Course Companion for MBA 8370 and MBA 8380

Scott Dressler

2023-01-08

Contents

VILLANO

UNIVERSI

Villanova School of

Preface

> "In ancient times they had no statistics so they had to fall back on lies."
>
> — Stephen Leacock

About this book...

This course companion is a collection of lecture notes I have compiled over my years of teaching *Analyzing and Leveraging Data* (MBA 8350). MBA 8350 was once a 3 credit hour, complete introduction to inferential statistics that covered univariate (*one variable*) and multivariate (*more that one variable*) analyses. In Fall 2022, this course was split into two halves. The first course (MBA 8370: Essential Business Statistics) now focuses on univariate analyses (with a small introduction to regression analysis at the end), while the second (MBA 8380: Analyzing and Leveraging Data) focuses on multivariate.

You will notice that this course companion covers both MBA 8370 and MBA 8380. Since students of MBA 8380 may want to refer back to the material covered they covered in MBA 8370, I thought it best to keep all material in one self-contained course companion. If you are currently in MBA 8370, feel free to look ahead to what the second course has in store for you.

I have custom made this course companion to provide a **free** resource to all students that covers all relevant topics of both courses. We **DO NOT** need an additional textbook.

Acknowledgements

This course companion would not be possible without the many students I have had the pleasure of teaching statistics to at VSB. Their questions, comments, and corrections to previous editions of these notes have made a significant contribution to what this product is. I am not going to call this a *final* product, because I hope my current and future students will continue to help me make improvements.

The original writing of this course companion was supported by a Villanova School of Business 2021 Teaching Innovation Grant.

MBA 8370

Chapter 1

Introduction

We are constantly bombarded with statistics on a daily basis (whether we like it or not). This chapter starts with some motivating examples that make heavy use of statistics, and an introduction to some important terminology and equations used to calculate descriptive measures (e.g., mean, standard deviation, etc.). We will start throwing around some data and R code in the applications, but we will not formally get into these details until later. When initially reading this chapter, you should skip or skim over the coding details and focus more on the conclusions. You can always come back to this chapter and focus on replicating the code after we cover R in subsequent chapters.

1.1 The "Big Picture" of Statistics

Question 1:

A wholesaler has an inventory light bulbs and wants to market them. What can be said about the average lifespan of the light bulbs in this inventory?

Question 2:

An economist wants to forecast the future state of output in the economy. What variables (i.e., leading indicators) should be included in her model? How much confidence should we place in her predictions?

Question 3:

A meteorologist wants to predict the path of a hurricane. How confident can we be in her predicted path of the storm?

Figure 1.1: How do we know so much about this unused lightbulb?

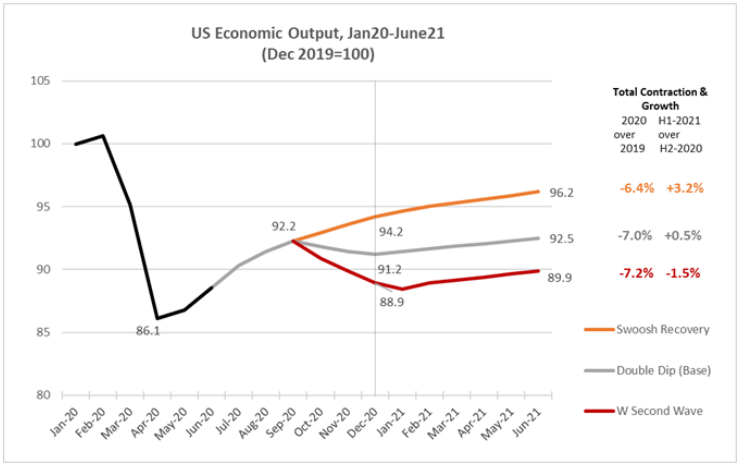


Figure 1.2: The Dismal Science

Figure 1.3: The Cone of Uncertainty

These and many more relevant questions can be answered (as best as possible) using **statistics**.

> **Statistics** is the branch of mathematics that transforms data into useful information for decision makers. Statistics are used to summarize data, draw conclusions, make forecasts, and make better sense of the world and the decisions we make within it.

Statistics is broken into two related branches.

> **Descriptive statistics** are used to summarize, analyze, and present data. Since the values of a dataset are in hand (i.e., observable), the descriptive statistics of a dataset are *facts*.

> **Inferential statistics** use the descriptive statistics from the data to draw conclusions about a larger group of observations that at the moment are *impossible* (or too expensive) to observe. Since this larger group of data is not in hand, the inferential statistics that stem from an analysis are *predictions*.

The ultimate goal of any statistical analysis is to take the information contained in the descriptive statistics (the facts) and use them to make *educated guesses* about what is going on in the larger group of observations (that you don't actually get to see). That is the power of inferential statistics - we draw conclusions

about observations that we never actually observe.

In order for us to adequately say what statistics is, we need to establish some terminology.

1.2 The Vocabulary of Statistics

A **variable** is a characteristic of an item or group that one wishes to analyze.

Data are the different values of the variable observed and recorded.

An **operational definition** establishes a meaningful use of the variable. Simply put, we need to establish that the data sufficiently captures what you want to analyze.

A **population** consists of all items you want to draw a conclusion from. The issue with a population is that it is the entire universe of observations that you are interested in, but they can never be fully observed.

- Sometimes a population is too costly to collect and analyze. For example, you won't call up every single voter for an election poll.
- Sometimes a population is impossible to collect because some observations have yet to be determined. For example, the population of end-of-day indices for the S&P 500 includes every observation that has ever existed as well as every observation **that has yet to exist**. This means that the true population is *infinitely* big!

A **sample** is the portion (i.e., subset) of a population selected for analysis. These are our observations in hand.

A **statistic** is a characteristic of a sample. Since we can observe the sample (i.e., our data), these are our descriptive statistics.

A **Parameter** is a characteristic of a population. Since we cannot observe the population, the best we can do is draw inferential statistics (or predictions) about them. While the value of a parameter exists, we would have to be omniscient in order to know it. The best we can do is use our sample statistics to construct an *educated guess* of what this value might be.

Recall the problem of the wholesaler who has a supply (i.e., population) of light bulbs. It would be great if we could state what the *average lifespan* of the light bulbs are, but that would require timing every light bulb until they burn out. This isn't very useful.

The seven terms stated above translate to our light bulb example as follows:

Term	Our light bulb problem
Variable	The lifespan of a light bulb
Data	The light bulbs that you actually plugged in and recorded the time it takes until burnt out
Operational Definition	The lifespan *in minutes*
Population	The entire group of light bulbs (all 100,000 of them)
Sample	The subset of the population selected for analysis. Sometimes referred to as the *data sample*.
Statistic	The average lifespan of every light bulb **in the sample**
Parameter	The average lifespan of every light bulb **in the population**

Inferential statistics allow us to describe the parameter of a population by using the corresponding statistic of a sample. We will **never** be able to truly know the population parameter, because the information available in the sample is all we got.

How do we know if the sample statistic is a GOOD predictor of the population parameter? The kicker is that since we cannot observe the population, the only thing we can do is try our best to ensure that the characteristics of the sample are the same as the population. This has to do with sample selection - a very important topic that will be addressed soon. Before that, we will discuss the descriptive measures of data.

1.3 Descriptive Measures

This section summarizes the measures we use to describe data samples.

- **Central Tendency**: the central value of a data set
- **Variation**: the dispersion (scattering) around a central value
- **Shape**: the distribution pattern of the data values

These measures will be used repeatedly in our analyses, and will affect how confident we are in our conclusions.

To introduce you to some numerical results in R, we will continue with our light bulb scenario and add some actual data. Suppose we randomly selected 60 light

bulbs from our population (i.e., our sample), turned them on, and timed each one until it burned out. If we recorded the lifetime of each light bulb, then we have a dataset (or data sample) of 60 observations of the lifetimes of light bulbs. This is what we will be using below.

1.3.1 Central Tendency

The **Arithmetic** or **sample mean** is the average value of a variable within the sample.

$$\bar{X} = \frac{\text{Sum of values}}{\text{Number of observations}} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

```
# the mean of our light bulb sample:
# First we load the data set and this will give us 60 observations of the lifespan of
load("data/Lightbulb.Rdata")
list(Lifetime)
```

```
## [[1]]
##  [1]  858.9164  797.2652 1013.5366 1064.8195  874.2275  825.1137  897.0879
##  [8]  924.0998  870.0674  966.2095  955.1281  977.2073  888.1690  826.6483
## [15]  776.7479  877.5691  998.7101  892.8178  886.0261  831.7615 1082.9650
## [22] 1034.9549  784.5026  919.2082 1049.1824  923.5767  907.7295  890.3758
## [29]  856.4240  808.8035 1009.7146  890.3709  930.9597  809.9274  919.9381
## [36]  793.7455  919.9824  948.8593  810.6887  846.9573  955.3873  833.2762
## [43]  892.4969  973.1861  913.7650  928.6057  940.7637  964.4341  914.2733
## [50]  880.3329  831.5395  967.2442 1030.7598  857.5421  889.3689 1094.1440
## [57]  927.7684  730.9976  918.8359  867.5931
```

```
(mean(Lifetime))
```

```
## [1] 907.5552
```

The average lifetime of our 60 ($n = 60$) observed light bulbs is 908 hours.

The **median** is the middle value of an ordered data set.

- If there is an odd number of values in a data set, the median is the middle value
- If there an even number, median is the average of the two middle values

```
# the median of our light bulb sample:
(median(Lifetime))
```

```
## [1] 902.4087
```

The median lifetime of our 60 observed light bulbs is 902 hours.

Percentiles break the ordered values of a sample into proportions (i.e., percentages of the group of individual observations).

- Quartiles split the data into 4 equal parts - with each group containing 25 percent of the observations.
- Deciles split the data into 10 equal parts - with each group containing 10 percent of the observations.
- In general, the pth percentile is given by: $(p * 100)^{th} = p(n + 1)$

A percentile delivers an observed value such that a determined proportion of observations are less than or equal to that value. You can choose any percentile value you wish. For example, the code below calculates the 4th, 40th, 50th, and 80th percentiles of our light bulb sample.

```
# You can generate any percentile (e.g. the 4th, 40th, 50th, and 80th) using the quantile functio
(quantile(Lifetime,c(0.04, 0.40, 0.50, 0.80)))
```

```
##       4%      40%      50%      80%
## 787.8301 888.8889 902.4087 966.4164
```

This result states that 4% of our observations are less that 788 hours, 40% of our observations are less than 889 hours, and 80% of our observations are less than 966 hours. Note that the median (being the middle-ranked observation) is by default the 50th percentile.

```
(quantile(Lifetime,0.50))
```

```
##      50%
## 902.4087
```

The main items of central tendency can be laid out in a Six-Number Summary. This summary delivers the *range* of our observations (the maximum observed value minus the minimum), the mean, as well as the quartiles of the observations.

- Minimum
- First Quartile (25th percentile)
- Second Quartile (median)
- Mean
- Third Quartile (75th percentile)
- Maximum

```
summary(Lifetime)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   731.0   857.3   902.4   907.6   955.2  1094.1
```

1.3.2 Variation

The **sample variance** measures the average (squared) amount of dispersion each individual observation has around the sample mean.

Dispersion is a very important concept in statistics, so take some time to understand exactly what this equation of variation is calculating. In particular, X represents a variable and X_i is a single observation of that variable from a sample. Once the mean ($\bar{X}$) is calculated, $X_i - \bar{X}$ is the difference between a single observation of X and the overall mean of X. Sometimes this difference is negative ($X_i < \bar{X}$) and sometimes this difference is positive ($X_i > \bar{X}$) - which is why we must square these differences before adding them all up. Nonetheless, once we obtain the average value of these differences, we get a sense of how these individual observations are scattered around the sample average. This measure of dispersion is relative. If this value was zero, then *every* observation of X is equal to $\bar{X}$. The greater the value is from zero, the greater the average dispersion of individual values around the mean.

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

The **sample standard deviation** is the square-root of the sample variance and measures the average amount of dispersion in the *same units as the mean*. This is done to back-out the fact that we had to square the differences of $X_i - \bar{X}$, so the variance is technically denoted in *squared units*.

$$S = \sqrt{S^2}$$

```
(var(Lifetime))
```

```
## [1] 6235.852
```

```
(sd(Lifetime))
```

```
## [1] 78.96741
```

The variance of our sample of light bulb lifetimes is 6236 squared-hours. After taking the square root of this number, we can conclude that the standard deviation of our sample is 79 hours. Is this standard deviation big or small? The answer to this comes when we get to statistical inference.

Discussion

- The term $(X_i - \bar{X})$ is squared because individual observations are either above or below the mean by design. If you don't square the terms (making the negative numbers positive) then they will sum to zero by design.
- The term $(n-1)$ appears in the denominator because this is a *sample* variance and not a *population* variance. In a population variance equation, $(n-1)$ gets replaced with n because we know the population mean. Since we had to estimate the population mean (i.e., used the sample mean), we had to deduct one **degree of freedom**. We will talk more about degrees of freedom later, but the rule of thumb is that we deduct a degree of freedom every time we build a sample statistic (like sample variance) using another sample statistic (like sample mean).

The **coefficient of variation** is a relative measure which denotes the amount of scatter in the data relative to the mean.

The coefficient of variation (or CV) is useful when comparing data on variables measured in different units or scales (because the CV reduces everything to percentages).

$$CV = \frac{S}{\bar{X}} * 100\%$$

Take for example the Gross Domestic Product (i.e., output) for the states of California and Delaware.

```
library(readxl)
CARGSP <- read_excel("data/CARGSP.xls")
DENGSP <- read_excel("data/DENGSP.xls")

CGDP <- CARGSP$CGDP
DGDP <- DENGSP$DGDP

(mean(CGDP))
```

```
## [1] 2094764
```

```
(sd(CGDP))
```

```
## [1] 397103.9
```

```
(mean(DGDP))
```

```
## [1] 56996.58
```

```
(sd(DGDP))
```

```
## [1] 12395.78
```

A quick analysis of the real annual output observations from these two states between the years 1997 and 2020 suggest that the average annual output of California is 2,094,764 million dollars (with a standard deviation of 397,104 million) and that of Delaware is 56,997 million dollars (with a standard deviation of 12,396 million). These two states have lots of differences between them, and it is difficult to tell which state has more volatility in their output.

If we construct coefficients of variation:

```
(sd(CGDP)/mean(CGDP))*100
```

```
## [1] 18.95698
```

```
(sd(DGDP)/mean(DGDP))*100
```

```
## [1] 21.74828
```

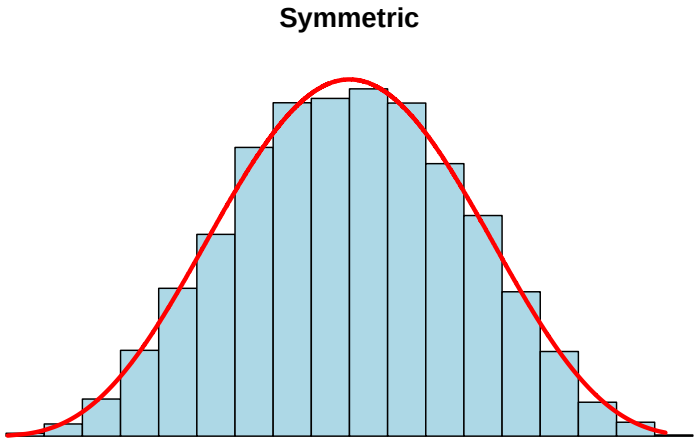
We can now conclude that Delaware's standard deviation of output is almost 22% that of its' average output, while California's standard deviation is 19%. This would suggest that Delaware has the more volatile output, relatively speaking, but they aren't that different.

1.3.3 Measures of shape

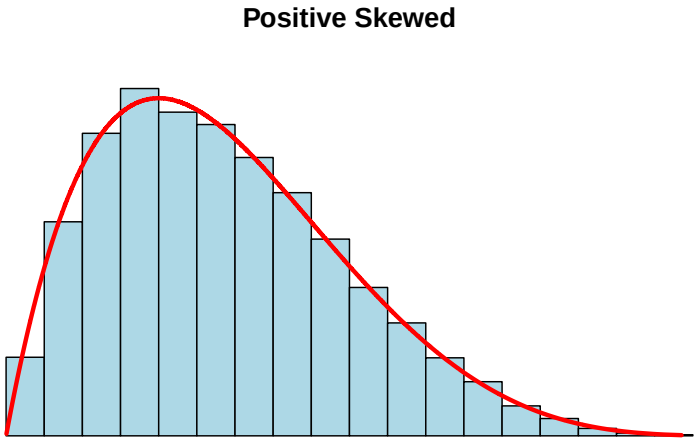
A **distribution** is a visual representation of a group of observations. We are going to be looking at plenty of distributions in the following chapters, so it will help to get an idea of their characteristics.

Comparing the mean and median of a sample will inform us of the skewness of the distribution.

- mean = median: a *symmetric* or *zero-skewed* distribution.

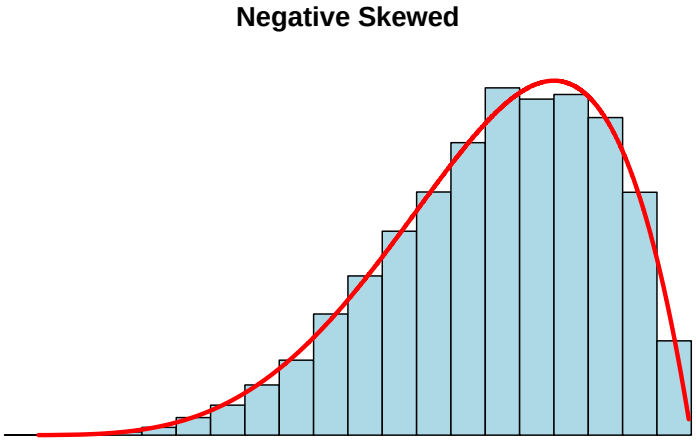


- mean > median: a *positive-skewed* or *right-skewed* distribution
  - the right-tail is pulled in the positive direction



- mean < median: a *negative-skewed* or a *left-skewed* distribution

– the left-tail is pulled in the negative direction



The degree of skewness is indicative of outliers (extreme high or low values) which change the shape of a distribution.

A classic economic example of a positive-skewed distribution is the average income distribution in the US. A large proportion of the individuals fall in the either the *low-income* or *middle-income* brackets, with a very small proportion falling in the *high-income* bracket. This high-income group pulls up the *average* amount of individual average income, but doesn't impact the median income because they are not in the bottom 50%. This is why mean > median in that distribution.

1.3.4 Covariance and Correlation

We won't be examining relationships between different variables (i.e., multivariate analyses) until later on in the companion, but we can easily calculate and visualize these relationships.

The **covariance** measures the strength of the relationship between two variables. This measure is similar to a variance, but it measures how the dispersion of one variable around its mean varies systematically with the dispersion of another variable around its mean. The covariance can be either positive or negative (or zero) depending on how the two variables move in relation to each other.

$$cov(X,Y) = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})$$

The **coefficient of correlation** transforms the covariance into a relative measure.

$$corr(X,Y) = \frac{cov(X,Y)}{S_X S_Y}$$

The correlation transformed the covariance relationship into a measure between -1 and 1.

- $corr(X,Y) = 0$: There is no relationship between X and Y. This corresponds to a covariance of zero.
- $corr(X,Y) > 0$: There is a positive relationship between X and Y - meaning that the two variables tend to move in the same direction. This corresponds to a large positive covariance.
- $corr(X,Y) < 0$: There is a negative relationship between X and Y - meaning that the two variables tend to move in the opposite direction. This corresponds to a large negative covariance.

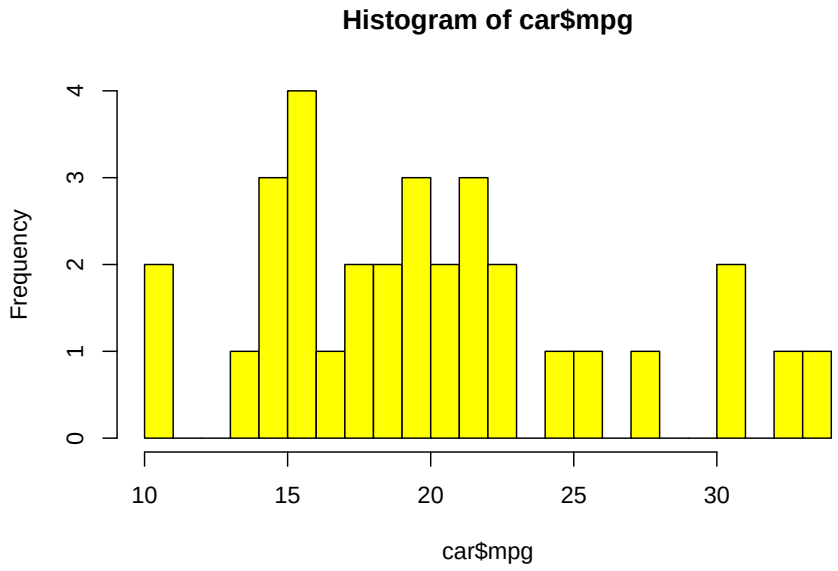
Extended Example:

This chapter concludes with a summary of all of the descriptive measures we discussed. Consider a dataset that is internal to R (called mtcars) that contains characteristics of 32 different automobiles. We will focus on two variables: the average miles per gallon (mpg) and the weight of the car (in thousands of pounds).

```
car <- mtcars # This command loads the dataset and calls it car

# Lets examine mpg first:
summary(car$mpg)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.40   15.43   19.20   20.09   22.80   33.90
```

```
hist(car$mpg,20,col = "yellow")
```




```
# Variance:
(var(car$mpg))
```

```
## [1] 36.3241
```

```
# Standard deviation:
(sd(car$mpg))
```

```
## [1] 6.026948
```

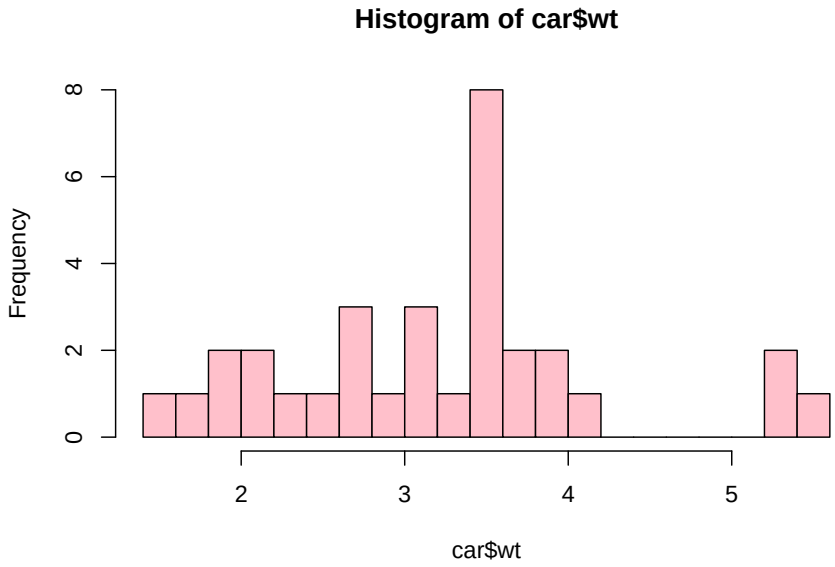
The above analysis indicates the following:

- The sample average MPG in the sample is 20.09, while the median in 19.20. This indicates that there is a slight positive skew to the distribution of observations.
- The lowest MPG is 10.4 while the highest is 33.90.
- The first quartile is 15.43 while the third is 22.80. This delivers the *inter-quartile range* (the middle 50% of the distribution)
- The standard deviation is 6.03 which delivers a 30 percent coefficient of correlation.

```
## Lets now examine weight:
summary(car$wt)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.513   2.581   3.325   3.217   3.610   5.424
```

```
hist(car$wt,20,col = "pink")
```




```
# Variance:
(var(car$wt))
```

```
## [1] 0.957379
```

```
# Standard deviation:
(sd(car$wt))
```

```
## [1] 0.9784574
```

The above analysis indicates the following:

- The sample average weight in the sample is 3.22 thousand pounds, while the median in 3.33. This indicates that there is a slight negative skew to the distribution of observations.
- The lowest weight is 1.51 while the highest is 5.42.
- The first quartile is 2.58 while the third is 3.61.
- The standard deviation is 0.99 which also delivers a 30 percent coefficient of correlation.

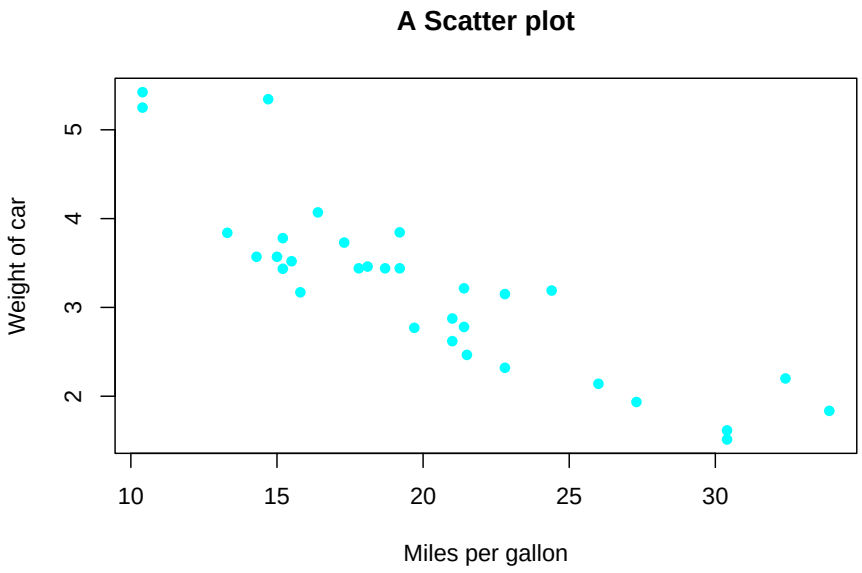
```
# We can now look at the relation between mpg and weight

(cov(car$mpg,car$wt))
```

```
## [1] -5.116685
```

```
(cor(car$mpg,car$wt))
```

```
## [1] -0.8676594
```

```
plot(car$mpg,car$wt, pch=16,
     xlab = "Miles per gallon",
     ylab = "Weight of car",
     main = "A Scatter plot",
     col = "cyan")
```



The negative correlation as well as the obviously negative relationship in the scatter-plot between the weight of a car and its miles per gallon should make intuitive sense - heavy cars are less efficient.

The Punchline

Suppose we want to learn about a relationship between a car's weight and its fuel efficiency. Our sample is 32 automobiles, but our population is EVERY automobile (EVER).[1] We would like to say something about the population mean Weight and MPG.

How does the sample variance(s) give us confidence on making statements about the population mean when we're only given the sample? That's where inferential statistics comes in. Before we get into that, we will dig into elements of collecting

[1] We could add more criteria such as every sedan, with a six-cylinder engine, etc.

data (upon which our descriptive statistics are based on) and using R (with which we will use to calculate our descriptive statistics using our collected data).

Chapter 2

Data Collection and Sampling

Always remember the ultimate goal of inferential statistics:

> We want to say something important about the characteristics of a population (parameters) without ever observing the entire population.

Since we can never get a hold of the population, the best thing we can do is to draw a subset of the population (i.e., a sample), use it to calculate the sample characteristics (i.e., statistics), and then draw inference on the population parameters.

The reason why we can say something about a population parameter of interest solely by looking at the statistics from a sample is because we are under the assumption that *the sample has the same characteristics as the population*. In other words, we say that the sample average is a good guess for the population average, the sample standard deviation is a good guess for the population standard deviation, etc. This is not an assumption that is simply made by wishful thinking. In fact, there is an entire field of statistics devoted to *sample selection*.

We won't spend a lot of time on this very important matter, and will instead assume in later chapters that the characteristics of the sample do in fact match those of the population. Nonetheless, this chapter will discuss a few sampling methods so you can rest assured that our crucial assumption of similar sample and population characteristics has a reasonable chance of holding.

2.1 Sampling Distributions

Recall that a **sample** is the subset of a **population** selected for analysis.

We are forced to analyze a sample rather than a population because:

1. selecting a sample is less time-consuming than selecting the population
2. selecting a sample is less costly
3. the resulting analysis is less cumbersome and more practical
4. sometimes obtaining the population is *impossible*! So the sample is the best we can do.

When making statements on the population parameters using the sample statistics, we are drawing **statistical inference**. In order for this inference to be reasonable, we must assume that the characteristics of the sample (i.e., the sample statistics) are reasonably close to the characteristics of the population (i.e., the population parameters). The problem with this assumption is that since we will never see the population, we will never be able to verify if the statistics are reasonably close to the parameters. This chapter discusses several different methods of drawing a sample from a population, as well as their pros and cons. The bottom line is that all of these methods attempt to get a sample to be the best possible subset of the population.

Failing to obtain a sample with the same characteristics as the population can fatally flaw a statistical analysis. If the sample statistics are not close to the population parameters, you are potentially over/under-representing important aspects of the population. When the sample statistics do not coincide with the population parameters, then the statistics are said to be *biased*. When this bias stems from a faulty sample, then this is called **sampling bias**.

2.2 Sampling Bias - two examples

There are quite a few glaring examples of sampling bias in history. One of them has to do with a rather famous photo:

2.2.1 Dewey Defeats Truman?

After defeating Thomas Dewey with a comfortable margin of 303 electoral college votes to Dewey's 189, President Harry Truman holds up a Chicago Daily Tribune stating the exact opposite. While the Truman Library would like to think that this iconic photo is an example of tenacity, perseverance, self-confidence, and success - it's actually a result of *sampling bias*.

The reporting error stems from the fact that the newspaper conducted a poll using phone numbers obtained from a list of vehicle registrations. Most people didn't have phones in 1948, and the people that were being polled had

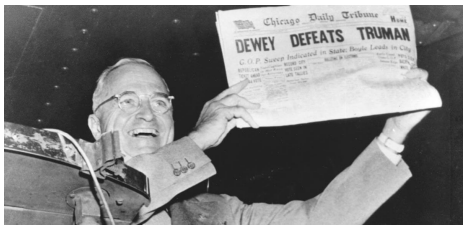


Figure 2.1: Dewey Defeats Truman? (1948)

both phones and automobiles. This skewed the sample distribution to wealthy, white, males - which obviously did not share similar views on the presidential candidates as the overall voting population.

2.2.2 98.6?

Everybody grew up with the following *fact* about their body…

> The average human body temperature is 98.6 degrees F (37 degrees C).

Is it really? To put this issue in the context of our terminology, the *average human body temperature* is a population parameter. The population here is every human that has ever lived and ever will live (i.e., an unobtainable amount of data). This average is actually a sample average obtained by a German physician in 1851 - a time believed by many current physicians to be one where many suffered from chronic infectious diseases resulting in mild fevers. Current studies are suggesting the average human body temperature is more like one degree lower than previously thought!

Now to be clear, there is a bit of a semantic argument about this last example. Some empiricists do not call this necessarily a sampling bias issue in 1851, because if a large portion of the population did regularly suffer from mild fevers then the sample was an accurate subset of the population at the time. Of course, if one is saying that the 1851 estimate of 98.6 degrees F is a representation of the *current* population - then that can be regarded as sampling bias.

2.3 Sampling Methods

A sampling process begins by defining the **frame** - a listing of items that make up the population. A frame could be population lists, maps, directories, etc. For our Truman example above, the frame was incorrectly chosen to be a list of registered vehicle owners (so the poll was doomed from the start).

A sample is drawn from a frame. The sample could be a **nonprobability sample** or a **probability sample**. The items in a nonprobability sample are selected without knowing their probabilities of selection. This is usually done

out of convenience (e.g., all voluntary responses to a survey or selecting the top or bottom of a frame). While these samples are quick, convenient, & inexpensive, they most likely suffer from selection bias. We can perform (albeit, incorrectly) inferential statistical analyses on these samples, but we are going to restrict attention to probability samples which are selected based on known probabilities.[1]

2.3.1 Simple random sampling

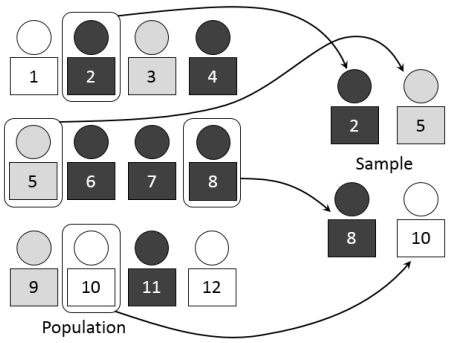


Figure 2.2: Simple Random Sampling

In a **simple random sample**, every item in a frame has an equal chance of being selected.

The chance (or probability) of being selected depends on if you're selecting…

- With replacement (1/N chance for all)
- Without replacement (1/N, 1/(N-1), 1/(N-2), …)

Examples of simple random sampling methods:

- Fishbowl methods
- random number indexing

Advantages:

- Simple random sampling is associated with the minimum amount of sampling bias compared to other sampling methods.
- If the sample frame is available, selecting a random sample is very easy.

Disadvantages:

[1]For example, if you were to analyze the results from individuals that took the time to fill out a voluntary online poll after a product purchase, you will not necessarily be making inferential statements on the entire population of individuals who purchased the product. You would only be discussing the population of individuals who purchased the product and would spend time to actually fill out the survey! These two populations may have different characteristics.

- Simple random sampling requires a list of all potential respondents (sampling frame) to be available beforehand - which can be costly.
- The necessity to have a large sample size (i.e., lots of observations) can be a major disadvantage in practical levels

2.3.2 Systematic Sampling

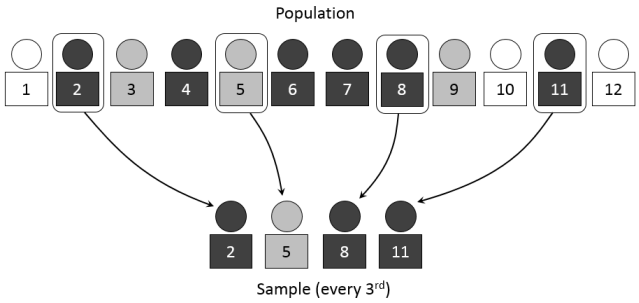


Figure 2.3: Systematic Sampling

A systematic sample begins with partitioning the N items in a frame into n groups of k items

$$k = \frac{N}{n}$$

- randomly select a number from 1 through k
- select the kth member from each of the n groups

For example: Suppose you want a sample n=40 out of N=800.

- Divide the population into k=20 groups.
- Select a number from 1-20 (e.g. 8)
- Sample becomes items 8,28,48,68,88,...

Advantages

- it will approximate the results of simple random sampling
- it is cost and time efficient

Disadvantages

- it can be applied only if the complete list of a population is available
- the sample will be biased if there are periodic patterns in the frame

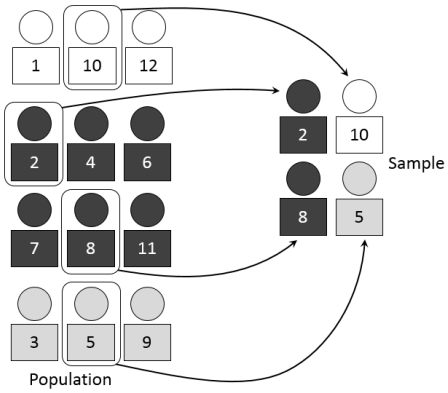


Figure 2.4: Stratified Sampling

2.3.3 Stratified Sampling

A stratified sample divides the N items in the frame into important subpopulations (strata)

- Each strata groups items according to some shared characteristic (gender, education, etc.)

Once these strata are constructed. A researcher selects a simple random sample from each strata and combines.

Advantages

- it is superior to simple random sampling because it reduces sampling error and ensures a greater level of representation
- ensures adequate representation of all subgroups
- when there is homogeneity within strata and heterogeneity between strata, the estimates can be as precise (or even more precise) as with the use of simple random sampling

Disadvantages

- requires the knowledge of strata membership
- process may take longer and prove to be more expensive due to the extra stage in the sampling procedure

2.3.4 Cluster Sampling

Cluster Sampling occurs when you break the sample frame into specific groups (i.e., clusters) and then randomly select several clusters as your sample. An example of this method is the consumer price index (CPI) which is a measure

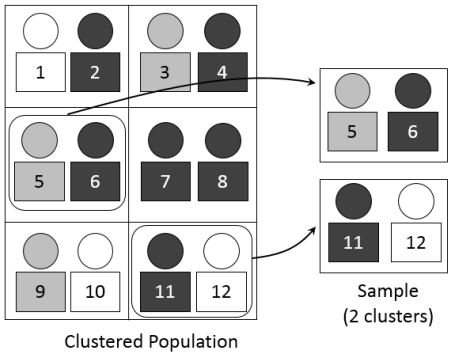


Figure 2.5: Cluster Sampling

of inflation calculated by the Bureau of Labor Statistics in the U.S. (US BLS). When trying to estimate the overall change in a *basket* of consumption goods across the US, the BLS breaks the US into metropolitan statistical areas (MSAs) and treats each one as a cluster. The BLS then goes and prices the various goods in the clusters selected for analysis.

Advantages:

- the most time-efficient and cost-efficient probability design for large geographical areas
- easy to use
- larger sample size can be used due to increased level of accessibility of perspective sample group members

Disadvantages

- requires group-level information to be known
- commonly has higher sampling error than other sampling techniques
- may fail to reflect the diversity in the sampling frame

2.4 Sampling in Practice

The sampling methods above deal with situations in which the population is hypothetically obtainable, but it is not feasible due to time or resource constraints. For example, a company could run a concrete election poll by calling up *every single registered voter* (the population), but that would cost too much and take too long. What happens in the situation where the population is unobtainable, meaning that at any point in time there will be some obtainable portion of the

population because it hasn't occurred yet. For example, if I wanted to analyze US unemployment rates, I couldn't possibly consider *future* rates that haven't been observed yet. In situation like these, one must take time to consider exactly what population you want to draw inferences from and draw their sample accordingly.

A quick example of data sampling in my own research is as follows. Some of my research deals with how bank lending responds to changes in the stance of monetary policy.[2] Since bank lending data is coming in daily, it is clear that the entire population is unobtainable. However, selecting a sample is not simply *collect as many observations as possible* because we must be clear about what population we want to actually talk about. In my example, I want to talk about how bank lending responds to monetary policy shocks *in normal times*. This means that observations in the sample cannot be impacted by episodes where monetary policy differed from what is currently considered normal. This restricts my sample to be after World War 2 and before episodes of unconventional monetary policy (i.e., anything post-2007).

What happens if characteristics of the population potentially changes? That's easy - you repeat the analysis with an updated sample and acknowledge that you are drawing inferences on a potentially different population. That is what I am currently researching. In particular, I am determining how bank lending responds to monetary policy under unconventional monetary policy practices of paying interest on excess reserves. This requires a data sample of observations appearing after 2007.

2.5 Sampling and Sampling Distributions

This chapter concludes with the hypothetical concept of a sampling distribution. Understanding this concept is crucial to understanding the entire point of inferential statistics.

Recall that we want to make statistical inferences that use statistics calculated from samples to estimate parameters of the population.

Plain statistics draws conclusions about the sample (those are facts) while statistical inference draws conclusions about the population.

In practice, a single sample is all that is selected. In other words, once you construct your sample it is all of the observations you have to use.

Since the actual observations inside your sample were selected at random, then the sample you constructed is in fact a *random* sample.

If the random observations were drawn from a sample frame, what was the resulting random sample drawn from? The answer is a **sampling distribution**.

[2] Dave, Chetan, Scott J. Dressler, and Lei Zhang, (2013). The bank lending channel: a FAVAR analysis. *Journal of Money, Credit, and Banking* 45(8). 1705-1720.

2.5.1 An Application

Consider the scenario discussed above where we want to determine the population average human body temperature. At a particular point in time, the population is every human. As the particular points in time change, new births implies that the population is changing as well! Clearly the overall population is unobtainable - so we need to draw a sample.

Suppose we decide on a sample size of 10,000 adults. Regardless of the sampling method chosen from the list above, we arrive at a data sample of 10,000 observations of human body temperatures. Since these individuals were selected *randomly*, then the sample mean calculated from the *random sample* is itself *random*. If we randomly draw another sample of 10,000 observations, we can get another sample average. We can do this repeatedly, getting a different sample average for every sample randomly drawn.

Note that this is purely hypothetical because we would never draw numerous samples… but bear with me.

We have established that our sample was a random draw from our population. Therefore, the sample mean calculated from our random sample is itself a random draw from a sampling distribution.

Think of a sampling distribution as a histogram showing you the outcomes of all possible sample means and their frequency of appearing. This distribution will have characteristics of its own. The mean of this distribution would be the mean value of all possible sample means. The standard deviation would be the amount of average dispersion all individual sample means around the overall mean.

What we will soon see is that this sampling distribution will be the foundation to inferential statistics. To see this, we will combine this concept of a sampling distribution with something called the Central Limit Theorem (CLT). The CLT is so important, it deserves its own chapter. However, before we get to that conceptual stuff, we will first get into the practical stuff. Namely, an introduction to the R project for Statistical Computing.

Chapter 3

Getting Started with R

This chapter is designed to get R on your machine (Section 1), introduce you to some basic commands for data and variable manipulation (Section 2), and introduce you to some introductory data visualization (Section 3). We will also be using a companion software called Rstudio which will make our interaction with R much more pleasant.

After the basics are covered in this chapter, you should be able to go back to Chapter 2 and have a better understanding of some of the code there. We will be learning additional R commands as they become needed in the subsequent chapters. By the end of the course, you should have a pretty solid understanding of working your way around R.

Let's get started!

3.1 The R Project for Statistical Computing

R is an open source (i.e. *free*) programming language and software environment for statistical computing, and is widely used among statisticians and data an-

alysts.[1] It is similar to other programming languages you may have heard of before (e.g., Python), but R is more suited to our data analysis needs. However, once you get the hang of one programming language, it is easier to adopt subsequent languages.

In addition to using R, we will also be using RStudio. RStudio is an integrated development environment (IDE) for R. The desktop version of R studio is also free, and comes with many useful features. In fact, this course companion you are currently reading was entirely formatted in R studio.

This section will walk you through downloading, installing, and preparing R and RStudio for our purposes. Once this is accomplished, you should be able to replicate every application contained in this course companion by using the code contained in the gray boxes. Let us begin!

3.2 Before you Install... RStudio Cloud?

Before venturing into downloading and installing R and RStudio, you should ask yourself a series of questions.

Do you have a Chromebook? The following section is intended for installation on either a PC or a Mac. For those of you using Chromebooks (which uses a Linux operating system), you will not be able to install these programs unless you partition your hard drive to run purely in Linux. If you don't know what that means, then I don't suggest you go down that route.

Do you have a rather old PC or old MAC? Installation of R and RStudio can be complicated on older computers. For example, if you have a PC with Windows 9 or earlier, you would need to download additional Microsoft programs to get R to install. If you have earlier MAC operating systems, you would need to install the specific version of R suitable for your OS.

Are you planning to use a work computer with a high level of security? Work computers are commonly shielded from the outside world with rather strong security firewalls. Being open source, R likes to regularly communicate with a *Mirror* server (explained below), and computers with high levels of security tend to prevent this.

If you answered *yes* to any of the above questions (or if you simply do not feel like installing more software on your computer), then the solution for you is the RStudio Cloud.

The website is https://rstudio.cloud/

Signing up for a free account with the RStudio Cloud has it's benefits…

[1] As of July 2020, R ranks 8th in the TIOBE index, a measure of popularity of programming languages.

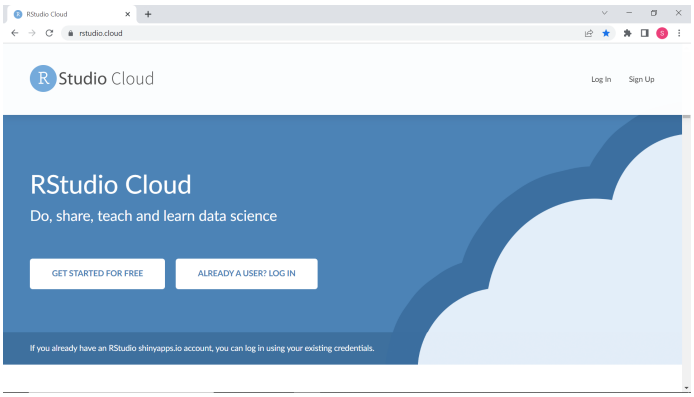


Figure 3.1: Welcome to RStudio Cloud!

1. You will be able to do everything for this class on the cloud-based version of the software without ever having to download anything.

2. Everything will be on your Cloud account, so any security firewalls will no longer be an issue.

3. You can run the most up-to-date version of R and RStudio no matter what operating system you have.

Of course, the RStudio Cloud also has a few drawbacks...

1. The RStudio Cloud requires an active internet connection when in use.

2. Your free account comes with limited usage time.[2] If you require more time, you are able to purchase additional usage time for a minimal fee.[3]

The bottom line is that the cloud-based version of RStudio is an excellent alternative to installing your own version of the course software. It is fairly new and prices are uncertain at the moment - and this is the only reason why this course has yet to *fully* adopt the cloud version for all students. Once prices have settled, this course will most likely shift to the cloud and no longer need students to install R and RStudio.

Note: If you decide to go with the RStudio option, you can skip the following section entitled *Downloading and installing R* and proceed to the section entitled *Taking Stock* and all remaining sections of the chapter. These sections are relevant for all students regardless of their version of RStudio.

[2]At the time of this writing, the account comes with 25 hours per month. This may be subject to change.

[3]At the time of this writing, you can purchase an additional 50 hours of usage time for $5 per month. This may be subject to change.

3.3 Downloading and installing R

This section is intended for students who are not going with the RStudio Cloud option discussed above, and are intending to download and install R and RStudio on your machine. Note that the directions below have you first downloading R and then RStudio. They are two separate programs that work together.

The first step to get R onto your machine is to go to the website, download the correct version of R, and install.

The website is https://www.r-project.org/.

3.3.1 Choosing a *Mirror*

Since R is open source, there are many different servers around the world where you can download it. You are welcome to choose any mirror (i.e., location) you wish, but you may want to be sure that you know the national language of whichever country you select. I was boring and simply chose a mirror in Pittsburgh, PA because it was closest to my location.

