical structure and large size, the Ruritanian Republican Army (RAvŘ) was best able to organize its actions to affect the elections.

The question is whether they did —

Did the RAvŘ react to the political ideology of the prime minister?

Unfortunately, the extant literature is divided on the direction of the effect. Some research suggests that the RAvŘ became more violent and killed more people when the Conservatives held power. Other research suggests that the RAvŘ became more violent under the Left party. Which is it?

The `terrorism` data set contains just three variables of import: `total` (the total number of deaths under that prime minister for the year, or part of the year), `days` (the number of days during the year that the prime minister was in power), and `riteleft` (the level of conservatism of the prime minister). The second variable is necessary to control for the fact that some prime ministers only ruled for a part of the year. The third variable is the research variable. The first variable is the response variable (dependent variable). The basic research model is

$$\text{deaths} \sim \text{riteleft} \tag{17.4.1}$$

However, we need to deal with `days`, the number of days the premier is in power. If we include `days` as a simple independent variable, we allow the effects of the `days` variable to freely vary to fit the data.

It is almost always better to treat `days` as the divisor for terrorist killings, thus ostensibly creating a variable of `killings per day`. But, this is no longer a count model (non-integer values), nor is it a proportion model (values can be greater than one). What should we do?

> Fear not! Through the magic of mathematics, we can handle it.

Recall in the section on the mathematics of count models that the link function we used was the logarithm: $\log[\lambda] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$. If, instead of the expected count, λ, we wanted to model the expected ratio, λ/days, we would have:

$$\log\left[\frac{\lambda}{\text{days}}\right] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \tag{17.4.2}$$

$$\log[\lambda] - \log[\text{days}] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \tag{17.4.3}$$

$$\log[\lambda] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \log[\text{days}] \tag{17.4.4}$$

As such, we now have a count model (the $\log[\lambda]$ is alone on the left and is a random variable) with an additional factor ($\log[\text{days}]$ on the right as a non-random variable). Note that *there is no parameter* to estimate for $\log[\text{days}]$. This is important in how we set up the model, as `days` is not a typical variable. Let us call it an *offset* variable.

Offset variables do not have parameters to estimate. They are direct effects with no multipliers. One can think of them as being subsumed in the constant term (which would be true if the offset variable were constant). Most statistical programs have an offset option available when you specify the model to be fit. In R, the offset is specified in the model call by the keyword `offset`.

For the `glm` function,

```
glm(pira~riteleft, offset=log(days), data=terror)
```

For `glm.nb`,

```
glm.nb(pira~riteleft, offset(log(days)), data=terror)
```

Option 1: Days as an independent variable

The first option is to treat the `days` variable as just another independent variable. This is not the best answer, as `days` has a specific meaning with respect to the number of terrorist deaths. The better option is to use Option 2 (below). However, for pedagogical purposes, let us first enter `days` as an independent variable. Performing regressions for each of the three count data families, we get the summarized results in the table below.

Table **17.4.1**: Results of three different "families:" Poisson, quasi-Poisson, and Negative Binomial. The numbers in parentheses, below the coefficient estimates, are the standard errors.

	Poisson		Quasi-Poisson		Negative Binomial	
Constant Term	2.2622	(0.1190)	2.2622	(0.7071)	2.0363	(0.6004)
Conservatism	-0.0115	(0.0011)	-0.0115	(0.0065)	-0.0142	(0.0093)
Days Served	0.0050	(0.0004)	0.0050	(0.0021)	0.0058	(0.0019)
AIC	1482.8		------		369.5	

Note that the direction of each of the effects is the same. This is not always true, especially when the variable has little effect or has no statistical significance. However, if the variable is significant *and* changes effect direction, then there is something severely wrong with your research model. Also note that the effects are the same between the Poisson and the quasi-Poisson families. The only difference is the size of the standard errors. The quasi-Poisson will *always* give a better estimate of the standard errors (and of the statistical significance) than the Poisson.

Note that the Poisson model is severely overdispersed — the residual deviance is much larger than the residual degrees of freedom (the residual deviance is 1298, the residual degrees of freedom is 36, the overdispersion factor is 36.06). As such, the Poisson family would be (very) inappropriate for this model. Thus, either the quasi-Poisson or the Negative Binomial model would be preferable.

If we had just used the Poisson family, we would have concluded that the level of conservatism of the prime minister is *highly* significant. However, looking at the more-appropriate results of fitting using Maximum Quasi-Likelihood Estimation (or using the Negative Binomial family), we see that the effect of conservatism is non-existent. Since the effect of conservatism on deaths was the purpose of this research question, it is extremely important to reach good conclusions about the effects of this variable.

As our research variable is not statistically significant at the usual level of significance, we will not even bother to predict and graph our predictions here.

Option 2: Days as an offset variable

The second (and preferred) option uses `days` as an offset (or "exposure") variable. This makes more sense than allowing it to freely enter the model as a typical independent variable. The results from fitting the data with the three model families are found in the table below.

Table **17.4.2**: Results of three different families: Poisson, quasi-Poisson, and Negative Binomial. The numbers represent the estimated coefficients. The numbers in parentheses are the estimated standard errors.

	Poisson		Quasi-Poisson		Negative Binomial	
Constant Term	-1.8280	(0.0254)	-1.8280	(0.1495)	3.8744	(0.0969)
Conservatism	-0.0106	(0.0011)	-0.0106	(0.0063)	-0.0069	(0.0041)
AIC	1479.6		------		2080.2	

According to the results, the Poisson family is not appropriate; the level of overdispersion is very high — on the order of 35. As such, using the MQLE method or the Negative Binomial family would make good substitutes. In the quasipoisson model, the parameter estimates remain the same, but the estimates of the standard errors change to reflect the overdispersion. Thus, while the effect of conservatism was statistically significant in the Poisson model, it was not in the quasipoisson model ($p = 0.1013$).

The Negative Binomial model echoes the qualitative conclusions of the quasipoisson: The level of conservatism has no statistically discernible effect on the level of deaths resulting from RAvŘ terrorism in Ruritania during the Troubles ($p = 0.0905$).

Bettering the Fit

Using the results from both the quasi-Poisson and the Negative Binomial model does offer you the ability to strengthen your conclusions. If one result gave statistical significance and the other did not, then you would realize that your conclusions depended on the assumptions you made about the underlying mechanism that produced the data, and not on the variables you chose to include (or exclude). It is never a good place to find yourself when your substantive results depend on the choice between two acceptable models.

Maybe, one should not stop here. Our formula is rather simplistic: it states that one independent variable is all we need to explain the dependent variable. It also assumes that the effect is linear between the independent and the dependent variable. If we believe that *extremist* prime ministers suffer from higher (or lower) levels of terrorist killings, then the research formula we have cannot capture that effect. To capture *that* effect, we will have to use the square (and/or higher powers) of the `riteleft` variable.

In fact, let us examine the effects of conservatism (up to the fourth power), *plus* the effects of having Labour in power, *plus* an interaction between having Labour in power and the level of conservatism in the Labour government. Thus, the research model we wish to fit will be

$$\begin{aligned} \texttt{pira} = \beta_0 + \beta_1 \texttt{riteleft} + \beta_2 \texttt{riteleft}^2 + \beta_3 \texttt{riteleft}^3 & \quad (17.4.5)\\ + \beta_4 \texttt{riteleft}^4 + \beta_5 \texttt{labour} & \quad (17.4.6)\\ + \beta_6 \texttt{labour} \times \texttt{riteleft} & \quad (17.4.7) \end{aligned}$$

Of course, we would need to have good theory to provide this model, but let's just have fun with this.

In most statistical programs, one would have to create new variables for each of the powers (three new variables) and a new variable for the interaction term ($\texttt{labour} \times \texttt{riteleft}$). In R, however, we can just write the formula to reflect what we want without having to worry about the additional step of creating new variables. As such, in R, the formula will be

$$\begin{aligned} \texttt{pira} \sim \texttt{riteleft + labour + I(riteleft}^2\texttt{) +} & \quad (17.4.8)\\ \texttt{I(riteleft}^3\texttt{) + I(riteleft}^4\texttt{) +} & \quad (17.4.9)\\ \texttt{I(labour*riteleft)} & \quad (17.4.10) \end{aligned}$$

The use of `I()` indicates that R should evaluate what is in the parentheses as a new variable. Fitting this model using Maximum Quasi-likelihood Estimation indicates that none of the terms have a statistically significant effect. This should not really surprise us, since there is a lot of correlation among the independent variables in that model. In the presence of high correlation, the standard errors tend to be larger than they should be.

Table 17.4.3: Results of two different models: fitting with MQLE and using the Negative Binomial family. The numbers are the parameter estimates; in parentheses, the estimated standard errors.

	Quasi-Poisson	Negative Binomial
Intercept	-12.51 (4.478)	-6.980 (2.396)
Labour	-4.742 (1.553)	-4.8430 (0.2856)
Conservatism	1.847 (0.07101)	1.8660 (0.3778)
Conservatism^2	-0.03830 (0.01425)	-0.03833 (0.00750)
Conservatism^3	-0.002585 (0.0009421)	-0.0026070 (0.0005005)
Conservatism^4	-0.00007314 (0.00002642)	-0.00007361 (0.00001398)

Since nothing was statistically significant, let us pare the model to reduce the effect of correlation and get at some more basic effects. The best first thing to remove from the model is the interaction term. Doing this gives us the research model:

$$\texttt{pira} = \beta_0 + \beta_1\ \texttt{riteleft} + \beta_2\ \texttt{riteleft}^2 + \beta_3\ \texttt{riteleft}^3 \tag{17.4.11}$$
$$+ \beta_4\ \texttt{riteleft}^4 + \beta_5\ \texttt{labour} + \epsilon \tag{17.4.12}$$

Fitting this model using both the quasi-Poisson family and the Negative Binomial family gives us the results in Table 17.4.3 above.

Notice that all of our variables are now statistically significant at the $\alpha = 0.05$ level. It turns out that the interaction term was so highly correlated with the other variables that it made it impossible to correctly estimate the effects of the individual research variables.

Now that we have two models that tell us, substantively, the same story, we should *show* the effect of the variables of interest. There are really only two independent variables involved here, with one being dichotomous. As such, we can show the effects on the same graph (one graph for each family), with two prediction curves per graph. The figure below shows the predictions from both the quasi-Poisson model (Left Panel) and the Negative Binomial model (Right Panel). The upper curve in both cases (red) corresponds to predictions when the Conservatives are in power.

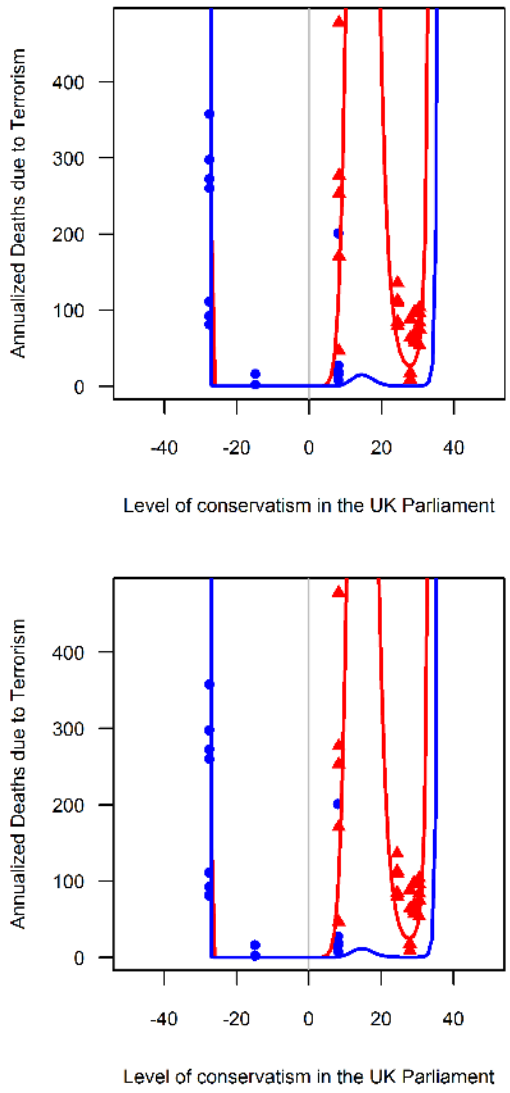


Figure 17.4.1: Plot of the number of deaths due to terrorism, caused by the Ruritanian Republican Army (RAvŘ), in Ruritania during the Troubles. The points are overlaid with the quasi-Poisson model (Left Panel) and the Negative Binomial model (Right Panel). In both cases, the upper curve (red) corresponds to the prediction when the Conservative Party is in power.

This page titled 17.4: Body Counts is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

17.5: The Bias-Variance Tradeoff

Note that these two models are completely worthless in *explaining* the effects of the variables on the population (or the "data generating process"). Because we used so many parameters, the model fits the data — noise and all — as opposed to the underlying reality (signal). This is a common problem. Since the goodness of our fit *increases* as we increase the number of variables in our models (see the effect of the number of covariates on the R^2 value), there is a pressure for us to increase the number of variables. However, as in this case, using too many variables (or interactions, or powers) usually tells us too little about the underlying *process* that gave rise to the data, which is the entire purpose of performing a statistical analysis.

Caution

Remember that we are only using the data (a sample) to help us better understand the process (model) that gave us the data (population). Fitting the data perfectly actually tells us little about the process we are trying to model. However, not using enough variables may not get at the process, either. This trade-off between increasing the number of variables (which increases the reliance of the parameter estimates on the actual data) and reducing the number of variables (which increases the errors in our model) is termed the Bias-Variance trade-off, and it is a problem we must keep in our minds at all times. On the one hand, we want a good model that fits the population, on the other hand, we only know the sample (the data collected).

In the terrorism example (see the section on terrorism), we can see that we used too many explanatory variables in our model. A glance at the graphs in the figure suggests that we should have gone with a quadratic model (second power) *at most*, even though the quartic model (fourth power) fit the data better. Avoiding over-fitting the data is as simple as being aware of the dataset and the model predictions (of course, a good graph helps).

This page titled 17.5: The Bias-Variance Tradeoff is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

17.6: Conclusion

In this chapter, we examined what we can do when our dependent variable is an unbounded count variable. As such variables are non-negative and discrete, nothing we have done thus far can properly handle them. While performing a log transform of the dependent variable as we did in Chapter 8: Fixing the Violations would allow us to actually make predictions that made sense (provided that there were no zero counts), the resultant model would probably violate one or more of the assumptions of the Classical Linear Model.

Two model families were introduced to handle count data. The Poisson family requires that the mean and the variance be equal (which translates to the residual deviance and the residual degrees of freedom be equal). This is rarely the case. When the residual variance is much larger than the mean, the data are overdispersed. The Negative Binomial family models overdispersed (and underdispersed) data, but it is a bit more difficult to fit with data.

As with Generalized Linear Models in general, the methods in this section model the expected value and not the actual outcome. As the parameters must be non-negative, we use a log link to ensure this condition holds. Note that we are *not* transforming the dependent variable, we are transforming the family parameter (or parameters) — λ, in the case of the Poisson and the quasi-Poisson; λ and θ for the Negative Binomial.

The last point of this chapter was a warning about the **Bias-Variance trade-off**: Including more variables fits the *data* better, not necessarily the *process* that gave rise to the data. Fewer variables may miss both the data and the underlying process. There is a happy medium — unfortunately, we cannot know what it is.

This page titled 17.6: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

17.7: End-of-Chapter Materials

R Functions

In this chapter, we were introduced to several R functions that will be useful in the future. These are listed here.

Packages

- `MASS`
  This package is another "book package –- a package created for a specific book. Here, that book is "Modern Applied Statistics with S," by William N. Venables and Brian D. Ripley (2004).

Statistics

- glm(formula)
  This function performs generalized linear model estimation on the given formula. There are three additional parameters that can (and often should) be specified.
  - The `family` parameter specifies the distributional family of the dependent variable. This chapter we looked at `poisson` and `quasipoisson`.
  - The `link` parameter specifies the link function for the distribution. If none is specified, the canonical link is assumed. For the Poisson, that canonical link is the logarithm
  - Finally, the `data` parameter specifies the data from which the formula variables come.

- `glm.nb(formula)`
  As Negative Binomial regression is fit using different methods, it cannot be included in the base `glm` command. To use the `glm.nb` command, you must include the (very helpful) `MASS` package in your script, `library(MASS)`. The output of the `glm.nb` function is similar to that of the normal `glm` command, with the inclusion of an estimate for θ and its standard error. If $\theta = 1$, then the Poisson model may be appropriate.
  - `offset`
    The offset function (or function parameter) allows us to include known varying values in our regression. The variable included as an offset will not have an effect parameter estimated for it.

- `predict(model, newdata)`
  As with almost all statistical packages, R has a predict function. It takes two parameters, the model, and a dataframe of the independent values from which you want to predict. If you omit `newdata`, then it will predict based on the independent variables of the data itself, which can be used to calculate residuals. The dataframe must list all independent variables with their associate new values. You can specify multiple new values for a single independent variable.

Exercises

1. Show that $E[Y] = \lambda$ and $V[Y] = \lambda$ using the methods of Section 14.2: The Requirements for GLMs.
2. The citizens' initiaitive example mentioned that California was an outlier in this model. First, plot the `initiative` data with California included. Second, appropriately fit the model with California included and interpret the coefficients. Finally, predict the number of initiatives Utah would have (a population of 1,722,850).
3. In Section 17.3: Overdispersion, we fit the `initiative` data using the Negative Binomial distribution. I made the statement that this model predicted 7.9 initiatives for Utah in the 1990s. Please graph the data, plot the prediction curve, and predict the number of initiatives Utah will have in the 1990s. Finally compare the results between the model with California and the model without California.

4. Estimate the number of initiatives that Utah had during the 1990s.
5. Given the probability mass function in Section 17.3: Overdispersion , prove $\E[Y] = \mu$ and $V[Y] = \mu + \mu^2/\theta$.
6. Given the definition of the Negative Binomial distribution, prove that an overdispersion of $\theta = \infty$ reduces the Negative Binomial to a Poisson.

Applied Readings

- Richard Berk and John M. MacDonald (2008). "Overdispersion and Poisson Regression ." *Journal of Quantitative Criminology*. 24(**3**): 269–284.
- M. Katherine Hutchinson and Matthew C. Holtman (2005). "Analysis of Count Data using Poisson Regression." *Research in Nursing & Health*. 28(**5**): 408–418.
- Dana Loomis, David B. Richardson, and L. Elliott (2005). "Poisson Regression Analysis of Ungrouped Data." *Occupational and Environmental Medicine*. 62(**5**): 325–329.
- Katarina A. McDonnell and Neil J. Holbrook (2004). "A Poisson Regression Model of Tropical Cyclogenesis for the Australian–Southwest Pacific Ocean Region." *Weather & Forecasting*. 19(**2**): 440–455.
- Ron Michener and Carla Tighe (1992). "A Poisson Regression Model of Highway Fatalities." *American Economic Review*. 82(**2**): 452–456.
- Marta N. Vacchino (1999). "Poisson Regression in Mapping Cancer Mortality." *Environmental Research*. 81(**1**): 1–17.
- Weiren Wang and Felix Famoye (1997). "Modeling Household Fertility Decisions with Generalized Poisson Regression." *Journal of Population Economics*. 10(**3**): 273–283.
- Lisa A. White (2009). *Predicting Hospital Admissions with Poisson Regression Analysis*. Masters Thesis. Naval Post-Graduate School.

Theory Readings

- Kurt Brannas (1992). "Limited Dependent Poisson Regression." *Journal of the Royal Statistical Society*, Series D (The Statistician). 41(**4**): 413–423.
- A. Colin Cameron and Pravin K. Trivedi. (1998) *Regression Analysis of Count Data*. New York: Cambridge University Press.
- Edward L. Frome (1981). "Poisson Regression Analysis." *The American Statistician*. 35(**4**): 262–263.
- Jie Q. Guoa and Tong Li (2002). "Poisson Regression Models with Errors-in-Variables: Implication and treatment." *Journal of Statistical Planning and Inference*. 104(**2**): 391–401.
- Alexander Kukush, Hans Schneeweis, and Roland Wolf (2004). "Three Estimators for the Poisson Regression Model with Measurement Errors." *Statistical Papers*. 45(**3**): 351–368.
- Alfonso Palmer, J. M. Losilla, J. Vives, and R. Jiménez (2007). "Overdispersion in the Poisson Regression Model: A comparative simulation study." *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences*. 3(**3**): 89–99.
- Tsung-Shan Tsou (2006). "Robust Poisson Regression." *Journal of Statistical Planning and Inference*. 136(**9**): 3173–3186.
- William N. Venables and Brian D. Ripley (2004). *Modern Applied Statistics with S*, 4th edition. New York: Springer.
- Rainer Winkelmann (2000). *Econometric Analysis of Count Data*. New York: Springer.

- Liming Xiang and Andy H. Lee (2005). "Sensitivity of Test for Overdispersion in Poisson Regression." *Biometrical Journal*. 47(**2**): 167–176.
- Feng-Chang Xie and Bo-Cheng Wei (2009). "Diagnostics for generalized Poisson regression models with errors in variables." *Journal of Statistical Computation & Simulation*. 79(**7**): 909–922.

This page titled 17.7: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

18: Nominal and Ordinal Dependent Variables



This chapter concludes our exploration of dependent variables in regression analysis. Having previously examined continuous, count, and binary outcomes, we now turn our attention to the final major category: categorical dependent variables.

Categorical variables describe qualities or characteristics that place observations into distinct groups. This chapter specifically addresses two fundamental types:

- **Nominal variables**: Categories without any inherent order (e.g., species of pet, brand of car, type of cuisine).
- **Ordinal variables**: Categories with a meaningful sequence or ranking (e.g., Likert-scale satisfaction ratings, education levels, disease severity stages).

The critical distinction between these types is not merely semantic; it has direct and significant implications for statistical modeling. Applying a technique designed for one type to the other can lead to inefficient models, loss of statistical power, or fundamentally incorrect interpretations. For instance, treating an ordered rating scale as a simple set of unordered labels discards valuable information about progression, while forcing arbitrary numeric values onto unordered categories can produce misleading results.

Therefore, the core objective of this chapter is to equip you with the appropriate regression frameworks for each scenario. We begin by extending the principles of logistic regression to handle nominal outcomes with more than two categories, focusing on how to model the probability of each category relative to a baseline. We then advance to specialized models for ordinal outcomes, which uniquely leverage the inherent ordering to provide more parsimonious and insightful results (Oooh! A GRE word!!).

✦•·····• 😻 •·····•✦

One of the most pervasive research questions in Political Science is to predict a person's vote based on demographic information. In other words, if you know a person's age, gender, income, education, and religion, how well can you predict how that individual will vote in the upcoming election?

At first glance, this question appears to be a binary dependent variable problem. After all, there are only two parties, right? Well, *even* if you ignore third parties, there is a third option: abstention. In each Ruritanian parliamentary election, a sizable number of registered voters decide not to vote. For instance, in the 2016 election, while Kuzněcov (of the royalist Král a Země party) received 48% of the vote cast and Ivanović (of the republican Republikánská Strana) received 46%, a full 45.3% of the eligible voters did not vote. Thus, the distribution of votes in this election is 26.3% Kuzněcov, 25.2% Ivanović, 3.2% other, and 45.3% none of the above. As such, conclusions based on those models that assume a binary outcome have definite issues with generalization to the voting public at large. They are ignoring important information.

A better alternative is to specifically add in "abstention" and model the three possible outcomes at once (or "abstention" and "other" and model the four). Such a regression model is called a **nominal regression model** or a multinomial regression model, because there is no inherent ordering among the levels of the dependent variable.

There is a second type of dependent variable that is closely related to the nominal case – the ordinal dependent variable. The difference between the nominal and the ordinal is that the ordinal has more information contained in it. There is no ordering in the nominal case, whereas there *is* an implicit ordering in the ordinal case. Examples of ordinal variables include ratings and indices.

If we just use our logistic regression methods (Chapter 15: Binary Dependent Variables), we come up with some odd results. If we force a nominal variable into just two categories, we lose information in the data. If we treat ordinal dependent variables simply as nominal, information is also lost. If we treat them as continuous, our conclusions may not match reality.

Thus, both nominal and ordinal dependent variables need their own modeling methods. This chapter examines how to model both the nominal dependent variable and the ordinal dependent variable more properly.

Caution

Statistics emphasizes both estimating the value (expected value) *and* the variance of that estimate (confidence interval).

Note

This chapter sits uneasily here. From the standpoint of the dependent variable type, this is its proper place. However, these are not generalized linear models (GLMs). They are particular expansions to the GLM paradigm. As such, if you are looking at the GLM modeling method as being the unifying theme to this part of the book, this chapter should not exist.

But, it does.

· · — · ✶ · — · ·

Chapter Sections

18.1: Nominal Dependent Variables
18.2: Ordinal Dependent Variables
18.3: Cattle Feed
18.4: The State University of Ruritania
18.5: Conclusion
18.6: End-of-Chapter Materials

This page titled 18: Nominal and Ordinal Dependent Variables is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

18.1: Nominal Dependent Variables

This section introduces the challenges and methods for modeling a nominal dependent variable — a categorical variable where the categories have no inherent ordering, such as occupation type or brand choice. It begins by framing nominal regression as a direct extension of the binary logistic regression model, but notes that simply running separate binary logits for each category leads to probabilities that sum to more than one. The solution lies in the multinomial distribution, which generalizes the binomial to handle multiple outcome categories with the key constraint that all category probabilities must sum to one. The section then demonstrates a complete multinomial logistic regression analysis using General Social Survey data to predict occupation from race, education, and experience. It walks through preliminary data examination, model fitting in R using the `nnet` package, interpretation of coefficients relative to a chosen baseline category, and calculation of predicted probabilities that properly sum to one.

Learning Objectives

By the end of this section, you will be able to:

1. Define a nominal dependent variable and explain why the binary logistic regression approach fails when extended directly to multiple categories, as the probabilities for each category, calculated separately, will not sum to one.
2. State the five requirements of the multinomial distribution and recognize it as the appropriate probability model for nominal outcomes, generalizing the binomial distribution to cases with J possible outcomes per trial.
3. Fit a multinomial logistic regression model in R using the `multinom` function from the `nnet` package, specifying the model formula and interpreting the output, which provides coefficients for each category relative to a user-selected baseline category.
4. Generate predicted probabilities for each outcome category for a given set of predictor values, confirming that these probabilities sum to one, and evaluate model fit using measures such as accuracy and the Akaike Information Criterion (AIC).

✦•·····• ✦ •·····•✦

A nominal variable is a categorical variable where **there does not exist a meaningful ordering** in the categories. Examples may include job type, presidential vote (and non-vote), and beer brand choice. These variables are categorical — not numeric — and the categories have no inherent ordering. White Collar is not "greater than" Professional. Voting Monarčista is not "more than" voting Republikán. Widmer is not "more than" Coors.

How do we model such dependent variables?

> Of course, there *may be* a time when you are predicting Republikán vote by examining an underlying level of conservatism. In such a case, Monarčista–Republikán would be ordered. Thus, it really depends on what you are modeling (as always).

There are a couple of ways of doing this. The first, easiest, and most understandable is to model the variable as a series of binary dependent variables. We already understand how this works, the testing of the model is already conceptually understood, and it works (not *really*, but **close**?). There are just a couple things to clarify.

Caution

Usually. Nothing in statistics *always* is best. As you have seen by now, there are always methods that work better, but with trade-offs. The science here is to be aware of the strengths with the weaknesses and balance them to get closer to the true process you are trying to model.

The Multinomial Distribution

As with the simply binary dependent variable case, let us layout the mathematical background to the nominal dependent variable case. As in the binary dependent variables case, we are actually modeling the underlying probabilities of each of the outcomes. Also, as in the binary case, there are five requirements for the random variable to follow a Multinomial distribution (*cf.* Section 22.4: Probability Distributions):

1. the number of trials, n, is known;
2. each trial has J possible outcomes;
3. the success probability for each trial, $\{\pi_1, \pi_2, \ldots, \pi_J\}$, is constant;
4. each trial is independent from the others; and
5. the random variable is the number of each type of outcome in those n trials.
6.

Thus, if we let π_j be the probability that category j is selected, then the following two conditions must hold:

$$0 < \pi_j < 1 \qquad \text{for all } j \in \{1, 2, \ldots, J\} \tag{18.1.1}$$

$$\sum_{j=1}^{J} \pi_j = 1 \tag{18.1.2}$$

Condition 18.1.1 must hold because we are dealing with probabilities bounded by 0 and 1, and Condition 18.1.2 holds because one of the J possible outcomes **must** happen. In the binary case, our two probabilities were $\pi_1 = \pi$ and $\pi_2 = 1 - \pi$, which satisfies the second condition by default and the first because it makes no sense to study phenomena that always or never occurs.

When we generalize the binary case, we need to select an appropriate probability distribution — one that can model J possible outcomes with J different probabilities. That distribution is called the multinomial distribution.

Recall that the distribution in the **bi**nary case was the **bi**nomial distribution.

The **probability density function** for the multinomial distribution in the general case is

$$f_{\mathbf{X}}(\mathbf{X}) = \frac{n!}{x_1! x_2! \cdots x_J!} \pi_1^{x_1} \pi_2^{x_2} \cdots \pi_J^{x_J} \tag{18.1.3}$$

Here, x_i are non-negative integers and $\sum x_j = n$. The expected value of this distribution for a specified outcome is

$$E[X_j] = n\pi_j \tag{18.1.4}$$

and the covariance between two outcomes is

$$Cov[X_i, X_j] = -n\pi_i\pi_j \tag{18.1.5}$$

Multinomial Outcomes

Be aware that $\mathbf{X}$ is a vector. So, if $n = 1$ and $J = 4$, the following could be outcomes from the Multinomial distribution:

$$x = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}; \quad x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}; \quad x = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{18.1.6}$$

In the first example, a 2 came up; in the second, a 1; in the third, a 3. Note that in each case, the sum of the entries is n and the number of entries is J.

Now, if $n = 4$ and $J = 3$, the following could be outcomes from a Multinomial distribution:

$$x = \begin{bmatrix} 0 \\ 3 \\ 1 \end{bmatrix}; \quad x = \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix}; \quad x = \begin{bmatrix} 0 \\ 0 \\ 4 \end{bmatrix} \tag{18.1.7}$$

In the first example, three 2s and a 3 came up; in the second, two 1s, a 2, and a 3 came up; in the last, four 3s came up.

> Be aware that the sum of the entries in each outcome vector is n and that the number of entries is J.

If the random variable $\mathbf{X}$ follows a Multinomial distribution with $n = 3$ and $\pi = [0.1, 0.5, 0.4]'$, then we could write it as

$$\mathbf{X} \sim Multi\left(n = 3,\ \pi = \begin{bmatrix} 0.1 \\ 0.5 \\ 0.4 \end{bmatrix}\right) \tag{18.1.8}$$

and the expected value of $\mathbf{X}$ would be

$$E[\mathbf{X}] = \begin{bmatrix} 0.3 \\ 1.5 \\ 1.2 \end{bmatrix} \tag{18.1.9}$$

The expected value of X_3 would be $E[X_3] = 1.2$.

Note

Make sure you see that this is just an extension of the Binomial distribution, where

$$f_X(x) = \frac{n!}{x!(n-x)!}\,\pi^x(1-\pi)^{n-x} \tag{18.1.10}$$

with

$$\mathbf{X} = \begin{bmatrix} x \\ n-x \end{bmatrix} \quad \text{and} \quad \pi = \begin{bmatrix} \pi \\ 1-\pi \end{bmatrix} \tag{18.1.11}$$

Using the Multinomial

Let us look at how to use the Multinomial distribution for modelling with two examples.

Example 18.1.1: Rolling Three Dice

Let us illustrate the multinomial distribution with a typical "rolling a die example." Assuming that the die is fair, then the probability of rolling each of the six outcomes is $1/6$. If we roll a fair die 3 times, what is the probability the outcome is $[1,0,1,0,0,1]'$ (that is, a 1, a 3, and a 6 come up)? What is the expected value of $\mathbf{X}$?

Solution.

This is a multinomial experiment. There are a fixed number of possible outcomes (six), the probabilities of each outcome are constant (they do not change as we roll the die), and the probabilities sum to one. As such, we know the probability mass function is

$$\underbrace{= \frac{3!}{x_1!\, x_2!\, x_3!\, x_4!\, x_5!\, x_6!} \left(\frac{1}{6}\right)^{x_1} \left(\frac{1}{6}\right)^{x_2} \left(\frac{1}{6}\right)^{x_3} \left(\frac{1}{6}\right)^{x_4} \left(\frac{1}{6}\right)^{x_5} \left(\frac{1}{6}\right)^{x_6}}_{f(\mathbf{x})} \tag{18.1.12}$$

Thus,

$$P\left[\mathbf{X} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}\right] = \frac{3!}{1!\,0!\,1!\,0!\,0!\,1!} \left(\frac{1}{6}\right)^{1} \left(\frac{1}{6}\right)^{0} \left(\frac{1}{6}\right)^{1} \left(\frac{1}{6}\right)^{0} \left(\frac{1}{6}\right)^{0} \left(\frac{1}{6}\right)^{1} \tag{18.1.13}$$

$$= 6\left(\frac{1}{6}\right)^{3} \tag{18.1.14}$$

$$= \frac{1}{36} \tag{18.1.15}$$

Thinking through the problem should get us to the same point.

Finally, we know the expected value is

$$E[\mathbf{X}] = n\pi = 3 \begin{bmatrix} 1/6 \\ 1/6 \\ 1/6 \\ 1/6 \\ 1/6 \\ 1/6 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{bmatrix} \tag{18.1.16}$$

■

Estimating π_j

As we have a formula for our expected value, we have our mechanism for estimating the several π_j: in an experiment (or set of data), count the number of times outcome j occurred and divide by the total number of trials (or records). This is actually the maximum likelihood estimator for π_j. Thus, our linear predictor is

$$\text{logit}(\pi_j) = \beta_{j,0} + \beta_{j,1}x_1 + \beta_{j,2}x_2 + \cdots + \beta_{j,k}x_k \tag{18.1.17}$$

Notice that this linear predictor has $k+1$ parameters to estimate for each of the j categories. Thus, you will need more than $j(k+1)$ pieces of data to fit it. There are ways to reduce the dimensionality of the problem (reduce the number of parameters in need of estimation); however, these are beyond the scope of this book.

We need the logit link (or something just like it) to force our linear predictions to be in the range $\pi_j \in (0,1)$. As any link that maps $g : \mathbb{R} \to (0,1)$ is acceptable, we could use the log-log link, the complementary log-log link, the probit link, or any of an infinite

number of others... in theory. As before, the choice of the link function is largely a matter of tradition. If you deviate from tradition, the burden of proof is on you to justify the selection. Furthermore, the differences are usually slight. If the differences are large, then there is something wrong with your research model. Because of this, it would behoove you to fit your research model using a couple different (appropriate) link functions to help determine the stability (robustness) of your results.

> Thus, there are two things that you need to take away from this discussion: First, we are able to fit the entire model at once because we have a distribution that can produce the necessary nominal results. Second, we model the underlying probabilities (like in the binary case), not the actual outcomes, as usual.

Modeling Nominal Variables

To see all of this in action, let us look at an extended example.

Example 18.1.2: Modeling Occupation

The General Social Survey (GSS) at the University of Chicago conducts an extensive survey of adult Americans every year. The data is freely available from NORC. In this *very* small subset of the data, `gssocc` , I would like to predict a person's occupation category (`occ`) based on race (`white`), years of education (`ed`), and years of experience (`exper`).

Representative Sample?

Before getting started, let us examine the variables involved. The race variable is binary, with a `1` representing the person identifying as "white" and a `0` otherwise. As a side note, this is a race variable, not an *ethnicity* variable. Thus, Hispanics may self-identify as either white or not-white. Also note that this is a *self-identification* variable; that is, the individual being surveyed decided his or her reported race. Looking at a frequency count, a full 91.69% of the respondents stated they were white. This is *significantly higher* than the population at large, where approximately 80% of Americans were white when the survey was conducted. When we do the final analysis, we need to keep this in mind, as it is not necessarily representative of the nation as a whole.

The median number of years of education in the sample is 12 years, which corresponds to graduating from high school. The mean number of years is 13.09, which indicates the sample is right skewed (the Hildebrand ratio is +0.37). Furthermore, it is interesting to note that 51.0% of the sample only graduated from high school. Additionally, 23.4% of the sample received a bachelor's degree or more (which is close to the population 27% who have received a bachelor's degree or higher). Finally, 18.7% of the sample did not graduate from high school, which is close to the 15% estimate of the population. From this, it appears as though the sample is representative of the population in terms of educational attainment.

Note

This was a safe assumption with respect to the `educ` variable, but not with respect to the `white` variable. As such, it needs to be mentioned that you are unable to check the representativeness of the experience variable.

The third independent variable is the years of experience in the job. There are no general statistics for the population, so we will have to make a large assumption that the sample represents the population. In the sample, the years of experience varies widely, from 2 to 66 years. The median is 17 years and the mean is 20.5 years. Thus, the sample is *also* right skewed. This makes sense as

this is a count variable. Count variables tend to be right skewed as they cannot take on negative values. In fact, there is nothing in the distribution of the experience variable that looks wrong. With that said, however, one still needs to mention the caveat that we cannot tell if this is representative of the population.

Correlations

Looking at the correlations amongst the independent variables can help us avoid any unpleasantness and surprises due to collinearity and multicollinearity. The correlation matrix below does not show any significant hint of multicollinearity. In fact, this correlation matrix suggests that these three variables are *effectively* independent of each other. Pearson's product-moment correlation test does indicate that the correlation between education and experience is statistically significant at the $\alpha = 0.05$ level ($t = -5.2152, df = 335, p \ll 0.0001$), but that just means we have sufficient evidence that $\rho \neq 0$. The coefficient of -0.2740 is a low level of correlation.

	White	Education	Experience
White	1.0000	0.0243	-0.0794
Education	0.0243	1.0000	-0.2740
Experience	-0.0794	-0.2740	1.0000

Nominal Regression Thoughts

Now, let us model the outcome variable with the three independent variables. Actually, we need to step back and really think about what we mean by "model the outcome." Do I want to predict the probability that a person will be Blue Collar given the input variables? Or: Do I want to predict the job category given the input variables? **These are different questions**. They require slightly different methods.

Option 1: Blue Collar vs. All Others

The first question actually asks a binary question: What is the probability that a person will be Blue Collar (compared to *all* of the other job categories)? This is very much like the questions asked in Chapter 15: Binary Dependent Variables. Here, the dependent variable takes on values `1` (Blue Collar) and `0` (not Blue Collar).

To answer *this* question, we need to create a variable called `bluecol` as an indicator variable for Blue Collared-ness. Thus, the model we fit will be

$$\texttt{bluecol} \sim \texttt{white} + \texttt{ed} + \texttt{exper} \tag{18.1.18}$$

We would fit it using a generalized linear model, a binomial family, and a logit link. The results of the regression are in the table here. From this model, we can perform all of the goodness of fit measures from Chapter 15: Binary Dependent Variables.

	Estimate	Std.Err	z-value	p-value
Intercept	3.1036	1.0110	3.07	0.0021
White	0.7090	0.6213	1.14	0.2538
Years of education	-0.3721	0.0640	-5.81	0.0000
Years of experience	-0.0259	0.0113	-2.30	0.0215

Looking at the results from running the model, we see that greater levels of education and greater levels of experience are associated with a *lower probability* of being a blue collar worker. For Bob, an individual who responded that he was white, had 20

years of education, and 10 years of experience in their current job, the probability of being a blue collar worker is approximately 2% (as compared to not being a blue collar worker).

Option 2: All vs. All

The second option is to try to estimate Bob's actual occupation type, not just whether he is Blue Collar. Note that answering this question is not as simple as running five logistic regressions. This is because the last parenthetical part in the previous section, "as compared to not being a blue collar worker", is subtle and extremely important.

Note

Here is why:

What is the probability that Bob is a white collar worker? If we do the same steps above, we get that the probability that Bob is a white collar worker (as compared to not being a white collar worker) is 13.1%. Similarly, if we continue performing separate logistic regressions, the probability that Bob is a professional is 96.9%; menial, 2.3%; and craft, 7.9%.

Note that all of these probabilities add up to **more than 100%**.

If we are trying to determine Bob's occupation, then there is something seriously wrong here, since the probability that Bob holds one of these five job types cannot be greater than 100%.

The problem is that we **kept changing the base category** in our probability calculations. In Chapter 15: Binary Dependent Variables, we never mentioned the need to specify the base category since it always defaulted to the opposite of what we were modeling. In other words, we were actually measuring the probability of an event as compared to the probability of "not the event" (a.k.a. the odds of the event). This ensured that the probabilities always added up to 100%.

The Lesson

Comparing probabilities of events is not as easy as when we were only working in the binary realm. It is doable — easily so, with one small change. We need to select a base category that **does not change** throughout our analysis. The choice is up to you, as all choices are equally acceptable from a statistics standpoint.

Practically speaking, to perform this modeling, you will have to load the `nnet` package. Since this comes with your base distribution of R, there is no need to install it. Once loaded with the `library(nnet)` command, to fit the better model, use the R command

```
multinom(occ ~ white + ed + exper)
```

Because of the large amount of output, the regression table is structured slightly different. The coefficients (in logit units) and the standard errors are still presented. The statistical significance (p-values) is not. However, a quick rule of thumb is that the variable is statistically significant (at the $\alpha = 0.05$ level) if the parameter estimate is more than twice the standard error. The table below presents the output from modeling the data in the form given in the output.

```
Coefficients:
               Constant     White   Educ Level   Experience
Craft           -1.8328   -0.7642       0.1933       0.0230
```

```
Menial           -0.7412   -1.2365       0.0994      -0.0074
Prof            -12.2595    0.5376       0.8783       0.0309
WhiteCol         -6.9800    0.3349       0.4526       0.0299


Std. Errors:
                Constant     White   Educ Level   Experience
Craft             1.1861    0.6324       0.0775       0.0126
Menial            1.5195    0.1996       0.1023       0.0174
Prof              1.6681    0.7996       0.1005       0.0144
WhiteCol          1.7144    0.9340       0.1023       0.0153
```

Note that one of the five job types is missing: Blue Collar (the **base** level). This is because all of the probabilities are measured *with respect to* Blue Collar. Thus, these percentages are directly comparable (after transforming from logit units).

The `Coefficients` part of the regression table provides you with four separate equations. For instance, the above says that the Craft equation is

$$\text{logit}\left(\frac{\texttt{Craft}}{\texttt{BlueCol}}\right) = -1.8328 - 0.7642\texttt{White} + 0.1933\texttt{Educ} + 0.0230\texttt{Experience} \tag{18.1.19}$$

That is rather tedious to calculate by hand. R is nice in that if you `predict` on a multinomial model, it will give you the category with the highest probability, by default. Thus, according to this model, Bob will most likely be a Professional (which was our conclusion above). If we want the probabilities for each of the possible job types for Bob, we need to add a `type="probs"` parameter to our function call:

```
predict(model.mn1, newdata=BOB, type="probs")
```

Such a call gives us the following probabilities (which sum to one, as they should):

```
  BlueCol   Craft   Menial     Prof   WhiteCol
   0.0020  0.0091   0.0020   0.9565     0.0304
```

Interpreting Estimates

The interpretation of the coefficients (parameter estimates) is the same as for the binary dependent variable case. Just remember that the coefficients are in "logit units," so you will want to speak of log-odds. So, for instance, interpreting the 0.3349 coefficient for White Collar worker, I would say something like

> Those identifying as White have a 33.49% increase in the odds of being a Whilte Collar worker than those who do not.

When interpreting the output from a `predict` function, however, you do not need to back-transform your predictions; they are already in probabilities/proportions. To remember this, just look at the output — it is in proportions already (a quick check is that they sum to one).

Goodness of fit

The first check of the goodness of the model is the relative accuracy (see also Section~???). The accuracy is the number of correct predictions divided by the number of cases. The relative accuracy divides this number by the accuracy of always selecting the modal category (\textbf{the null model}). For this dataset, the modal category is Professional, with 140 out of 337 cases belonging to Professionals. Thus, the relative accuracy is $\frac{169}{337} / \frac{140}{337} = 1.207$. Thus, this model improves accuracy by 21% over the null model. Is this good? It depends on your other models.

Also, using the basic accuracy score means that you are asserting that your data are representative of the population. Without this assertion, the accuracy means virtually nothing!

As Maximum Likelihood Estimation is used, the Akaike Information Criteria (AIC) score is also reported. For this model, $AIC = 885$. Is this good? Again, **it depends on your other models**. In other words, model comparison needs another model. I leave it as an exercise to see that the null model has $AIC = 1027$. Thus, our model is much better than the null model.

Interpreting the Model

Now that we have looked at our model, let us look at the parameter estimates. According to our model, Whites have a higher probability of being Professionals and White Collar workers than they are to be Craft or Menial laborers. As for education, higher levels of education are associated with higher odds of being a Professional or a White Collar worker (both of these are statistically significant) than being a Blue Collar worker. Finally, years of experience are not a statistically significant predictor of job type, as none of the coefficients are statistically significant (coefficient / standard error > 2).

So, we have a picture of Professionals and White Collar workers, when compared to Blue Collar workers: they are White and well educated. Not an earth-shattering conclusion, but it is encouraging to see that our conclusions do seem to reflect reality.

Note on the Rule of Thumb

This rule of thumb comes from the fact that in a Normal distribution, the ratio needs to exceed 1.96 to be statistically significant at the $\alpha = 0.05$ level. These parameter estimates are not guaranteed to be Normally distributed. As such, the rule of thumb is to be more conservative.

Even with the rule of thumb, do not bet the farm. 

This page titled 18.1: Nominal Dependent Variables is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

18.2: Ordinal Dependent Variables

This section introduces methods for handling ordinal dependent variables, where the categories have a meaningful order but the distances between them are unknown. Examples include movie ratings (one star to five stars) or survey responses like Strongly Disagree to Strongly Agree. The section outlines four possible approaches, ranging from treating the ordinal variable as nominal (simple but inefficient) to pretending it is continuous and using OLS (simple but potentially misleading). It focuses on the most statistically sound approach: assuming an underlying continuous latent variable is being measured, and that we only observe which threshold-defined category the latent value falls into. This method, known as ordinal regression or proportional odds logistic regression, estimates both the regression coefficients for the latent variable and the threshold values separating the categories. A worked example using GSS data on warmth of feeling toward President Obama demonstrates how to fit such a model in R using the `polr` function from the `MASS` package, interpret the coefficient estimates and thresholds, and generate predicted probabilities for each outcome category.

Learning Objectives

By the end of this section, you will be able to:

1. Identify an ordinal dependent variable and describe at least four potential approaches for modeling it: treating it as nominal, modeling cumulative probabilities, assuming an underlying continuous latent variable with thresholds, or treating the categories as continuous and using OLS.
2. Explain the logic of the latent variable approach to ordinal regression, where an unobserved continuous process ($\boldsymbol{Y^* = XB + E}$) is assumed, and the observed ordinal category is determined by which of several threshold values (τ_1, τ_2, τ_3, etc.) the latent variable falls between.
3. Fit an ordinal regression model in R using the `polr` function from the `MASS` package, specifying the appropriate formula and (optionally) the link function (e.g., logistic for proportional odds).
4. Interpret the output of an ordinal regression model, including the coefficient estimates (which describe effects on the underlying latent variable) and the estimated threshold values (which separate the ordinal categories), and generate predicted probabilities for each outcome category for a given set of predictor values.

✦•·····• ✦ •·····•✦

Another variety of categorical dependent variables is ordinal. A variable is **ordinal** if it is *both* categorical *and* the categories have an underlying order to them. Examples include movie ratings (number of stars), hurricane intensity, and so forth.

There are actually *at least* four ways of handling ordinal dependent variables:

1. **Treat them as nominal.** This allows us to fit ordinal data using previous techniques. Unfortunately, it is inefficient as it ignores important aspects of the data itself.
2. **Treat their cumulative level as nominal.** If the ordinal variable takes on values 1 – 5, then create nominal variables corresponding to Level 1, Levels 1 and 2, Levels 1–3, Levels 1–4, and Levels 1–5. This preserves much of the underlying information *and* allows us to fit it using a previous method.
3. **Assume that there is an underlying continuous process** that you wish to fit. The ordinal nature is just several threshold values along the possible values. This reduces to a pseudo-OLS, where you also need to fit the threshold values, not just the slopes and intercepts. Using Maximum Likelihood methods, this is trivial to solve.
4. **Pretend that the ordinal values are continuous** and fit it using ordinary least squares or one of its offsprings. This has the advantage of being easily fit.

Three of these ways have already been discussed, and you are quite adept at using them (Options 1, 2, and 4). Only the third option is completely new to you. This chapter focuses on how to fit Option Three.

Option Three

Let us assume that there is an underlying continuous process. We only experience (observe) this process through the ordinal variable. This is very similar to how we first looked at binary variables: underlying process exhibited only in the 0/1 outcomes. Here, there is more than just the one threshold (which traditionally defaulted to 0.500). Thus, we have two sets of parameters to fit. The first is the parameters which describe the process (the βs). The second is the position of those threshold values (the τs).

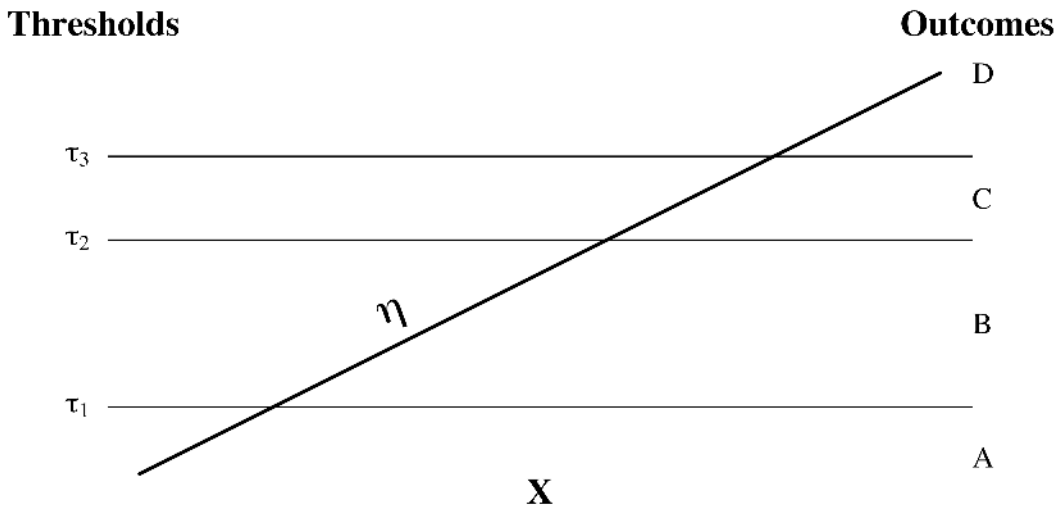


Figure 18.2.1: Schematic diagram of the thresholding process. The line represents the linear continuous process. The τs represent the threshold values. A, B, C, and D represent the ordinal outcomes.

Without going into the details, we will use Maximum Likelihood Estimation as our fitting method because it has many nice properties. Thus, our underlying process is

$$\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \tag{18.2.1}$$

Our thresholding process is illustrated in the figure above. The line represents the underlying continuous process that you are trying to model. The A, B, C, and D represent the observed ordinal values. The threshold values, τ_1, τ_2, and τ_3 are the values of η that separate the observed ordinal values.

This model is very straight forward and understandable. Using R to obtain the fitting is also straight forward. The results presented are also relatively straight forward.

Example 18.2.1: Warmth for Obama

Let us use some more data from the GSS. This data explores the "warmth of feeling" the respondent has for President Obama. The demographic information is the gender (`male`), the race (`white`), the age, and the number of years of education (`ed`). The response variable has four **ordered** levels: Strongly Disagree (SD), Disagree (D), Agree (A), and Strongly Agree (SA). Our goal is to explain a person's feelings toward the president based solely on demographic information.

Solution.

Let us fit this data with ordinal regression. The function in R is `polr` , which stands for "proportional odds logistic regression"

(although the probit is an option as a link function). This function requires the **MASS** package. Thankfully, since **MASS** also comes with the base distribution of R, there is no need to install it, only to load it via the `library(MASS)` command.

The actual command to fit this model using ordinal regression is

```
polr( warm~male + white + age + ed )
```

Table **18.2.1**: Result of ordinal regression in R. Note that the women tend to view President Obama in a more favorable light; whites, less; older, less; and higher educated, more. All of these agree with multiple surveys throughout his tenure as President.

		Value	Std. Error	t-value
Variables:				
	Woman	0.743	0.078	9.50
	White	-0.400	0.118	-3.39
	Age	-0.020	0.0024	-8.17
	Years of Education	0.098	0.013	7.52
Thresholds:				
	SD | D	-1.700	0.237	-7.18
	D | A	0.111	0.233	0.48
	A | SA	1.979	0.236	8.37

This command will give the coefficients of the underlying linear regression and the threshold values separating the four categories. From the table, we see that the equation for the underlying linear process is

$$\eta = 0.743\mathtt{Woman} - 0.400\mathtt{white} - 0.020\mathtt{age} + 0.098\mathtt{ed} \tag{18.2.2}$$

The thresholds are also listed. The threshold between Strongly Disagree and Disagree is at $\tau_1 = -1.700$. The threshold between Disagree and Agree is $\tau_2 = 0.111$. The threshold between Agree and Strongly Agree is $\tau_3 = 1.979$. Thus, to calculate our prediction, we calculate the prediction based on the linear model, η, and compare that value to the intervals described by the thresholds. Thus, for Bob, who is Male, White, 40 years old and has 20 years of education, we have

$$\eta = 0.740 \times 0 - 0.400 \times 1 - 0.020 \times 40 + 0.098 \times 20 = 0.76 \tag{18.2.3}$$

As $\eta = 0.76$, we have our prediction that Bob agrees with the president. If we actually want probabilities that Bob Strongly Disagrees, Disagrees, Agrees, or Strongly Agrees, we would have to back-transform using the inverse of the logit function and calculate each probability using integral calculus… or we could just ask the computer to do it for us:

```
BOB = data.frame(male="Men", white="White", age=40, ed=20)
predict(model.ol1, newdata=BOB, type="probs")
```

This gives the probabilities as

```
    SD       D       A      SA
0.0785   0.263   0.429   0.229
```

Thus, it is far from certain that Bob agrees (or strongly agrees) with the president, although that probability is rather high: $0.429 + 0.229 = 0.658$.

Accuracy

Finally, let us look at the accuracy of the model. I leave it as an exercise to show that the relative accuracy is 1.105, which indicates that the model is about 10.5% better than the null model (the modal category is "Agree"). This is not a fantastic increase in accuracy, but we do know how certain demographics feel about the president: Whites tend to disagree, Males tend to disagree, older people tend to disagree, and lesser educated people tend to disagree.

Of course, we could have added in a quadratic education term to the model to see if both the more-educated and the less-educated both support the president. I also leave this as an exercise to show that there is no evidence of this. Thus, we have no evidence that the relationship between education and presidential support is anything other than linear.

This page titled 18.2: Ordinal Dependent Variables is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

18.3: Cattle Feed

This section presents a practical application of multinomial logistic regression for a nominal dependent variable. Using a dataset on cattle, the goal is to predict which brand of feed a cow received based on its ranch, age, and weight. The section walks through the complete workflow: loading and exploring the data, fitting a multinomial model using the `nnet` package in R, and interpreting the results. It highlights how exploratory analysis, such as cross-tabulations, can reveal perfect predictors (like ranch) that dominate the model. The section then demonstrates how to use the fitted model to predict the feed type probabilities for a new cow (a "mysteryMoo") and, crucially, how to create informative graphics that visualize the effect of a continuous predictor (weight) on the predicted probabilities for each outcome category, making the model's implications clear and actionable.

Learning Objectives

By the end of this lengthy example, you will be able to:

1. Fit a multinomial logistic regression model in R using the multinom function from the `nnet` package for a nominal dependent variable (feed type) with multiple categorical and continuous predictors (ranch, age, weight).
2. Interpret the coefficient estimates from a multinomial model, recognizing that they are estimated relative to a baseline category (Accuration in this example) and that extremely large coefficients can indicate perfect or near-perfect predictors revealed in exploratory cross-tabulations.
3. Generate predicted probabilities for each outcome category for a new observation (a "mystery cow") using the predict function with `type="prob"`, and identify the most likely category.
4. Create and interpret a visualization that plots the predicted probabilities for each feed type against a continuous predictor (weight), holding other predictors constant, to understand how the likelihood of each outcome changes across the range of that predictor.

✦•·····• ✦ •·····•✦

Example: Cattle and their Feed

Previously, we attempted to model the weight of cattle based on a few factors. Let us try something different. Let us predict the brand of food used by the cattle based on the ranch, age, and weight.

Specifically, let's first model feed type. Then, let's say that the RUR ranch sent a 21-year-old cow to slaughter at 1197 pounds. Which food brand was most likely used? What are the probabilities of each brand being used?

Solution.

Since the food brand is a nominal variable, we will use multinomial regression. The data file is `cattleData`. Let's load it and look at some summary statistics on it. In other words, let's **know our data**.

```
library(nnet)
cowz = read.csv("http://rur.kvasaheim.com/data/cattleData.csv")
attach(cowz)

summary(cowz)

cor.test(weight,age)
table(ranch,feedType)
```

Note that there is (as expected) a strong correlation between age and weight. If we are doing model selection, we will need to keep this in mind as this multicollinearity will decrease the statistical significance of those two variables. Note from the cross-tabulation that the EVA ranch only used Purina and the TCL ranch only used Rangeland (in this sample). That fact would make it really easy to predict the feed type for those ranches. The other ranches use a combination of all of the brands.

With this information to guide us, we fit the nominal regression model:

```
cowModel = multinom(feedType ~ weight + ranch + age)
summary(cowModel)
```

The first line fits the model. Note that the model did converge, so we can pay attention to the results. If it had not converged, we should first change the link function, then realize that the multicollinearity is a problem. Dropping one or more variables would be an appropriate action in that case.

- The results of the `summary(cowModel)` command gives some insight into the relationships. First, note that the coefficient estimate for `ranchEVA` for estimating `Purina` is 20.37. This is extremely high, meaning it is *almost guaranteed* that a cow from the EVA ranch will used Purina.
  But, from the cross-tabulation above, we already knew this.
- Similarly, the coefficient estimate for `ranchTCL` for Rangeland is a huge 22.32. This indicates a cow from the TCL ranch will most certainly use Rangeland food. Again, we knew this from our cross-tabulation.
- Note from the regression table that `Accuration` is missing. All feed measurements are taken *with respect to that level*. This is important to keep in mind if we do this by hand. It is just something to note if we are using the computer to do our calculations.

So, let's estimate the food used by our mystery cow. First, let's define it:

```
mysteryMoo = data.frame(weight=1197, age=21, ranch="RUR")
```

Now, let's predict the probabilities it used each of the feed types:

```
predict(cowModel, mysteryMoo, type="prob")
```

The results tell us that the mystery cow most likely used Steakmaker. In fact, the probability it used Steakmaker was 79%. The second most likely feed type was Accuration (13%).

Graphics

Let us talk about graphics for a bit. A two-dimensional scatter plot looks at two numeric variables. We can therefore easily plot a prediction curve when dealing with only a single dependent variable and single independent variable. If there is a second independent variable, we can plot several curves, one for each level in that second independent variable. Once we move beyond two independent variables, graphics are more difficult to do. A simple regression model like the one above may require dozens of graphics to illustrate each aspect.However, we can simplify things by focusing on only a couple independent variables at a time. The choice depends on the story you are trying to learn (or tell).

Graphic: Feed Type versus Weight

For this first graphic, I am consciously making the decision to plot the **predicted probability** on the y-axis, the **cattle weight** on the x-axis, and have a prediction curve for each feed type. This will allow me to see the **effect of weight on the predicted food type**. This means I need to select values for the other two independent variables. For the numeric age, I would typically use its mean or median, whichever was the "typical" age for these cattle.

For the selected value of the ranch, I would either select the ranch to which the mystery cow belonged (to continue *that* story) or the most popular ranch (to try to *generalize* the story). It is best to do separate graphics for all ranches so that you, the researcher, can better understand the effect of ranch on the probabilities. It is always better to do more to understand.

So, here is the code to create the predictions:

```
theWeights = seq(1019,1579, length=1e4)
theAge = median(age)
prRUR = predict(cowModel, newdata=data.frame(weight=theWeights, age=theAge, ranch="RUR
```

The `prRUR` variable now contains 10,000 rows (one for each weight) and 5 columns (one for each feed type). The entries are the probabilities.

Now, we just plot the data and these predictions:

```
par(family="serif", las=1)
par(xaxs="i", yaxs="i")
par(mar=c(4,4,0,0)+0.5)
par(cex.lab=1.2, font.lab=2)

plot.new()
plot.window( xlim=c(1000,1600), ylim=c(0,1))

axis(1); axis(2)
title(xlab="Weight [lb]")
title(ylab="Probability at RUR Ranch")

lines(theWeights,prRUR[,1], col=1) # Accuration
lines(theWeights,prRUR[,2], col=2) # Purina
lines(theWeights,prRUR[,3], col=3) # Rangeland
lines(theWeights,prRUR[,4], col=4) # Steakmaker
lines(theWeights,prRUR[,5], col=5) # Wind and Rain

legend("topright", bty="n", col=1:5, lwd=2,
    legend=c("Accuration","Purina","Rangeland","Steakmaker","Wind and Rain")
    )
```

Note that this graphic includes a legend that lets the reader know which probability curve belongs to which feed type. Legends are rather important to include on a graphic. Remember that graphics should be stand-alone with their caption. Because a legend contains so much information, it requires a large function. To see all a legend can do, run `?"legend"` in R.

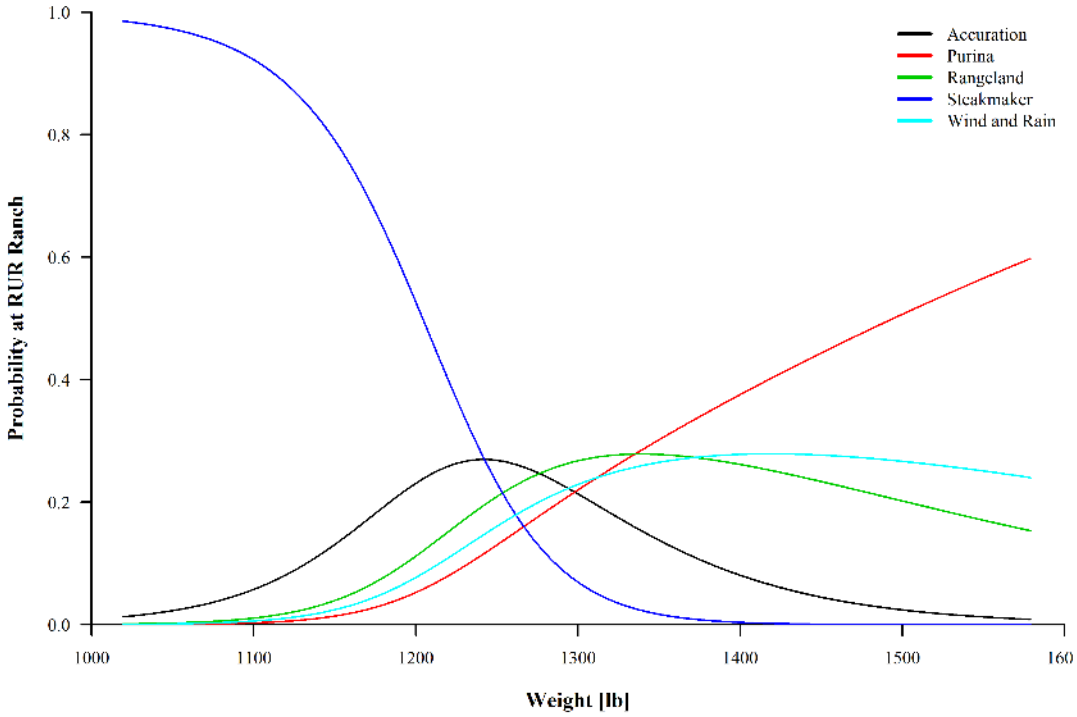


Figure **18.3.1**: Graphic of the probability for each feed type at the RUR ranch. The probabilities vary with the cow's weight. The age is held at the median, 21.

The figure above is the resulting graphic. Note that the predicted feed type tends to be either Steakmaker, for light cows, or Purina, for heavy cows. When the weight of the cow is middling, there is great uncertainty in which feed type it used.

It is interesting that this analysis gives us additional insight on how we can create big cows for slaughter. This suggests we should use Purina brand. This conclusion, however, is based only on the RUR ranch and a middle-aged cow. More importantly, this conclusion assumes that the data are representative of the population of interest. As this data was originally collected in conjunction with a dissertation in Animal Science, I tend to think it is representative.

From a strictly statistical standpoint, the additional insight is limited. However, if you are hired by RUR ranch to determine the best feed type, this graphic would be *very* persuasive for you, the decision-maker. While we could create a similar graphic for feed type against age, I am not convinced it would be helpful. Age is not something one would like to optimize like weight. In other words, I am not sure what story I would tell about it.

Important!

Don't make graphics just for fun. Make sure you create them knowing how to interpret them.

This page titled 18.3: Cattle Feed is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

18.4: The State University of Ruritania

This section presents a practical case study in modeling an ordinal dependent variable, specifically a student's class standing (Freshman, Sophomore, Junior, Senior) at the State University of Ruritania. Using the `polr` function from the `MASS` package in R, it demonstrates how to fit a proportional odds logistic regression model with gender and grade point average (GPA) as predictors. The section walks through the crucial step of properly defining the ordinal nature of the response variable in R, interpreting the model summary output, and comparing the model's AIC to a null model to assess improvement. It then shows how to generate predicted probabilities for each class level for a new individual (Eliska) and discusses the uncertainty inherent in these predictions. Finally, it explores strategies for visualizing the results of an ordinal model when both numeric and categorical predictors are present.

Learning Objectives

By the end of this section, you will be able to:

1. Prepare an ordinal dependent variable for analysis in R by using the `ordered()` function to define the correct sequence of levels, ensuring the model respects the natural ordering of the categories.
2. Fit a proportional odds logistic regression model using the `polr` function from the `MASS` package, specifying an ordinal response and one or more predictors (both categorical and continuous).
3. Interpret the model summary output, including coefficient estimates and their signs, and compare the AIC of the fitted model to that of a null model to evaluate whether the predictors improve model fit.
4. Generate predicted probabilities for each outcome category for a new observation using the `predict()` function with `type="prob"`, and explain how to graphically represent the model results by plotting predicted probabilities against a continuous predictor, accounting for categorical predictors by using separate lines or facets.

✦•·····• ✦ •·····•✦

Example: Eliska in School

Previously, we modeled the grade point average of students at the State University of Ruritania (Státní Univerzita v Ruritánii). Let us turn this around and model the student's class (Freshman, Sophomore, Junior, Senior) given only their reported gender and the current GPA of the student.

Let us also predict the class of Eliska, a female student with a 3.33 GPA.

Solution.

As usual, the first step is to import the data and look at a summary, including a cross-tabulation of our categorical independent variable and the dependent variable:

```
library(MASS)

suvrData = read.csv("http://rur.kvasaheim.com/data/suvr.csv")
summary(suvrData)
```

Let us pause here.

Note that the `class` variable is an **ordinal variable**. We need to let R know this, along with the order:

```
suvrData$class = ordered(suvrData$class, levels=c("Non-Matriculated", "Freshman", "Sop
summary(suvrData)
```

There we go, the levels for the `class` variable are in the right order.

Let's continue.

```
attach(suvrData)
table(gender, class)
```

Note that none of the non-matriculated students are female. This is just something to know and remember as we get results.

Now, we can fit our model and look at the summary results:

```
suvrModel = polr(class ~ gender + gpa, data=suvrData)
summary(suvrModel)
```

A quick check that you have ordered the levels correctly is to look at the second table in the summary output. The rows should describe subsequent levels.

The AIC of this model is 1547. The AIC of the null model

```
suvrNullModel = polr(class~1, data=suvrData)
summary(suvrNullModel)
```

is 1566. Thus, our model is an improvement.

The model predicts that Eliska is a Junior (44.7%) or a Senior (36.5%):

```
eliska = data.frame(gender="Female", gpa=3.33)
predict(suvrModel, eliska, type="prob")
```

Here are the (abbreviated) raw results

```
Non-Mat    Fresh      Soph       Junior     Senior
0.00211    0.02356    0.16293    0.44656    0.36483
```

Thus, we do have an estimate for Eliska's class level, but there is a second option which is rather close. I'm not sure I would bet any money on where to put Eliska. Regardless, it is highly unlikely for Eliska to be either non-matriculated or a Freshman. Those probabilities, while non-zero, are *very* low.

Beware!

Remember that the data are not representative of the population. The distribution of the classes is quite similar to the probabilities predicted for Eliska. This is not surprising. The effect of the independent variables on the dependent variable are not statistically significant. Thus, these probabilities are essentially the relative proportions of the classes in the sample.

Graphic: Class against GPA

As for a graphic, we need our dependent variable to be the probability of each class. Since there is only one numeric independent variable, GPA, that will be the variable we graph along the x-axis.

The ultimate **question** is:

> What do we do with the gender variable?

One option is to plot the effect of gender on the same graphic. That means we will have 5×2 curves on the same plot (the number of levels by the number of genders recorded). That *may be* problematic as it may **overwhelm the graphic**. The figure below is this figure. Note that it does allow us to compare everything at once. However, you may find it overwhelming… or not.

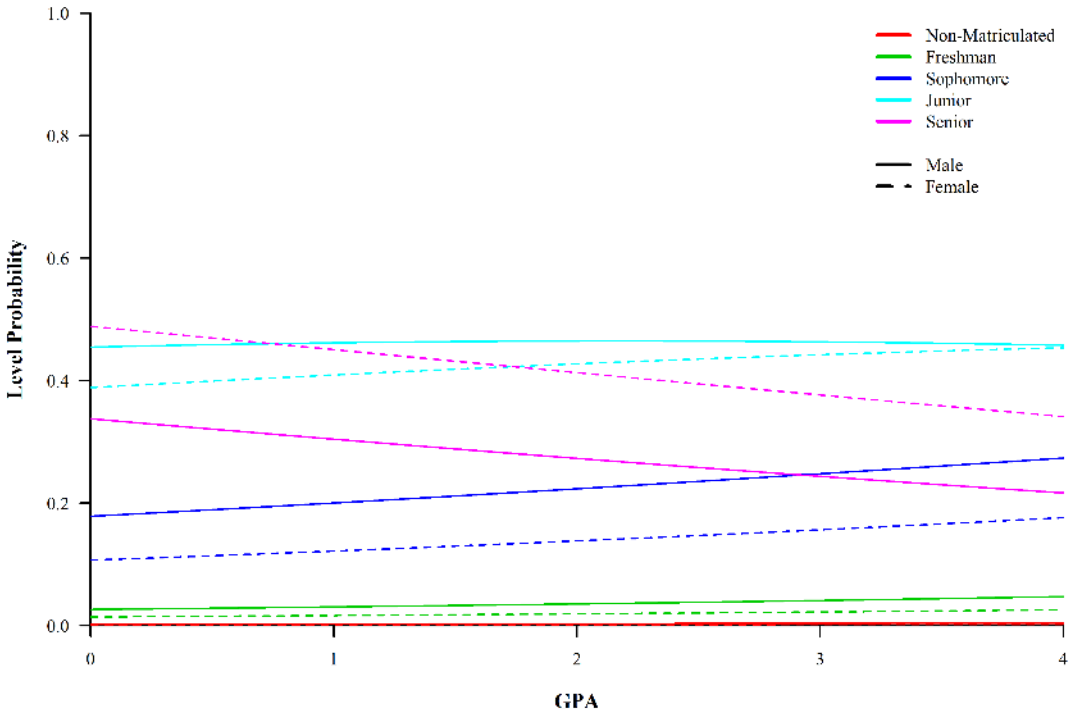


Figure 18.4.1: Graphic of the probability for each class level for each gender. Note that the non-matriculated and the Freshman levels uniformly have low probability. This is due to the nature of the data; only 2 non-matriculated and 22 Freshmen are in the sample of size $n = 661$. This limits what we can say about the population, unless the level distributions are similar to the population.

For the higher GPA values, it is most likely that the student is a Junior, regardless of the gender. At no place is it likely the student is either a Freshman or non-matriculated. This is supported by the data, as the number of non-matriculated students is just 2 and the number of Freshman is just 22 — out of a sample size of $n = 661$.

We can also use this graphic to estimate the various probabilities for Eliska. Remember she has a GPA of **3.33**. Since Eliska is female, we look at the dashed lines. Going to **3.33** on the x-axis and move vertically, we see that Eliska is most likely a member of the cyan level — Junior — with a close second being the magenta level — Senior. This conclusion agrees with our prediction above.

This page titled 18.4: The State University of Ruritania is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

18.5: Conclusion

In this chapter, we explored specialized regression techniques for modeling categorical dependent variables, distinguishing between nominal and ordinal data types. Each type presents unique analytical challenges, and we examined tailored approaches to address them.

- For **nominal dependent variables**, where categories lack inherent order, we extended the logistic regression framework to accommodate multiple outcomes. This multinomial approach essentially fits a series of binary logistic models, each comparing one category against a consistent reference level. Maintaining this consistent baseline is essential for producing coherent, interpretable results that allow meaningful comparison across all possible outcomes. This method preserves the flexibility of logistic modeling while adapting to the complexity of multiple unordered categories.
- For **ordinal dependent variables**, where categories follow a natural progression, we introduced a different conceptual model: the assumption of an underlying continuous latent variable. Rather than treating each category independently, this approach estimates a single linear function along with threshold values that partition the continuum into the observed ordinal levels. This method respects the data's ordered structure while leveraging the efficiency of continuous modeling techniques, providing both statistical power and intuitive interpretation.

Practical implementation in R simplifies both processes significantly. The prediction functions for these models are very powerful: they can directly predict the most likely outcome category for a given observation, while also providing the complete probability distribution across all possible categories. This capability supports both decision-making and probabilistic reasoning, allowing analysts to assess not just the predicted outcome but also the confidence and uncertainty associated with that prediction.

In summary, these specialized regression techniques expand our analytical toolkit for categorical data, bridging the gap between qualitative outcomes and quantitative modeling. By selecting the appropriate method based on whether categories are nominal or ordinal, you can extract meaningful insights while respecting the fundamental nature of their data, leading to more nuanced interpretations and more-informed decisions. Please remember all of this as you move ahead in your college (and post-college) life.

Be Aware!

Pay attention to the dependent variable and the data-generating process.

This page titled 18.5: Conclusion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

18.6: End-of-Chapter Materials

R Functions

In this chapter, we were introduced to several R functions that will be useful in the future. These are listed here.

Packages

- `KnoxStats`
  This package still contains some helpful functions. Here, we used it for the `set.base` and the `accuracy` functions.
- `MASS`
  This package is also a "book package," a package created for a specific book. Here, that book is *Modern Applied Statistics with S*, by Venables and Ripley.
- `nnet`
  This package contains many functions dealing with neural networks. For this chapter, we use it to fit multinomial models.

Statistics

- `multinom()`
  This modeling function allows you to fit nominal dependent variables. Its structure is standard in that its main argument is the formula. In order to use the `multinom` function, you must load the `nnet` package.
- `polr()`
  This modeling function allows you to fit ordinal dependent variables when there is an underlying linear function that drives the process. In order to use the `polr` function, you must load the `MASS` package.
- `predict(model, newdata)`
  As with almost all statistical packages, R has a predict function. It takes two parameters, the model, and a dataframe of the independent values from which you want to predict. If you omit `newdata`, then it will predict based on the independent variables of the data itself, which can be used to calculate residuals. The dataframe must list all independent variables with their associate new values. You can specify multiple new values for a single independent variable.
- `set.base()`
  This allows one to change the base category from which all other levels are estimated. It is a member of the `KnoxStats` package.
- `accuracy()`
  This calculates the basic accuracy of a confusion matrix. To be meaningful, this assumes the data are representative of the population.

Exercises

This section offers suggestions on things you can practice from this chapter.

1. In Section 18.1: Nominal Dependent Variables, we fit a multinomial model to the `gssocc` data. The base used was "Blue Collar." Refit the model using "Craft" as the base category.
2. Determine the AIC of the null model in Section 18.1: Nominal Dependent Variables.
3. As mentioned in Section 18.2: Ordinal Dependent Variables, calculate the relative accuracy of the model of the example.
4. As mentioned in Section 18.2: Ordinal Dependent Variables, add a *quadratic* education term to the model in the example to see if both the highly educated and the lesser educated both support the president.

Applied Readings

- Paul D. Allison and Nicholas A. Christakis (1994). "Logit Models for Sets of Ranked Items." *Sociological Methodology*. 24: 199–228.
- John Fox and Robert Andersen (2006). "Effect Displays for Multinomial and Proportional-Odds Logit Models." *Sociological Methodology*. 36: 225–55.
- Daniel Carson Johnson (1997). "Formal Education vs. Religious Belief: Soliciting New Evidence with Multinomial Logit Modeling." *Journal for the Scientific Study of Religion*. 36(**2**): 231–46.
- Mark R. Killingsworth and Cordelia W. Reimers (1983). "Race, Ranking, Promotions, and Pay at a Federal Facility: A Logit Analysis." *Industrial and Labor Relations Review*. 37(**1**): 92–107.
- Alan B. Lowther and John R. Skalski (1998). "A Multinomial Likelihood Model for Estimating Survival Probabilities and Overwintering for Fall Chinook Salmon Using Release: Recapture Methods." *Journal of Agricultural, Biological, and Environmental Statistics*. 3(**2**): 223–36.
- Christopher Winship and Robert D. Mare (1984). "Regression Models with Ordinal Variables." *American Sociological Review*. 49(**4**): 512–25.
- Judith E. Zeh, Daijin Ko, Bruce D. Krogman and Ronald Sonntag (1986). "A Multinomial Model for Estimating the Size of a Whale Population from Incomplete Census Data." *Biometrics*. 42(**1**): 1–14.

Theory Readings

- B. R. Bhat and N. V. Kulkarni (1966). "On Efficient Multinomial Estimation." *Journal of the Royal Statistical Society*. Series B (Methodological). 28(**1**): 45–52.
- Zhen Chen and Lynn Kuo (2001). "A Note on the Estimation of the Multinomial Logit Model with Random Effects." *The American Statistician*. 55(**2**): 89–95.
- Jean-Yves Dauxois and Syed N. U. A. Kirmani (2003). "Testing the Proportional Odds Model under Random Censoring." *Biometrika*. 90(**4**): 913–22.
- Byung Soo Kim and Barry H. Margolin (1992). "Testing Goodness of Fit of a Multinomial Model Against Overdispersed Alternatives." *Biometrics*. 48(**3**): 711–19.
- Bercedis Peterson and Frank E. Harrell, Jr. (1990). "Partial Proportional Odds Models for Ordinal Response Variables." *Journal of the Royal Statistical Society*. Series C (Applied Statistics). 39(**2**): 205–17.
- G. A. F. Seber and S. O. Nyangoma (1974). "Residuals for Multinomial Models." *Biometrika*. 87(**1**): 183–91.
- M. Stone (1974). "Cross-Validation and Multinomial Prediction." *Biometrika*. 61(**3**): 509–15.
- Y. K. Tse (1987). "A Diagnostic Test for the Multinomial Logit Model." *Journal of Business & Economic Statistics*. 5(**2**): 283–86.

This page titled 18.6: End-of-Chapter Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

19: Appendices

Appendices serve as a supplementary section at the end of a document, such as a research paper, report, or (here) book. Their primary purpose is to include material that is too detailed, lengthy, or tangential to belong in the main body of the work, but which nonetheless supports its credibility, provides essential background, or offers further insight for interested readers. In general, this can include raw data sets, extensive technical calculations, detailed maps, lengthy transcripts of interviews, survey questionnaires, or copies of relevant primary documents. Here, I use the appendices to include material needed to understand the material in the textbook. By placing this particular material in an appendix, I try to maintain the clarity and flow of the main argument while still providing support for material you do not remember.

In this textbook, I designed the appendices as technical repositories and practical manuals. They house the foundational tools and explicit computational instructions that underpin the methods discussed in the main chapters. To support your work, we have included three targeted appendices.

- **The Appendix of Matrices** provides a concise review of matrix algebra operations crucial for understanding the general linear model.
- **The Appendix of R** serves as a hands-on guide, offering specific code and workflow examples for implementing analyses in the R statistical environment.
- **The Appendix of Statistics** collects key formulas, distribution tables, and proofs of fundamental theorems referenced throughout the book.

Together, these three appendices allow the core narrative to focus on conceptual understanding and application, while ensuring you have immediate access to the mathematical, computational, and statistical reference material needed to complete your work.

> In other words, check these appendices for gaps in your knowledge.

This page titled 19: Appendices is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

20: The Appendix of Matrices

Before we can properly delve into the mechanics and meaning of regression analysis, it is absolutely essential that we first lay a solid mathematical foundation. The entire framework of linear models is built upon this specific groundwork, and attempting to proceed without it would be like trying to build a kingdom in Ruritania.

For you to get the most out of this book, my expectation is that you have already successfully completed a course in matrix algebra. I'm not just referring to a passing familiarity, but a comfortable, working knowledge. This means you should be proficient in the fundamental operations: You can confidently add and multiply matrices *and* you understand how these operations combine and interact. Furthermore, you should grasp the conceptual idea that matrices represent linear transformations — that they can rotate, stretch, and squash vectors and vector spaces. Your experience should also include having worked directly with eigenvalues and eigenvectors, understanding their significance in revealing a matrix's core properties. Finally, you should be capable of calculating fundamental characteristics like the rank and the trace of a matrix, knowing what these values tell you about the matrix's structure and behavior.

None of these topics is a footnote; they are critically important for moving beyond a superficial application of formulas and toward a deeper, more intuitive understanding of the mathematics that underpin linear models. They are, in effect, the language in which the theory is written. Therefore, the purpose of this appendix is twofold:

- First, it will serve as a refresher, reviewing some of these pivotal concepts from your prior matrix algebra class to ensure they are fresh in your mind.
- Second, it will introduce and elaborate upon a few additional items and perspectives that you may not have encountered previously, thereby bridging the gap between a standard matrix algebra course and the specific demands of statistical modeling.

20.1: Matrix Basics
20.2: Addition
20.3: Multiplication
20.4: Other Terms and Operations
20.5: Consequences
20.6: Statistics in Matrices
20.7: End-of-Appendix Materials

This page titled 20: The Appendix of Matrices is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

20.1: Matrix Basics

At its core, a matrix is a simple but powerful rectangular array of numbers, symbols, or expressions, arranged in rows and columns. Think of it as a structured data table or a digital spreadsheet, where each individual item within the grid is called an element. We typically define a matrix by its dimensions — the number of rows followed by the number of columns. These arrays are far more than just organizational tools; they are the fundamental language for representing and performing linear transformations and data.

✦•·····• ✦ •·····•✦

A matrix is just a rectangular array of **scalars**. It is used to simplify many mathematical calculations. Throughout this book, I will use it in such a sense. The following is an example of a matrix:

$$\mathbf{A} = \begin{bmatrix} 3 & 5 & 2 \\ a & 1 & 18 \end{bmatrix} \tag{20.1.1}$$

Because a matrix is a *rectangular* array, it has a **dimension**. The matrix $\mathbf{A}$ above has dimension 2×3 because there are 2 rows and 3 columns. We could also write this as

$$\mathbf{A} \in \mathcal{M}_{2\times 3} \tag{20.1.2}$$

This can be read in a few ways:

- "$\mathbf{A}$ is a 2×3 matrix"
- "$\mathbf{A}$ is a matrix with dimension 2×3"
- "$\mathbf{A}$ is an element of the set of 2×3 matrices"

As an aside, note that the symbol $\in$ means "is an element of" and $\mathcal{M}_{2\times 3}$ is "the set of all matrices of dimension 2×3."

Also note that the dimension order is very important and is always written as rows $\times$ columns. $\mathcal{M}_{2\times 3}$ and $\mathcal{M}_{3\times 2}$ are entirely different sets of matrices.

A matrix is **square** if the number of rows equals the number of columns. That is, $\mathbf{B}$ is square if

$$\mathbf{B} \in \mathcal{M}_{n\times n} \tag{20.1.3}$$

for some number $n \in$. If a matrix is square, the set is often denoted simply by $\mathcal{M}_n$. The matrix $\mathbf{A}$ above is *not* square because the number of rows does not equal the number of columns.

Representation

The next sections cover the algebra of matrices. To ease the notation, let me show you two ways of representing matrices. First, here is matrix \mathbf{A} written out.

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,c} \\ a_{2,1} & a_{2,2} & a_{2,3} & \cdots & a_{2,c} \\ a_{3,1} & a_{3,2} & a_{3,3} & \cdots & a_{3,c} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{r,1} & a_{r,2} & a_{r,3} & \cdots & a_{r,c} \end{bmatrix} \tag{20.1.4}$$

Note that every element in the $\mathbf{A}$ matrix is represented by a lowercase a and its r, c position in the matrix. This allows us to simplify representation at times:

$$\mathbf{A} = [a_{ij}] \tag{20.1.5}$$

Here, i is the row index and j the column index.

Now, with these two representations, we can enter the realm of the algebra of matrices.

This page titled 20.1: Matrix Basics is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

20.2: Addition

Matrix addition is crucial for linear models, as it provides the mechanism for incorporating an error or residual term. In the classic linear model formulation, the vector of observed responses is expressed as the *sum* of the systematic component (**XB**), and the random component (**E**). Without the simple, element-wise operation of matrix addition, we could not structurally separate the predictable pattern in the data from the inherent, unexplained variability.

✦•·····• ✦ •·····•✦

Matrix addition is closed. This means that the sum of two matrices will always give you another matrix... as long as it makes sense to add two matrices. Two matrices are can be added if they have the same dimension. That is, two matrices are commensurate for addition if they have the same dimension.

Definition: Commensurate

When the matrices have the correct dimension to perform the mathematical operation, they are called "**commensurate**." For addition, commensurate matrices have the same dimension. For multiplication, the requirement is much different (see here).

Let $\mathbf{A} \in \mathcal{M}_{r \times c}$ and $\mathbf{B} \in \mathcal{M}_{r \times c}$ for some values of r and c. $\mathbf{A}$ and $\mathbf{B}$ are commensurate and can be summed. Matrix addition is element-by-element (**elementwise**) addition. Thus,

$$\mathbf{A}+\mathbf{B} = \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} & a_{13}+b_{13} & \cdots & a_{1c}+b_{1c} \\ a_{21}+b_{21} & a_{22}+b_{22} & a_{23}+b_{23} & \cdots & a_{2c}+b_{2c} \\ a_{31}+b_{31} & a_{32}+b_{32} & a_{33}+b_{33} & \cdots & a_{3c}+b_{3c} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{r1}+b_{r1} & a_{r2}+b_{r2} & a_{r3}+b_{r3} & \cdots & a_{rc}+b_{rc} \end{bmatrix} \tag{20.2.1}$$

This can also be symbolized (shortened) as

$$\mathbf{A}+\mathbf{B} = [a_{i,j}] + [b_{i,j}] = [a_{i,j}+b_{i,j}] \tag{20.2.2}$$

Matrix addition has a zero (**additive identity**). It is the commensurate matrix with all elements equal to zero:

$$\mathbf{0} = \begin{bmatrix} 0_{1,1} & 0_{1,2} & 0_{1,3} & \cdots & 0_{1,c} \\ 0_{2,1} & 0_{2,2} & 0_{2,3} & \cdots & 0_{2,c} \\ 0_{3,1} & 0_{3,2} & 0_{3,3} & \cdots & 0_{3,c} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0_{r,1} & 0_{r,2} & 0_{r,3} & \cdots & 0_{r,c} \end{bmatrix} = [0_{i,j}] \tag{20.2.3}$$

How does the additive identity work in addition? Just as you would expect:

$$\mathbf{A}+\mathbf{0} = \begin{bmatrix} a_{11}+0 & a_{12}+0 & a_{13}+0 & \cdots & a_{1c}+0 \\ a_{21}+0 & a_{22}+0 & a_{23}+0 & \cdots & a_{2c}+0 \\ a_{31}+0 & a_{32}+0 & a_{33}+0 & \cdots & a_{3c}+0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{r1}+0 & a_{r2}+0 & a_{r3}+0 & \cdots & a_{rc}+0 \end{bmatrix} = \mathbf{A} \tag{20.2.4}$$

This can also be symbolized (shortened) as

$$\mathbf{A}+\mathbf{0} = [a_{ij}+0] = [a_{ij}] \tag{20.2.5}$$

I leave it as an exercise to prove $\mathbf{A}+\mathbf{0} = \mathbf{0}+\mathbf{A} = \mathbf{A}$.