3.3.2 Download and install the correct version

R is available for PCs, Macs, and Linux systems. You will most likely want one of the first two options.[4] Be sure to choose the option in the top box that offers *Precompiled binary distributions*.

For Macs:

- Click on the "Download R for (Mac) OS X" link at the top of the page.
- Click on the file containing the latest version of R under "Files."
- Save the .pkg file, double-click it to open, and follow the installation instructions.

For PCs:

- Click on the "Download R for Windows" link at the top of the page.
- Click on the "install R for the first time" link at the top of the page.
- Click "Download R for Windows" and save the executable file somewhere on your computer. Run the .exe file and follow the installation instructions.

Once this is complete, you will **never need to actually open R**. We will be using RStudio to communicate with R - and the next section directs you through the installation of RStudio.

[4]While a Chromebook is technically a Linux system, you cannot install the software unless you partition the hard drive. This is why Chromebook users were directed to use the RStudio Cloud in the previous section.

3.3.3 Downloading and installing RStudio

While R is open source, RStudio is a company that sells versions of its Integrated Digital Environment (IDE) to use along with R. In short, it is an easy-to-use interface that makes working with R easier. We will be using the free version of RStudio Desktop which is available here:

https://rstudio.com/products/rstudio/download/

For Macs:

- Click on "Download RStudio Desktop."
- Click on the version recommended for your system, or the latest Mac version, save the .dmg file on your computer
- Double-click it to open, and then drag and drop it to your applications folder.

For PCs:

- Click on "Download RStudio Desktop."
- Click on the version recommended for your system, or the latest Windows version, and save the executable file.
- Run the .exe file and follow the installation instructions.

3.4 Taking Stock

If you have either installed the software or signed up to the RStudio Cloud correctly, you should now be able to open Rstudio (or your Cloud account) and see a screen that looks like this:

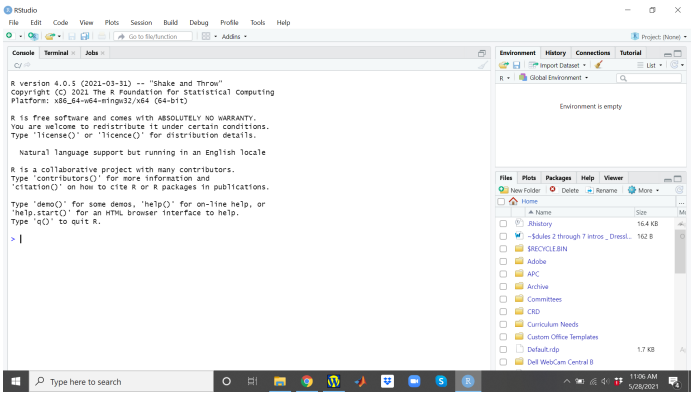


Figure 3.2: Welcome to R!

The window on the left is your *Console* which is exactly what you would see if you opened up R directly.[5] The window on the upper-right is your *Global Environment*. It will show you all of the data sets, variables, and result objects that are currently in R and available to you. Note that it is currently empty because we haven't done anything yet. The window on your bottom-right has several useful tabs that let us look at our folder directory (as shown) as well as any figures we generate and R packages at our disposal.

This is the default mode of Rstudio. You can input commands into the console right at the ">" and R will execute them line by line. This is fine if you wish to execute one single command at a time, but it becomes tedious if we have a series of commands we need to execute before we arrive at our desired result. We can therefore alter this default mode by adding R-scripts.

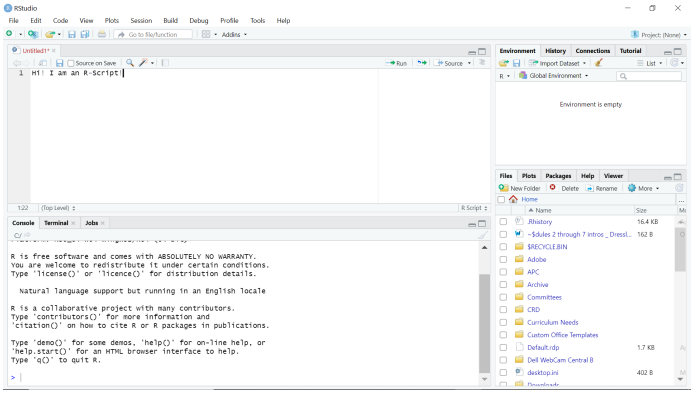


Figure 3.3: An R-script!

Clicking on the green plus in the upper left of the screen will give you the option of opening an R-script. An R-script window will now appear and take up half of what was once our console space. An R-script is really nothing more than a text file. We can type several commands in sequence without running them line by line (which we would need to do if we typed them into the console). Once the commands are typed out, we can highlight them all and hit that run button on top. The commands get sent to the console and you're off...

The picture above is just a quick example of what an R-script can do. Line 3 tells R to plot all of the variables in a dataset called *mtcars*. Highlighting that line and hitting the run button sends the command to the console below, and the plot figure shows up in the Plots window. That's that!

[5] Note that we will never open R by itself, because it is easier to communicate with R through Rstudio.

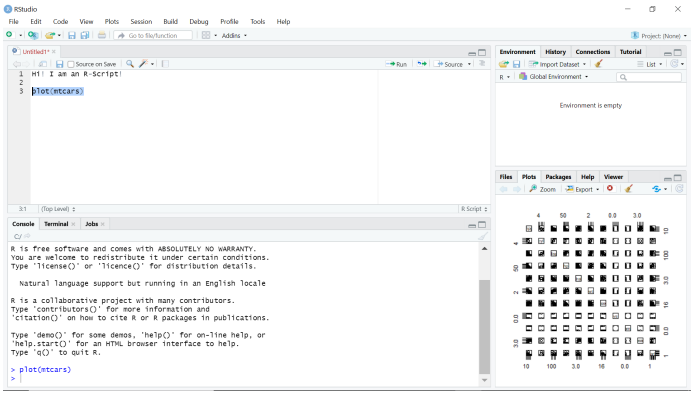


Figure 3.4: Running Commands from R-scripts!

3.5 Coding Basics

Now that your software is ready to go, this section introduces you to how R likes to be talked to. Note that the subsequent chapters are full of commands that you will need to learn when the time comes. In the meantime, here are just a few general pointers.

R is what is known as a line command computing language - meaning that it doesn't need to compile code prior to execution. That being said, try the following command at the prompt in your console (>):

```
12 + 4
```

```
## [1] 16
```

See? Just a big calculator.

3.5.1 Installing *Packages*

In order for R to be able to do some of the sophisticated things we will be doing in the course, we need to install source code called *packages*.

Whenever you need a package, all you need to do is type:

```
install.packages("name of package")
```

Once this is done, the package is installed in your version of R and you will never need to install it again.[6] You will need to *unpack* the packages each time you want to use them by calling a *library* command, but we will get to that later.

The first R-script you will run as part of your first assignment is called *Install_Packages.R*. The executable portion of the code looks like this:

[6] You need to do this for your Cloud-based version of R as well.

```
install.packages( c("AER", "car", "dplyr",
"fastDummies", "readxl", "xtable", "vars",
"WDI", "xts", "zoo", "wooldridge") )
```

This is a simple command that asks R to download 11 packages and install them. You can easily make your own RScript by opening up a blank RScript in R, and copying the code in the gray box above.[7] Highlight the portion above in your RScript, and hit the *Run* tab at the top of the upper-left window of RStudio. A bunch of notifications will appear on the R console (lower-left window) while the list of packages will be downloaded from the mirror site you selected earlier and installed. This can take some time (about 20 mins) depending on your internet connection, so it is advised you do this when you can leave your computer alone for awhile.

3.5.2 Assigning Objects

We declare variable names and other data objects by *assigning* things names. For example, we can repeat the calculation above by first assigning some variables the same numbers:

```
BIG <- 12
SMALL <- 4
(TOTAL <- BIG + SMALL)
```

```
## [1] 16
```

Notice that all of these variable names should now be in your global environment (upper-right window). The reason why 16 was returned on the console is because we put the last command in parentheses. That is the *print to screen* command.

You might be asking why R simply doesn't use an equal sign in stead of the assign sign. The answer is that we will be assigning names to output objects that contain much more than a single number. Things like regression output is technically *assigned* a name, so we are simple being consistent. You can use an equal sign in place of the assign sign for some cases, and everything will go through equally well. However, this doesn't work for every command we will use in this class.

3.5.3 Listing, Adding, and Removing

We can list all objects in our global environment using the list command: ls()

```
ls()
```

```
##   [1] "AL"          "alpha"       "AUTO"        "B1"          "Bhat0"
##   [6] "Bhat1"       "Bhat2"       "BIG"         "car"         "CARDATA"
##  [11] "CARDATA2"    "CARGSP"      "CDdata"      "CGDP"        "CM"
##  [16] "CREG"        "D"           "DENGSP"      "df"          "DGDP"
```

[7] You can also get the code directly from the first assignment.

```
##  [21] "DS"          "DTRND"       "e"           "eps"         "Fcrit"
##  [26] "fit"         "fitpoints"   "Fstat"       "grid.lines"  "h"
##  [31] "hprice1"     "i"           "i1"          "i2"          "j"
##  [36] "k"           "left"        "LEFT"        "left90"      "left99"
##  [41] "LFT"         "Lifetime"    "Lifetime1"   "Lifetime2"   "m"
##  [46] "M"           "MDAT"        "Mode"        "mtcars"      "mu"
##  [51] "MULTI2"      "n"           "N"           "P"           "probability"
##  [56] "Pval"        "R"           "R2r"         "R2u"         "Rate"
##  [61] "REG"         "REG1"        "REG2"        "REG3"        "REG4"
##  [66] "RES"         "Revenue"     "RHT"         "right"       "RIGHT"
##  [71] "right90"     "right99"     "RREG"        "S"           "SBhat1"
##  [76] "Sig"         "sigma"       "SMALL"       "t"           "t_values"
##  [81] "tcrit"       "TOTAL"       "tstat"       "UREG"        "wage1"
##  [86] "x"           "X"           "x.pred"      "X1"          "X2"
##  [91] "X3"          "Xbar"        "Xcrap"       "xfit"        "xtick"
##  [96] "xy"          "Y"           "y.pred"      "Y1"          "Y2"
## [101] "Y3"          "yfit"        "Yhat"        "Yz"          "Z"
## [106] "z.pred"      "Zcrit"       "Zstat"
```

As we already showed, we can add new variables by simply assigning names to our calculations.

```
TOTAL.SQUARED <- TOTAL^2
```

If you ever wanted to remove some variables from your global environment, you can use the remove command: rm(*name of variable*)

```
rm(TOTAL.SQUARED)
```

3.5.4 Loading Data

R can handle data in almost any format imaginable. The main data format we will consider in this class is a trusty old MS Excel file. Note that there is a zip file associated with this companion that contains all data files needed for replication. You will need to download that zip file onto your computer and unzip it before proceeding. It is recommended that you put all of you data files somewhere easy to access. Like a single folder directly on your C drive.[8]

There are two ways to load data...

1. The Direct Way

Once you locate a data file on your computer, you can direct R to import the file and assign it any name you want. The example below imports a dataset of automobile sales called AUTO_SA.xlsx and names it *CARDATA*.

[8] A folder on your *desktop* is the worst place for a data folder, because the file path is very messy. The closer to the C: drive, the better.

```
library(readxl)
CARDATA <- read_excel("data/AUTO_SA.xlsx")
```

The term *"data/AUTO_SA.xlsx"* is the exact location on my computer for this data file. Once you change the file path to your specification… you're done!

2. The Indirect (but easy) Way

You can also import data directly into R through Rstudio.

1. Use the files tab (bottom-right window) and locate the data file you want to import.
2. Left-click on file and select *Import Dataset…*
3. The import window opens and previews your data.
4. If everything looks good, hit *Import* and your done.

Note that the import window in step 3 has a *code preview* section which is actually writing the code needed to import the dataset. It will look exactly like what your code would need to look like in order to import data the direct way. You can refer to that for future reference.

3.5.5 Manipulating Data

You should now have a dataset named *CARDATA* imported into your global environment. You can examine the names of the variables inside the dataset using the list command - only this time we reference the name of the dataset.

```
ls(CARDATA)
```

```
## [1] "AUTOSALE" "CPI"      "DATE"     "INDEX"    "MONTH"    "YEAR"
```

When referencing a variable within a dataset, you must reference both the names of the dataset and variable so R knows where to get it. The syntax is:

Dataset$Variable

For example, if we reference the variable *AUTOSALE* by stating that it is in the CARDATA dataset.

CARDATA$AUTOSALE

We can now manipulate and store variables within the dataset by creating variables for what ever we need. For example, we can create a variable for real auto sales by dividing autosales by the consumer price index (CPI).

```
CARDATA$RSALES <- CARDATA$AUTOSALE / CARDATA$CPI
ls(CARDATA)
```

```
## [1] "AUTOSALE" "CPI"      "DATE"     "INDEX"    "MONTH"    "RSALES"
## [7] "YEAR"
```

3.5.6 Subsetting Data

Sometimes our dataset will contain more information than we need. Let us narrow down our dataset to see how we can get rid of unwanted data. You should see a little Excel looking icon to the left of the name CARDATA up in the global environment window. If you click on it, you should see the following:

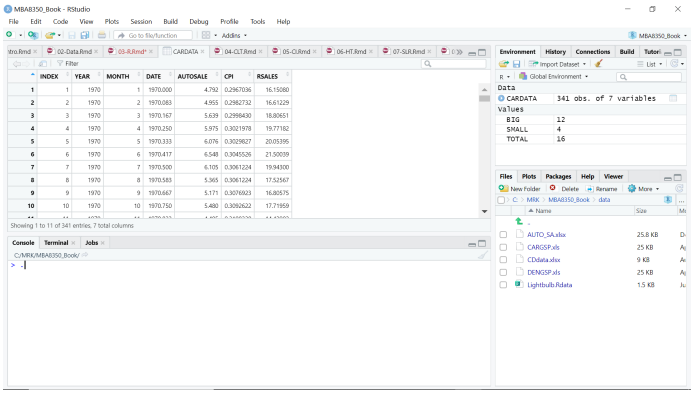


Figure 3.5: A Dataset in R

Thinking of the data set as a matrix with 341 rows and 7 columns will help us understand the code needed to select specific portions of this data.

Note that the variable MONTH cycles from 1 to 12 indicating the months of the year. Suppose we only want to analyze the 12th month of each year (i.e., December). We can do this by creating a new dataset that keeps only the rows associated with the 12 month.

```
CARDATA2 <- CARDATA[CARDATA$MONTH==12,]
```

What the above code does is treat the dataset CARDATA as a matrix and lists it as [rows,columns]. The rows instruction is to only keep rows where the month is 12. The columns instruction is left blank, because we want to keep all columns.

3.6 Data Visualization

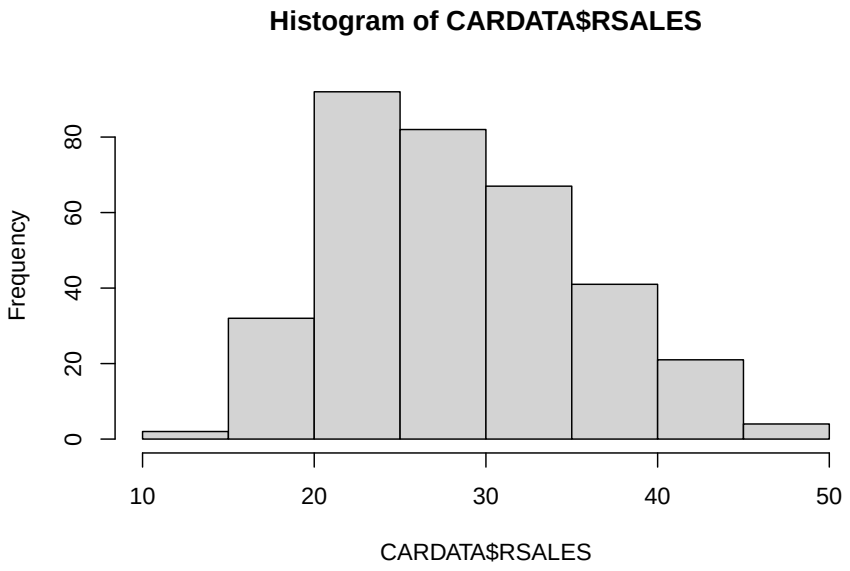
R is absolutely brilliant when it comes to data visualization, and this section will only scratch the surface of what it can do. We will go over some basic data visualizations using the built-in features of R. There are a lot of resources out there that covers a separate R package called ggplot. It's a handy package, but knowing the features discussed here will be sufficient for our course as well as give you some background that will help you push ggplot farther (if need be).

3.6.1 Histograms

A histogram breaks data observations into bins (or breaks) and shows the frequency distribution of these bins. We will use this to consider probability distributions, but it also helps us get an idea of the distributional properties of any data sample.

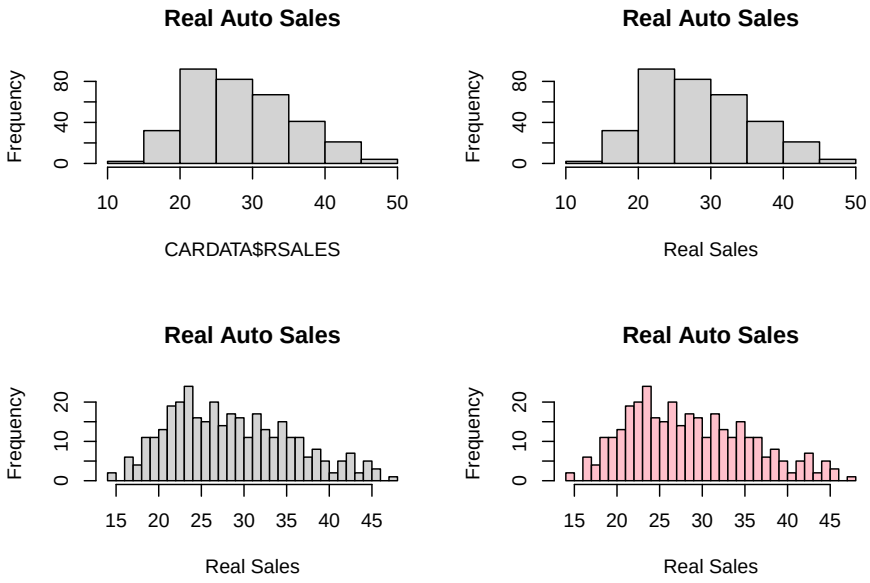
Let us continue to analyze the car dataset we created above:

```
hist(CARDATA$RSALES)
```



We can fancy this up by changing the title (main), labels (xlab), number of bins (breaks), and color (col). We will do this one at a time by creating a 2 by 2 set of figures using the par(mfrow=c(2,2)) command. This command *partitions* the plot window into a 2x2 series of subplots.

```
par(mfrow=c(2,2))
hist(CARDATA$RSALES,main = "Real Auto Sales")
hist(CARDATA$RSALES,main = "Real Auto Sales",
     xlab = "Real Sales")
hist(CARDATA$RSALES,main = "Real Auto Sales",
     xlab = "Real Sales",
     breaks = 40)
hist(CARDATA$RSALES,main = "Real Auto Sales",
     xlab = "Real Sales",
     breaks = 40,
     col = "pink")
```

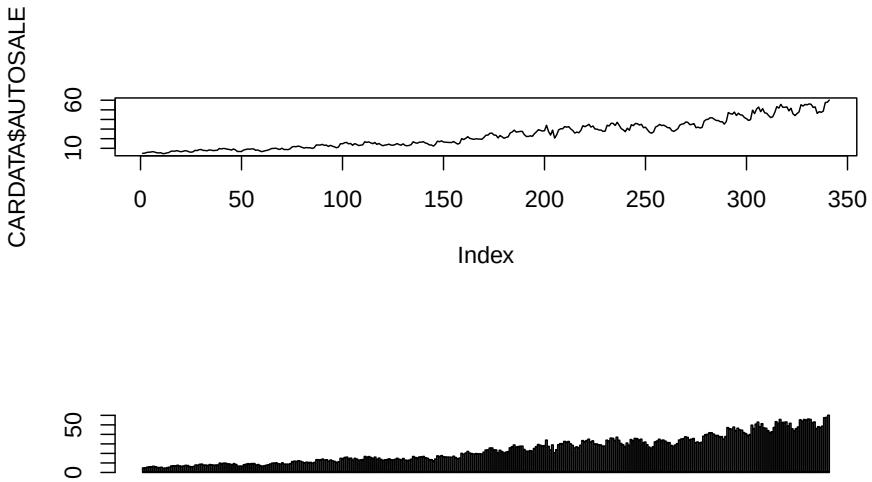


3.6.2 Line, bar, and Scatter Plots

The plot command can visualize the relationship between two variables or just one variable in order. The barplot command is similar to a single-variable plot.

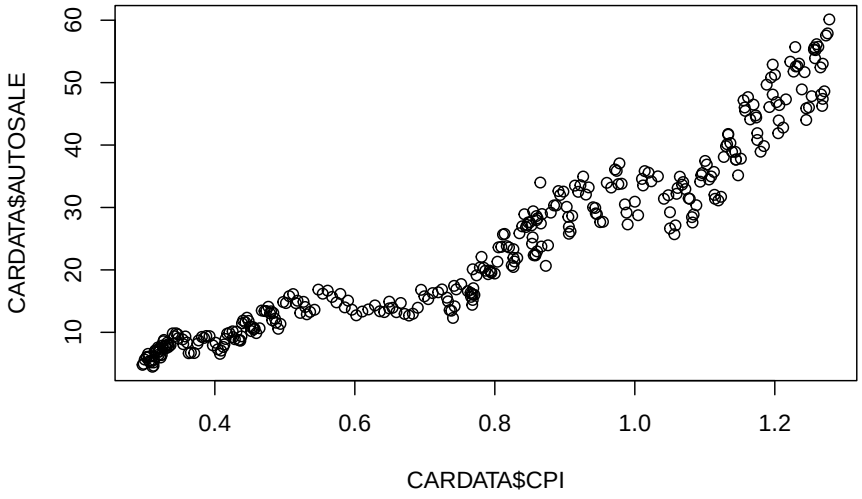
We can look at the nominal sales data in a line plot by specifying the type of plot as "l". A barplot delivers the same information, but just looks different.

```
par(mfrow=c(2,1))
plot(CARDATA$AUTOSALE, type = "l")
barplot(CARDATA$AUTOSALE)
```



We can look at relationships using the default values of the plot command.

```
plot(CARDATA$CPI,CARDATA$AUTOSALE)
```



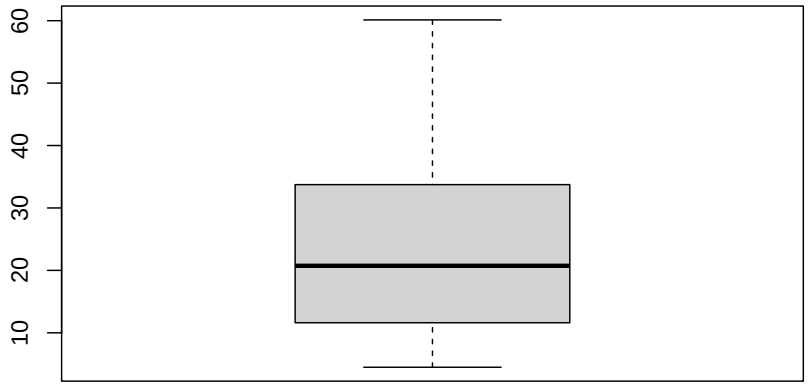
You will see plenty of these plots throughout these notes, and they will get

increasingly more sophisticated with titles, colors, etc.

3.6.3 Boxplots

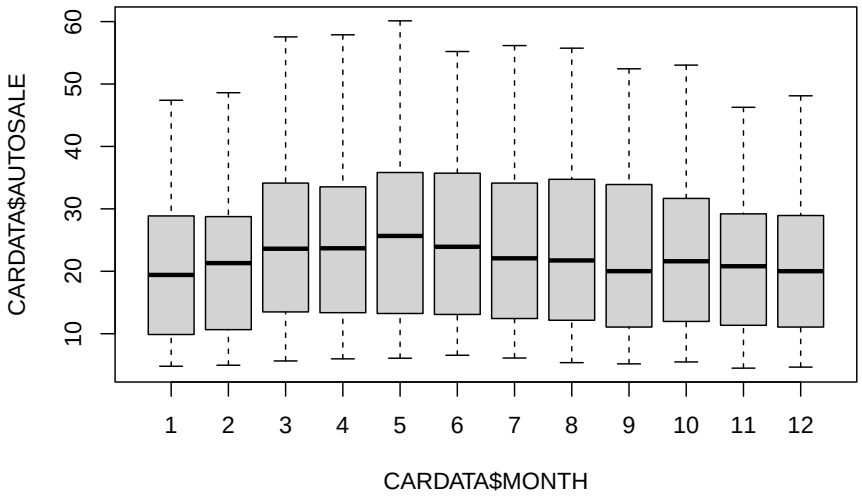
Box Plot illustrate the minimum, the 25th, 50th (median), 75th percentiles and the maximum. It is useful for visualizing the spread of the data.

```
boxplot(CARDATA$AUTOSALE)
```



We can also examine these five numbers within groups according to some other variable. Lets look at this breakdown of auto sales per month of the year.

```
boxplot(CARDATA$AUTOSALE~CARDATA$MONTH)
```



3.6.4 Much more out there

While this basically covers most of the plots we will need for the course, there is a ton more out there. The interested reader can consult a *free* book on the matter.

https://rkabacoff.github.io/datavis/

However deep you want to go, I hope you have seen that data visualization in R is a heck of a lot easier than in Excel.[9]

[9] For example, a histogram in MS Excel takes about 20 minutes for me to create each one!

Chapter 4

The Central Limit Theorem

The **Central Limit Theorem** (henceforth, CLT) is one of the most important conceptual foundations of inferential statistics. It is the essential reason why we can make educated guesses regarding the parameters of a population using information on the statistics of a sample. The CLT will going on in the background of every subsequent chapter of this course companion. Doing statistics without fully understanding the CLT is simply going through the motions. You will not be able to fully appreciate inferential statistics without knowing what is really going on beneath the hood.

4.1 The CLT (Formally)

Recall the concept of a sampling distribution briefly covered in chapter 2. For every randomly selected sample (i.e., a subset of the population), you can calculate a sample mean. If you were to repeatedly collect random samples and record their sample means, then you would be able to construct a *sampling distribution* of the sample mean values. This is would amount to a histogram (or frequency distribution) of all of your recorded sample means. Looking at this frequency of values would give you an idea of where you think the mean value from the next sample you would randomly draw will be. We will discuss this sampling distribution further below, but you should note that the statistical properties of this sampling distribution is where our educated guessing is coming from.

So here is the CLT formally…

> The central limit theorem states that if you have a population with mean μ and standard deviation σ and take sufficiently large random samples of size n from the population with replacement, then the distribution of the sample means will be approximately normally distributed.

There are some finer details to note.

- Given the population parameters μ and σ, the resulting sampling distribution will be a normal distribution with mean μ and standard deviation $\sigma/\sqrt{n}$.
- This will hold true regardless of whether or not the source population is normal, provided the sample size is sufficiently large (usually $n > 30$).
- If the population distribution is normal, then the theorem holds true even for samples smaller than 30. This however, is a rather extreme *if*.
- This means that we can use the normal probability distribution to quantify uncertainty when making inferences about a population mean based on the sample mean.

Now, the CLT can be proven - but I think it's better to illustrate the CLT with a couple of examples.

4.2 Application 1: A Sampling Distribution with a Known Population

The first application presents sampling distributions for a random process where we know the underlying process of the population: The rolling of two die.

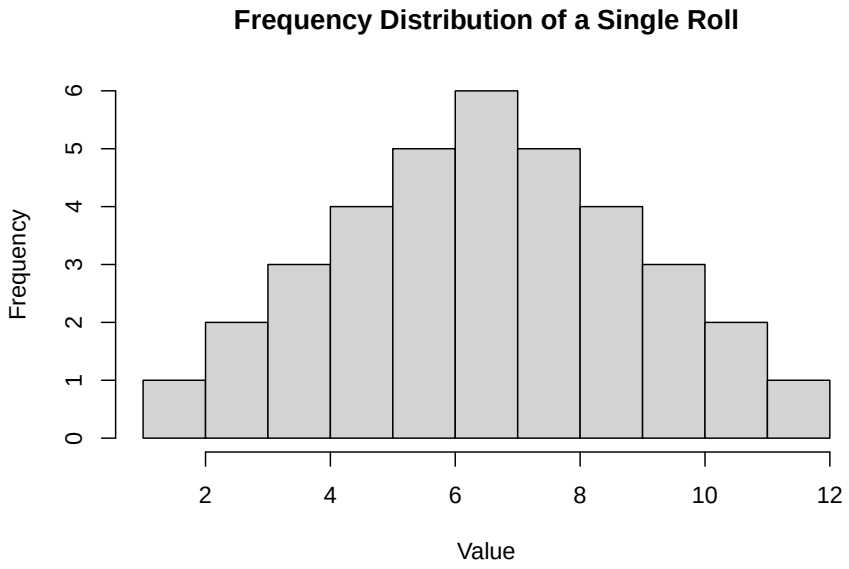
Suppose you worked for a gaming commission and placed in charge of making sure the dice at a casino were fair. **We know** that the (population) average roll of 2 fair die is 7 while the standard deviation is 2.45.[1]

It wouldn't be fair for you to test a set of dice by rolling them once because there is a large probability of rolling a number other than 7. In particular, there are 36 possible outcomes of rolling two die and only 6 of those outcomes equal 7. This means that although 7 is the highest probability single outcome, there is a much higher probability of rolling a number other than 7 (*ever play craps?*).

[1]The mean of a single dice throw is 3.5,

$$3.5 = (1 + 2 + 3 + 4 + 5 + 6)/6$$

and the expected value of two independent dice is the sum of expected values of each die. Standard deviation can be calculated using this mean value and the formula presented earlier.



The figure above is the population distribution of rolling two die. The average (mean) value is 7, the range of possible outcomes are between 2 and 12, and the standard deviation is a number that represents the dispersion of individual values around the mean. If you were to roll two die, then the outcome of that roll is conceptually a draw from this distribution.

Since we don't want to wrongfully accuse the casino of cheating, we need to roll the dice a few times to get an idea of what the average roll value is. If it is fair dice, then we know they will roll a 7 on average - but that means we would need to roll the dice an *infinite* amount of times to achieve this. To be realistic, lets settle on a number of rolls to be generally given by n. If we choose $n = 5$, then that means we roll the dice 5 times, record the roll each time, and then record the average. This is a sample average of a sample of size 5. We could do this for $n = 10$, $n = 30$, $n = 300$, etc.

The figure is illustrating four potential *sampling distributions*. For example, if you were to collect a sample of 5 rolls, then you would technically be drawing a sample average from the distribution in the upper left. On the other hand, if you decide to roll the dice 300 times, then you are technically drawing a sample average from the distribution in the lower-right.

There are two main takeaways from the above illustration.

1. Sampling distributions *appear* to approximate normal distributions. The normal distribution is the classic *bell-curve* distribution that tends to magically show up in empirical analyses. The CLT is the reason why. Note that even though the original distribution didn't look like a normal distri-

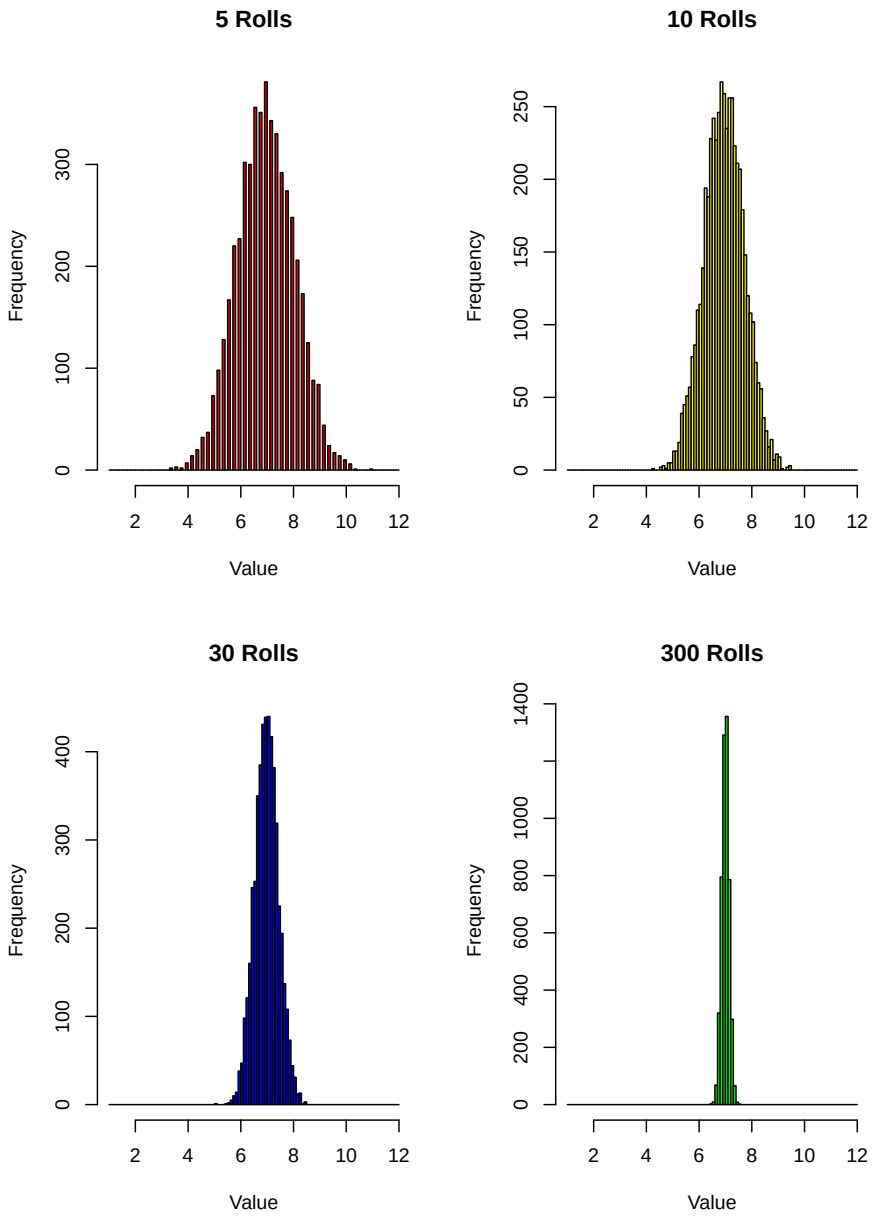


Figure 4.1: Sampling Distributions

bution at all, you still can construct sampling distributions that appear normal. This holds regardless of the initial population distribution.

2. Sampling distributions become *more* normal and have a lower standard deviation when the sample size gets bigger. Notice that as the sample size goes up, the distributions become narrower. This means that when there is a big sample size there is a very low probability that your going to see sample averages near 2 or 12. This should make sense: If you roll two dice 300 times and take the average, there is no way you are going to record a sample average of 2 unless you roll *snake eyes* 300 times in a row. As the sample size increases, the *extreme* events start getting diluted. This reduces the standard deviation of the sampling distribution.

3. The sampling distributions (for $n \geq 30$) are distributed normal with mean μ and standard deviation $\sigma/\sqrt{n}$. Technically this means that your random sample will produce a random outcome (a sample mean) which we denote $\bar{X}$.

$$\bar{X} \sim N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$$

You can see these two properties in the four sampling distributions illustrated above. All four sampling distributions are centered around 7, which is the population mean. As the sample size gets larger, the sampling distributions get *narrower* around the population mean. This illustrates why a larger sample has a better shot at becoming a better representation of the population.

4.3 Application 2: A Sampling Distribution with an Unknown Population

In most applications, we will not be as lucky as in the first application and we will know nothing about the underlying population. We won't know the distributional properties of the population, we won't know any of the population parameters… nothing. The beauty of the CLT is that this doesn't matter. We can still apply the CLT to set the stage for statistical inference.

In light of school closings back in 2020, the city of Philadelphia considered sending out $100 EBT cards to every student registered in public school.

A key question at the beginning of deliberation is how much would this policy cost?

- We know that there are 352,272 families in Philadelphia, and the city has records on how many students are registered in public schools.
  - Suppose it is too costly (at the initial stage) to determine the exact total number of children.

- If we knew the average number of children registered per family, we can get an estimate of the cost of the policy.

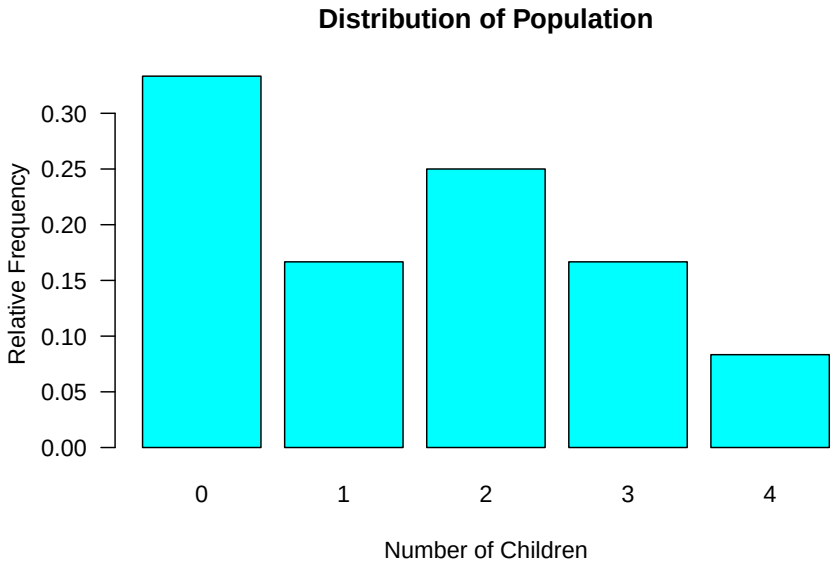
Pretend we are *omniscient*...

- The **POPULATION** average number of children per family is...

$$\mu = 1.5$$

$$\sigma = 1.38$$

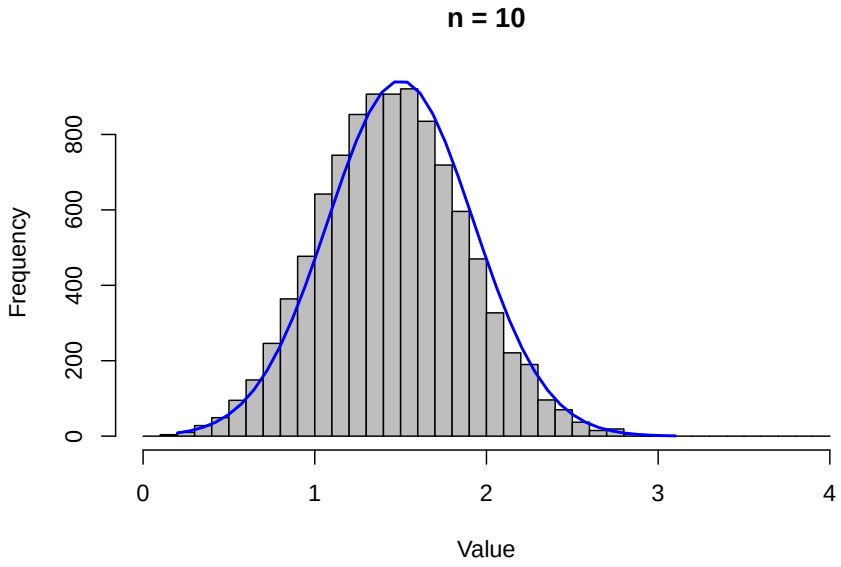
NOTE: We do not know these population parameters in reality. I am simply stating them here so we can refer to them later for verification. In reality, we will **never** know the true population parameters. That's why we need inferential statistics.

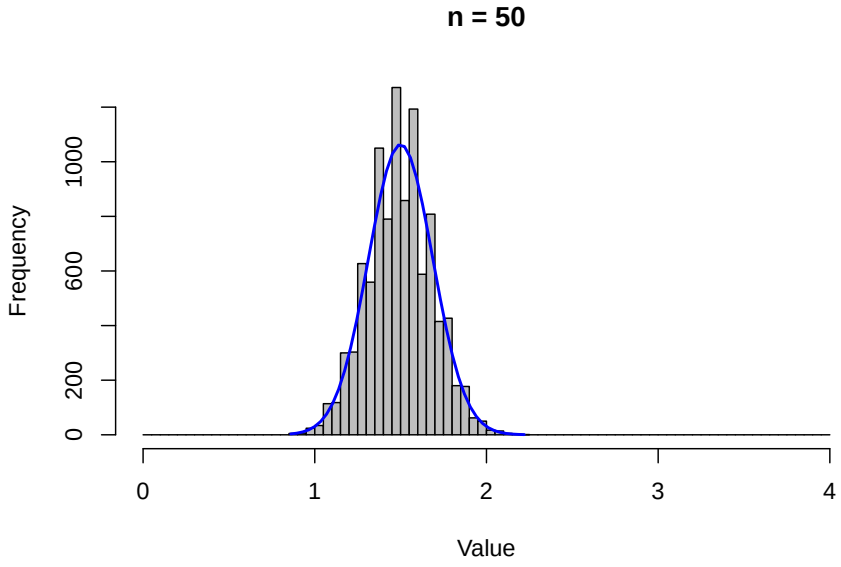


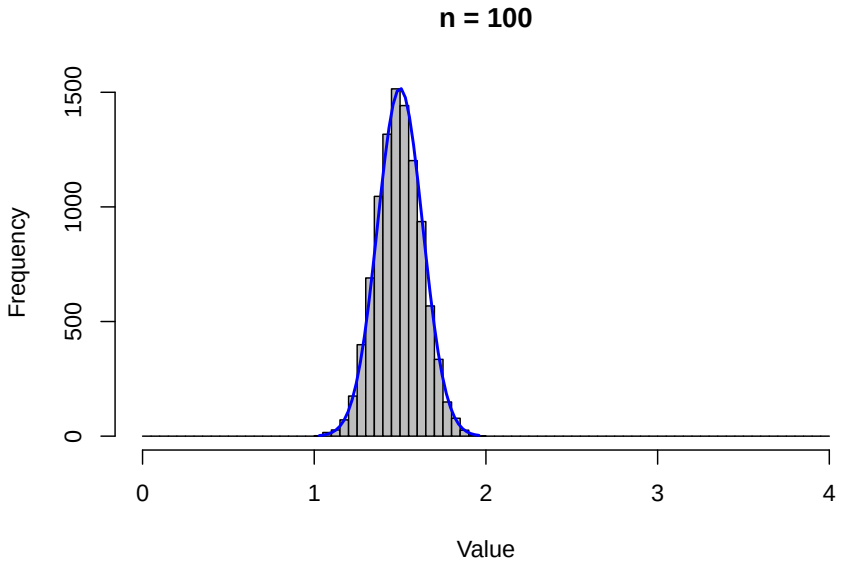
4.3.1 The Sample

- Since it is too costly to examine the entire population (at the initial stage), we draw a single sample.
- We use the sample to calculate sample statistics
- Since the sample is randomly drawn from the population, the sample statistics are randomly drawn from a sampling distribution.

The characteristics of the sampling distribution depends on the sample size n.


n = 10
Frequency
0
200
400
600
800
0
1
2
3
4
Value


n = 50
Frequency
0
200
600
1000
0
1
2
3
4
Value



The figures above show sampling distributions of various sample sizes. Note that all of these distributions are centered around the same number (of 1.5), and the dispersion around the mean is getting smaller as n is getting larger. In other words, the standard deviation $\sigma/\sqrt{n}$ is getting smaller because n is getting larger (while σ remains unchanged).

4.4 The Punchline

Once you determine a sample size (n), you get **one random draw** of a sample mean $\bar{X}$ from the appropriate sampling distribution.

- The distribution is approximately *normal*
- The mean is μ
- The standard deviation $\sigma/\sqrt{n}$

$$\bar{X} \sim N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$$

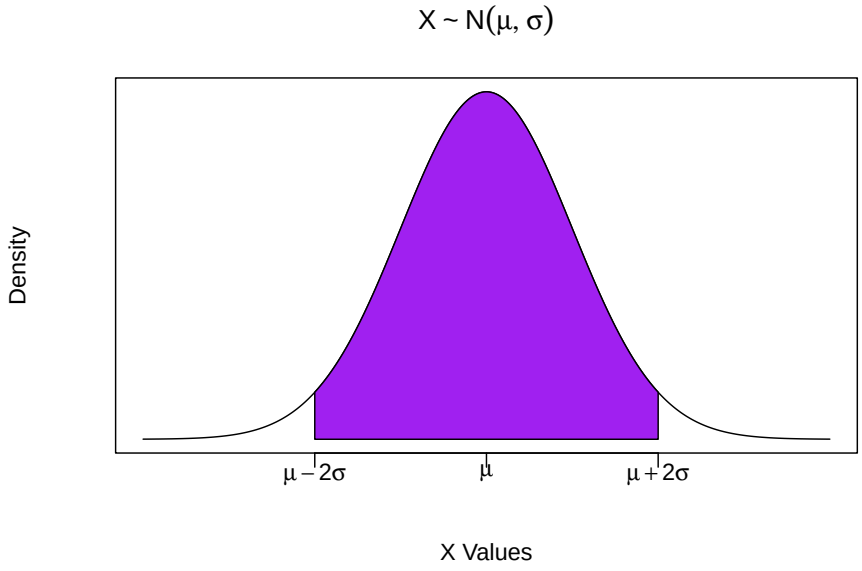
What does this buy us? The answer is *everything* if we want to apply any form of *confidence* (i.e., stating a probability of occurring).

The reason is that the normal distribution has a lot of useful properties.

1. The distribution is **symmetric**. The shape of the distribution to the right of the mean is identical to the shape of the distribution to the left of the mean.