Matrices also have an **additive inverse**. As with scalar arithmetic, a matrix plus its additive inverse equals the zero matrix; that is, if $\mathbf{B}$ is the additive inverse of $\mathbf{A}$, then $\mathbf{A}+\mathbf{B} = \mathbf{0}$.

Two things about the additive inverse: First, it is commensurate with the original matrix. Second, it is *unique* (just as in scalar arithmetic).

To calculate the additive inverse of $\mathbf{A}$, just negate each element of $\mathbf{A}$. Thus, if $\mathbf{B} = [-a_{ij}]$ then $\mathbf{B}$ is the additive inverse of $\mathbf{A}$.

Finally, as with all elementwise operations, matrix addition is both **commutative** and **associative**:

- $\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$
- $(\mathbf{A} + \mathbf{B}) + \mathbf{C} = \mathbf{A} + (\mathbf{B} + \mathbf{C})$

In summation, matrix addition behaves like scalar addition, as long as the matrices are commensurate.

Caution

As you may be able to guess, a lot of the proofs rely on checking if the matrices involved are commensurate. So, keep this in mind as you proceed through proofs.

This page titled 20.2: Addition is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

20.3: Multiplication

Matrix multiplication is not a single operation but rather a family of distinct products, each with its own rules and purpose. The most fundamental is the "usual" matrix product for linear transformations. However, there are others. Alongside the usual, we have the Hadamard product for element-wise multiplication, the Kronecker product for creating block matrices, and scalar multiplication which uniformly scales every element in a matrix.

This diversity of products is the computational bedrock of linear models. Without these specific operations, the elegant, compact matrix equations that define regression would be impossible, forcing us to rely on cumbersome and inefficient scalar notations... which we will experience.

✦•·····• ✦ •·····•✦

There are many, many, *many* types of multiplication with matrices. As expected, the version used depends on the need. As only the scalar product and the matrix product are typically seen in an undergraduate linear models course, we will only discuss those two here.

Scalar Product

As in arithmetic, the scalar product arose from needing to repeatedly add a matrix to itself. Thus, instead of writing $\mathbf{A}+\mathbf{A}+\mathbf{A}+\mathbf{A}+\mathbf{A}+\mathbf{A}+\mathbf{A}+\mathbf{A}$, one could write $8\mathbf{A}$, where 8 is a scalar. This was quickly generalized to non-integer values for the scalar multiple, just as $3\times a$ was quickly generalized to things like $4.25\times a$.

Scalar multiplication is defined as

$$c\mathbf{A}=[ca_{ij}] \tag{20.3.1}$$

Scalar products do not change the **dimension** of the matrix. That is, if c is a scalar and $\mathbf{A}\in\mathcal{M}_{r\times c}$, then $c\mathbf{A}\in\mathcal{M}_{r\times c}$.

Scalar products are **commutative**. That is, if c is a scalar and $\mathbf{A}$ is a matrix, then $c\mathbf{A}=\mathbf{A}c$. This will come in handy later, so be aware of it. Note that c does not need to be a natural number.

Scalar products are also **associative**. That is, if c is a scalar, then the following are equivalent:

- $c\mathbf{AB}$
- $\mathbf{A}c\mathbf{B}$
- $\mathbf{AB}c$
- $(c\mathbf{A})\,\mathbf{B}$
- $c\,(\mathbf{AB})$

Scalar multiplication is also **distributive** over matrix addition. Thus, $c\,(\mathbf{A}+\mathbf{B})=c\mathbf{A}+c\mathbf{B}$.

Matrix Product

The matrix product is the multiplication that is (usually) meant when one just says "matrix multiplication." Its definition arises from linear algebra and repeated linear transformations. It has many nice properties. Calculation is not one of them.

Let us define two matrices $\mathbf{A}$ and $\mathbf{B}$ such that the number of columns of $\mathbf{A}$ equals the number of rows of $\mathbf{B}$. Their product is defined as

$$\mathbf{AB}=[ab_{i,j}]=\left[\sum_k a_{i,k}b_{k,j}\right] \tag{20.3.2}$$

Here, k ranges between 1 and the number of columns of $\mathbf{A}$. The dimension of the product is the number of rows of $\mathbf{A}$ by the number of columns of $\mathbf{B}$. That is, let $\mathbf{A}\in\mathcal{M}_{r1\times c1}$ and $\mathbf{B}\in\mathcal{M}_{r2\times c2}$. Then, $\mathbf{A}$ and $\mathbf{B}$ are commensurate for matrix multiplication if $c1=r2$. The dimension of the product is $r1\times c2$.

Example 20.3.1

Let us define our two matrices as

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \tag{20.3.3}$$

and

$$\mathbf{B} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & m \end{bmatrix} \tag{20.3.4}$$

Find the product $\mathbf{AB}$.

Solution:

The first step is to check that the matrices are commensurate for multiplication: Do the number of columns of $\mathbf{A}$ equal the number of rows of $\mathbf{B}$? Note that $\mathbf{A} \in \mathcal{M}_{2\times 3}$ and $\mathbf{B} \in \mathcal{M}_{3\times 3}$. Because the "inners" of $\mathbf{AB}$ are equal to each other, the two matrices are commensurate.

Second, we determine the dimension of the product. It is the number of rows of $\mathbf{A}$ by the number of columns of $\mathbf{B}$: 2×3, the "outers."

Third, since we know the dimension of the product, we just have to fill in the blanks in the matrix:

$$\mathbf{AB} = \begin{bmatrix} - & - & - \\ - & - & - \end{bmatrix} \tag{20.3.5}$$

The top-right element in the product matrix is element $1,1$. Thus, by our definition, it equals

$$ab_{1,1} = \left[\sum_k a_{1,k}b_{k,1}\right] \tag{20.3.6}$$

$$= [a_{1,1}b_{1,1} + a_{1,2}b_{2,1} + a_{1,3}b_{3,1}] \tag{20.3.7}$$

$$= [1a + 2d + 3g] \tag{20.3.8}$$

The top-middle element, $ab_{1,2}$, is

$$ab_{1,2} = \left[\sum_k a_{1,k}b_{k,2}\right] \tag{20.3.9}$$

$$= [a_{1,1}b_{1,2} + a_{1,2}b_{2,2} + a_{1,3}b_{3,2}] \tag{20.3.10}$$

$$= [1b + 2e + 3h] \tag{20.3.11}$$

Note what is happening here. The elements of the "top-right" cell is the inner product of the *top* row and the *right* column. Similarly, the bottom-center element is the inner product of the *bottom* row and the *center* column.

I leave it as an exercise for you to finish the multiplication. Here is the final product:

$$\mathbf{AB} = \begin{bmatrix} 1a + 2d + 3g & 1b + 2e + 3h & 1c + 2f + 3m \\ 4a + 5d + 6g & 4b + 5e + 6h & 4c + 5f + 6m \end{bmatrix} \tag{20.3.12}$$

In scalar arithmetic, we have a multiplicative identity, multiplicative inverse, and multiplication is commutative, associative, and distributes over addition. All of these hold for matrix multiplication --- *except* commutativity. In general, $\mathbf{AB} \neq \mathbf{BA}$, even when the multiplications both make sense.

The multiplicative **identity**, which we will symbolize by $\mathbf{I}$, has the property that $\mathbf{AI} = \mathbf{IA} = \mathbf{A}$. We will call $\mathbf{I}$ the identity matrix.

Caution

Technically, this statement is only true if $\mathbf{A}$ is square. If it is not square, then the two identity matrices will have different dimension. We will restrict ourselves to square matrices. The "generalized inverse" is beyond the scope of this text.

If a multiplicative identity exists, then a **multiplicative inverse** also exists. The inverse of a matrix $\mathbf{A}$, denoted $\mathbf{A}^{-1}$, is a matrix that satisfies these two requirements:

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I} \tag{20.3.13}$$

Not all matrices have inverses. Those that do *not* are called **singular**. Those that *do* are called **invertible**.

Only square matrices *can be* invertible. However, not all square matrices *are* invertible. From linear algebra, a matrix is invertible if and only if it is square and is of **full rank**.

A consequence of this is that a matrix is invertible if its **determinant** is non-zero. In general, the calculation of the determinant and the inverse are computationally intensive. However, they are rather straight-forward for 2×2 matrices. Since much of the initial exploration of linear models takes place in the realm of one dependent and one independent variable, it behooves us to focus a bit on 2×2 matrices.

Theorem 20.3.1

Let $\mathbf{A} \in \mathcal{M}_{2\times 2}$. Then, the determinant of $\mathbf{A}$ is

$$\det \mathbf{A} = a_{11}a_{22} - a_{12}a_{21}$$

From your linear algebra class, you may have learned that the determinant of a matrix is a single **scalar** value that reveals the matrix's scaling factor for area (or volume or hyper-volume) when it acts as a linear transformation. A determinant of zero indicates that the transformation compresses the space into a lower dimension, making the matrix non-invertible, while a non-zero determinant confirms the transformation preserves the full dimensionality of the space. Since the rank of a matrix is important to statisticians, this last feature is important in terms of the data (information).

Theorem 20.3.2

If $\mathbf{A} \in \mathcal{M}_{2\times 2}$ then its inverse is

$$\mathbf{A}^{-1} = \frac{1}{\det \mathbf{A}} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix} \tag{20.3.14}$$

Proof

To prove this, we will show that $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$ and that $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$.

$$\mathbf{A}\mathbf{A}^{-1} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad \frac{1}{\det \mathbf{A}} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix} \tag{20.3.15}$$

$$= \frac{1}{\det \mathbf{A}} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix} \tag{20.3.16}$$

$$= \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{11}a_{22} - a_{12}a_{21} & -a_{11}a_{12} + a_{11}a_{12} \\ a_{21}a_{22} - a_{21}a_{22} & -a_{12}a_{21} + a_{11}a_{22} \end{bmatrix} \tag{20.3.17}$$

$$= \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{11}a_{22} - a_{12}a_{21} & 0 \\ 0 & -a_{12}a_{21} + a_{11}a_{22} \end{bmatrix} \tag{20.3.18}$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{20.3.19}$$

$$= \mathbf{I} \tag{20.3.20}$$

Thus, we have shown that $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$. This is just one-half of the proof. I leave it as an exercise for you to prove the second half: $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$.

Note

The reason I provide the mathematics for 2×2 matrices is that in our study of simple linear regression, many of the important calculations will be done with 2×2 matrices. See Chapter 4: Matrices and Linear Regression.

Formulas *do* exist for 3×3 matrices, too. However, once we move beyond that, hand calculations are time-prohibitive. At the end of this section, I provide some examples of performing these calculations in R.

Now that we have a mechanism to calculate a multiplicative inverse, let us see that not all square matrices have one.

Example 20.3.2

Let $\mathbf{A} \in \mathcal{M}_{2\times 2}$ be defined as

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & 6 \end{bmatrix} \tag{20.3.21}$$

Calculate $\mathbf{A}^{-1}$.

Solution:

Let us calculate $\mathbf{A}^{-1}$ using our formula,

$$\mathbf{A}^{-1} = \frac{1}{\det \mathbf{A}} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix} \tag{20.3.22}$$

Applying this formula is straight-forward:

$$\mathbf{A}^{-1} = \frac{1}{0} \begin{bmatrix} 6 & -3 \\ -2 & 1 \end{bmatrix} \tag{20.3.23}$$

Yes, the determinant of $\mathbf{A}$ is $\det \mathbf{A} = 1 \times 6 - 3 \times 2 = 6 - 6 = 0$. Since the determinant is 0, the inverse does not exist (one cannot divide by 0). The matrix is singular.

What is it about the $\mathbf{A}$ matrix that makes it singular?

Note that the second column is just 3 times the first column (or the second row is twice the first). This means the matrix is not full rank. The columns are not linearly independent. When we get to using these matrices with real data, we will see this as the second column gives us **no knowledge** about the world that is not already contained in the first column. The second column is **redundant**.

So far, we have seen the multiplicative identity and the multiplicative inverse. It is time to note that matrix multiplication is *not* commutative, in general.

Theorem 20.3.3

Matrix multiplication is *not* commutative. That is, there exist $\mathbf{A}$ and $\mathbf{B}$ such that $\mathbf{AB} \neq \mathbf{BA}$.

Not that this is not saying that it is *impossible* for $\mathbf{AB} = \mathbf{BA}$. It is merely saying that we cannot assume it. There are definitely times that the product is commutative. In fact, knowing those times will help us simplify some of the more-complicated models.

Proof:

The proof is simple. It is simply a counter-example. Let

$$\mathbf{A} = \begin{bmatrix} 3 & 1 \\ 2 & 7 \end{bmatrix} \tag{20.3.24}$$

and

$$\mathbf{B} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \tag{20.3.25}$$

Note that $\mathbf{AB} \neq \mathbf{BA}$.

Since we have found a counter-example, we have shown that matrix multiplication is not commutative, in general.

Note

While technically correct, this proof leaves us feeling a little empty. So we found one counter-example. Cool beans. But we learned precious little about commutativity with matrix multiplication. Let us explore a bit and see if we can determine *when* matrix multiplication is commutative. We may learn something interesting.

Exploring Matrices:

First, let us assume $\mathbf{A}$ and $\mathbf{B}$ are square and commensurate. This ensures that $\mathbf{AB}$ and $\mathbf{BA}$ can be calculated. For instance, let both be 2×2 matrices. Then, $\mathbf{AB}$ is

$$\mathbf{AB} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \tag{20.3.26}$$

$$= \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix} \tag{20.3.27}$$

and $\mathbf{BA}$ is

$$\mathbf{BA} = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \tag{20.3.28}$$

$$= \begin{bmatrix} b_{11}a_{11} + b_{12}a_{21} & b_{11}a_{12} + b_{12}a_{22} \\ b_{21}a_{11} + b_{22}a_{21} & b_{21}a_{12} + b_{22}a_{22} \end{bmatrix} \tag{20.3.29}$$

$$= \begin{bmatrix} a_{11}b_{11} + a_{21}b_{12} & a_{12}b_{11} + a_{22}b_{12} \\ a_{11}b_{21} + a_{21}b_{22} & a_{12}b_{21} + a_{22}b_{22} \end{bmatrix} \tag{20.3.30}$$

By comparing the two product matrices, $\mathbf{AB}$ and $\mathbf{BA}$, we can determine some restrictions under which multiplication *is* commutative.

For instance, if $a_{12} = a_{21} = 0$ and $b_{12} = b_{21} = 0$, then the two product matrices are the same. In other words, if both $\mathbf{A}$ and $\mathbf{B}$ are *both diagonal matrices*, multiplication will be commutative. That's an interesting consequence we would have missed if we just stopped with our proof.

In fact, now that we know what to look for, it is trivial to show that multiplication of diagonal matrices is commutative in general... as we do here:

Theorem 20.3.4

Let $\mathbf{A}$ and $\mathbf{B}$ be diagonal matrices of the same size. The product is commutative; that is, $\mathbf{AB} = \mathbf{BA}$.

Proof:

Since \mathbf{A} and \mathbf{B} are diagonal and of the same shape, they can be represented as

$$\mathbf{A} = \begin{bmatrix} a_1 & 0 & 0 & \cdots & 0 \\ 0 & a_2 & 0 & \cdots & 0 \\ 0 & 0 & a_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_n \end{bmatrix} \tag{20.3.31}$$

and

$$\mathbf{B} = \begin{bmatrix} b_1 & 0 & 0 & \cdots & 0 \\ 0 & b_2 & 0 & \cdots & 0 \\ 0 & 0 & b_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & b_n \end{bmatrix} \tag{20.3.32}$$

By definition, the product $\mathbf{AB}$ is

$$\mathbf{AB} = \begin{bmatrix} a_1 & 0 & 0 & \cdots & 0 \\ 0 & a_2 & 0 & \cdots & 0 \\ 0 & 0 & a_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_n \end{bmatrix} \begin{bmatrix} b_1 & 0 & 0 & \cdots & 0 \\ 0 & b_2 & 0 & \cdots & 0 \\ 0 & 0 & b_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & b_n \end{bmatrix} \tag{20.3.33}$$

$$= \begin{bmatrix} a_1 b_1 & 0 & 0 & \cdots & 0 \\ 0 & a_2 b_2 & 0 & \cdots & 0 \\ 0 & 0 & a_3 b_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_n b_n \end{bmatrix} \tag{20.3.34}$$

Similarly, the product $\mathbf{BA}$ is

$$\mathbf{BA} = \begin{bmatrix} b_1 & 0 & 0 & \cdots & 0 \\ 0 & b_2 & 0 & \cdots & 0 \\ 0 & 0 & b_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & b_n \end{bmatrix} \begin{bmatrix} a_1 & 0 & 0 & \cdots & 0 \\ 0 & a_2 & 0 & \cdots & 0 \\ 0 & 0 & a_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_n \end{bmatrix} \tag{20.3.35}$$

$$= \begin{bmatrix} b_1 a_1 & 0 & 0 & \cdots & 0 \\ 0 & b_2 a_2 & 0 & \cdots & 0 \\ 0 & 0 & b_3 a_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & b_n a_n \end{bmatrix} \tag{20.3.36}$$

$$= \begin{bmatrix} a_1 b_1 & 0 & 0 & \cdots & 0 \\ 0 & a_2 b_2 & 0 & \cdots & 0 \\ 0 & 0 & a_3 b_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_n b_n \end{bmatrix} \tag{20.3.37}$$

$$= \mathbf{AB} \tag{20.3.38}$$

Thus, we have shown $\mathbf{AB} = \mathbf{BA}$ for two diagonal matrices of the same size.

Finally, I leave it as an exercise for you to prove that matrix multiplication is **associative** (when the multiplication can be done). That is, if $\mathbf{ABC}$ can be calculated, then it can be calculated as either $(\mathbf{AB})\,\mathbf{C}$ or as $\mathbf{A}\,(\mathbf{BC})$.

This page titled 20.3: Multiplication is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

20.4: Other Terms and Operations

There exist other helpful operations on matrices. Already, we have come across the determinant as being especially helpful in determining if a matrix is invertible or singular. However, there are more. I'm not sure where to put them, so I'm putting all of them in this one section.

Enjoy!

✦•·····• ✦ •·····•✦

Trace

Another useful function is the trace. It is just the sum of the diagonal elements. That is:

$$\text{tr } \mathbf{A} = \sum_{i=1}^{r} a_{ii} \tag{20.4.1}$$

The formula is simple... perhaps deceptively so. In fact, one may wonder what the trace actually tells us about a matrix. Quite simply, it provides another fundamental scalar descriptor of the matrix. While it is less intuitive than the determinant, it offers several key insights:

- Invariant Under Change of Basis: The trace is a similarity invariant, meaning that even if you change the coordinate system (e.g., perform a similarity transformation like $\mathbf{P^{-1}AP}$), the trace of the matrix remains unchanged. This points to it capturing some intrinsic property of the underlying data, independent of how it is represented.
- Sum of Eigenvalues: The trace is equal to the sum of the matrix's eigenvalues. This is a profoundly important connection, as the eigenvalues represent the fundamental scaling factors of the transformation in its principal directions. Therefore, the trace gives you a single number that summarizes the total "spread" of these scaling factors.
  [Recall that the determinant is the *product* of the eigenvalues.]
- A Measure of "Total Effect": In certain contexts, the trace can be interpreted as a measure of the total variance explained. For example, in a projection matrix used in regression, the trace (which equals its rank) indicates the number of dimensions in the projected subspace. In other words, the trace of the hat matrix is the number of **degrees of freedom** needed to estimate the parameters.

In essence, while the determinant tells you about the *multiplicative* scaling of volume, the trace tells you about the *additive* sum of the matrix's core scaling components. Its invariance makes it a crucial tool for understanding the essential nature of a linear transformation.

Transpose

The transpose of a matrix is just the matrix where the rows and columns are switched. Thus, if $\mathbf{B}$ is the transpose of $\mathbf{A}$, then $b_{ij} = a_{ji}$. In symbols, we indicate the transpose as

$$\mathbf{B} = \mathbf{A}' \tag{20.4.2}$$

In some sources, it is also symbolized as

$$\mathbf{B} = \mathbf{A}^{\top} \tag{20.4.3}$$

There are three quick results that you will need to prove:

1. $\mathbf{A'A}$ and $\mathbf{AA'}$ both exist and are symmetric.
2. $\mathbf{rank A'A} = \mathbf{rank AA'}$, the matrix and its transpose contain the same amount of information.
3. $(\mathbf{A}')^{-1} = (\mathbf{A}^{-1})'$, the inverse of a transposed matrix is the transpose of the inverted matrix. This presupposed that $\mathbf{A}^{-1}$ exists.

Symmetric

A matrix is symmetric if it is equal to its transpose, $\mathbf{A} = \mathbf{A}'$.

$$[a_{ij}] = [a_{ji}] \tag{20.4.4}$$

Note that only square matrices *can be* symmetric. Symmetric matrices have some nice properties with respect to calculations.

Also note that one can "**symmetrize**" any square matrix. That is, one can form symmetric matrix $\mathbf{X}$ from a square matrix $\mathbf{A}$ as

$$\mathbf{X} = \frac{\mathbf{A} + \mathbf{A}'}{2} \tag{20.4.5}$$

I leave it as an exercise to prove that $\mathbf{X}$ is symmetric.

One important feature of symmetric matrices is that they can be transformed into a **diagonal** matrix. If $\mathbf{A}$ is symmetric, then there exists a matrix $\mathbf{Q}$ such that $\mathbf{AQ}$ is diagonal.

Note

Why is this helpful?

First, remember that multiplication of diagonal matrices is commutative.

Second, as you will see in the text, diagonal covariance matrices indicate independence of the variables. This means that any set of variables can be linearly transformed into a set of independent variables. This fact is the basis for a procedure called "principal component analysis."

Idempotency

A matrix $\mathbf{A}$ is **idempotent** if $\mathbf{AA} = \mathbf{A}$. This means that multiplying the matrix by itself yields the original matrix; i.e., once the transformation is applied a first time, applying it again will not change the result.

The concept of idempotency is the formal mathematical expression of a **projection** (see below). Imagine projecting a 3D object onto a 2D wall — the shadow is a "flattened" version of the object. If you then try to project that 2D shadow onto the same wall again, it remains unchanged. The idempotent matrix captures this exact property: it "squashes" vectors onto a subspace, and any vector already in that subspace is left untouched by a subsequent application of the transformation.

The Determinant

The determinant is a mathematical concept that serves as a diagnostic tool for assessing the fundamental viability of multivariate models. In regression, the determinant of the $\mathbf{X}'\mathbf{X}$ matrix determines whether the ordinary least squares solution exists at all. If the determinant equals zero, the matrix is singular, indicating perfect multicollinearity among predictors, which prevents inversion and makes estimation impossible. Beyond mere existence, the magnitude of the determinant also conveys the **overall collinearity** in the data; a near-zero determinant suggests high multicollinearity, even if not perfect, leading to unstable coefficient estimates with inflated standard errors. Thus, the determinant acts as a gatekeeper for model validity, a gauge of data stability, and a key component in many advanced statistical procedures.

- det M = 0
  M is rank deficient; its inverse does not exist
- det M ≠ 0
  M is full rank; it has an inverse
- |det(A)|
  This is the "volume scaling factor" of the linear transformation (of the matrix)

Calculating the determinant of a general square matrix by hand is a bit beyond the needs of this course. However, the determinant of a 2 times; 2 matrix is useful:

$$\det A = a_{1,1}a_{2,2} - a_{1,2}a_{2,1} \tag{20.4.6}$$

In R, the function is `det` :

```
A = matrix( c(1,2,3,4,5,6,7,8,9), ncol=3)
det(A)
```

Caution

There are a couple of ways to indicate the determinant of a matrix. Some sources use `|A|` to indicate the determinant of the matrix A. This can be problematic if one also is taking absolute values.

Because of this possible confusion, I prefer `det` , where `det A` indicates the determinant of the matrix A.

As always, be aware of the symbols and terminology used by the author you are reading.

Matrix Rank

In statistics, especially when we move into multivariable methods like linear regression, we rely heavily on linear algebra. At the heart of many of these techniques is a key operation: solving for a set of unknown parameters. The **rank of a matrix** is the concept that tells us whether a unique and sensible solution exists.

Think of it this way: the rank of a matrix is the number of **linearly independent** rows or columns it contains. "Linearly independent" means that no row or column can be created by a linear combination of the others. It's a measure of the matrix's **true informational content**.

Its importance becomes clear in two fundamental areas:

1. **Invertibility**: Can We Compute Our Estimates?

In ordinary least squares (OLS) regression, we estimate the coefficients using the formula:

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$$

The critical step is inverting the matrix $\boldsymbol{X'X}$. A matrix can be inverted **if and only if it is** "full rank."

- Full Rank: The rank equals the number of columns in $\mathbf{X}$. This means all our predictor variables are linearly independent. $\mathbf{X'X}$ is invertible, the formula works, and we get unique estimates for all our coefficients.
- Rank Deficient (Less-Than-Full Rank): The rank is less than the number of columns. This signals *perfect* multicollinearity — one predictor is an exact linear combination of others (e.g., including variables for "weight in kg" and "weight in lbs," or the sum of all dummy variables for a categorical factor). In this case, $\mathbf{X'X}$ is **singular**; it cannot be inverted. The formula breaks down, and no unique set of coefficients exists—the computer will either throw an error or drop a variable arbitrarily.

This leads to a very important **take-away**:

- The rank is a diagnostic for perfect multicollinearity. It tells us if our model's design is fundamentally flawed before we even look at the data.

2. **Dimensionality**: What Are We Actually Measuring?

Rank also tells us about the *effective dimensionality* of our data. Imagine you have 10 survey questions, but due to redundant phrasing, they only capture 3 distinct underlying attitudes. Your data matrix has 10 columns, but its rank might be close to 3. This has two major (and useful) implications:

- Principal Component Analysis (PCA): PCA finds new, uncorrelated variables (components). The number of meaningful components you can extract is exactly the rank of your standardized data matrix. It reveals the true number of dimensions needed to explain most of the variation.

- Linear Models and Hypothesis Testing: When testing complex hypotheses (e.g., in ANOVA or MANOVA), we compare models using matrices. The rank of these hypothesis matrices determines **the degrees of freedom** for the test. It essentially counts how many independent restrictions we are placing on our model parameters.

This leads to another very important **take-away**:

- Rank moves beyond just inversion. It quantifies the true, non-redundant information in your dataset, guiding dimensionality reduction and validating the structure of statistical tests.

Determining Matrix Rank

Remember that the rank of a matrix is the **maximum number of linearly independent rows or columns** in the matrix. In simpler terms, it tells us how many "truly unique" dimensions of information exist in your data, free from redundancy. For a data matrix, the rank has a practical interpretation: it's the **maximum number of uncorrelated variables** you could theoretically extract from your data.

Two columns are linearly dependent if one can be expressed as an exact linear combination of the others. When this happens, the matrix is rank deficient.

As an example, let us say we have data on 5 students with 4 variables:

Table **20.4.1**: Listing of the data

Student ID	Math Score (X_1)	Reading Score (X_2)	Total Score (X_3)	Double Math (X_4)
A	80	85	165	160
B	90	80	170	180
C	70	90	160	140
D	85	75	160	170
E	95	85	180	190

From this data, our 5×4 data matrix $\mathbf{X}$ is:

```
X = matrix( c(80, 85, 165, 160,
              90, 80, 170, 180,
              70, 90, 160, 140,
              85, 75, 160, 170,
              95, 85, 180, 190), ncol=4)
```

So, how do we determine the rank of this matrix? There are a few ways. Some easier than others. I will only show the first one in detail. The others can be found in your notes from Linear Algebra.

Method 1: Finding Dependencies

Looking at our variables and how they are defined, we see

1. Total Score (X_3) = Math Score (X_1) + Reading Score (X_2)

   165 = 80 + 85
   170 = 90 + 80
   So $X_3 = X_1 + X_2 \rightarrow$ a linearly dependent group

2. Double Math (X_4) = 2 × Math Score (X_1)

   160 = 2 × 80
   180 = 2 × 90
   So $X_4 = 2X_1 \rightarrow$ a linearly dependent group

So we have 4 columns, but:

- X_3 depends on X_1 and X_2
- X_4 depends on X_1

So, we really only have two variables giving information. Thus, rank $\mathbf{X} = 2$.

Mathematically, we can say

$$\sum_{i=1}^{4} a_n \mathbf{X}_n = 0 \tag{20.4.7}$$

for the vectors

$$a = [1, 1, -1, 0]'$$

$$a = [2, 0, 0, -1]'$$

Because we found (at least) one vector a, we know that $\mathbf{X}$ is rank deficient.

Note that these vectors are not unique. Any scalar multiple of these will give the same result.

Other Methods

The direct method is not the only one. While it makes it easy to show a matrix is rank deficient, it is not so good at determining the actual rank of the matrix. These are methods for determining the matrix rank more directly:

- Gaussian Elimination:
  The rank is the number of rows with non-zero leading element.
- Invertible Submatrices:
  The matrix's rank equals the size of the largest square sub-matrix with a non-zero determinant.
- Singular Value Decomposition:
  In practice, software uses SVD. The rank equals the number of non-zero singular values.

Rank in R

Of course, R has a built-in function to calculate the rank of a matrix. The function is `qr`. The function returns a lot of information, including the rank. If all you want is the rank of the matrix, just access rank from the output. Here is an example.

```
# Define a matrix
M <- matrix(c(1, 2, 3, 4, 5, 6, 7, 8, 9), ncol = 3)

# Calculate the rank
matrix_rank <- qr(M)$rank
print(matrix_rank) # Output: 2
```

Note that the `$rank` part of `qr(M)$rank` just returns the *rank* of the matrix. This is because `qr(M)` is a list-type variable with several variables included (just like with a data frame). The `$` accesses the variables internal to the list.

Positive Definite Matrices

A matrix $\mathbf{A}$ is a positive definite (pd) if $\mathbf{q}'\mathbf{A}\mathbf{q} > 0$ for all non-zero vectors $\mathbf{q}$. It is usually difficult to determine if a matrix is positive definite (pd). However, once you know it is, there are some important properties, which we look at in the next section.

Their Importance

So, why are positive definite matrices important?

Positive definite matrices are important because they guarantee the stability, uniqueness, and interpretability of many core multivariate procedures. They ensure that quadratic forms — such as those appearing in Mahalanobis distance, likelihood functions, and other optimization criteria — yield positive values for any non-zero vector, which is essential for meaningful variance estimation and probability calculations.

- In practical terms, the covariance matrix (of any non-degenerate random vector) **must be** positive definite; if it were not, it would imply zero or negative variance for some linear combination of the variables, which is nonsensical.

This property also underpins the feasibility of methods like Cholesky decomposition for simulation, the convergence of optimization algorithms in maximum likelihood estimation, and the validity of confidence ellipsoids in multivariate inference. In essence, positive definiteness acts as a mathematical certificate that a multivariate model is well-posed and that the underlying data structure is coherent and admissible.

Projection Matrices

A matrix $\mathbf{P}$ is a **projection matrix** if it is idempotent. The purpose of projection matrices is to project a higher space onto a subspace. If $\mathbf{P}$ is also symmetric, it is called an orthogonal projection matrix. This means it projects the larger space orthogonally (perpendicularly) onto the subspace. Think of shining a flashlight on a plant. If you put the flashlight directly over the plant, it will project the plant orthogonally onto the floor. If you do it at an angle, then the projection is called oblique. Here are comments on both types and why they matter to us.

Oblique Projections

A projection matrix is defined by the property $\mathbf{P}^2 = \mathbf{P}$ (idempotency). This simply means that once you project a vector onto a subspace, projecting it again doesn't change anything. An oblique projection satisfies this condition but does not require the projection to be at a *right angle*. You can project a vector onto a line or plane by "casting a shadow" from a non-right angle. The resulting projection matrix for an oblique projection is idempotent but *not* symmetric.

Orthogonal Projections

An orthogonal projection is a special case of a projection where the "shadow" is cast at a right angle onto the subspace. This is the type of projection most frequently used in statistics and linear algebra for tasks like least squares regression. For a matrix to be an orthogonal projection matrix, it must be both idempotent and symmetric. The symmetry condition is what guarantees that the projection is perpendicular.

Why do we care?

In linear regression, the famous "hat matrix" $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ is the orthogonal projection matrix that projects the observed data vector $\mathbf{Y}$ onto the column space of the design matrix $\mathbf{X}$ to obtain the fitted values $\hat{\mathbf{Y}}$.

> The key in both instances is that you are simplifying a complicated reality (3-D object) onto a simpler model (2-D shadow).

Three Special Numeric Matrices

There are three special matrices that are used in computer matrix calculations. These are interesting in themselves, and there are some exercises for you based on these.

- The vector $\mathbf{j}$ is a vector of 1s. It is used to calculate row sums (if pre-multiplying) or column sums (if post-multiplying).
- The matrix $\mathbf{J}$ is a matrix of 1s. It does what $\mathbf{j}$ does, but puts the sums in a matrix.
- The vector \mathbf{e_i} vector is a vector of 0s, with a 1 in the ith position. It is used in proofs, as it can be used to select an individual row, column, or element of a matrix.

Other Linear Algebra Topics

There are a few remaining topics that I cannot put elsewhere. Here they are

- Matrices $\mathbf{A}$ and $\mathbf{B}$ are **orthogonal**, $\mathbf{A} \perp \mathbf{B}$, if $\mathbf{A}'\mathbf{B} = \mathbf{0}$. That is, if the inner products of the columns of $\mathbf{A}$ and $\mathbf{B}$ are orthogonal, then the matrices themselves are orthogonal.
- The **eigenvalues** of a matrix $\mathbf{A}$ are those values λ that solve the equation $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$.
- The vector $\mathbf{v}$ corresponding to each eigenvalue is called its **eigenvector**.

This page titled 20.4: Other Terms and Operations is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

20.5: Consequences

The formal definitions of sample statistics, their distributions, and their relationship to population parameters are not merely academic distinctions — they are the very engine of statistical inference. From these foundational ideas flow consequences that govern how we learn from data. These consequences directly underpin the theory, diagnostics, and interpretation of linear models.

This section explores the most important of these implications, including the behavior of estimators, the decomposition of variance, the derivation of standard errors, and the theoretical justification for key assumptions. Understanding these logical outcomes is what transforms simple calculation into reliable and defensible statistical analysis.

✦•·····• ✦ •·····•✦

Basic Matrix Results

First, when taking the **transpose of a product**, you switch the order of the multiplication:

$$(\mathbf{AB})' = \mathbf{B}'\mathbf{A}' \tag{20.5.1}$$

A similar result holds with taking the **transpose of inverses**. The only difference is that all *three* inverses must exist:

$$(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1} \tag{20.5.2}$$

Lemma 20.5.1

For any matrix $\mathbf{X}$, the matrix $\mathbf{X}'\mathbf{X}$ is symmetric.

Proof.
I leave the proof as an exercise for you.

If you know the determinant of a matrix, you can easily calculate the **determinant of a scalar multiple** of that matrix.

Lemma 20.5.2

If $c \in \mathbb{R}$ and $\mathbf{A} \in \mathcal{M}_n$, then $\det c\mathbf{A} = c^n \det \mathbf{A}$.

There is a similar result with the **determinant of a trace** of a matrix.

Lemma 20.5.3

If $c \in \mathbb{R}$ and $\mathbf{A} \in \mathcal{M}_n$, then $\operatorname{tr} c\mathbf{A} = c \operatorname{tr} \mathbf{A}$.

Positive Definite Matrices

There are a lot of interesting properties of positive definite (pd) matrices. So, let us break these into a separate subsection.

Lemma 20.5.4

The diagonal elements of a pd matrix are all positive. That is, let $\mathbf{A} \in \mathcal{M}_n$ be positive definite, then $a_{ii} > 0, \ \forall i \in \{1, 2, \dots, n\}$.

Proof.
This can easily be shown by letting the $\mathbf{q}'$ row vector be $\mathbf{e}_i$. The quadratic form $\mathbf{q}'\mathbf{A}\mathbf{q}$ would therefore equal the diagonal element at position i. Since $\mathbf{A}$ is positive definite, that element must be greater than 0.

■

Note that the *converse* is not true, in general. Just because the diagonal elements are all positive does not mean that the matrix is positive definite. For an example, note

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \tag{20.5.3}$$

is not positive definite. To see this, its determinant is 0.

A few additional results:

- The **inverse of a pd matrix** is also pd. That is, if $\mathbf{A}$ is positive definite, then so is $\mathbf{A}^{-1}$.
- Since the determinant of a pd matrix is positive, all of the **eigenvalues are positive**. And, since all of the diagonal elements of a pd matrix are positive, then the **trace** is positive. Since the trace is used to calculate the degrees of freedom, the matrix must be positive definite.
- If $\mathbf{X}$ is a **full rank matrix**, even if not square, $\mathbf{X}'\mathbf{X}$ is positive definite.
- And, most importantly, the **covariance matrix** is positive definite if the design matrix, $\mathbf{X}$, is full rank. Otherwise, it is positive semi-definite and has a determinant of zero.

This page titled 20.5: Consequences is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

20.6: Statistics in Matrices

While the formulas you learned in introductory statistics are perfectly suited for calculation, a matrix formulation offers a more general and powerful perspective that scales to advanced methods like linear models. This **optional section** re-expresses fundamental sample statistics — such as the mean, variance, and covariance — using matrix notation.

The **purpose** is twofold: to provide essential practice with matrix operations, and to reveal the elegant underlying structure that unifies these concepts. In what follows, we will represent a data sample of size n as a column vector $\mathbf{Y}$. Through this lens, familiar quantities like the sum of squares emerge naturally as specific quadratic forms, building a critical bridge between basic statistics and the linear algebra that drives modern data analysis.

✦•·····• ✦ •·····•✦

Lemma 20.6.1

The sample mean using matrices: $\overline{Y} = \frac{1}{n}\mathbf{j}'\mathbf{Y}$.

Proof.

$$\frac{1}{n}\mathbf{j}'\mathbf{Y} = \frac{1}{n}\begin{bmatrix}1 & 1 & 1 & \cdots & 1\end{bmatrix}\begin{bmatrix}Y_1\\Y_2\\Y_3\\\vdots\\Y_n\end{bmatrix} \tag{20.6.1}$$

$$= \frac{1}{n}\left(1Y_1 + 1Y_2 + 1Y_3 + \cdots + 1Y_n\right) \tag{20.6.2}$$

$$= \frac{1}{n}\sum_{i=1}^{n} Y_i \tag{20.6.3}$$

$$= \overline{Y} \tag{20.6.4}$$

■

Theorem 20.6.2

The sum of squared values using matrices: $\mathbf{Y}'\mathbf{Y}$.

Proof.

$$\mathbf{Y}'\mathbf{Y} = \begin{bmatrix}y_1 & y_2 & y_3 & \cdots & y_n\end{bmatrix}\begin{bmatrix}y_1\\y_2\\y_3\\\vdots\\y_n\end{bmatrix} \tag{20.6.5}$$

$$= \left(y_1y_1 + y_2y_2 + y_3y_3 + \cdots + y_ny_n\right) \tag{20.6.6}$$

$$= \sum_{i=1}^{n} y_i^2 \tag{20.6.7}$$

■

Lemma 20.6.3

The sample variance using matrices: $s_y^2 = \frac{1}{n-1}\left(\mathbf{Y} - \overline{Y}\mathbf{j}\right)'\left(\mathbf{Y} - \overline{Y}\mathbf{j}\right)$.

Proof.

$$\frac{1}{n-1}\left(\mathbf{Y} - \overline{Y}\mathbf{j}\right)'\left(\mathbf{Y} - \overline{Y}\mathbf{j}\right) \tag{20.6.8}$$

$$= \frac{1}{n-1}\begin{bmatrix} y_1 - \overline{Y}, & y_2 - \overline{Y}, & y_3 - \overline{Y}, & \cdots, & y_n - \overline{Y}, \end{bmatrix}\begin{bmatrix} y_1 - \overline{Y} \\ y_2 - \overline{Y} \\ y_3 - \overline{Y} \\ \cdots \\ y_n - \overline{Y} \end{bmatrix} \tag{20.6.9}$$

$$= \frac{1}{n-1}(y_1 - \overline{Y})(y_1 - \overline{Y}) + (y_2 - \overline{Y})(y_2 - \overline{Y}) + \cdots + (y_n - \overline{Y})(y_n - \overline{Y}) \tag{20.6.10}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \overline{Y})^2 \tag{20.6.11}$$

■

Lemma 20.6.4

The sample covariance using matrices: $s_{xy} = \frac{1}{n-1}\left(\mathbf{Y} - \overline{Y}\mathbf{j}\right)'(\mathbf{X} - \bar{x}\mathbf{j})$.

Proof.

This proof echoes the previous proof.

$$\frac{1}{n-1}\left(\mathbf{Y} - \overline{Y}\mathbf{j}\right)'(\mathbf{X} - \bar{x}\mathbf{j}) \tag{20.6.12}$$

$$= \frac{1}{n-1}\begin{bmatrix} y_1 - \overline{Y}, & y_2 - \overline{Y}, & y_3 - \overline{Y}, & \cdots, & y_n - \overline{Y}, \end{bmatrix}\begin{bmatrix} x_1 - \bar{x} \\ x_2 - \bar{x} \\ x_3 - \bar{x} \\ \cdots \\ x_n - \bar{x} \end{bmatrix} \tag{20.6.13}$$

$$= \frac{1}{n-1}(y_1 - \overline{Y})(x_1 - \bar{x}) + (y_2 - \overline{Y})(x_2 - \bar{x}) + \cdots + (y_n - \overline{Y})(x_n - \bar{x}) \tag{20.6.14}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \overline{Y})(x_i - \bar{x}) \tag{20.6.15}$$

■

Note that this notation leads to the synonym $s_y^2 = s_{yy}$. It also leads to a nice proof that the covariance matrix is symmetric.

Definition: Variance-Covariance Matrix

Let $\mathbf{Y}$ be a random vector (a column vector whose elements are random variables). The quantity $V[\mathbf{Y}]$ is called the variance-covariance matrix of $\mathbf{Y}$.

Note that it is often just called the covariance matrix of $\mathbf{Y}$.

Lemma 20.6.5

Let $\mathbf{Y} \in \mathscr{M}_{r,c}$. If $\mathbf{Y}' = [\mathbf{Y_1}, \mathbf{Y_2}, \mathbf{Y_3}, \cdots, \mathbf{Y_r}]$, then the elements of $V[\mathbf{Y}]$ are $[\sigma_{ij}]$, where σ_{ij} is the covariance between Y_i and Y_j and $\sigma_{i,i}$ is the variance of Y_i.

Lemma 20.6.6

Covariance matrices are symmetric.

Lemma 20.6.7

If $\mathbf{Y}$ is a random vector and $\mathbf{X}$ is not, then

$$V[\mathbf{X}'\mathbf{Y}] = \mathbf{X}'V[\mathbf{Y}]\mathbf{X} \tag{20.6.16}$$

This assumes that the multiplication makes sense (the matrices are commensurate).

This page titled 20.6: Statistics in Matrices is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

20.7: End-of-Appendix Materials

Exercises

1. Prove $\mathbf{A} + \mathbf{0} = \mathbf{0} + \mathbf{A} = \mathbf{A}$.
2. Prove matrix addition is commutative.
3. Prove matrix addition is associative.
4. Prove that scalar multiplication is associative.
5. Prove that scalar multiplication is distributive over addition.
6. Using a counterexample, prove that matrix multiplication is not commutative when only one of the matrices is diagonal (thus showing that \emph{both} must be diagonal).
7. Let $\mathbf{A}$ be any square matrix. Show that $\frac{(\mathbf{A}+\mathbf{A}')}{2}$ is symmetric.
8. Determine the determinant of $\mathbf{J}_{10}$.
9. Determine the rank of $\mathbf{J}_{10}$.
10. Prove that $\mathbf{j}_{10}\,\mathbf{j}_{10}'$ is not positive definite.
11. Prove that $\mathbf{j}_{10}'\,\mathbf{j}_{10}$ \emph{is} positive definite.
12. Prove $(\mathbf{AB})' = \mathbf{B}'\mathbf{A}'$ from Section 20.5.
13. Prove $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$ from Section 20.5.
14. Prove Lemma 20.5.1.
15. Prove Lemma 20.5.2.
16. Prove Lemma 20.5.3.
17. Prove Lemma 20.6.5.
18. Prove Lemma 20.6.6.
19. Prove Lemma 20.6.7.