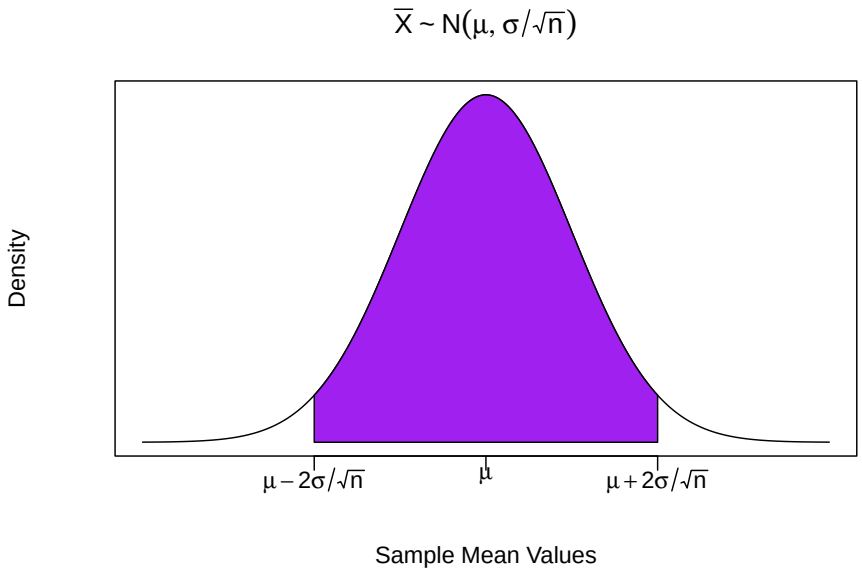
2. Approximately 95% of all possible outcomes are within 2 standard deviations of the mean.[2]

To illustrate these two properties, consider the generic normal distribution illustrated below. You can easily see the symmetry of the distribution, while the shaded area represents 95% of the distribution. In probability terms, 95% of the area of the probability distribution means that there is a 95% chance of drawing a value within this range.



So what does this really buy us? Consider the application above about the Philadelphia policy where we would have in reality have no idea what the population parameters (μ, σ) are, or what the population distribution even looks like. However, the CLT says that if we decide on a sample size n, then we will draw from a sampling distribution that is a normal distribution with mean μ and standard deviation $\sigma/\sqrt{n}$.

[2]Technically, 1.96 standard deviations.



So what we know is that once we draw a random sample and construct a sample mean, we can say with 95% confidence that that sample mean was drawn from the shaded region of the above distribution. We know what the sample mean value is because we just calculated it. What we don't know is what μ is. However, we can construct a probabilistic range (a *confidence interval*) around where we think this population parameter lies. This is where we are going next.

Chapter 5

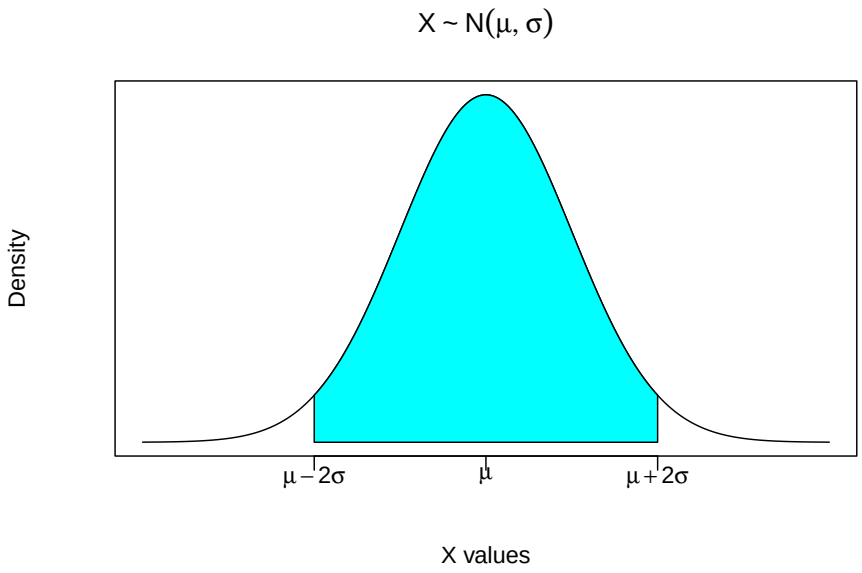
Confidence Intervals

With the concept of the Central Limit Theorem (CLT) under our belts, we can discuss our first application of statistical inference: *confidence intervals*. The main concept is that when a statistician discusses *confidence*, they are actually saying how likely something is going to happen. In other words, 95% confidence in a statement means that something is going to happen 95 out of 100 replications. We only get one shot (not 100 replications), so it is the likelihood that it's going to happen in this one shot.

In order for us to get into statistical inference, we need to first have a refresher on probability. Thanks to the CLT, our probability calculations are going to come from the normal probability distribution.

5.1 A Refresher on Probability

Suppose I give you a random variable (X) and tell you that this random variable comprises a normal distribution with an *arbitrary* mean equal to μ and an *arbitrary* standard deviation equal to σ. We can denote this generally as $X \sim N(\mu, \sigma)$ and we can draw this generally as



This picture is the normal probability density of this random variable. It is very much like a histogram, only we can consider a continuum of possible numbers (i.e., unlimited number of histogram bins). A normal probability density has several useful properties.

1. It is centered at the mean. It is a **symmetric** distribution with 50% of the probability being on either side of the mean.

2. Like all probability distributions, it must add up to 1 (or 100%). This is like saying that the probability of reaching into a hat full of numbers and pulling out a number between positive and negative infinity is equal to 100%.

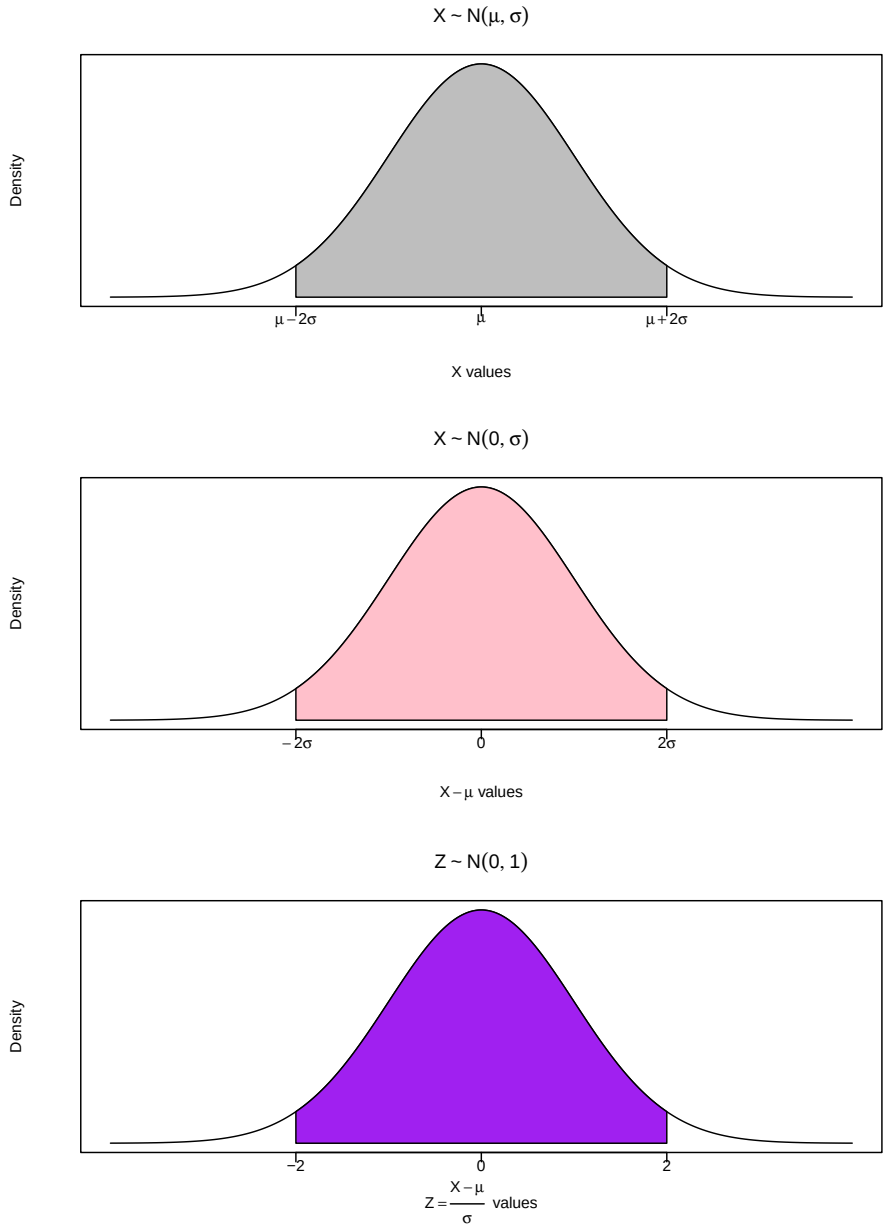
3. A normal distribution has the nice property that approximately 95% of the density area is between two standard deviation above and below the mean. This is the shaded area in the above figure. It roughly states that if you reached into a bowl full of numbers that comprised this distribution, then you have a 95% chance of pulling out a number between $\mu - 2\sigma$ and $\mu + 2\sigma$.

This is very useful, but for our purposes we need to take this arbitrary normal distribution and transform it into a **standard normal distribution**. We do this by applying what is known as a Z-transformation:

$$Z = \frac{X - \mu}{\sigma}$$

The figure below illustrates how this transformation changes an otherwise arbitrary normal distribution. The top figure is the arbitrary random variable with a mean of μ and a standard deviation of σ ($X \sim N(\mu, \sigma)$). The second figure shows what happens to the distribution when we subtract the mean from every number in the distribution. This effectively shifts the distribution such that it is now centered around zero, so we now have a normally distributed random variable with a mean of zero and a standard deviation of σ ($X - \mu \sim N(0, \sigma)$). The third figure shows what happens when we divide every number in the distribution by σ. Recall that σ is a positive number that can be greater or less than one. If you divide a number by a number less than one then the number gets bigger. If you divide a number by a number greater than one then the number gets smaller. This means that dividing every number by σ will either increase or decrease the dispersion of values such that the standard deviation is equal to one. A normally distributed random variable with a mean of zero and a standard deviation is said to be a standard normal random variable ($Z \sim N(0, 1)$).

Note that this transformation shifts the distribution, but **does not** change its properties. This was done on purpose to get you to see that a standard normal transformation shifts the mean and alters the dispersion, but does not change the facts that the distribution is still symmetric, still adds to one, and still has the property that 95% of the probability area is between 2 standard deviations to the right and left of the mean.



What does this transformation do? It takes a normally distributed random variable with arbitrary μ and σ, and transforms the distribution into one with mean 0 and standard deviation 1.

Why is this useful? It can easily be used for numerical probability calculations - but this isn't as useful nowadays since we have computers. However, this

transformation will be essential to put the normal distribution on the same level as other distributions we will soon encounter.

5.1.1 Application 1

Suppose there exists a bowl of numbered cards. The numbers on these cards comprises a normal distribution where the mean value is 5, and the standard deviation is 3.

$$X \sim N(5, 3)$$

We now have everything we need to calculate the probability of any outcome from this data-generating process. For example, suppose we wanted to determine the probability of reaching into this bowl and picking a number between 2 and 3. In probability terms:

$$Pr(2 \leq x \leq 3)$$

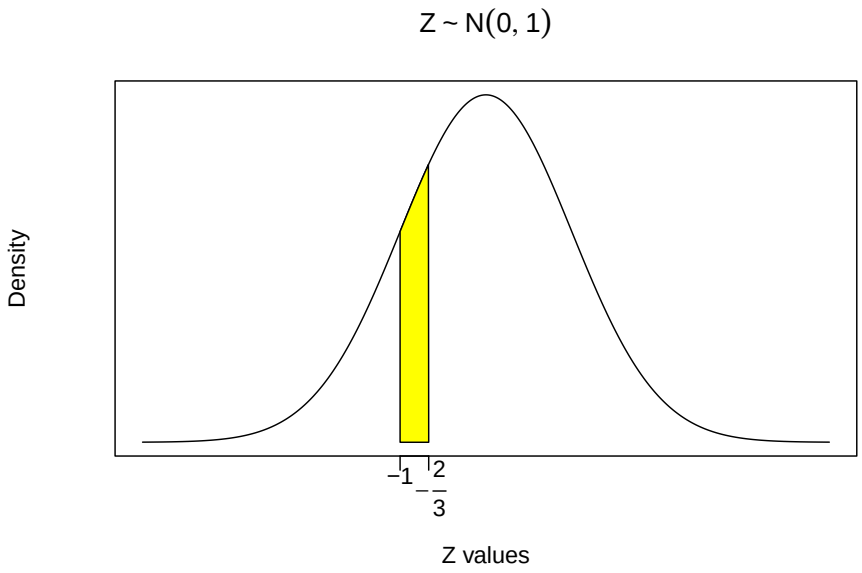
1. First we perform a standard normal transformation $Z = \frac{X-\mu}{\sigma} = \frac{X-5}{3}$, so our probability question gets transformed as well:

$$Pr(2 \leq x \leq 3) = Pr\left(\frac{2-5}{3} \leq \frac{x-5}{3} \leq \frac{3-5}{3}\right)$$

This delivers the same probability question, only in standard normal terms:

$$Pr(2 \leq x \leq 3) = Pr\left(-1 \leq z \leq -\frac{2}{3}\right)$$

2. Next we illustrate exactly what this probability question looks like in our distribution. In other words, indicate what *slice* of the distribution answers the probability question. This slice is illustrated in the figure below by shading in the probability area of the distribution between -1 and $-\frac{2}{3}$.



3. Finally, we calculate the probability in R. Now this is where the illustration above will help get us organized, because we can exploit the distributional properties of symmetry and the distribution summing to one. This is important because we can use R to calculate the same number in several different ways. All of these routes to the answer are acceptable, so we will go through them all here.

First thing to do is introduce you to the R command "pnorm"

```
pnorm(q, mean = 0, sd = 1, lower.tail = TRUE)
```

The command requires a number (*quantity*) for the variable q. It will then use a normal distribution with a mean of 0 and a standard error of 1 (*by default*) and calculate the area to the **left of the number q**. Note that this is the default action which is given by "lower.tail = TRUE". If you want to turn off this default action, then you need to set "lower.tail = FALSE" and the command will calculate the area to the **right of the number q**. For example, we can calculate $Pr(z \leq -1)$ or the area to the left of -1.

```
pnorm(-1)
```

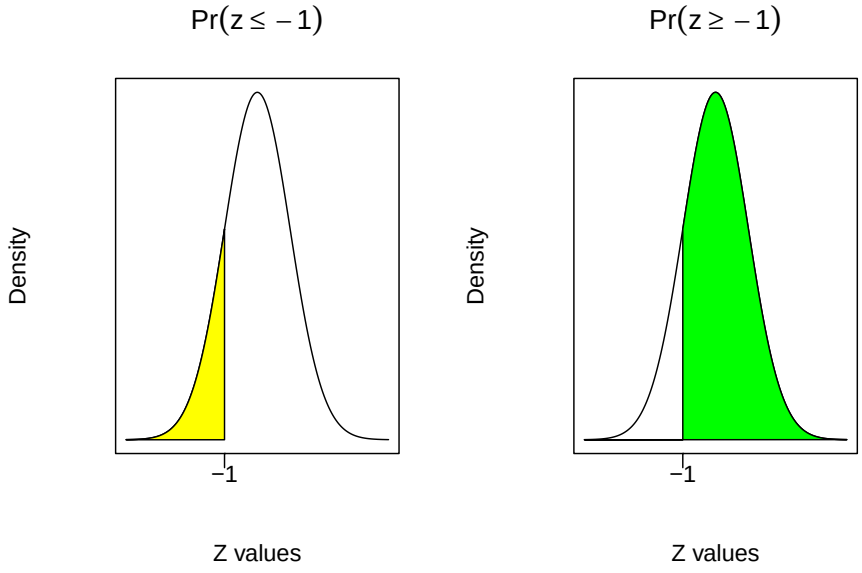
```
## [1] 0.1586553
```

We could also calculate $Pr(z \geq -1)$ or the area to the right of -1.

```
pnorm(-1,lower.tail = FALSE)
```

```
## [1] 0.8413447
```

These two probability areas sum to 1 (as they should), and are illustrated below. The left figure illustrates that 15.9% of the area under the distribution is to the left of -1, so you have a 15.9% chance of picking a number less than or equal to -1. Conversely, the right figure illustrates that 84.1% of the area under the distribution is to the right of -1, so you have a 84.1% chance of picking a number greater than or equal to -1.



Now that we know how R likes to calculate probabilities, we can use it to determine $Pr(-1 \leq z \leq -\frac{2}{3})$ which is the shaded slice of the distribution in the previous figure.

1. Using the *default* setting: suppose you want to calculate all of the probabilities using the default setting of calculating areas to the left. The shaded slice of the distribution is then the difference between the area to the left of $-\frac{2}{3}$ and the area to the left of -1.

$$Pr\left(-1 \leq z \leq -\frac{2}{3}\right) = Pr\left(z \leq -\frac{2}{3}\right) - Pr\left(z \leq -1\right)$$

```
pnorm(-2/3)-pnorm(-1)
```

```
## [1] 0.09383728
```

2. Removing the *default* setting. If you want to calculate probabilities from the right (which might come in handy), then the same slice of the distribution is the difference between the area to the right of -1 and the area to the right of $-\frac{2}{3}$.

$$Pr\left(-1 \leq z \leq -\frac{2}{3}\right) = Pr(z \geq -1) - Pr\left(z \geq -\frac{2}{3}\right)$$

```
pnorm(-1,lower.tail = FALSE)-pnorm(-2/3,lower.tail = FALSE)
```

```
## [1] 0.09383728
```

3. Exploiting that the area sums to 1. Yet another way to arrive at the same answer is to calculate the area to the left of -1, the area to the right of $-\frac{2}{3}$, and arrive at the slice by subtracting these areas from 1.

$$Pr\left(-1 \leq z \leq -\frac{2}{3}\right) = 1 - Pr(z \leq -1) - Pr\left(z \geq -\frac{2}{3}\right)$$

```
1-pnorm(-1)-pnorm(-2/3,lower.tail = FALSE)
```

```
## [1] 0.09383728
```

As you can see, each procedure delivers the same answer - you have a 9.4% chance of picking a number from a standard normal distribution between -1 and $-\frac{2}{3}$.

Note that this is the same answer to the original question (before we transformed the distribution). The take away from this exercise is that there are plenty of straightforward ways of calculating probabilities in R, and we will be making a fair amount of use of them.

$$Pr\left(-1 \leq z \leq -\frac{2}{3}\right) = 0.094$$

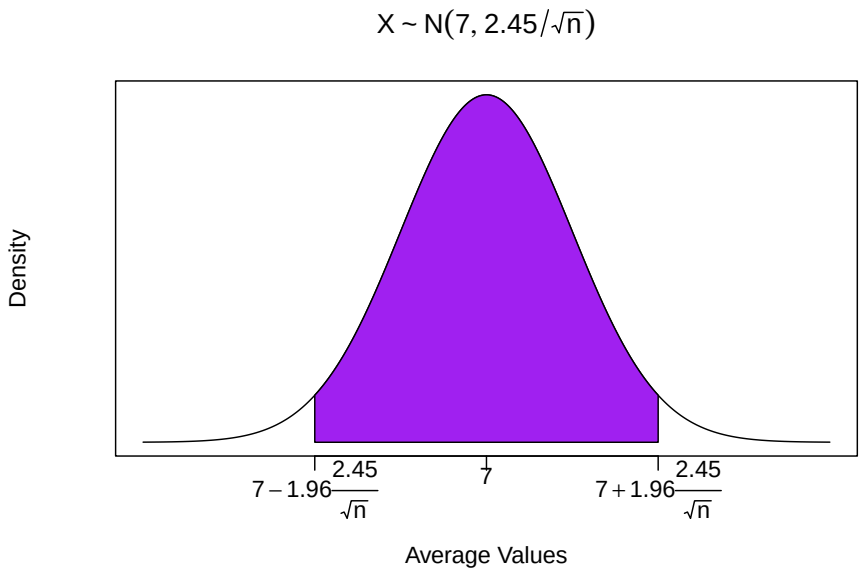
5.1.2 Application 2

Let us look deeper into our dice example. In particular, if I were to roll two fair dice a n number of times and calculated the average, what range of values should I expect to see?

Recall that the distribution of the population has a mean of 7 and a standard deviation of 2.45. This means that for $n \geq 30$, the *sampling distribution* is normal and given by

$$\bar{X} \sim N\left(7, \frac{2.45}{\sqrt{n}}\right)$$

Recall that in this (rare) example, we *know* the population parameters. Therefore, we can build a range where we expect sample averages to reside.



So if we collected a sample on $n = 100$, meaning we rolled two dice 100 times, recorded the total each time, and calculated the mean value, then…

$$Pr\left(7 - 1.96\frac{2.45}{\sqrt{100}} \leq \bar{X} \leq 7 + 1.96\frac{2.45}{\sqrt{100}}\right) = 0.95$$

```
Z = qnorm(0.975,lower.tail = FALSE)
n = 100
mu = 7
sigma = 2.45

(LFT = mu - Z * sigma / sqrt(n))
```

```
## [1] 7.480191
```

```
(RHT = mu + Z * sigma / sqrt(n))
```

```
## [1] 6.519809
```

$$Pr(6.52 \leq \bar{X} \leq 7.48) = 0.95$$

This means that with 95% confidence, the single outcome from your *experiment* will be within 6.52 and 7.48 if the die you are rolling are in fact fair.[1]

[1] Note that the code used a command called *qnorm*. This gets described further below.

As stated earlier, this example is rare because we know the population parameters. When we don't, we reverse engineer the probability statement so we can take what we know (the sample statistics) and use them to say something about what we don't. This is known as a confidence interval.

5.2 Deriving a Confidence Interval

Recall when we randomly draw a sample from a sampling distribution and use it to calculate a sample mean ($\bar{X}$), we essentially state that a sample mean is a random variable. Since the CLT states that the sampling distribution is a normal distribution, then this further states that the sample mean is a normally distributed random variable with a mean of μ and a standard deviation of $\frac{\sigma}{\sqrt{n}}$.

$$\bar{X} \sim N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$$

We can apply our standardization trick so we have

$$Z = \frac{\bar{X} - \mu}{(\sigma/\sqrt{n})}$$

Suppose we have a sample of size n, calculated a sample mean $\bar{X}$, and that we *know* the population standard deviation σ (more on this later). If we know these values, then we can use the standard normal distribution as well as the normalization above to draw statistical inference on the population parameter μ.

What can we say about μ?

- Since our sample has the same characteristics as the population, we would like to say $\mu = \bar{X}$ (i.e., $Z = 0$), but this is not likely. Recall the dice example discussed earlier, while 7 (the population mean) is the most likely average of a sample, there is a larger likelihood of a sample average *close* to, but not exactly equal to the population mean.
- Since $\bar{X}$ is a single draw from a normal distribution, we can construct a *probabilistic* range around μ. This range requires an arbitrary level of confidence $(1 - \alpha)$ - which provides bounds for the Z distribution (i.e., it gives us the area under the curve).

We therefore start with a probabilistic statement using a standard normal distribution:

$$Pr\left(-Z \leq \frac{\bar{X} - \mu}{(\sigma/\sqrt{n})} \leq Z\right) = 1 - \alpha$$

This states (in general terms) that the probability of realizing a value of $\frac{\bar{X}-\mu}{(\sigma/\sqrt{n})}$ drawn from a standard normal distributed random variable to be between the values -Z and Z is equal to $1-\alpha$. To put this into context, suppose I set $\alpha = 0.05$ so $1-\alpha = 0.95$ implies that I am looking for something that will occur with 95% probability. Recall the normal distribution has the nice property that 95% of the probability space is approximately between two standard deviations above and below the mean. By approximately, I mean it is actually 1.96 and not 2.[2]

Finding these numbers requires another R command: qnorm.

```
qnorm(p, mean = 0, sd = 1, lower.tail = TRUE)
```

Just like how pnorm takes a quantity and returns a probability, qnorm takes a probability and returns a quantity.

```
qnorm(0.025)
```

```
## [1] -1.959964
```

```
qnorm(0.975,lower.tail=FALSE)
```

```
## [1] -1.959964
```

$$Pr\left(-1.96 \leq \frac{\bar{X}-\mu}{(\sigma/\sqrt{n})} \leq 1.96\right) = 0.95$$

Now back to our statement for a general α and Z:

$$Pr\left(-Z \leq \frac{\bar{X}-\mu}{(\sigma/\sqrt{n})} \leq Z\right) = 1-\alpha$$

Suppose for the moment that we know the value of σ. Given that the only thing we **do not** know is the other population parameter μ, we can rearrange the inequalities inside the probability statement to deliver a probabilistic range where we think this parameter will reside.

$$Pr\left(-Z \leq \frac{\bar{X}-\mu}{(\sigma/\sqrt{n})} \leq Z\right) = 1-\alpha$$

$$Pr\left(-Z\frac{\sigma}{\sqrt{n}} \leq \bar{X}-\mu \leq Z\frac{\sigma}{\sqrt{n}}\right) = 1-\alpha$$

$$Pr\left(-\bar{X}-Z\frac{\sigma}{\sqrt{n}} \leq -\mu \leq -\bar{X}+Z\frac{\sigma}{\sqrt{n}}\right) = 1-\alpha$$

[2]This explains the 1.96 used in the dice application above.

$$Pr\left(\bar{X} - Z\frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{X} + Z\frac{\sigma}{\sqrt{n}}\right) = 1 - \alpha$$

This statement is a **confidence interval**, which can be written concisely as

$$\bar{X} - Z_{\frac{\alpha}{2}}\frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{X} + Z_{\frac{\alpha}{2}}\frac{\sigma}{\sqrt{n}}$$

or

$$\bar{X} \pm Z_{\frac{\alpha}{2}}\frac{\sigma}{\sqrt{n}}$$

It explicitly states that given the characteristics of the sample $(\bar{X}, n, \sigma)$ and an arbitrary level of confidence that gives us the probability limits from the standard normal distribution $(Z_{\frac{\sigma}{2}})$, then we can build a range of values where we can state with $(1-\alpha)\times 100$ confidence that the population parameter resides within.

Welcome to statistical inference!

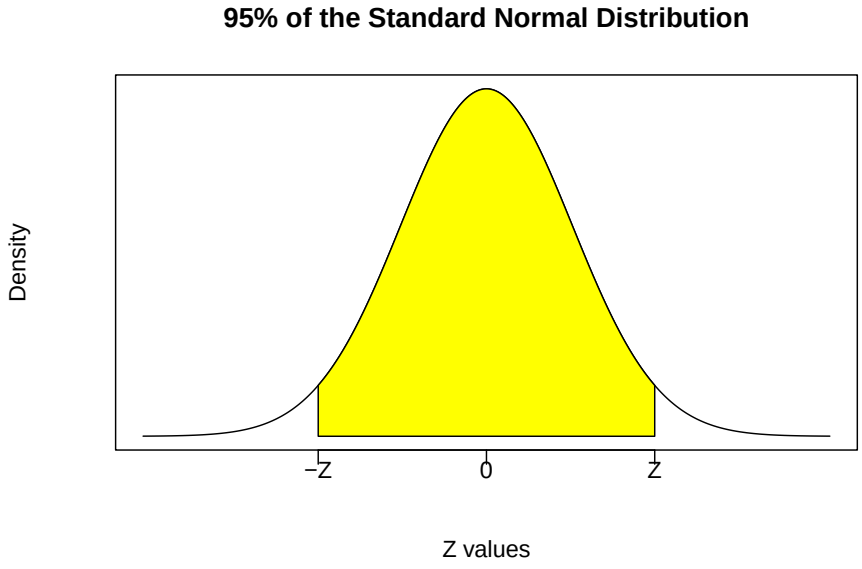
5.2.1 Application 3

A paper manufacturer produces paper expected to have a mean length of 11 inches, and a known standard deviation of 0.02 inch. A sample of 100 sheets is selected to determine if the production process is still adhering to this length. If it isn't, then the machine needs to go through the costs of being taken off line and recalibrated. The sample was calculated to have a average value of 10.998 inches.

$$\bar{X} = 10.998, \quad n = 100, \quad \sigma = 0.02$$

Construct a 95% confidence interval around the average length of a sheet of paper in the population.

1. Since we want 95% confidence, we know that $\alpha = 0.05$ and we need the critical values from a standard normal distribution such that 95% of the probability distribution is between them. These critical values were calculated previously to -1.96 and 1.96 and are illustrated below. Note that since the shaded region is 95% of the *central* area of the distribution, we are chopping of 5% of the *total* area from both tails combined. That means 2.5% is chopped off of each tail.



2. Now using the *positive* critical Z value in our confidence interval equation, we have:

$$\bar{X} \pm Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}$$

$$10.998 \pm 1.96 \frac{0.02}{\sqrt{100}}$$

Using R for the calculations:

```
Xbar = 10.998
n = 100
Sig = 0.02
alpha = 0.05
Z = qnorm(alpha/2,lower.tail = FALSE)
(left = Xbar - Z * Sig / sqrt(n))
```

```
## [1] 10.99408
```

```
(right = Xbar + Z * Sig / sqrt(n))
```

```
## [1] 11.00192
```

$$10.99408 \leq \mu \leq 11.00192$$

Conclusion: I am 95% confident that the mean paper length in the population is somewhere between 10.99408 and 11.00192 inches.

Note that any value within this range is equally likely!

5.2.2 What if we want to change confidence?

If we want to increase the confidence of our statement to 99% or lower it 90%, then we change α and calculate a new critical Z value. Everything else stays the same.

```
alpha = 0.01 # increase confidence to 99%
Z = qnorm(alpha/2,lower.tail = FALSE)
(left99 = Xbar - Z * Sig / sqrt(n))
```

```
## [1] 10.99285
```

```
(right99 = Xbar + Z * Sig / sqrt(n))
```

```
## [1] 11.00315
```

```
alpha = 0.10 # decrease confidence to 90%
Z = qnorm(alpha/2,lower.tail = FALSE)
(left90 = Xbar - Z * Sig / sqrt(n))
```

```
## [1] 10.99471
```

```
(right90 = Xbar + Z * Sig / sqrt(n))
```

```
## [1] 11.00129
```

5.2.3 What happens to the size of the confidence interval when we increase our *confidence*?

Our three previous conclusions are as follows:

> I am 90% confident that the mean paper length in the population is somewhere between 10.9947103 and 11.0012897 inches.
>
> I am 95% confident that the mean paper length in the population is somewhere between 10.9940801 and 11.0019199 inches.
>
> I am 95% confident that the mean paper length in the population is somewhere between 10.9928483 and 11.0031517 inches.

Notice that as our (arbitrary) level of confidence gets larger, the range of our expected population value gets wider. This illustrates a very important distinction in the world of inferential statistics between *confidence* and *precision*. If you want me to make a statement with a higher level confidence (i.e., a higher probability of being correct), then I will sacrifice the precision of my answer and

simply increase the range of possible values. Suffice it to say, we can increase our confidence up to the point where our confidence interval becomes useless.

$$Pr(-\infty \leq \mu \leq \infty) = 1.00$$

I am 100% confident that the mean paper length in the population is somewhere between $-\infty$ and ∞ inches.

5.3 What to do when we do not know σ

In most instances, if we know nothing about the population parameter μ then we know nothing about any of the other parameters (like σ). In this case, we are forced to use our best guess of σ. Since we are assuming that our sample has the same characteristics of the population, then our best guess for σ is the sample standard deviation S.

Put plainly, we substitute the statistic (S) for the population parameter (σ). Because S is an estimate of σ, this will slightly change our probability distribution. In particular, If $\bar{X}$ is normally distributed as per the CLT, then a standardization using S instead of σ is said to have a Student's t distribution with $n-1$ degrees of freedom.

$$t = \frac{\bar{X} - \mu}{(S/\sqrt{n})}$$

Note that this looks almost exactly like our Z transformation, only with S replaced for σ, and t replaced for Z. However, this statistic is said to be drawn from a distribution with $n-1$ degrees of freedom. We mentioned degrees of freedom before, and we stated that we lose a degree of freedom when we build statistics on top of each other. In other words, we lose a degree of freedom for every statistic we use to calculate another statistic. Consider the standard deviation equation needed to calculate S.

$$S = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2}$$

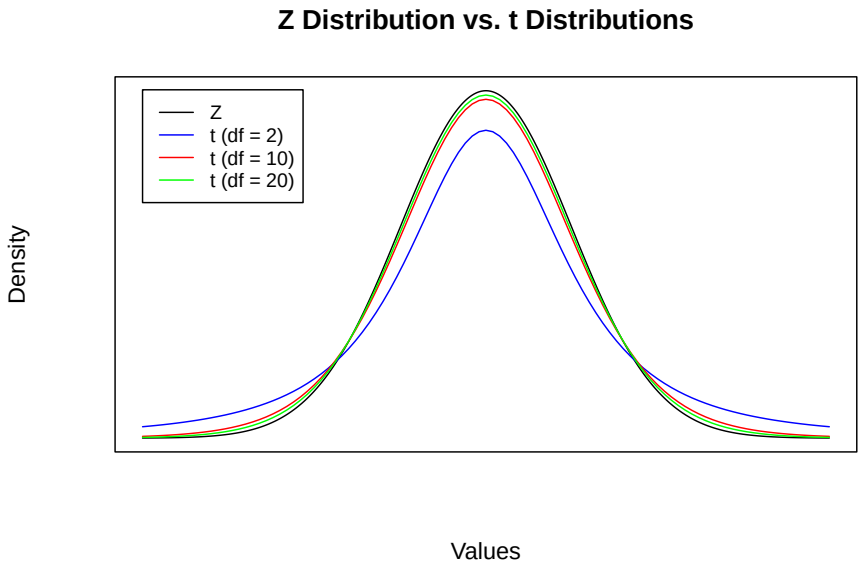
The equation states that the sample mean ($\bar{X}$) is used to calculate the sample standard deviation. Therefore one statistic is used to calculate a subsequent statistic… and that is why we lose one degree of freedom.

5.3.1 Student's t distribution versus Z distribution…

A Student's t distribution (henceforth, t distribution) and Z distribution have very much in common: they are both symmetric, both centered at a mean of

0, and both sum to one (because they are both probability distributions). The main difference is that a t-distribution has *fatter tails* than a Z distribution, and the fatness of the tails depends on the degrees of freedom (which in turn depends on the sample size n).[3]

The figure below compares the standard normal (Z) distribution with several t distributions that differ in degrees of freedom. Notice that tail thickness of the t distributions are inversely related to sample size. As the the degrees of freedom get larger (i.e., the larger the sample size), the closer the t distribution gets to the Z distribution. This is because as n gets larger, S becomes a better estimate of σ.



$$\bar{X} - Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{X} + Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}$$

$$\bar{X} - t_{(\frac{\alpha}{2}, df=n-1)} \frac{S}{\sqrt{n}} \leq \mu \leq \bar{X} + t_{(\frac{\alpha}{2}, df=n-1)} \frac{S}{\sqrt{n}}$$

In a nutshell, the only difference encountered when not knowing σ is that we have a slightly different probability distribution (which requires knowing the degrees of freedom and uses a different R command). The new R commands are `qt` and `pt` which requires degrees of freedom but otherwise has all of the same properties of `qnorm` and `pnorm` discussed above.

[3] Next time you raise a pint of Guiness, you can thank William Sealy Gosset for being *Student* and creating the t distribution. You can read more here

```
pt(q, df, lower.tail = TRUE)
qt(p, df, lower.tail = TRUE)
```

5.3.2 Application 4

Suppose you manage a call center and just received a call from Quality Control asking for the *average call length* at your facility. They are asking for the average call length in the population, so the best you can do is provide a confidence interval around this population parameter. You select a random sample of 50 calls from your facility and calculate a sample average of 5.8 minutes and a **sample standard deviation** of 2.815 minutes.

$$\bar{X} = 5.8, \quad n = 50, \quad S = 2.815$$

Calculate a 95% confidence interval around the population average call length.

```
Xbar = 5.8
n = 50
df = n-1
S = 2.815
alpha = 0.05
t = qt(alpha/2,df,lower.tail = FALSE)
(left = Xbar - t * S / sqrt(n))
```

```
## [1] 4.999986
```

```
(right = Xbar + t * S / sqrt(n))
```

```
## [1] 6.600014
```

With 95% confidence, the population average call length is between 5 minutes and 6.6 minutes.

As before, if we want to change our level of confidence then we change α and recalculate the t statistic. Notice that the relationship remains that a lower confidence level delivers a narrower confidence interval.

```
alpha = 0.01 # increase confidence to 99%
t = qt(alpha/2,df,lower.tail = FALSE)
(left = Xbar - t * S / sqrt(n))
```

```
## [1] 4.733108
```

```
(right = Xbar + t * S / sqrt(n))
```

```
## [1] 6.866892
```

```
alpha = 0.10 # decrease confidence to 90%
t = qt(alpha/2,df,lower.tail = FALSE)
(left = Xbar - t * S / sqrt(n))
```

```
## [1] 5.132563
```

```
(right = Xbar + t * S / sqrt(n))
```

```
## [1] 6.467437
```

5.4 Determining Sample Size

It was previously stated that the sample size should always be as big as possible in order to deliver the most *precise* conclusions. This isn't always a satisfactory answer, because collecting observations might be possible (but costly).

How big should n be?

Selecting an appropriate sample size could be determined by many constraints

- budget, time, … (things that cannot really be dealt with statistically)
- *acceptable* sampling error (we can deal with this)

Recall our confidence interval equation:

$$\bar{X} - Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{X} + Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}$$

or

$$\bar{X} \pm Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}$$

The term $Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}$ is one-half the width of the confidence interval. This is called the *sampling error* (or *margin of error*).

$$e = \pm Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}$$

In our previous exercises, we were given a sample size (n) and used our calculations to determine the width of the confidence interval ($2e$). If we instead wanted to fix the margin of error, then we can let the above identify determine how big our sample size needs to be.

$$n = \left(\frac{Z_{\frac{\alpha}{2}} \sigma}{e} \right)^2$$

Going back to our call center example, suppose that quality control demanded a 95% confidence interval with a 15 second (0.25 minute) margin of error. This means that the 95% confidence interval can only be 0.5 minutes wide. How many calls need to be in the sample?

```
alpha = 0.05
Z = qnorm(alpha/2,lower.tail = FALSE)
Xbar = 5.8
Sig = 2.815
e = 0.25

(n = (Z*Sig/e)^2)
```

```
## [1] 487.0493
```

```
# Round up since you can't have a fraction of an observation
ceiling(n)
```

```
## [1] 488
```

Our analysis indicates that if you want this particular a margin of error, then you will need to collect a sample of 488 calls.

You might have noticed that we did something a bit incorrect in the last exercise. We specified a Z distribution and called the sample standard deviation σ. Note that only in these sort of applications that determine a sample size is this permissible. The reason is because a sample standard deviation obviously depends on the sample in question. We therefore need to assume that the standard deviation is fixed when calculating the sample size (even though this isn't the case). Once you determine a sample size, then you collect a sample, calculate the sample standard deviation, and calculate the appropriate confidence interval. The margin of error should be reasonably close to what was required.

5.5 Concluding Applications

5.5.1 Light Bulbs (Last Time)

Let's go back one last time to our light bulb example

```
load("C:/Data/MBA8350/Lightbulb.Rdata")
(n = length(Lifetime))
```

```
## [1] 60
```

```
(Xbar = mean(Lifetime))
```

```
## [1] 907.5552
```

```
(S = sd(Lifetime))
```

```
## [1] 78.96741
```

We have the following information from our sample:

$$\bar{X} = 907.6, \quad n = 60, \quad S = 78.967$$

Use the above information to calculate a 95% confidence interval around the population average lifespan of the light bulbs you have left to sell. You can put this on information on the box!

```
alpha = 0.05
df = n-1
t = -qt(alpha/2,df)

(left = Xbar - t * S / sqrt(n))
```

```
## [1] 887.1557
```

```
(right = Xbar + t * S / sqrt(n))
```

```
## [1] 927.9546
```

5.5.2 Returning to the Philadelphia School Policy Application

Let us return to the Philadelphia school policy example to provide one final discussion of a confidence interval. This application may appear redundant, but it is intended to provide an alternative approach to the confidence interval concept. It has helped students in the past, so it might do some good.

In early February 2020, the city of Philadelphia considered sending out $100 EBT cards to every student registered in public school due to the school closings brought on by the pandemic.

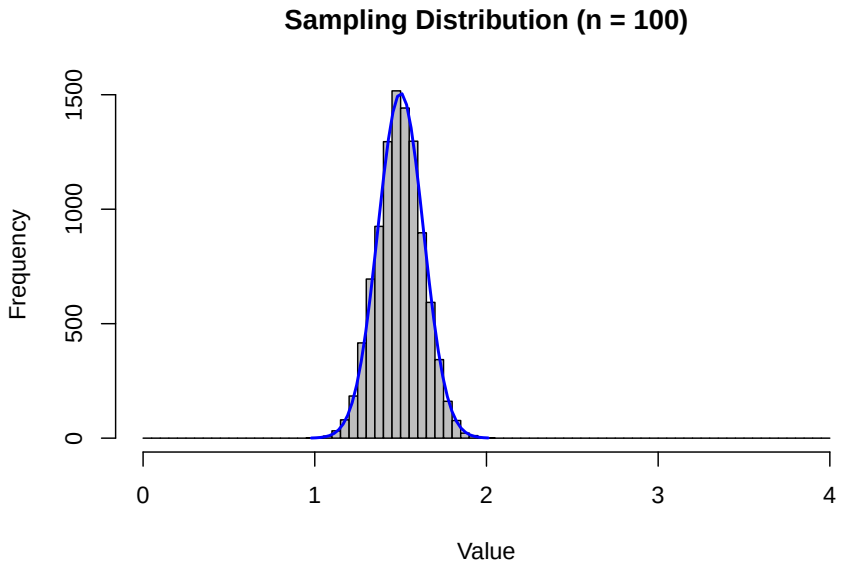
How much would this policy cost?

The Frame

There are 352,272 families in Philadelphia, and the city has records on how many students are registered in public schools.

However, suppose it is too costly (at this stage) to determine the total number of children registered in public schools. If we knew the average number of children registered per family, we can get an estimate of the cost of the policy.

Since it is too costly to examine the entire population (at the moment), we draw a single sample and use the sample to calculate sample statistics. Since the sample is randomly drawn from the population, the sample statistics are randomly drawn from a sampling distribution.



Your Sample

Once you determine a sample size (n), you get **one random draw** from the appropriate sampling distribution.

- The distribution is approximately *normal*
- The mean is μ
- The standard deviation $\sigma/\sqrt{n}$

We use this information and our sample characteristics to say something about the population parameters…

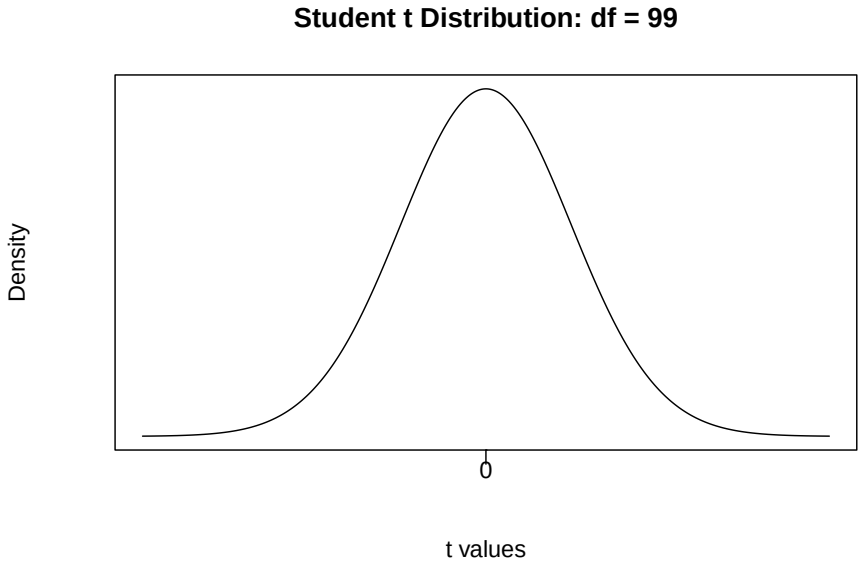
Suppose you select a sample of $n = 100$ families and calculate

$$\bar{X} = 1.7$$

$$S = 1.5$$

Since we have an *estimate* of the population standard deviation from our sample, our sampling distribution is now a t distribution with $n - 1 = 99$ degrees of freedom.

$$t = \frac{\bar{X} - \mu}{S/\sqrt{n}} = \frac{1.7 - \mu}{1.5/\sqrt{100}}$$



What we know...

- CLT: the true population average is the central point of our sampling distribution

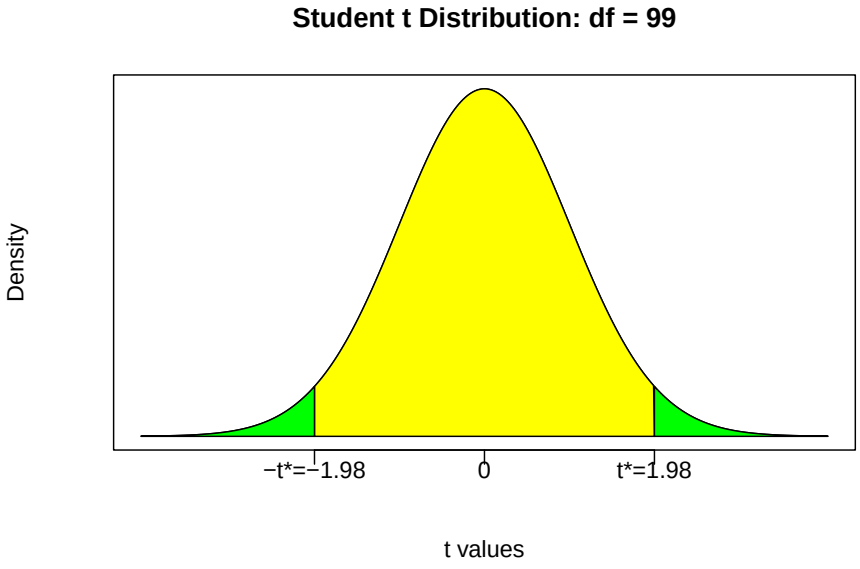
- We can choose an *arbitrary* level of confidence $(1 - \alpha)$ to limit where we think our statistic from a single draw will fall.

$$Pr(-t^* \leq \frac{1.7 - \mu}{1.5/\sqrt{100}} \leq t^*) = 1 - \alpha$$

Suppose we want 95% confidence ($\alpha = 0.05$)

```
(tcrit <- qt(0.05/2,99))
```

```
## [1] -1.984217
```



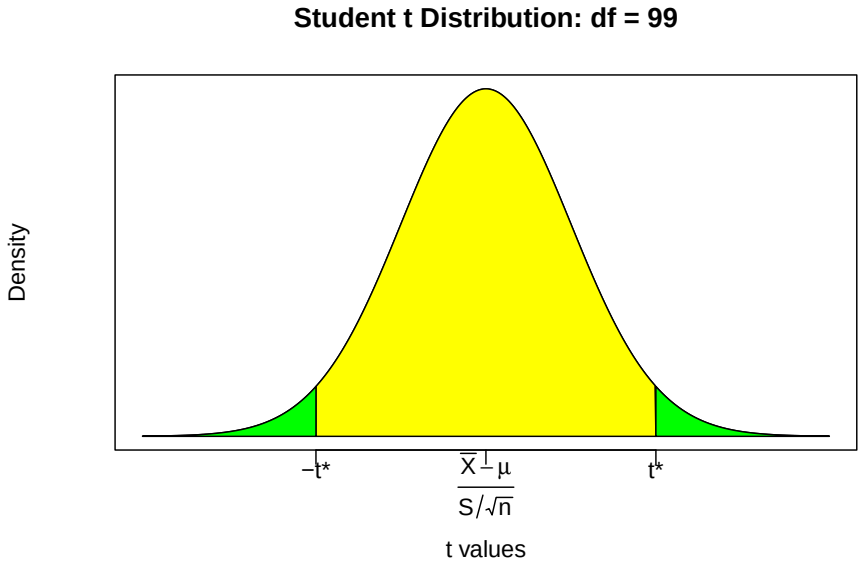
What we DON'T know...

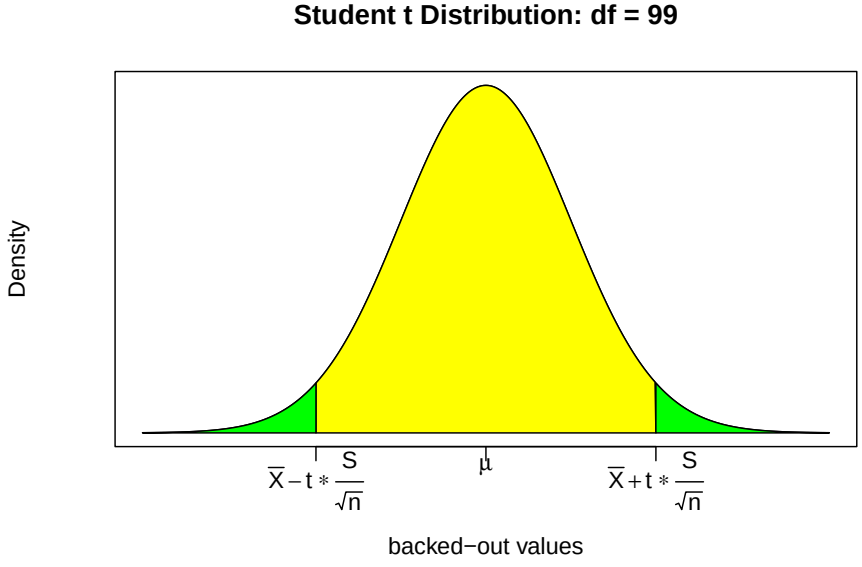
- We don't know the numerical value of μ...
- We don't know where our value of $\bar{X}$ falls in relation to μ
  - $\bar{X} = \mu$?
  - $\bar{X} > \mu$?
  - $\bar{X} < \mu$?

The fact that we don't know where $\bar{X}$ is in relation to μ is why we end up with an *interval* around where we think the population parameter resides.

$$Pr(\bar{X} - t^* \frac{S}{\sqrt{n}} \leq \mu \leq \bar{X} + t^* \frac{S}{\sqrt{n}}) = 1 - \alpha$$

$$Pr(1.7 - 1.98 \frac{1.5}{\sqrt{100}} \leq \mu \leq 1.7 + 1.98 \frac{1.5}{\sqrt{100}}) = 0.95$$






```
Xbar = 1.7; S = 1.5; n = 100; AL = 0.05
tcrit <- -qt(AL/2,n-1)

(LFT <- Xbar - tcrit * S / sqrt(n))
```

```
## [1] 1.402367
```

```
(RHT <- Xbar + tcrit * S / sqrt(n))
```

```
## [1] 1.997633
```

$$Pr(1.40 \leq \mu \leq 2.00) = 0.95$$

With 95% confidence, the average number of children per family is between 1.4 and 2.

Total cost is between…

$$1.4 \times \$100 \times 352,272 = \$49,318,080$$

and

$$2 \times \$100 \times 352,272 = \$70,454,400$$

Chapter 6

Hypothesis Tests

Confidence intervals determine a range where our population mean resides given the characteristics of a sample and a desired level of confidence. Recall that the population parameter can be *anywhere* within the range dictated by a confidence interval.

Hypothesis testing is a similar inferential method, but it approaches the problem from the opposite direction.

1. You start with an *unambiguous claim* on the value of the population parameter. This claim is nonarbitrary, and dictated from either theory or a past observation.

2. You test to see if the sample statistics are consistent with the claim (or refute it)

The general idea is that you begin with some *nonarbitrary* statement on what value you believe (or do not believe) the population parameter to be. You then test if the characteristics of your sample suggest that it is likely or not that a population with your proposed parameter values generated a sample similar to the one you currently have.

If this seems a bit vague at the moment, it will hopefully be more concrete soon. The main thing to keep in mind is that hypothesis tests are quite simple and structured. Once you learn how to perform one hypothesis test - you can essentially perform them all. This chapter guides you through some basic steps that once mastered - you'll have a powerful tool of statistical inference under your belt.

6.1 Anatomy of a Hypothesis Test

A hypothesis test begins with a claim about the value of particular a population parameter.

This statement takes the form of a **null hypothesis**

$$H_0 : \mu = x$$

This statement gets contrasted against an **alternative hypothesis**

$$H_1 : \mu \neq x$$

The null hypothesis (H_0) represents a belief of a population parameter that you would like to *disprove*, while the alternative hypothesis (H_1) is the opposite of the null and represents a claim you would like to show.

A hypothesis test uses the characteristics of the sample to determine if the statement about the population parameter in the null appears consistent (or inconsistent) with the characteristics of the sample. Recall that we are still under the assumption that the characteristics of the sample are similar to the **true** characteristics of the population. Therefore, if the sample characteristics are inconsistent with the statement in the null hypothesis, then you are likely to **reject the null hypothesis**. This means that the null hypothesis does not capture the **true** characteristics of the population (because the sample does). If the sample characteristics are *similar* to those stated in the null hypothesis, then you do not have evidence to reject the null and you conclude to **not reject the null hypothesis**.

In other words, if you *reject the null*, you have statistical evidence that H_1 is correct (and the null hypothesis cannot be correct). If you *do not reject the null*, you have failed to prove the alternative hypothesis. Note that failure to prove the alternative does NOT mean that you have proven the null. In other words, there IS a difference between *do not reject* and *accept*!

This distinction between *do not reject* and *accept* cannot be emphasized enough. First, if anyone concludes that they accept the null in this class - you will get marked incorrect. If you conclude to accept the null outside of this class - then people will suspect that you don't fully understand what you are talking about. Second, we can never say accept the null because it is simply too strong of a statement to make regarding a population parameter.

Suppose we believe that the population mean life span of our light bulbs is x hours. A hypothesis test will give us a specific way of testing this belief, and allows us to conclude whether or not this statement is consistent with our sample.

$$H_0 : \mu = x \quad versus \quad H_1 : \mu \neq x$$

We will discuss how to formally conduct a hypothesis tests in a bit. For now, lets compare these hypotheses with the confidence interval we calculated in the previous section.

$$887 \leq \mu \leq 928$$

Recall that our confidence interval states that with 95% confidence, the population average life span of the light bulbs is *somewhere* between 887 hours and 928 hours - meaning that any value within this range is equally likely. It also states that there is only a 5% chance that the population parameter lies outside of this range.

If we were to test that the population average lifespan was 1000 hours,

$$H_0 : \mu = 1000 \quad versus \quad H_1 : \mu \neq 1000$$

then our confidence interval would give us evidence to **reject** the null because there would be less than a 5% chance for the null to be true. The sample characteristics and the statement in the null hypothesis are therefore inconsistent.

If we were to test that the population average lifespan was 900 hours, then we will reach a different conclusion.

$$H_0 : \mu = 900 \quad versus \quad H_1 : \mu \neq 900$$

Since 900 is a value *inside* our confidence interval, then we would not have evidence to reject the null and we therefore conclude **do not reject** the null. The reason why we never say accept is that while 900 is within the confidence interval, there are also a *continuum* of other values in there. The true population mean might be 901, 900.0001, 910, etc. If you were to *accept* the null, then you are explicitly stating that the population parameter is exactly 900 - we do not have enough evidence for this.

Note that while we are seeing a clear connection between hypothesis tests and confidence intervals, hypothesis tests can get more sophisticated than this. It is therefore worthwhile to consider a formal solution methodology.

6.2 Steps to a hypothesis test

A hypothesis test is a simple, step-by-step procedure that can be very powerful. The sections below will provide details to performing hypothesis tests, and every application will follow four basic steps.

1. State the **null** and **alternative** hypotheses
2. Calculate a **test statistic under the null**

3. Calculate either the **rejection region(s)** or the **p-value**
4. Use the information in steps 2 and 3 to **conclude**: *Reject* or *Do Not Reject*

You will notice that step 3 offers two equivalent ways of conducting a hypothesis test. We will discuss them in turn.

6.3 Two methods for conducting a hypothesis test (when σ is known)

We will consider two equivalent methods for conducting a hypothesis test. The first is called the *rejection region* method and is very similar to confidence intervals. The second is the *p-value* method and delivers some very useful results that we will come in handy for almost every application of inferential statistics. We will consider these methods in turn.

6.3.1 Rejection Region Method

All hypothesis tests start with a statement of the null and alternative hypotheses (Step 1). The null makes an explicit statement regarding the value of a population parameter (μ). Once this value of μ is established under the null, all hypothesis tests construct a *test statistic under the null* (Step 2). In particular, assuming we know the population standard deviation σ, we can construct a Z *statistic under the null* using our familiar z-transformation:

$$Z = \frac{\bar{X} - \mu}{(\sigma/\sqrt{n})}$$

Note that we already know values from the sample ($\bar{X},\ n,\ \sigma$) and we have a *hypothesized* value of μ from the null hypothesis. We can therefore directly calculate this Z-statistic *under the null*. This would be the value of a Z-statistic given our sample characteristics under the *assumption* that our null hypothesis is **correct**.

The rejection region method takes this Z-statistic under the null and sees where it falls in a standard normal sampling distribution. The sampling distribution of a test statistic is first divided into two regions…

1. A region of rejection (a.k.a., critical region): values of the test statistic that are unlikely to occur **if the null hypothesis is true**. These values are inconsistent with the null hypothesis.
2. A region of nonrejection: values of the test statistic that are likely to occur if the null hypothesis is true. These values are *consistent with the null hypothesis*.

The regions of rejection and nonrejection are identified by determining *critical values* from the normal probability distribution. These are particular values of the sampling distribution that divides the entire distribution into rejection and nonrejection regions. This is why some people refer to the *rejection region* approach as the *critical value* approach.

Once the different regions are established, then you simply see where the calculated test statistic under the null falls and conclude (Step 4). If it falls inside the rejection region, then you **reject the null** because the characteristics of the sample are *too inconsistent* with the population parameter stated inside the null hypothesis. If it falls inside the nonrejection region, then you **do not reject the null** because the characteristics of the sample are such that the population parameter stated inside the null is *possible* (but we can't say if it's necessarily true).

The Steps of a Hypothesis Test (Rejection Region Method)

1. State the null and alternative hypotheses
2. Calculate a test statistic under the null
3. Determine the rejection and nonrejection regions of a standardized sampling distribution
4. Conclude (reject or do not reject)

Application 1

Suppose a fast-food manager wants to determine whether the waiting time to place an order has changed from the previous mean of 4.5 minutes. We want to see if our sample characteristics are consistent with an average of 4.5 minutes or not. We start with a statement of the two hypotheses.

$$H_0 : \mu = 4.5 \quad versus \quad H_1 : \mu \neq 4.5$$

The null explicitly states that $\mu = 4.5$, so we can use this value to construct a test statistic using the information from our sample. Suppose that a sample of $n = 25$ observations delivered a sample mean of $\bar{X} = 5.1$ minutes. Suppose further that we know the population standard deviation of the wait time process to be $\sigma = 1.2$. We can calculate a test statistic under the null.

```
mu = 4.5
Xbar = 5.1
Sig = 1.2
n = 25

(Zstat = (Xbar - mu)/(Sig/sqrt(n)))
```

```
## [1] 2.5
```

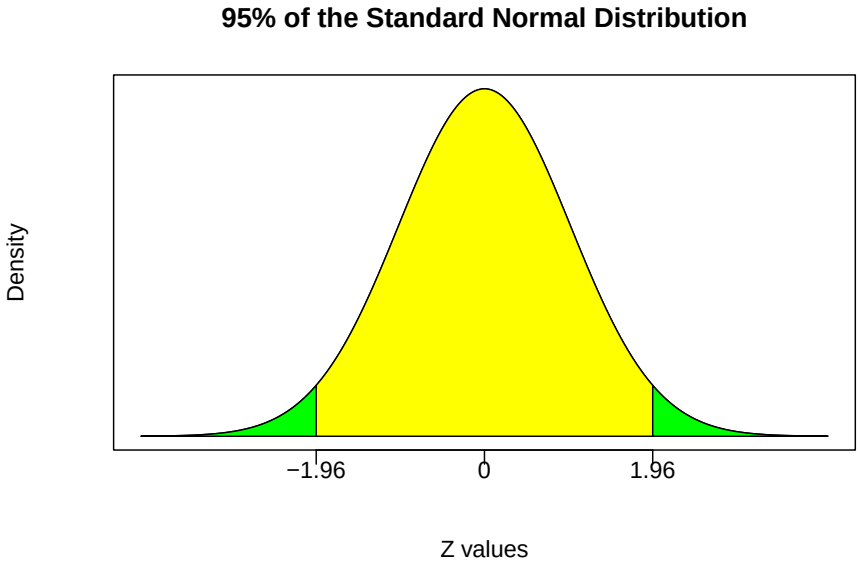
$$Z = \frac{\bar{X} - \mu}{(\sigma/\sqrt{n})} = \frac{5.1 - 4.5}{\left(1.2/\sqrt{25}\right)} = 2.5$$

The next step involves taking a standard normal sampling distribution and breaking it up into regions of rejection and nonrejection. Before we do this, lets take a step back and think about what it means for a sample to be consistent or inconsistent with the null hypothesis. If you had a sample average ($\bar{X}$) that was the same value as the parameter value stated in the null hypothesis (μ) then you could state that the value of μ in the null hypothesis is *very consistent* with the characteristics of the sample. In fact, you can't be more consistent than having $\bar{X}$ and μ being the exact same number. Furthermore, a test statistic under the null would be equal to zero because $(\bar{X} - \mu)$ is in the numerator. This implies that a test statistic under the null equal to zero is very consistent with the null hypothesis, and you will therefore *not reject* the null hypothesis. However, the farther away the test statistic under the null gets from zero, the farther away it gets from the center of the *do not reject region* and the closer it gets to one of the *rejection regions*.

Lets illustrate this supposing that we want to test the hypothesis at the 95% confidence level ($\alpha = 0.05$). The central 95% of a standard normal distribution is given by

$$Pr(-1.96 \leq Z \leq 1.96) = 0.95$$

This means that if you reached into a bowl of numbers comprising a standard normal distribution, then 95% of the time you will draw a number between -1.96 and 1.96. The remaining numbers outside of this range will show up only 5% of the time. These regions are the bases for confidence intervals and also the bases for the nonrejection and rejection regions.



The yellow-shaded region is centered on zero and represents the *nonrejection region*. It tells you that if you calculate a test statistic under the null to be between -1.96 and 1.96, then you do not have enough evidence to reject the null. However, the green-shaded regions are the *rejection regions*. It tells you that if you calculate a test statistic under the null that is greater than 1.96 or less than -1.96, then you have enough evidence to reject the null (with 95% confidence). In other words, it is so unlikely to have the null be correct while simultaneously randomly selecting a sample with the observed sample characteristics, so we conclude that the statement in the null cannot be true. In the fast food example above, the test statistic under the null of 2.5 falls in the rejection region. This means that we can *reject* the null hypothesis of $\mu = 4.5$ minutes with 95% confidence. In other words, we are 95% confident that the population mean is some number other than 4.5 minutes - the statement made in the alternative hypothesis.

Changing the level of confidence (α)

The hypothesis test above was concluded under a specified 95% confidence level ($\alpha = 0.05$). This level of confidence effectively delivered our rejection and nonrejection regions. So… what happens when we change α?

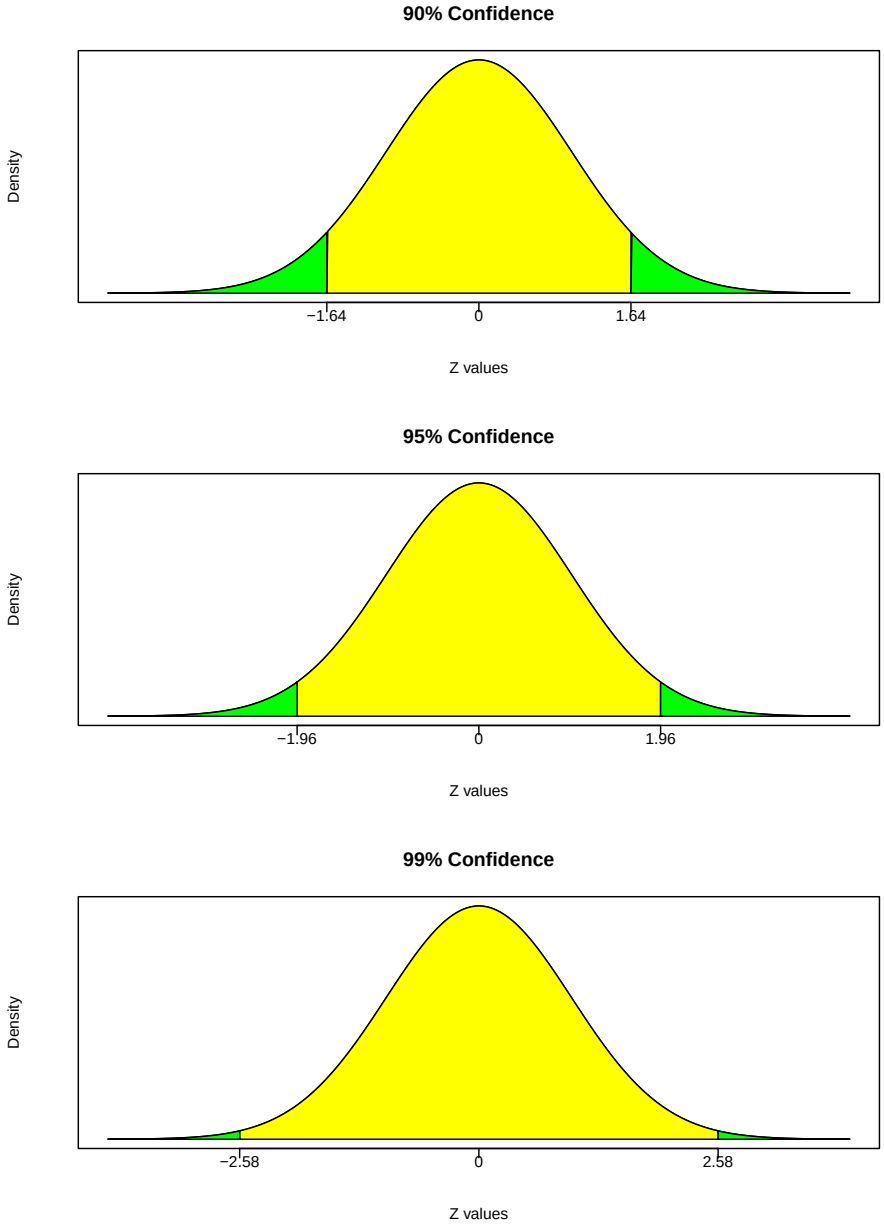
The first thing to understand is that the level of confidence does not impact the hypotheses or the test statistic under the null. The **only** thing the level of confidence impacts is the shaded regions in the sampling distribution. The figure below illustrates rejection and nonrejection regions for α values of 0.10, 0.05, and 0.01. Note that as α gets smaller, the size of the nonrejection region

gets larger. This means that *do not reject* is becoming a more likely conclusion. This should make sense because *do not reject* is a wishy-washy conclusion, while *reject* is definitive. Do not reject states that the null may or may not be true. It's a punt! A *"Maybe, maybe not"* answer! Therefore, if you are placing more confidence on your conclusion, the more likely you are to make the wishy-washy conclusion.

The fast food example had a test statistic under the null of 2.5. This test statistic falls in the rejection region for both 90% and 95% levels of confidence. This suggests that if you can reject a null hypothesis with a certain level of confidence, then you can automatically reject at all lower levels of confidence. However, the do not reject region under 99% confidence is given by

$$Pr(-2.58 \leq Z \leq 2.58) = 0.99$$

and our test statistic of 2.5 falls inside it. We therefore conclude that we *do NOT reject* the null with 99% confidence. In other words, we do not have enough evidence to say that the null hypothesis is incorrect at 99% confidence, so we conclude that it *may or may not* be true (i.e., we punt) at this confidence level.



Using the rejection region method, we were able to reject the null with 95% confidence ($\alpha = 0.05$) but unable to reject with 99% confidence ($\alpha = 0.01$). This begs the question as to the highest confidence level at which we can reject the null. We know it is some confidence level between 95% and 99%, but what is it exactly? We can use the rejection region approach multiple times by choosing

various values of α and narrow things down, or we can conduct the hypothesis test using the *p-value* approach.

6.3.2 P-value Approach

The P-value is an extremely useful and often misunderstood number. I therefore have THREE equivalent ways of explaining it. Each one works - so just stick with the one that works for you. Before we get to those, lets talk explicitly about what we mean when we make statements based on confidence.

When using a sample statistic to draw conclusions about a population parameter, there is always the risk of reaching an incorrect conclusion. In other words, you can make an **error**.

When making a conclusion about a hypothesis test, one can either reject a null hypothesis or not. Therefore, there are two possible types of errors to be made.

1. A **Type I error** occurs when a researcher incorrectly rejects a true hypothesis. (*You rejected something that shouldn't have been rejected.*)
2. A **Type II error** occurs when a researcher incorrectly fails to reject a false hypothesis. (*You did not reject something that should have been rejected.*)

The **acceptable** probability of committing either one of these errors depends upon an arbitrary confidence level α. To be precise, when you reject a hypothesis with 95% confidence, then you are implicitly stating that you are accepting a 5% chance of being wrong. That is where $\alpha = 0.05$ (or 5% comes from). If you decrease α to 0.01 (or 1%), then you can reject a hypothesis with 99% confidence and implicitly accept a 1% chance of being wrong. The kicker is that the more you decrease the probability of committing a type I error, the more you increase the chance of not rejecting a hypothesis that you should be rejecting (a type II error). For example, if you want a conclusion with 100% confidence, then you will *never* reject a hypothesis no matter how wrong it actually is.[1]

The main take away from the previous statement is that α states the *acceptable* probability of committing a type one error. Recall in our fast food example that we rejected the null hypothesis with 95% confidence (i.e., a 5% acceptable probability of being wrong), but we did not reject the null hypothesis with 99% confidence (i.e., a 1% acceptable probability of being wrong). This means that the *actual* probability of committing a type one error is somewhere in between 0.05 and 0.01 (i.e., 5% and 1%). This actual probability of committing a type I error is called the **p-value**.

```
# Fast Food Example Revisited
mu = 4.5
Xbar = 5.1
```

[1] This point touches on the idea of confidence in statistics. If you want me to make a statement with 100% confidence, then I'll simply say *anything can happen* because it is a statement that has zero chance of being wrong.

```
Sig = 1.2
n = 25

(Zstat = (Xbar - mu)/(Sig/sqrt(n)))
```

```
## [1] 2.5
```

```
# 95% confidence:
alpha = 0.05
(Zcrit = qnorm(alpha/2,lower.tail = FALSE))
```

```
## [1] 1.959964
```

```
# 99% confidence:
alpha = 0.01
(Zcrit = qnorm(alpha/2,lower.tail = FALSE))
```

```
## [1] 2.575829
```

```
# p-value:
(Pval = pnorm(Zstat,lower.tail = FALSE)*2)
```

```
## [1] 0.01241933
```

```
# Actual confidence level:
((1-Pval)*100)
```

```
## [1] 98.75807
```

The calculations regarding the fast food example were repeated and continued to include a p-value. Recall that the null hypothesis stated that the population mean was equal to 4.5, and the test statistic under the null is equal to 2.5. The critical values marking the boundaries between the do not reject region and the reject regions was ± 1.96 for $\alpha = 0.05$ and ± 2.58 for $\alpha = 0.01$. Our test statistic falls inside the rejection region for $\alpha = 0.05$ and inside the nonrejection region for $\alpha = 0.01$. Our test statistic falls *right on the boundary* of a rejection and nonrejection region when $p = 0.0124$. This is the p-value of the problem. It states that you can reject the null hypothesis with *at most* 98.76% confidence and you will incur a 1.24% chance of being wrong. As expected, it is between 5% and 1% and gives you a tailor-made confidence level for the hypothesis test at hand.

The definitions of a P-value

> The p-value is the probability of getting a test statistic equal to or more extreme than the sample result, given that the null hypothesis (H_0) is true.

While this is the technical definition of a p-value, it is a bit vague. There are some roughly equivalent definitions that might be easier to digest.

> The p-value is the probability of committing a Type I error. If the p-value is greater than some arbitrarily given α, then you cannot reject the null.
>
> The p-value is the probability that your null hypothesis is *correct*. The HIGHEST level of confidence at which you can reject the null is therefore $1 - p$.

These two definitions essentially state the same thing from two different perspectives. Suppose you calculated a p-value to a hypothesis and determined the value to be 0.08. This means that if you reject the null, you have an 8% chance of being wrong (i.e., committing a type I error). This also means that if you reject the null, you have a 92% chance of being correct. That is why you can reject the null with 92% confidence.

6.4 Two-sided vs One-sided Test

$$H_0 : \mu = 4.5 \quad versus \quad H_1 : \mu \neq 4.5$$

The hypothesis test considered above is known as a **two-sided** test because the null gets rejected if the mean of the sample is either significantly greater than or less than the value stated in the null hypothesis. If you notice from the illustrations above, a two-sided test has **TWO** rejection regions - one in each tail (hence the name). Note that this is also why we calculated critical values using half of the value of α and doubled the calculated probability value in order to arrive at a p-value.

In the fast food example above, suppose we want to show that the service time *increased*. In other words, we want statistical evidence that the wait time actually increased from a previous time of 4.5 minutes. We can provide statistical evidence by rejecting a null hypothesis that the new population average wait time is 4.5 minutes *or less*. This scenario delivers us a **one-sided hypothesis test**.

$$H_0 : \mu \leq 4.5 \quad versus \quad H_1 : \mu > 4.5$$

As the name implies, a one-sided hypothesis test only has one rejection region. This means that the entire value of α is grouped into either the right or left tail. The tail containing the rejection region depends upon the exact specification of the hypothesis test.

The hypothesis test above is called a *right-tailed* test because the rejection region is in the right tail. To see this, consider several hypothetical sample averages and see if they are consistent with the null $\mu \leq 4.5$.

- Suppose you observe $\bar{X} = 4$. Is this sample average consistent with $\mu \leq 4.5$?
- What about $\bar{X} = 2$?
- What about $\bar{X} = 1$?

Your answer should be yes to all of these sample averages. In fact, *any* sample average less than or equal to 4.5 is consistent with $\mu \leq 4.5$.

Next, recall the test statistic under the null:

$$Z = \frac{\bar{X} - 4.5}{(\sigma/\sqrt{n})}$$

For any of the hypothetical values considered above (4, 2, or 1), the test statistic would be a negative number. Since we said that all of these sample averages are consistent with the null being true, then we would never reject the null in any of these instances. Therefore, the rejection region cannot be in the left tail because that is where the negative values of the distribution reside. The rejection region must therefore be in the right tail. Only when a sample average is sufficiently greater than 4.5 is when we can consider rejecting the null. Note this is true reasoning behind why the above test is a right-tail test. For a quick short cut approach - look at the alternative hypothesis. If the inequality points to the right (e.g., >), then you have yourself a right-tail test. If the inequality points to the left (e.g., <), then you have yourself a left-tail test. It's just that simple.

Now that we already have the null and alternative hypotheses down as well as the test statistic under the null, the next step is to determine the critical value that divides the distribution into rejection and nonrejection regions.

```
# Fast Food Example Revisited
mu = 4.5
Xbar = 5.1
Sig = 1.2
n = 25

(Zstat = (Xbar - mu)/(Sig/sqrt(n)))
```

```
## [1] 2.5
```

```
# 95% confidence:
alpha = 0.05
(Zcrit = qnorm(alpha,lower.tail=FALSE))
```

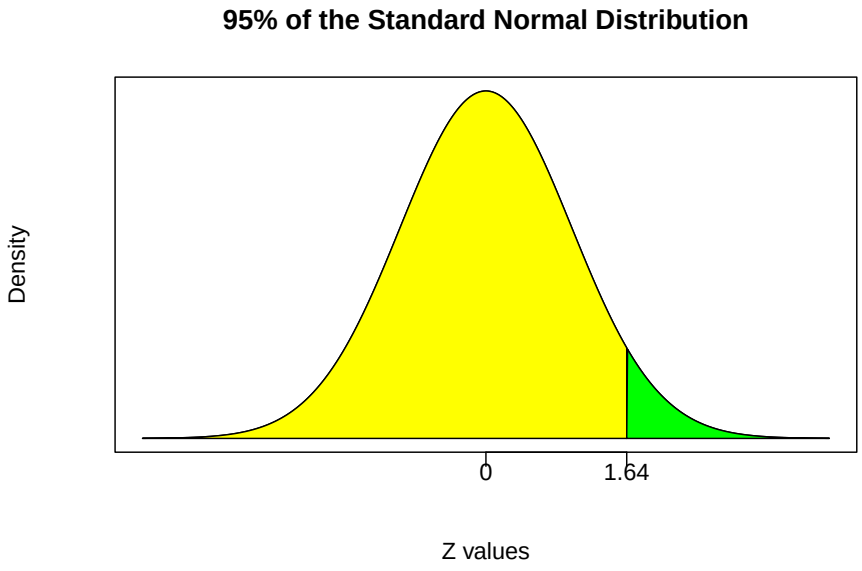
```
## [1] 1.644854
```

```
# P-value:
(Pval = pnorm(Zstat,lower.tail=FALSE))
```

```
## [1] 0.006209665
```

```
# highest Confidence Level for rejection:
((1-Pval)*100)
```

```
## [1] 99.37903
```



If we conducted this hypothesis test at the 95% confidence level ($ =0.05), you will see that the rejection region is the 5% of the curve in the right tail. That means you reject all test statistics greater than or equal to 1.64. Since our test statistic is 2.5, we can reject with 95% confidence. We can also conduct this hypothesis test using the p-value approach which delivers a p-value of 0.0062. This means that if we reject the null, we only incur a 0.62% chance of being wrong. This equivalently means that we can reject the null with up to 99.38% confidence.

6.5 Conducting a hypothesis test (when σ is unknown)

When the population standard deviation (σ) is unknown, it must be estimated. Just like with confidence intervals, When you replace σ with it's estimate S, you change the distribution from Z to t (and need to mind the degrees of freedom).

That's the only difference

Let's go through some applications when σ is unknown. You will see that the

only difference is that we use a t distribution with $n-1$ degrees of freedom to calculate rejection / nonrejection regions and p-values.

Application 2

The Saxon Home Improvement Co. has had a mean per sales invoice of $120 over the last 5 years and would like to know if the mean amount per sales invoice has significantly changed. This is enough information to state our hypotheses for a two-sided test.[2]

$$H_0 : \mu = 120 \quad versus \quad H_0 : \mu \neq 120$$

You collected a sample of 12 observations, and concluded that the sample mean was $112.85 and the sample standard deviation was $20.80.

$$\bar{X} = 112.85, \quad n = 12, \quad S = 20.80$$

This information allows us to calculate a t-test statistic under the null. The only difference is that we now have a sample standard deviation (S) were we once had a population standard deviation (σ).

```
Xbar = 112.85
n = 12
S = 20.80
mu = 120

(t = (Xbar - mu) / (S/sqrt(n)))
```

```
## [1] -1.190785
```

$$t = \frac{\bar{X} - \mu}{(S/\sqrt{n})} = \frac{112.85 - 120}{\left(20.80/\sqrt{12}\right)} = -1.19$$

Now that we have our test statistic, we need to determine if it falls into our nonrejection or rejection regions. The important thing to realize is that these regions are now part of a t distribution with 11 $(n-1)$ degrees of freedom. If we consider 95% confidence…

```
alpha = 0.05
(tcrit = qt(alpha/2,n-1,lower.tail=FALSE))
```

```
## [1] 2.200985
```

[2]Note the language - *significantly changed* means that the value could have either gone up or down. This is why it is a two-sided test.

The calculations suggest that the nonrejection region is between ± 2.2. Since our test statistic falls within this region, we do not reject the null. This implies that we do not have evidence that the population average sales invoice has significantly changed from \$120 with 95% confidence. The conclusion is therefore *do not reject*.

We could also calculate a p-value for the test:

```
(Pval = pt(t,n-1)*2)
```

```
## [1] 0.2588003
```

```
# Highest confidence interval for rejection:
((1-Pval)*100)
```

```
## [1] 74.11997
```

Notice here that the p-value states that if we were to reject the null, then we would incur a 25.88% chance of being wrong. This means that we could only reject the null with 74.12% confidence.

Note that the calculations uses a new R command: `pt(q,df)`. This command calculates the probability under a t distribution the same way the `pnorm(q)` command calculates the probability under a standard normal distribution. In addition, I again encourage you to always visualize the distribution and explicitly draw the rejection and nonrejection regions. This is extremely helpful when first getting started. Below you will also see a note I wrote for a previous class reinforcing how R likes to calculate probabilities. It is for reference if needed.

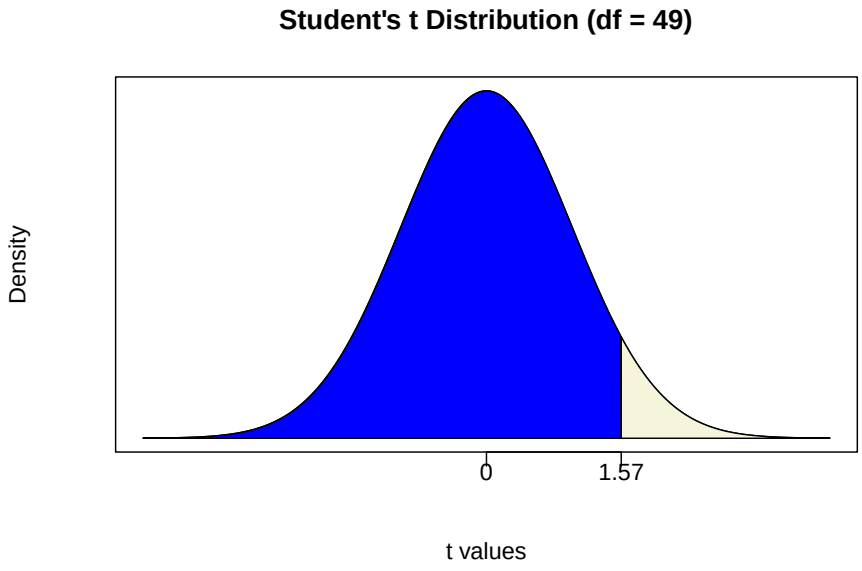
6.6 Appendix: A note on calculating P-values

6.6.1 The Problem

Suppose you were performing a right-tailed hypothesis test (using a t distribution) and you arrived at a test statistic under the null of 1.57. This means that the rejection region is in the right tail, and if you wished to calculate the p-value, then it would be the area of the curve to the right of 1.57.

If you have a sample of $n = 50$, then you would use a t-distribution with 49 or $(n - 1)$ degrees of freedom.

An illustration is below:



6.6.2 How to calculate p-values

In case you haven't noticed by now, R has a default way of calculating probability areas...

IT ALWAYS CALCULATES AREAS FROM THE LEFT!

In other words, the default is to give you the area to the left of a number...

```
(Pval = pt(1.57,49))
```

```
## [1] 0.9385746
```

Don't be annoyed about this, because all software does this (including Excel).

We can use this default to calculate the p-value (i.e. the area to the right of 1.57) in THREE different ways by relying on two properties of our probability distributions.

Property 1: The distribution is centered at zero and symmetric.

This means that the area to the right of 1.57 is the same as the area to the left of -1.57. So we can use the pt function with the default setting to this effect:

```
(Pval = pt(-1.57,49))
```

```
## [1] 0.06142544
```

Property 2: The distribution always adds up to one.

This means that you have a 100% chance of pulling a number between negative and positive infinity. So if you use 1.57 and the default setting which gives you the area to the left, then subtract that number from 1 to get the area to the right:

```
(Pval = 1-pt(1.57,49))
```

```
## [1] 0.06142544
```

Final Option: Undo the default setting...

The full command for calculating a p-value from a t-distribution (for our purposes) is as follows:

pt(q, df, lower.tail = TRUE)

Note that q is the *quantity*, and df is the *degrees of freedom*. All other entries (if not specified) go to their default values. This is where *lower.tail* comes in. It is set to *TRUE* by default, meaning that whatever number you input for q, you will get the area to the left. If you change this entry to *FALSE*, then the default is switched off and you will calculate the area to the right.

```
(Pval = pt(1.57,49,lower.tail = FALSE))
```

```
## [1] 0.06142544
```

Notice that all three ways of calculating a p-value give you the exact same result. Therefore, you do not need to master all three - just pick whichever method works best for you.

MBA 8380

Chapter 7

Simple Linear Regression

Suppose you have two homes that are the same in *every way* except for size. Our intuition would suggest that bigger homes cost more (*all else equal*) so we would expect that there is a positive relationship between house size and house price.

Saying *bigger homes cost more* is a **qualitative** statement because all we are saying is that the relationship between house size and house price is positive. What if we want to make a **quantitative** statement? In other words, while we are fairly confident that the actual house price (say, in dollars) will increase for every unit increase in house size (say, an additional square foot) - we want to know exactly what this *average-price-per-square-foot* is.

A **Regression** can measure the relationship between the mean value of one variable and corresponding values of other variables. In other words, it is a statistical technique used to explain average movements of one (dependent) variable, as a function of movements in a set of other (independent) variables.

This chapter will discuss the estimation, interpretation, and statistical inference of a *simple* linear regression model, which means that we will attempt to explain the movements in a dependent variable by considering **one** independent variable. This is the simplest regression model we can consider in order to understand what is going on under the hood of a regression. The next chapter will extend this analysis to *multiple* regression models where the only real difference is that the number of independent variables are greater than one.

7.1 A Simple Linear Regression Model

A Linear Regression model is a line equation.

The simplest example of a line equation is:

$$Y_i = \beta_0 + \beta_1 X_i$$

The *betas*, β_0 and β_1 are called line coefficients.

- β_0 is the *constant* or *intercept term*
- β_1 is the *slope* term - it determines the change in Y given a change in X

$$\beta_1 = \frac{Rise}{Run} = \frac{\Delta Y_i}{\Delta X_i}$$

7.1.1 What does a regression model imply?

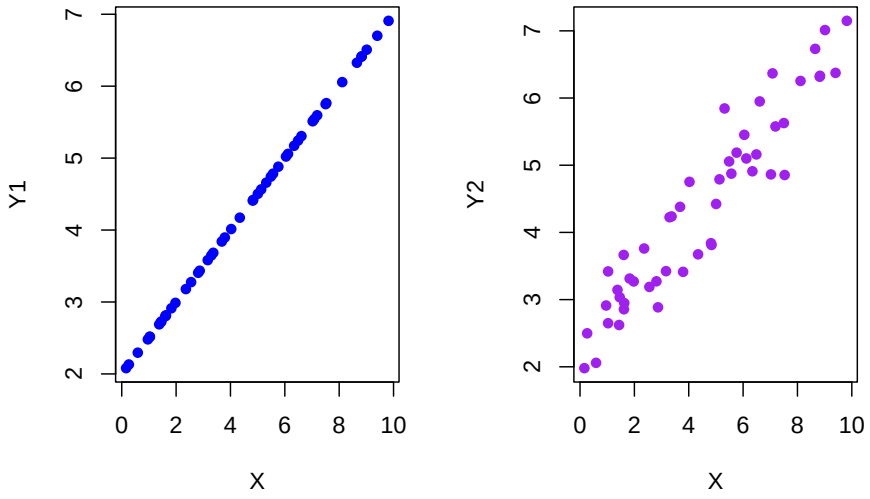
$$Y_i = \beta_0 + \beta_1 X_i$$

When we write down a model like this, we are imposing a huge amount of assumptions on how we believe the world works.

First, there is the **Direction of causality**. A regression implicitly assumes that changes in the independent variable (X) *causes* changes in the dependent variable (Y). This is the ONLY direction of causality we can handle, otherwise our analysis would be confounded (i.e., is don't know *what causes what*) and not useful.

Second, the equation assumes that information on the independent variable (X) is all the information you need to explain the dependent variable (Y). In other words, if we were to look at pairs of observations of X and Y on a plot, then the above equation assumes that all observations (data points) line up exactly on the (straight) regression line.

This would be great if the observations look like the figure on the left, but not if they look like the figure on the right.



It would be extremely rare for the linear model (as detailed above) to account for all there is to know about the dependent variable Y…

1. There might be other independent variables that explain different parts of the dependent variable (i.e., multiple dimensions). *(more on this next chapter)*
2. There might be measurement error in the recording of the variables.
3. There might be an incorrect functional form - meaning that the relationship between the dependent and independent variable might be more sophisticated than a straight line. *(more on this next chapter)*
4. There might be purely random and therefore totally unpredictable variation in the dependent variable.

This last item can be easily dealt with!

Adding a stochastic error term (ε_i) to the model will effectively take care of all sources of variation in the dependent variable (Y) that is not explicitly captured by information contained in the independent variable (X).

7.1.2 The *REAL* Simple Linear Regression Model

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

The Linear Regression Model now explicitly states that the explanation of the dependent variable (Y_i) can be broken down into two components:

- A **Deterministic** Component: $\beta_0 + \beta_1 X_i$
- A **Random / Stochastic / Explainable** Component: ε_i

Lets address these two components in turn.

The Deterministic Component

$$\hat{Y}_i = \beta_0 + \beta_1 X_i$$

The deterministic component delivers the expected (or *average*) value of the dependent variable (Y) given a values for the coefficients (β_0 and β_1) and a value of the dependent variable (X).

Since X is given, it is considered *deterministic* (i.e., non-stochastic)

In other words, the deterministic component determines the mean value of Y associated with a particular value of X. This should make sense, because the average value of Y is the best guess.

Technically speaking, the deterministic component delivers the *expected value of Y conditional on a value of X* (i.e., a conditional expectation).

$$\hat{Y}_i = \beta_0 + \beta_1 X_i = E[Y_i|X_i]$$

The Unexpected (*Garbage Can*) Component

$$\varepsilon_i = Y_i - \hat{Y}_i$$

Once we obtain the coefficients, we can compare the observed values of Y_i with the expected value of Y_i conditional on the values of X_i.

The difference between the true value (Y_i) and the expected value ($\hat{Y}_i$) is by definition… *unexpected!*

This unexpected discrepancy is your **prediction error** - and everything your deterministic component cannot explain is deemed *random* and *unexplainable*.

If a portion of the dependent variable is considered random and unexplainable - then it gets thrown away into the *garbage can* (ε_i).

This is a subtle but crucial part of regression modeling…

- Your choice of the independent variable(s) dictate what you believe to be important in explaining the dependent variable.
- The unimportant (or random) changes in the dependent variable that your independent variables cannot explain end up in the garbage can *by design*.
- Therefore, **the researcher** essentially chooses what is important, and what gets thrown away into the garbage can.

YOU are the researcher, so you determine what parts of the dependent variable goes into the garbage can by your selection of the independent variable(s)!

7.2 Application: Predicting House Price Based on House Size

Let's consider an application where we attempt to explain the price of a house (in thousand US$) by the size of a house (in square feet). We start by establishing the theory and relating it to our statistical terminology.

The Population Regression Function:

$$price_i = \beta_0 + \beta_i \; sqrft_i + \varepsilon_i$$

- $price_i$ is the dependent variable (Y_i)
- $sqrft_i$ is the independent variable (X_i)
- The equation above is the true (but unknown), population regression function.
- The coefficients (β_0 and β_1) are the population regression coefficients!
  - They are the coefficients you would obtain if you had *every* possible observation (i.e., the population)
  - This ain't gonna happen…
- We need to obtain the estimated, sample regression coefficients. To do this, we need to collect a sample of observations.

The Sample

In order to obtain *sample estimates* of our regression model above, we must obtain a sample of observations. We collect a (random) sample of size n. This is where the subscript i comes in - indicating that in general, each individual observation can be identified as $i = 1, ..., n$. The sample estimates are based on the sample.

Hypothetically, we can obtain different estimated coefficients for every different sample… but we will address that later.

To facilitate this application, we will use a data set internal to R, called *hprice1*.

```
data(hprice1,package='wooldridge')
ls(hprice1)
```

```
##  [1] "assess"   "bdrms"    "colonial" "lassess"  "llotsize" "lotsize"
##  [7] "lprice"   "lsqrft"   "price"    "sqrft"
```

This data set contains 88 observations of homes where each home has 10 pieces of information called *variables*. We are only concerned with two variables at the moment - the house price (*price*) and the house size (*sqrft*).

```
summary(hprice1$price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   111.0   230.0   265.5   293.5   326.2   725.0
```

```
summary(hprice1$sqrft)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1171    1660    1845    2014    2227    3880
```

We know that:[1]

1. the average house price is $293,500
2. 50% of the observations are between the 1st and 3rd quartiles of $230,000 and $326,200
3. the minimum house price in the sample is $111,000
4. the maximum house price in the sample is $725,000. You can look at the summary output for the size variable and make similar statements.