This page titled 20.7: End-of-Appendix Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

CHAPTER OVERVIEW

21: The Appendix of R



A deep understanding of statistics and randomness is not merely absorbed — it is built through **active experimentation**. The true power of concepts like sampling distributions, variance, and inference reveals itself when you can simulate them, manipulate data, and visualize results in real time. This appendix introduces you to the R Statistical Environment, a premier tool for this very purpose. R is more than just software; it is a comprehensive platform for statistical computing and graphics, designed to turn theory into insight.

R is exceptionally well-suited for statistics and data science for several key reasons. First, it is open-source and free, ensuring broad accessibility and a massive, collaborative global community. Second, it was **built by statisticians, for statisticians**, meaning its core design and thousands of specialized packages (like those for linear models `lmtest` and `car`, `ggplot2` for visualization, and `dplyr` for data manipulation) are tailored directly to the needs of statistical analysis. Third, its powerful scripting nature makes reproducibility and automation central to your analysis — every analysis can be documented and rerun exactly, which is a cornerstone of rigorous scientific practice.

The purpose of this appendix is to provide foundational support for using R. We will begin with practical setup in a "How to Install" section, then progress to core programming concepts that demonstrate R's power. By the end, you will be equipped to use R not just as a calculator, but as a dynamic laboratory for exploring the wonder of statistics firsthand.

21.1: Installing R
21.2: R Packages
21.3: R Functions
21.4: Programming Practice
21.4.1: Plotting a Distribution
21.4.2: Functions of RVs
21.4.3: Effects of Rounding
21.4.4: Algorithmic Thinking
21.4.5: Distribution of Extreme Values
21.4.6: Non-Parametric Bootstrap
21.4.7: Distribution of Sample Means
21.4.8: Distribution of Sample Sums

This page titled 21: The Appendix of R is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.1: Installing R

The first step in using the R Statistical Environment is to install it on your computer. The process is relatively straight-forward. You download the installation program, then run the installation program.

You can find the installation program at the CRAN website:

https://cran.r-project.org/



Once there, download the appropriate installation program for your specific type of computer. Once it is downloaded, find the installation program and run it. Selecting all of the default options is fine. It's what I do.

The specifics are dependent on the type of computer operating system, so I will not get into the specifics. Let me point you in the direction of an Internet search engine and/or a site with a lot of videos. Learning how to access the information in these sites is a skill in itself.

So, enjoy the exploration and the learning.

R Studio and Other IDEs

Some prefer to install both R *and* an integrated developer environment (IDE). The most common IDE for R is called R Studio. The URL to access R Studio is



https://posit.co/download/rstudio-desktop/

Note that I use the IDE that comes with the usual R download. It does everything I need. In fact, I find the special features of most IDEs useless at best, and counter-productive at worst.

But, feel free to use them.

Note

R Studio *runs* R. It is not a different programming language. It is merely a IDE that you may prefer to use to ease your studies into statistics. There are many other IDEs available, including some that are generic enough to use with other programming languages.

This page titled 21.1: Installing R is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.2: R Packages

R is free software. In this case, that means two things: it costs nothing to obtain and you are able to modify it as you see fit. this modification can take the form of incorporating the code into an IDE. More frequently, it takes the form of creating packages that extend the basic functionality of R. As of this moment, while I am writing this, there are 23,035 packages available at CRAN.



R packages can be found in a few places: CRAN is the usual place to locate them. However, there are several on GitHub. Also, many focused on bioinformatics and computational biology, can be found at Bioconductor.

Why **the three repositories**?

CRAN is the *official* repository for all R packages.

Bioconductor is a dedicated repository for R packages focused on these areas, offering a wide range of tools and pipelines for analyzing -omics data, such as genomics, transcriptomics, and proteomics. While Bioconductor is the main source for bioinformatics-related R packages, it is important to note that the broader Comprehensive R Archive Network (CRAN) also hosts many R packages, some of which may be relevant to biological data analysis.

Platforms like **GitHub** are frequently used by developers (like me) to host cutting-edge or in-development versions of R packages, including those for biological applications.

While the base R that you installed has many statistical features, it is important to learn how to extend that base R installation. One way to do this is to install and load packages (two *separate* processes).



This book relies on these packages:

- KnoxStats
- car
- lawstat
- lmtest
- MASS
- nnet
- quantreg
- VGAM

Note that this book is *not* a part of the so-called tidyverse. Perhaps a future edition will be, but not yet, not now.



Installing Packages...

To use a package in R, you need to do two things. **First**, you need to *install* the package to your computer. This only needs to be done once for your computer. **Second**, you need to *load* the package, which happens in each script in which you use the package.

So, how do you install a package...

... from CRAN

The overall scheme for installing packages is the same for each operating system. The specifics are slightly different to take advantage of what the operating systems allow. That scheme is

1. Run the `install.packages` function
2. Select a place to download the package from
3. ????
4. Profit

For instance, if I want to install the `car` package, I would run the following code in the Console window

```
install.packages("car")
```

Running this will download the `car` package from the Internet (the mirror site you selected) and install it on your computer. [If you have already specified the mirror site, then step two will not happen.]

It will also download all packages that are needed to run the `car` functions (called "dependencies" because `car` depends on them). You can see what other packages are installed by watching the Console window.



... from GitHub

The idea is the same as above, but R needs to use the `devtools` package to install packages from GitHub.

1. If you do *not* have `devtools` already installed, run this in the Console:

```
install.packages("devtools")
```

If you *do* have `devtools` already installed, skip that step.

2. Now, run the following in the Console window

```
devtools::install_github("ojforsberg/KnoxStats")
```

A lot will happen, but you can just sit and watch it all happen. Once the Console window is done doing its thing, that package (`KnoxStats` , taken from the `ojforsberg` repository) is installed on your computer.

The code to install other packages from GitHub will differ in the repository and package. Most GitHub repositories are well documented, it should have the code available.

... from Bioconductor

While the Comprehensive R Archive Network (CRAN) hosts thousands of essential packages, some specialized tools — particularly for bioinformatics, genomic data analysis, and high-throughput sequencing — are developed and maintained through **Bioconductor**. This open-source project provides rigorously curated, well-documented packages tailored for biological data science. Installing packages from Bioconductor requires a slightly different procedure than the standard `install.packages()` command used for CRAN.

The process has been streamlined through the `BiocManager` package. To begin, you must first install `BiocManager` from CRAN. Once installed, you can use it to install *any package* available in the Bioconductor repository. This method ensures that all necessary dependencies are correctly resolved and that you receive a version of the package compatible with your current version of R.

Below is the standard workflow. It is recommended to run these commands directly in your R console, not within a function or a larger script, especially the first time you set up Bioconductor.

```
# First, install the BiocManager package from CRAN
install.packages("BiocManager")

# Then, use BiocManager to install any desired Bioconductor package.
```

```
# Replace 'packagename' with the actual name of the package you need.
BiocManager::install("packagename")

# Example: To install the core 'limma' package for differential expression analysis:
BiocManager::install("limma")
```

When you run `BiocManager::install()` for the first time, you may be prompted to update some previously installed packages. You can typically proceed by typing `y` (for "yes") in the console. To install a specific version of a package or to explore the full catalog, visit the official Bioconductor website.

This method integrates Bioconductor's powerful, domain-specific tools seamlessly into your R workflow, greatly expanding your analytical capabilities for specialized statistical challenges.

> **Note**
>
> Nothing in *this* course requires anything from Bioconductor. However, if you are interested in bioinformatics, I encourage you to read their website. There is a lot of information there that you should find interesting!

Loading Packages

The installation needs to be done **just once** for your computer. However, if you need to use the package in a script, you will need to **load** it into the working memory. Since this is something tied to the script, you should have this line in your script if you are using the `car` package:

```
library("car")
```

Once that line is run, R is able to access all functions (and/or data) in the package. Usually, there are a lot of functions in a package. To see what is available in the package, run this in the Console window:

```
help(package="car")
```

After running this, a page will pop up showing all available functions and links to their help pages. This particular package has a lot of functions associated with it. Some other packages have just a couple. The number of functions depends on the purpose of the package.

This page titled 21.2: R Packages is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.3: R Functions

There are a plethora of functions available in R. The key is to use functions to become familiar with them. Also, because you will always need to use new functions, it is very important to be familiar with the R help files for the functions.



Basic Functions

The following are basic functions in R. You will use these frequently.

Function	Description
`source()`	Run an external R script
`head()`	Show the first six elements in the vector
`tail()`	Show the last six elements in the vector
`length()`	Calculate the length of the vector
`seq()`	Create a sequence of values
`summary()`	Print the six-number summary
`mean()`	Calculate the arithmetic mean
`median()`	Calculate the median
`sum()`	Calculate the sum of the elements in the vector
`sd()`	Calculate the standard deviation
`var()`	Calculate the variance

Please know what each does. If necessary, use the help file for the given function. The more you familiarize yourself with the help files, the more they will tell you.

Matrix Functions

R can also do arithmetic on matrices. Since the internal computer calculations are all based on matrices, it is important to be familiar with matrix operations to make sure you know what R is doing (can check the output).

Element-wise Functions

`+`	usual matrix addition
`*`	Hadamard product
`^`	exponentiation

Additional Matrix Functions

`%*%`	usual matrix product
`t()`	transpose
`solve()`	matrix inverse

Probability Functions

At the core of statistics is probability and probability distributions. These will be important in helping you better understand the effects of randomness on the estimates... and the effects of violating procedure requirements on those estimates.

Generic Functions

`set.seed()`	set the specified random number seed
`sample()`	randomly sample from a vector. This can also be used to sample from a discrete Uniform distribution

Probability Function Naming Logic

Each named probability function in R can be parsed into two parts, the stem and the prefix. The **stem** specifies the probability distribution, whereas the **prefix** specifies what aspect of the distribution you wish to access.

This is an exhaustive list of the standard **prefixes**:

`d`	specifies the likelihood value. If the distribution is discrete, then this will also be the probability, otherwise it is the density.
`p`	specifies the *cumulative* probability, P[X ≤ x]. This prefix will rarely be available for multivariate functions.
`q`	specifies the quantile, the value of x that produces the probability. This prefix will rarely be available for multivariate functions.
`r`	specifies a random value. In simulation, this is the most important prefix, as it produces a random sample of a given size from the specified distribution.

The second part is the **stem**. This specifies the distribution involved. Here is a list of some of the more interesting stems available:

`binom`	Binomial distribution. One needs to specify the number of trials, `size` , and the success probability, `prob` .
`cauchy`	Cauchy distribution. Optionally, one can specify the location and the spread. The default is the standard Cauchy with `location` of 0 and `scale` of 1.
`chisq`	Chi-squared distribution. One needs to specify the number of degrees of freedom, `df` .
`exp`	Exponential distribution. One needs to specify the `rate` , λ.
`f`	Snedecor's F distribution. One needs to specify both degrees of freedom, with the numerator preceding the denominator degrees of freedom, `df1` and `df2` .
`gamma`	Gamma distribution. One *must* specify the `shape` parameter, α, since there is no default. The `rate` parameter has a default value of β = 1.
`norm`	Normal (Gaussian) distribution. Optionally, one can specify the mean `m` and standard deviation `s` of the Normal. This defaults to a standard Normal distribution.
`pois`	Poisson distribution. One needs to specify the expected value, `lambda` , λ.
`t`	Student's t distribution. One needs to also specify the number of degrees of freedom, `df` .
`unif`	Continuous Uniform distribution. Optionally, one can specify the minimum and maximum value. The default is the standard Uniform with `min` of 0 and `max` of 1.

Testing Functions

Because R is a statistical program, it is able to perform all of the basic statistical tests and procedures. These are the related functions.

`aov`	Analysis of Variance procedure
`lm`	OLS regression
`summary.aov`	ANOVA table of a linear model
`summary.lm`	Regression table of a linear model
`residuals`	calculated residuals from a model
`confint`	confidence interval for estimated parameters
`predict`	estimation and prediction of a value

Non-Base Testing Functions

`runs.test`	`KnoxStats`	the runs test
`hetero.test`	`KnoxStats`	univariate test of heteroskedasticity
`fligner.test`	`stats`	test of heteroskedasticity across groups
`bptest`	`lmtest`	Breusch-Pagan test of heteroskedasticity

Control Functions

`for`	for-loops
`if`	if-then statements
`ifelse`	ternary operator
`numeric`	creates a vector in memory

Graphical Functions

R is a full-fledged graphical system. In fact, this is what set R apart from its competitors (and still does!). Every pixel of a graphic can be modified in R. This book relies on the basic R graphic engine. There are two other graphical engines (metaphors): `grid`

and `ggplot2` . Base-R graphics will always serve you. With that being said, `ggplot2` is the modern graphics engine for R. It serves as a wrapper for the basic graphics, making some graphics much easier to create.

`qqnorm`	Q-Q plot for a Normal target
`qqline`	plots the diagonal line in a normal Q-Q plot
`barplot`	bar chart
`boxplot`	box-and-whiskers plot
`hist`	histogram
`histogram`	histogram*
`overlay`	histogram with a distribution overlaying it*
`plot`	plots a basic graphic
`par`	sets the parameters for the future graphic
`plot.new`	starts a new graphic
`plot.window`	sets the size of a new window
`axis`	creates axes
`title`	plots a title on the specified axis
`points`	plots points on the graphic
`lines`	plots lines on the graphic

* Those graphical functions marked with an asterisk are found in the KnoxStats package.

This page titled 21.3: R Functions is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

SECTION OVERVIEW

21.4: Programming Practice

This section provides a series of practical examples to help you apply statistical concepts using the R Statistical Environment, one of the most versatile programming languages for statistical analysis and data visualization. R is particularly well-suited for solving a wide range of statistical problems, from basic descriptive statistics to advanced modeling techniques. As you progress through these examples, you will gain hands-on experience with R's powerful functions, enabling you to approach upcoming statistical challenges with confidence (pun intended).



I designed each example in this section to demonstrate the practical application of key statistical methods while also showcasing the capabilities of R. To help you fully understand the solutions, each example includes explanations, annotated code snippets, and discussions of the outputs. Even if you are new to R, the examples are structured to quickly build your proficiency with the language.

By working through these examples, you will not only deepen your understanding of statistical methods but also start developing a toolkit of practical R skills. Whether you aim to analyze data for research, business, or personal projects, these examples will equip you with the knowledge and techniques needed to leverage R effectively. Take your time to experiment with the code, explore variations, and see how the results change — this active learning approach will enhance both your statistical intuition and your programming expertise.

R Programming Examples

21.4.1: Plotting a Distribution

21.4.2: Functions of RVs

21.4.3: Effects of Rounding

21.4.4: Algorithmic Thinking

21.4.5: Distribution of Extreme Values

21.4.6: Non-Parametric Bootstrap

21.4.7: Distribution of Sample Means

21.4.8: Distribution of Sample Sums

This page titled 21.4: Programming Practice is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.4.1: Plotting a Distribution

Probability distributions sit at the center of statistics. We use them to describe the errors around us. Because of this, it is helpful to be able to "see" the distribution. Keeping a mental image of the distributions allows us greater insight into how they arise and the consequences of their randomness.

Problem:

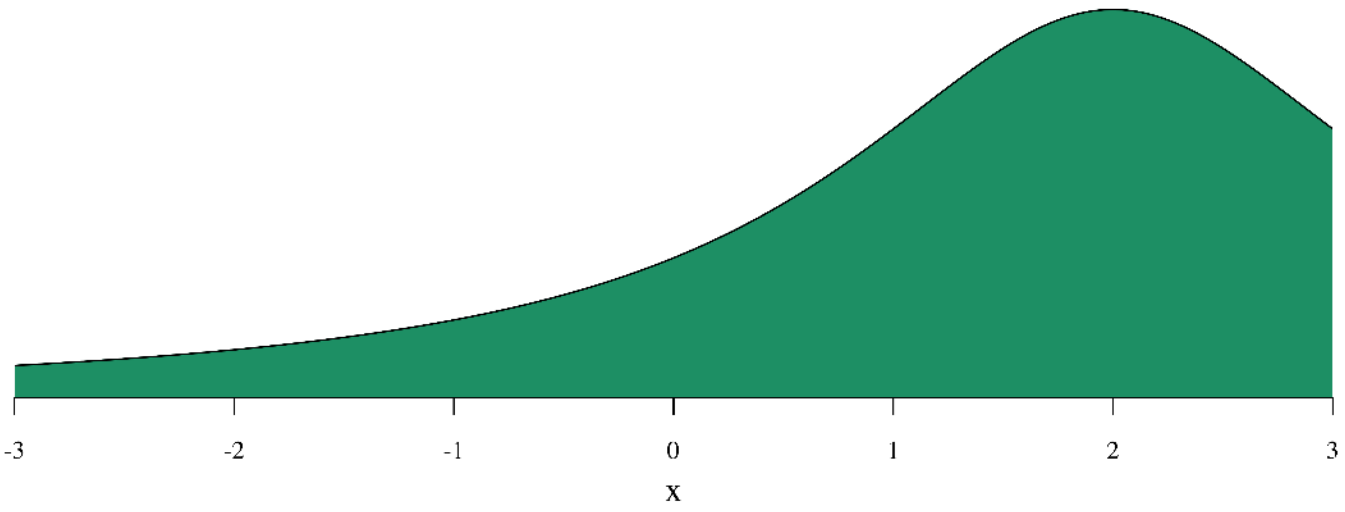
Produce a basic density plot of a Cauchy distribution between -3 and +3, where the Cauchy is centered at x=2 and has an inter-quartile range (IQR) of 3.

Solution:

Since we are working with a probability distribution, let us refer to Section 21.3: R Functions. The function to calculate the density of the Cauchy centered at 2 with IQR 3 is `dcauchy(x,2,1.5)` . Thus, some code to produce a basic plot between -3 and +3 is

```
x = seq(-3,3,length=1000)
y = dcauchy(x, location=2, scale=1.5)
plot(x,y)
```

With a little bit of work, you can make a graphic like this.



The fill color is the green in a pallette of three colors designed to be safe for those with the usual color blindness, #1b9e77. When possible, it is best to accommodate those with color blindness and those who print out the graphic in shades of grey.

Note

The three safe colors are Green (#1b9e77), Orange (#d95f02), and Blue (#7570b3). These colors are from the Color Brewer site. There are other options for such color-blindness-safe colors. Just stay aware of your audience.

Extension:

Plot the pdf of a standard Normal distribution from -3 to +3 on the same graphic as a standard Cauchy. Looking at the two distributions, which has a higher variance?

This page titled 21.4.1: Plotting a Distribution is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.4.2: Functions of RVs

If you cannot derive the exact distribution of a statistic mathematically, then you cannot analyze its behavior — unless you simulate it. Simulation lets you generate the sampling distribution empirically, allowing you to estimate bias, variance, and probabilities without solving complex equations. It is a practical — and powerful — workaround when math falls short.

Problem:

Estimate a density graph of the volume of a cylinder with radius following a standard Uniform distribution and a height following a Normal distribution with mean 10 and standard deviation 0.1.

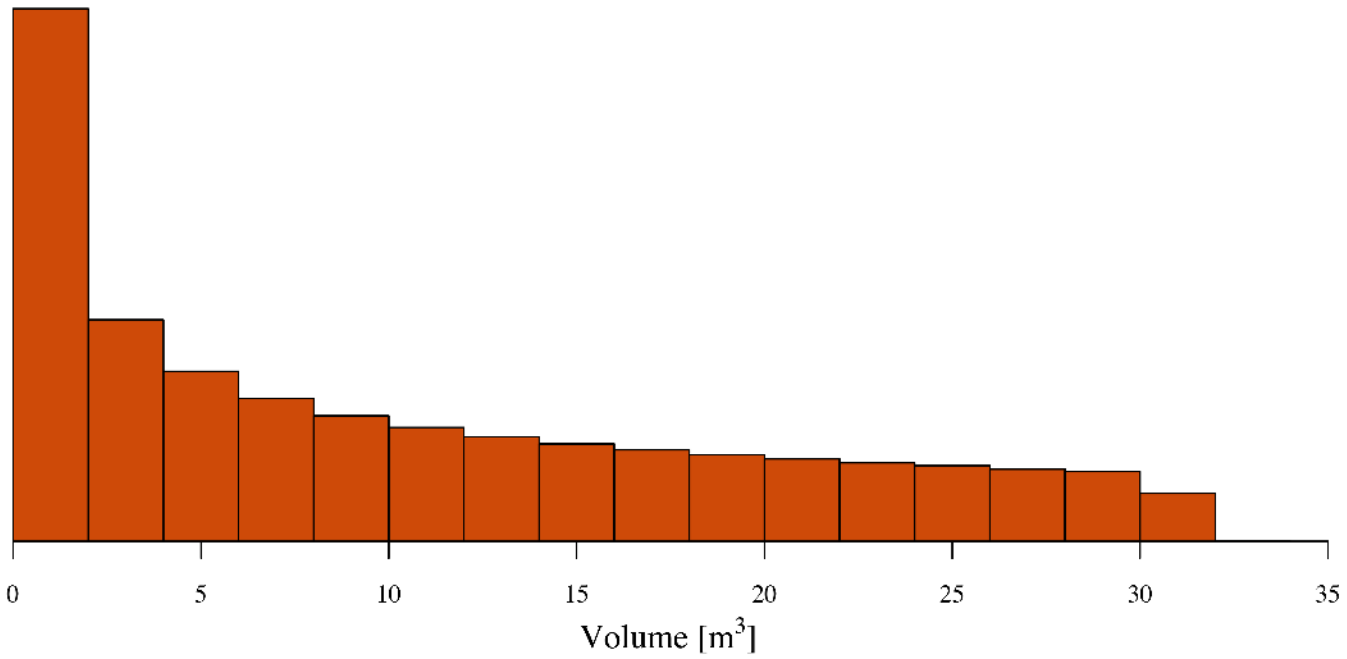
Solution:

Since we are working with a probability distribution, I again refer you to Section 21.3: R Functions.

```
radius = runif(n=1e6)
height = rnorm(n=1e6, m=10, s=0.1)
volume = pi * radius^2 * height

hist(volume)
```

This is an example where using simulation easily obtains the approximate distribution. After using my R skills, I obtained the histogram below. See how close you can come to this.



The orange color (#d95f02) is also color-safe for those with the most likely form of colorblindness.

Extension:

Given the same heights and radii, find the distribution of the surface area of a cylinder with a bottom, but not a top.

This page titled 21.4.2: Functions of RVs is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.4.3: Effects of Rounding

Rounding can significantly affect statistical decisions because it alters the precision of numerical data (the inputs), which can lead to changes in calculated values such as means, variances, or p-values (the outputs). In hypothesis testing, for instance, rounding may cause a test statistic to fall on the border of a critical value, potentially shifting the conclusion about rejecting or failing to reject the null hypothesis. Similarly, in regression analysis, rounding predictor or response variables can affect the accuracy (and precision) of parameter estimates and the overall model fit.

These impacts are particularly pronounced in datasets with small sample sizes or values that are close to decision thresholds. Therefore, careful consideration of rounding practices is essential to ensure that statistical conclusions remain valid and reliable. This example explores the effects of rounding on the one-sample t-test.

This is another example of simulation.

Problem:

Determine the effect of rounding on the appropriateness of the one-sample t-test.

Solution:

This leaves a lot of decisions to us. The one-sample t-test requires data being generated from Normal distribution. So, let's generate data from a Normal(5, 1) distribution. We need to round the data, so we will need to use the `round` function. Finally, we will need to determine if the distribution of the resulting p-values is standard Uniform distribution.

The following code does this. Make sure you understand every piece of this code and why the output tests the effects of rounding.

```
pval = numeric()

for(i in 1:1e4) {
 x = rnorm(10, m=5, s=1)
 y = round(x)
 pval[i] = t.test(y, mu=5)$p.value
}

ks.test(pval,"punif")
binom.test(sum(pval<0.05), n=length(pval), p=0.05)
```

The Kolmogorov-Smirnov test indicates that the distribution of the p-values is not standard Uniform. Thus, rounding in this situation breaks the t-test. Note that if we only care about $\alpha = 0.05$, we would use the Binomial test results. Given that p-value, I would still conclude that I should not use the t-test in this situation.

What about increasing the sample size from 10 to 50? The Kolmogorov-Smirnov test still indicates that the test is no longer acceptable. Note that if all we care about is $\alpha = 0.05$, then the t-test *does* appear to be appropriate under these conditions.

What about increasing the variability in the data? Having a wider spread to the data may make the rounding less important. Let's change the standard deviation from 1 to 10 (and return the sample size to 10). From my run, the distribution of the p-value is still not standard Uniform, but the rejection rate for $\alpha = 0.05$ is close enough to 0.05.

Extension:

What if we increase the sample size back to 50? Show that the distribution of the p-values is close enough to standard Uniform that the rounding does not affect the quality of the t-test conclusions.

This page titled 21.4.3: Effects of Rounding is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.4.4: Algorithmic Thinking

Algorithmic thinking is crucial in statistics as it enables a structured approach to solving complex problems by breaking them into smaller, logical steps. This mindset is particularly valuable when designing workflows for data analysis, from data preprocessing and visualization to modeling and interpretation. By thinking algorithmically, statisticians can systematically address challenges such as cleaning messy data, optimizing computational efficiency, or automating repetitive tasks.

Moreover, algorithmic thinking facilitates the translation of statistical concepts into code, allowing for reproducibility, scalability, and adaptability in analysis. In a field increasingly reliant on computational tools, developing this skill ensures that statisticians can efficiently tackle problems and adapt to new methods or technologies. This example illustrates algorithmic thinking to arrive at an interesting result.

Problem:

If the inter-arrival time is Exponentially distributed with average time of 20 minutes, then what is the distribution of the number of people who show up in an 8 hour period?

Solution:

This is a tough question but let's break it down into its parts, then simulate it.

First, let's simulate the inter-arrival time (the time between arrivals).

```
iat = rexp(100, rate=3)     ### 'iat' is in hours
```

From this, we can see that the total time between the 1st and the 20th arrival would be the sum of 20 of those inter-arrival times:

```
iat = rexp(100, rate=3)
sum(iat[1:20])
```

On the other hand, the *number* of arrivals in an hour would be

```
iat = rexp(100, rate=3)
iatCS = cumsum(iat)
max(which(iatCS <= 1))
```

So, the number of arrivals in *eight* hours would be

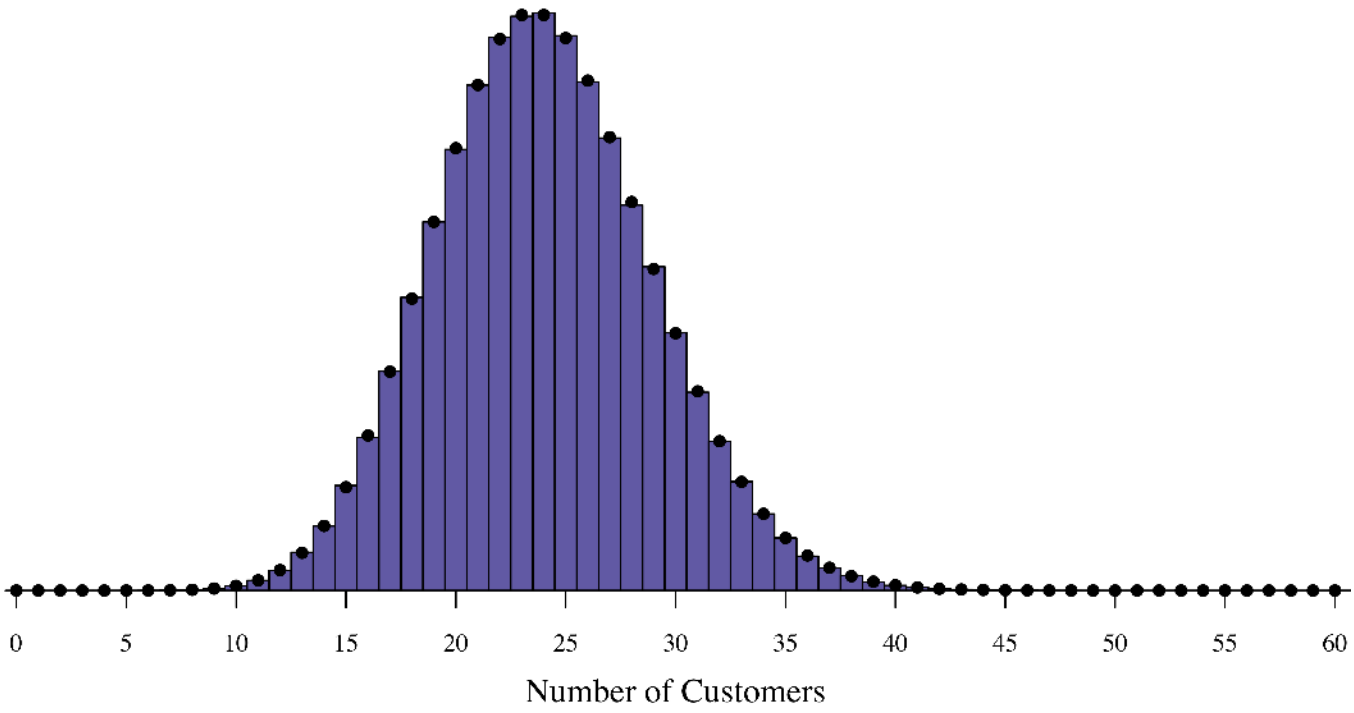
```
iat = rexp(100, rate=3)
iatCS = cumsum(iat)
max(which(iatCS <= 8))
```

To get the *distribution* of those "number of arrivals in eight hours," you just need to repeat the code **many times**, saving the number of arrivals each time:

```
arrNum = numeric()

for(i in 1:1e6) {
  iat = rexp(100, rate=3)
  iatCS = cumsum(iat)
  arrNum[i] = max(which(iatCS <= 8))
}
```

The histogram below shows the estimated distribution of the number of customers arriving in those 8 hours.



Closely examining the histogram *and* thinking about the distributions you should have already learned, it is clear that the number of arrivals in 8 hours follows this distribution

$$\text{Number of Arrivals} \sim \text{Poisson}(\lambda=24)$$

To make it even more manifest, overlaying the histogram with a graph of that Poisson distribution illustrates this (the dots on the histogram).

```
hist(arrNum, freq=FALSE, breaks=seq(1,50)-0.5)
points(1:50, dpois(1:50, lambda=24), pch=16)
```

Note that this is not *proof* of the relationship between the two distributions. It merely suggests the relationship. Your probability theory course will give you the tools to actually prove the relationship.

Extensions:

Extension 1: Change all of the time measurements in the previous example from hours to minutes. Make sure that the conclusions are the same.

Extension 2: Change the inter-arrival time to 10 minutes. What is the distribution of the number of arrivals in three hours? Show it using the histogram.

This page titled 21.4.4: Algorithmic Thinking is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.4.5: Distribution of Extreme Values

If we know the distribution of the lifetimes of each component, does this mean we know the distribution of the minimum lifetime? Sometimes. Note that sometimes we may be curious about the distribution of a *function* of random variables... which is also a random variable.

But, it goes beyond simple curiosity. Calculating the distribution of functions of random variables is essential because it allows us to understand how transformations or combinations of random variables behave and how their uncertainty propagates. This knowledge is fundamental in many applications, such as deriving the sampling distribution of an estimator, which forms the basis for hypothesis testing and constructing confidence intervals. Additionally, understanding the distribution of these functions enables probabilistic modeling in scenarios where direct distributions are unavailable, such as in operations research or risk management. It also helps in determining the likelihood of complex events, optimizing decision-making, and evaluating reliability in systems involving random inputs. By studying these distributions, statisticians can make more accurate predictions and draw meaningful inferences in both theoretical and applied contexts.

Problem:

A flashlight uses five batteries. The lifetime of each battery is independent and follows a Gamma distribution with mean 50 days and standard deviation 5 days (`shape` = 100, `scale` = 0.500). The flashlight will show light until the *first* battery dies.

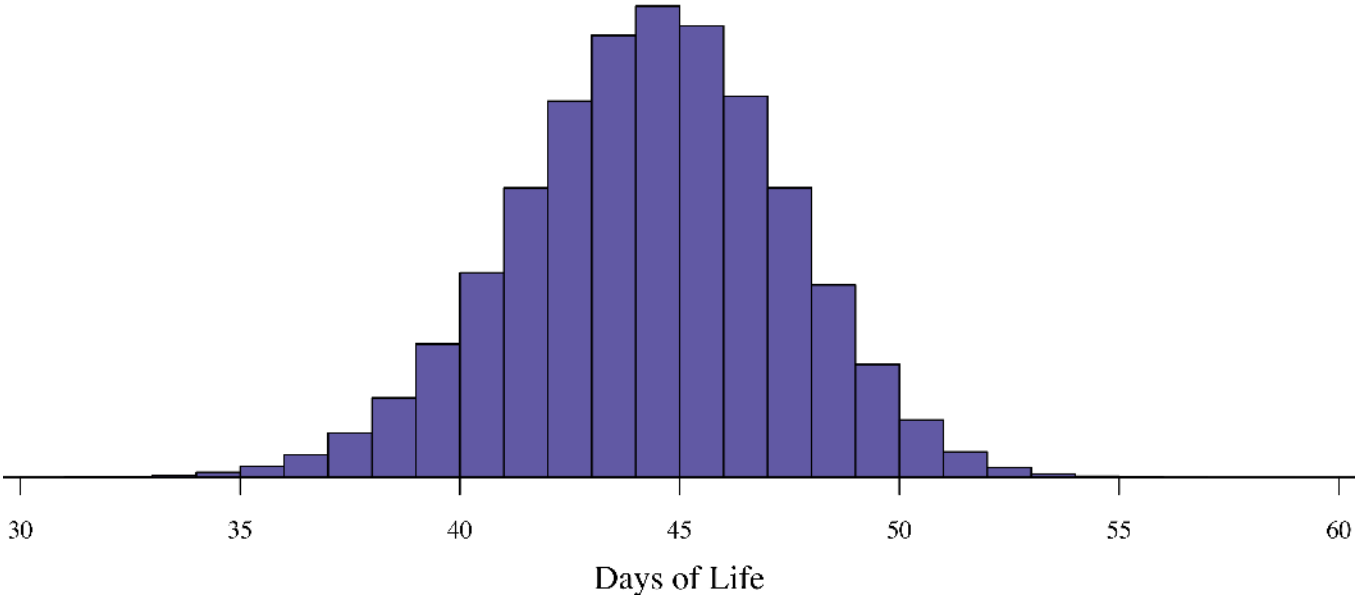
What is the expected time the flashlight will work after receiving five new batteries?

Solution:

This is a great place for you to think through the problem (algorithmic thinking). What is the first step modeling this physical event? What is the second? Etc.? Write out the steps... and the code... in your notebook. If done correctly, you will obtain the graphic at the bottom of the page.

Conclusions:

The question actually asked for the expected lifetime of the flashlight. The expected lifetime (mean) of the flashlight is 44.3 days, with a standard deviation of 3.1 days (sd). It might be nice to know that 90% of the flashlights survive between 39.1 and 49.2 days (quantile).



Extension:

King Rudolph wants to know the effect on flashlight lifetime if the batteries had a lifetime that followed a Gamma distribution with mean 50 days and standard deviation 2 days. What answer should I give to him?

This page titled 21.4.5: Distribution of Extreme Values is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.4.6: Non-Parametric Bootstrap

The **bootstrap** is a powerful and versatile tool in statistics because it provides a non-parametric method for estimating the sampling distribution of a statistic without requiring strong assumptions about the underlying population distribution. By repeatedly resampling with replacement from the observed data, the bootstrap allows statisticians to approximate the variability of estimators, construct confidence intervals, and conduct hypothesis tests.

This approach is especially valuable when theoretical distributions are difficult to derive, such as with complex estimators or small sample sizes. Furthermore, the bootstrap is widely applicable across diverse statistical methods, making it a critical tool for robustness and flexibility in real-world data analysis. Its ability to leverage computational power to provide insights where traditional methods may falter has made the bootstrap indispensable in modern statistics.

Problem:

A sample of n=10 counts is drawn from a population with unknown distributional characteristics. From the collected data, estimate a 95% confidence interval for the population mean.

Here are the observations:

4, 3, 4, 5, 4, 4, 1, 4, 9, 8, 3

Solution:

Note that the sample size is small in this example. As such, we cannot rely on the Central Limit Theorem to assume the sample mean is (close to) Normally distributed. We do not even know if the population has a finite variance. The only thing we know is that we collected that particular sample. Thus, we will use the bootstrap.

That is, we will treat the sample as being representative of the population. Without this assumption, there is absolutely nothing we can do. With that assumption, we can effectively (or essentially) "recreate" the population as being an infinite repetition of this data. Then, to estimate the variability in the population, we simply redraw a sample of the same size from that population.

This is called the ***non-parametric*** bootstrap because it does not assume a specific (named) distribution of the data.

Here is the code to accomplish this once.

```
theData = c(4, 3, 4, 5, 4, 4, 1, 4, 9, 8, 3)

newData = sample(theData, replace=TRUE)
mean(newData)
```

That code gives us just one sample mean. It takes thousands of them to understand the distribution (centers and variabilities).

Thus, the non-parametric bootstrapping code will be

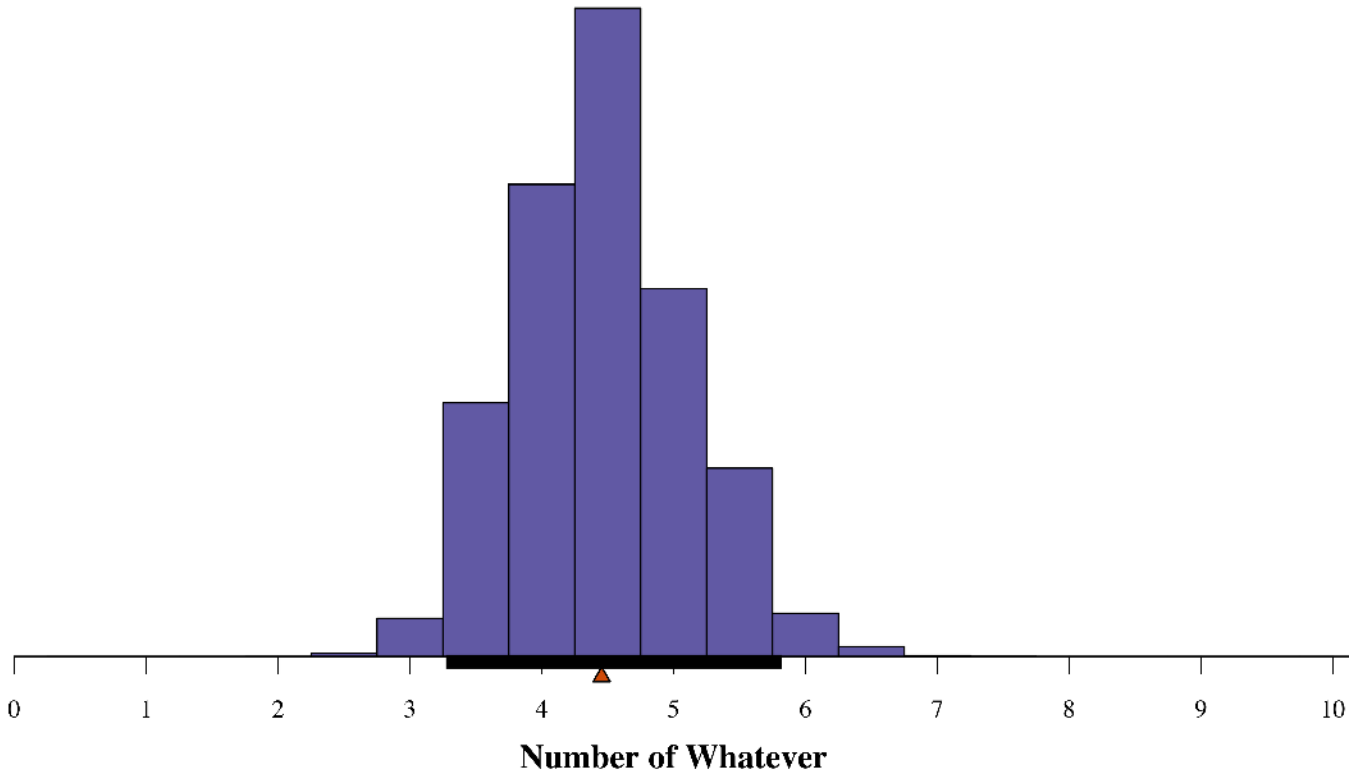
```
theData = c(4, 3, 4, 5, 4, 4, 1, 4, 9, 8, 3)
newMeans = numeric()

for(i in 1:1e6) {
  newData = sample(theData, replace=TRUE)
  newMeans[i] = mean(newData)
```

```
}

mean(newMeans)
sd(newMeans)
quantile(newMeans, c(0.025,0.975))
```

From this code, I estimate the population mean is 4.45, with an estimated 95% confidence interval from 3.27 to 5.82. The figure below shows the histogram of the sample means from the simulation. The triangle on the x-axis represents the mean of these sample means. The thick bar along the axis represents the estimated 95% confidence interval.



Note

The non-parametric bootstrap is used when all you know about the population is that you randomly selected the sample. It is a *very* flexible procedure that should be a part of your statistical toolbox.

However, statistically, it tends to be of low power (confidence intervals are too wide; p-values are too high). The reason for this lower power is that you are making fewer assumptions on the population. If you know more about the population and are able to use it, then that procedure will have higher power (all things being equal).

How Good *is* the Non-Parametric Bootstrap?

Problem:

How good is the non-parametric bootstrap in terms of *covering* the population mean? For this investigation, let's assume the population really is standard Normal.

Solution:

The first step is to draw a sample of size n=10 from that population, then do the non-parametric bootstrap, then determine how frequently the population mean is in the confidence interval. It should happen 95% of the time.

Here is the code:

```
estLCL = numeric()
estUCL = numeric()

for(j in 1:1e6) {     # The loop testing the NPB

 theData = rnorm(10)
 newMeans = numeric()

 for(i in 1:1e6) {.   # The loop performing the NPB
  newData = sample(theData, replace=TRUE)
  newMeans[i] = mean(newData)
 }

 estLCL[j] = quantile(newMeans,0.025)
 estUCL[j] = quantile(newMeans,0.975)
}
```

Note that this will take quite some time to run. There are a total of 1,000,000,000,000 iterations taking place. On a new laptop, one should probably expect it to run overnight. On an older one, it may take an entire day. However, the results will be rather precise.

Let me take this opportunity to (re-) introduce you to a measure of the quality of a confidence interval: **coverage**. Coverage is defined as the proportion of the time that the confidence interval contains the true mean.

For our example, the true mean is 0. So, the coverage will be the proportion of the time that 0 is between the two estimated confidence bounds.

```
mean( 0>estLCL & 0<estUCL )
```

When I run this, I obtain 0.896, which is *quite* different from the hoped-for 0.95. It means I am rejecting at a rate of 10.4% instead of the claimed 5%. This is not good.

Having **too low of a coverage** (as here) is problematic because it increases the risk that the true parameter value lies outside the confidence interval or prediction range. Coverage probability reflects the proportion of intervals that, over repeated sampling, are expected to contain the true value. If the coverage is too low, it means that the interval is overly narrow or poorly calibrated, leading to an *underestimation* of uncertainty. This can result in overconfidence in statistical conclusions, which may cause critical errors in decision-making or policy formulation, particularly in fields like medicine, finance, or engineering where risks are high.

Note

While the bootstrap does a fine job of estimating the center of a distribution (mean or median), it tends to be biased for other measures (quantiles, variances, etc.). Because the bootstrap is so useful to statisticians, we have spent a lot of time trying to fix these biases. This has led to several "Bias-Corrected" bootstrap estimates.

These are, however, beyond the scope of this book. But, if you are interested, check out the `boot` package in R and the book "Bootstrap Methods and Their Application" by A. C. Davison and D. V. Hinkley (1997).

Extension:

King Rudolph's cabinet consists of several members. Their ages are $\{57, 62, 95, 44, 68, 65, 66, 72\}$. Repeat the above analysis to determine the coverage rate of the non-parametric bootstrap using this new data.

This page titled 21.4.6: Non-Parametric Bootstrap is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

21.4.7: Distribution of Sample Means

21.4.7: Distribution of Sample Means is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

21.4.8: Distribution of Sample Sums

The Central Limit Theorem is a cornerstone of statistical inference because it guarantees that the sampling distribution of the sum will be approximately Normal, regardless of the original population's distribution (more or less), provided the sample size is large enough. This is crucial because it allows us to make probabilistic claims about sample estimates using the Normal distribution — even when we know little about the underlying population.

In practice, this means we can construct confidence intervals, conduct hypothesis tests, and quantify uncertainty without needing to derive exact distributions every time. This set of simulation experiments visually confirms this powerful result: By summing random variables from any finite-variance distribution, you can see the Normal distribution emerge from chaos.

Problem:

The data are not generated from a Normal process, but OLS estimation requires it. How bad is it if I ignore this violation?

Solution:

Let us generate observations (data) under several different distributions, calculate the sums of those samples, and repeat many, many, many times. and both look at the distribution of the sums and test if that distribution is distinguishable from Normal.

Normal (Gaussian) Data:

First, let us see the process with the data being generated from a Normal distribution. The first step is to generate the data.

```
set.seed(370)
y = rnorm(100*1000)
```

Now, to make things faster, let's put this in a matrix and calculate the row sums

```
s = matrix(y, ncol=100)
t = rowSums(s)
```

The variable `t` now contains 1000 values, with each being the sum of n = 100 random values from a Normal distribution.

Unsurprisingly, the Shapiro-Wilk Normality test does not conclude that the sums are not Normally distributed (p-value = 0.710). The histogram shows why:

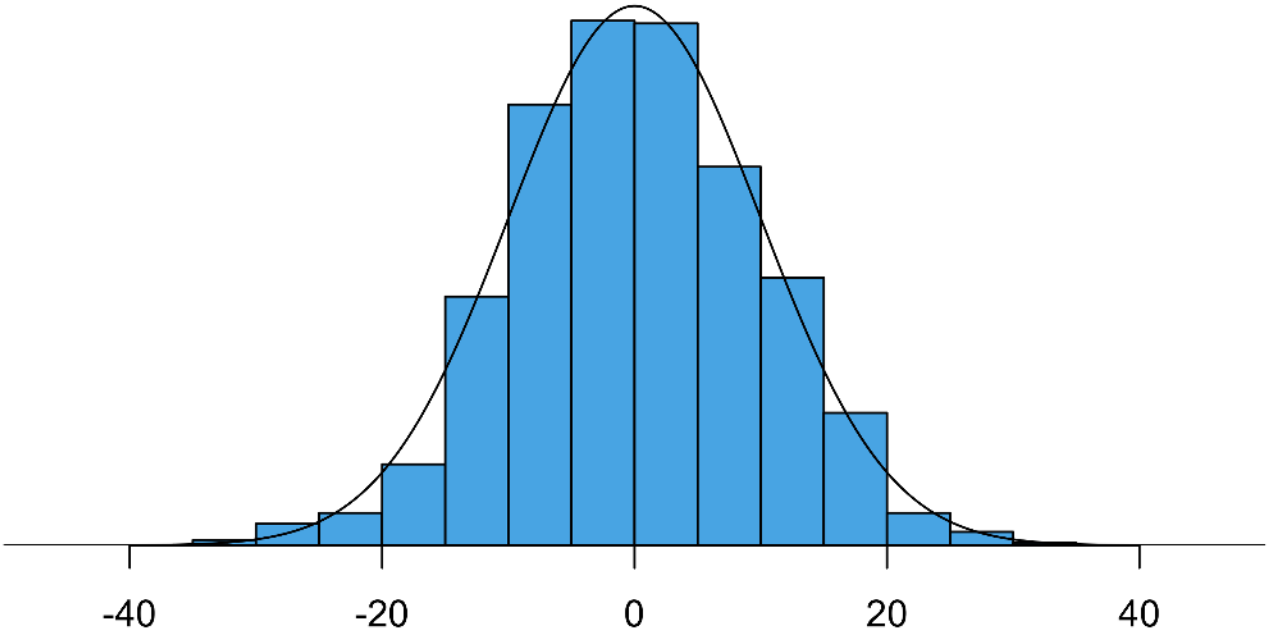


Figure 21.4.8.1: Distribution of the sums of Normally-distributed values, overlain with a Normal curve. Note that the empirical distribution (the histogram) does not exactly match the Normal curve. This is because of random variation. It is close, however.

Student's t Data:

The previous was boring. We completely expected the sums of Normally-distributed values to also be Normally-distributed. Let us do the same analysis, but with data generated from a Student's t distribution. The only thing that changes in the code is the distribution:

```
set.seed(370)
y = rt(100*1000, df=10)
s = matrix(y, ncol=100)
t = rowSums(s)
```

The Shapiro-Wilk test is, again, unable to that the sums are not Normally distributed (p-value = 0.889).

Chi-Square Data:

The t distribution is symmetric. What happens if we use an asymmetric distribution? Here, we use a Chi-Squared distribution.

```
set.seed(370)
y = rchisq(100*1000, df=10)
s = matrix(y, ncol=100)
t = rowSums(s)
```

Again, the Shapiro-Wilk test fails to detect non-Normality (p-value = 0.104).

Cauchy Data:

Well, all of the above data-generating distributions have one thing in common: Their variances are finite. In other words, the Central Limit Theorem applies to them. Larger sample sizes mean that the distribution of sample sums (and means) converge to a Normal distribution. What about a distribution that has a non-finite variance, like the Cauchy?

```
set.seed(370)
y = rcauchy(100*1000)
s = matrix(y, ncol=100)
t = rowSums(s)
```

Note that the Shapiro-Wilk test does detect the non-Normality in the sum of Cauchy random variables. Here is the histogram of the sum of Cauchy random variables.

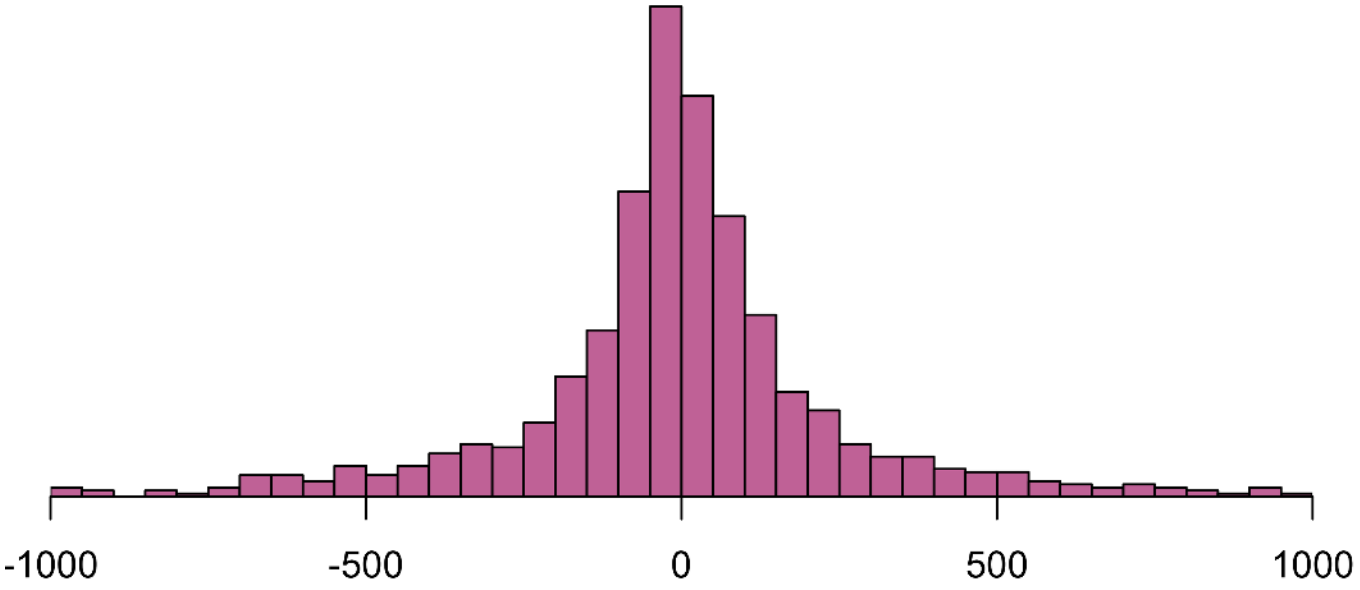


Figure **21.4.8.2**: Distribution of the sums of Cauchy random variables. Compare this to Figure **21.4.8.1**, above. Note the difference in the horizontal scale. Note that it can be shown that the sum of n standard Cauchy random variables follows a Cauchy distribution with scale parameter n. How would you show this?

Concluding Thoughts:

In this series of examples, we have seen that the distribution of the sample sums is less dependent on the distribution of the original data than we may think. As long as the sample size is large enough (and the variance of the distribution is finite), the sample sums are sufficiently close to Normal.

Extensions:

1. Repeat the above experiments, but change the sample size from 100 to 10. Which sample sums remain sufficiently Normal?

2. Repeat the Cauchy experiment above, but change the sample size from 100 to 1000. Are the sample sums sufficiently Normal now? Are the sufficiently Normal if the sample size is 10,000? ... what about 100,000? Explain what your conclusions mean.

21.4.8: Distribution of Sample Sums is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

CHAPTER OVERVIEW

22: The Appendix of Statistics

This appendix covers most of the things you need to remember from your introductory statistics course — and more. Treat this as more than just a review. It includes all you need from your statistics course, *plus some additional* items that you may find helpful. Starred sections, as usual, are optional for understanding linear models at a basic level. Note that I *do* include a section on moment generating functions (MGFs) and a proof on the Central Limit Theorem (CLT). Neither tend to be included in introductory statistics courses. Also, neither proof is overly important for this course. Both are offered up on the altar of "this is kinda interesting." The consequences of the CLT, however, are immense, as such, I encourage you to read through those sections.

The Random Variable

At the very center of statistics... and of *understanding* statistics... is the **random variable**. There are many ways of defining a random variable. This is important, because the random variable is central to our understanding of probability and statistics. The "official" definition of a random variable is

Definition: Random Variable

A random variable $X : \Omega \to E$ is a measurable function from a set of possible outcomes Ω to a measurable space E.

While this definition has the advantage of being mathematically precise, it is not necessarily helpful in terms of understanding what a random variable *really is*. For me, a random variable is just an unknown outcome of an experiment. The "variable" you were introduced to in high school algebra is a value that is unknown until more information is available. A *random variable* is a value that is unknown until that experiment is performed. Repeated experiments may produce different values.

Here, I am using the term "experiment" broadly. It is any action, including simple observation. Thus, my height right now is not a random variable. However, my height in three years *is* a random variable. I will not know its value until I perform the experiment (measure it in three years).

> Random variables have distributions. Nothing else has a distribution in the sense we are using it here — *nothing* else. Since random variables have distributions, they also have expected values, variances, minimums, maximums, medians, and many other measures on the variable.

Samples are drawn from random variables to better understand how they behave (a.k.a. their distribution, Section 22.5). Thus, if I want to understand the relationship between engine displacement (a fixed/set variable) and mileage (a random variable) for automobiles in general, I would measure engine displacements and mileages from a sample of automobiles. That sample would give me information about the relationship between those two variables.

Note

There is a difference between the variable (what we measure) and the values (*results* from our measurements). The values constitute our sample. This means that the engine displacement is a variable and some values are $\{1.76, 2.25, 1.8, 2.5, 0.75\}$ liters. Similarly, mileage is a variable and $\{25.3, 17.8, 34.4, 14.5, 30.5\}$ are some values.

To help with **notation**, I *try* to follow these rules:

- Observations (data, values) are denoted with lowercase Latin letters like x, y, and z;
- random variables (unobserved), with uppercase Latin letters like X, Y, and Z;
- population parameters in need of estimating, with Greek lowercase letters like μ, σ, and π; and

- the set of possible parameter values, with Greek uppercase letters like M, Σ, and Π.

22.1: Importantly Confusing Points
22.2: Sample Statistics
22.3: Population Parameters
22.4: Probability Distributions
22.5: Distributions of Sample Statistics
22.6: Other Topics
22.7: End-of-Appendix Materials

This page titled 22: The Appendix of Statistics is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

22.1: Importantly Confusing Points

So, here is something that is extremely confusing because we use the same word to mean/imply *different* things. This section can be summarized as "Make sure you know what the terms mean in the different contexts." However, since I like to write, here is a whole section of the book trying to make this idea manifest. While I focus on the variance, the same issue arises with things like "mean" and "skew."

Note that in statistics (in all disciplines), few things are as fundamental as terminology. We use the same word to describe the spread of a fixed set of numbers, the variability within a specific data sample, and a fixed property of a probability distribution. All three of these are termed the "variance." However, while each is a valid measure of dispersion, they represent fundamentally different ideas: a calculated value, a random statistic, and a deterministic parameter.

This section unpacks this critical terminology.

✦•·····• ✦ •·····•✦

A set of numbers has a variance. A sample from a random variable has a variance. A distribution has a variance. All three measures are termed "variance;" however, they mean different things.

> One can also say that a "random variable has a variance." While a random variable is *not* a distribution (it *follows* one, however), we will often conflate the random variable and its distribution.

Here is a set of numbers I specify (i.e., they are *not* randomly generated): $\{1, 2, 3, 4, 5, 6, 7\}$. They have a mean and a variance. The mean is 4. The variance is 4.667. Those measures only tell me about the numbers *I* specified. Since they are not realizations of a random variable, it does not make sense to use those values for anything other than **mathematical calculations** on those particular values.Here is a set of numbers generated from a Poisson distribution (see the Poisson Section) with mean $\lambda = 1$: $\{1, 2, 1, 2, 0\}$. These values (this *sample*) also have a mean ($\mu = 1.20$) and a variance ($\sigma^2 = 0.70$). Since these numbers are a **realization of** a random variable, these two values can be called "statistics." They are *functions* of observed random variables. Because they are statistics, the mean should be close to the expected value of the random variable, and the variance should be close to the variance of the random variable.

Caution

Since these statistics are functions of a random sample, those statistics are also random variables. As such, they have distributions. In fact, understanding and knowing the distribution of sample statistics is one of the goals of statistics. This study led you to confidence intervals and test statistics back in your previous statistics course.

Here is a Chi-square distribution with parameter "4 degrees of freedom": $\chi^2_{\nu=4}$. It has an expected value ($\mu = 4$) and a variance ($\sigma^2 = 8$). These are parameters and are *not* a function of a sample (hence they are represented by lowercase Greek letters). These are genuine numbers that have no distribution.

> They have no distribution because they do not vary. Our *understanding* of them may vary, but the value does not.

Population parameters are also numbers that we do not know when we are applied statisticians. These are, however, numbers that we *want* to know — or at least obtain a good estimate of. To get those estimates, we obtain a sample from the distribution and measure the corresponding sample statistics. While the sample statistics will not (usually) exactly equal the population parameters, they will tend to be close. How close depends on the sample size and the statistic measured.

Caution

This "issue" with the variance is *not* unique to the variance.

The lesson to take away from this is: You need to be aware of the symbols and what they mean. This is especially true when it comes to values, random variables, and distributions.

This page titled 22.1: Importantly Confusing Points is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

22.2: Sample Statistics

A statistic is simply a function of your data. Since data are typically a random sample, any statistic you calculate — like a mean or a variance or a skew — is *itself* a random variable. This means your sample mean has its own distribution and will vary from one sample to the next. This section reviews key sample statistics essential for linear models, framing them not just as static summaries but as the fundamental, random building blocks of statistical inference. Understanding this duality — between a calculated value and a random quantity — is crucial for moving from simple description to meaningful modeling.

✦•·····• ✦ •·····•✦

Sample Mean (Arithmetic Mean)

The sample mean is the usual "average" that we have calculated many times in the past. If we want to use just one number to summarize the data, the mean is the usual value.

Definition: Sample Mean

Let Y_i be a random sample from a distribution. The sample mean is defined as

$$\overline{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i \tag{22.2.1}$$

Note that this definition is equivalent to

$$\sum_{i=1}^{n} Y_i = n\overline{Y} \tag{22.2.2}$$

This form is useful in many proofs, and you will see one later. Before we get to that, however, here is an elementary property of the sample mean that needs to be provided.

Lemma 22.2.1: Linear Functional

The sample mean is a linear functional.

Proof.

This means we have to show that the expected value of $(aY + b)$ is $a\overline{Y} + b$ for non-stochastic scalars a and b. Thankfully, this proof is very straightforward. We just find the mean of $aY + b$ using the definition.

$$\overline{aY+b} = \frac{1}{n}\sum_{i=1}^{n}(aY_i + b) \tag{22.2.3}$$

$$= \frac{1}{n}\sum_{i=1}^{n} aY_i + \frac{1}{n}\sum_{i=1}^{n} b \tag{22.2.4}$$

$$= a\frac{1}{n}\sum_{i=1}^{n} Y_i + \frac{1}{n}nb \tag{22.2.5}$$

$$= a\overline{Y} + b \tag{22.2.6}$$

Thus, through the transitive property, the expected value of a linear combination of the variable is just a linear combination of the expected values.

■

You should go through each of the steps in the previous proof and give a reason the step is mathematically correct. This will give you some practice in the mathematics underlying linear models

Now, we show that the average deviance from the mean is zero.

Lemma 22.2.2: Mean Deviation

Let Y_i be a sample of size n from a random variable.

$$\sum_{i=1}^{n} \left(Y_i - \overline{Y}\right) = 0 \tag{22.2.7}$$

Proof.
I leave the proof as an exercise.

While it seems as though this lemma is not important, you will see its application **over and over again**. It will serve you well to learn it by sight *as quickly as possible*.

Sample Variance

If we use the mean to summarize the entire data set using a single number, the measure of spread is used to see how well that number represents each element of the data. There are many measures of spread that you probably have come across. These include the variance, standard deviation, and interquartile range. Here we will focus on the variance.

Definition: Sample Variance

Let Y_i be a random sample of size n from a distribution. The sample variance is defined as

$$S_y^2 = \frac{1}{n-1} \sum_{i=1}^{n} \left(Y_i - \overline{Y}\right)^2 \tag{22.2.8}$$

Here are some elementary properties of the sample variance that need to be provided.

Lemma 22.2.3: Quadratic Functional

The sample variance is a quadratic functional.

Proof.
This means we need to show that the variance of $(aY + b)$ is $a^2 S_y^2$ for non-stochastic scalars a and b. This is *also* a relatively straight forward application of the definition.

Thus, let Y_i be a random sample of size n from a distribution. With this, we have the following

$$S^2_{aY+b} = \frac{1}{n-1}\sum_{i=1}^{n}\left((aY_i+b)-(\overline{aY+b})\right)^2 \tag{22.2.9}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}\left(aY_i+b-a\overline{Y}-b\right)^2 \tag{22.2.10}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}\left(aY_i-a\overline{Y}\right)^2 \tag{22.2.11}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}\left(a(Y_i-\overline{Y})\right)^2 \tag{22.2.12}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}a^2\left(Y_i-\overline{Y}\right)^2 \tag{22.2.13}$$

$$= a^2\,\frac{1}{n-1}\sum_{i=1}^{n}\left(Y_i-\overline{Y}\right)^2 \tag{22.2.14}$$

$$= a^2\,S_y^2 \tag{22.2.15}$$

Thus, since $S^2_{aY+b} = a^2\,S_y^2$, we know that the variance is a quadratic functional.

■

The proofs of Lemmas **22.2.1** and **22.2.3** both exemplify one important method for proving statements: Substitute and simplify. It also provides to us an important result. Go over both proofs a few times. Realize the importance of the scalar multiple on the variance, as well as the scalar addend on the variance.

Also, think about **why it makes sense** for a translation to have no effect on the variance, but a scale change to have a significant effect.

What is so important about the previous lemma? Think about the variance of heights of students in this room. First, measure in feet, then in inches, then in centimeters. Why does the variance change in the three measurements?

Now, have everyone stand on the same chair while measuring them in inches. Why does the variability when measured to the floor does not depend on whether they are standing on the chair or not?

Sample Covariance

The variance measures the variability of a single value. The *covariance* measures the variability of the *relationship* between two numeric variables.

Definition: Sample Covariance

Let X_i be a random sample from a distribution. Let Y_i be a random sample from a distribution (same as X or different). The sample covariance is defined as

$$S_{X,Y} = \frac{1}{n-1}\sum_{i=1}^{n}\left(X_i-\overline{X}\right)\left(Y_i-\overline{Y}\right) \tag{22.2.16}$$

The covariance is also denoted by $Cov[X,Y]$. However, using this symbol may lead to confusion about whether you are talking about a sample or a population. It is safer to stay with $S_{X,Y}$ as the symbol for the sample covariance and with $\sigma_{X,Y}$ as the symbol for the population covariance.

Here are some elementary properties of the sample covariance.

Lemma 22.2.4

$S_{X+Y,Z} = S_{X,Z} + S_{Y,Z}$

Proof.

This proof is also a direct application of the definition and of the proof style of Lemma 22.2.3.

$$S_{X+Y,Z} = \frac{1}{n-1}\sum_{i=1}^{n}\left((X_i+Y_i)-(\overline{X}-\overline{Y})\right)\left(Z_i-\overline{Z}\right) \tag{22.2.17}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}\left(X_i-\overline{X}+Y_i-\overline{Y}\right)\left(Z_i-\overline{Z}\right) \tag{22.2.18}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}\left(X_i-\overline{X}\right)\left(Z_i-\overline{Z}\right) + \frac{1}{n-1}\sum_{i=1}^{n}\left(Y_i-\overline{Y}\right)\left(Z_i-\overline{Z}\right) \tag{22.2.19}$$

$$= S_{X,Z} + S_{Y,Z} \tag{22.2.20}$$

Thus, by the transitive property, $S_{X+Y,Z} = S_{X,Z} + S_{Y,Z}$.

■

Note

Remember that proving things is important. As you work through this material, you need to be able to prove things mathematically. In statistics, however, not all things can be proven mathematically. Some things need simulation to give provisional results. However, proving things mathematically is preferred.

I leave the following three lemmas as exercises for you.

Lemma 22.2.5

$S_{X,Y} = S_{Y,X}$

Lemma 22.2.6

$S_{aX,bY} = ab\ S_{X,Y}$

Lemma 22.2.7

$S_{X,X} = S_X^2$

Sample Correlation

Like the covariance, the correlation measures **the strength of the relationship** between two numeric variables. However, where the covariance cannot be meaningfully compared across variables, the correlation can. A correlation value of 0.25 indicates the same exact thing whether you are looking at the relationship between height and weight or between population and GDP. A covariance of 100 means something different if one is looking at the relationship between height and weight or if one is looking at the relationship between population and GDP. This is because the correlation scales the covariance to ensure that it ranges between -1 and +1.

Definition: Sample Correlation

Let X_i be a random sample of size n from a distribution with finite variance. Let Y_i be a random sample from a distribution, also with finite variance. The sample correlation is defined as

$$R_{X,Y} = \frac{\sum_{i=1}^{n} \left(X_i - \overline{X}\right)\left(Y_i - \overline{Y}\right)}{\sqrt{\sum_{i=1}^{n} \left(X_i - \overline{X}\right)^2 \sum_{i=1}^{n} \left(Y_i - \overline{Y}\right)^2}} \tag{22.2.21}$$

The correlation is also symbolized using $Cor[X,Y]$. However, this may be confusing as this symbol is also used for the population correlation. It is better to use $R_{X,Y}$ and $r_{X,Y}$ for the sample correlation and $\rho_{X,Y}$ for the population correlation.

Alternate formulas for the correlation include:

$$R_{X,Y} = \frac{S_{X,Y}}{\sqrt{S_X^2 S_Y^2}} \tag{22.2.22}$$

$$R_{X,Y} = \frac{S_{X,Y}}{S_X S_Y} \tag{22.2.23}$$

I leave the following lemmas as exercises for you.

Lemma 22.2.8

$R_{X,Y} = R_{Y,X}$

Lemma 22.2.9

$R_{aX,Y} = R_{X,Y}$

Lemma 22.2.10

$R_{aX+b,Y} = R_{X,Y}$

Lemma 22.2.11

$-1 \leq R_{X,Y} \leq 1$

Lemma **22.2.10** is *very important*. Do not just think about them in terms of the mathematics. Think about what these mean in terms of the reality: If we apply a linear transformation to the two variables, how does that affect the correlation? What does that *really* mean?

Caution

In each of the above definitions, the numeric divisor is chosen to ensure that the sample statistic is an unbiased estimator of the population parameter. In other words, we divide by $n-1$ so that **this equation is true**:

$$E[\text{statistic}] = \text{parameter}$$

It is interesting that those denominators are the "degrees of freedom" for that statistic. This gives one *helpful* "definition" for the term **degrees of freedom**.

This page titled 22.2: Sample Statistics is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

22.3: Population Parameters

Let us spend a section examining population parameters. A population parameter is a measurement on the *population*, analogous to the sample statistic being a measurement on the sample. Note that it is the population parameters we seek to know. The sample statistics are only used to give us information about them. Some usual population parameters of interest are the mean, variance, covariance, and correlation.

✦•·····• ✦ •·····•✦

Population Mean

The population mean is a measure of center (or "typicalness") for the *entire distribution* (or random variable). It is known by a few different names: population mean, expected value, and first moment. It is represented by some variation of either μ or $E[Y]$.

If the distribution is discrete, the formula for the expected value is defined as

$$E[Y] \stackrel{\text{def}}{=} \sum_{i \in \mathcal{S}} y_i \; P[Y = y_i] \tag{22.3.1}$$

If the distribution is continuous, the formula for the expected value is

$$E[Y] \stackrel{\text{def}}{=} \int_{\mathcal{S}} y \; f(y) \; \mathrm{d}y \tag{22.3.2}$$

Note the similarities between the two formulas. The difference is due solely to the mathematics underlying Equations 22.3.1 and 22.3.2. Those mathematics tend to be covered in an undergraduate **real analysis course**.

> Another interpretation of the expected value is the "long-run average" of the outcomes. I frequently find this useful in checking that my results match my expectations.

✔ Example 22.3.1

Let Y be the number of heads in one flip of a fair coin. What is the expected number of heads?

Solution.
The random variable Y is discrete with sample space $\mathcal{S} = \{0, 1\}$. Thus, the formula for the expected value gives us

$$E[Y] \stackrel{\text{def}}{=} \sum_{i \in \mathcal{S}} y_i \; P[Y = y_i] \tag{22.3.3}$$

$$= 0 \; P[Y = 0] + 1 \; P[Y = 1] \tag{22.3.4}$$

$$= 0(0.500) + 1(0.500) \tag{22.3.5}$$

Thus, the expected number of heads is $E[Y] = 0.500$.

> The expected value of a random variable does *not* need to be an element of the sample space (as here). Thus, the interpretation as a **long-run average** helps in understanding and interpreting the mean.

✔ Example 22.3.2

Let Y be the time I spend at a stoplight. If the light has a 120-second cycle, spending 55s on green, 5s on yellow, and 60s on red, calculate the expected time I wait, ***given*** that I have to stop.

Solution.
The random variable Y is continuous with sample space $\mathcal{S} = [0, 60]$. Without any information beyond knowing that there is a lower

and an upper bound, we should assume the random variable follows a Uniform distribution.

Thus, the formula for the expected value gives us

$$E[Y] \stackrel{\text{def}}{=} \int_{\mathcal{S}} y\ f(y) \mathrm{d}y \tag{22.3.6}$$

$$= \int_0^{60} y\ \frac{1}{60} \mathrm{d}y \tag{22.3.7}$$

$$= \frac{1}{60} \int_0^{60} y \mathrm{d}y \tag{22.3.8}$$

$$= \frac{1}{60} \frac{y^2}{2} \Big|_{y=0}^{60} \tag{22.3.9}$$

$$= \frac{1}{60} \left(\frac{60^2}{2} - \frac{0^2}{2} \right) \tag{22.3.10}$$

$$= 30 \tag{22.3.11}$$

Thus, the expected time I spend at the stop light is $E[Y] = 30$ seconds. Why does that not surprise me?

■

> For those interested in understanding *why* the Uniform is the distribution, please read up on "maximum entropy distributions."

Example 22.3.3

Let Y be the time I spend at a stoplight. If the light has a 120-second cycle, spending 55s on green, 5s on yellow, and 60s on red, calculate the expected time I wait.

Solution.
This is quite *different* from the previous example. There, we knew the light was red. Now, we do not know if it is green (no time stopped), yellow (no time stopped), or red.

One can follow the above procedure to solve the problem. However, I would like to use a **more helpful procedure**. To motivate the procedure, note that the mean has been calculated as the sum of each value times its probability. Thus, we could calculate the expected time to wait as

$$E[\text{wait}] = P[\text{green}]\ E[\text{green}] + P[\text{yellow}]\ E[\text{yellow}] + P[\text{red}]\ E[\text{red}] \tag{22.3.12}$$

$$= \frac{55}{120}\ 0 + \frac{5}{120}\ 0 + \frac{60}{120}\ 30 \tag{22.3.13}$$

$$= \frac{1800}{120} \tag{22.3.14}$$

Thus, *without* the additional information that I actually stopped at the stoplight, the expected wait time is only 15s.

■

———— ⋆·☆·⋆ ————

With formulas for the expected value (equations 22.3.1 and 22.3.2), one can find the expected value of any *function* of a random variable, too. For instance, $E[Y^2]$ is the expected value of the square of the random variable. It is calculated as

$$E[Y^2] = \sum_{i \in \mathcal{S}} y_i^2\ P[Y = y_i] \tag{22.3.15}$$

if Y is discrete, and

$$E[Y^2] = \int_{\mathcal{S}} y_i^2\ f(y)\ \mathrm{d}y \tag{22.3.16}$$

if y is continuous.

This will come in handy in terms of notation.

Example 22.3.4

Let $Y \sim Bern(\pi = 0.500)$. Calculate $E[Y]$, $E[Y^2]$, and $E[Y^3]$.

Solution.

Recall that the probability mass function for Y is given by

$$f(y) = \begin{cases} 0.500 & y = 0 \\ 0.500 & y = 1 \end{cases} \tag{22.3.17}$$

Thus, we have

$$E[Y] \stackrel{\text{def}}{=} \sum_{y \in S} p(y)\ y \tag{22.3.18}$$

$$= (0.500)\ 0 + (0.500)\ 1 \tag{22.3.19}$$

$$= 0.500 \tag{22.3.20}$$

and

$$E[Y^2] = \sum_{y \in S} p(y)\ y^2 \tag{22.3.21}$$

$$= (0.500)\ 0^2 + (0.500)\ 1^2 \tag{22.3.22}$$

$$= 0.500 \tag{22.3.23}$$

and

$$E[Y^3] = \sum_{y \in S} p(y)\ y^3 \tag{22.3.24}$$

$$= (0.500)\ 0^3 + (0.500)\ 1^3 \tag{22.3.25}$$

$$= 0.500 \tag{22.3.26}$$

Clearly, that these three moments are equal is a feature of the Bernoulli distribution. Such will not necessarily be true for any other distribution.

Lemma 22.3.1: Bernoulli Moments

If $Y \sim Bern(\pi)$, then $E[Y^n] = \pi$ for any positive, finite-valued n.

Proof.

Recall that the probability mass function for Y is given by

$$f(y) = \begin{cases} 1 - \pi & y = 0 \\ \pi & y = 1 \end{cases} \tag{22.3.27}$$

Thus, we have

$$E[Y^n] = \sum_{y \in S} p(y)\ y^n \tag{22.3.28}$$

$$= (1 - \pi)\ 0^n + (\pi)\ 1^n \tag{22.3.29}$$

$$= (1 - \pi)\ 0 + (\pi)\ 1 \tag{22.3.30}$$

$$= \pi \tag{22.3.31}$$

Thus, it is proven.

■

When discussing the random variable, one will tend to refer to $E[Y]$. When discussing the *distribution* of the random variable, one will tend to refer to μ. Since random variables "follow" (or "have") a distribution, the two terms tend to be treated as being interchangeable, especially since $E[Y] = \mu$.

Formula 22.3.1 for the expected value is the one we all see in our introductory statistics course. There is *another formula* that may be useful to know. Given $F(y)$ is the cumulative distribution function (CDF) for Y, then this is true:

$$E[Y] = \int_0^\infty \Big(1 - F(y)\Big)\, dy \tag{22.3.32}$$

Note that this formula *only works* if the support of the random variable is non-negative.

Example 22.3.5

Let Y be the time I spend at a stoplight. If the light has a 120-second cycle, spending 55s on green, 5s on yellow, and 60s on red, calculate the expected time I wait, given that the light is red. Use Formula 22.3.32.

Solution.
This is the same problem as above, so we should get the same answer. This time, however, we need to use equation 22.3.32. We know that the CDF for the above Uniform distribution is

$$F(y) = \begin{cases} 0 & y \le 0 \\ y/60 & 0 < y \le 60 \\ 1 & y > 60 \end{cases} \tag{22.3.33}$$

Thus, $E[Y]$ is

$$E[Y] = \int_0^\infty \Big(1 - F(y)\Big)\, dy \tag{22.3.34}$$

$$= \int_0^\infty \begin{pmatrix} 1 - y/60 & 0 < y \le 60 \\ 0 & y > 60 \end{pmatrix} dy \tag{22.3.35}$$

$$= \int_0^{60} 1 - y/60 \, dy \tag{22.3.36}$$

...

I leave the rest as an exercise for you.

■

Population Variance

The population variance is a measure of uncertainty (or of spread or uncertainty) for the distribution. It is symbolized as σ^2 and as $V[Y]$. When discussing the random variable, one will tend to refer to $V[Y]$; however, when discussing the *distribution* of the random variable, one will tend to refer to σ^2, as above.

The typical population variance formulas are

$$V[Y] = \sum_{i \in S} (y_i - \mu)^2 \; P[Y = y_i] \tag{22.3.37}$$

$$= \int_S (y_i - \mu)^2 \; f(y) dy \tag{22.3.38}$$

Note that these are merely

$$V[Y] = E[(Y - \mu)^2] \tag{22.3.39}$$

In other words, the variance is defined as **the second central moment**.

This last definition may be more helpful in our understanding of the variance, especially as it leads to a different formula for the variance:

$$E[Y^2] = \sigma^2 + \mu^2 \tag{22.3.40}$$

Theorem 22.3.2: Another Variance Formula

Let Y be a random variable with finite mean and variance.

$$E[Y^2] = V[Y] + E[Y]^2 \tag{22.3.41}$$

Proof.
I leave the proof as an exercise.

Another Variance Formula

The variance formulas given above in Equations 22.3.37 and 22.3.38 are the typical ones provided in undergraduate statistics. They depend on the probability mass (or density) function (pmf or pdf). There is a formula for the population variance that relies on the cumulative distribution function (CDF) instead (much like formula 22.3.32). Here it is.

$$\sigma^2 \;=\; 2\int_0^\infty y\Big(1-F(y)\Big)\,\mathrm{d}y \;-\; \left(\int_0^\infty \Big(1-F(y)\Big)\,\mathrm{d}y\right)^2 \tag{22.3.42}$$

Note that this formula *only works* for a non-negative random variable... that is, for random variables whose possible values must be positive. Also note that this formula *should only be used* if the CDF is easier to work with than the pdf. This is a rare event, which illustrates why few see this formula.

Example 22.3.6

Show that the variance of a standard Uniform distribution is

$$V[Y] = \frac{1}{12} \tag{22.3.43}$$

Use both the pdf and the CDF method.

Solution.
I leave this as an exercise for you.

Population Covariance and Correlation

The formula for the population **covariance** is

$$\sigma_{xy} = E[(X-\mu_x)(Y-\mu_y)] \tag{22.3.44}$$

Note this reduces to $V[Y] = Cov[Y,Y]$ and $\sigma_y^2 = \sigma_{yy}$.

The formula for the population **correlation** is

$$\rho_{xy} = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \tag{22.3.45}$$

The Greek letter ρ is "rho." It is *not* a "p."

Example 22.3.7

What is the covariance between a random variable and a constant?

Solution.

I leave this as an exercise for you.

Example 22.3.8

What is the covariance between two constants?

Solution.

I leave this as an exercise for you.

This page titled 22.3: Population Parameters is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

22.4: Probability Distributions

In the study of statistics, we do not navigate uncertainty without a map. Our guides are probability distributions — mathematical functions that describe the likelihood of different outcomes for a random process. You have likely encountered several of these foundational distributions already, such as the **Binomial** for counts of success, the **Poisson** for events over time or space, and the ubiquitous **Normal** distribution that models so many natural phenomena. Others, like the **Chi-square**, **Student's t**, and **F distributions**, emerge powerfully as essential tools for statistical inference, particularly when forming confidence intervals and testing hypotheses.

Each distribution tells a story: some arise naturally from the world around us, while others are derived as crucial consequences of our need to analyze data and draw conclusions. The coming sections will explore these distributions — and others — in detail. We will examine their properties, their origins, and, most importantly, their practical applications, building the necessary toolkit to model variability and make informed statistical decisions.

✦•·····• ✦ •·····•✦

The Uniform Distribution

The most important distribution is the Uniform distribution. All other distributions can be generated from it, either directly or in a limiting process. This distribution is so important that statistics textbooks from the 20th century had random values from a Uniform as the Table 1 (usually inside the front cover).

The Uniform distribution is a continuous probability distribution with a fixed upper and lower bound and with all elements of its sample space having the same likelihood.

$$f(y;a,b) = \begin{cases} \frac{1}{b-a} & y \in (a,b) \\ 0 & \text{otherwise} \end{cases} \tag{22.4.1}$$

This probability distribution has two parameters, a and b, where a is the minimum possible value returned, and b is the maximum possible value returned.

From this description, we have the following results if $Y \sim Unif(a,b)$:

- The sample space is $\mathcal{S} = [a,b]$.
- The expected value is $E[Y] = \frac{b+a}{2}$.
- The variance is $\sigma^2 = \frac{b-a}{12}$.
- The skew is zero.

Finally, the cumulative distribution function, CDF, is

$$F(y;a,b) = \begin{cases} 0 & y \le a \\ \frac{y}{b-a} & x \in (a,b) \\ 1 & y \ge b \end{cases} \tag{22.4.2}$$

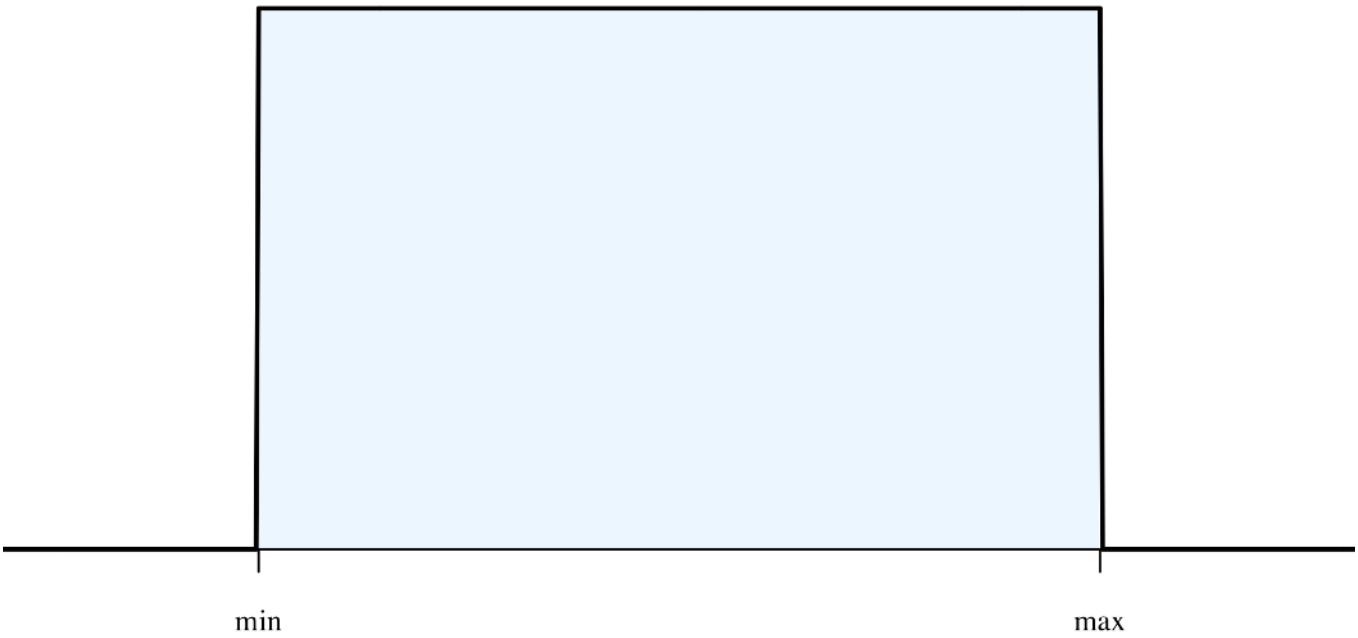


Figure 22.4.1: A typical Uniform distribution. Note that the likelihood (vertical axis) is zero outside the shown sample space of $\mathcal{S} = [\min, \max] = [a, b]$. Also note that the likelihood within its sample space is a constant.

The Bernoulli Distribution



Jacob Bernoulli, c.1687
(Source: Wikipedia)

Arguably, the Bernoulli is the grandfather of all discrete distributions. It models a random variable that has two possible outcomes, which we call "failure" and "success," or 0 and 1. This is its "**sample space**," the set of all outcomes with non-zero likelihood:

$$\mathcal{S} = \{0, 1\} \tag{22.4.3}$$

The Bernoulli has a single **parameter**, π, which is the probability of the random variable being 1. We will frequently refer to π as the "success probability." The success probability is a real number between 0 and 1, inclusive. Thus, we can write $\pi \in \Pi = [0, 1]$.

The **probability mass function** (pmf) for the Bernoulli is

$$f(y) = \begin{cases} 1 - \pi & y = 0 \\ \pi & y = 1 \\ 0 & \text{otherwise} \end{cases} \tag{22.4.4}$$

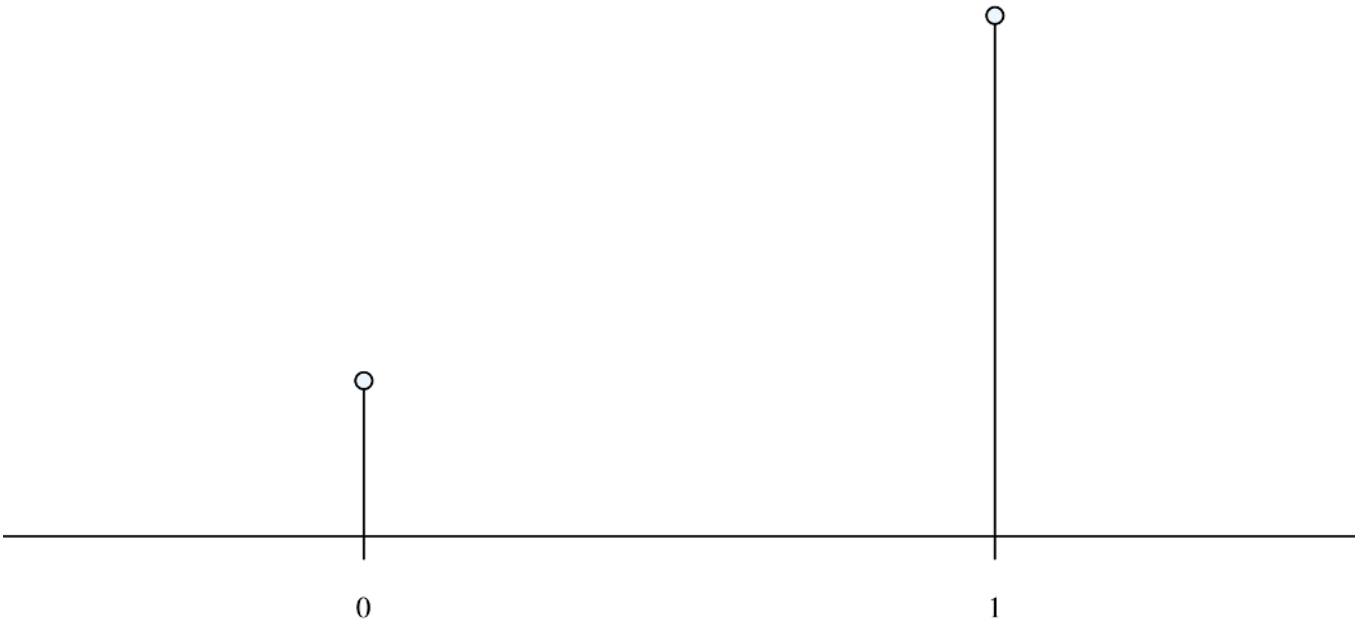


Figure 22.4.2: A Bernoulli distribution. Here, the probability of success is $\pi = 0.77$, hence the probability of a success (the spike at 1) is greater than the probability of a failure (spike at 0).