The Sample Regression Function

We combine our Population Regression Function (PRF) and our data sample to estimate a *Sample Regression Function* (SRF).

$$price_i = \hat{\beta}_0 + \hat{\beta}_1 \; sqrft_i + e_i$$

The difference between the SRF and the PRF are very important.

1. The PRF coefficients are *population parameters* while the SRF coefficients are *sample statistics*. In other words, the SRF coefficients are actual numbers that correspond to our sample, and we use them to draw inference on the things we really want to talk about - the PRF coefficients.[2]
2. The difference between the SRF residual (e_i) and the PRF residual (ε_i) is along the same lines as the difference between the SRF and PRF coefficients. The SRF residual contains the unexplained variability of the dependent variable in the sample while the PRF residual theoretically contains the unexplained variability in the population.

We will get into the details about how these regression estimates can be obtained later. Right now, lets just arrive at our estimates and shed light on the big picture.

[1] We covered the *summary* of a single variable in Chapter 1.

[2] Think of this along the lines of our univariate analyses earlier in the companion: μ is the population parameter while $\bar{X}$ is the sample statistic.

```
REG <- lm(price~sqrft,data = hprice1)
coef(REG)
```

```
## (Intercept)       sqrft
##   11.204145    0.140211
```

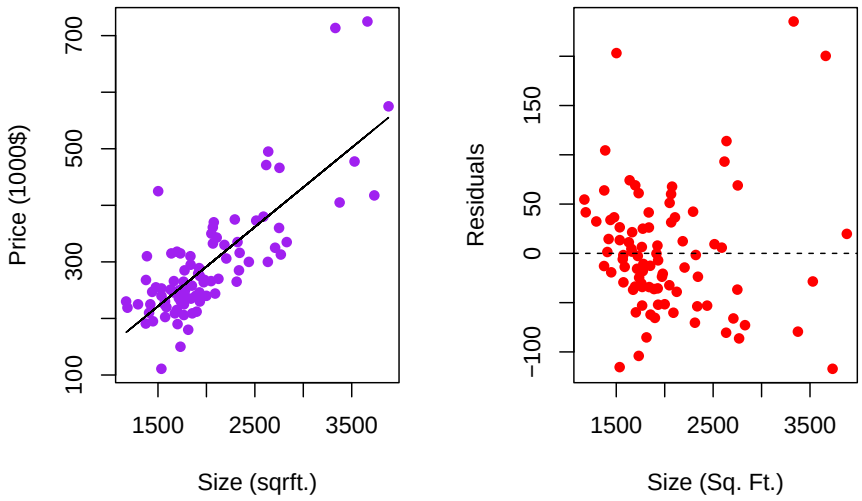
Our regression estimates are $\hat{\beta}_0 = 11.2$ and $\hat{\beta}_1 = 0.14$. This delivers a prediction equation from our SRF as:

$$\hat{price}_i = 11.2 + 0.14\ sqrft_i$$

Where $\hat{Y}_i = \hat{Price}_i$ is the expected house price conditional on a particular size.

We can illustrate the results of the regression as follows:

```
par(mfrow = c(1,2))
plot(hprice1$sqrft, hprice1$price,
     pch = 16, col = "purple",
     xlab = "Size (sqrft.)",
     ylab = "Price (1000$)")
lines(hprice1$sqrft,fitted(REG),col = 'black')
plot(hprice1$sqrft,residuals(REG),
     pch = 16, col = "red",
     xlab = "Size (Sq. Ft.)",
     ylab = "Residuals")
abline(h = 0,lty = "dashed")
```



In the left figure, the *dots* are a scatter-plot of the actual observations of house price (Y) and house size (X) while the blue line is our estimated regression which delivers the expected house price ($\widehat{price}_i$) for each observation of house size. Note that every time an actual house price is different than the expected value from the regression ($price_i - \widehat{price}_i$) - then that difference is considered *unexpected* and ends up in the *garbage can* (residual). The residual values are in the right figure. Note that the residual values are centered around the zero line - this means that the unexpected component of house price is equal to zero *on average*.

Analysis of the SRF

We can get plenty of mileage out of our estimated SRF.

1. We can interpret the estimated coefficients (one at a time) to get a sense of how house size influences house price.
    - $\hat{\beta}_0 = 11.2$ is the estimated intercept term. Mathematically, it is the expected value of the dependent variable conditional on the independent variable being 0 ($E[Y_i|X_i = 0] = 11.2$). In the context of this problem, we are saying that the *expected price of a house that has 0 square feet in size is 11.2 thousand dollars*. If that sounds funny to you... it should. The take away is that an intercept term always has a mathematical interpretation, but it might not always make sense. The key is if an independent value of zero (i.e., $X = 0$) makes sense.
    - $\hat{\beta}_1 = 0.14$ is the estimated slope term. Mathematically, it is the expected change in value of the dependent variable given a unit-increase in the independent variable ($\Delta Y_i / \Delta X_i = 0.14$). In the context of this problem, we are saying that the *expected price of a house will increase by 0.14 thousand dollars ($140) for every (square-foot) increase in house size*. If you were a realtor, you can now state that somebody looking for a home would be paying $140 per square foot of house size *on average*.
2. We can use the model for forecasting purposes.

To illustrate a forecast, suppose you came across a 1,800 square-foot house with a selling price of $250,000. Does this seem like a fair price? In order to answer this question with our estimated results, we simply plug 1800 square-feet as a value for our independent variable and arrive at an expected price conditional on this house size.

$$\widehat{price}_i = 11.2 + 0.14(1800) = 263.6$$

```
Bhat0 <- summary(REG)$coef[1,1]
Bhat1 <- summary(REG)$coef[2,1]
(Yhat = Bhat0 + Bhat1 * 1800)
```

```
## [1] 263.5839
```

Our regression forecast states that an 1,800 square-foot house should have an *average* price of $263,000. Since this is more than the $250,000 of the house in question, then the regression model suggests that this price is *below average*, and is a decent price based on the size of the house.

Discussion

While our model appears useful, we must always be mindful of its limitations. For starters, our regression **assumes** that house size is the **only** thing that matters when predicting house price. Our candidate house is more than $10,000 below the average 1,800 square-foot house price in the sample, but this might be due to very relevant things that our model considers *unpredictable*.

- Is the house located next to the town dump?
- Is the house built on top of an ancient burial ground?
- Does it have a really ugly kitchen?
- Does the roof leak?

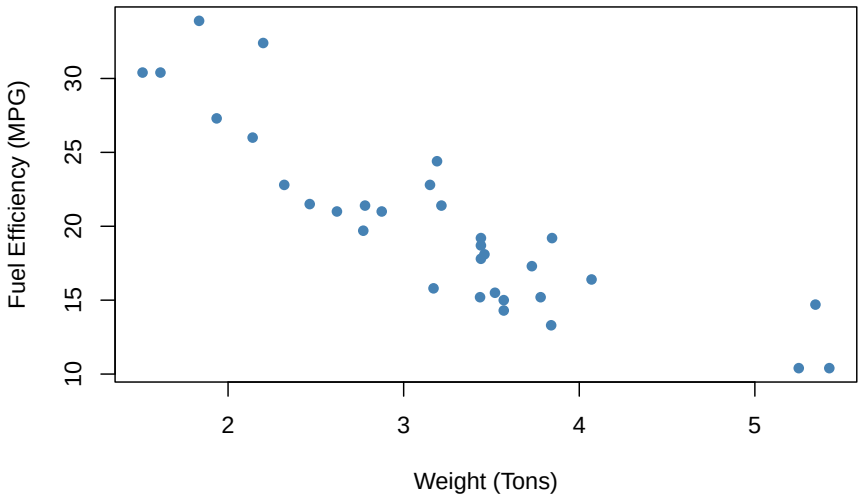
The bottom line is that one should always view our regression estimates within the lens of its limitations. This isn't to say that the estimates are *incorrect* or *wrong*, because they are actually quite useful. However, understanding how far one can take regression results is important.

7.3 Ordinary Least Squares (OLS)

Ordinary Least Squares (OLS, for short) is a popular method for estimating the unknown parameters in a linear regression model. OLS chooses the parameters of a linear function by minimizing the sum of the squared differences between the observed dependent variable (values of the variable being observed) in the given data set and those predicted by the linear function.

To illustrate this, consider a sample of observations and a PRF:

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$



Look at the figure and try to imagine the *best fitting* straight line that goes through all observations in the scatter plot. This line has two features: and intercept term ($\hat{\beta}_0$) and a slope term ($\hat{\beta}_1$). Which values would you assign?

We can go about this a little more formally. First, if we had values for $\hat{\beta}_0$ and $\hat{\beta}_1$, then we can determine the residual (error) for each pair of Y_i and X_i.

$$e_i = Y_i - (\hat{\beta}_0 + \hat{\beta}_1 X_i)$$

We can then sum across all observations to get the *total error*.

$$\sum_i e_i = \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)$$

The problem we face now is that error terms can be both positive and negative. That means that the individual error terms will start to wash each other out when we sum them up and we therefore get an incomplete measure of the total error. To prevent the positive and negative error terms from washing each other out, we square each of the terms. This makes the negative errors positive, while the positive errors stay positive.[3]

$$\sum_i e_i^2 = \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)^2$$

[3]Note: this is where the *sum of squared errors* comes in.

Notice that this function now states that we can calculate the sum of squared errors for any given values of $\hat{\beta}_0$ and $\hat{\beta}_1$. We can therefore find the *best* values of $\hat{\beta}_0$ and $\hat{\beta}_1$ that deliver the *lowest* sum of squared errors. The line that delivers the lowest squared errors is what we mean by the best line.

$$min \sum_i e_i^2 = min_{(\hat{\beta}_0, \hat{\beta}_1)} \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)^2$$

This function is called an *objective function*, and we can minimize the sum of squared errors by taking first-order conditions (i.e., the derivative of the objective function with respect to $\hat{\beta}_0$ and $\hat{\beta}_0$).

$$\hat{\beta}_1 = \frac{\sum_i (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_i (X_i - \bar{X})^2} = \frac{cov(X,Y)}{var(X)}$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

Where a 'bar' term over a variable represents the mean of that variable (i.e., $\bar{X} = \frac{1}{n} \sum_i X_i$)

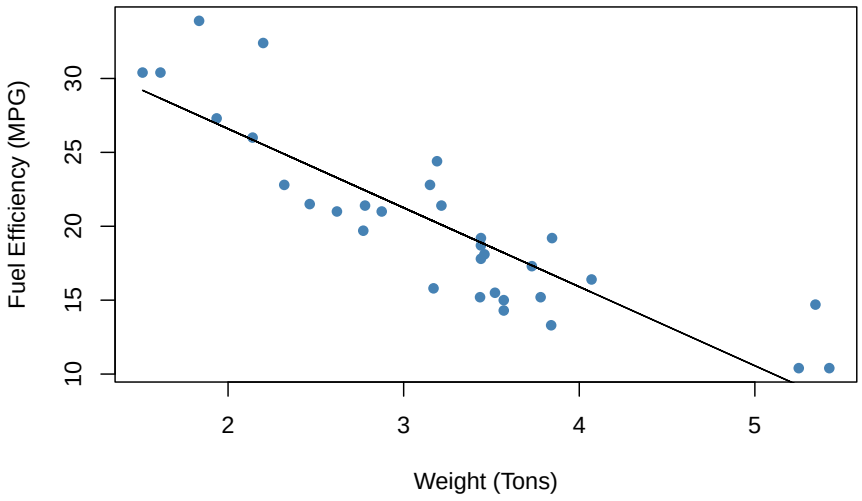
These two equations are important. The first equation states that the slope of the line equation ($\hat{\beta}_1$) is equal to the ratio between the covariance of Y and X and the variance of X. Remember that a covariance measures how two variables systematically move together. If they tend to go up at the same time, then they have a positive covariance. If they tend to go down - a negative covariance. If they do not tend to move together in any systematic way, then they have zero covariance. This systematic movement is precisely what helps determine the slope term. The second equation states that with $\hat{\beta}_1$ determined, we can determine $\hat{\beta}_0$ such that the regression line goes through the means of the dependent and independent variables.

Lets see what these estimates and the resulting regression line look like.

```
REG <- lm(mpg~wt,data = mtcars)
coef(REG)
```

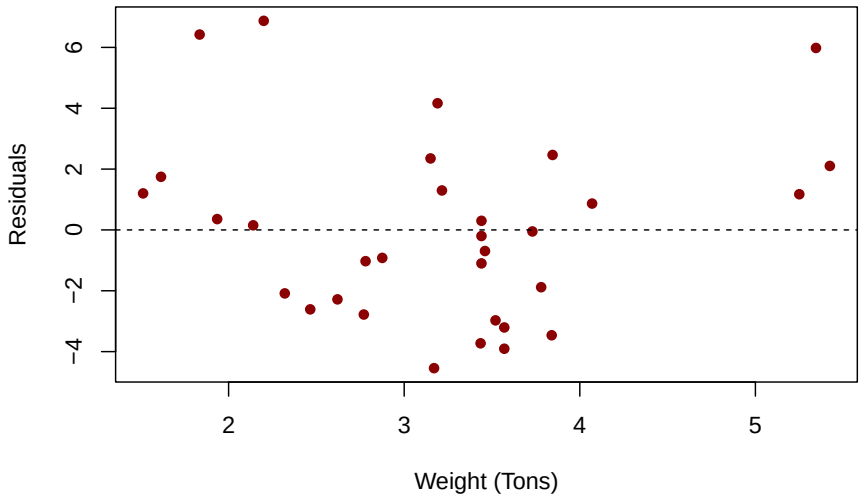
```
## (Intercept)          wt
##   37.285126   -5.344472
```

```
plot(mtcars$wt,mtcars$mpg,
     pch = 16, col = "steelblue",
     xlab = "Weight (Tons)",
     ylab = "Fuel Efficiency (MPG)")
lines(mtcars$wt,fitted(REG),col='black')
```



Now you probably imagined a line that looked kinda like this, but we know that this line (with these coefficients) is the absolute best line that minimizes the total difference between the observations (the *dots*) and the predictions (the *line*). Any other line we could draw would have a larger sum of squared errors. We can see what this difference looks like by looking at the residuals.

```
plot(mtcars$wt,residuals(REG),
     pch = 16, col = "darkred",
     xlab = "Weight (Tons)",
     ylab = "Residuals")
abline(h = 0,lty = "dashed")
```



Notice that these residual values are distributed both above and below the zero line. If you were to sum them all up - then you get zero ALWAYS. It is a mathematical outcome of finding a minimum of the objective function!

$$\sum_i e_i = 0$$

This mathematical outcome is actually important. First, if the residuals or *forecast errors* sum up to zero, then that means that they have a *mean* that is also 0 ($\bar{e} = 0$). This further means that they are zero *on average*, so the *expected value* is zero!

$$E[e_i] = 0$$

If the expected value of the forecast error is zero, then this means that our regression line is correct *on average*. If we think about it, this is the best we can ask for out of a regression function.[4]

7.3.1 B.L.U.E.

OLS is a powerful estimation method that delivers estimates with the following properties.

[4]If you said that your regression model is *wrong on average* - you would probably want a different regression model.

1. They are the **BEST** in a minimized mean-squared error sense. We just showed this.
2. They are **LINEAR** insofar as the OLS method can be quickly used when the regression model is a linear equation.
3. They are **UNBIASED** meaning that the sample estimates are true estimates of the population parameters.

Therefore, **BEST**, **LINEAR**, **UNBIASED**, **ESTIMATES** is why the output of an OLS method is said to be **B.L.U.E.**

7.4 Decomposition of Variance

Using our regression estimates and sample information, we can construct one of the most popular (and most abused) measures of *goodness of fit* for a regression. We will construct this measure in pieces.

First, the **total sum of squares** (or TSS) can be calculated to measure the total variation in the dependent variable:

$$TSS = \sum_{i=1}^{N} (Y_i - \bar{Y})^2$$

This expression is similar to a variance equation (without averaging), and since the movements in the dependent variable are ultimately what we are after, this measure delivers *the total variation in the dependent variable that we would like our model to explain.*

Next, we can use our regression estimates to calculate an **estimated sum of squares** (or ESS) which measures the total variation in the dependent variable that our model *actually* explained:

$$ESS = \sum_{i=1}^{N} (\hat{Y}_i - \bar{Y})^2$$

Note that this measure uses our conditional forecasts from our regression model in place of the actual observations of the dependent variable.

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$$

Finally, we can use our regression estimates to also calculate a **residual sum of squares** (or RSS) which measures the total variation in the dependent variable that our model *cannot* explain:

$$RSS = \sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{N} e_i^2$$

Note that this is a measure of the variation in the garbage can, and the garage can is where all of the variation in the dependent variable that your model cannot explain ends up.

7.4.1 The R^2

Our regression breaks the variation in Y_i (the TSS) into what can be explained (the ESS) and what cannot be explained (the RSS). This essentially means $TSS = ESS + RSS$. Furthermore, our OLS estimates attempt to maximize the ESS and minimize the RSS. This delivers our first measure of how well our model explains the movements in the dependent variable or *goodness of fit*

$$R^2 = \frac{ESS}{TSS} = 1 - \frac{RSS}{TSS}$$

This **coefficient of determination** or R^2 should be an intuitive measure. First, it is bounded between 0 and 1. If the measure is 0 then the model explains **NOTHING** and all variation is in the garbage can. If the measure is 1 then the model explains **EVERYTHING** and the garbage can is empty. Any number in between is simply the proportion of the variation in the dependent variable explained by the model.

```
REG3 <- lm(price ~ sqrft, data = hprice1)
summary(REG3)$r.squared
```

```
## [1] 0.6207967
```

```
pander(summary(REG3))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	11.2	24.74	0.4528	0.6518
sqrft	0.1402	0.01182	11.87	8.423e-20

Table 7.2: Fitting linear model: price ~ sqrft

Observations	Residual Std. Error	R^2	Adjusted R^2
88	63.62	0.6208	0.6164

Returning to our house price application above, you can see that our coefficient of determination (R^2) is 0.62.[5] This states that approximately 62 percent of

[5]Note that this number is sometimes called the *multiple* R^2

the variation in the prices of homes in our sample is explained by the size of the house (in square feet), while the remaining 38 percent is *unexplained* by our model and shoved into the garbage can. That is all it says… no more and no less.

7.4.2 What is a *good* R^2?

Is explaining 62 percent of the variation in house prices *good*? The answer depends on what you want the model to explain. We know that the house size explains a majority of the variation in house prices while *all other* potential independent variables will explain at most the remaining 38 percent. If you want to explain everything there is to know about house prices, then an R^2 of 0.62 leaves something to be desired. If you only care to understand the impact of size, then the R^2 tells you how much of the variation in house prices it explains. There really isn't much more to it than that.

7.4.3 Standard Error of the Estimate

$$S_{YX} = \sqrt{\frac{RSS}{n-2}} = \sqrt{\frac{\sum_{i=1}^{N} e_i^2}{n-2}}$$

The standard error of the estimate is much like a standard deviation equation. However, while the standard deviation measures the variability around a mean, the standard error of the estimate measures the variability around the prediction line.

Note that the denominator of this measure is $n-2$. The reason that we are *averaging* the sum of squared errors by $n-2$ is because we lost **two degrees of freedom**. Recall that we lose a degree of freedom whenever we need to estimate something based on other estimates. When we consider how we calculated the residuals in the first place,

$$e_i = Y_i - \hat{Y}_i = Y_i - \hat{\beta}_0 - \hat{\beta}_1 \, X_i$$

you will see that we had to estimate **two** line coefficients before we can determine what the prediction error is. That is why we deduct two degrees of freedom.[6]

7.5 Assumptions of the Linear Regression Model

An empirical regression analysis always begins with a statement of the population regression function (PRF). The PRF explicitly states exactly how you (the

[6] NOTE: this line of reasoning implies that we will lose more degrees of freedom when we estimate models with more independent variables. The general formula for degrees of freedom is $n-k-1$ where k is the number of estimated slope coefficients.

researcher) believes the independent variable is related to the dependent variable. One thing to be clear about when stating a PRF is that you are imposing a great deal of assumptions on how the world works. If your assumptions are correct, then the PRF is a reasonable depiction of reality and OLS will uncover accurate estimates of the PRF parameters. If your assumptions are incorrect, then the estimates are highly unreliable and might actually be misleading.

Verifying the assumptions of a linear regression model is a majority of the work involved with an empirical analysis, and we will be doing this for the rest of the course. Before getting into the details of *how* to verify the assumptions, we first need to know what they are.

One should note that these are not assumptions of our model, because these assumption are actually imposed on our model. These assumptions are made on what we think are going on in reality - at least on the relationship between the dependent and independent variables that actually occurs in the world.

The main assumptions of a linear regression model that we will focus on are as follows.

1. Linearity: the true relationship (in the world) is in fact linear. This assumption must hold because you are estimating a linear model (hence the linearity assumption is imposed on reality because we selected a linear model for analysis).
2. Independence of Errors: the forecast errors (e_i) are not correlated with each other
3. Equal Variance (*homoskedasticity*): the variance of the error term is constant
4. Normality of Errors: the forecast errors comprise a normal distribution

7.5.1 Linearity

If we write down the following PRF:

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

we are explicitly assuming that this accurately captures the real world. In particular,

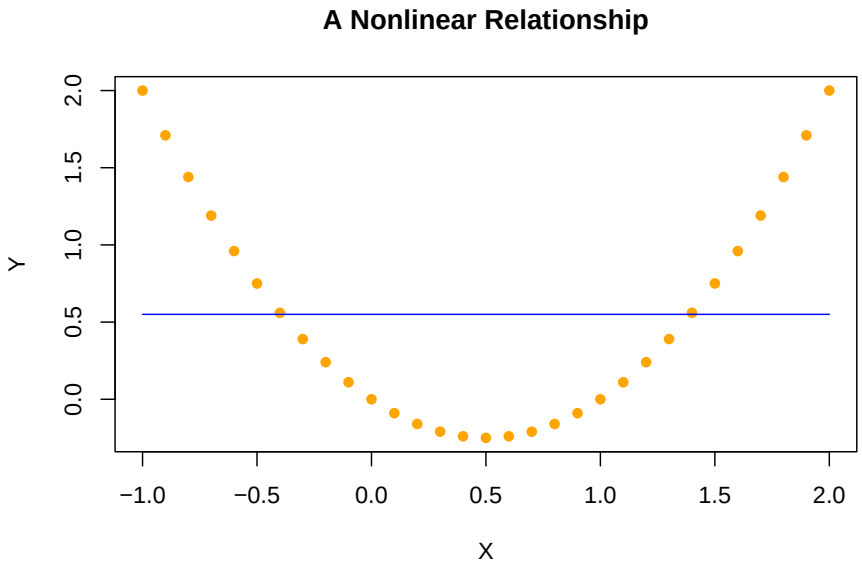
- The relationship between Y_i and X_i is in fact linear. This means that the a straight-line (i.e., a constant slope) fits the relationships between the dependent variable and independent variables better than a nonlinear relationship.
- The error term (i.e., the garbage can) is additive, meaning that the forecast errors are separable from the forecasts.

If these assumptions differ from the relationship that is going on in reality, then our model will suffer from *bias*. The SRF estimates will not be good representations of the PRF parameters, and they should not be interpreted as such.

There is an entire chapter devoted to relaxing the linearity assumption later on, but consider the following application to illustrate a violation of this assumption. Suppose you consider a simple linear regression model to uncover the relationship between a dependent variable and an independent variable.

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

However, suppose the true relationship (shown in the figure) is clearly nonlinear, and the blue line in the figure is the estimated (linear) SRF. As the line suggests, it is horizontal suggesting that the linear relationship between Y and X is *zero*. This doesn't mean that there is no relationship - because there clearly is. However, our assumption of this relationship being linear is incorrect because the results tell us that there is no *linear* relationship.



7.5.2 Independence of Errors

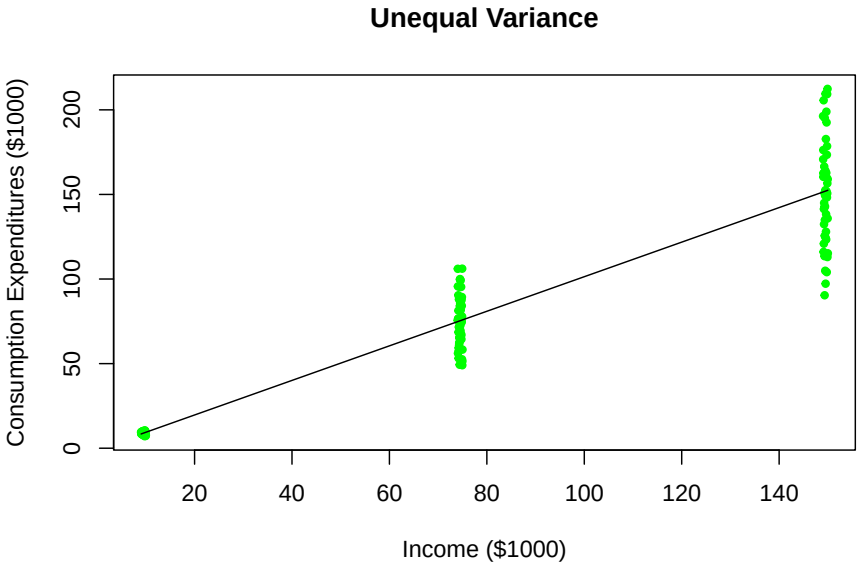
Serial correlation exists when an observation of the error term is correlated with past values of itself. This means that the errors are not independent of each other.

$$\varepsilon_t = \rho\varepsilon_{t-1} + \nu_t$$

If this is the case, the model violates the idea that the errors are completely unpredictable. If we would be able to view our past mistakes and improve upon our predictions - why wouldn't we?[7]

7.5.3 Equal Variance

The error term must have a constant variance throughout the range of each independent variable because we will soon see that the confidence we place in our estimates are partially determined by this variance. We are unable to change our confidence in the estimates throughout the observed range of independent values - it is one size fits all. Therefore, the size of the errors (i.e., the dispersion in the garbage can) must be constant throughout.



Suppose you wanted to estimate how much of an additional dollar of income the population would spend on consumption.[8] Your data set has 50 household observations from each of three annual income levels: $10,000, $75,000, and $150,000 as well as their annual consumption expenditures. As the figure illustrates, households earning around $10,000 a year all have roughly the same consumption level (because they all save very little). As income levels increase,

[7]Note that serial correlation is potentially a problem in time-series data (i.e., data that must be in chronological order).

[8]In economics this is called the *Marginal Propensity to Consume* and is an important measure for considering who should and should not get hit with a tax.

you see more *dispersion* in consumption expenditures because more income is paired with more options. Households earning $150,000 annually could choose to save a majority of it or even go into debt (i.e., spend more than $150,000). This data could be used to estimate a regression line (illustrated in black), but you can see that the model looks like it does a poorer job of predicting consumption expenditures as the income levels increase. This means that the forecast errors are increasing (becoming more disperse) as income levels increase, and this is *heteroskedasticity*. We will briefly come back to potential solutions to this later in the advanced topics section.

7.5.4 Normality of Errors

We know that OLS will produce forecast errors that have a mean of zero as well as a variance that is as low as possible by finding the *best fitting* straight line. The assumption that these are now the two moments that can be used to describe a normal distribution comes directly from the Central Limit Theorem and the concept of a sampling distribution. Recall that the *population* error term is zero on average and has some nonzero variance. A random sample of these error terms should have similar characteristics, as well as comprising a normal distribution.

7.6 Appendix: Statistical Inference

Note that statistical inference of a linear regression model is officially discussed in the next chapter dealing with multiple regressions. This appendix is a similar discussion only dealing with simple regressions. It is intended to be used as an additional reference if needed.

Once the assumptions of the regression model have been verified, we are able to perform statistical inference. Similar to the univariate analysis in the previous section of the book, we are able to calculate confidence intervals and conduct hypothesis tests on the population coefficients. However, since a regression is a multivatiate analysis too that considers a causal relationship between the independent and dependent variables, we are able to perform statistical inference on the forecasts of the model as well.

7.6.1 Confidence Intervals (around population parameters)

Recall our earlier formula for calculating a confidence interval in a single-variable context:

$$Pr\left(\bar{X} - t_{(\frac{\alpha}{2}, df=n-1)}\frac{S}{\sqrt{n}} \leq \mu \leq \bar{X} + t_{(\frac{\alpha}{2}, df=n-1)}\frac{S}{\sqrt{n}}\right) = 1 - \alpha$$

We used the CLT to ultimately state that $\bar{X}$ was drawn from a normal distribution with a mean of μ and standard deviation $\sigma/\sqrt{n}$ (but we only have S which makes this a t distribution). This line of reasoning is *very* similar to what we have with regression analyses.

First, $\hat{\beta}$ is an estimate of β just like $\bar{X}$ is an estimate of μ. However, the standard error of the sampling distribution of $\hat{\beta}$ is derived from the standard deviation of the residuals.

$$S_{\hat{\beta}} = \frac{S_{YX}}{\sum (X_i - \bar{X})^2}$$

This means that we construct a *standardized* random variable from a t distribution with $n-2$ degrees of freedom.

$$t = \frac{\hat{\beta} - \beta}{S_{\hat{\beta}}}$$

We have already derived a confidence interval before, so we can skip to the punchline.

$$Pr\left(\hat{\beta} - t_{(\frac{\alpha}{2}, df=n-2)} S_{\hat{\beta}} \leq \beta \leq \hat{\beta} + t_{(\frac{\alpha}{2}, df=n-2)} S_{\hat{\beta}}\right) = 1 - \alpha$$

This is the formula for a confidence interval around the *population* slope coefficient β given the estimate $\hat{beta}$ and the regression characteristics. It can also be written compactly as before.

$$\hat{\beta} \pm t_{(\frac{\alpha}{2}, df=n-2)} S_{\hat{\beta}}$$

Recall our regression explaining differences in house prices given information on house sizes.

```
pander(summary(REG3))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	11.2	24.74	0.4528	0.6518
sqrft	0.1402	0.01182	11.87	8.423e-20

Table 7.4: Fitting linear model: price ~ sqrft

Observations	Residual Std. Error	R^2	Adjusted R^2
88	63.62	0.6208	0.6164

The information included in the regression summary is all that is needed for us to construct a 95 percent ($\alpha = 0.05$) confidence interval around the *population* slope coefficient β_1.

```
# Back out all of the needed information:

Bhat1 <- summary(REG3)$coef[2,1]
SBhat1 <- summary(REG3)$coef[2,2]
N <- length(residuals(REG3))

# Find the critical t-distribution values... same as before

AL <- 0.05
df <- N-2
tcrit <- qt(AL/2,df,lower.tail = FALSE)

# Use the formula... same as before

(LEFT <- Bhat1 - tcrit * SBhat1)
```

```
## [1] 0.1167203
```

```
(RIGHT <- Bhat1 + tcrit * SBhat1)
```

```
## [1] 0.1637017
```

$$Pr(0.1167 \leq \beta_1 \leq 0.1637) = 0.95$$

This states that while an increase in house size by one square foot will increase the house price by \$140 ($\widehat{\beta}_1$) on average in the sample, we can also state that an increase in house size by one square foot will increase the house price *in the population* somewhere between \$116.70 and \$163.70 with 95% confidence.

While the code above showed you how to calculate a confidence interval from scratch as we did before, there is an easier (one-line) way in R:

```
confint(REG3)
```

```
##                   2.5 %     97.5 %
## (Intercept) -37.9825309 60.3908210
## sqrft         0.1167203  0.1637017
```

7.6.2 Hypothesis Tests

We are able to conduct hypothesis tests regarding the values of the population regression coefficients. For example:

$$H_0 : \beta_1 = 0 \quad vs. \quad H_1 : \beta_1 \neq 0$$

In the context of our house price application, this null hypothesis states that the population slope between house price and size is zero... meaning that there is *no* relationship between the two variables.

Given the null hypothesis above, we follow the remaining steps laid out previously: we calculate a test statistic under the null, calculate a p-value, and conclude.

The test statistic under the null is given by

$$t = \frac{\hat{\beta}_1 - \beta_1}{S_{\hat{\beta}_1}}$$

and this test statistic is drawn from a t distribution with $n-2$ degrees of freedom. Concluding this test is no more difficult that what we've done previously.

```
B1 = 0
(tstat <- (Bhat1 - B1)/SBhat1)
```

```
## [1] 11.86555
```

```
(Pval <- pt(tstat,N-2,lower.tail=FALSE)*2)
```

```
## [1] 8.423405e-20
```

```
(1-Pval)
```

```
## [1] 1
```

Our results state that we can reject this null hypothesis with approximately 100% confidence, meaning that there is a statistically significant relationship between house prices and house sizes.

As with the confidence interval exercise above, we actually do not need to conduct hypothesis tests where the null sets the population parameter to zero because R does this automatically. If you look again at columns to the right of the estimated coefficient $\hat{\beta}_1$, you will see a t value that is exactly what we calculated above and a p value that is essentially zero. This implies that a test with the null hypothesis set to zero is always done for you.

```
summary(REG3)
```

```
##
## Call:
## lm(formula = price ~ sqrft, data = hprice1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -117.112  -36.348   -6.503   31.701  235.253
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.20415   24.74261   0.453    0.652
## sqrft        0.14021    0.01182  11.866   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 63.62 on 86 degrees of freedom
## Multiple R-squared:  0.6208, Adjusted R-squared:  0.6164
## F-statistic: 140.8 on 1 and 86 DF,  p-value: < 2.2e-16
```

This isn't to say that all hypothesis tests are automatically done for you. Suppose a realtor believes that homes sell for \$150 per square foot. This delivers the following hypotheses, followed by a test statistic, p-value, and conclusion.

$$H_0 : \beta_1 = 0.150 \quad vs. \quad H_1 : \beta_1 \neq 0.150$$

```
B1 = 0.150
(tstat <- (Bhat1 - B1)/SBhat1)
```

```
## [1] -0.8284098
```

```
(Pval <- pt(tstat,N-2)*2)
```

```
## [1] 0.4097316
```

```
(1-Pval)
```

```
## [1] 0.5902684
```

Our p-value of 0.41 implies that there is a 41% chance of being wrong if we reject the null hypothesis. We therefore do not have evidence that the population slope is different from 0.150… so we do not reject.

One-sided tests are also like before. Suppose a realtor believes that homes sell *more than* \$160 per square foot. This delivers the following hypotheses, followed by a test statistic, p-value, and conclusion.

$$H_0 : \beta_1 \leq 0.160 \quad vs. \quad H_1 : \beta_1 > 0.160$$

```
B1 = 0.160
(tstat <- (Bhat1 - B1)/SBhat1)
```

```
## [1] -1.674674
```

```
(Pval <- pt(tstat,N-2))
```

```
## [1] 0.04881561
```

```
(1-Pval)
```

```
## [1] 0.9511844
```

Our test concludes that we can reject the null with at most 95.11% confidence.

7.6.3 Confidence Intervals (around forecasts)

A regression can also build confidence intervals around the conditional expectations (i.e., forecasts) of the dependent variable.

Suppose you want to use our model to predict the price of a 1000 square foot house. The conditional expectation is calculated by using our regression coefficients, a value of house size of 1000, and setting our forecast error to zero.

```
X = 1000
Bhat0 = summary(REG3)$coef[1,1]
Bhat1 = summary(REG3)$coef[2,1]

(Yhat = Bhat0 + Bhat1 * X)
```

```
## [1] 151.4151
```

Another way to calculate this forecast is using the predict command in R. This command creates a new data frame that includes only the value for the independent variable you want to make a prediction with. The rest is done for you.

```
predict(REG3,data.frame(sqrft = 1000))
```

```
##        1
## 151.4151
```

Our model predicts that a 1,000 square foot house will sell for $151,415 on average. While this is an expected value based on the sample, what is the prediction in the population? We are able to build a confidence interval around this forecast in a number of ways.

- A confidence interval for the mean response
- A confidence interval for an individual response

The mean response: a confidence interval

Suppose you want to build a confidence interval around the mean price for a 1000 square foot house in the population. This is a conditional mean. In other words, we want the average house price but *only* for homes with a particular size. This conditional mean is generally given by $\mu_{Y|X=X_i}$ and in this case by $\mu_{Y|X=1000}$. Building a confidence interval for the mean response is given by

$$\hat{Y}_{X=X_i} \pm t_{(\frac{\alpha}{2},df=n-2)} S_{YX} \sqrt{h_i}$$

or

$$\hat{Y}_{X=X_i} - t_{(\frac{\alpha}{2},df=n-2)} S_{YX} \sqrt{h_i} \leq \mu_{Y|X=X_i} \leq \hat{Y}_{X=X_i} + t_{(\frac{\alpha}{2},df=n-2)} S_{YX} \sqrt{h_i}$$

where

- $\hat{Y}_{X=X_i}$ is the expectation of the dependent variable conditional on the desired value of X_i.
- S_{YX} is the standard error of the estimate (calculated previously)
- $t_{(\frac{\alpha}{2},df=n-2)}$ is the critical t statistic (calculate previously)
- $h_i = \frac{1}{n} + \frac{(X_i - \bar{X})^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}$

This last variable h_i is what is new to us and increases the size of the confidence interval when the desired value of X_i is farther away from the average value of the observations $\bar{X}$. This variable can sometimes be difficult to calculate, but R again does it for you. In R, a confidence interval around the population mean is simply called a *confidence* interval.

```
predict(REG3,
        data.frame(sqrft = 1000),
        interval = "confidence",
        level = 0.95)
```

```
##        fit      lwr      upr
## 1 151.4151 124.0513 178.7789
```

$$Pr(124.05 \leq \mu_{Y|X=1000} \leq 178.78) = 0.95$$

We can now state with 95% confidence that the *population mean house price* of all 1000 square-foot houses is somewhere between \$124,050 and \$178,780. Note that the confidence interval around the mean response is centered at our conditional expectation ($\hat{Y}$) just like all confidence intervals are centered around its estimate.

An individual response: a prediction interval

Suppose that instead of building a confidence interval around the conditional average in the population, we want to determine the range within which we are confident to draw a *single* home value. This calculation is almost identical to the mean response above, but with one slight difference.

$$\hat{Y}_{X=X_i} \pm t_{(\frac{\alpha}{2},df=n-2)} S_{YX} \sqrt{1+h_i}$$

or

$$\hat{Y}_{X=X_i} - t_{(\frac{\alpha}{2},df=n-2)} S_{YX} \sqrt{1+h_i} \leq Y_{X=X_i} \leq \hat{Y}_{X=X_i} + t_{(\frac{\alpha}{2},df=n-2)} S_{YX} \sqrt{1+h_i}$$

where

- $\hat{Y}_{X=X_i}$ is the expectation of the dependent variable conditional on the desired value of X_i.
- S_{YX} is the standard error of the estimate (calculated previously)
- $t_{(\frac{\alpha}{2},df=n-2)}$ is the critical t statistic (calculate previously)
- $h_i = \frac{1}{n} + \frac{(X_i - \bar{X})^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}$

The only difference is that we replace $\sqrt{h_i}$ with $\sqrt{1+h_i}$. Conceptually, we inserted the one in the formula because we are selecting a *single* home with a specified size out of the population. This is very different from building a confidence interval around a population mean, but in R it is simply the change of one word.

```
predict(REG3,
        data.frame(sqrft = 1000),
        interval = "prediction",
        level = 0.95)
```

```
##        fit      lwr      upr
## 1 151.4151 22.02204 280.8082
```

$$Pr(22.02 \leq Y_{X=1000} \leq 280.81) = 0.95$$

We can now state with 95% confidence that *a single draw of a house price* from the population of all 1000 square-foot houses will be somewhere between $22,020 and $280,810. Note that the prediction interval is also centered at our conditional expectation ($\hat{Y}$), but now the interval is much wider than in the previous calculation. This should make sense, because when you are selecting a single home then you have a positive probability of selecting either very cheap homes or very expensive homes. A mean would wash these extreme values out.

7.7 Up Next...

This chapter covered the basics of a simple linear regression model. The next step is to turn to **multiple** linear regression models where we get to explain the dependent variable by using more than one independent variable. This will allow us to make more sophisticated models that have a better chance of capturing the relationships that are actually going on in the real world. The thing to keep in mind is that **ALL** of the information laid out in this chapter will still apply.

Chapter 8

Multiple Linear Regression

Sometimes one independent variable just doesn't cut it.

$$PRF: \ Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + ... + \beta_k X_{ki} + \varepsilon_i$$

$$SRF: \ Y_i = \hat{\beta}_0 + \hat{\beta}_1 X_{1i} + \hat{\beta}_2 X_{2i} + ... + \hat{\beta}_k X_{ki} + e_i$$

A **Multiple Regression Model** is a direct extension of the **Simple Regression Model** by adding additional independent variables. Adding additional independent variables allows the regression to use more information when trying to explain movements in the single dependent variable. In other words, multiple independent variables can explain changes in the dependent variable along different *dimensions*.

The multiple regression model has a lot in common with the simple regression model.

1. It is still the case that we establish a *population regression function* (PRF) that we believe holds in the population, but we are forced to estimate a *sample regression function* (SRF) because we can only observe a sample (i.e., subset) of the population.
2. The SRF is still solved via OLS. The first-order conditions are a bit more complicated than those stemming from a simple regression, but are conceptually the same.
3. The PRF and SRF still each contain a single intercept term and a single residual term.
4. The model we are examining is still a *line* equation - only it is a multi-dimensional line equation (i.e., a plane in the case of two dimensions).

The only significant change we need to make is with respect to the interpretation of the slope coefficients of our model. These slope coefficients still deliver the *expected or average change in the dependent variable given a unit change in an independent variable*. However, since we are looking at multiple independent variables simultaneously, we need to be **explicit** that we are examining these relationships *one independent variable at a time*. In other words, when we examine the relationship between the dependent variable and a particular independent variable, we need to explicitly state that we are holding all other independent variables *constant*.

$$\beta_k = \frac{\Delta Y}{\Delta X_k}$$

In the population: a PRF slope coefficient indicates the EXPECTED or AVERAGE change in the dependent variable associated with a one-unit increase in the kth explanatory variable holding all other explanatory variables constant.

$$\hat{\beta}_k = \frac{\Delta Y}{\Delta X_k}$$

In the sample: a SRF slope coefficient indicates the expected or average change in the dependent variable associated with a one-unit increase in the kth explanatory variable holding all other explanatory variables constant.

8.1 Application: Explaining house price in a multiple regression

Let us revisit the relationship between house price and house size, but extend the regression model to include a second independent variable: the number of bedrooms.

Our PRF becomes:

$$price_i = \beta_0 + \beta_1\ sqrft_i + \beta_2\ bdrms_i + \varepsilon_i$$

Our SRF becomes:

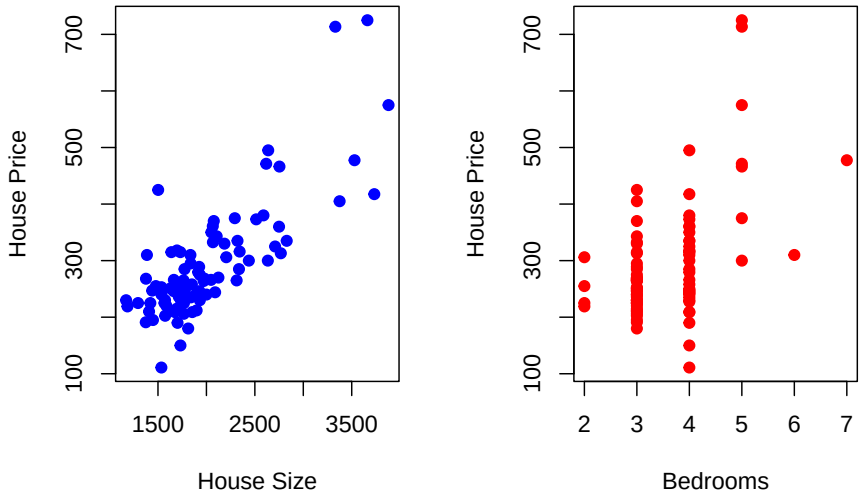
$$price_i = \hat{\beta}_0 + \hat{\beta}_1\ sqrft_i + \hat{\beta}_2\ bdrms_i + e_i$$

To visualize what we are about to do, lets start with scatter plots looking at the relationships between the dependent variable and each independent variable.

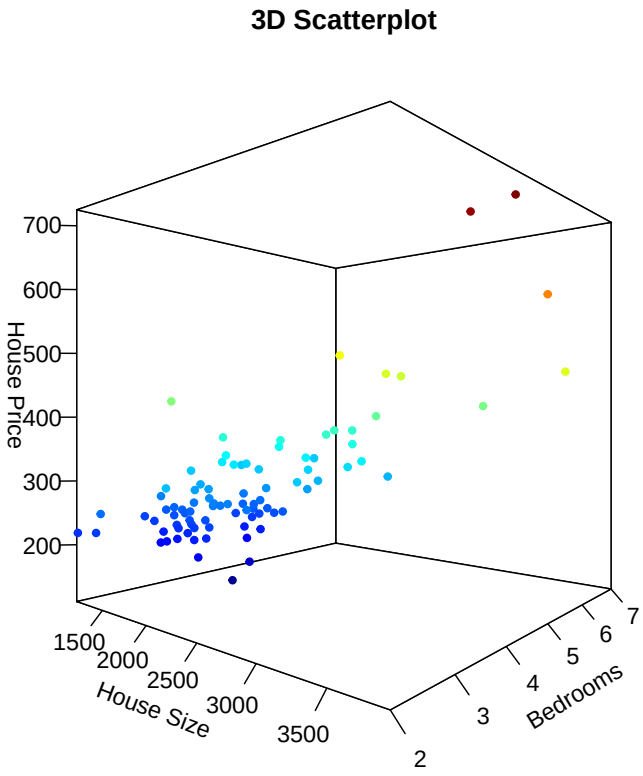
The figure on the left is the scatter plot between the House Price and House Size. This positive relationship is exactly what we have looked at previously.

The figure on the right is the scatter plot between the same House Price but the number of bedrooms each house has. This figure illustrates that the houses in our sample have between 2 and 7 bedrooms (with no half-rooms), and homes with more bedrooms generally have higher prices (as expected). Note that we are looking at the same dependent variable along different *dimensions*. We can combine these dimensions into a singe (3-Dimensional) figure to see how the relationships between the dependent variable and each independent variable appear simultaneously.

```
data(hprice1,package='wooldridge')
Y <- hprice1$price
X1 <- hprice1$sqrft
X2 <- hprice1$bdrms

par(mfrow = c(1,2))
plot(X1,Y, col = "blue",
          pch = 19, cex = 1,
          xlab = "House Size", ylab = "House Price")
plot(X2,Y, col = "red",
          pch = 19, cex = 1,
          xlab = "Bedrooms", ylab = "House Price")
```




```
scatter3D(X1, X2, Y, pch = 20, cex = 1, phi = 0,
          colkey=FALSE, ticktype = "detailed",
          xlab = "House Size", ylab = "Bedrooms",
          zlab = "House Price", main = "3D Scatterplot")
```



For comparison, suppose that we consider these independent variables one at a time in two simple regressions. In particular, we can examine one simple regression model where House Size is the only independent variable, and another simple regression model where Bedrooms is the only independent variable.[1]

```
REG1 <- lm(hprice1$price ~ hprice1$sqrft)
coef(REG1)
```

```
##   (Intercept) hprice1$sqrft
##     11.204145      0.140211
```

```
REG2 <- lm(hprice1$price ~ hprice1$bdrms)
coef(REG2)
```

[1]Note that this is only for explanatory purposes only. If this were an actual analysis, we really wouldn't gain much from considering each independent variable separately.

```
##   (Intercept) hprice1$bdrms
##      72.23111      62.02456
```

The regression considering house size alone has a slope coefficient of 0.14. Remember that since house price was denoted in thousands of dollars and house size was denoted in square feet, this slope coefficient states that an additional square foot of house size will increase the average house price by $140.

The regression considering number of bedrooms alone has a slope coefficient of 62. This slope coefficient states that an additional bedroom will increase the average house price by $62,000.

While the results from these two simple regressions make sense, we need to realize that a simple regression model only considers a single independent variable (and throws all of the other information into the garbage can). This means the first regression takes no notice of the number of rooms a house has, while the second regression takes no notice of the size of the home. Since it is reasonable to assume that bigger homes have more bedrooms, then a regression model that is only given one of these pieces of information might be overstating the quantitative impact of the single independent variable.

To illustrate this, let us run a multiple regression model where both house size and number of bedrooms are considered.

```
REG3 <- lm(price ~ sqrft + bdrms, data = hprice1)
coef(REG3)
```

```
## (Intercept)       sqrft       bdrms
## -19.3149958   0.1284362  15.1981910
```

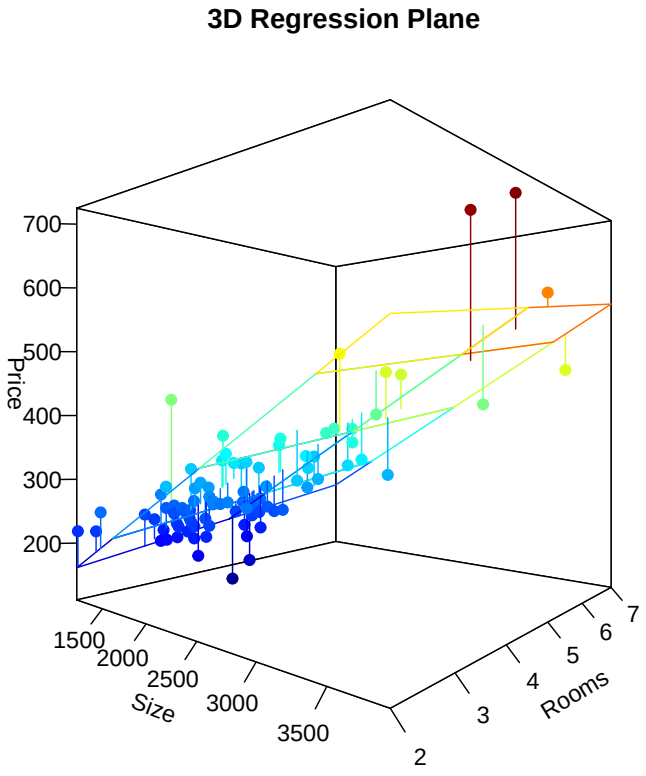
The slope with respect to house size is now 0.128 (down from 0.14) while the slope with respect to number of bedrooms is now 15.2 (down from 62). In order to make sense of these changes, let us explicitly interpret these slope coefficients within the context of a multiple regression model (where we can hold all other independent variables constant).

> Holding number of bedrooms constant, an additional square foot of house size will increase a house price by $128, on average.
>
> Holding house size constant, an additional bedroom will increase a house price by $15,200, on average.

The power of a multiple regression comes through when you look at the second slope interpretation. Multiple regression allows us to consider two houses that have the same house size but one house has an additional bedroom. In other words, imagine building a wall that turns one bedroom into two smaller bedrooms (and doesn't change the house size). This will increase the expected house price by $15,200. This is much smaller than the simple regression relationship of $62,000 because the simple regression could not differentiate the impact of a bedroom from the impact of an increase in house size. The multiple regression

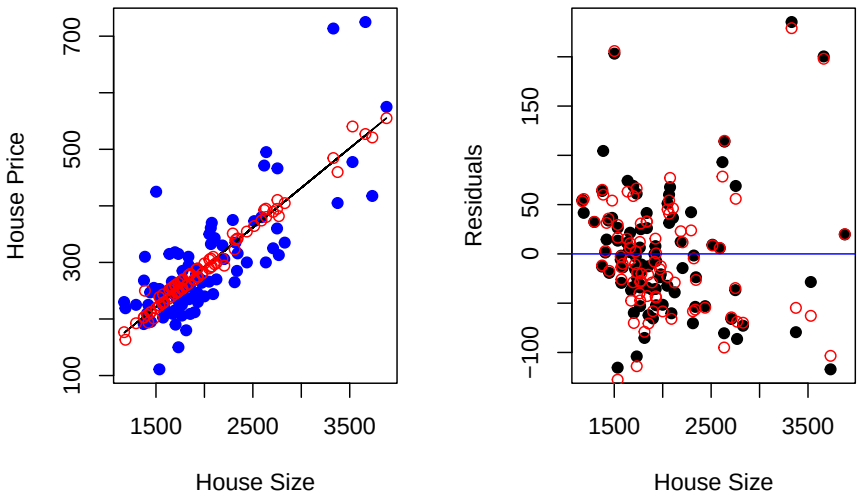
model can.

The final figure shows the 3-dimensional regression line (i.e., plane) that best fits the sample. You can see that considering multiple dimensions increases the performance of the deterministic component of the model and therefore reduces the amount of information that goes into the garbage can as *unpredictable*. This can be shown in the last picture that only looks at the relationship from the "sqrft" dimension. In the figure on the left, the blue dots are the observations in the data, the black line is the regression line from the simple regression model (without Bedrooms), while the red dots are the model predictions from the multiple regression model with Bedrooms included as an additional independent variable. Notice how this allows the regression predictions to veer off of a straight line. This results in slightly less prediction errors showing up in your garbage can - as illustrated in the figure on the right.




```
par(mfrow = c(1,2))
plot(hprice1$sqrft,hprice1$price, col = "blue",
```

```
          pch = 19, cex = 1,
          xlab = "House Size", ylab = "House Price")
lines(hprice1$sqrft,fitted(REG1))
points(hprice1$sqrft,fitted(REG3),col = "red")

plot(hprice1$sqrft,residuals(REG1), col = "black",
          pch = 19, cex = 1,
          xlab = "House Size", ylab = "Residuals")
points(hprice1$sqrft,residuals(REG3),col = "red")
abline(h = 0,col="blue")
```



8.1.1 The Importance of "Controls"

One very important item to point out in the last application is exactly why the coefficient on number of bedrooms dropped from $62,000 to $15,200 when the size of the house was added to the regression. The reason can be broken up into two categories.

1. The independent variables are correlated

It seems reasonable to believe that bigger houses have more bedrooms. This means that the size of a house and the number of bedrooms are correlated with each other.

2. "All Else Equal" in a Multiple Regression is more than just words

A multiple regression can separately identify the average impact of each independent variable on the dependent variable.

Put together, these two items suggest that when two independent variables are correlated, then they should both appear in the regression model. If not, then the correlation between an included independent variable and an omitted independent variable might lead to *omitted variable bias*. This is what we saw above in the regression with only number of bedrooms as an independent variable. The coefficient of $62,000 is giving you the combined impact of an additional room *and* a bigger house. When you add house size as another independent variable, you are now able to determine the expected increase in house price for an additional bedroom *holding house size constant*.

Bottom line is that even though you are concerned with the results from a particular independent variable, it is important to try and include all independent variables that might be correlated with the independent variable of interest. This attempts to alleviate omitted variable bias.

8.2 Adjusted R^2

Regardless of the number of independent variables, the variance of a regression model can be decomposed and a R^2 can be calculated.

$$TSS = \sum_{i=1}^{N}(Y_i - \bar{Y})^2$$

$$ESS = \sum_{i=1}^{N}(\hat{Y}_i - \bar{Y})^2$$

$$RSS = \sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{N} e_i^2$$

$$R^2 = \frac{ESS}{TSS} = 1 - \frac{RSS}{TSS}$$

The R^2 still delivers the proportion of the variation in the dependent variable explained by the model, only now the model is comprised of multiple independent variables. In other words, we are using more independent variables when comprising the deterministic component of our model and making expectation of the dependent variable.

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 \, X_{1i} + \hat{\beta}_2 \, X_{2i} + ... + \hat{\beta}_k \, X_{ki}$$

An R^2 is a very intuitive calculation, but it sometimes might be misleading.

8.2.1 Abusing an R^2

No matter how hard I try to downplay the importance of an R^2, students always have the tendency to shoot for that measure to be as close to 1 as possible. The problem with this goal is that an R^2 equal to 1 in not necessarily a good thing. Furthermore, achieving an R^2 of one might be *impossible*.

An R^2 of 1 is sometimes impossible because our PRF actually has a residual term. This means that we understand that there is forecast error in the population. In other words, suppose that we are trying to understand the movements of a rather *noisy* dependent variable and 20 percent of its variation is entirely random and unpredictable. This would imply that a value of R^2 equal to 0.80 is the **highest** value we can hope to achieve. If we somehow received a value higher than that… our sample regression function is not a valid representation of our population regression function.

Let us illustrate how an R^2 value can be misleading by way of an application. Consider a previous regression where we explained house prices with only the number of bedrooms.

```
REG1 <- lm(price ~ bdrms, data = hprice1)
summary(REG1)$r.squared
```

```
## [1] 0.2581489
```

The coefficient of determination states that the number of bedrooms explains slightly around 0.26 percent of the variation in house prices. If we include the size of the house in the regression,

```
REG2 <- lm(price ~ bdrms + sqrft, data = hprice1)
summary(REG2)$r.squared
```

```
## [1] 0.6319184
```

we see that the R^2 increases to 0.63 as before. If we include yet another variable such as the size of the property,

```
REG3 <- lm(price ~ bdrms + sqrft + lotsize, data = hprice1)
summary(REG3)$r.squared
```

```
## [1] 0.6723622
```

we see that the regression now explains about 0.67 percent of the variation in house prices.

What we are seeing is that the more variables you add the higher the R^2 is getting. While this might lead you to believe that we are adding *important* independent variables to the regression, the problem is that the R^2 will go up

no matter what variable you add. The increase might be slight, but the R^2 will never go down.

```
Xcrap <- rnorm(88)
```

```
REG4 <- lm(price ~ bdrms + sqrft + lotsize + Xcrap, data = hprice1)
summary(REG4)$r.squared
```

```
## [1] 0.6753351
```

The exercise above adds a completely random variable as a fourth independent variable. It should have nothing to do with explaining house prices. However, if you generate *the correct* random variables, then you might get an increase in the R^2 by as much as an entire percentage point. Does this say that the random variable actually helps explain variations in house prices? Of course not. What it does show is that sometimes we can abuse the R^2, so we need an additional measure of goodness of fit.

8.2.2 An *Adjusted* R^2

The problem with an R^2 is that it will increase no matter what independent variable you throw into the regression. If you think about it, if a regression with two independent variables explains 63 percent of the variation in the dependent variable, then adding a third variable (no matter how silly) will deliver a regression that will explain *no less* than 63 percent of the variation. We therefore cannot use the R^2 as an informal measure for whether or not we should include an independent variable because we don't know how *big* an increase in R^2 needs to be. We therefore need a goodness of fit measure that not only has the potential to increase when the added variable is deemed important, but has the potential to decrease when the variable is unimportant. This is called an *adjusted* R^2.

$$\bar{R}^2 = 1 - \frac{RSS/(N-k-1)}{TSS/(N-1)}$$

The main difference between the adjusted R^2 and it's unadjusted measure are the degrees of freedom in the numerator. When you add an additional independent variable, k goes up by one but N stays constant. Also, when adding an additional independent variable, the RSS goes down (which is what delivers an increase in the standard R^2). What you have in the numerator is a cost / benefit analysis. In other words, if the decrease in RSS is greater - then the $\bar{R}^2$ increases and the independent variable of question *might be somewhat important.* However, if the decrease in $N-k-1$ is greater, then the $\bar{R}^2$ decreases and the independent variable of question is *not important.*

Conclusion: for informal use only!

While the R^2 and adjusted R^2 are two common measures of goodness of fit, they are informal at best. One can interpret them along the lines of how we did above, but there will more formal measures of whether or not an independent variable improves the forecasts of the regression model. Bottom line: these measures can give some insight to the results of a regression model, but they aren't anything worth hanging your final conclusions on.

8.3 Statistical Inference

The course officially discusses statistical inference using the multiple regression model as opposed to the simple regression model, so this section should contain everything that is needed. If any preliminary material desired, there is an appendix to Chapter 7 that discusses statistical inference specifically with respect to the simple (one independent variable) regression model. One might also want to briefly review the chapters of statistical inference from MBA 8370 (i.e., Chapters 5 and 6).

8.3.1 Recalling the Concept of Statistical Inference

Back in MBA 8370, we wanted to get an idea about the *parameters* of a *population* (i.e., the population mean μ and the population standard deviation σ), but only had concrete information on the *statistics* of a *sample* (i.e., the sample mean $\bar{X}$ and the sample standard deviation S). We were able to make probabilistic statements (i.e., *educated guesses*) concerning the population parameters given the sample statistics along the lines of *confidence intervals* and *hypothesis tests*.

- Confidence Intervals allowed us to make general statements concerning the range of values in which the population mean μ will reside given the characteristics of the sample ($\bar{X},\ S$) and a particular probability or *level of confidence* α.
- Hypothesis Tests allowed us to determine if nonarbitrary statements concerning the value of the population mean μ are consistent or inconsistent with the characteristics of the sample ($\bar{X},\ S$).

The same concept of statistical inference can be applied to regression models. A population regression model contains parameters such as the intercept, slope coefficients, and residual standard error (σ_{XY}).

$$PRF:\ Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + ... + \beta_k X_{ki} + \varepsilon_i$$

We would like to know these population parameters, but we won't know them for sure because we cannot analyze the population. Since we can only observe

a sample, we can estimate a sample regression model containing statistics such as the intercept, slope coefficients, and residual standard error (S_{XY}).

$$SRF: \ Y_i = \hat{\beta}_0 + \hat{\beta}_1 \ X_{1i} + \hat{\beta}_2 \ X_{2i} + ... + \hat{\beta}_k \ X_{ki} + e_i$$

Our statistical inference will again amount to using our sample characteristics to make probabilistic statements about our population parameters. Statistical inference will take the form of our familiar confidence intervals and hypothesis tests, and a new statistical inference tool of forecasting.

8.3.2 Confidence Intervals (around population parameters)

Recall our earlier formula for calculating a confidence interval in a univariate context:

$$Pr\left(\bar{X} - t_{(\frac{\alpha}{2}, df=n-1)}\frac{S}{\sqrt{n}} \leq \mu \leq \bar{X} + t_{(\frac{\alpha}{2}, df=n-1)}\frac{S}{\sqrt{n}}\right) = 1 - \alpha$$

We used the Central Limit Theorem (CLT) to ultimately state that $\bar{X}$ was drawn from a normal distribution with a mean of μ and standard deviation $\sigma/\sqrt{n}$ (but we only have S which makes this a t distribution). This line of reasoning is *very* similar to what we have with regression analyses.

First, $\hat{\beta}$ is an estimate of β just like $\bar{X}$ is an estimate of μ. However, the standard error of the sampling distribution of $\hat{\beta}$ is derived from the standard deviation of the residuals.

$$S_{\hat{\beta}} = \frac{S_{YX}}{\sum (X_i - \bar{X})^2}$$

with

$$S_{YX} = \sqrt{\frac{\sum e_i^2}{n-k-1}}$$

This means that we construct a *standardized* random variable from a t distribution with $n-k-1$ degrees of freedom, where k is the number of independent variables (or slope coefficients) in the regression model.[2]

$$t = \frac{\hat{\beta} - \beta}{S_{\hat{\beta}}}$$

[2]Note that the appendix in Chapter 7 states that the t distribution in the simple regression case has $n-2$ degrees of freedom. This is because there is only one independent variable in a simple regression, so $k = 1$ and $n - k - 1 = n - 2$.

We have already derived a confidence interval before, so we can skip to the punchline.

$$Pr\left(\hat{\beta} - t_{(\frac{\alpha}{2}, df=n-k-1)} S_{\hat{\beta}} \leq \beta \leq \hat{\beta} + t_{(\frac{\alpha}{2}, df=n-k-1)} S_{\hat{\beta}}\right) = 1 - \alpha$$

This is the formula for a confidence interval around the *population* slope coefficient β given the estimate $\hat{\beta}$ and the regression characteristics. It can also be written compactly as before.

$$\hat{\beta} \pm t_{(\frac{\alpha}{2}, df=n-k-1)} S_{\hat{\beta}}$$

Recall our regression explaining differences in house prices given information on house sizes and number of bedrooms.

```
REG <- lm(price ~ sqrft + bdrms, data = hprice1)
summary(REG)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-19.31	31.05	-0.6221	0.5355
sqrft	0.1284	0.01382	9.291	1.394e-14
bdrms	15.2	9.484	1.603	0.1127

Table 8.2: Fitting linear model: price ~ sqrft + bdrms

Observations	Residual Std. Error	R^2	Adjusted R^2
88	63.04	0.6319	0.6233

The information included in the regression summary is all that is needed for us to construct a 95 percent ($\alpha = 0.05$) confidence interval around the *population* slope coefficient β_1. In other words, we can build a range where the population slope between house price and size will reside with 95 percent confidence.

```
# Back out all of the needed information:
Bhat1 <- summary(REG)$coef[2,1]
SBhat1 <- summary(REG)$coef[2,2]
n <- length(residuals(REG))
k = 2

# Find the critical t-distribution values... same as before
AL <- 0.05
df <- n-k-1
tcrit <- qt(AL/2,df,lower.tail = FALSE)
```

```
# Use the formula... same as before
(LEFT <- Bhat1 - tcrit * SBhat1)
```

```
## [1] 0.1009495
```

```
(RIGHT <- Bhat1 + tcrit * SBhat1)
```

```
## [1] 0.1559229
```

$$Pr(0.101 \leq \beta_1 \leq 0.156) = 0.95$$

This states that while an increase in house size by one square foot will increase the house price by \$128 ($\hat{\beta}_1$) on average in the sample, we can also state that an increase in house size by one square foot will increase the house price on average *in the population* somewhere between \$101 and \$156 with 95% confidence.

While the code above showed you how to calculate a confidence interval from scratch as we did before, there is an easier (one-line) way in R:

```
confint(REG)
```

```
##                   2.5 %     97.5 %
## (Intercept) -81.0439924 42.4140009
## sqrft         0.1009495  0.1559229
## bdrms        -3.6575816 34.0539635
```

8.3.3 Hypothesis Tests

We are able to conduct hypothesis tests regarding the values of the population regression coefficients. For example:

$$H_0 : \beta_1 = 0 \quad vs. \quad H_1 : \beta_1 \neq 0$$

In the context of our house price application, this null hypothesis states that the population slope between house price and size is zero… meaning that there is *no* relationship between the two variables in the population.

Given the null hypothesis above, we follow the remaining steps laid out previously: we calculate a test statistic under the null, calculate a p-value, and conclude.

The test statistic under the null is given by

$$t = \frac{\hat{\beta}_1 - \beta_1}{S_{\hat{\beta}_1}}$$

and this test statistic is drawn from a t distribution with $n - k - 1$ degrees of freedom. Concluding this test is no more difficult that what we've done previously.

```
B1 = 0
(tstat <- (Bhat1 - B1)/SBhat1)
```

```
## [1] 9.290506
```

```
(Pval <- pt(tstat,df,lower.tail=FALSE)*2)
```

```
## [1] 1.393748e-14
```

```
(1-Pval)
```

```
## [1] 1
```

Our results state that we can reject this null hypothesis with approximately 100% confidence, meaning that there is a statistically significant relationship between house prices and house sizes. By *statistically significant*, we are essentially saying that the population relationship is some number other than zero.

As with the confidence interval exercise above, we actually do not need to conduct hypothesis tests where the null sets the population parameter to zero because R does this automatically. If you look again at the columns to the right of the estimated coefficient $\widehat{\beta}_1$ in the regression summary above, you will see a t value that is exactly what we calculated above and a p value that is essentially zero. This implies that a test with the null hypothesis set to zero is always done for you.

```
summary(REG)
```

```
##
## Call:
## lm(formula = price ~ sqrft + bdrms, data = hprice1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -127.627  -42.876   -7.051   32.589  229.003
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -19.31500   31.04662  -0.622    0.536
## sqrft         0.12844    0.01382   9.291 1.39e-14 ***
## bdrms        15.19819    9.48352   1.603    0.113
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 63.04 on 85 degrees of freedom
```

```
## Multiple R-squared:  0.6319, Adjusted R-squared:  0.6233
## F-statistic: 72.96 on 2 and 85 DF,  p-value: < 2.2e-16
```

This isn't to say that *all* hypothesis tests are automatically done for you.

Suppose a realtor believes that homes sell for \$150 per square foot. This is a non-arbitrary statement on a population parameter that delivers the following hypotheses, followed by a test statistic, p-value, and conclusion.

$$H_0 : \beta_1 = 0.150 \quad vs. \quad H_1 : \beta_1 \neq 0.150$$

```
B1 = 0.150
(tstat <- (Bhat1 - B1)/SBhat1)
```

```
## [1] -1.559829
```

```
(Pval <- pt(tstat,df)*2)
```

```
## [1] 0.122516
```

```
(1-Pval)
```

```
## [1] 0.877484
```

Our p-value implies that there is a 12 percent chance of being wrong if we reject the null hypothesis. In other words, we can reject the null with at most 88 percent confidence. We therefore do not have evidence that the population slope is different from 0.150 with any traditional level of confidence (e.g., $\alpha \leq 0.10$).

One-sided tests are also like before. We can consider *right-tailed* tests where the rejection region (and p-value) are in the right tail, as well as *left-tailed* tests where the rejection region (and p-value) are in the left tail. Let us examine one of each.

Suppose a realtor believes that homes sell for *more than* \$120 per square foot.[3] Since we can lend statistical support to this claim by rejecting everything else, This delivers the following hypotheses, which gives rise to a **right-tailed** test.

$$H_0 : \beta_1 \leq 0.120 \quad vs. \quad H_1 : \beta_1 > 0.120$$

We calculate a test statistic under the null as always. But since this is a right-tailed test, we calculate the p-value (and conclusion) by always calculating the area to the **right** of the test statistic.

```
B1 = 0.120
(tstat <- (Bhat1 - B1)/SBhat1)
```

```
## [1] 0.610238
```

[3] Note that while the estimate from our sample is greater than 0.120, the statement we are testing is regarding what is going on in the population.

```
(Pval <- pt(tstat,df,lower.tail = FALSE))
```

```
## [1] 0.2716661
```

```
(1-Pval)
```

```
## [1] 0.7283339
```

Our test concludes that we can reject the null with at most 73 percent confidence.

Suppose a different realtor believes that homes sell for *less than* \$130 per square foot.[4] Since we can lend statistical support to this claim by rejecting everything else, This delivers the following hypotheses, which gives rise to a **left-tailed** test.

$$H_0 : \beta_1 \geq 0.130 \quad vs. \quad H_1 : \beta_1 < 0.130$$

We calculate a test statistic under the null as always. But since this is a left-tailed test, we calculate the p-value (and conclusion) by always calculating the area to the **left** of the test statistic.

```
B1 = 0.130
(tstat <- (Bhat1 - B1)/SBhat1)
```

```
## [1] -0.1131176
```

```
(Pval <- pt(tstat,df,lower.tail = TRUE))
```

```
## [1] 0.455102
```

```
(1-Pval)
```

```
## [1] 0.544898
```

Our test concludes that we can reject the null with at most 54 percent confidence.

8.3.4 Confidence Intervals (around forecasts)

A regression can also build confidence intervals around the conditional expectations (i.e., forecasts) of the dependent variable.

Suppose you want to use our model to predict the price of a 1000 square foot house with 3 bedrooms. The conditional expectation is calculated by using our regression coefficients, a value of house size of 1000, a value of bedrooms of 3, and setting our forecast error to zero.

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 \, 1000 + \hat{\beta}_2 \, 3$$

[4] Note that while the estimate from our sample is less than 0.130, the statement we are testing is regarding what is going on in the population.

```
Bhat0 = summary(REG)$coef[1,1]
Bhat1 = summary(REG)$coef[2,1]
Bhat2 = summary(REG)$coef[3,1]

(Yhat = Bhat0 + Bhat1 * 1000 + Bhat2 * 3)
```

```
## [1] 154.7158
```

This calculation suggests that we expect a house with 1000 square feet and 3 bedrooms to sell for approximately 154.716 (thousand) dollars. Another way to calculate this forecast is using the predict command in R. This command creates a new data frame that includes only the value for the independent variable you want to make a prediction with. The rest is done for you.

```
predict(REG,data.frame(sqrft = 1000, bdrms = 3))
```

```
##        1
## 154.7158
```

While this is an expected value based on the sample, we need to appreciate that we want to see what the prediction is in the population. We are able to build a confidence interval around this forecast in a number of ways.

- A confidence interval for the mean response
- A confidence interval for an individual response

The mean response: a confidence interval

Suppose you want to build a confidence interval around the mean price for a 1000 square foot house with 3 bedrooms in the population. This is a conditional mean. In other words, we want the average house price but *only* for homes with a particular size. This conditional mean is generally given by $\mu_{Y|X}$ and in this case by $\mu_{Y|X_1=1000,\ X_2=3}$. Building a confidence interval for the mean response is given by

$$\hat{Y}_X \pm t_{(\frac{\alpha}{2},df=n-k-1)} S_{YX} \sqrt{h_i}$$

or

$$\hat{Y}_X - t_{(\frac{\alpha}{2},df=n-k-1)} S_{YX} \sqrt{h_i} \leq \mu_{Y|X} \leq \hat{Y}_X + t_{(\frac{\alpha}{2},df=n-k-1)} S_{YX} \sqrt{h_i}$$

where

- $\hat{Y}_X$ is the expectation of the dependent variable conditional on the desired value of X.
- S_{YX} is the standard error of the estimate (calculated previously)

- $t_{(\frac{\alpha}{2},df=n-k-1)}$ is the critical t statistic for a given value of α (calculate previously)
- $h_i = \frac{1}{n} + \frac{(X_i - \bar{X})^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}$

This last variable h_i is what is new to us and increases the size of the confidence interval when the desired value of X_i is farther away from the average value of the observations $\bar{X}$. This variable can sometimes be difficult to calculate, but R again does it for you.[5] In R, a confidence interval around the population mean is simply called a *confidence* interval.

```
predict(REG,
        data.frame(sqrft = 1000,bdrms = 3),
        interval = "confidence",
        level = 0.95)
```

```
##        fit      lwr      upr
## 1 154.7158 127.2862 182.1454
```

$$Pr(127.29 \leq \mu_{Y|X} \leq 182.15) = 0.95$$

We can now state with 95% confidence that the *population mean house price* of all 1000 square-foot houses with 3 bedrooms is somewhere between $127,290 and $182,150. Note that the confidence interval around the mean response is centered at our conditional expectation ($\hat{Y}$) just like all confidence intervals are centered around its estimate.

An individual response: a prediction interval

Suppose that instead of building a confidence interval around the conditional average in the population, we want to determine the range within which we are confident to draw a *single* home value. This calculation is almost identical to the mean response above, but with one slight difference.

$$\hat{Y}_X \pm t_{(\frac{\alpha}{2},df=n-k-1)} S_{YX} \sqrt{1 + h_i}$$

or

$$\hat{Y}_X - t_{(\frac{\alpha}{2},df=n-k-1)} S_{YX} \sqrt{1 + h_i} \leq Y_X \leq \hat{Y}_X + t_{(\frac{\alpha}{2},df=n-k-1)} S_{YX} \sqrt{1 + h_i}$$

where

[5] Note that this equation is provided for only one independent variable. It becomes even more messy in a multivariate setting. However, the important concept is that this value gets larger when we consider values of X that are farther away from the average values in the sample.

- $\hat{Y}_X$ is the expectation of the dependent variable conditional on the desired value of X_i.
- S_{YX} is the standard error of the estimate (calculated previously)
- $t_{(\frac{\alpha}{2}, df=n-k-1)}$ is the critical t statistic for a given value of α (calculate previously)
- $h_i = \frac{1}{n} + \frac{(X_i - \bar{X})^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}$

The only difference is that we replace $\sqrt{h_i}$ with $\sqrt{1 + h_i}$. Conceptually, we inserted the one in the formula because we are selecting a *single* home with a specified size and number of bedrooms out of the population. This is very different from building a confidence interval around a population mean, but in R it is simply the change of one word.

```
predict(REG,
        data.frame(sqrft = 1000, bdrms = 3),
        interval = "prediction",
        level = 0.95)
```

```
##        fit      lwr      upr
## 1 154.7158 26.39973 283.0318
```

$$Pr(26.40 \leq Y_X \leq 283.03) = 0.95$$

We can now state with 95% confidence that *a single draw of a house price* from the population of all 1000 square-foot houses will be somewhere between \$26,400 and \$283,030. Note that the prediction interval is also centered at our conditional expectation ($\hat{Y}$), but now the interval is much wider than in the previous calculation. This should make sense, because when you are selecting a single home then you have a positive probability of selecting either very cheap homes or very expensive homes. A mean would wash these extreme values out.

Chapter 9

Collinearity

We showed in the previous chapter that one wants to include correlated variables into the same regression in order for us to *control* for particular sample characteristics. For example, the size of a house and the number of bedrooms in the house are correlated, so including them both in the same regression that focuses on explaining house price was a good thing.

However, a significant problem in regression analysis arises when two more independent variables are *too* correlated with each other. This is known as collinearity (or multicollinearity). A correlation means that two or more variables systematically move together. In regression analysis, movement is *information* that we use to explain differences or changes in the dependent variable. If independent variables have the exact same movements due to large correlations, then they contain similar (i.e., redundant) information.

Another issue with collinearity is that when two or more variables systematically move together, then it goes against the very interpretation of our estimates: *holding all else constant* If the variables aren't held constant in the data due to collinearity (i.e., they are always moving systematically with each other), then our estimates cannot differentiate the impact of these variables along separate dimensions. Since the information from these independent variables are shared and redundant, then the dimensions from these collinear variables becomes blurred.

9.1 An Application

Consider an application that compares simulated data where two independent variables have different degrees of correlation. The simulated data was generated from the following model:

$$Y_i = 1 + 1\ X_{1i} + 1\ X_{2i} + \varepsilon_i$$

In other words, the simulated data **should** return the same coefficients above **if** there are no problems with the estimation. The exercise will show you how collinearity can become a problem.

```
# 1) Regression: correlation = 0.3289
cor(MDAT$X31,MDAT$X32)
```

```
## [1] 0.3289358
```

```
CREG <- lm(Y3~X31+X32,data=MDAT)
coeftest(CREG)
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 1.0015735  0.0214445  46.705 < 2.2e-16 ***
## X31         1.0152385  0.0401048  25.315 < 2.2e-16 ***
## X32         0.9905016  0.0099267  99.781 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# 2) Regression: correlation = 0.938
cor(MDAT$X21,MDAT$X22)
```

```
## [1] 0.9380521
```

```
CREG <- lm(Y2~X21+X22,data=MDAT)
coeftest(CREG)
```

```
##
## t test of coefficients:
##
##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 1.001574   0.021445  46.705 < 2.2e-16 ***
## X21         1.100724   0.109304  10.070 < 2.2e-16 ***
## X22         0.905016   0.099267   9.117 1.082e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# 3) Regression: correlation = 0.999
cor(MDAT$X11,MDAT$X12)
```

```
## [1] 0.9992777
```

```
CREG <- lm(Y1~X11+X12,data=MDAT)
coeftest(CREG)
```

```
##
## t test of coefficients:
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.001574   0.021445 46.7053  < 2e-16 ***
## X11         1.955579   0.996657  1.9621  0.05261 .
## X12         0.050161   0.992672  0.0505  0.95980
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# 4) Regression: Highest correlation = 1
cor(MDAT$X41,MDAT$X42)
```

```
## [1] 1
```

```
CREG <- lm(Y4~X41+X42,data=MDAT)
coeftest(CREG)
```

```
##
## t test of coefficients:
##
##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 1.003483   0.021342  47.019 < 2.2e-16 ***
## X41         2.002616   0.037857  52.900 < 2.2e-16 ***
## X42               NA         NA      NA        NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The above application considers four sets of data where the only difference is the degree of collinearity between the two independent variables. The first regression has a degree of correlation between X_{1i} and X_{2i} equal to 0.33, and you will see that the regression does a fairly good job at recovering the regression coefficients. The second regression has a degree of correlation between X_{1i} and X_{2i} equal to 0.94, and you will see that the regression is beginning to suffer a bit where both slope estimates are now off by about 10 percent. The Third regression has a degree of correlation between X_{1i} and X_{2i} equal to just shy of perfect (1), and you will see that the regression is now *way* off from the expected estimates. Finally, the fourth regression has **perfect collinearity** between X_{1i} and X_{2i}, and the regression actually chokes by providing an *NA* (meaning, not a number) as an answer for the second coefficient. Mathematically, perfect collinearity asks for a computer to divide a number by zero (which computers don't like to do).

9.2 What does Collinearity do to our regression?

The takeaway from our application is that collinearity can become a significant problem if the degree of correlation among the independent variables is large enough. What the application does not show is that collinearity also results in

excessively large standard errors of the coefficient estimates. Intuitively, if the regression doesn't know which variable is providing the (redundant) information, then it shows this by placing little precision on the estimate - meaning an excessively large standard deviation. This standard deviation is *positively biased* - meaning that it is larger due to the presence of collinearity. This artificially large standard error will impact the significance of estimates via confidence intervals and hypothesis tests.

9.3 How to test for Collinearity?

Note that most variables are correlated to some degree (even if completely at random). Therefore, the question is really how much collinearity exists in our data? Is it not enough so we can disregard (as in the first example in the previous section) or enough to cause issues (as in the third or fourth example)?

There are two data characteristics that help detect the degree of collinearity in a regression:

- High simple correlation coefficients
- High Variance Inflation Factors (VIFs)

Correlation Coefficients

$$Cov(X_1, X_2) = \frac{1}{n-1}\sum_{i=1}^{n}(X_{1i} - \bar{X}_1)(X_{2i} - \bar{X}_2)$$

$$S_{X_1} = \frac{1}{n-1}\sum_{i=1}^{n}(X_{1i} - \bar{X}_1)^2$$

$$S_{X_2} = \frac{1}{n-1}\sum_{i=1}^{n}(X_{2i} - \bar{X}_2)^2$$

$$\rho(X_1, X_2) = \frac{Cov(X_1, X_2)}{S_{X_1} S_{X_2}}$$

If a simple correlation coefficient between any two explanatory variables, $\rho(X_1, X_2)$, is high in absolute value, then collinearity is a potential problem. Like we saw in the application, high is rather arbitrary. Therefore, researchers settle on a threshold of 0.80. In other words, if you have a correlation of 0.80 or higher, then you are running the risk of having your estimates biased by the existence of collinearity.

The problem with looking at simple correlations is that they are only *pairwise* calculations. In other words, you can only look at two variables at a time. What if a collinearity problem is bigger than just two variables?

Variance Inflation Factors (VIFs)

Suppose you want to estimate a regression with three independent variables, but you want to test for collinearity first.

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_{3i} + \varepsilon_i$$

Correlation coefficients, being pairwise, will not be able to uncover a correlation structure that might exist across *all three* independent variables.

Take for example three independent variables: a pitcher's ERA, the number of earned runs, and the number of innings pitched. For those of you (like me) who are unfamiliar with baseball, a pitcher's ERA is essentially, their earned runs divided by the number of innings pitched. This means that ERA might be positively correlated with earned runs and negatively correlated with innings pitched, but you wouldn't realize that the correlation is *perfect* (meaning, equal to 1) unless you consider both variables *simultaneously* - and correlation coefficients cannot look at this issue simultaneously. A Variance Inflation Factor (or VIF) is a method for examining a complete correlation structure on a list of three or more independent variables.

A Variance Inflation Factor (VIF) is calculated in two steps:

First, run an OLS regression where an independent variable (say, X1) takes a turn at being a dependent variable.

$$X_{1i} = a_0 + a_1 X_{2i} + a_2 X_{3i} + u_i$$

Note that the original dependent variable (Y_i) is NOT in this equation!

The purpose of this auxiliary regression is to see if there is a sophisticated correlation structure between X_{1i} and the right-hand side variables. Conveniently, we already have an R^2 which will indicate exactly how much the variation in the left-hand variable is *explained* by the right-hand variables.

The second step takes the R^2 from this regression and calculates the VIF for independent variable X_{1i}. Since the VIF impacts the estimated coefficient of β_1 in the original regression, it is sometimes referred to as $VIF(\hat{\beta}_1)$:

$$VIF(\hat{\beta}_1) = \frac{1}{1 - R^2}$$

If we did this for every independent variable in the original regression, we would arrive at three VIF values.

$$X_{1i} = a_0 + a_1 X_{2i} + a_2 X_{3i} + u_i \rightarrow VIF(\hat{\beta}_1) = \frac{1}{1 - R^2}$$

$$X_{2i} = a_0 + a_1\, X_{1i} + a_2\, X_{3i} + u_i \rightarrow VIF(\hat{\beta}_2) = \frac{1}{1 - R^2}$$

$$X_{3i} = a_0 + a_1\, X_{1i} + a_2\, X_{2i} + u_i \rightarrow VIF(\hat{\beta}_3) = \frac{1}{1 - R^2}$$

These VIF values will deliver the amount of bias in each of the standard errors of the estimated coefficients due to the presence of collinearity. For example, if a VIF number is 2, then this means that the degree of collinearity will result in a standard error that is *twice* as large as it would have been without collinearity. In order to determine if there is a problem, we again resort to an arbitrary threshold of $VIF \geq 5$. Note that since an R^2 value is comparable to a correlation coefficient, this VIF measure corresponds to a correlation above 0.8.

9.3.1 An Application:

```
library(readxl)
MULTI2 <- read_excel("data/MULTI2.xlsx")
names(MULTI2)
```

```
## [1] "Team"          "League"        "Wins"          "ERA"
## [5] "Runs"          "Hits_Allowed"  "Walks_Allowed" "Saves"
## [9] "Errors"
```

Suppose that you want to explain why some baseball teams recorded more wins than others by looking at the season statistics listed above. Before we run a full regression with *Wins* as the dependent variable and the other right variables as independent variables, we need to test for collinearity.

If we were to follow the steps above for each independent variable, we will need to calculate seven VIF values (Team isn't a variable... it's a name). This is a lot easier done than said in R:

```
# Estimate the 'intended' model:
REG <- lm(Wins ~ League + ERA + Runs + Hits_Allowed +
            Walks_Allowed + Saves + Errors, data = MULTI2)

# Use REG object to determine the VIFS:
library(car)
vif(REG)
```

```
##        League           ERA          Runs  Hits_Allowed Walks_Allowed
##      1.221101     11.026091      1.279997      6.342662      3.342659
##         Saves        Errors
##      1.762577      1.548678
```

The output above shows a VIF for each of the independent variables. The largest are for ERA and Hits Allowed, and these are problematic given that they are

above our threshold of 5.[1] So now that we detected collinearity… what do we do about it?

9.4 How do we remove Collinearity?

There are several ways to remove or reduce the degree of collinearity that vary in degrees of feasibility and effectiveness.

First, is the collinearity problem due to the inherent nature of the variables themselves or is it a coincidence with your current sample? If it is coincidence, then the problem might go away if you collected more observations. Note that this might not always work, and sometimes more data isn't even available. However, it is a easy first pass if feasible.

Second, one could always **ignore** collinearity and proceed with the analysis. The reason for this is that while collinearity might bias the standard errors of the estimates, the bias might not be that bad. Think of increasing the value of zero by 100 times.

For example, lets try the ignorance approach with the baseball application above:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	69.28	13.64	5.077	4.37e-05
League	1.847	1.012	1.825	0.08168
ERA	-6.058	3.441	-1.76	0.09225
Runs	0.08855	0.007688	11.52	8.703e-11
Hits_Allowed	-0.02523	0.01411	-1.788	0.08761
Walks_Allowed	-0.02665	0.01178	-2.262	0.03393
Saves	0.5378	0.07606	7.071	4.297e-07
Errors	0.004109	0.04188	0.0981	0.9227

Table 9.2: Fitting linear model: Wins ~ League + ERA + Runs + Hits_Allowed + Walks_Allowed + Saves + Errors

Observations	Residual Std. Error	R^2	Adjusted R^2
30	2.503	0.9611	0.9488

The results suggest that the population coefficients for the variables League, ERA, Hits Allowed, and Errors are all insignificantly different from zero with

[1] Note that the handy command *vif* is located in the *car* package. That is why we needed to open the car package using the library command. See the chapter on R basics for more details.

95% confidence. Now if they were all significant, then we could possibly ignore any potential collinearity issues because the bias would not be *enough* for us to see if there was a problem. However, since two of these insignificant variables are ones we already identified as having a collinearity problem, then we are unable to go this route.

The third option for removing collinearity is to remove the correlated independent variables until the correlation structure is removed. The way to proceed down this route is to remove the variables (one-at-a-time) with the highest VIF values first until all remaining values have VIF values below 5. The good side of this analysis is that you can now proceed with the main regression knowing that collinearity is not a problem. The bad side is that you might have had to remove variables that you really wanted to have in the regression.

The VIF values from the baseball analysis suggest that ERA and Hits Allowed are two variables that potentially need to be removed from the analysis due to collinearity. The way to proceed is that if we were to only remove one variable at a time, we will remove the variable with the *highest* VIF because it is the one that has the most redundant information.

```
REG <- lm(Wins ~ League + Runs + Hits_Allowed + Walks_Allowed + Saves + Errors, data =

vif(REG)
```

```
##        League          Runs  Hits_Allowed Walks_Allowed         Saves
##      1.149383      1.279914      1.365583      1.235945      1.665172
##        Errors
##      1.546465
```

```
summary(REG)
```

```
##
## Call:
## lm(formula = Wins ~ League + Runs + Hits_Allowed + Walks_Allowed +
##     Saves + Errors, data = MULTI2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8127 -2.0776  0.0551  2.0168  4.9951
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   83.214595  11.607524   7.169 2.67e-07 ***
## League         2.278948   1.026010   2.221   0.0365 *
## Runs           0.088445   0.008031  11.013 1.20e-10 ***
## Hits_Allowed  -0.047231   0.006840  -6.905 4.86e-07 ***
## Walks_Allowed -0.043122   0.007485  -5.761 7.22e-06 ***
## Saves          0.569301   0.077227   7.372 1.69e-07 ***
```

```
## Errors          0.001322   0.043722   0.030   0.9761
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.615 on 23 degrees of freedom
## Multiple R-squared:  0.9557, Adjusted R-squared:  0.9441
## F-statistic: 82.65 on 6 and 23 DF,  p-value: 2.119e-14
```

The regression with ERA removed now is free of collinearity. We can confirm this by the fact that all VIF values of the remaining independent variables are well below 5. The regression results suggest that after removing ERA, ERA and Hits Allowed now have population coefficients that were significantly different than zero with 95% confidence. Errors is still an insignificant variable. This suggests that the insignificance wasn't due to collinearity. It's simply the fact that Errors do not significantly help us explain why some teams win more games than others.

Sometimes removing collinearity might involve multiple rounds

You will note from the application above that we only needed to remove one independent variable, so only one round of VIF calculations displayed values above 5. It might sometimes be the case that even after you remove an independent variable, the next round of VIF values reports reports one or more with value of 5 or more. If this happens, you simply repeat the process by removing the variable with the highest VIF and check again. In general, a complete removal of multicollinearity involves the following:

1. calculate VIFs for your data set
2. drop the variable with the highest VIF (greater than 5)
3. calculate VIFs on your data again (with the dropped variable no longer in the data set)
4. drop the variable with the highest VIF (greater than 5)
5. this is repeated until all VIFs are less than 5

Chapter 10

Qualitative (Dummy) Variables

Quantitative variables are easy to model and interpret because they take on numerical values and are readily dealt with by computers. Qualitative variables, however, are variables that do not naturally deliver numerical values. In other words, they are more like *categories*. Examples of qualitative variables are:

- Gender (male, female)
- Marital status (yes, no)
- Ethnicity (white, Hispanic, Asian, etc.)

Qualitative variables are made operational for regression analysis by creating **dummy variables**. A dummy variable can only take on two values (i.e., 0 or 1) and should be thought of as a *switch*.

- 1 implies the switch is *on*, meaning that the designated trait is present for an individual observation.
- 0 implies the switch is *off*, meaning that the trait is absent for an individual observation.

We can consider two different types of dummy variables depending on if we model the presence or absence of a trait to impact the intercept of the model or the relationship (or slope) between the dependent variable and other independent variables. We will cover these in turn.

10.1 Intercept dummy variable

An intercept dummy variable is a qualitative variable that *stands alone* in a regression just like other quantitative variables we have encountered. Let us

illustrate this by adding an intercept dummy variable to a wage analysis.