Technically, the probability mass function must return a value for every real y, hence the need for the third line in the definition of the Bernoulli pmf. With that being said, it is frequently left off and assumed. Thus, we will often see it as

$$f(y) = \begin{cases} 1-\pi & y = 0 \\ \pi & y = 1 \end{cases} \tag{22.4.5}$$

without any loss of understanding.

Its **cumulative distribution function** (CDF) of the Bernoulli distribution is

$$F(y) = \begin{cases} 0 & y < 0 \\ 1-\pi & 0 \le y < 1 \\ 1 & 1 \le y \end{cases} \tag{22.4.6}$$

Similarly, this will frequently be written as

$$F(y) = 1-\pi \qquad \text{for } 0 \le y < 1 \tag{22.4.7}$$

without confusion.

The mean (**expected value**) of a Bernoulli random variable is

$$E[Y] \stackrel{\text{def}}{=} \sum_{i \in \mathcal{S}} y_i \, f(y_i) \tag{22.4.8}$$

$$= 0 \times (1-\pi) + 1 \times (\pi) \tag{22.4.9}$$

$$= \pi \tag{22.4.10}$$

Its **variance** is

$$V[Y] \stackrel{\text{def}}{=} \sum_{i \in \mathcal{S}} (y_i - \mu)^2 \, f(y_i) \tag{22.4.11}$$

$$= (0-\pi)^2 \times (1-\pi) + (1-\pi)^2 \times (\pi) \tag{22.4.12}$$

$$= \pi^2(1-\pi) + \pi(1-\pi)^2 \tag{22.4.13}$$

$$= \pi^2 - \pi^3 + \pi - 2\pi^2 + \pi^3 \tag{22.4.14}$$

$$= \pi(1-\pi) \tag{22.4.15}$$

Skew (or **skewness**) measures the asymmetry of a probability distribution around its mean. Positive skew (right-skew) indicates a long tail on the right, meaning most data is clustered on the left with some high outliers, while negative skew (left-skew) has a long tail on the left with low outliers. Knowing skew is important for several reasons, some of which are:

- Interpretation of Averages
  In a skewed distribution, the mean is pulled toward the tail. Knowing the skew warns you that the mean may be misleading (e.g., average income is often skewed high by outliers, so the median is more representative).
- Informing Data Transformations & Model Choice
  Many statistical models assume symmetry. Identifying skew helps you decide if you need to transform the data (e.g., using a log transformation for positive skew) or choose a non-parametric test.
- Risk and Decision-Making
  In finance, positive skew of returns might be desirable (small chance of huge gains), while negative skew signals a risk of large losses.

The skew of a Bernoulli is

$$\gamma_3(Y) \stackrel{\text{def}}{=} E\left[\left(\frac{Y-\mu}{\sigma}\right)^3\right] \tag{22.4.16}$$

$$= \sum_{i \in \{0,1\}} \left(\frac{y_i - \mu}{\sigma}\right)^3 f(y_i) \tag{22.4.17}$$

$$= \left(\frac{0-\pi}{\pi(1-\pi)}\right)^3 \times (1-\pi) + \left(\frac{1-\pi}{\pi(1-\pi)}\right)^3 \times (\pi) \tag{22.4.18}$$

... some algebra ...

$$= \frac{1-2\pi}{\sqrt{\pi(1-\pi)}} \tag{22.4.19}$$

Here, we are defining skew as the "third standardized moment." Other definitions are available, including the Hildebrand ratio, which you may have learned in your introductory statistics course,

$$H \stackrel{\text{def}}{=} \frac{\overline{Y} - \widetilde{Y}}{S} \tag{22.4.20}$$

Example 22.4.1

What is the Hildebrand ratio for a Bernoulli distribution?

Solution.

Because the median of the Bernoulli is not a smooth function, let us break this proof into three cases. Note, we will assume $\pi \in (0,1)$. If $\pi = 0$ or $\pi = 1$, the statistics are not interesting.

Case 1: $\pi > 0.500$
Let $Y \sim Bern(\pi)$. Let $\pi > 0.500$. From the probability mass function and the definition of the Hildebrand ratio, we have the following:

$$H \stackrel{\text{def}}{=} \frac{\overline{Y} - \widetilde{Y}}{S} \tag{22.4.21}$$

$$= \frac{\pi - 1}{\pi(1-\pi)} \tag{22.4.22}$$

$$= -\frac{1}{\pi} \tag{22.4.23}$$

Since this is always negative, we know that the Bernoulli is negatively skewed if $\pi > 0.500$.

Case 2: $\pi > 0.500$
As previously, we have the following:

$$H \stackrel{\text{def}}{=} \frac{\overline{Y} - \widetilde{Y}}{S} \tag{22.4.24}$$

$$= \frac{\pi - 0}{\pi(1-\pi)} \tag{22.4.25}$$

$$= \frac{1}{1-\pi} \tag{22.4.26}$$

Since this is always positive, we know that the Bernoulli is positively skewed if $\pi < 0.500$.

Case 3: $\pi = 0.500$
As previously, we have the following:

$$H \stackrel{\text{def}}{=} \frac{\overline{Y} - \widetilde{Y}}{S} \tag{22.4.27}$$

$$= \frac{\pi - \pi}{\pi(1-\pi)} \tag{22.4.28}$$

$$= 0 \tag{22.4.29}$$

Thus, the Bernoulli is symmetric when $\pi = 0.500$.

■

Note

In the previous example, I required $\pi \notin \{0, 1\}$. This is not a restriction for applied statistics. Why? What does it mean if $\pi = 0$ or $\pi = 1$? When would such a thing happen? Why would we need to study it?

Kurtosis measures the "tailedness" of a probability distribution, indicating how much data is in the tails compared to the center. A high kurtosis means heavy tails and more outliers, while low kurtosis means lighter tails and fewer outliers. Knowing kurtosis is valuable for a few reasons:

- Risk and Outlier Analysis
  In fields like finance and quality control, high kurtosis signals a higher probability of extreme events (both crashes and windfalls). This is critical for stress-testing and risk management, as it warns of more frequent outliers than a Normal process would predict.

- Statistical Modeling
  Many statistical tests (e.g., ANOVA, regression) assume Normally distributed errors. High kurtosis can violate this assumption,

leading to incorrect conclusions. Checking kurtosis helps validate your model and guides you toward more robust analytical methods.

- Process Understanding
  In engineering or manufacturing, the shape of a distribution reveals process behavior. Low kurtosis might indicate a process with consistent, controlled variation, while high kurtosis could suggest intermittent, severe faults.

The kurtosis of a Bernoulli is

$$\gamma_4(Y) \stackrel{\text{def}}{=} E\left[\left(\frac{Y-\mu}{\sigma}\right)^4\right] \tag{22.4.30}$$

$$= \sum_{i\in\{0,1\}} \left(\frac{y_i-\mu}{\sigma}\right)^4 f(y_i) \tag{22.4.31}$$

$$= \left(\frac{0-\pi}{\pi(1-\pi)}\right)^4 \times (1-\pi) + \left(\frac{1-\pi}{\pi(1-\pi)}\right)^4 \times (\pi) \tag{22.4.32}$$

... some algebra ...

$$= \frac{1-3\pi(1-\pi)}{\pi(1-\pi)} \tag{22.4.33}$$

Here, we are defining kurtosis as the "fourth standardized moment."

The usual measure is called the "**excess kurtosis**." This is just the kurtosis minus 3. Why 3? The kurtosis of the Normal distribution is 3. Thus, the excess kurtosis measures how its kurtosis differs from the Normal. Thus, the excess kurtosis of the Bernoulli is

$$\frac{1-6\pi(1-\pi)}{\pi(1-\pi)} \tag{22.4.34}$$

Note

While the Bernoulli distribution is heavily used only in Chapter 15: Binary Dependent Variables, it is a simple distribution that serves as a basis for better understanding other probability distributions

The Binomial Distribution

The Binomial distribution arises from modeling independent repeated trials where the variable is the number of successes out of a known number of trials.

Definition: Binomial Random Variable

Let $X_i \overset{\text{iid}}{\sim} Bern(\pi)$. If we define $Y = \sum_{i=1}^{n} X_i$, then $Y \sim Bin(n, \pi)$.

Here, the symbol $\overset{\text{iid}}{\sim}$ indicates the random variables are **independent and identically distributed**. In other words, they constitute a *random sample*.

Interpreting the above definition gives us the following **five requirements** for a random variable to follow a Binomial distribution:

1. The number of trials, n, is known.
2. Each trial has two possible outcomes: success and failure.
3. The probability of a success, $\pi \in \Pi = (0, 1)$, does not change from trial to trial.
4. The trials are independent.

5. The random variable is the number of successes in those n trials.

If your random variable follows all five of these conditions, then it follows a Binomial distribution with parameters n and π.

If you are interested, the **probability mass function** (pmf) of a Binomial random variable is

$$P[Y = y;\ n, \pi] \ = \ \binom{n}{y} \pi^y (1-\pi)^{n-y} \tag{22.4.35}$$

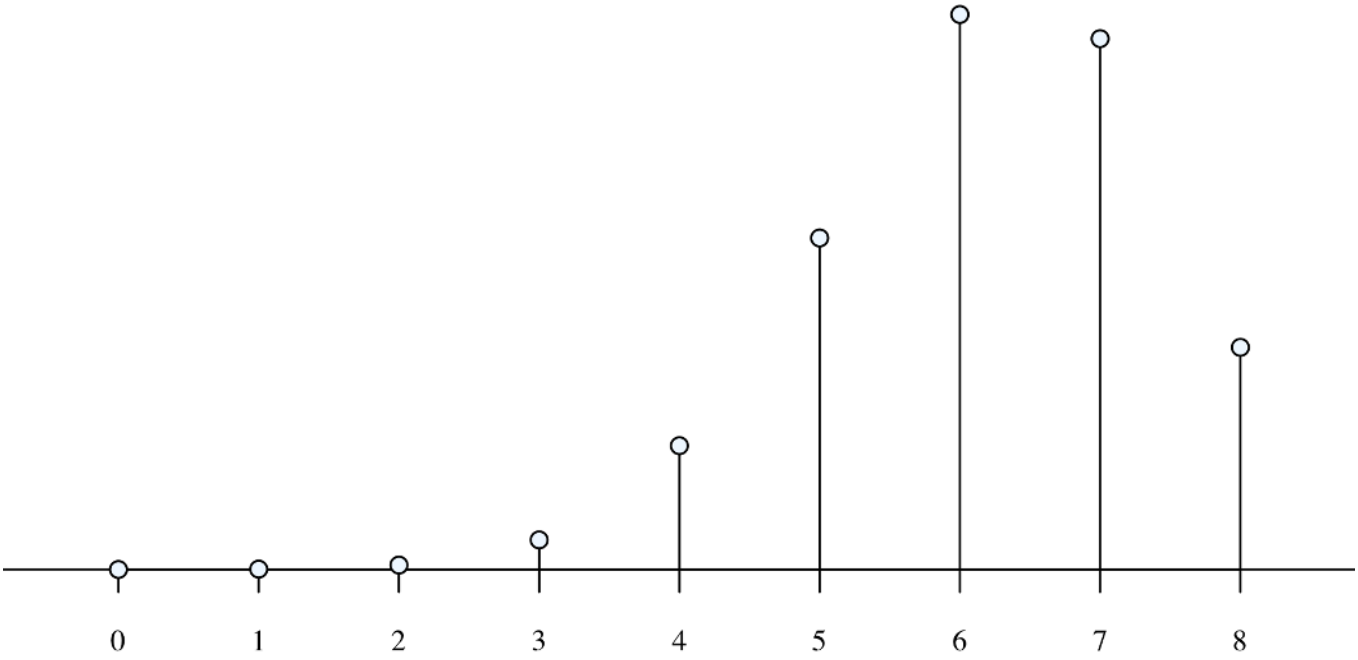


Figure 22.4.3: A graphic of the probability mass function (pmf) of a typical Binomial distribution. Here, $n = 8$ and $\pi = 0.77$. What is the mean?

The Binomial distribution is symmetric only when $\pi = 0.500$. If $\pi < 0.500$, then it is skewed right. Otherwise, it is skewed left. The sample space of a Binomial random variable is $\mathcal{S} = \{0, 1, 2, \ldots, n\}$. The expected value is $E[Y] = n\pi$ and the variance is $V[Y] = n\pi(1 - \pi)$.

A Simple Normal Approximation

Under certain circumstances, the Binomial distribution can be **approximated** with the Normal distribution. This arises from the Central Limit Theorem (Section: The Central Limit Theorem). This is *especially important* if we are examining the distribution of a proportion instead of a count.

Theorem 22.4.1: Normal Approximation to the Binomial

Let $Y \sim Bin(n,\ \pi)$. The distribution of the number of successes is

$$Y \dot{\sim} N\Big(n\pi,\ n\pi(1-\pi)\Big) \tag{22.4.36}$$

Here, the symbol $\dot{\sim}$ indicates the distribution is approximate. As one would expect, the approximation improves as the sample size, n, increases. This is due entirely to the Central Limit Theorem (Section: The Central Limit Theorem) and the fact that a Binomial random variable is the sum of independent and identically distributed (iid) Bernoulli random variables.

Also, the approximation improves if any of a variety of "continuity corrections" are used.

Theorem 22.4.2: Distribution of Sample Proportion

Let $Y \sim Bin(n,\ \pi)$. The distribution of the sample proportion, $P \overset{\text{def}}{=} \frac{Y}{n}$ is

$$P \dot{\sim}\ N\left(\pi,\ \frac{\pi(1-\pi)}{n}\right) \tag{22.4.37}$$

As one would expect from the Central Limit Theorem, the approximation improves as the sample size, n, increases.

Proof.

This proof proceeds from approximating the distribution of Y with a Normal distribution (Theorem: Normal Approximation to the Binomial), then using the characteristics of the Normal distribution to obtain the answer. I leave this as an exercise.

The Poisson Distribution

The Poisson distribution arises from counting the number of successes over a time period or an area. Contrast this with the Binomial, where the successes were counted over the number of trials.



Siméon Denis Poisson, c.1840
(Source: Wikipedia)

The probability mass function of the Poisson distribution is

$$P[Y = Y;\ \lambda] = \frac{e^{-\lambda}\lambda^{y}}{y!} \tag{22.4.38}$$

The expected value is $E[Y] = \lambda$, and the variance is $V[Y] = \lambda$. The sample space is $\mathcal{S} = \{0, 1, 2, ...\}$. It is skewed right, regardless of the value of λ, however that skew goes to zero as $\lambda \to \infty$. If the skew is defined as the standardized third moment, we can "easily" show that

$$\gamma_3(Y) = E\left[\left(\frac{Y-\mu}{\sigma}\right)^3\right] = \lambda^{-1/2} \tag{22.4.39}$$

Similarly, if we define the kurtosis as the standardized fourth moment, we know the *excess kurtosis* converges to zero as $\lambda \to \infty$

$$\text{excess kurtosis} = E\left[\frac{(X-\mu)^4}{\sigma^4}\right] - 3 = \lambda^{-1} \tag{22.4.40}$$

These previous results show that the Poisson distribution becomes more and more Normal as $\lambda \to \infty$.

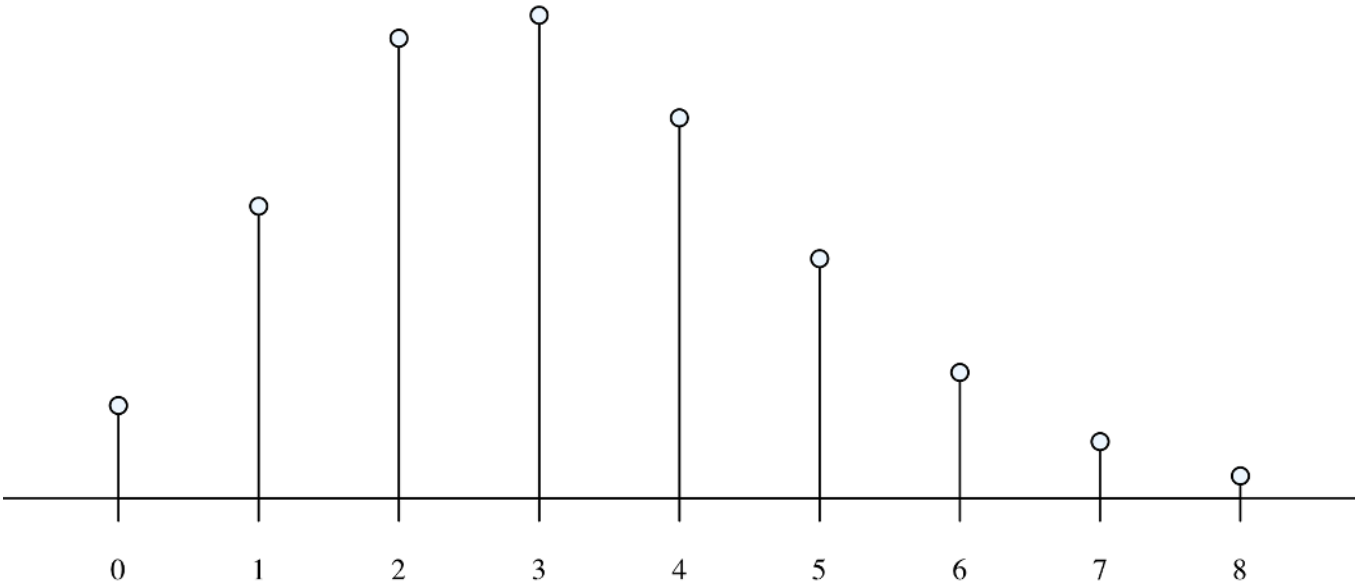


Figure 22.4.4: Here, not that the Poisson continues *ad infinitum* to the right. This Poisson has parameter $\lambda = 3.15$.

Note

The Poisson distribution is an example of an "**infinitely divisible**" distribution. This means that any Poisson distribution is the sum of two other Poisson distributions. This characteristic is rare, but it also holds for the Normal distribution. It is important because the Central Limit Theorem applies to sums of random variables. Since the Poisson distribution is a sum of other Poisson distributions, the CLT tells us that the Poisson distribution converges to the Normal distribution (as $\lambda \to \infty$).

The Poisson distribution is the number of successes over a time period or an area. This suggests that the Poisson distribution arises as a limiting case of the Binomial distribution when $n \to \infty$, and $\mu = n\pi$ is a constant value. We prove this in the next theorem.

Theorem 22.4.3: Poisson as the Limit of a Binomial

If $Y_n \sim Bin(n, \pi)$, then

$$\lim_{n\to\infty} Y_n \sim Pois(\lambda) \tag{22.4.41}$$

as long as $\mu = n\pi = \lambda$ remains constant.

Proof.

This proof heavily relies on Sterling's approximation, $n! \approx \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$, which you could have seen in your calculus course.

https://stats.libretexts.org/@go/page/57825

$$P[Y = y;\ n, \pi] = \binom{n}{y} \pi^y (1-\pi)^{n-y} \tag{22.4.42}$$

$$= \frac{n!}{(n-y)!\ y!} \pi^y (1-\pi)^{n-y} \tag{22.4.43}$$

$$\approx \frac{\sqrt{2\pi n}\left(\frac{n}{e}\right)^n}{\left(\sqrt{2\pi(n-y)}\left(\frac{n-y}{e}\right)^{n-y}\right)\ y!} \pi^y (1-\pi)^{n-y} \tag{22.4.44}$$

$$= \sqrt{\frac{n}{n-y}}\ \frac{n^n e^{-y}}{(n-y)^{n-y}\ y!} \pi^y (1-\pi)^{n-y} \tag{22.4.45}$$

Now, let $n \to \infty$ to give us

$$= 1\ \frac{n^n e^{-y}}{(n-y)^{n-y}\ y!} \pi^y (1-\pi)^{n-y} \tag{22.4.46}$$

and holding $n\pi = \lambda$ (i.e., $\pi = \lambda/n$), we have

$$P[Y = y;\ n, \pi] = \frac{n^n\ e^{-y}}{n^{n-y}\left(1-\frac{y}{n}\right)^{n-y}\ y!} \left(\frac{\lambda}{n}\right)^y \left(1-\frac{\lambda}{n}\right)^{n-y} \tag{22.4.47}$$

$$= \frac{\lambda^y \left(1-\frac{\lambda}{n}\right)^{n-y} e^{-y}}{\left(1-\frac{y}{n}\right)^{n-y}\ y!} \tag{22.4.48}$$

As $n \to \infty$, we have $n - y \approx n$. This gives

$$P[Y = y;\ n, \pi] \approx \frac{\lambda^y \left(1-\frac{\lambda}{n}\right)^{n} e^{-y}}{\left(1-\frac{y}{n}\right)^{n}\ y!} \tag{22.4.49}$$

Remember from calculus that $n \to \infty$ means $\left(1-\frac{y}{n}\right)^n \to e^{-y}$, *by definition*. Now, applying this limit, we have

$$P[Y = y;\ n, \pi] = \frac{\lambda^y e^{-\lambda} e^{-y}}{e^{-y}\ y!} \tag{22.4.50}$$

$$= \frac{\lambda^y e^{-\lambda}}{y!} \tag{22.4.51}$$

Since this is the probability mass function of the Poisson distribution, the theorem is proven.

■

Thus, we have shown that the Poisson distribution is the limiting distribution of a Binomial distribution when the number of trials goes to infinity and the expected value remains constant (the probability of success goes to zero).

> Stop and think about *why* this fact suggests that the Poisson can model successes over time or space.

Note

The reason for understanding the Poisson distribution is that it is used to model counts (number of successes over time or space). It is the focus of count regression of Chapter 17: Count Dependent Variables.

The Exponential Distribution

The exponential distribution is widely used in various fields due to its property of modeling the time between events in a Poisson process, where events occur continuously and independently at a constant average rate. In reliability engineering and survival analysis, it is a fundamental tool for modeling the lifespan of components or the time until failure of a system, particularly for items

with a constant hazard rate, meaning they are memoryless. In queueing theory, it is used to model the inter-arrival times of customers at a service point or the service times themselves, forming the basis for models like M/M/1 queues. Other key applications include telecommunications, where it models the time between incoming calls or data packets; finance, for modeling the time until a default event in credit risk modeling; and epidemiology, for describing the incubation period of certain diseases. Its mathematical simplicity and the memoryless property make it a convenient and powerful first-choice model for analyzing waiting times and durations.



Pierre-Simon de Laplace, c.1850
(Source: Wikipedia)

Its **sample space** is all positive real numbers, $\mathcal{S} = (0, \infty)$.

The **probability density function** (pdf) of an Exponential is either

$$f(y; \lambda) = \lambda \, e^{\lambda y} \tag{22.4.52}$$

or

$$f(y; \theta) = \frac{1}{\theta} \, e^{y/\theta} \tag{22.4.53}$$

Note that there are two main parameterizations of the Exponential distribution. I tend to use the λ parameterization because it's more fun to write a λ than a θ. Note that the θ is infrequently a β and represents the "scale" of the distribution. So, that's three parameterizations for the Exponential. That's not a record, by any means. All it means to us is that we need to know what parameterization is being used. I will stick with the λ parameterization for fun and profit.

The **expected value** and **standard deviation** are $\mu = \sigma = 1/\lambda$. The median is $\ln 2/\lambda$. The **skew** is $\gamma_1 = 2$, and the **excess kurtosis** is 6.

The **cumulative distribution function** (CDF) is

$$F(y; \lambda) = \begin{cases} 1 - e^{-\lambda x} & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{22.4.54}$$

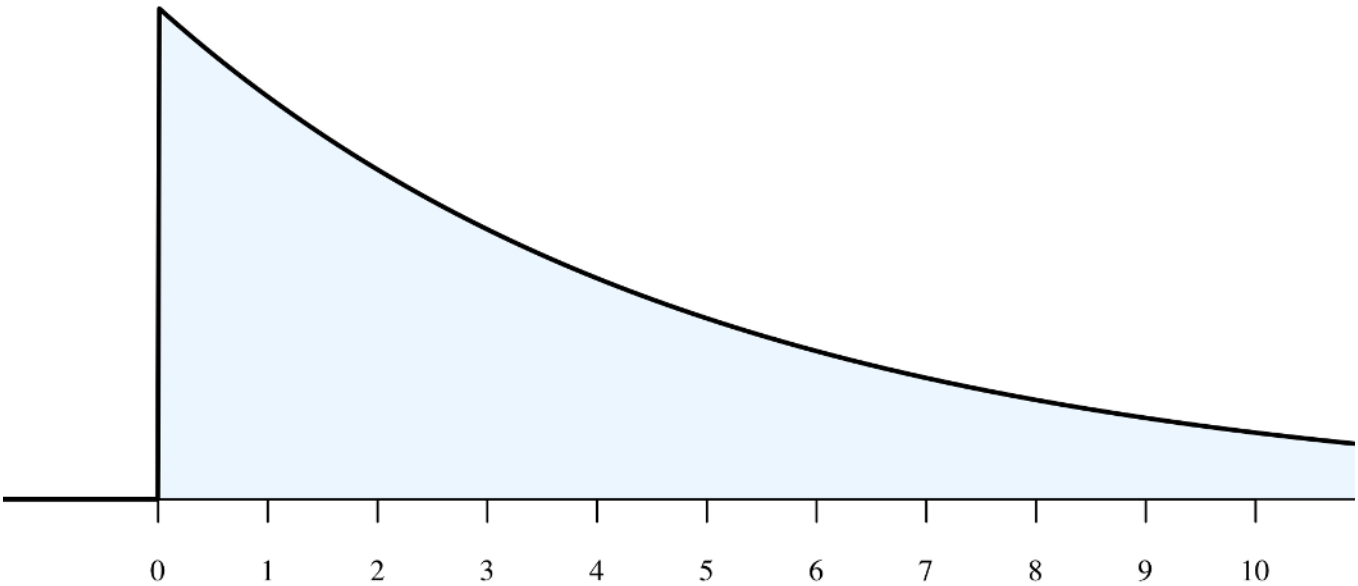


Figure 22.4.5: The probability density function for an Exponential distribution. Here, $\lambda = 0.2$. Thus, the expected value is 5... and the height of the Exponential at its mode is 0.2.

https://stats.libretexts.org/@go/page/57825

The Memoryless Property

Because it is one of the defining hallmarks of the Exponential distribution, let's look at its memoryless property. In terms of linear regression, it's little more than a curiousity. However, for statistics and data science in general, it is quite useful. Note that the exponential distribution is *uniquely* memoryless among continuous distributions, meaning the probability of an event occurring in the next interval is independent of how much time has already passed. Formally, for $X \sim \text{Exp}(\lambda)$ and $s, t > 0$, the property states $P(X > s + t \mid X > s) = P(X > t)$. This characteristic makes it ideal for modeling waiting times where the future is independent of the past, such as radioactive particle decay or the arrival of the next customer at a service desk.

Proof.
Using the definition of conditional probability and the exponential survival function $P(X > x) = e^{-\lambda x}$:

$$\begin{aligned} P(X > s + t \mid X > s) &= \frac{P(X > s + t \cap X > s)}{P(X > s)} \\ &= \frac{P(X > s + t)}{P(X > s)} \quad (\text{since } s + t > s) \\ &= \frac{e^{-\lambda(s+t)}}{e^{-\lambda s}} \\ &= e^{-\lambda t} \\ &= P(X > t). \end{aligned}$$

As a Transformed Uniform???

The last thing I want to discuss is not unique to the Exponential. I mentioned above in the Uniform distribution section that it was the grandfather of all distributions in that all distributions can be generated from the Uniform. Frequently, this is done through easy definitions (like the Bernoulli) or limiting cases (like the Poisson and Normal). However, it can also be done through the "**Probability Integral Transform**," which was was introduced by Ronald Fisher in his 1932 edition of the book *Statistical Methods for Research Workers*.

What it means is that the standard Uniform distribution can be transformed into (just about) **any other distribution**. For instance, an Exponential random variable can be generated from a Uniform through this transformation:

$$-\frac{1}{\lambda}\log(U) \sim Exp(\lambda) \tag{22.4.55}$$

Here, $U \sim Unif(0, 1)$, a standard Uniform distribution.

This is not unique to the Exponential. Also, you should just file this under "hmmmm... interesting" and nothing more.

The Normal (Gaussian) Distribution

The Normal distribution, also known as the Gaussian distribution and the Gauss-Laplace distribution, is ubiquitous in statistics. This is due to the Central Limit Theorem (see Section: The Central Limit Theorem), which states that the distribution of the sample sum (or sample mean) approaches a Normal distribution, regardless of how the original variable is distributed (as long and the variance is finite and the sample is random).

The **probability density function** (pdf) of the Normal is

$$f(y;\ \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\ \frac{(y-\mu)^2}{\sigma^2}\right] \tag{22.4.56}$$



Carl Friedrich Gauss
c.1840
(Source: Wikipedia)

Note

The probability mass function (pmf) describes the probability of a discrete random variable taking on a specific exact value. The probability density function (pdf) describes the relative likelihood of a continuous random variable falling near a particular value, where probability is given by the area under the pdf curve over an interval.

In short: a pmf gives actual probabilities for distinct outcomes, while a pmf gives a density that must be integrated to find probabilities for a range of values.

The Normal distribution is symmetric, has expected value $E[Y] = \mu$, variance $V[Y] = \sigma^2$, and sample space $\mathcal{S} = \mathbb{R}$, all real values.

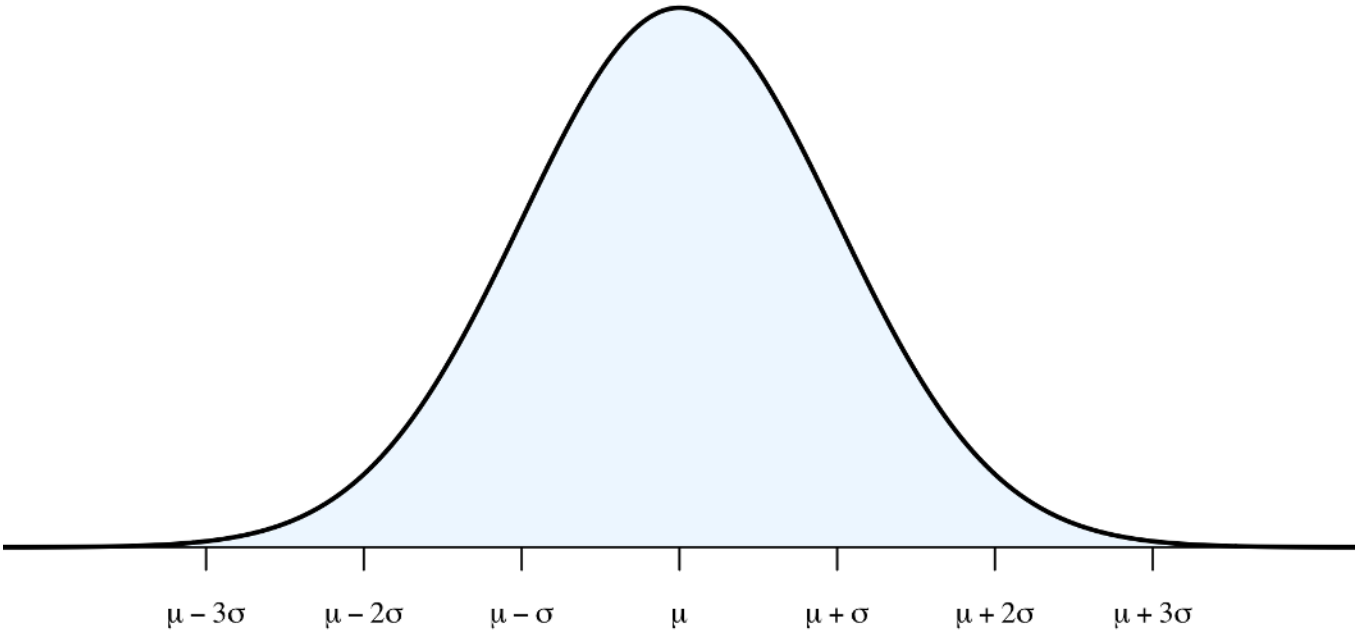


Figure 22.4.6: A Normal distribution showing the mean and several other values. Recall from your introductory statistics course that about 68% of the distribution is between $\mu - \sigma$ and $\mu + \sigma$; 95% between $\mu - 2\sigma$ and $\mu + 2\sigma$; and 99.5% between $\mu - 3\sigma$ and $\mu + 3\sigma$;

Example 22.4.2

Calculate the skew of the Normal distribution.

Solution.

There are several ways to show that the Normal is symmetric (skew = 0). Here are three.

Method 1: Math

The mathematical definition of symmetry is that $f(y)$ is symmetric about the vertical line μ if $f(\mu - y) = f(\mu + y)$. Here is the proof:

$$f(\mu - y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\frac{(\mu - y - \mu)^2}{\sigma^2}\right] \tag{22.4.57}$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\frac{(-y)^2}{\sigma^2}\right] \tag{22.4.58}$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\frac{(y)^2}{\sigma^2}\right] \tag{22.4.59}$$

and

$$f(\mu + y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\frac{(\mu + y - \mu)^2}{\sigma^2}\right] \tag{22.4.60}$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\frac{(y)^2}{\sigma^2}\right] \tag{22.4.61}$$

Thus since $f(\mu - y) = f(\mu + y)$, we have shown $f(y)$ is symmetric about μ.

Method 2: Hildebrand

We can also use the Hildebrand rule to show that the Normal distribution is symmetric:

$$H = \frac{\overline{Y} - \widetilde{Y}}{S} \tag{22.4.62}$$

$$= \frac{\mu - \mu}{\sigma} \tag{22.4.63}$$

$$= 0 \tag{22.4.64}$$

Thus, since $H = 0$, the Normal distribution is a symmetric distribution.

Note that I skipped over the part where I prove $\widetilde{Y} = \mu$. I leave that for later (Theorem: The Median of a Normal).

Method 3: Third Standardized Moment

We can also use the third standardized moment to show that the Normal distribution is symmetric:

$$f(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\ \frac{(y-\mu)^2}{\sigma^2}\right] \tag{22.4.65}$$

$$\gamma_3 = E\left[\left(\frac{Y-\mu}{\sigma}\right)^3\right] \tag{22.4.66}$$

$$= \int_{\mathbb{R}} f(y)\ \left(\frac{y-\mu}{\sigma}\right)^3\ \mathrm{d}y \tag{22.4.67}$$

I leave it as an exercise to expand the cube, separately calculate $E[Y^2]$ and $E[Y^3]$, and do the integration. There is a lot of algebra, but not much more than that if you are careful.

■

There is a wealth of information on the Normal distribution. Arguably, it is the most studied distribution in statistics. It is also the most important distribution in statistics because of the Central Limit Theorem. For these reasons, I offer little to say about it beyond beyond the CLT.

Theorem 22.4.4: The Median of a Normal

The median of a Normal distribution is μ.

Proof.

By definition for a continuous distribution, the median is the value $\tilde{y}$ such that $F(\tilde{y}) = 0.500$. Thus, this proof reduces to a "mere calculation:"

$$F(\mu) = \int_{-\infty}^{\mu} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\ \frac{(y-\mu)^2}{\sigma^2}\right]\ \mathrm{d}y \tag{22.4.68}$$

Since this equals 0.500, we have shown that μ is the median of a Normal distribution.

■

The Chi-Square Distribution

The Chi-square distribution arose from multiple areas. One was the need to model the variation of the sample. A second was in examining categorical variables. It is used in the Chi-square goodness-of-fit test and the Chi-square test of independence. In both of those cases, the test statistic only *approximately* follows a Chi-square distribution.

If you want it (and I'm not entirely sure why you would), here is the probability density function of the Chi-square distribution based on its one parameter, ν, the "number of degrees of freedom:"

$$f(y;\ \nu) = \frac{1}{2^{\nu/2}\,\Gamma\left(\frac{\nu}{2}\right)} y^{\nu/2-1} e^{-y/2} \tag{22.4.69}$$

where $0 < \nu$ and the **gamma function** is defined as

$$\Gamma(y) \stackrel{\text{def}}{=} \int_0^\infty t^{y-1} e^{-t}\ \mathrm{d}t \tag{22.4.70}$$



Prasanta C. Mahalanobis, 1962
(Source: Wikipedia)

Note

The Chi-square distribution is actually the distribution of the second fraction in the exponent in the Normal probability density function. That is, if $Y \sim N(\mu,\ \sigma^2)$, then

$$\frac{(Y-\mu)^2}{\sigma^2} \sim \chi^2_{\nu=1} \tag{22.4.71}$$

The square root of this fraction is called the "Mahalanobis distance." The Mahalanobis distance is unitless, scale-invariant, and takes into account the correlations of the data set. One sees it in data science applications, especially in clustering and outlier detection.

When you read probability papers from around the turn of the 20th century, you may see χ as a variable of interest. Originally, it was just a curvey, fancy x used to illustrate that the Normal distribution could approximate the Binomial distribution:

$$\chi = \frac{x - n\pi}{\sqrt{n\pi(1-\pi)}} \tag{22.4.72}$$

With this, $\chi \dot{\sim} N(0,\ 1)$.

At the start of the 20th century, statisticians were able to turn this equation into a **definition** for the Chi-square distribution:

Definition: The Chi-Square Distribution

Let Z_i be a random sample of size ν from a standard Normal distribution, that is $Z_i \overset{\text{iid}}{\sim} N(0,\ 1)$, then

$$\sum_{i=1}^{\nu} Z_i^2 \sim \chi^2_\nu \tag{22.4.73}$$

In this definition, ν (pronounced "nu") is the number of degrees of freedom, the number of those Normal distributions that are independent.

It is **this definition** that is most helpful in determining what is (and what is *not*) a Chi-square random variable.

Note the following:

- The expected value of a Chi-square distribution is ν, and the variance is 2ν.
- The sample space is $\mathcal{S} = (0, \infty)$.
- It is *always* positively skewed ($\gamma_3(Y) = \sqrt{8/\nu}$), although that skew goes to zero as $\nu \to \infty$.
- Its excess kurtosis also goes to 0 as $\nu \to \infty$ (ex.kurt. = $12/\nu$).
- If the number of degrees of freedom are 2 or less, then the probability density function is strictly decreasing (its first derivative is always less than zero). Among other things, this means that the mode is zero if $\nu \leq 2$.

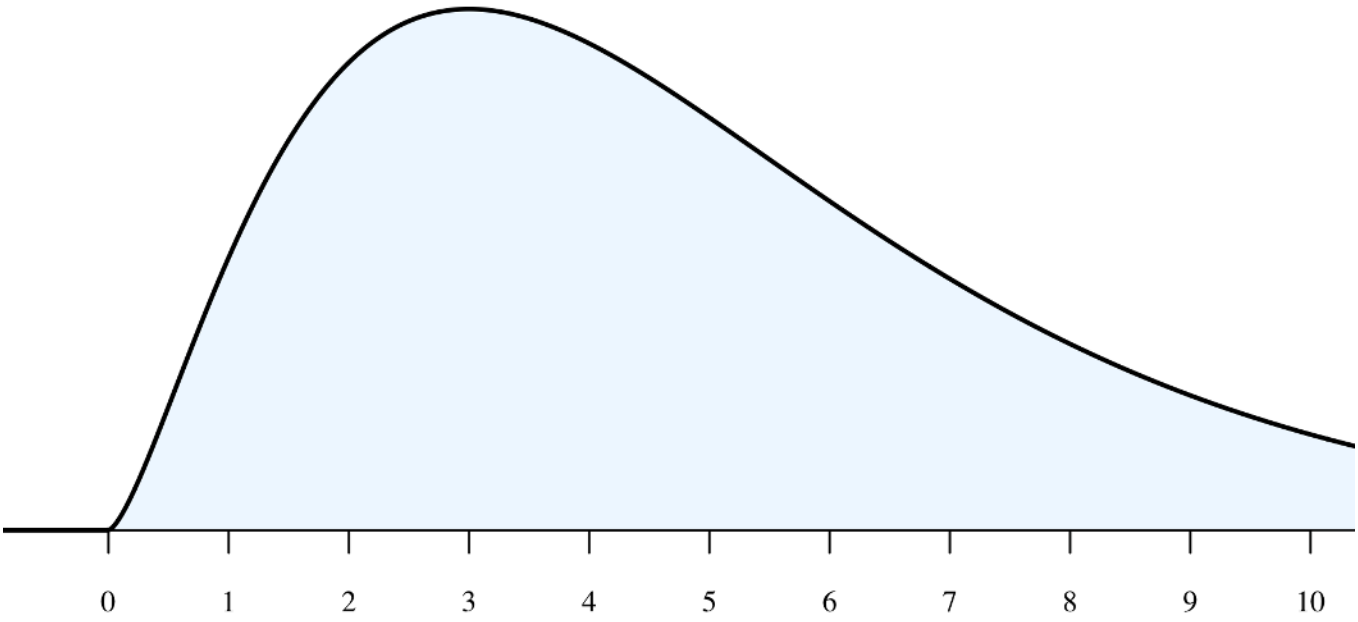


Figure 22.4.7: A Chi-square distribution. Here, $\nu = 5$.

The Student's t Distribution


William Sealy Gosset in 1908
(Source: Wikipedia)

Arguably, most of the previous distributions arise from Nature. The Student's t distribution arises from a need to test certain hypotheses. This distribution is named after its creator, William Sealy Gosset (as pictured in the figure to the right). Gosset worked for Guinness Brewery as a master brewer (and statistician) near the end of the 19th century. He applied the well-known statistical techniques in his job, but found that these techniques had major flaws. Apparently, he felt that he was rejecting far too many bushels of grain based on current statistical techniques.

Specifically, Gosset determined that the test statistic

$$Z = \frac{Y - \mu}{s/\sqrt{n}} \tag{22.4.74}$$

did *not* follow a standard Normal distribution as expected, when the sample size was small. In fact, for his small samples, it was not even "sufficiently" close.

The problem was that Gosset was working with small samples of barley, while most statistics at the time were concerned with large samples. In my opinion, this is Gosset's greatest contribution: paying attention to small-sample properties of estimators. While Guinness supported him, they did not want a repeat of a previous employee who published trade secrets in a scientific journal. Thus, to get his discoveries out there, Gosset had to publish under a pseudonym. He chose "Student."

The following is Gosset's **definition** of his Student's t distribution.

Definition: The Student's t Distribution

Let $Z \sim N(0, 1)$ and $V \sim \chi^2_\nu$ with $Z \perp V$. Define the following ratio:

$$T = \frac{Z}{\sqrt{V/\nu}} \tag{22.4.75}$$

The random variable T follows a Student's t distribution with ν degrees of freedom.

If you care to see it (and some do like the mathematics), this is the probability density function calculated by Fisher:

$$f(y;\ \nu) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi}\,\Gamma\left(\frac{\nu}{2}\right)}\left(1+\frac{y^2}{\nu}\right)^{-\frac{\nu+1}{2}} \tag{22.4.76}$$

Again, $\Gamma(\cdot)$ is the gamma function.

Here are some results:

- Its sample space is $\mathcal{S} = \mathbb{R}$.
- The mean of the t distribution is $E[Y] = 0$, if $\nu > 1$.
- The variance is $V[Y] = \frac{\nu}{\nu-2}$, if $\nu > 2$; ∞, if $\nu \in (1, 2]$; and undefined elsewhere.
- It has zero skew, $\gamma_3(Y) = 0$.
- Its excess kurtosis is $\frac{6}{\nu-1}$, if $\nu > 4$; ∞, if $2 \leq \nu \leq 4$; and undefined elsewhere.