Suppose you are a consultant hired by a firm to help determine the underlying features of the current wage structure for their employees. You want to understand why some individuals have wage rates that are different from others. Let our dependent variable be *wage* (the hourly wage of an individual employee) and the independent variables be given by…

- *educ* is the total years of education of an individual employee
- *exper* is the total years of experience an individual employee had prior to starting with the company
- *tenure* is the number of years an employee has been working with the firm.

These independent variables are all *quantitative* because they directly translate to numbers. We can also add a qualitative variable to this list of independent variables to see if gender can help explain why some people earn a higher wage than others. In particular, consider the qualitative variable *female* which equals 1 if the individual is female and 0 if the individual is not (i.e., male).

The Specified model (the PRF) now becomes

$$wage_i = \beta_0 + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \beta_4 female_i + \varepsilon_i$$

Note that the slope of the three quantitative variables are completely standard. The slope with respect to the dummy variable is similar, but needs to be interpreted in a specific manner. In particular, since we normally interpret slopes with respect to a *unit increase* in the independent variable, and the fact that a dummy variable can only go up one unit (i.e., from 0 to 1), we therefore interpret a dummy variable accordingly.

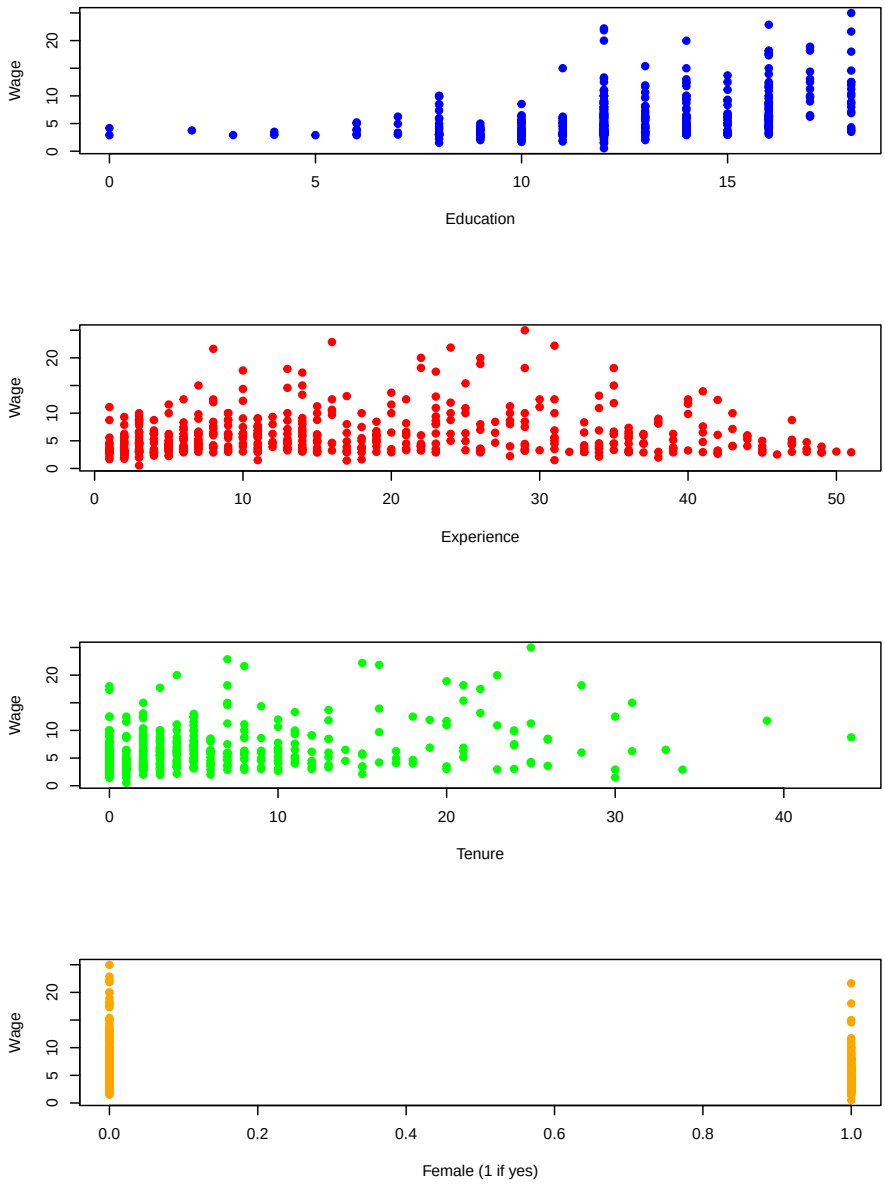
$$\beta_4 = \frac{\Delta wage}{\Delta female}$$

> Holding education, tenure, and experience constant, a female earns a β_4 difference in wage relative to a male, on average

Note that the dummy variable is constructed such that males receive a 0 while females receive a 1. This implies that β_4 will denote the average change in a female's wage *relative* to a male's wage. If $\beta_4 < 0$, then this would imply that a female's average wage is less than a male's.

After loading the *wage1* data directly from the *wooldridge* package (see code below), the four independent variables are illustrated in the scatter plots below. Notice that even though the dummy variable takes on only two numbers by design, we can still see how it effectively *splits* the observations into the two groups.

```
data(wage1, package = "wooldridge")

par(mfrow = c(4,1))
plot(wage1$educ,wage1$wage,
          col = "blue", pch = 19, cex = 1,
          xlab = "Education", ylab = "Wage")
plot(wage1$exper,wage1$wage,
          col = "red", pch = 19, cex = 1,
          xlab = "Experience", ylab = "Wage")
plot(wage1$tenure,wage1$wage,
          col = "green", pch = 19, cex = 1,
          xlab = "Tenure", ylab = "Wage")
plot(wage1$female,wage1$wage,
          col = "orange", pch = 19, cex = 1,
          xlab = "Female (1 if yes)", ylab = "Wage")
```



There is no difference between estimating quantitative and qualitative variables as far as R in concerned.

```
REG <- lm(wage~educ+exper+tenure,data=wage1)
coef(REG)
```

```
## (Intercept)        educ       exper      tenure
## -2.87273482  0.59896507  0.02233952  0.16926865
```

```
REG <- lm(wage~educ+exper+tenure+female,data=wage1)
coef(REG)
```

```
## (Intercept)        educ       exper      tenure      female
## -1.56793870  0.57150477  0.02539587  0.14100506 -1.81085218
```

Interpretations of the other independent variables are unchanged. However, $\hat{\beta}_4 = -1.81$ suggests the following:

> Holding education, tenure, and experience constant, a female earns $1.81 less in wages relative to a male, on average

This states that we can compare two individuals with the same education, experience, and tenure levels but differ in gender and conclude that the male earns more.

Let us examine this further to show exactly why this type of qualitative variable is called an *intercept dummy variable*. Since the dummy variable can only take on the values 1 or 0, we can write down the PRF for both cases. In particular, the PRF for a male has $female_i = 0$ while the PRF for a female has $female_i = 1$.

$$PRF: wage_i = \beta_0 + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \beta_4 female_i + \varepsilon_i$$

$$Male: wage_i = \beta_0 + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \varepsilon_i$$

$$Female: wage_i = (\beta_0 + \beta_4) + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \varepsilon_i$$

Notice that β_4 does not appear in the PRF for males because the female variable equals 0, while it appears *alone* in the PRF for females because the female variable equals 1. After rearranging a bit, you can see that the intercept term of the PRF for males is β_0 while the intercept term of the PRF for females is $(\beta_0 + \beta_4)$. This illustrates that if you hold the other three independent variables constant, the difference between the wage rates of a male and female is β_4 on average. In other words, if you plug in the same numbers for education, experience, and tenure in the two PRFs above, then the difference in wages between men and women who share these traits will be β_4.

10.2 Slope dummy variable

While an intercept dummy variable is a very powerful modeling tool, it makes one glaring assumption. Consider the regression results above, namely the estimated slope coefficient with respect to tenure

$$\hat{\beta}_3 = 0.14$$

The interpretation of this slope coefficient is as follows:

> Holding education, experience, and gender constant, an individual will receive \$0.14 more in wages for every additional year of tenure, on average.

In particular, holding gender constant implies that a female receives the same annual raise than a male. This is an *assumption* of the model, because the model is incapable of differentiating the annual wage with respect to gender. Do men and women get significantly different average annual raises? We can extend the model to explicitly test this assumption with the use of a *slope dummy variable*.

A slope dummy variable is an example of an *interaction term*. In other words, it is a new variable that arises from taking the product of two variables. In this case, in order for us to examine the gender difference of tenure, we consider the product between female and tenure.

$$\begin{aligned} wage_i = \beta_0 + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \beta_4 female_i + ... \\ \beta_5 (tenure_i * female_i) + \varepsilon_i \end{aligned}$$

Like our illustration of an intercept dummy, we can see what this PRF looks like for males and females.

$$Male: wage_i = \beta_0 + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \varepsilon_i$$

$$Female: wage_i = (\beta_0 + \beta_4) + \beta_1 educ_i + \beta_2 exper_i + (\beta_3 + \beta_5) tenure_i + \varepsilon_i$$

For males, the PRF looks *exactly* as it does when we only considered an intercept dummy because both β_4 and β_5 drop out when $female_i = 0$. For females, we can see the potential change in the intercept (as before), but we can now see a potential change in the slope with respect to tenure.

```
REG <- lm(wage~educ+exper+tenure+
            female+female*tenure,data=wage1)
coef(REG)
```

```
##   (Intercept)          educ         exper        tenure        female
##   -2.00229568    0.58279061    0.02834532    0.17780235   -1.17787884
## tenure:female
##   -0.14359567
```

Our extended model now gives a better picture of the gender impact on wages.

$$\hat{\beta}_4 = -1.18$$

> Holding all else constant, a female earns \$1.18 less than a male on average.

When considering the impact of tenure on wages, we could show the difference explicitly:

$$Males: \frac{\Delta wage}{\Delta tenure} = \hat{\beta}_3 = 0.18$$

$$Females: \frac{\Delta wage}{\Delta tenure} = \hat{\beta}_3 + \hat{\beta}_5 = 0.18 - 0.14 = 0.04$$

The regression states that males receive an \$0.18 increase in wages on average for every additional year in tenure (holding all else constant), while females receive only a \$0.04 increase in wages on average.

Note that we could also consider slope dummy variables with respect to education as well as experience. You should do those on your own.

10.3 What if there are more than two categories?

Since a dummy variable can take on either a zero or a one, it is perfectly designed to identify two categories. This might be fine for some variables like yes / no or win / lose, but what if a variable has more than two categories? Examples would be direct extensions of the above variables: yes / no / maybe or win / lose / draw.

The rule of thumb (to be explained in detail soon) is:

A variable containing N categories requires $N-1$ dummy variables.

This rule actually applies to our standard case, because we can model $N = 2$ categories with $N - 1 = 1$ dummy variables. In our example above, we wanted to identify 2 categories of gender (male or female) so we needed 1 dummy variable. However, we need to take a little more care and follow additional steps when dealing with more than two categories. Suppose we extended our gender characteristics to identify a third gender category (*nonbinary*) in order to account for individuals who do not subscribe to one of the two traditional categories. We will use this scenario to illustrate how our model gets extended.

1. Identify a *benchmark* category

A benchmark category is one of the characteristics that the researcher identifies as the category that all other categories get compared against. In our gender example, suppose we choose *male* as our benchmark characteristic. You will find that this choice is arbitrary, but it may have implications.

2. Construct appropriate dummy variables

Once the benchmark category has been established as male, we need two dummy variables: one that identifies individuals as *female* and one that identifies individuals as *nonbinary*.

$$female_i = 1 \text{ if female; } 0 \text{ if male or nonbinary}$$

$$\text{non-binary}_i = 1 \text{ if nonbinary; } 0 \text{ if male or female}$$

Note that each dummy variable is still a switch that signals the presence or absence of a characteristic. However, when **BOTH** dummy variables are zero at the same time… you have your benchmark category. That is how you can identify three categories with only two dummy variables.

To illustrate, consider the original model restricting attention to intercept dummies.

$$wage_i = \beta_0 + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \beta_4 female_i + \beta_5 (nonbinary_i) + \varepsilon_i$$

We can write down what the model looks like for each of our three categories:

$$Male : wage_i = \beta_0 + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \varepsilon_i$$

$$Female : wage_i = (\beta_0 + \beta_4) + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \varepsilon_i$$

$$\text{Non-binary} : wage_i = (\beta_0 + \beta_5) + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \varepsilon_i$$

When comparing these three equations, you can hopefully see how the benchmark category comes into play. The first equation is essentially the benchmark equation, indicating that β_0 is the intercept term for males. The second equation is for females, and shows how the intercept for females differs from males (given by β_4). The third equation is for those identifying as nonbinary, and shows how the intercept for these individuals differs from males (given by β_5). Note that all of the other slopes are *assumed* to be identical here (but we could consider slope dummies like above).

One detail worth mentioning about the application above is that the coefficients β_4 and β_5 show how each category compares to the benchmark category. We can test if these coefficients are significantly different from zero with standard hypothesis tests. For example:

$$H_0 : \beta_4 = 0 \quad H_1 : \beta_4 \neq 0$$

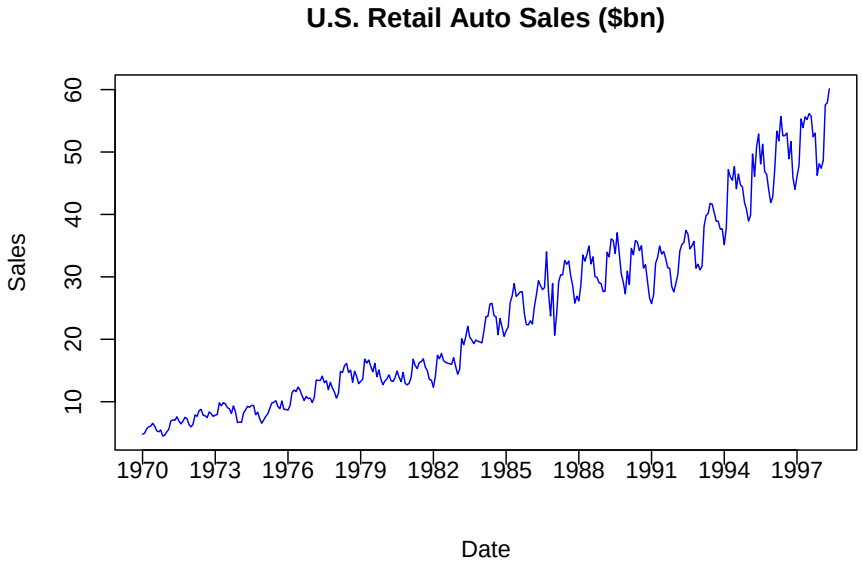
However, if we show that β_4 and β_5 were significantly different than zero, we can only conclude that females and nonbinary individuals are treated differently than males (because it was the benchmark category). We cannot determine if *female* and *nonbinary* are significantly different from each other without a joint hypothesis test (examined below) or a choice of a new benchmark category. For example, you can easily change the benchmark category to be female and end up with a formal test of the difference between female and nonbinary.

10.4 A Final Application

Let us consider an in-depth application where dummy variables are essential for making time-series variables ready for analysis by removing a *seasonal cycle*.

Consider the following time series data:

```
library(readxl)
AUTO <- read_excel("data/AUTO_SA.xlsx")

plot(AUTO$INDEX,AUTO$AUTOSALE,
     type = "l", main = "U.S. Retail Auto Sales ($bn)",
     col = "blue", xlab = "Date", ylab = "Sales", xaxt = "n")
xtick <- seq(1,length(AUTO$INDEX), by = 36)
axis(side=1, at=xtick, labels = FALSE)
text(x=xtick, par("usr")[3],
     labels = c("1970","1973","1976","1979","1982","1985",
                "1988","1991","1994","1997"),
     pos = 1, xpd = TRUE)
```



The figure illustrates retail automobile sales, denoted in billions of dollars, for the US between 1970 and 1998. As with most time series, this data is actually a combination of several components.

- **Trend**: The long-term (i.e., average) increase or decrease in value over time.
- **Seasonality**: The repeating (i.e., predictable) short-term cycle in the series caused by the seasons or months of the year
- **Random**: The information in the series that is not due to a long-term trend or a short-term cyclical pattern is what we would actually like to explain.

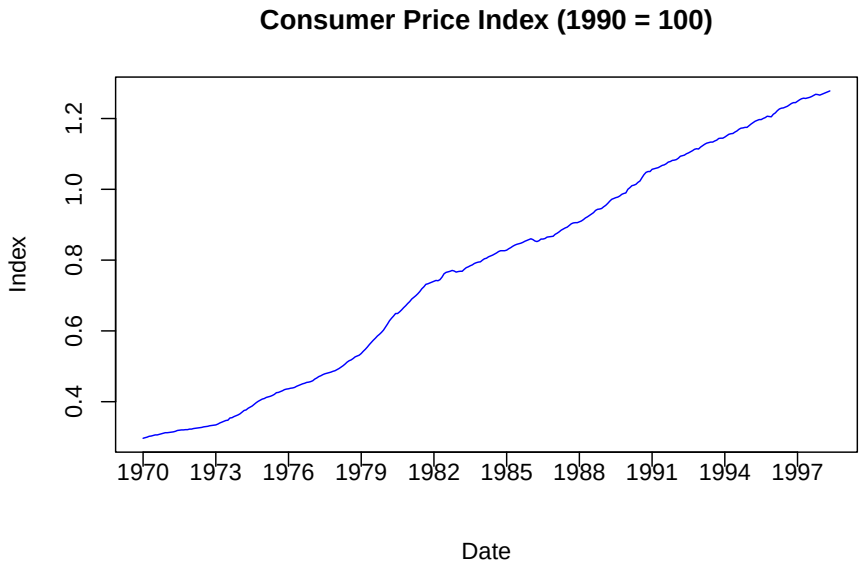
Lets us decompose this series in several steps to not only give us more exposure to dummy variables, but to also learn a bit more about time series data.

Make a *Nominal* Series a *Real* Series

The auto sales series above is known as a *nominal* series because the dollar values for each time period are expressed in the prices of that time period. For example, the data indicates that the US had $4.79 billion in auto sales in January 1970 and $47.4 billion in January 1998. We cannot say that auto sales increased by ten times during this time period, because the US also experienced *inflation* during this time period. In particular, $4.79 billion is denoted in 1970 dollars while $47.4 billion is denoted in 1998 dollars. In order to remove any inflationary distortions from the data, we need to divide these numbers by some

measure of how average prices have evolved. There are many ways of doing this, but a direct method is to use the *consumer price index* or CPI. The CPI tells us how average prices have evolved relative to a benchmark year that is set to 100 (or 1). If the CPI differs in a particular year, then we know how prices have changes relative to the benchmark year.

```
plot(AUTO$INDEX,AUTO$CPI,
     type = "l", main = "Consumer Price Index (1990 = 100)",
     col = "blue", xlab = "Date", ylab = "Index", xaxt = "n")
xtick <- seq(1,length(AUTO$INDEX), by = 36)
axis(side=1, at=xtick, labels = FALSE)
text(x=xtick, par("usr")[3],
     labels = c("1970","1973","1976","1979","1982","1985",
                "1988","1991","1994","1997"),
     pos = 1, xpd = TRUE)
```

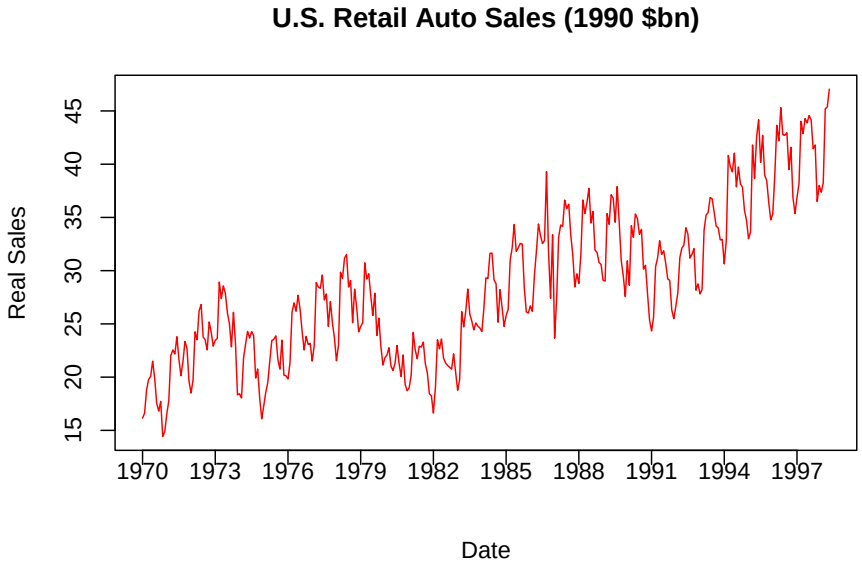


The figure above illustrates the CPI where 1990 is denoted as the benchmark year (because it is set to 1). All other time periods now have prices calculated relative to the benchmark. For example, the CPI in January 1970 is 0.30 which means that average prices were 70 percent lower than what they were in 1990.

We use the CPI to transform a nominal series into a real series. For example:

$$\text{Real Auto Sales} = \frac{\text{Nominal Auto Sales}}{\text{CPI}}$$

```
AUTO$RAUTO = AUTO$AUTOSALE / AUTO$CPI

plot(AUTO$INDEX,AUTO$RAUTO,
     type = "l", main = "U.S. Retail Auto Sales (1990 $bn)",
     col = "red", xlab = "Date", ylab = "Real Sales", xaxt = "n")
xtick <- seq(1,length(AUTO$INDEX), by = 36)
axis(side=1, at=xtick, labels = FALSE)
text(x=xtick, par("usr")[3],
     labels = c("1970","1973","1976","1979","1982","1985",
                "1988","1991","1994","1997"),
     pos = 1, xpd = TRUE)
```



This figure now shows the *Real* US Auto Sales denoted in 1990 prices. For example, January 1970 experienced $16.15 billion in auto sales while January 1998 experienced $47.05. Now that these two numbers are both stated using the same price level, we can say that car sales increased by three times (not ten) over the time period.

Remove a Trend

Our real sales data still shows signs of both a trend and a seasonal cycle that need to be removed. Let us start by removing the trend.

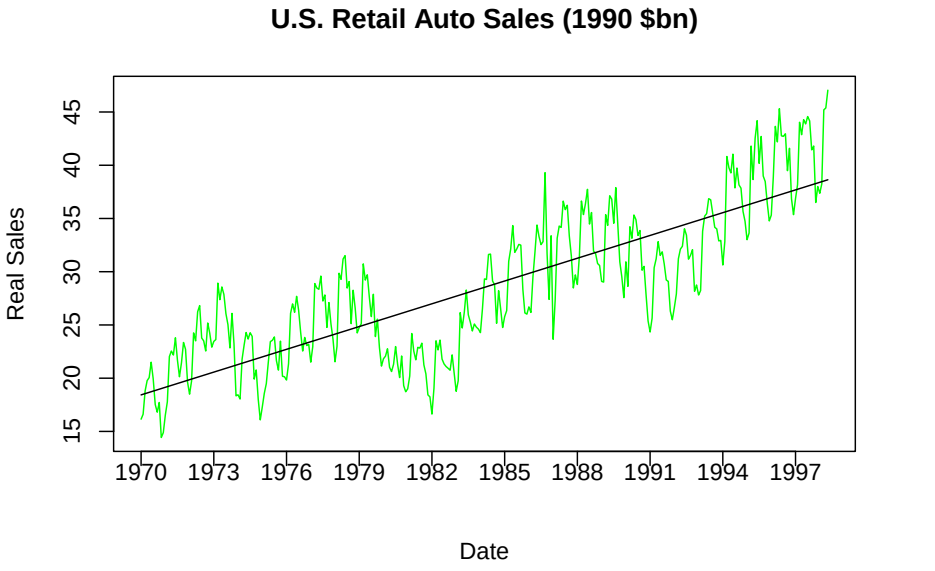
Given that a trend is defined as the average change in a time series, we are technically attempting to identify (and remove) the average change in the series

given a one-unit increase in time. Since this data is monthly, we are attempting to identify the average monthly change in the series. We can identify the trend with a regression equation.

```
AUTO$TREND <- seq(1,length(AUTO$INDEX), by = 1)

DTRND <- lm(AUTO$RAUTO ~ AUTO$TREND)
coef(DTRND)
```

```
## (Intercept)  AUTO$TREND
## 18.37195789  0.05942082
```

```
plot(AUTO$INDEX,AUTO$RAUTO,
     type = "l", main = "U.S. Retail Auto Sales (1990 $bn)",
     col = "green", xlab = "Date", ylab = "Real Sales", xaxt = "n")
xtick <- seq(1,length(AUTO$INDEX), by = 36)
axis(side=1, at=xtick, labels = FALSE)
text(x=xtick, par("usr")[3],
     labels = c("1970","1973","1976","1979","1982","1985",
                "1988","1991","1994","1997"),
     pos = 1, xpd = TRUE)
lines(AUTO$INDEX,fitted(DTRND), col = "black")
```



The code above does three things. First, it creates a variable called *TREND* which is simply an increasing list of numbers from 1 (the first observation) to 341 (the last observation). Each increase is an additional month. Second, it runs a regression where real auto sales is the dependent variable while trend is

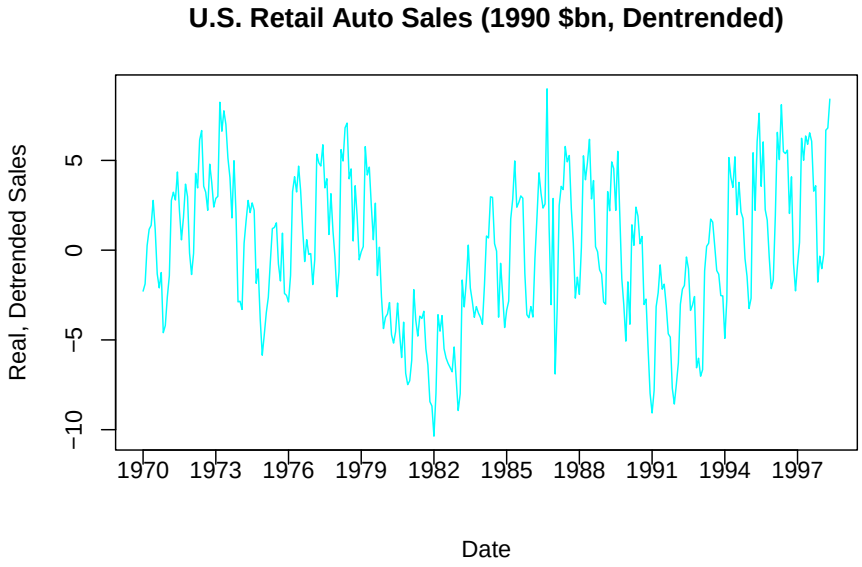
the only independent variable.

$$Real\ Sales_t = \beta_0 + \beta_1\ Trend_t + \varepsilon_t$$

The slope coefficient with respect to the trend is 0.059 which means that average auto sales increase by roughly 0.06 billion 1990 dollars each month on average. Finally, it plots the real series as well as our calculated trend together. Notice how the *predicted* sales coming from the trend line is straight - indicating how this is only the expected sales for a particular month given information only on the evolution of time.

Comparing these two lines in the figure should give you an idea how the trend gets removed from a time series. If we want to remove the predictable change in a series over time, then we can subtract these numbers from the original series once we estimate the trend. Note however that this is already done for you, because the residual of the above regression is actually the information in auto sales that cannot be explained by the predictable evolution of time.

```
AUTO$RAUTO_DT = residuals(DTRND)
plot(AUTO$INDEX,AUTO$RAUTO_DT,
     type = "l",
     main = "U.S. Retail Auto Sales (1990 $bn, Dentrended)",
     col = "cyan", xlab = "Date",
     ylab = "Real, Detrended Sales", xaxt = "n")
xtick <- seq(1,length(AUTO$INDEX), by = 36)
axis(side=1, at=xtick, labels = FALSE)
text(x=xtick, par("usr")[3],
     labels = c("1970","1973","1976","1979","1982","1985",
                "1988","1991","1994","1997"),
     pos = 1, xpd = TRUE)
```



The above figure illustrates the detrended data, where negative numbers indicate that observations are *below trend* while positive numbers indicate that observations are *above trend*.

Remove Seasonality

The figure above still includes a seasonal component which needs to be removed. We will do this using dummy variables.

Identifying seasonality generally refers to the short-run average pattern observed in the series. Since this is monthly data, we would like to observe the average sales in each month. If this were quarterly series, we would like to observe the average sales in each *season* (summer, fall, winter, spring). We can identify these average amounts by using dummy variables to identify if each observation falls into a particular month.

The first step is to establish a benchmark month. This is essentially an arbitrary decision, so lets just go with December (i.e., the twelfth month of the year).

```
head(AUTO)
```

```
## # A tibble: 6 x 9
##   INDEX  YEAR MONTH  DATE AUTOSALE   CPI RAUTO TREND RAUTO_DT
##   <dbl> <dbl> <dbl> <dbl>    <dbl> <dbl> <dbl> <dbl>    <dbl>
## 1     1  1970     1 1970      4.79 0.297  16.2     1   -2.28
## 2     2  1970     2 1970.     4.96 0.298  16.6     2   -1.88
## 3     3  1970     3 1970.     5.64 0.300  18.8     3    0.256
```

```
## 4     4  1970     4 1970.     5.98 0.302  19.8     4    1.16
## 5     5  1970     5 1970.     6.08 0.303  20.1     5    1.38
## 6     6  1970     6 1970.     6.55 0.305  21.5     6    2.77
```

Note that our dataset already has a variable called month which identifies 1 as January, 2 as February, etc. This will make the creation of dummy variables very easy.

Since we want to break this data into 12 categories, then we will need to construct 11 dummy variables. One dummy variable will deliver a 1 every time the observation is in January (0 elsewhere), one dummy variable will deliver a 1 every time the observation is in February (0 elsewhere), and so on. We can do this by hand (which is tedious), or we can use a new package called *fastDummies*.

If you are using *fastDummies* for the first time, you will want to install it:

```
install.packages("fastDummies")
```

This package is designed to accept a variable and construct dummy variables for however many categories it can identify. For example:

```
library(fastDummies)
AUTO <- dummy_cols(AUTO, select_columns = 'MONTH')
names(AUTO)
```

```
##  [1] "INDEX"    "YEAR"     "MONTH"    "DATE"     "AUTOSALE" "CPI"
##  [7] "RAUTO"    "TREND"    "RAUTO_DT" "MONTH_1"  "MONTH_2"  "MONTH_3"
## [13] "MONTH_4"  "MONTH_5"  "MONTH_6"  "MONTH_7"  "MONTH_8"  "MONTH_9"
## [19] "MONTH_10" "MONTH_11" "MONTH_12"
```

Note how the dataset *AUTO* now contains 21 variables when it previously contained 9. This is because the above lines of code created 12 new variables - a dummy variable for each month of the year (1-12). Since we are considering the 12th month as our benchmark, we simply do not include it in our regression.

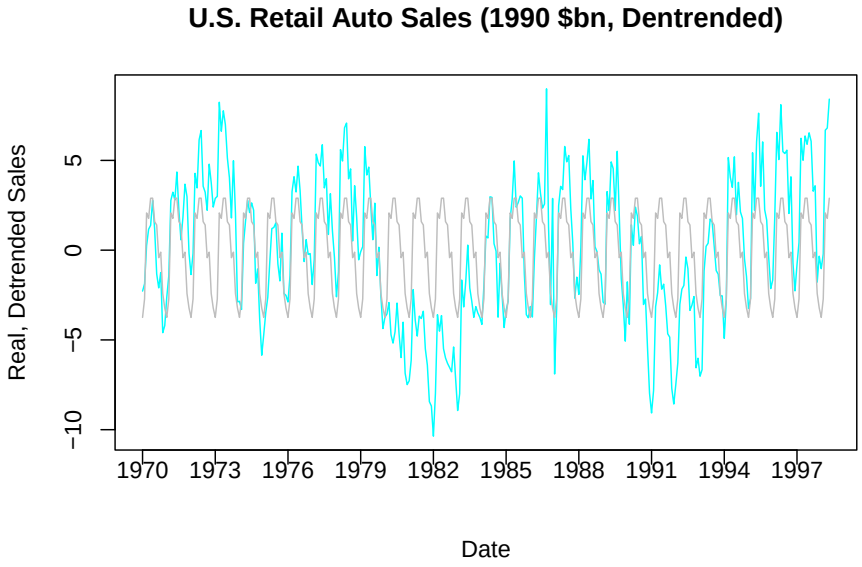
```
DS <- lm(RAUTO_DT ~ MONTH_1 + MONTH_2 + MONTH_3 + MONTH_4 +
           MONTH_5 + MONTH_6 + MONTH_7 + MONTH_8 + MONTH_9 +
           MONTH_10 + MONTH_11, data = AUTO)

summary(DS)
```

```
##
## Call:
## lm(formula = RAUTO_DT ~ MONTH_1 + MONTH_2 + MONTH_3 + MONTH_4 +
##     MONTH_5 + MONTH_6 + MONTH_7 + MONTH_8 + MONTH_9 + MONTH_10 +
##     MONTH_11, data = AUTO)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.3877 -1.8444  0.4071  2.3868  9.4046
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -3.2222     0.6231  -5.172 4.04e-07 ***
## MONTH_1      -0.5227     0.8735  -0.598 0.550014
## MONTH_2       0.5072     0.8735   0.581 0.561902
## MONTH_3       5.2955     0.8735   6.062 3.68e-09 ***
## MONTH_4       4.9933     0.8735   5.716 2.44e-08 ***
## MONTH_5       6.1152     0.8735   7.001 1.44e-11 ***
## MONTH_6       6.1181     0.8811   6.943 2.05e-11 ***
## MONTH_7       4.8101     0.8811   5.459 9.45e-08 ***
## MONTH_8       4.6222     0.8811   5.246 2.79e-07 ***
## MONTH_9       2.8030     0.8811   3.181 0.001607 **
## MONTH_10      3.1100     0.8811   3.530 0.000476 ***
## MONTH_11      0.8048     0.8811   0.913 0.361724
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.297 on 329 degrees of freedom
## Multiple R-squared:  0.3468, Adjusted R-squared:  0.325
## F-statistic: 15.88 on 11 and 329 DF,  p-value: < 2.2e-16
```

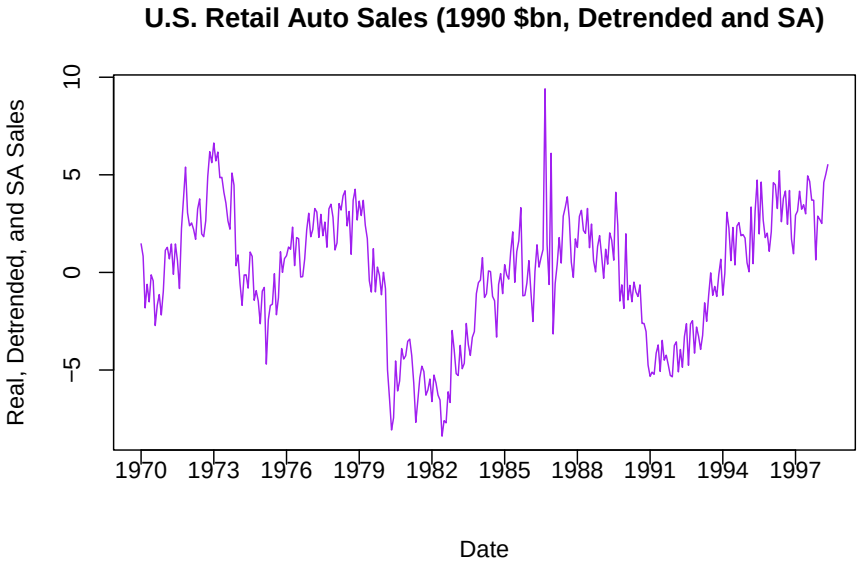
The dummy variable coefficients presented above show how the average auto sales for a particular month differ from the benchmark month of December. For example, June sales (*MONTH_6*) are the largest above December on average with 6.12 billion, while January sales (*MONTH_1*) are actually lower than December average trend sales by 0.52 billion on average. However, note that the difference between June and December is significantly different from zero, while the difference between January and December is not.

```
plot(AUTO$INDEX,AUTO$RAUTO_DT,
     type = "l",
     main = "U.S. Retail Auto Sales (1990 $bn, Dentrended)",
     col = "cyan", xlab = "Date",
     ylab = "Real, Detrended Sales", xaxt = "n")
xtick <- seq(1,length(AUTO$INDEX), by = 36)
axis(side=1, at=xtick, labels = FALSE)
text(x=xtick, par("usr")[3],
     labels = c("1970","1973","1976","1979","1982","1985",
                "1988","1991","1994","1997"),
     pos = 1, xpd = TRUE)
lines(AUTO$INDEX,fitted(DS),col = "gray")
```



The figure above compares the actual detrended series (composed of seasonal and random components) and the seasonal component estimated from our use of dummy variables. As with removing the trend, we can now remove the seasonal component by taking the difference between these two series or simply using the residuals of the regression (since this is the part of the series that cannot be explained by the repeating of months).

```
plot(AUTO$INDEX,residuals(DS),
     type = "l",
     main = "U.S. Retail Auto Sales (1990 $bn, Detrended and SA)", col = "purple",
     xlab = "Date",
     ylab = "Real, Detrended, and SA Sales", xaxt = "n")
xtick <- seq(1,length(AUTO$INDEX), by = 36)
axis(side=1, at=xtick, labels = FALSE)
text(x=xtick, par("usr")[3],
     labels = c("1970","1973","1976","1979","1982","1985",
                "1988","1991","1994","1997"),
     pos = 1, xpd = TRUE)
```



This final figure illustrates the random component of US auto retail sales once we removed price distortions, a long-run trend, and a seasonal cycle. What remains is the component that cannot be explained by these predictable (and uninteresting) things - and this is exactly what analysts what to explain with other more interesting variables (e.g., interest rates, exchange rates, bond prices, etc.). Notice how you can make out the two recessions that occurred during the time frame quite easily.

Chapter 11

Functional Forms

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \varepsilon_i$$

The regression model we have studied thus far has two main features. First, the model is linear in the coefficients. This important property allows us to estimate the model using OLS. Second the model is linear in the variables. This property imposes a linear relationship between the dependent and independent variables. In other words, the relationship is a straight line.

A linear relationship between the independent and a dependent variable results in a slope that is *constant*.

$$\beta_1 = \frac{\Delta Y_i}{\Delta X_{1i}}$$

This means that (holding X_{2i} constant, of course) the expected value of Y will increase by β_1 units in response to any unit-increase in X_{1i}. The same increase occurs on average no matter where in the range of the independent variable you are. Sometimes this assumption is valid if the range of the independent variable is small enough such that a constant slope is appropriate. Sometimes it isn't. If this assumption is not valid, then we are committing a specification error even if we have included all of the necessary independent variables. The specification error involves the assumption of a linear model.

The types of models we will consider here will be non-linear in the variables but will still be linear in the coefficients. This means that we can still estimate the models using OLS, but we will be extending the model to uncover some highly non-linear relationships between the dependent and independent variables. We will do this by transforming the variables prior to estimation, and the names of these models are given by the types of transformations we perform. We then run a simple OLS estimation on the transformed variables, and *back-out* the

non-linear relationships afterwards. This last bit is what will be new to us, but you will see that it only involves a brief refresher of… calculus.

11.1 Derivatives

In calculus, the *slope* of a function is a simplistic term for it's *derivative*. Take for example the very general function $f(X) = aX^b$. This function uses two parameters (a and b) and one variable (X) to return a number or function value $f(X)$. If you think about it, this is exactly what the deterministic component of our regression does. When $b \neq 1$, this function is non-linear. Therefore, to determine the slope - the increase in the function value given a unit-increase in X - we need to take the derivative. The general formula for a derivative is given by

$$\frac{\Delta f(X)}{\Delta X} = abX^{b-1}$$

Note that we have used this derivative formula before, only we used it when $f(X) = Y$ and the formula was linear (i.e. $b = 1$) and $a = \beta$.

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

$$\frac{\Delta Y_i}{\Delta X_i} = \beta_1$$

11.2 Why consider non-linear relationships?

The models we will consider below will generally use this extension (and a few other calculus tools) to transform our variables and get at specific non-linear relationships. The reason we do this is to get at the *true* relationship between the dependent and independent variables. If the relationship is in fact linear, then none of these sophisticated models are necessary. If the relationship is not linear, then linear models are by definition incorrect and will deliver misleading results. The models in this section are therefore used only when the data requires them. Nobody wants to make a model more complicated than it needs to be.

The sole purpose of introducing a sophisticated functional form into an otherwise straight-forward regression model is because the relationship between a dependent an independent variable is not linear *in the data*. Recall that a linear relationship is one where the slope is constant. If the slope is constant (say, β), then a one-unit increase in X will deliver a β unit increase in Y on average *no matter where in the range of X you are*. There are a lot of instances in the real world where this doesn't make sense. Take for example the impact of apartment rental price on its size. One can imagine that if an apartment is small (e.g. 50 sq

ft), then one might be willing to pay a lot more for a slight increase in size. If an apartment is ridiculously huge (e.g. 5000 sq ft), then one might not be willing to pay anything for an increase in size. This means that the relationship between apartment rental price and size is conceptually non-linear - the slope is dependent upon the actual size of the apartment. However, if your data has a small range (e.g., between 300 and 400 sq ft), then you might never need to consider a non-linear relationship because a linear one does a good job of approximating the relationship that you observe. This chapter deals with situations where you observe a non-linear relationship in your actual data, so it needs to be modeled.

Once we have established that a relationship is non-linear, we next need to take a stand on what *type* of non-linear relationship we are attempting to uncover. Answering this question depends upon how you think the slope is going to behave.

- Is the slope not constant in unit changes, but constant in percentage changes?
- Does the slope qualitatively change? In other words, is there a relationship between a dependent and independent variable that starts out as positive (or negative) and eventually turns negative (or positive)?
- Does the slope start out positive or negative and eventually *dies out* (i.e., goes to zero)?

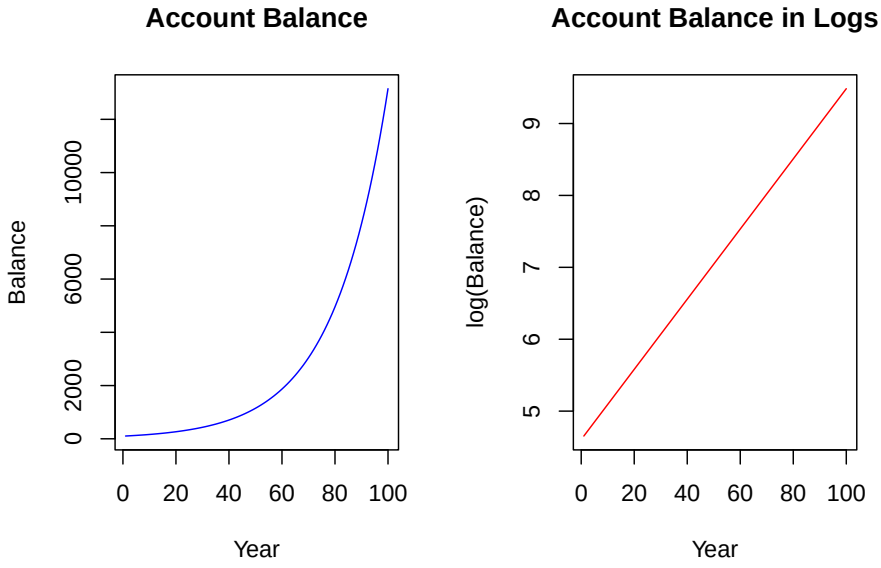
This chapter details three types of non-linear transformations each designed to go after one of these three scenarios. Non-linear transformations are not *one size fits all*, so having a good idea of how to handle each type of relationship is essential.

11.3 The Log transformation

The natural log transformation is used when the relationship between a dependent and independent variable is not constant in units but constant in percentage changes (or growth rates). Imagine putting $100 in a bank at 5 percent interest. If you kept the entire balance in the account, then after one year you will have $105 (a $5 increase), after two years you will have $110.25 (a $5.25 increase), after three years you will have $115.76 (a $5.76 increase), and so on. What is happening is that the account balance is not growing in constant dollar units, but it is growing in constant percentage units. In fact, the balance is said to be growing *exponentially*. Things like a country's output, aggregate prices, population all grow exponentially because they build on each other just like the compound interest story.

If we kept our $100 dollars in the bank for a very long time, the balance would evolve according to the figure below on the left. The figure illustrates a non-linear relationship between account balance and time - and the slope is getting steeper as time goes on. While we know that the account balance is increasing by

larger and larger dollar increments, we also know that it is growing at a constant five percent. We can uncover this constant percentage change by applying the natural log to the balance - as we did to the right figure. You can see that the natural log function *straightens* the exponential relationship - so the transformed relationship is linear and ready for our regression model.



The derivative of the log function

The natural log function has a very specific and meaningful derivative:

$$\frac{dln(Y)}{dY} = \frac{\Delta Y}{Y}$$

This formula is actually a generalization of the percentage change formula. Suppose you wanted to know the difference between Y_2 and Y_1 in percentage terms relative to Y_1. The answer is

$$\frac{Y_2 - Y_1}{Y_1} * 100\%$$

Therefore, the only thing missing from the log transformation is the multiplication of 100%, which we can do after estimation.

For example, suppose that you didn't know the *average* percentage change (or average growth rate) of your account. If Y was your account balance and X was

number of years in the account, then you could estimate what it was. Notice that the *slope* is 0.05. If you multiply that by 100% then you have your 5% interest rate back.

```
R <- lm(log(Y)~X)
```

(Intercept)	X
4.605	0.04879

Log-log and Semi-log models

Recall that a standard slope is the *change* in Y over a *change* in X. Combine this fact with the log of a variable delivers a percentage change in the derivative (provided you multiply by 100%), and you have several options for which variables you want to consider the logs of. The question you ask yourself is if you want to consider the change of a variable in units **or** the percentage change of a variable.

Log-log model

A Log-log model is one where both the dependent and the independent variable are logged.

$$ln(Y_i) = \beta_0 + \beta_1 ln(X_i) + \varepsilon_i$$

The slope coefficient (β_1) details the percentage change in the dependent variable given a one *percent* change in the independent variable. To see this, apply the derivative formula above to the entire formula.

$$\begin{aligned}
\frac{dln(Y_i)}{dY} &= \beta_1 \frac{dln(X_i)}{dX} \\
\frac{dln(Y_i)}{dY} * 100\% &= \beta_1 \frac{dln(X_i)}{dX} * 100\% \\
\%\Delta Y_i &= \beta_1 \%\Delta X_i \\
\beta_1 &= \frac{\%\Delta Y_i}{\%\Delta X_i}
\end{aligned}$$

Semi-log models

Sometimes it makes no sense to take the log of a variable because the percentage change makes no sense. For example, it wouldn't make sense to take the log of the year in the bank account example above because time is not relative. In other words, a percentage change in time doesn't make sense. In addition, variables that reach values of zero or lower *cannot* be logged because the natural

log is only defined on positive values. In either case, it would only make sense to not take the log of some of the variables.

A Log-lin model is a semi-log model where only the dependent variable is logged. This is like the case with the bank account example above.

$$ln(Y_i) = \beta_0 + \beta_1 X_i + \varepsilon_i$$

$$\frac{dln(Y_i)}{dY} = \beta_1 \Delta X$$

$$\frac{dln(Y_i)}{dY} * 100\% = (\beta_1 * 100\%)\Delta X$$

$$\frac{\%\Delta Y}{\Delta X} = \beta_1 * 100\%$$

Note that the 100% we baked into the interpretation is explicitly accounted for in order to turn the derivative of the log function into a percentage change.

A Lin-Log model is a semi-log model where only the independent variable is logged. This might come in handy when you want to determine the average change in the dependent variable in response to a percentage-change in the independent variable.

$$Y_i = \beta_0 + \beta_1 ln(X_i) + \varepsilon_i$$

$$\Delta Y = \beta_1 \frac{dln(X_i)}{X_i}$$

$$\Delta Y = \beta_1 \frac{dln(X_i)}{X_i} * \frac{100}{100}$$

$$\Delta Y = \frac{\beta_1}{100} \%\Delta X$$

$$\frac{\Delta Y}{\%\Delta X} = \frac{\beta_1}{100}$$

Note that the derivation for the lin-log model suggests that you must divide the estimated coefficient by 100 in order to state the expected change in the dependent variable due to a *percentage* change in the independent variable.

It isn't ALL OR NOTHING!!!

To be clear, if you have a multiple regression model with several independent variables, you get to treat each independent variable however you wish. In other words, if you log one independent variable, you do not need to automatically log the others. This is especially the case when some can be logged while others cannot. The bottom line is that if you have one of the relationships detailed above with the dependent variable and a single independent variable, then you use the correct derivative form and provide the correct interpretation.

In particular, suppose you had the following model

$$ln(Y_i) = \beta_0 + \beta_1 X_{1i} + \beta_2 ln(X_{2i}) + \varepsilon_i$$

This model is a combination between a log-lin model (with respect to X_{1i}) and a log-log model (with respect to X_{2i}). The derivatives are therefore

$$\beta_1 * 100\% = \frac{\%\Delta Y}{\Delta X_1}$$

$$\beta_2 = \frac{\%\Delta Y}{\%\Delta X_2}$$

Application

If we ran a regression with hourly wage as the dependent variable and tenure (i.e., years on the job) as the independent variable, then we are estimating the average change in dollars for an additional year of tenure. However, it might be more worthwhile to consider an annual average percentage change in wage as opposed to a dollar change. That is what happens for most people, anyway.

```
data(wage1,package="wooldridge")

REG <- lm(log(wage) ~ tenure, data = wage1)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	1.501	0.02687	55.87	7.261e-223
tenure	0.02395	0.003039	7.881	1.89e-14

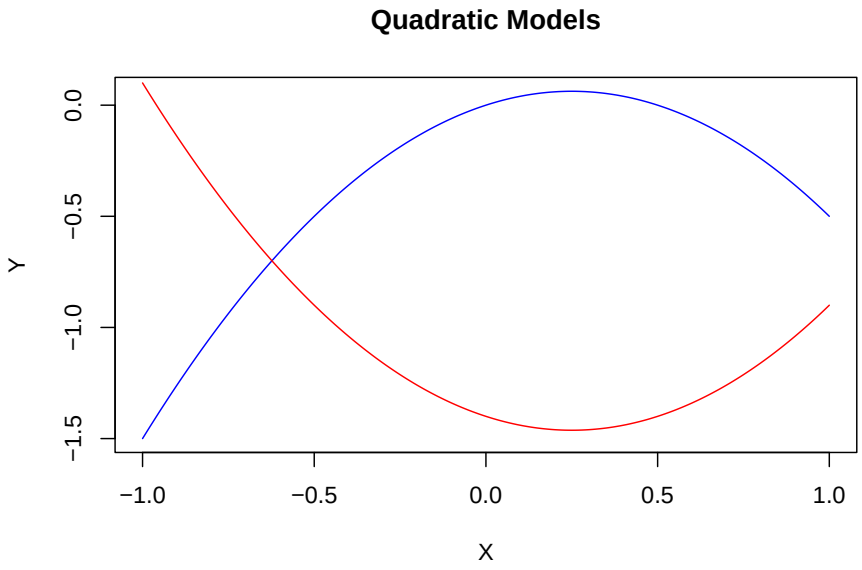
Table 11.3: Fitting linear model: log(wage) ~ tenure

Observations	Residual Std. Error	R^2	Adjusted R^2
526	0.5031	0.106	0.1043

The slope estimate gets multiplied by 100% so we can state that wages increase by 2% on average for every additional year of tenure.

11.4 The Quadratic transformation

A quadratic transformation is used when the relationship between a dependent and independent variable *changes direction*. The slope can start off positive and become negative (the blue line in the figure), or begin negative and become positive (the red line).



Quadratic models are designed to handle functions featuring slopes that change qualitatively.

$$Y_i = \beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \varepsilon_i$$

Notice that this regression has one variable showing up twice: once linearly and once squared. The regression is still linear in the coefficients, so we can estimate this regression as if the regression contained any two independent variables of interest. In particular, if you defined a new variable $X_{2i} = X_i^2$, then the model would look like a standard multiple regression.

Once the estimated coefficients are obtained, they are combined to calculate one non-linear slope equation.

$$\frac{\Delta Y}{\Delta X} = \beta_1 + 2\beta_2 X$$

A few things to note about this slope equation.

1. It is a function of X - meaning that you need to choose a value of the independent variable to calculate a numerical slope. This means you are calculating the expected change in X from a unit increase in X *from a particular value*.

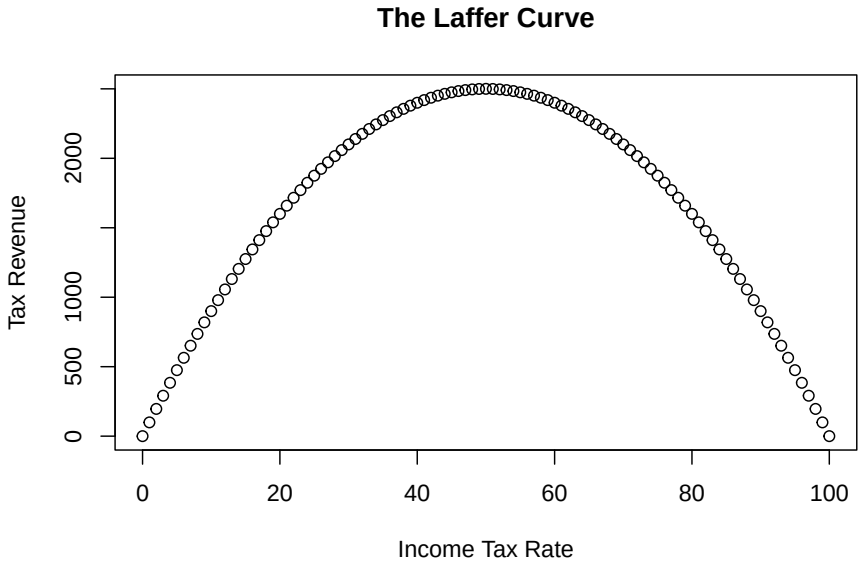
2. The slope is increasing or decreasing depending on the values of the coefficients and the independent value of X. The coefficients β_1 and β_2 are usually of opposite sign. Therefore, the slope is negative if $\beta_1 < 0$ and X is small so $\beta_1 + 2\beta_2 X < 0$, or positive if $\beta_1 > 0$ and X is small so $\beta_1 + 2\beta_2 X > 0$.

Note: just as in the log transformation, this quadratic transformation does not need to be done to every independent variable in the regression. Only those that are suspected to have a relationship with the dependent variable that changes direction.[1]

Application

Consider some simulated data illustrating the relationship between tax rates and the amount of tax revenue collected by the government. One can intuitively imagine that the government will collect zero revenue if they tax income at zero percent. However, they will also collect zero revenue if they tax at 100 percent because nobody will work if their entire income is taxed away. Therefore, there should be a relationship between tax rate and tax revenue that looks something like the figure below.

[1] For example, you will never be able to look at a quadratic transformation for a dummy variable because when the observations are only 0 and 1, then X and X^2 are technically the same variable!



This figure illustrates the infamous Laffer Curve of supply-side economics.[2]

Suppose you had the data illustrated in the figure. If you assumed a linear relationship between tax revenue and rate, then you will get a very misleading result.

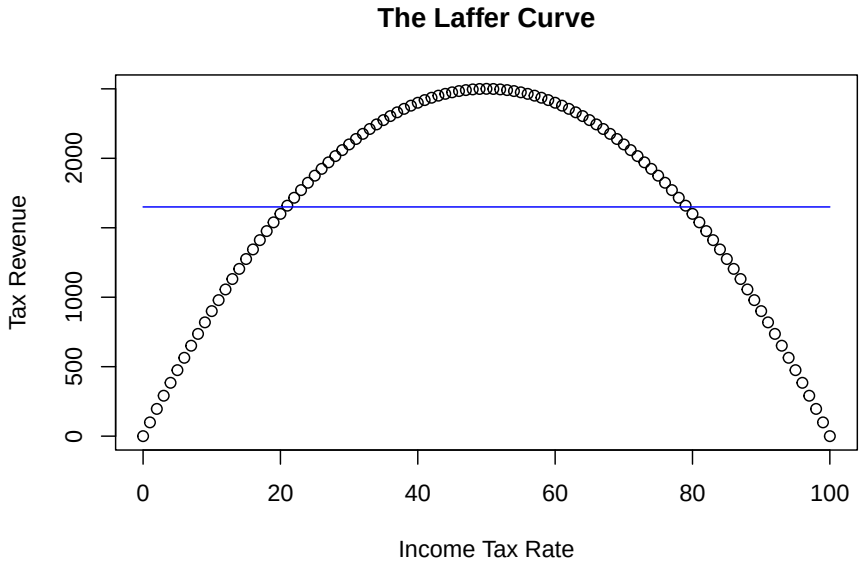
```
REG <- lm(Revenue ~ Rate)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	1650	151.7	10.88	1.336e-18
Rate	5.432e-15	2.62	2.073e-15	1

Table 11.5: Fitting linear model: Revenue ~ Rate

Observations	Residual Std. Error	R^2	Adjusted R^2
101	767.8	1.603e-31	-0.0101

[2]We are using this as an example of a quadratic relationship only - I will spare you my tirade on the detrimental use of this (admittedly intuitive) idea by supply-siders.



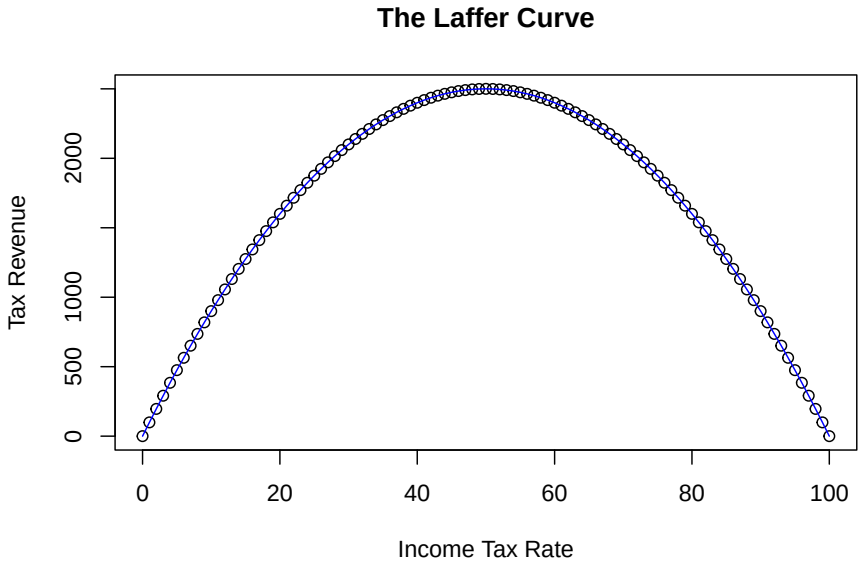
The figure shows that the best fitting straight line is *horizontal* - meaning that the slope is zero. This linear model would suggest that there is no relationship between tax revenue and tax rate. It is partly true - there is just no *linear relationship*.

```
REG <- lm(Revenue ~ Rate + I(Rate^2))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-2.896e-12	5.803e-13	-4.991	2.613e-06
Rate	100	2.682e-14	3.729e+15	0
I(Rate^2)	-1	2.595e-16	-3.853e+15	0

Table 11.7: Fitting linear model: Revenue ~ Rate + I(Rate^2)

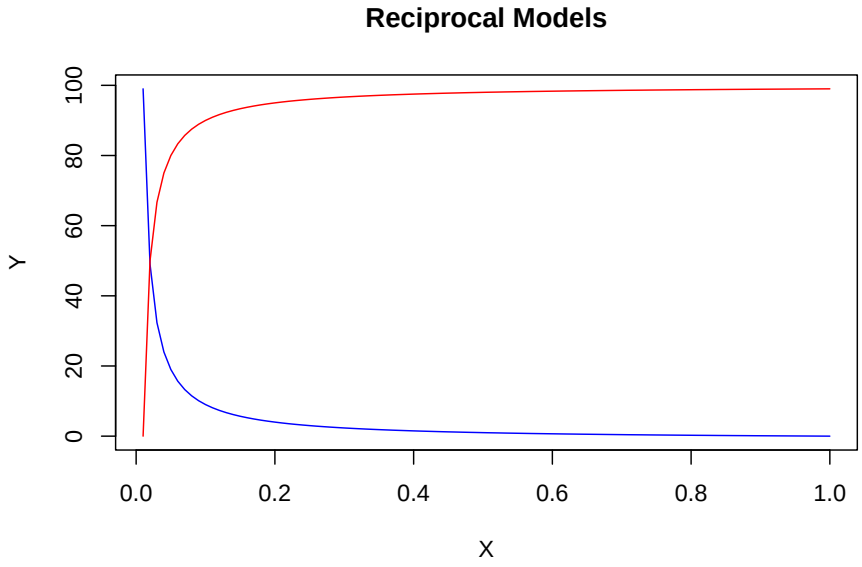
Observations	Residual Std. Error	R^2	Adjusted R^2
101	1.983e-12	1	1



11.5 The Reciprocal transformation

Suppose a relationship doesn't change directions as much as it *dies out*.

$$Y_i = \beta_0 + \beta_1 \frac{1}{X_i} + \varepsilon_i$$



As in the quadratic transformation, this model is easily estimated by redefining variables (i.e., $X_{2i} = 1/X_i$). The slope of this function can be obtained using our standard derivative function and noting that $1/X_i = X_i^{-1}$

$$Y_i = \beta_0 + \beta_1 \frac{1}{X_i} + \varepsilon_i$$

$$Y_i = \beta_0 + \beta_1 X_i^{-1} + \varepsilon_i$$

$$\frac{\Delta Y}{\Delta X} = -\beta_1 X_i^{-2} = \frac{-\beta_1}{X_i^2}$$

Notice again that a value of the independent variable is needed to calculate the slope of the function at a specific point. However, as X gets larger, the slope approaches zero. A slope of zero is a horizontal line - and that is when the relationship between Y and X dies out. If $\beta_1 > 0$ then the slope begins negative and approaches zero from above (the blue line). If $\beta_1 < 0$ then the slope begins positive and approaches zero from below (the red line).

Application

The richer a nation is, the better a nation's health services are. However, a nation can eventually get to a point that health outcomes cannot improve no matter how rich it gets. This application considers child mortality rates of developing and developed countries and uses the wealth of a country measured

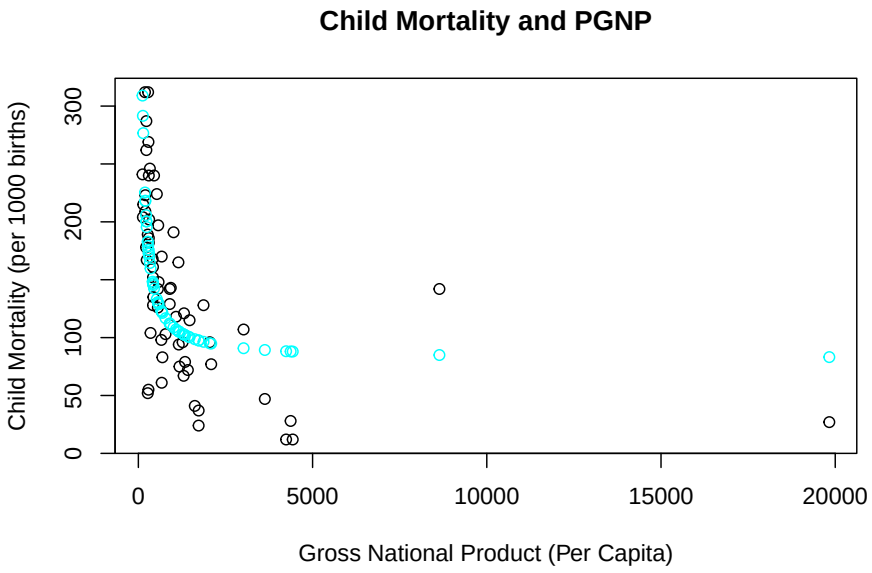
by per-capita gross national product (PGNP) to help explain why the child mortality rate is different across countries.

```
library(readxl)
CM <- read_excel("data/CM.xlsx")

REG <- lm(CM$CM ~ I(1/CM$PGNP))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	81.79	10.83	7.551	2.38e-10
I(1/CM$PGNP)	27273	3760	7.254	7.821e-10

Table 11.9: Fitting linear model: CMCM $I(1/CM\text{PGNP})$

Observations	Residual Std. Error	R^2	Adjusted R^2
64	56.33	0.4591	0.4503



The estimated coefficient is large and positive. However, this isn't the entire slope, because you need to consider the derivative formula above.

$$\frac{\Delta Y}{\Delta X} = \frac{-27,273}{X_i^2}$$

If you wanted to consider the impact on child mortality of making a relatively poor nation richer, you plug a low value for PGNP like 100. If you want to consider a richer nation, consider a larger value like 1000.

$$\frac{\Delta Y}{\Delta X} = \frac{-27,273}{100^2} = -2.73$$
$$\frac{\Delta Y}{\Delta X} = \frac{-27,273}{1000^2} = -0.0273$$

These calculations show that increasing the wealth of richer countries has a smaller impact on child mortality rates. This should make sense, and it takes a reciprocal model to capture it.

11.6 Conclusion

This section introduced you to three different ways of adding potential non-linear relationships into our model while still preserving linearity in the coefficients. This allows us to retain our OLS estimation procedure, and only requires some calculus steps after estimation to get at our answers.

One take away is that one can easily map out a functional form in theory, but it might not be entirely captured by the data sample. In other words, while we can always tell a story that a relationship might become non-linear *eventually*, if that extreme range is not in the data then a non-linear relationship isn't necessary.

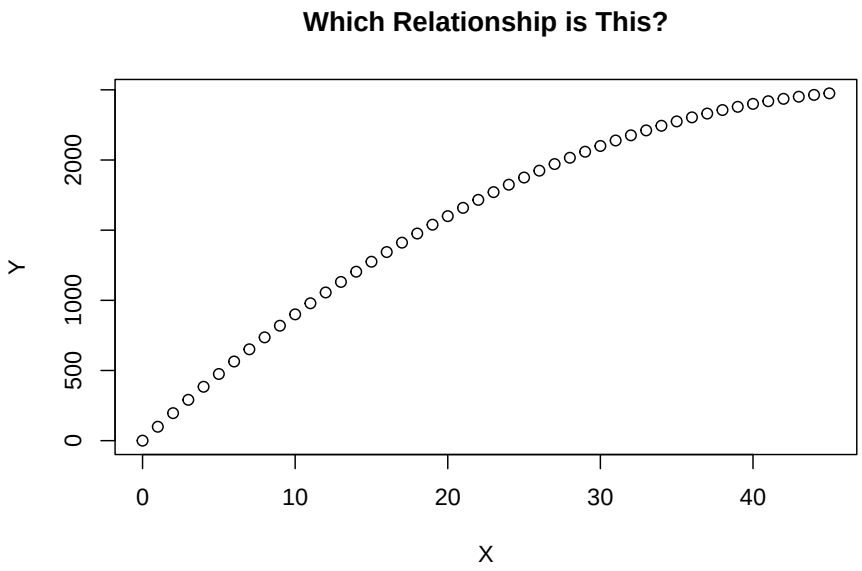
On the other hand, if there is a non-linear relationship in the data, then it might be the case that more than one functional form might fit. While it is true that the three transformations handle different *right-side* behaviors, notice that the *left-side* of the relationships look quite similar.

Take the figure below for example. We do not have enough data to see if the relationship stays increasing (requiring a log transformation), changes direction (requiring a quadratic transformation), or dies out (requiring a reciprocal transformation). If this is the case then trial and error combined with a lot of care is required.

- Which model has the highest R^2? (Provided that the dependent variable is not transformed.)
- Which model makes the most sense theoretically?
- What is the differences in the out-of-sample forecasts between models? What is the *cost* of being wrong?

```
X <- seq(0,45,1)
Y <- 100 * X - X^2

plot(X,Y,
```

```
     main = "Which Relationship is This?",
     xlab = "X",
     ylab = "Y")
```



The bottom line is that choosing the wrong non-linear transformation will still lead to some amount of specification bias, but it might not be as much as a linear specification.

Chapter 12

Joint Hypothesis Tests

This final chapter deals with a powerful tool of statistical inference: *joint hypothesis tests*. The concept of a joint hypothesis test is detailed in contrast to *simple hypothesis tests* - which are the types of hypothesis tests we have learned thus far. Once we establish what a joint hypothesis test can do, we then go about conducting the tests. This does introduce a new probability distribution that we need to use when calculating p-values. However, once a p-value is calculated we can use it to make a conclusion just like before.

12.1 Simple versus Joint Hypothesis Tests

We have already considered all there is to know about *simple* hypothesis tests.

$$H_0 : \beta = 0 \quad \text{versus} \quad H_1 : \beta \neq 0$$

With the established (one-sided or two-sided) hypotheses, we were able to calculate a test statistic given a nonarbitrary value of β, calculate a p-value, and conclude. There is nothing more to it than that.

A simple hypothesis test follows the same constraints as how we interpret single coefficients: *all else equal*. In particular, when we conduct a simple hypothesis test, we must calculate a test statistic under the null while assuming that all other coefficients are unchanged. This might be fine under some circumstances, but what if we want to test the population values of multiple regression coefficients at the same time? Doing this requires going from simple hypothesis tests to **joint** hypothesis tests.

Joint hypothesis tests consider a stated null involving multiple PRF coefficients simultaneously. Consider the following general PRF:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_{3i} + \varepsilon_i$$

A simple hypothesis test such as

$$H_0 : \beta_1 = 0 \quad \text{versus} \quad H_1 : \beta_1 \neq 0$$

is conducted under the assumption that β_2 and β_3 are left to be whatever the data says they should be. In other words, a simple hypothesis test can only address a value for one coefficient at a time while being silent on all others.

A joint hypothesis states a null hypothesis that considers multiple PRF coefficients simultaneously. The statement in the null hypothesis can become quite sophisticated and test some very interesting statements.

For example, we can test if *all* population coefficients are equal to zero - which explicitly states that none of the independent variables are important.

$$H_0 : \beta_1 = \beta_2 = \beta_3 = 0 \quad \text{versus} \quad H_1 : \beta_1 \neq 0,\ \beta_2 \neq 0,\ \text{or}\ \beta_3 \neq 0$$

We don't have to be so extreme and test that just two of the three coefficients are simultaneously zero.

$$H_0 : \beta_1 = \beta_3 = 0 \quad \text{versus} \quad H_1 : \beta_1 \neq 0 \text{ or } \beta_3 \neq 0$$

If we have a specific theory in mind, we could also test if PRF coefficients are simultaneously equal to specific (nonzero) numbers.

$$H_0 : \beta_1 = 1 \text{ or } \beta_3 = 4 \quad \text{versus} \quad H_1 : \beta_1 \neq 1 \text{ or } \beta_3 \neq 4$$

Finally, we can test if PRF coefficients behave according to some relative measures. Instead of stating in the null that coefficients are equal to some specific number, we can state that they are equal (or opposite) to each other or they behave according to some mathematical condition.

$$H_0 : \beta_1 = -\beta_3 \quad \text{versus} \quad H_1 : \beta_1 \neq -\beta_3$$

$$H_0 : \beta_1 + \beta_3 = 1 \quad \text{versus} \quad H_1 : \beta_1 + \beta_3 \neq 1$$

$$H_0 : \beta_1 + 5\beta_3 = 3 \quad \text{versus} \quad H_1 : \beta_1 + 5\beta_3 \neq 3$$

As long as you can state a hypothesis involving multiple PRF coefficients in a linear expression, then we can test the hypothesis using a joint test. There are

an infinite number of possibilities, so it is best to give you a couple of concrete examples to establish just how powerful these tests can be.

Application

One chapter of my PhD dissertation concluded with a single joint hypothesis test. The topic I was researching was the *Bank-Lending Channel of Monetary Policy Transmission*, which is a bunch of jargon dealing with how banks respond to changes in monetary policy established by the Federal Reserve. A paper from 1992 written by Ben Bernanke and Alan Blinder established that aggregate bank lending volume responded to changes in monetary policy (identified as movements in the Federal Funds Rate).[1] A simplified version of their model (below) considers the movement in bank lending as the dependent variable and the movement in the Fed Funds Rate (FFR) as the independent variable.

$$L_i = \beta_0 + \beta_1 FFR_i + \varepsilon_i$$

While this is a simplification of the model actually estimated, you can see that β_1 will concisely capture the change in bank lending given an increase in the Fed Funds Rate.

$$\beta_1 = \frac{\Delta L_i}{\Delta FFR_i}$$

Since an increase in the Federal Funds Rate indicates a tightening of monetary policy, the authors proposed a simple hypothesis test to show that an increase in the FFR delivers a decrease in bank lending.

$$H_0 : \beta_1 \geq 0 \quad \text{versus} \quad H_1 : \beta_1 < 0$$

Their 1992 paper rejects the null hypothesis above, which gave them empirical evidence that bank lending responds to monetary policy changes. The bank lending channel was established!

My dissertation tested an implicit assumption of their model: *symmetry*.

$$\beta_1 = \frac{\Delta L_i}{\Delta FFR_i}$$

The interpretation of the slope of this regression works for both increases and decreases in the Fed Funds Rate. Assuming that $\beta_1 < 0$, a one-unit increase in the FFR will deliver an expected decline of β_1 units of lending on average. However, it also states that a one-unit *decrease* in the FFR will deliver an

[1] Bernanke, B., & Blinder, A. (1992). The Federal Funds Rate and the Channels of Monetary Transmission. *The American Economic Review*, 82(4), 901-921.

expected *increase* of β_1 units of lending on average. This symmetry is baked into the model. The only way we can explicitly test this assumption is to extend the model and perform a joint hypothesis test.

Suppose we separated the FFR variable into increases in the interest rate and decreases in the interest rate.

$$FFR_i^+ = FFR_i > 0 \quad \text{(zero otherwise)}$$
$$FFR_i^- = FFR_i < 0 \quad \text{(zero otherwise)}$$

If we were to put both of these variables into a similar regression, then we could separate the change in lending from increases and decreases in the interest rate.

$$L_i = \beta_0 + \beta_1 FFR_i^+ + \beta_2 FFR_i^- + \varepsilon_i$$

$$\beta_1 = \frac{\Delta L_i}{\Delta FFR_i^+}, \quad \beta_2 = \frac{\Delta L_i}{\Delta FFR_i^-}$$

Notice that both β_1 and β_2 are still hypothesized to be negative numbers. However, the first model imposed the assumption that they were the *same* negative number while this model allows them to be different. We can therefore test the hypothesis that they are the same number by performing the following joint hypothesis:

$$H_0 : \beta_1 = \beta_2 \quad \text{versus} \quad H_1 : \beta_1 \neq \beta_2$$

In case you were curious, the null hypothesis get rejected and this provides evidence that the bank lending channel is indeed *asymmetric*. This implies that banks respond more to monetary tightenings than monetary expansions, which should make sense given all of the low amounts of bank lending in the post-global recession of 2008 despite interest rates being at all time lows.

12.2 Conducting a Joint Hypothesis Test

A joint hypothesis test involves four steps:

1. Estimate an *unrestricted* model
2. Impose the null hypothesis and estimate a *restricted* model
3. Construct a *test statistic under the null*
4. Determine a p-value and conclude

1. Estimate an Unrestricted Model

An analysis begins with a regression model that can adequately capture what you are setting out to uncover. In general terms, this is a model that doesn't impose any serious assumptions on the way the world works so you can adequately test these assumptions. Suppose we have a hypothesis that two independent variables impact a dependent variable by the same quantitative degree. In that case, we need a model that does not impose this hypothesis.

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \varepsilon_i$$

The model above allows for the two independent variables to impact the dependent variable in whatever way the data sees fit. Since there is no imposition of the hypothesis on the model, or no restriction that the hypothesis be obeyed, then this model is called the *unrestricted* model.

2. Estimate a Restricted Model

A restricted model involves both the unrestricted model and the null hypothesis. If we wanted to test if the two slope hypotheses were the same, then our joint hypothesis is just like the one in the previous example:

$$H_0 : \beta_1 = \beta_2 \quad \text{versus} \quad H_1 : \beta_1 \neq \beta_2$$

With the null hypothesis established, we now need to construct a *restricted* model which results from imposing the null hypothesis on the unrestricted model. In particular, starting with the unrestricted model and substituting the null, we get the following:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \varepsilon_i$$

$$Y_i = \beta_0 + \beta_2 X_{1i} + \beta_2 X_{2i} + \varepsilon_i$$

$$Y_i = \beta_0 + \beta_2 (X_{1i} + X_{2i}) + \varepsilon_i$$

$$Y_i = \beta_0 + \beta_2 \tilde{X}_i + \varepsilon_i \quad \text{where} \quad \tilde{X}_i = X_{1i} + X_{2i}$$

Imposing the null hypothesis restricts the two slope coefficients to be identical. If we construct the new variable $\tilde{X}_i$ according to how the model dictates, then we can use the new variable to estimate the *restricted* model.

3. Construct a test statistic under the null

Now that we have our unrestricted and restricted models estimated, the only two things we need from them are the R^2 values from each. We will denote the R^2 from the unrestricted model as the *unrestricted* R^2 or R^2_u, and the R^2 from the restricted model as the *restricted* R^2 or R^2_r.

These two pieces of information are used with *two* degrees of freedom measures to construct a test statistic under the null - which is conceptually similar to how we perform simple hypothesis tests. However, while simple hypothesis tests are performed assuming a Student's t distribution, joint hypothesis tests are performed assuming an entirely new distribution: An F distribution.

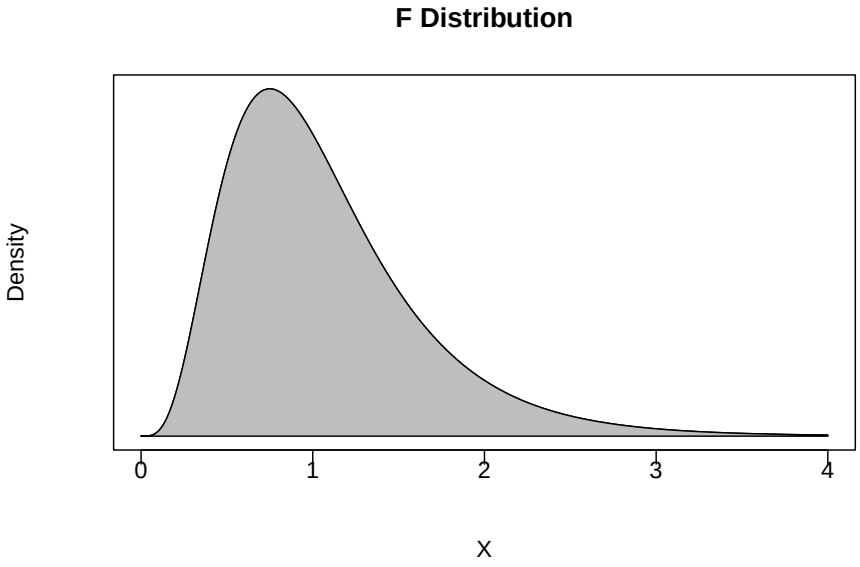
Roughly speaking, an F distribution arises from taking the square of a t distribution. Since simple hypothesis tests deal with t distributions, and the joint hypothesis deals with R^2 values, you get the general idea. An F-statistic under the null is given by

$$F = \frac{(R^2_u - R^2_r)/m}{(1 - R^2_u)/(n - k - 1)} \sim F_{m,\, n-k-1}$$

where

- R^2_u is the unrestricted R^2 - the R^2 from the unrestricted model.
- R^2_r is the restricted R^2 - the R^2 from the restricted model.
- m is the numerator degrees of freedom - the number of restrictions imposed on the restricted model. In other words, count up the number of equal signs in the null hypothesis.
- $n - k - 1$ is the denominator degrees of freedom - this is the degrees of freedom for a simple hypothesis test performed on the *unrestricted* model.

In simple hypothesis tests, we constructed a t-statistic that is presumably drawn from a t-distribution. We are essentially doing the same thing here by constructing a F-statistic that is presumably drawn from a F-distribution.



The F-distribution has a few conceptual properties we should discuss.

An F statistic is restricted to be non-negative.

This should make sense because the expressions in both the numerator and denominator of our F-statistic calculation are both going to be non-negative. The numerator is always going to be non-negative because $R_u^2 \geq R_r^2$. In other words, the unrestricted model will always explain more or at least as much of the variation in the dependent variable as the restricted model does. When the two models explain the same amount of variation, then the R^2 values are the same and the numerator is zero. When the two models explain different amounts of variation, then this means that the restriction prevents the model from explaining as much of the variation in the dependent variable it otherwise would when not being restricted.

The Rejection Region is Always in the Right Tail

If we have $R_u^2 = R_r^2$, then this implies that the restricted model and the unrestricted model are explaining the same amount of variation in the dependent variable. Think hard about what this is saying. If both models have the same R^2, then they are essentially *the same model*. One model is unrestricted meaning it can choose any values for coefficients it sees fit. The other model is restricted meaning we are forcing it to follow whatever is specified in the null. If these two models are the same, then the *restriction doesn't matter*. In other words, the model is choosing the values under the null whether or not we are imposing the null. If that is the case, then the f-statistic will be equal to or close to zero.

If we have $R_u^2 > R_r^2$, then this implies that the restriction imposed by the null hypothesis is hampering the model from explaining as much of the volatility in the dependent variable than it otherwise would have. The more $R_u^2 > R_r^2$, the more $F > 0$. Once this F-statistic under the null becomes large enough, we reject the null. This means that the difference between the unrestricted and restricted models is so large that we have evidence to state that the null hypothesis is simply not going on in the data. This implies that the rejection region is *always* in the right tail, and the p-value is always calculated from the right as well.

4. Determine a P-value and Conclude

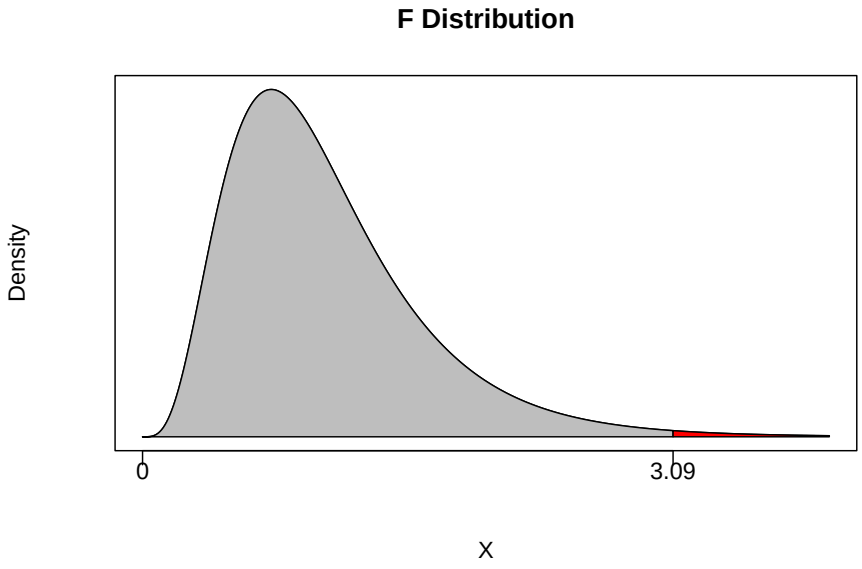
Again, we establish a confidence level α as we would with any hypothesis test. This delivers an acceptable probability of a type I error and breaks the distribution into a rejection region and a non-rejection region.

For example, suppose you set $\alpha = 0.05$ and have $m = 2$ and $n - k - 1 = 100$. This means that the non-rejection region will take up 95% of the area of the F-distribution with 2 and 100 degrees of freedom.

```
(Fcrit <- qf(0.95,2,100))
```

```
## [1] 3.087296
```

If an F-statistic is greater than 3.09 then we can reject the null of the joint hypothesis with at least 95% confidence.



As in any hypothesis test, we can also calculate a p-value. This will deliver the maximum confidence level at which we can reject the null.

```
pf(q, df1, df2, lower.tail = TRUE)
```

Notice that since the probability is calculated from the left by default (like the other commands), we can use the above code to automatically calculate $1 - p$.

12.3 Applications

Lets consider two applications. The first application is not terribly interesting, but it will illustrate a joint hypothesis test that is *always* provided to you free of charge with any set of regression results. The second application is more involved and delivers the true importance of joint tests.

Application 1: A wage application

This is the same scenario we considered for the dummy variable section, only without gender as a variable.

Suppose you are a consultant hired by a firm to help determine the underlying features of the current wage structure for their employees. You want to understand why some wage rates are different from others. Let our dependent variable be *wage* (the hourly wage of an individual employee) and the independent variables be given by…

- *educ* be the total years of education of an individual employee
- *exper* be the total years of experience an individual employee had prior to starting with the company
- *tenure* is the number of years an employee has been working with the firm.

The resulting PRF is given by…

$$wage_i = \beta_0 + \beta_1 educ_i + \beta_2 exper_i + \beta_3 tenure_i + \varepsilon_i$$

Suppose we wanted to test that none of these independent variables help explain movements in wages, so the resulting joint hypothesis would be

$$H_0 : \beta_1 = \beta_2 = \beta_3 = 0 \quad \text{versus} \quad H_1 : \beta_1 \neq 0,\ \beta_2 \neq 0,\ \text{or } \beta_3 \neq 0$$

The unrestricted model is one where each of the coefficients can be whatever number the data wants them to be.

```
data(wage1, package = "wooldridge")
UREG <- lm(wage~educ+exper+tenure,data=wage1)
(R2u <- summary(UREG)$r.squared)
```

```
## [1] 0.3064224
```

Our unrestricted model can explain roughly 30% of the variation in wages.

The next step is to estimate the restricted model - the model with the null hypothesis imposed. In this case you will notice that setting all slope coefficients to zero results in a rather strange looking model:

$$wage_i = \beta_0 + \varepsilon_i$$

This model contains no independent variables. If you were to estimate this model, then the intercept term would return the average wage in the data and the error term will simply be every deviation from the individual wage observations with it's average value. Since it is impossible for the deterministic component of this model to explain *any* of the variation in wages, then this implies that the restricted R^2 is zero by definition. Note that this is only a special case because of what the restricted model looks like. There will be more interesting cases where the restricted R^2 will need to be determined by estimating a restricted model.

```
R2r <- 0 # By definition
```

Now that we have the restricted and unrestricted R^2, we need the degrees of freedom to calculate an F-statistic under the null. The numerator degrees of freedom (m) denotes how many restrictions we placed on the restricted model. Since the null hypothesis sets all three slope coefficients to zero, we consider this to be 3 restrictions. The denominator degrees of freedom ($n-k-1$) is taken directly from the unrestricted model. Since $n = 526$ and we originally had 3 independent variables ($k = 3$), the denominator degrees of freedom is $n-k-1 = 522$. We can now calculate our F statistic under the null as well as our p-value.

```
m = 3; n = 526; k = 3

(Fstat <- ((R2u - R2r)/m)/((1-R2u)/(n-k-1)))
```

```
## [1] 76.87317
```

```
(Pval <- pf(Fstat,m,n-k-1,lower.tail = FALSE))
```

```
## [1] 3.405862e-41
```

```
(1-Pval)
```

```
## [1] 1
```

Note that since our F-statistic is far from 0, we can reject the null with approximately 100% confidence (i.e. the p-value is essentially zero).

What can we conclude from this?

Since we rejected the null hypothesis, that means we have statistical evidence that the alternative hypothesis is true. However, take a look at the what the alternative hypothesis actually says. It says that *at least one* of the population coefficients are statistically different from zero. It doesn't say which ones. It doesn't say how many. That's it…

Is there a short cut?

Remember that all regression results provide the simple hypothesis that each slope coefficient is equal to zero.

$$H_0 : \beta = 0 \quad \text{versus} \quad H_1 : \beta \neq 0$$

All regression results also provide the joint hypothesis that all slope coefficients are equal to zero. You can see the result at the bottom of the summary page. The last line delivers the same F-statistic we calculated above as well as a p-value that is essentially zero.

Note that while this uninteresting joint hypothesis test is done by default. Other joint tests require a bit more work.

```
summary(UREG)
```

```
##
## Call:
## lm(formula = wage ~ educ + exper + tenure, data = wage1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6068 -1.7747 -0.6279  1.1969 14.6536
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.87273    0.72896  -3.941 9.22e-05 ***
## educ         0.59897    0.05128  11.679  < 2e-16 ***
## exper        0.02234    0.01206   1.853   0.0645 .
## tenure       0.16927    0.02164   7.820 2.93e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.084 on 522 degrees of freedom
## Multiple R-squared:  0.3064, Adjusted R-squared:  0.3024
## F-statistic: 76.87 on 3 and 522 DF,  p-value: < 2.2e-16
```

Application 2: Constant Returns to Scale

Suppose you have data on the Gross Domestic Product (GDP) of a country as well as observations on two aggregate inputs of production: the nation's capital stock (K) and aggregate labor supply (L). One popular regression to run in growth economics is to see if a nation's aggregate production function possesses *constant returns to scale*. If it does, then if you scale up a nation's inputs by a particular percentage, then you will get the exact same percentage increase in output (i.e., double the inputs results in double the outputs). This has implications for what the size an economy should be, but we won't get into those details now.

The PRF is given by

$$lnGDP_i = \beta_0 + \beta_K \, lnK_i + \beta_L \, lnL_i + \varepsilon_i$$

where

- $lnGDP_i$ is an observation of total output
- lnK_i is an observation of total capital stock
- lnL_i is an observation of total labor stock.

These variables are actually in *logs*, but we will ignore that for now.

If we are testing for constant returns to scale, then we want to show that increasing all of the inputs by a certain amount will result in the same increase in output. Technical issues aside, this results in the following null hypothesis for a joint test:

$$H_0 : \beta_K + \beta_L = 1 \quad \text{versus} \quad H_1 : \beta_K + \beta_L \neq 1$$

We now have all we need to test for CRS:

```
# Load data...

library(readxl)
CDdata <- read_excel("data/CDdata.xlsx")

# Run unrestricted model, get R^2...

UREG <- lm(lnGDP ~ lnK + lnL, data = CDdata)
(R2u <- summary(UREG)$r.squared)
```

```
## [1] 0.9574247
```

The unrestricted model can explain around 96% of the variation in the dependent variable. For us to determine how much the restricted model can explain,

we first need to see exactly what the restriction does to our model. Starting from the unrestricted model, imposing the restriction delivers the following:

$$lnGDP_i = \beta_0 + \beta_K\, lnK_i + \beta_L\, lnL_i + \varepsilon_i$$
$$lnGDP_i = \beta_0 + (1-\beta_L)\, lnK_i + \beta_L\, lnL_i + \varepsilon_i$$

$$(lnGDP_i - lnK_i) = \beta_0 + \beta_L\, (lnL_i - lnK_i) + \varepsilon_i$$
$$\tilde{Y}_i = \beta_0 + \beta_L\, \tilde{X}_i + \varepsilon_i$$

where

$$\tilde{Y}_i = lnGDP_i - lnK_i \quad \text{and} \quad \tilde{X}_i = lnL_i - lnK_i$$

Notice how these derivations deliver exactly how the variables of the model need to be transformed and what the restricted model needs to be estimated.

```
Y = CDdata$lnGDP - CDdata$lnK
X = CDdata$lnL - CDdata$lnK

RREG <- lm(Y~X)
(R2r <- summary(RREG)$r.squared)
```

```
## [1] 0.9370283
```

The restricted model can explain roughly 94% of the variation in the dependent variable. To see if this reduction in R^2 is enough to reject the null hypothesis, we need to calculate an F-statistic. The numerator degrees of freedom is $m = 1$ because there is technically only one restriction in the null. The denominator degrees of freedom uses $n = 24$ and $k = 2$.

```
m = 1; n = 24; k = 2

(Fstat <- ((R2u - R2r)/m)/((1-R2u)/(n-k-1)))
```

```
## [1] 10.0604
```

```
(Pval <- pf(Fstat,m,n-k-1,lower.tail = FALSE))
```

```
## [1] 0.004594084
```

```
(1-Pval)
```

```
## [1] 0.9954059
```

As in the previous application, we received a very high F-statistic and a very low p-value. This means we *reject* the hypothesis that this country has an aggregate production function that exhibits constant returns to scale with slightly over 99.5% confidence.