As $\nu \to \infty$, the Student's t distribution converges to the Normal distribution. The proof of this is a nice exploration of the mathematics you already know. You should make sure you can prove it.

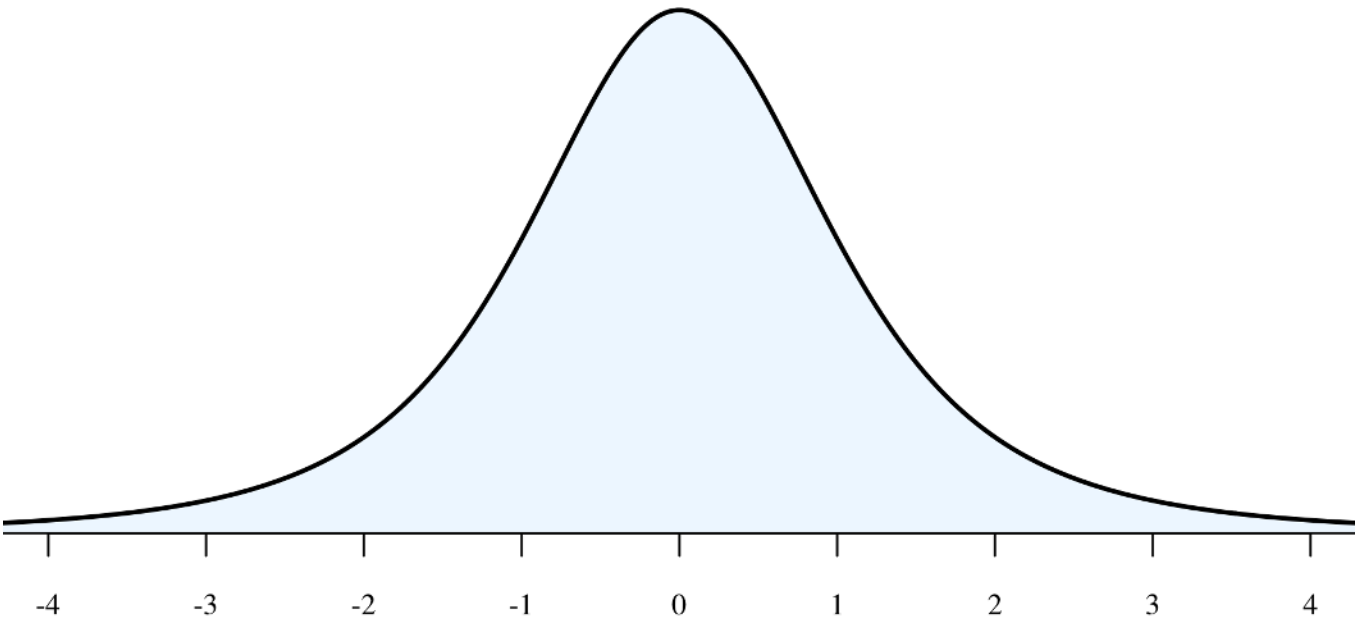


Figure 22.4.8: A typical Student's t distribution. Here, $\nu = 4$.

Note

While the t-distribution was formulated by Gosset to deal with the distribution of a test statistic, it was actually first seen as a posterior distribution a couple decades earlier. However, as Bayesian inference and frequentist statistics were rarely aware of each other, the results by Helmert and Lüroth remained unknown to Gosset and Fisher.

The Cauchy Distribution

The Cauchy distribution is named after Augustin-Louis Cauchy, a French mathematician who specialized in complex analysis and abstract algebra. The distribution is originally (and most helpfully) **defined** as



Augustin-Louis Cauchy
c.1840
(Source: Wikipedia)

Definition: The Cauchy Distribution

Let $Z_1 \sim N(0,\ 1)$, $Z_2 \sim N(0,\ 1)$, and $Z_1 \perp Z_2$. Then, the ratio

$$Y \overset{\text{def}}{=} \frac{Z_1}{Z_2} \tag{22.4.77}$$

follows a standard Cauchy distribution.

Note that this is just a t distribution with one degree of freedom. As such, the probability density function for the **standard Cauchy** is

$$f(y) = \frac{1}{\pi\left(1+y^2\right)} \tag{22.4.78}$$

Regardless of the fact that the standard Cauchy is symmetric with median 0, neither its mean nor its variance exist. I leave this proof as an exercise for you.

The standard Cauchy can be generalized to different medians (locations) and spreads (scales):

$$f(y;\ \eta,\gamma) \overset{\text{def}}{=} \frac{1}{\pi\gamma\left(1+\left(\frac{y-\eta}{\gamma}\right)^2\right)} \tag{22.4.79}$$

Again, neither the mean nor the variance exist, however the median and mode are now η ("eta") and the interquartile range is 2γ ("gamma").

Finally, its cumulative distribution function (CDF) is

$$F(y;\ \eta,\gamma) = \frac{1}{\pi}\operatorname{atan}\left(\frac{y-y_0}{\gamma}\right) + \frac{1}{2} \tag{22.4.80}$$

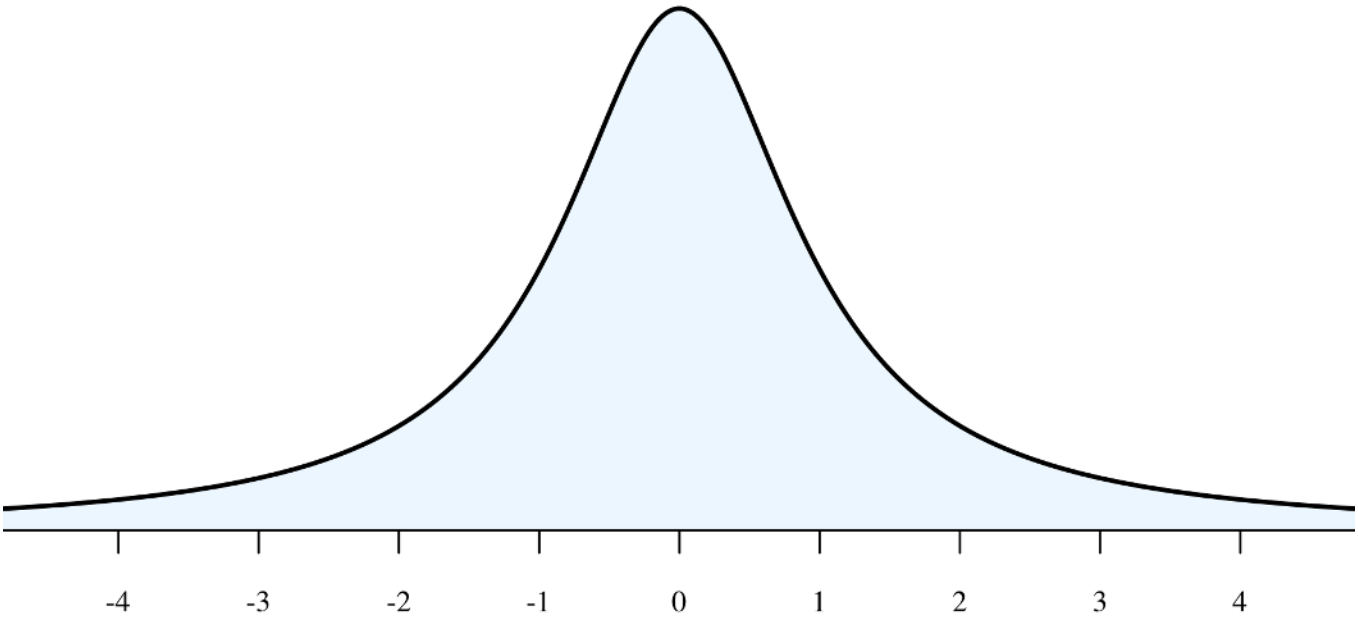


Figure **22.4.9**: This is a Cauchy distribution. It is also a t distribution with 1 degree of freedom. Note that, while this distribution is symmetric and bell-shaped, it has neither a variance nor an expected value. This is because the tails of the distribution are especially heavy (large probability of observing events in the tails). Also, note that the Central Limit Theorem does not apply to the Cauchy distribution. Why?

As befits the magical world of mathematics, the Cauchy distribution pops up in many places. For me, the most interesting place is as the Witch of Agnesi.

I find it very interesting to see how frequently mathematical equations find their way into many different areas of mathematics and science. The key, as I see it, is to understand **how the mathematical expressions came into being**, then see what applies to the current probability/statistics problem. A trip through history may suggest that there is "no progress in the history of [mathematical] knowledge —merely a continuous and sublime recapitulation" (Umberto Eco, *The Name of the Rose*, 1980).

Note

The Cauchy distribution is one of the most helpful distributions available. Its first (and above) moments are not finite. This means the Central Limit Theorem does *not* apply to it. As such, it is frequently used to illustrate the importance of finite variances.

Snedecor's F Distribution

The F distribution, developed by George W. Snedecor (*Calculation and Interpretation of Analysis of Variance and Covariance*, 1934) and named for Ronald Fisher, is frequently used for testing compound hypotheses, notably in the analysis of variance (ANOVA) procedure (see the Ruritanian Crops example).

Definition: Snedecor's F Distribution

Let $V_1 \sim \chi^2(\nu_1)$, $V_2 \sim \chi^2(\nu_2)$, and $V_1 \perp V_2$. Define the following ratio:

$$F = \frac{V_1/\nu_1}{V_2/\nu_2} = \frac{\nu_2\, V_1}{\nu_1\, V_2} \tag{22.4.81}$$

The random variable F follows Snedecor's F distribution with ν_1 and ν_2 degrees of freedom.



George W. Snedecor in 1915
(Source: Wikipedia)

Here are some results:

- The support set is $\mathcal{S} = (0, \infty)$.
- If $\nu_2 > 2$, then the mean of the F distribution is

$$E[F] = \frac{\nu_2}{\nu_2 - 2} \tag{22.4.82}$$

  If $\nu_2 \leq 2$, then the expected value is infinite.

- When $\nu_2 > 4$, its variance is

$$V[F] = \frac{2\nu_2^2(\nu_1 + \nu_2 + 2)}{\nu_1(\nu_2 - 2)^2(\nu_2 - 4)} \tag{22.4.83}$$

  If $\nu_2 \leq 4$, then the variance is infinite.

- The distribution (function) is always right-skewed.

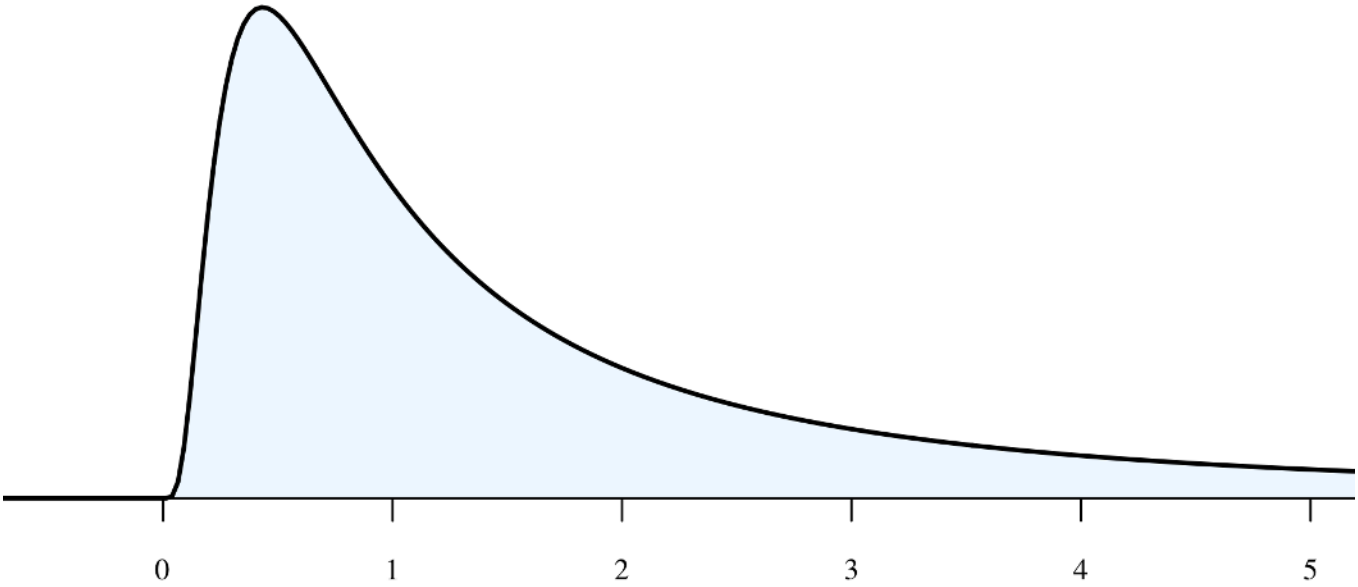


Figure 22.4.10: An F distribution. Here, $\nu_1 = 15$ and $\nu_2 = 2$.

If you want it, here is its probability density function. Note, however, that the definition of the F distribution is *much more helpful*.

$$f(y;\ \nu_1, \nu_2) = \frac{1}{y\ B\left(\frac{\nu_1}{2}, \frac{\nu_2}{2}\right)} \sqrt{\frac{y^{\nu_1}\ \nu_1^{\nu_1}\ \nu_2^{\nu_2}}{(y\ \nu_1 + \nu_2)^{\nu_1+\nu_2}}} \tag{22.4.84}$$

In this formula, $B(\cdot, \cdot)$ is the **beta function**, defined as

$$B(x, y) \stackrel{\text{def}}{=} \int_0^1 t^{x-1}(1-t)^{y-1}\ \mathrm{d}t \tag{22.4.85}$$

Interesting... There seems to be some relationship between the beta function and the Binomial distribution.

By the way, if you would prefer writing the beta function in terms of the **gamma function**,

$$B(x, y) = \frac{\Gamma(x)\ \Gamma(y)}{\Gamma(x+y)} \tag{22.4.86}$$

Most likely, you will have seen the gamma function in your calculus class. If not, there is nothing to worry about, since the computer will be performing the calculations for you. Focus on *how these distributions arise* (i.e., their actual *definition*) and how they are used.

Making These Graphics

So, how did I generate these graphics? Of course, I used R. This allowed me to save the script and share it with y'all. The script actually requires no additional packages; it runs on the base R you download from CRAN.

- Here is the script.

Enjoy!

This page titled 22.4: Probability Distributions is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

22.5: Distributions of Sample Statistics

Now, let us synthesize the ideas from the previous sections. We have established that sample statistics are random variables, and that each population parameter is a fixed target. The true power of statistical inference emerges when we connect these concepts by knowing the distributions of our sample statistics. This knowledge — understanding how a statistic like the sample mean or variance behaves across all possible samples — is what allows us to move from a single, observed data point to broader conclusions. It enables us to quantify uncertainty, construct confidence intervals, and test hypotheses.

Ultimately, it is through the known distribution of a statistic that we can rigorously draw conclusions about an unknown population parameter using only the information contained in a random sample.

✦•·····• ✦ •·····•✦

Sample Mean

The distribution of the sample mean is one of the first sampling distributions you would have come across in your introductory statistics course. It usually serves as the link between the Normal distribution and confidence intervals.

Theorem 22.5.1: Distribution of the Sample Mean

If $Y_i \sim N(\mu,\ \sigma^2)$, and we collect an independent and identically distributed (iid) sample of size n, then

$$\overline{Y} \sim N\left(\mu,\ \frac{\sigma^2}{n}\right) \tag{22.5.1}$$

Proof.

This proof proceeds in three parts. The first part notes that a linear combination of independent Normal random variables is also a Normal random variable. (The proof of this can be found at Theorem: Sum of Normals.)

Since the Normal distribution has two parameters, mean and variance, the second and third parts determine the expected value and variance of that random variable.

$$E\left[\overline{Y}\right] = E\left[\frac{1}{n}\sum_{i=1}^{n} Y_i\right] \tag{22.5.2}$$

$$= \frac{1}{n}\sum_{i=1}^{n} E[Y_i] \tag{22.5.3}$$

$$= \frac{1}{n}\sum_{i=1}^{n} \mu \tag{22.5.4}$$

$$= \frac{1}{n} n\mu \tag{22.5.5}$$

$$= \mu \tag{22.5.6}$$

Thus, the expected value of $\overline{Y}$ is μ. Now for the variance:

$$V\left[\overline{Y}\right] = V\left[\frac{1}{n}\sum_{i=1}^{n} Y_i\right] \tag{22.5.7}$$

$$= \frac{1}{n^2} V\left[\sum_{i=1}^{n} Y_i\right] \tag{22.5.8}$$

$$= \frac{1}{n^2}\sum_{i=1}^{n} V[Y_i] \tag{22.5.9}$$

$$= \frac{1}{n^2}\sum_{i=1}^{n} \sigma^2 \tag{22.5.10}$$

$$= \frac{1}{n^2} n\sigma^2 \tag{22.5.11}$$

$$= \frac{\sigma^2}{n} \tag{22.5.12}$$

Putting these parts together gives us our conclusion.

■

Note

Notice the procedure in the previous proof. The first step is to determine the distribution. The remaining steps determine the values of the distribution's parameters. We needed to determine the values of μ and σ^2 because the Normal distribution uses both as parameters.

Note

That the expected value of $\overline{Y}$ is μ means that the sample mean is an *unbiased* estimator of the population mean.

Sample Variance

The distribution of the sample variance is rarely covered in introductory statistics. However, it is needed when determining some important test statistics.

Theorem 22.5.2: Distribution of the Sample Variance

If $Y_i \overset{\text{iid}}{\sim} N(\mu,\ \sigma^2)$, and we collect a random sample of size n, then

$$\frac{(n-1)S^2}{\sigma^2} \sim \chi^2_{\nu=n-1} \tag{22.5.13}$$

Proof.

Since we need to show that it follows a Chi-square distribution, we simply use the definition of that distribution, performing some algebra to ensure it is in the right form.

First, since we will need it in the future of this proof, please recall

$$S_y^2 = \frac{1}{n-1}\sum_{i=1}^{n} \left(Y_i - \overline{Y}\right)^2 \tag{22.5.14}$$

This is equivalent to

$$(n-1)S_y^2 = \sum_{i=1}^{n} \left(Y_i - \overline{Y}\right)^2 \tag{22.5.15}$$

Now, remember the definition of a Chi-square random variable:

$$\sum_{i=1}^{n} \left(\frac{Y_i - \mu}{\sigma}\right)^2 \sim \chi_n^2 \tag{22.5.16}$$

Note that this is very similar to our definition of S^2. The difference is the presence of μ. So, for reasons that will become obvious later, let us subtract and add $\overline{Y}$ to the numerator of 22.5.16, then expand the square:

$$\chi_n^2 \sim \sum_{i=1}^{n} \left(\frac{Y_i - \overline{Y} + \overline{Y} - \mu}{\sigma}\right)^2 \tag{22.5.17}$$

$$= \sum_{i=1}^{n} \left(\frac{(Y_i - \overline{Y}) + (\overline{Y} - \mu)}{\sigma}\right)^2 \tag{22.5.18}$$

$$= \sum_{i=1}^{n} \left(\frac{Y_i - \overline{Y}}{\sigma}\right)^2 + \sum_{i=1}^{n} \left(\frac{\overline{Y} - \mu}{\sigma}\right)^2 + 2\sum_{i=1}^{n} \left(\frac{(Y_i - \overline{Y})(\overline{Y} - \mu)}{\sigma}\right) \tag{22.5.19}$$

Note that the third term is zero:

$$2\sum_{i=1}^{n} \left(\frac{(Y_i - \overline{Y})(\overline{Y} - \mu)}{\sigma}\right) = 2\left(\frac{(\overline{Y} - \mu)}{\sigma}\right) \sum_{i=1}^{n} (Y_i - \overline{Y}) \tag{22.5.20}$$

$$= 2\left(\frac{(\overline{Y} - \mu)}{\sigma}\right) 0 \tag{22.5.21}$$

$$= 0 \tag{22.5.22}$$

With that, we have:

$$\chi_n^2 \sim \sum_{i=1}^{n} \left(\frac{Y_i - \overline{Y}}{\sigma}\right)^2 + \sum_{i=1}^{n} \left(\frac{\overline{Y} - \mu}{\sigma}\right)^2 \tag{22.5.23}$$

$$= \frac{\sum_{i=1}^{n} (Y_i - \overline{Y})}{\sigma^2} + n\left(\frac{\overline{Y} - \mu}{\sigma}\right)^2 \tag{22.5.24}$$

$$= \frac{(n-1)S^2}{\sigma^2} + n\left(\frac{\overline{Y} - \mu}{\sigma}\right)^2 \tag{22.5.25}$$

And: $$\chi_n^2 \sim \frac{(n-1)S^2}{\sigma^2} + \left(\frac{\overline{Y} - \mu}{\sigma/\sqrt{n}}\right)^2 \tag{22.5.26}$$

The second term is the square of a standard Normal distribution; that is, it follows a χ_1^2 distribution. Since the sum of two Chi-square distributions is another Chi-square distribution (with ν being the sum of the degrees of freedom), we have our result:

$$\frac{(n-1)S^2}{\sigma^2} \sim \chi_{n-1}^2 \tag{22.5.27}$$

Note

Do you remember from the Chi-square section that the expected value of a Chi-square random variable is ν? Use that to see why we needed to divide by $n-1$ in the formula for the sample variance to ensure S^2 is an unbiased estimator of σ^2.

That is, you should be able to prove $E[S^2] = \sigma^2$.

By the way, there is *another* proof using algebra, but it gives little insight into probability. Because of this, I forgo it. However, it is a really nice exercise in high school algebra and statistics definitions.

? Exercise 22.5.1

Prove $E[S^2] = \sigma^2$.

Note

The sample variance can be extended to this thing called the "**Mean Squared Error**."

$$\text{MSE} = \frac{1}{n-p}\sum_{i=1}^{n}\left(Y_i - \overline{Y}\right)^2 \tag{22.5.28}$$

The difference is that the variance uses $n-1$ degrees of freedom, while the MSE uses $n-p$ degrees of freedom. Why the difference? When using the sample variance, you are only estimating **one** parameter (μ), whereas the MSE is used when there is more than one parameter needing to be estimated (usually, β_0, (\beta_1\), etc.). The MSE will be very important in linear regression, because it estimates the residual variance.

Theorem 22.5.3: Distribution of the Mean Squared Error

If $Y_i \overset{\text{iid}}{\sim} N(\mu,\ \sigma^2)$, and we collect a random sample of size n, then

$$\frac{(n-p)\ \text{MSE}}{\sigma^2} \sim \chi^2_{\nu=n-p} \tag{22.5.29}$$

This page titled 22.5: Distributions of Sample Statistics is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

22.6: Other Topics

This section holds some topics that I do not know where to put otherwise. You will have discussed the effects of multiple comparisons on the legitimacy of statistical conclusions. That is why you had to use analysis of variance instead of multiple t-tests for comparing more than two population means.

Next, I am introducing the runs test here. The runs test is important in testing the claim that the model systematically fits the observed data. The logic behind it is very straight-forward, being based on a Binomial random variable.

Finally, I provide a proof of the Central Limit Theorem. You will most certainly have heard of the Central Limit Theorem in your earlier statistics course but not seen its proof. However, the work below with moment generating functions is provided solely for those who are interested in how to prove the Central Limit Theorem.

✦•·····• ✦ •·····•✦

A Check for Skewness

There are several tests for skewness, and for estimating the level of skew in a distribution. However, most are beyond the scope of this book. In lieu of these, let us use the Hildebrand Rule as a good *Rule of Thumb* for determining if the data are "sufficiently" skewed (Hildebrand 1986, p43).

For the sample, the ratio is defined as

$$H = \frac{\bar{x} - \tilde{x}}{s} \tag{22.6.1}$$

According to Hildebrand, if this ratio is greater than 0.20, then the data (or population) are positively skewed. If this ratio is less than -0.20, then the data (or population) are *negatively* skewed. Otherwise, there is no significant skew.

Of course, this is just a Rule of Thumb that completely ignores the sample size and precision of the estimate. However, it is not overly terrible. If, however, the skew of the data is important, one should use a genuine statistical test.

Multiple Comparisons

Remember from your previous statistics class the problem with multiple comparisons. When performing multiple tests, one needs to adjust the value of the p-value to ensure that you do not reject true null hypotheses at too high a rate (α).

The first part illustrates why. The second part introduces the Bonferroni adjustment. Note that the Bonferroni *always* "works," but it tends to be rather conservative. That is, it controls the Type I error rate, but at the expense of reducing the **power** of the test (higher Type II error rate, β).

Multiple Testing

To start this section, recall the meaning of α. It is the proportion of the time we wrongly reject a true null hypothesis (i.e., commit a Type I error). Since we are basing our statistical inference on our *selected* value of α, we want to ensure that we are rejecting at the right frequency (or with the right probability). At the very least, we want to ensure that we do not reject too frequently. If we reject too frequently, we are biasing our conclusions towards rejecting null hypotheses when we should not.

Note

If we reject too **in**frequently, then we will fail to reject the null hypothesis too often. This could also be a bad thing. It really depends on the penalty for being each type of wrong. In the framework of the traditional hypothesis testing, we wish to avoid rejecting the null hypothesis too frequently. This leads to the focus on ensuring we reject the null hypothesis at the α rate or less. Chapter 15: Binary Dependent Variables explores consequences of only controlling α.

To see the effect of multiple tests on the probability of rejecting a true null hypothesis (committing a Type I error), let us look at the following two examples.

Example 22.6.1

Let the null hypothesis be true. Let us also collect a sample and test the null hypothesis using that sample. What is the probability we reject the true null hypothesis if we state our accepted Type I error rate is α?

Solution.
By definition, as long as the test is appropriate for the null hypothesis, the probability of wrongly rejecting the null hypothesis is α.

Example 22.6.2

Let the null hypothesis be true. Let us collect a sample and test the null hypothesis using that sample. Let us now collect a second, independent, sample and test the null hypothesis. What is the probability we reject the true null hypothesis in *either* sample if we state our Type I error rate is α?

Solution.
By definition, for an appropriate test of the null hypothesis, the probability of wrongly rejecting the null hypothesis in *each* test is α.

Let us define Y as the number of samples for which the test rejects the null hypothesis. Clearly, if the tests are independent (if the samples are independent), then $Y \sim Bin(2, \alpha)$, and we need to calculate $P[Y \geq 1]$.

As such, the probability is

$$P[Y \geq 1] = 1 - P[Y = 0] \tag{22.6.2}$$
$$= 1 - \binom{2}{0} \alpha^0 (1-\alpha)^2 \tag{22.6.3}$$

If $\alpha = 0.05$, then $P[Y \geq 1] = 0.0975$. In other words, we claim the Type I error rate is 0.05, but it is really 0.0975 almost twice as much!!

■

Allow me to repeat that result: If we test the null hypothesis twice, then the *actual* Type I error rate is almost *twice* what we claim. This is very problematic, especially for hypotheses that require several sub-tests to fully test, or for times we test multiple hypotheses using the same data.

The Bonferroni Adjustment

One of the first ways for ensuring that the real Type I error rate is not larger than the claimed is the Bonferroni adjustment (Bonferroni 1936; Dunn 1958; Dunn 1961). While the Bonferroni adjustment *always* controls the Type I error rate, it is only the best option when nothing else is available; it reduces the power of the test (i.e., it increases the probability of a Type II error).

Theorem 22.6.1

Let the target Type I error rate be α, with k tests being performed. To ensure that the experimentwise Type I error rate is *no greater than* α, each test should be rejected at the α/k level.

Proof.
If we reject each test at the α/k level, the experiment-wise error rate for the k tests is

$$\text{EWER} = P\left[\bigcup_{i=1}^{k} \mathbb{1}_{P_i \leq \frac{\alpha}{k}}\right] \tag{22.6.4}$$

$$\leq \sum_{i=1}^{k} P\left[P_i \leq \frac{\alpha}{k}\right] \tag{22.6.5}$$

$$= \sum_{i=1}^{k} \frac{\alpha}{k} \tag{22.6.6}$$

$$= k\,\frac{\alpha}{k} = \alpha \tag{22.6.7}$$

Note that $\mathbb{1}_{(P_i \leq \frac{\alpha}{k})}$ equals 1 when $P_i \leq \frac{\alpha}{k}$ and 0 otherwise. The function $\mathbb{1}$. is called the "indicator function" for this very reason. It "indicates" if the condition in the braces is true. (What is the distribution of the indicator function?)

Also note that P_i is the p-value for each test. If the test is appropriate, then the p-value follows a standard uniform distribution, $Unif(0,1)$

With those reminders, the rest is just algebra.

■

Note

Note the inequality. Under what circumstances is it an equality?

The Bonferroni adjustment always "works." In other words, applying the Bonferroni adjustment *always* ensures that the Type I error rate for the entire battery of tests is at most α. However, it is usually a poor adjustment in that it over-corrects for multiple comparisons.

To see this, let us make the assumption that the tests are independent. This additional information allows us to create a tighter adjustment.

Theorem 22.6.2

Let the target Type I error rate be α, with k **independent** tests being performed. To ensure that the experiment-wise Type I error rate is no greater than α, each test should be rejected at the $1-(1-\alpha)^{1/k}$ level.

Proof.
If we reject each test at the $1-(1-\alpha)^{1/k}$ level, the experiment-wise error rate for the k tests is

$$\text{EWER} = P\left[\bigcup_{i=1}^{k} \mathbb{1}_{P_i \leq 1-(1-\alpha)^{1/k}}\right] \tag{22.6.8}$$

By De Morgan's Laws, this is

$$= 1 - P\left[\bigcap_{i=1}^{k} \mathbb{1}_{P_i > 1-(1-\alpha)^{1/k}}\right] \tag{22.6.9}$$

By independence, this is

$$= 1 - \prod_{i=1}^{k} P\left[P_i > 1 - (1-\alpha)^{1/k}\right] \tag{22.6.10}$$

$$= 1 - \prod_{i=1}^{k} \left(1 - P\left[P_i \leq 1 - (1-\alpha)^{1/k}\right]\right) \tag{22.6.11}$$

$$= 1 - \prod_{i=1}^{k} \left(1 - \left(1 - (1-\alpha)^{1/k}\right)\right) \tag{22.6.12}$$

Finishing with algebra

$$\text{EWER} = 1 - \prod_{i=1}^{k} (1-\alpha)^{1/k} \tag{22.6.13}$$

$$= 1 - \left((1-\alpha)^{1/k}\right)^{k} \tag{22.6.14}$$

$$= 1 - (1-\alpha) \tag{22.6.15}$$

$$= \alpha \tag{22.6.16}$$

Thus, if we know that the tests are independent, we can create a smaller multiplier. This will still protect the Type I error rate, while not affecting the Type II error rate as much.

■

Theorem 22.6.3

The original Bonferroni adjustment is a larger divisor (smaller multiplier; lower power) than the adjustment that assumes independence among the tests. In other words,

$$\frac{\alpha}{k} \leq 1 - (1-\alpha)^{1/k} \tag{22.6.17}$$

I leave this proof as an exercise.

Theorem 22.6.1 (far above) shows that we can adjust any case of multiple testing. However, Theorem 22.6.2 and Theorem 22.6.3 (just above) show that additional information (or assumptions) can help make adjustments that affect the test's power less.

In other words, additional information can help. We should not throw anything out.

Note

This section explored the need for adjusting the α to ensure that the claimed Type I error rate is no larger than the true Type I error rate. This is extremely important in inferential statistics. It is not, however, the only thing that needs to be examined.

Power (the probability of rejecting a false null hypothesis) is also quite important. Because of this, while the Bonferroni adjustment always works, it may reduce the power of the test to unacceptable levels. As such, understanding the relationships between and among the various tests is important in creating a superior adjustment.

The Runs Test

This section introduces the runs test (Bradley 1968; Mood 1940; Wald and Wolfowitz 1940). While you will not have seen it in your introductory statistics course, it is based on elementary concepts.

To begin, let us define a couple of terms.

Definition: Run

A run is a sequential set of values all either above or below zero.

Definition: Run Length

The length of a run is the number of elements in that run.

Example 22.6.3

Let the following be residuals from a model.

$$1.2, 3.5, -0.5, 8.9, -1.2, -1.6, -4.5, 0.6, 1.5, -7.9 \tag{22.6.18}$$

Let us calculate the number of runs.

Solution.

The first run is of length 2: $\{1.2, 3.5\}$. The second run is of length 1: $\{-0.5\}$. The third run is also of length 1: $\{8.9\}$. The fourth through sixth runs are $\{-1.2, -1.6, -4.5\}$, $\{0.6, 1.5\}$, and $\{-7.9\}$.

Thus, there are six runs in those 10 residuals.

■

Before continuing, remind yourself of Figure 6.2.1 and Figure 6.2.2. In both graphics, the residual is colored blue if it is positive and pink otherwise. In Figure 6.2.1, there are long unbroken streaks (runs) of blue and pink. There are just three runs.

In Figure 6.2.2, the length of those runs is much reduced and the number is increased. There are now 12 runs in the 20 residuals.

Since the distribution of the residuals is assumed/required to be Normal, the probability of each residual being above the line $e = 0$ is $\pi = 0.500...$ a Binomial random variable! Because we know the distribution of the number of residuals above the $e = 0$ line, we know *everything* about the distribution of residuals, including the distribution of the number of runs. Too few runs indicates that one positive residual will tend to be followed by another. This usually suggests the model is misspecified. In Figure 6.2.1, the number of runs was just three. In the properly specified model of Figure 6.2.2, the number of runs is 12.

A statistician will pay attention to the number of runs as well as *the distribution of the number of runs* ***under the null hypothesis***. In other words, what is the distribution of the number of runs?

The following estimates that distribution. Working through the code may help you better understand the runs test, the test statistic, and its distribution.

```
resids = 20

calculateRuns = function(r) {
  n = length(r)
  x = sign(r)
  y = x[-n]-x[-1]
runs = 1+sum(y!=0)
}

numRuns = numeric()
for(i in 1:1e5) {
  x = rnorm(resids)
  numRuns[i] = calculateRuns(x)
}

par(mar=c(2,1,1,1))
par(yaxt="n")
```

```
par(family="serif")
barplot(table(numRuns), col=rgb(0.85, 0.37, 0.01))
abline(h=0)
```

In the code above, the first line specifies the number of residuals. Since I will be using this distribution to analyze the residuals from Section~???, I set the number to 20. The `calculateRuns` function calculates the number of runs in a sequence of values. The for-loop repeatedly generates values from the null hypothesis and calculates the number of runs.

The last chunk of code plots the histogram, which is shown in Figure **22.6.1**. Note that it is bell-shaped. This should give you an insight into how we can create an approximate test statistic based on the Normal distribution, which is what the functions do.

From this, the probability of having 7 or more runs is 95%. So, we would claim too few runs if we observed 6 or fewer runs in our 20 residuals. The p-value for observing 3 runs is 0.0003. Since this p-value is less than $\alpha = 0.05$, we reject the null hypothesis that the residuals are randomly distributed about the zero line. Thus, we would conclude that the relationship is not linear.

The following lines calculated the critical value and the p-value for the runs distribution.

```
quantile(numRuns, 0.05)     ## critical value
mean(numRuns<=3)            ## p-value
```

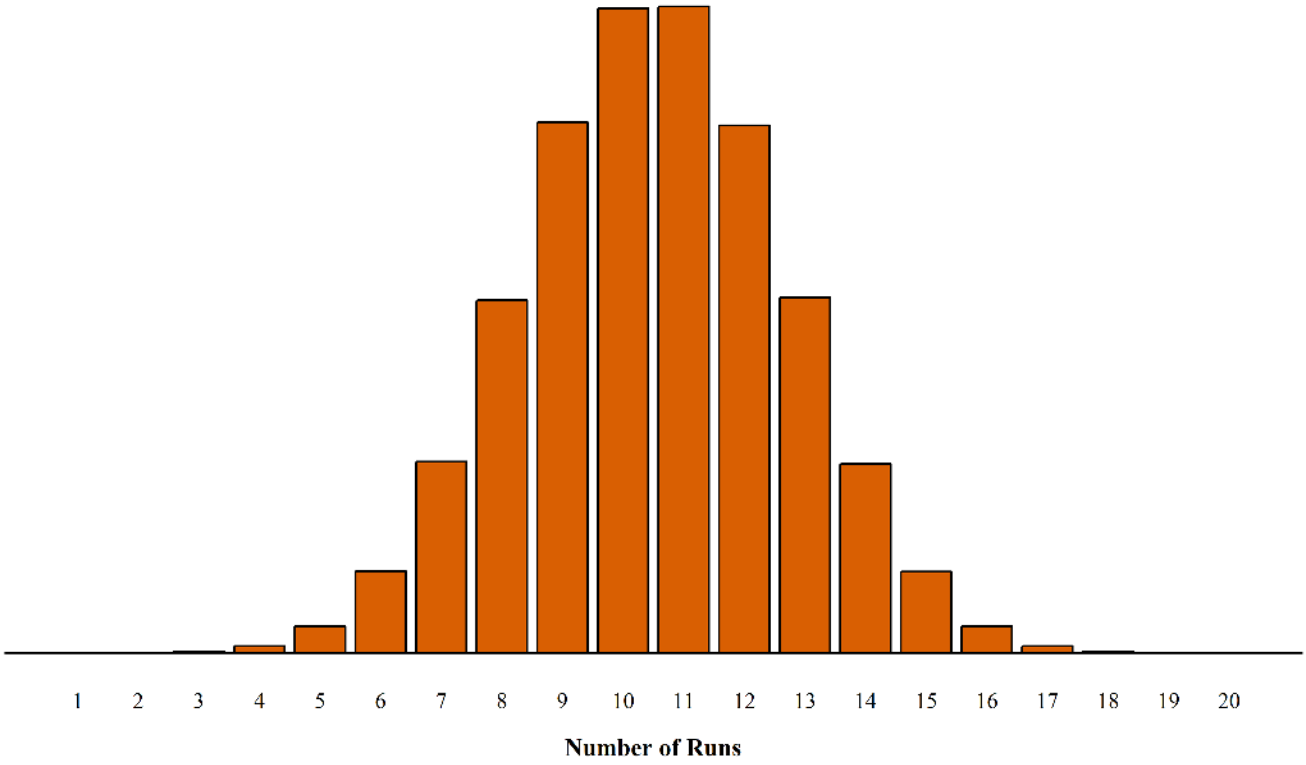


Figure **22.6.1**: Distribution of the number of runs with 20 residuals. This assumes that the individual residuals are independent of each other, which is the null hypothesis of the runs test.

Another Normal Approximation*

In addition to the simulation experiment above, Wald and Wolfowitz (1940) provided a Normal approximation that did not require simulation. They concluded that the number of runs, R, approximately follows the following distribution:

$$R \overset{\cdot}{\sim} N\left(\frac{2N^+N^-}{N}+1;\ \frac{(\mu-1)(\mu-2)}{N-1}\right) \tag{22.6.19}$$

In these, N^+ is the number of positive values, N^- is the number of negative values, and N is the number of values. As with any "Normal approximation," the approximation improves as the sample size increases.

The Exact Distribution*

While Wald and Wolfowitz (1940) did create an approximate distribution for the number of runs, we can also provide the exact distribution that preceded them using the hypergeometric distribution. All that we need is to remember combinatorics and the "choose" (or nCr or binomial coefficient) function.

Under the null hypothesis, the probability of having exactly r runs, given a sample size of N, of which N^+ are positive values and N^- are negative, is

$$P[R=r] = 2\,\frac{\binom{N^+-1}{m-1}\binom{N^--1}{m-1}}{\binom{N}{N^+}} \tag{22.6.20}$$

if r is even and $r=2m$. *However*, if r is odd and $r=2m+1$, then

$$P[R=r] = \frac{\binom{N^+-1}{m}\binom{N^--1}{m-1}+\binom{N^--1}{m}\binom{N^+-1}{m-1}}{\binom{N}{N^+}} \tag{22.6.21}$$

Those look rather "simple" to calculate. Remember, however, that p-values are *cumulative* probabilities. Thus, to calculate p-values, one would need to calculate $P[R \le r]$, which means a lot more calculation.

> Computers are awesome!

The R Functions

Instead of going through the above steps every time to obtain p-values and critical values, the function is implemented in a couple packages in R. First, it is implemented in the `lawstat` package as the function `runs.test`. It takes just one piece of information: the residuals in the order of the independent variable. It is also implemented in the `randtests` and `snpar` packages, as well as `KnoxStats`.

Note that order matters. Thus, you will need to order the residuals based on each independent variable separately. That is, if you have three independent variables, you will need to run the runs test three times, once for each independent variable.

The Central Limit Theorem

The Central Limit Theorem (CLT) is one of the most important theorems in statistics. It explains why we can rely so heavily on the Normal distribution for inferential statistics, even if the data do not arise from a Normal process.

Theorem 22.6.4: The Central Limit Theorem

Let X be a random variable following a distribution with finite variance σ^2. Let $\{X_1, X_2, \ldots, X_n\}$ be a random sample of size n from this distribution. Finally, define T as

$$T_n \overset{\text{def}}{=} \sum_{i=1}^{n} X_i \tag{22.6.22}$$

As $n \to \infty$, the distribution of T_n converges to the Normal distribution.

Note that this theorem requires that the random variable has a finite variance *and* that the sample is a random sample (independent and identically distributed). If either of these two conditions is not met, then this version of the Central Limit Theorem does not

hold.

Other versions exist for some different types of dependent sampling. However, even those require the variance be finite.

In your previous statistics course, you may remember the Central Limit Theorem as saying something about the distribution of the sample mean. Well, that is actually a consequence of the CLT.

Corollary 22.6.5: Distribution of Sample Mean

Given the conditions of the Central Limit Theorem, define $\overline{X}$ as

$$\overline{X}_n \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} X_i \tag{22.6.23}$$

The distribution of the sample mean, $\overline{X}_n$, converges to the Normal distribution.

To use the Central Limit Theorem, we need to ensure the data are **randomly generated** from a distribution with a **finite variance**. We also need to ensure that we are trying to describe the sample total or sample mean. If those conditions are met, then we can use the CLT.

How would we *prove* the Central Limit Theorem? To do that, we need to introduce something called "moment generating functions." This we do in the next section. Continue, if you wish. Proving the Central Limit Theorem is usually beyond the scope of this type of course.

Moment Generating Functions*

This section introduces moment generating functions. You most certainly did not experience these in your introductory statistics course. For this course, they are used solely to prove the Central Limit Theorem and to prove that a linear combination of Normally distributed random variables is also a Normally distributed random variable (Theorem: The Sum of Normals).

A moment generating function is a function that can be used to generate all of the **moments** of a distribution. We have already experienced moments. The expected value is the first moment, for instance. Here are the definitions of three types of moments: raw, central, and standardized:

Definition: Moment

The kth (raw) moment is $E[X^k]$.

Definition: Central Moment

The kth *central* moment is $E[(X-\mu)^k]$, where $\mu = E[X]$.

Definition: Standardized Moment

The kth *standardized* moment is $E\left[\left(\frac{X-\mu}{\sigma}\right)^k\right]$, where $\mu = E[X]$ and $\sigma = V[X]$.

Thus, by definition, the first raw moment is $E[X]$. This is just the **mean**. The second central moment is $E[(X-\mu)^2]$. This is just the **variance**. Similarly, the third standardized moment is the **skew**, and the fourth standardized moment is the **kurtosis**.

Prove to yourself that the first central moment is always 0 as long as $E[X]$ exists.

$$\cdot\cdot-\cdot\ast\cdot-\cdot\cdot$$

Moments are interesting in their own right. However, the following theorem makes them important in the realm of probability theory.

Theorem 22.6.6

If two distributions have all moments the same, then the two distributions are equal, with probability 1.

The "with probability 1" clause is added to take into consideration the inherent issues with continuous distributions. [It also clearly specifies the difference between probability (and statistics) and mathematics that some mathematicians may not appreciate.]

Here is an example of that inherent issue: Let us define two distributions X and Y:

$$f(x) = \begin{cases} 1 & 0 < x < 1 \\ 0 & \text{otherwise} \end{cases} \tag{22.6.24}$$

$$g(y) = \begin{cases} 1 & 0 \le y \le 1 \\ 0 & \text{otherwise} \end{cases} \tag{22.6.25}$$

The random variables X and Y are *not* identical. The support for Y includes the entire support of X as well as the values 0 and 1. Thus, they are mathematically not the same distribution.

However, they are the same *with probability 1*. In other words, it does not matter which of the distributions you use, the calculated probabilities will be the same. This is because $P[X = 0] = P[Y = 0] = 0$ and $P[X = 1] = P[Y = 1] = 0$. Thus, for all intents and purposes, X and Y are the same. In probability, we would say $X = Y$ with probability 1, symbolized as $X \overset{wp0}{=\!=} Y$.

With Theorem 22.6.6, it becomes clear that if two distributions have the same moment generating function, then those distributions are the same (with probability 1).

And so, at this point, we know what a moment is and what we can use them for. We know that if two distributions have all the same moments, then the two distributions are identical (with probability 1). This led to us knowing that if two distributions have the same moment generating functions, then the two distributions are the same. All that remains is to define a moment generating function (MGF).

Note

There is a little more to the theory, but the details are not important at this point in your statistical life.

Definition: Moment Generating Function

Let X be a random variable. Define the function $M_x(t)$ as

$$M_x(t) \overset{\text{def}}{=} E\left[e^{Xt}\right] \tag{22.6.26}$$

Then $M_x(t)$ is called the moment generating function of X.

Why is it called a "**moment generating function**"? It generates the *raw* moments. The kth *raw* moment of X is calculated from

$$E[X^k] = \frac{d^k}{dt^k} M_x(t)\Big|_{t=0} \tag{22.6.27}$$

Note that the moment generating function must be defined in a small neighborhood of 0 for us to apply this formula.

Example 22.6.4

Let X be a Bernoulli random variable with success probability π. What is its moment generating function?

Before starting the solution, please review the information on the Bernoulli distribution.

Solution.

The moment generating function is defined as $M_x(t) = E\left[e^{Xt}\right]$. Since X is a discrete random variable, the MGF is a sum weighted by its probability mass function (pmf).

$$E\left[e^{Xt}\right] = \sum_i e^{xt}\, P[X = x_i] \tag{22.6.28}$$

$$= \sum_{x=0}^{1} e^{xt}\, \pi^x (1-\pi)^{1-x} \tag{22.6.29}$$

$$= e^{0t}\, \pi^0 (1-\pi)^{1-0} + e^{1t}\, \pi^1 (1-\pi)^{1-1} \tag{22.6.30}$$

$$= 1 \cdot 1 \cdot (1-\pi) + e^t \cdot \pi^1 \cdot 1 \tag{22.6.31}$$

And, simplification gives us the MGF for a Bernoulli random variable:

$$E[e^{Xt}] = (1-\pi) + \pi\, e^t \tag{22.6.32}$$

Now, with that, we can calculate the mean of X:

$$E[X^1] = \frac{d^1}{dt^1} M_x(t)\Big|_{t=0} \tag{22.6.33}$$

$$= \frac{d}{dt}\left((1-\pi) + \pi\, e^t\right)\Big|_{t=0} \tag{22.6.34}$$

$$= \pi\, e^t\,|_{t=0} \tag{22.6.35}$$

$$= \pi\, e^0 \tag{22.6.36}$$

$$= \pi \tag{22.6.37}$$

It should come as no surprise that the expected value of a Bernoulli random variable is the success probability. The second moment is

$$E[X^2] = \frac{d^2}{dt^2} M_x(t)\Big|_{t=0} \tag{22.6.38}$$

$$= \frac{d^2}{dt^2}\left((1-\pi) + \pi\, e^t\right)\Big|_{t=0} \tag{22.6.39}$$

$$= \frac{d}{dt}\left(\pi\, e^t\right)\Big|_{t=0} \tag{22.6.40}$$

$$= \pi\, e^t\,|_{t=0} \tag{22.6.41}$$

$$= \pi \tag{22.6.42}$$

This is the second raw moment. Recall that $E[X^2] = \sigma^2 + \mu^2$. Thus, we can calculate the variance of X as

$$V[X] = E[X^2] - \mu^2 \tag{22.6.43}$$

$$= \pi - \pi^2 \tag{22.6.44}$$

$$= \pi(1-\pi) \tag{22.6.45}$$

Thus, we have obtained our usual variance formula for a Bernoulli random variable... but in a different way.

■

Example 22.6.5

Determine the moment generating function for a Binomial random variable with parameters n and π.

Solution.

By our definition of the Binomial random variable, we know that it is just the sum of n independent and identically distributed Bernoulli random variables. Let us use that to make the calculations easier.

Thus, let X_i be a series of Bernoulli random variables and $Y = \sum_{i=1}^{n} X_i$ be the resulting Binomial random variable. The moment generating function of Y is

$$M_y(t) = E\left[\exp\left(Yt\right)\right] \tag{22.6.46}$$
$$= E\left[\exp\left[\sum_{i=1}^{n} X_i\ t\right]\right] \tag{22.6.47}$$
$$= E\left[\prod_{i=1}^{n} \exp\left[X_i\ t\right]\right] \tag{22.6.48}$$
$$= \prod_{i=1}^{n} E\left[\exp\left[X_i\ t\right]\right] \tag{22.6.49}$$
$$= \prod_{i=1}^{n} \left((1-\pi) + \pi e^t\right) \tag{22.6.50}$$

Thus, the MGF for a Binomial random variable is

$$M_y(t) = \left((1-\pi) + \pi e^t\right)^n \tag{22.6.51}$$

With that, we can calculate the various moments, should we desire.

■

Theorem 22.6.7

Let X and Y be two *independent* random variables. Define $Z = X + Y$. The moment generating function of Z is the product of the moment generating functions of X and Y:

$$M_z(t) = M_x(t)\ M_y(t) \tag{22.6.52}$$

Proof.
The proof is simply an exercise in definitions and algebra:

$$M_z(t) \stackrel{\text{def}}{=} E[e^{tZ}] \tag{22.6.53}$$
$$= E[e^{(X+Y)t}] \tag{22.6.54}$$
$$= E[e^X\ e^Y] \tag{22.6.55}$$
$$= E[e^X]\ E[e^Y] \tag{22.6.56}$$
$$= M_x(t)\ M_y(t) \tag{22.6.57}$$

Note that we needed the assumption of independence in the fourth line (22.6.56), where we split the expected value.

■

Theorem 22.6.8: MGF of a Normal

The moment generating function for the Normal distribution is

$$M_x(t) = \exp\left[\mu t + \frac{1}{2}\sigma^2 t^2\right] \tag{22.6.58}$$

I leave this proof as an exercise. It is a simple application of the definition of moment generating function and the probability density function of the Normal distribution.

With that last theorem, we are able to prove that a linear combination of independent Normal random variables also is a Normal random variable.

Theorem 22.6.9: Sum of Normals

Let $X \sim N\left(\mu_x,\ \sigma_x^2\right)$ and $Y \sim N\left(\mu_y,\ \sigma_y^2\right)$. Let X and Y be independent; that is, $X \perp Y$. If a, b, and c be scalars, then

$$aX + bY + c \sim N\left(a\mu_x + b\mu_y + c,\ a^2\sigma_x^2 + b^2\sigma_y^2\right) \tag{22.6.59}$$

Proof.

As expected, the proof relies on moment generating functions. First, the moment generating function of $aX + bY + c$ is the product of the MGFs of aX, bY, and c because they are independent. So, let us examine the MGFs of each.

$$\begin{aligned} M_{ax}(t) &= M_x(at) && (22.6.60)\\ &= \exp\left[\mu_x(at) + \frac{1}{2}\sigma_x^2(at)^2\right] && (22.6.61)\\ M_{by}(t) &= M_y(bt) && (22.6.62)\\ &= \exp\left[\mu_y(bt) + \frac{1}{2}\sigma_y^2(bt)^2\right] && (22.6.63)\\ M_c(t) &= M(ct) && (22.6.64)\\ &= \exp[ct] && (22.6.65) \end{aligned}$$

Putting these together gives

$$\begin{aligned} M_{ax+by+c}(t) &= \exp\left[\mu_x(at) + \frac{1}{2}\sigma_x^2(at)^2\right] \cdot \exp\left[\mu_y(bt) + \frac{1}{2}\sigma_y^2(bt)^2\right] \cdot \exp[ct] && (22.6.66)\\ &= \exp\left[\mu_x(at) + \frac{1}{2}\sigma_x^2(at)^2 + \mu_y(bt) + \frac{1}{2}\sigma_y^2(bt)^2 + ct\right] && (22.6.67)\\ &= \exp\left[(a\mu_x + b\mu_y + c)\,t + \frac{1}{2}\left(a^2\sigma_x^2 + b^2\sigma_y^2\right)t^2\right] && (22.6.68) \end{aligned}$$

Note that this is the moment generating function of a Normal distribution with mean $a\mu_x + b\mu_y + c$ and variance $a^2\sigma_x^2 + b^2\sigma_y^2$. Thus, we have proven this theorem.

■

Note on the Degenerate Distribution

We are not quite done; there is one missing piece.

How can we have a moment generating function for the scalar value c? Aren't MGFs only for random variables? Well, there is a distribution called the "**degenerate distribution**" that takes on a single value with probability 1. Think of a two-headed coin. The probability mass function for a degenerate distribution is $\delta(X - c)$, where $\delta(\cdot)$ is the Dirac delta function (Calculus I), X is the random variable, and c is the only possible value of X.

If you have taken a course in differential equations, you will certainly have come across the Dirac delta function. If not, then you may not have. In my experience, I saw $\delta(\cdot)$ in Calculus I... and II and III and a few physics courses.

The take-away is that there is a lot of statistics that you have not even seen. You have just caught little glimpses. Sometimes, you have seen probability without actually noticing it. For instance, the probability density functions you have seen are just solutions to certain differential equations.

Astounding!

Note

Alright, I'm not done with that. If you do not like thinking about the degenerate distribution as an actual distribution, think of it as a *limiting* distribution. For instance, define

$$f_s(x;\ \mu) \stackrel{\text{def}}{=} \frac{1}{\sqrt{2\pi s^2}}\ \exp\left[-\frac{1}{2}\ \frac{(x-\mu)^2}{s^2}\right] \qquad (22.6.69)$$

The limit of this (Normal) distribution as $s \to 0$ is the degenerate distribution at the point $x = \mu$.

Neat-o skeet-o!

Note that the previous distribution showed that the sum of two Normally-distributed random variables also has a Normal distribution. The converse, that a Normally-distributed random variable can only be decomposed into the sum of two other Normally-distributed random variables is much more difficult to prove. It is the purpose of Cramér's Theorem.

In other words, Cramér's Theorem states that if $Z = X + Y$ and if Z follows a Normal distribution (and a few other requirements), then both X and Y must also follow Normal distributions. The proof can be found in Cramér (1936).

A Proof of the Central Limit Theorem*

And so, with all of this preparation, we are now ready to prove the Central Limit Theorem.

Theorem 22.6.10: The Central Limit Theorem

Let X be a random variable with expected value μ and finite variance σ^2. Let $\{X_1, X_2, \ldots, X_n\}$ be a *random sample* of size n from this distribution. *Finally*, define T_n as

$$T_n \stackrel{\text{def}}{=} \sum_{i=1}^{n} X_i \tag{22.6.70}$$

Then,

$$T_n \stackrel{d}{\to} N\left(n\mu,\ n\sigma^2\right) \tag{22.6.71}$$

as $n \to \infty$.

Here is an **overview of this proof**: This proof of the Central Limit Theorem relies on moment generating functions and Taylor's theorem. It starts with the distribution of T, which we need to determine. It then creates two subsequent distributions, Y_n and Z_i, to help with the calculations. The moment generating function for Z_i is approximated using Taylor's theorem. From that, the moment generating function for Y_n is determined. That provides us the approximate distribution of Y_n, which is $N(0,\ 1)$. With that, we determine the approximate distribution of T, as required.

With that overview, let us start the actual proof.

Proof.
Let X be a random variable with mean μ and finite variance σ^2. Let $\{X_1, X_2, \ldots, X_n\}$ be a random sample from X. The sum $T_n \stackrel{\text{def}}{=} \sum X_i$ has mean $n\mu$ and variance $n\sigma^2$.

Define the random variable Y_n as

$$Y_n \stackrel{\text{def}}{=} \frac{T_n - n\mu}{\sqrt{n\sigma^2}} \tag{22.6.72}$$

In other words, the Y_n random variable is just the T_n random variable *standardized*.

Note that this definition is equivalent to

$$Y_n = \sum_{i=1}^{n} \frac{X_i - \mu}{\sqrt{n\sigma^2}} \tag{22.6.73}$$

Eventually, we will show that the distribution of Y_n converges to standard Normal as $n \to \infty$. For now, we use it as an intermediate between T and Z_i. Define Z_i as

$$Z_i \stackrel{\text{def}}{=} \frac{X_i - \mu}{\sigma} \tag{22.6.74}$$

which is the z-score for the X_i random variable. It is useful, because we can now see

$$Y_n = \sum_{i=1}^{n} \frac{1}{\sqrt{n}} Z_i \tag{22.6.75}$$

This link between Z_i and Y_n will be exploited later, when we finally approximate the distribution of Z_i and use it to approximate the distribution of Y_n.

With that literary foreshadowing, note that the moment generating function of Y_n, in terms of the Z_i, is

$$M_y(t) = \left(M_z\left(\frac{t}{\sqrt{n}}\right) \right)^n \tag{22.6.76}$$

This result relies on the Z_i being *independent and identically distributed* (i.e., a random sample). The moment generating function for Z_1 (or any of the Z_i) can be approximated by using Taylor's theorem (expanded around $t = 0$):

$$M_z\left(\frac{t}{\sqrt{n}}\right) = \sum_{i=0}^{\infty} \left(\frac{t}{\sqrt{n}}\right)^i \frac{M_Z^{(i)}(0)}{i!} \tag{22.6.77}$$

$$= \left(\frac{t}{\sqrt{n}}\right)^0 \frac{M_Z^{(0)}(0)}{0!} + \left(\frac{t}{\sqrt{n}}\right)^1 \frac{M_Z^{(1)}(0)}{1!} + \left(\frac{t}{\sqrt{n}}\right)^2 \frac{M_Z^{(2)}(0)}{2!} + \cdots \tag{22.6.78}$$

$$= 1\frac{1}{0!} + \frac{t}{\sqrt{n}}\frac{E[Z_i]}{1!} + \left(\frac{t}{\sqrt{n}}\right)^2 \frac{E[Z^2]}{2!} + \cdots \tag{22.6.79}$$

$$= 1 + \frac{t}{\sqrt{n}}\frac{0}{1!} + \left(\frac{t}{\sqrt{n}}\right)^2 \frac{1}{2!} + \cdots \tag{22.6.80}$$

$$= 1 + 0 + \frac{t^2}{2n} + \cdots \tag{22.6.81}$$

If we include the remainder term from Taylor (the $\cdots$ above), then we have this conclusion

$$M_z\left(\frac{t}{\sqrt{n}}\right) \approx 1 + \frac{t^2}{2n} + \xi\frac{t^3}{6n^{3/2}} + o\left(\frac{t^3}{n^{3/2}}\right) \tag{22.6.82}$$

The last two terms are the remainder. Here, ξ is a constant and $o\left(\frac{t^3}{n^{3/2}}\right)$ is "little-o" notation indicating that the remainder (whatever it is) goes to 0 faster than does $\frac{t^3}{n^{3/2}}$ (as $t \to 0$).

Now, we return to Y_n. It was defined as $\sum_{i=1}^n \frac{1}{\sqrt{n}} Z_i$, where the Z_i were independent and identically distributed. This means the moment generating function of Y_n is

$$M_y\left(\frac{t}{\sqrt{n}}\right) \approx \left(1 - \frac{t^2}{2n} + \xi\frac{t^3}{6n^{3/2}} + o\left(\frac{t^3}{n^{3/2}}\right)\right)^n \tag{22.6.83}$$

The third term goes to 0 as $n \to \infty$, as does the fourth term. The remaining two terms are

$$M_y\left(\frac{t}{\sqrt{n}}\right) \approx \left(1 - \frac{t^2/2}{n}\right)^n; \qquad n \to \infty \tag{22.6.84}$$

Note that this is just the definition of the exponential function $e^{\frac{1}{2}t^2}$. This means we have the MGF for Y_n:

$$M_y\left(\frac{t}{\sqrt{n}}\right) \approx e^{\frac{1}{2}t^2} \tag{22.6.85}$$

\ldots\ and this is just the moment generating function for a standard Normal distribution, $N(0,\ 1)$. Thus, Y_n will approach $N(0,\ 1)$ as $n \to \infty$.

Finally, using our definition of Y_n (Eqn 22.6.73), we can conclude that the distribution of T_n also converges in distribution to $N(\mu,\ \sigma^2)$ as $n \to \infty$.

■

And so, in this final section of the appendix, we were finally able to prove the Central Limit Theorem. To do so, we needed to learn about moment generating functions... and remember our Taylor Series expansion.

Again, the only purpose of moment generating functions for this course of study is to be able to prove the Central Limit Theorem. I included the proof of the Central Limit Theorem simply because many have asked for it in the past.

Et voilà !

This page titled 22.6: Other Topics is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

22.7: End-of-Appendix Materials

R Functions

In this appendix, we were introduced to several R functions that will be useful. These are listed here.

Packages

- `KnoxStats`
  This is a "book package" with several supporting functions to make learning statistics a bit easier. See here for how to install it.



Statistics

- `runs.test(r, order=x)`
  The runs test tests if the residuals `r`, as ordered by `x`, are sufficiently distributed around the zero line to suggest that they are independent. This is the version in the `KnoxStats` package.

Exercises

1. Prove the Mean Deviance Lemma.
2. Prove Lemma 22.2.5.
3. Prove Lemma 22.2.6.
4. Prove Lemma 22.2.7.
5. Prove Lemma 22.2.8.
6. Prove Lemma 22.2.9.
7. Prove Lemma 22.2.10.
8. Prove Lemma 22.2.11. Note that this proof usually requires the Cauchy-Schwarz Inequality.
9. Prove that the Cauchy distribution is equivalent to the Student's t distribution with 1 degree of freedom.
10. Prove that the Normal distribution is equivalent to the Student's t distribution as $\nu \to \infty$.
11. Prove that the interquartile range of the standard Cauchy is 2.
12. Prove $E[S^2] = \sigma^2$ when $Y \sim N(\mu, \sigma^2)$.
13. Prove that the first central moment is always 0, as long as $E[X]$ exists.
14. Prove that the third central moment (skew) for the Normal distribution is zero.
15. Use moment generating functions to determine when the Binomial distribution has zero skew.
16. Calculate the moment generating function for the Poisson distribution. Check that it generates the first two moments. Use it to determine the variance of a Poisson random variable.
17. Prove Theorem: MGF of a Normal.
18. Use the moment generating function of the Bernoulli distribution to determine the moment generating function of the degenerate distribution.
19. Use moment generating functions to calculate the variance of a degenerate distribution.

Theory Readings

- Carl E. Bonferroni (1936). "*Teoria statistica delle classi e calcolo delle probabilità*." *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commerciali di Firenze* (in Italian). 8: 3–62.
- James V. Bradley (1968). *Distribution-Free Statistical Tests*, Chapter 12. Prentice-Hall.
- Harald Cramér (1936). "*Über eine Eigenschaft der Normalen Verteilungsfunktion*." *Mathematische Zeitschrift* (in German). 41(**1**): 405–414. doi:10.1007/BF01180430.
- Olive Jean Dunn (1958). "Estimation of the Means for Dependent Variables." *Annals of Mathematical Statistics*. 29(**4**): 1095–1111.

doi:10.1214/aoms/1177706374.

- Olive Jean Dunn (1961). "Multiple Comparisons Among Means." *Journal of the American Statistical Association*. 56(**293**): 52–64. doi:10.1080/01621459.1961.10482090.
- David K. Hildebrand (1986). *Statistical Thinking for Behavioral Scientists*. Duxbury Press.
- Alexander M. Mood (1940). "The Distribution Theory of Runs." *Annals of Mathematical Statistics*. 11(**4**): 367–392. https://www.jstor.org/stable/pdf/2235718.pdf.
- George W. Snedecor. (1934). "Calculation and Interpretation of Analysis of Variance and Covariance." *Agronomy Journal* 26(3): 255–56. https://doi.org/10.2134/agronj1934.0...2002600030019x.
- Abraham Wald and Jacob Wolfowitz (1940). "On a test whether two samples are from the same population." *Annals of Mathematical Statistics*. 11(2): 147–162. doi: 10.1214/aoms/1177731909.

This page titled 22.7: End-of-Appendix Materials is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Ole Forsberg.

Index

D

dire

Glossary

Sample Word 1 | Sample Definition 1

Detailed Licensing

Overview

Title: Linear Models and Rurità Kràlovstvì

Webpages: 159

Applicable Restrictions: Noncommercial

All licenses found:

- CC BY-NC-SA 4.0: 96.9% (154 pages)
- Undeclared: 3.1% (5 pages)

By Page

- Linear Models and Rurità Kràlovstvì - *CC BY-NC-SA 4.0*
  - Front Matter - *CC BY-NC-SA 4.0*
    - TitlePage - *CC BY-NC-SA 4.0*
    - InfoPage - *CC BY-NC-SA 4.0*
    - Table of Contents - *Undeclared*
    - Licensing - *Undeclared*
    - Preface - *CC BY-NC-SA 4.0*
  - 1: An Introduction to Ruruitania - *CC BY-NC-SA 4.0*
    - 1.1: The Background of Ruritania - *CC BY-NC-SA 4.0*
    - 1.2: The Geography - *CC BY-NC-SA 4.0*
    - 1.3: The Government - *CC BY-NC-SA 4.0*
    - 1.4: The Economy - *CC BY-NC-SA 4.0*
    - 1.5: US-Ruritanian Relations - *CC BY-NC-SA 4.0*
    - 1.6: Illustrations of Analyses - *CC BY-NC-SA 4.0*
      - 1.6.1: Illustration- The Missing Kraj - *CC BY-NC-SA 4.0*
      - 1.6.2: Illustration- Cows in Děčín - *CC BY-NC-SA 4.0*
      - 1.6.3: Illustration- Ruritanian Crops - *CC BY-NC-SA 4.0*
      - 1.6.4: Illustration- Wealth in Ruritania - *CC BY-NC-SA 4.0*
      - 1.6.5: Illustration- Elections in Ruritania - *CC BY-NC-SA 4.0*
      - 1.6.6: Illustration- Insurance in Ruritania - *CC BY-NC-SA 4.0*
      - 1.6.7: Illustration- Warmth for the King - *CC BY-NC-SA 4.0*
    - 1.7: Conclusion - *CC BY-NC-SA 4.0*
  - 2: Part One- Least Squares - *CC BY-NC-SA 4.0*
  - 3: Intro to Linear Regression - *CC BY-NC-SA 4.0*
    - 3.1: Scalar Representation - *CC BY-NC-SA 4.0*
    - 3.2: First Results - *CC BY-NC-SA 4.0*
    - 3.3: Our First Assumptions - *CC BY-NC-SA 4.0*
    - 3.4: PRE Measures - *CC BY-NC-SA 4.0*
    - 3.5: Conclusion - *CC BY-NC-SA 4.0*
    - 3.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 4: Matrices and Linear Regression - *CC BY-NC-SA 4.0*
    - 4.1: Matrix Representation - *CC BY-NC-SA 4.0*
    - 4.2: Predictions and the Hat Matrix - *CC BY-NC-SA 4.0*
    - 4.3: The Geometry of OLS - *CC BY-NC-SA 4.0*
    - 4.4: Multicollinearity and Categorical Independent Variables - *CC BY-NC-SA 4.0*
    - 4.5: Conclusion - *CC BY-NC-SA 4.0*
    - 4.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 5: Improved! Now with Probabilities - *CC BY-NC-SA 4.0*
    - 5.1: Probability Distributions - *CC BY-NC-SA 4.0*
    - 5.2: Test Statistics and Hypothesis Testing - *CC BY-NC-SA 4.0*
    - 5.3: Confidence Intervals - *CC BY-NC-SA 4.0*
    - 5.4: The Working-Hotelling Bands - *CC BY-NC-SA 4.0*
    - 5.5: Conclusion - *CC BY-NC-SA 4.0*
    - 5.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 6: Dood! Check the Requirements - *CC BY-NC-SA 4.0*
    - 6.1: Normality - *CC BY-NC-SA 4.0*
    - 6.2: Functional Fit - *CC BY-NC-SA 4.0*
    - 6.3: Homoskedasticity - *CC BY-NC-SA 4.0*
    - 6.4: Multicollinearity - *CC BY-NC-SA 4.0*
    - 6.5: Conclusion - *CC BY-NC-SA 4.0*
    - 6.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 7: Time for Some Examples - *CC BY-NC-SA 4.0*
    - 7.1: Violent Crime - *CC BY-NC-SA 4.0*
    - 7.2: Violent Crime, Wealth, Region - *CC BY-NC-SA 4.0*
    - 7.3: Cows in Děčín - *CC BY-NC-SA 4.0*
    - 7.4: Conclusion - *CC BY-NC-SA 4.0*
    - 7.5: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 8: Fixing the Violations - *CC BY-NC-SA 4.0*

- 8.1: The Issue of Boundedness - *CC BY-NC-SA 4.0*
- 8.2: The South Sudanese Referendum - *CC BY-NC-SA 4.0*
- 8.3: Adjustments for Heteroskedasticity - *CC BY-NC-SA 4.0*
- 8.4: Conclusion - *CC BY-NC-SA 4.0*
- 8.5: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

9: Beyond the Ordinary - *CC BY-NC-SA 4.0*

10: Other Least Squares - *CC BY-NC-SA 4.0*

- 10.1: Ordinary Least Squares - *CC BY-NC-SA 4.0*
- 10.2: Weighted Least Squares - *CC BY-NC-SA 4.0*
- 10.3: Generalized Least Squares - *CC BY-NC-SA 4.0*
- 10.4: May the (Strong) Force be with You - *CC BY-NC-SA 4.0*
- 10.5: Elections in Ruritania - *CC BY-NC-SA 4.0*
- 10.6: Conclusion - *CC BY-NC-SA 4.0*
- 10.7: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

11: Quantile Regression - *CC BY-NC-SA 4.0*

- 11.1: Parameter Estimation - *CC BY-NC-SA 4.0*
- 11.2: Quantile Regression - *CC BY-NC-SA 4.0*
- 11.3: Conclusion - *CC BY-NC-SA 4.0*
- 11.4: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

12: Maximizing the Likelihood - *CC BY-NC-SA 4.0*

- 12.1: The Likelihood - *CC BY-NC-SA 4.0*
- 12.2: The MLE and the CLM - *CC BY-NC-SA 4.0*
- 12.3: Conclusion - *CC BY-NC-SA 4.0*
- 12.4: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

13: Beyond the Classical Model - *CC BY-NC-SA 4.0*

14: Generalized Linear Models - *CC BY-NC-SA 4.0*

- 14.1: The CLM and the GLM - *CC BY-NC-SA 4.0*
- 14.2: The Requirements for GLMs - *CC BY-NC-SA 4.0*
- 14.3: Assumptions of GLMs - *CC BY-NC-SA 4.0*
- 14.4: The Gaussian Distribution - *CC BY-NC-SA 4.0*
- 14.5: Generalized Linear Models in R - *CC BY-NC-SA 4.0*
- 14.6: Conclusion - *CC BY-NC-SA 4.0*
- 14.7: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

15: Binary Dependent Variables - *CC BY-NC-SA 4.0*

- 15.1: Binary Dependent Variables - *CC BY-NC-SA 4.0*
- 15.2: Latent Variable Modeling - *CC BY-NC-SA 4.0*
- 15.3: The Mathematics - *CC BY-NC-SA 4.0*
- 15.4: Modeling with the Logit - *CC BY-NC-SA 4.0*
- 15.5: Measuring Accuracy - *CC BY-NC-SA 4.0*
- 15.6: Modeling with Other Links - *CC BY-NC-SA 4.0*
- 15.7: Model Selection - *CC BY-NC-SA 4.0*
- 15.8: Conclusion - *CC BY-NC-SA 4.0*
- 15.9: End-of-Chapter Material - *CC BY-NC-SA 4.0*

16: Binomial Dependent Variables - *CC BY-NC-SA 4.0*

- 16.1: Binomial Distribution - *CC BY-NC-SA 4.0*
- 16.2: The Mathematics - *CC BY-NC-SA 4.0*
- 16.3: O Canada! - *CC BY-NC-SA 4.0*
- 16.4: Sri Lanka in 2010 - *CC BY-NC-SA 4.0*
- 16.5: Beta-Binomial Regression - *CC BY-NC-SA 4.0*
- 16.6: Conclusion - *CC BY-NC-SA 4.0*
- 16.7: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

17: Count Dependent Variables - *CC BY-NC-SA 4.0*

- 17.1: Linear or Poisson Regression? - *CC BY-NC-SA 4.0*
- 17.2: The Mathematics - *CC BY-NC-SA 4.0*
- 17.3: Overdispersion - *CC BY-NC-SA 4.0*
- 17.4: Body Counts - *CC BY-NC-SA 4.0*
- 17.5: The Bias-Variance Tradeoff - *CC BY-NC-SA 4.0*
- 17.6: Conclusion - *CC BY-NC-SA 4.0*
- 17.7: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

18: Nominal and Ordinal Dependent Variables - *CC BY-NC-SA 4.0*

- 18.1: Nominal Dependent Variables - *CC BY-NC-SA 4.0*
- 18.2: Ordinal Dependent Variables - *CC BY-NC-SA 4.0*
- 18.3: Cattle Feed - *CC BY-NC-SA 4.0*
- 18.4: The State University of Ruritania - *CC BY-NC-SA 4.0*
- 18.5: Conclusion - *CC BY-NC-SA 4.0*
- 18.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

19: Appendices - *CC BY-NC-SA 4.0*

20: The Appendix of Matrices - *CC BY-NC-SA 4.0*

- 20.1: Matrix Basics - *CC BY-NC-SA 4.0*
- 20.2: Addition - *CC BY-NC-SA 4.0*
- 20.3: Multiplication - *CC BY-NC-SA 4.0*
- 20.4: Other Terms and Operations - *CC BY-NC-SA 4.0*
- 20.5: Consequences - *CC BY-NC-SA 4.0*
- 20.6: Statistics in Matrices - *CC BY-NC-SA 4.0*
- 20.7: End-of-Appendix Materials - *CC BY-NC-SA 4.0*

21: The Appendix of R - *CC BY-NC-SA 4.0*

- 21.1: Installing R - *CC BY-NC-SA 4.0*
- 21.2: R Packages - *CC BY-NC-SA 4.0*
- 21.3: R Functions - *CC BY-NC-SA 4.0*
- 21.4: Programming Practice - *CC BY-NC-SA 4.0*
  - 21.4.1: Plotting a Distribution - *CC BY-NC-SA 4.0*
  - 21.4.2: Functions of RVs - *CC BY-NC-SA 4.0*
  - 21.4.3: Effects of Rounding - *CC BY-NC-SA 4.0*
  - 21.4.4: Algorithmic Thinking - *CC BY-NC-SA 4.0*

        - 21.4.5: Distribution of Extreme Values - *CC BY-NC-SA 4.0*
        - 21.4.6: Non-Parametric Bootstrap - *CC BY-NC-SA 4.0*
        - 21.4.7: Distribution of Sample Means - *Undeclared*
        - 21.4.8: Distribution of Sample Sums - *Undeclared*
    - 22: The Appendix of Statistics - *CC BY-NC-SA 4.0*
        - 22.1: Importantly Confusing Points - *CC BY-NC-SA 4.0*
        - 22.2: Sample Statistics - *CC BY-NC-SA 4.0*
        - 22.3: Population Parameters - *CC BY-NC-SA 4.0*
        - 22.4: Probability Distributions - *CC BY-NC-SA 4.0*
        - 22.5: Distributions of Sample Statistics - *CC BY-NC-SA 4.0*
        - 22.6: Other Topics - *CC BY-NC-SA 4.0*
        - 22.7: End-of-Appendix Materials - *CC BY-NC-SA 4.0*
    - Back Matter - *CC BY-NC-SA 4.0*
        - Index - *CC BY-NC-SA 4.0*
        - Glossary - *CC BY-NC-SA 4.0*
        - Detailed Licensing - *CC BY-NC-SA 4.0*
        - Detailed Licensing - *Undeclared*

Detailed Licensing

Overview

Title: Linear Models and Rurità Kràlovstvì

Webpages: 159

Applicable Restrictions: Noncommercial

All licenses found:

- CC BY-NC-SA 4.0: 96.9% (154 pages)
- Undeclared: 3.1% (5 pages)

By Page

- Linear Models and Rurità Kràlovstvì - *CC BY-NC-SA 4.0*
  - Front Matter - *CC BY-NC-SA 4.0*
    - TitlePage - *CC BY-NC-SA 4.0*
    - InfoPage - *CC BY-NC-SA 4.0*
    - Table of Contents - *Undeclared*
    - Licensing - *Undeclared*
    - Preface - *CC BY-NC-SA 4.0*
  - 1: An Introduction to Ruruitania - *CC BY-NC-SA 4.0*
    - 1.1: The Background of Ruritania - *CC BY-NC-SA 4.0*
    - 1.2: The Geography - *CC BY-NC-SA 4.0*
    - 1.3: The Government - *CC BY-NC-SA 4.0*
    - 1.4: The Economy - *CC BY-NC-SA 4.0*
    - 1.5: US-Ruritanian Relations - *CC BY-NC-SA 4.0*
    - 1.6: Illustrations of Analyses - *CC BY-NC-SA 4.0*
      - 1.6.1: Illustration- The Missing Kraj - *CC BY-NC-SA 4.0*
      - 1.6.2: Illustration- Cows in Děčín - *CC BY-NC-SA 4.0*
      - 1.6.3: Illustration- Ruritanian Crops - *CC BY-NC-SA 4.0*
      - 1.6.4: Illustration- Wealth in Ruritania - *CC BY-NC-SA 4.0*
      - 1.6.5: Illustration- Elections in Ruritania - *CC BY-NC-SA 4.0*
      - 1.6.6: Illustration- Insurance in Ruritania - *CC BY-NC-SA 4.0*
      - 1.6.7: Illustration- Warmth for the King - *CC BY-NC-SA 4.0*
    - 1.7: Conclusion - *CC BY-NC-SA 4.0*
  - 2: Part One- Least Squares - *CC BY-NC-SA 4.0*
  - 3: Intro to Linear Regression - *CC BY-NC-SA 4.0*
    - 3.1: Scalar Representation - *CC BY-NC-SA 4.0*
    - 3.2: First Results - *CC BY-NC-SA 4.0*
    - 3.3: Our First Assumptions - *CC BY-NC-SA 4.0*
    - 3.4: PRE Measures - *CC BY-NC-SA 4.0*
    - 3.5: Conclusion - *CC BY-NC-SA 4.0*
    - 3.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 4: Matrices and Linear Regression - *CC BY-NC-SA 4.0*
    - 4.1: Matrix Representation - *CC BY-NC-SA 4.0*
    - 4.2: Predictions and the Hat Matrix - *CC BY-NC-SA 4.0*
    - 4.3: The Geometry of OLS - *CC BY-NC-SA 4.0*
    - 4.4: Multicollinearity and Categorical Independent Variables - *CC BY-NC-SA 4.0*
    - 4.5: Conclusion - *CC BY-NC-SA 4.0*
    - 4.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 5: Improved! Now with Probabilities - *CC BY-NC-SA 4.0*
    - 5.1: Probability Distributions - *CC BY-NC-SA 4.0*
    - 5.2: Test Statistics and Hypothesis Testing - *CC BY-NC-SA 4.0*
    - 5.3: Confidence Intervals - *CC BY-NC-SA 4.0*
    - 5.4: The Working-Hotelling Bands - *CC BY-NC-SA 4.0*
    - 5.5: Conclusion - *CC BY-NC-SA 4.0*
    - 5.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 6: Dood! Check the Requirements - *CC BY-NC-SA 4.0*
    - 6.1: Normality - *CC BY-NC-SA 4.0*
    - 6.2: Functional Fit - *CC BY-NC-SA 4.0*
    - 6.3: Homoskedasticity - *CC BY-NC-SA 4.0*
    - 6.4: Multicollinearity - *CC BY-NC-SA 4.0*
    - 6.5: Conclusion - *CC BY-NC-SA 4.0*
    - 6.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 7: Time for Some Examples - *CC BY-NC-SA 4.0*
    - 7.1: Violent Crime - *CC BY-NC-SA 4.0*
    - 7.2: Violent Crime, Wealth, Region - *CC BY-NC-SA 4.0*
    - 7.3: Cows in Děčín - *CC BY-NC-SA 4.0*
    - 7.4: Conclusion - *CC BY-NC-SA 4.0*
    - 7.5: End-of-Chapter Materials - *CC BY-NC-SA 4.0*
  - 8: Fixing the Violations - *CC BY-NC-SA 4.0*

- 8.1: The Issue of Boundedness - *CC BY-NC-SA 4.0*
- 8.2: The South Sudanese Referendum - *CC BY-NC-SA 4.0*
- 8.3: Adjustments for Heteroskedasticity - *CC BY-NC-SA 4.0*
- 8.4: Conclusion - *CC BY-NC-SA 4.0*
- 8.5: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

9: Beyond the Ordinary - *CC BY-NC-SA 4.0*

10: Other Least Squares - *CC BY-NC-SA 4.0*

- 10.1: Ordinary Least Squares - *CC BY-NC-SA 4.0*
- 10.2: Weighted Least Squares - *CC BY-NC-SA 4.0*
- 10.3: Generalized Least Squares - *CC BY-NC-SA 4.0*
- 10.4: May the (Strong) Force be with You - *CC BY-NC-SA 4.0*
- 10.5: Elections in Ruritania - *CC BY-NC-SA 4.0*
- 10.6: Conclusion - *CC BY-NC-SA 4.0*
- 10.7: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

11: Quantile Regression - *CC BY-NC-SA 4.0*

- 11.1: Parameter Estimation - *CC BY-NC-SA 4.0*
- 11.2: Quantile Regression - *CC BY-NC-SA 4.0*
- 11.3: Conclusion - *CC BY-NC-SA 4.0*
- 11.4: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

12: Maximizing the Likelihood - *CC BY-NC-SA 4.0*

- 12.1: The Likelihood - *CC BY-NC-SA 4.0*
- 12.2: The MLE and the CLM - *CC BY-NC-SA 4.0*
- 12.3: Conclusion - *CC BY-NC-SA 4.0*
- 12.4: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

13: Beyond the Classical Model - *CC BY-NC-SA 4.0*

14: Generalized Linear Models - *CC BY-NC-SA 4.0*

- 14.1: The CLM and the GLM - *CC BY-NC-SA 4.0*
- 14.2: The Requirements for GLMs - *CC BY-NC-SA 4.0*
- 14.3: Assumptions of GLMs - *CC BY-NC-SA 4.0*
- 14.4: The Gaussian Distribution - *CC BY-NC-SA 4.0*
- 14.5: Generalized Linear Models in R - *CC BY-NC-SA 4.0*
- 14.6: Conclusion - *CC BY-NC-SA 4.0*
- 14.7: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

15: Binary Dependent Variables - *CC BY-NC-SA 4.0*

- 15.1: Binary Dependent Variables - *CC BY-NC-SA 4.0*
- 15.2: Latent Variable Modeling - *CC BY-NC-SA 4.0*
- 15.3: The Mathematics - *CC BY-NC-SA 4.0*
- 15.4: Modeling with the Logit - *CC BY-NC-SA 4.0*
- 15.5: Measuring Accuracy - *CC BY-NC-SA 4.0*
- 15.6: Modeling with Other Links - *CC BY-NC-SA 4.0*
- 15.7: Model Selection - *CC BY-NC-SA 4.0*
- 15.8: Conclusion - *CC BY-NC-SA 4.0*
- 15.9: End-of-Chapter Material - *CC BY-NC-SA 4.0*

16: Binomial Dependent Variables - *CC BY-NC-SA 4.0*

- 16.1: Binomial Distribution - *CC BY-NC-SA 4.0*
- 16.2: The Mathematics - *CC BY-NC-SA 4.0*
- 16.3: O Canada! - *CC BY-NC-SA 4.0*
- 16.4: Sri Lanka in 2010 - *CC BY-NC-SA 4.0*
- 16.5: Beta-Binomial Regression - *CC BY-NC-SA 4.0*
- 16.6: Conclusion - *CC BY-NC-SA 4.0*
- 16.7: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

17: Count Dependent Variables - *CC BY-NC-SA 4.0*

- 17.1: Linear or Poisson Regression? - *CC BY-NC-SA 4.0*
- 17.2: The Mathematics - *CC BY-NC-SA 4.0*
- 17.3: Overdispersion - *CC BY-NC-SA 4.0*
- 17.4: Body Counts - *CC BY-NC-SA 4.0*
- 17.5: The Bias-Variance Tradeoff - *CC BY-NC-SA 4.0*
- 17.6: Conclusion - *CC BY-NC-SA 4.0*
- 17.7: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

18: Nominal and Ordinal Dependent Variables - *CC BY-NC-SA 4.0*

- 18.1: Nominal Dependent Variables - *CC BY-NC-SA 4.0*
- 18.2: Ordinal Dependent Variables - *CC BY-NC-SA 4.0*
- 18.3: Cattle Feed - *CC BY-NC-SA 4.0*
- 18.4: The State University of Ruritania - *CC BY-NC-SA 4.0*
- 18.5: Conclusion - *CC BY-NC-SA 4.0*
- 18.6: End-of-Chapter Materials - *CC BY-NC-SA 4.0*

19: Appendices - *CC BY-NC-SA 4.0*

20: The Appendix of Matrices - *CC BY-NC-SA 4.0*

- 20.1: Matrix Basics - *CC BY-NC-SA 4.0*
- 20.2: Addition - *CC BY-NC-SA 4.0*
- 20.3: Multiplication - *CC BY-NC-SA 4.0*
- 20.4: Other Terms and Operations - *CC BY-NC-SA 4.0*
- 20.5: Consequences - *CC BY-NC-SA 4.0*
- 20.6: Statistics in Matrices - *CC BY-NC-SA 4.0*
- 20.7: End-of-Appendix Materials - *CC BY-NC-SA 4.0*

21: The Appendix of R - *CC BY-NC-SA 4.0*

- 21.1: Installing R - *CC BY-NC-SA 4.0*
- 21.2: R Packages - *CC BY-NC-SA 4.0*
- 21.3: R Functions - *CC BY-NC-SA 4.0*
- 21.4: Programming Practice - *CC BY-NC-SA 4.0*
  - 21.4.1: Plotting a Distribution - *CC BY-NC-SA 4.0*
  - 21.4.2: Functions of RVs - *CC BY-NC-SA 4.0*
  - 21.4.3: Effects of Rounding - *CC BY-NC-SA 4.0*
  - 21.4.4: Algorithmic Thinking - *CC BY-NC-SA 4.0*

- 21.4.5: Distribution of Extreme Values - *CC BY-NC-SA 4.0*
- 21.4.6: Non-Parametric Bootstrap - *CC BY-NC-SA 4.0*
- 21.4.7: Distribution of Sample Means - *Undeclared*
- 21.4.8: Distribution of Sample Sums - *Undeclared*

22: The Appendix of Statistics - *CC BY-NC-SA 4.0*

- 22.1: Importantly Confusing Points - *CC BY-NC-SA 4.0*
- 22.2: Sample Statistics - *CC BY-NC-SA 4.0*
- 22.3: Population Parameters - *CC BY-NC-SA 4.0*
- 22.4: Probability Distributions - *CC BY-NC-SA 4.0*
- 22.5: Distributions of Sample Statistics - *CC BY-NC-SA 4.0*
- 22.6: Other Topics - *CC BY-NC-SA 4.0*
- 22.7: End-of-Appendix Materials - *CC BY-NC-SA 4.0*

Back Matter - *CC BY-NC-SA 4.0*

- Index - *CC BY-NC-SA 4.0*
- Glossary - *CC BY-NC-SA 4.0*
- Detailed Licensing - *CC BY-NC-SA 4.0*
- Detailed Licensing - *Undeclared*