

Computational Probability and Statistics

Ken Horton

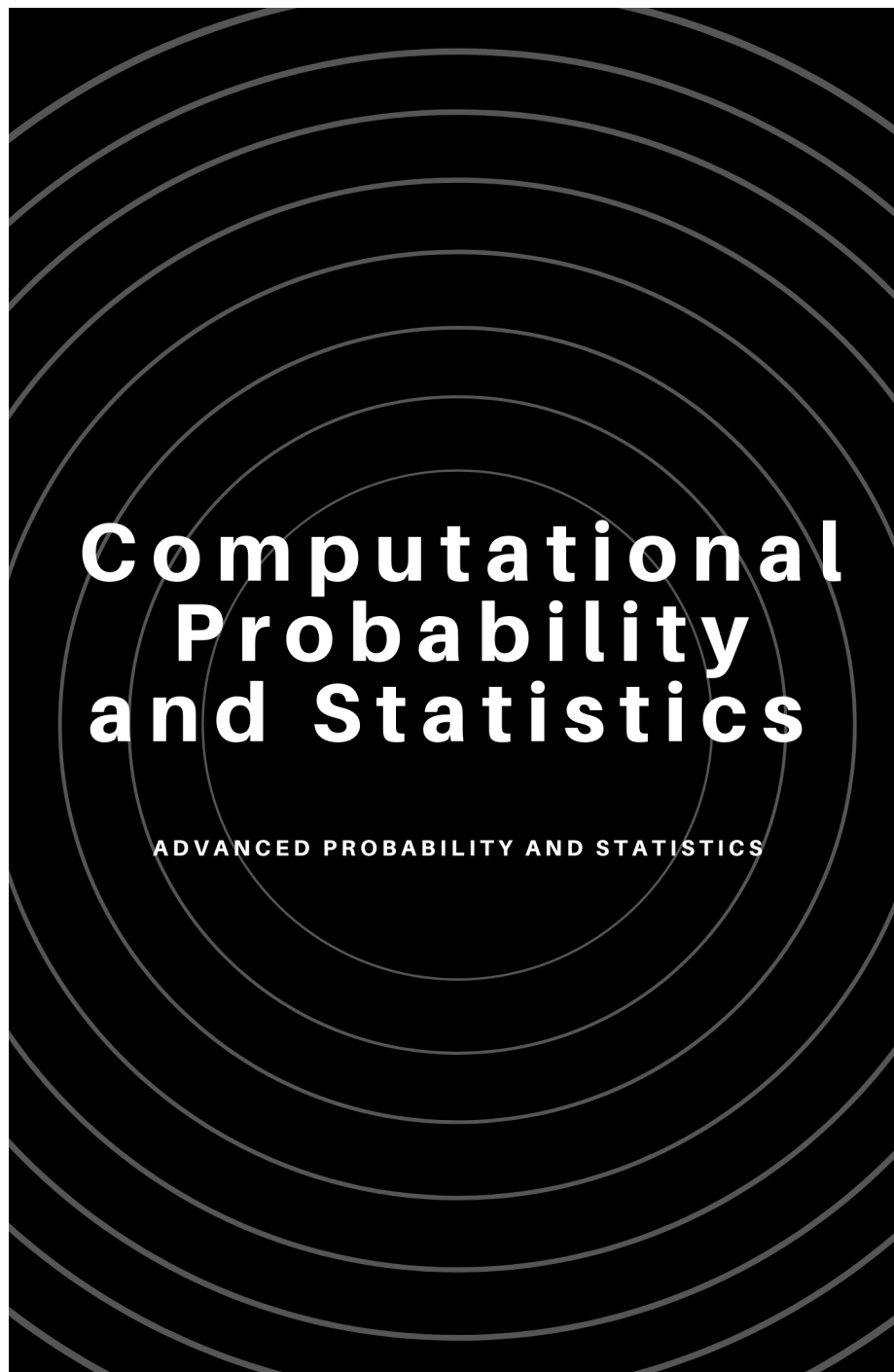
Kris Pruitt

Bradley Warner

2021-07-26

Contents

Preface



This book is based on the notes we created for our students as part of a one semester course on probability and statistics. We developed these notes from three primary resources. The most important is the Openintro Introductory Statistics with Randomization and Simulation (?) book. In parts we have used their notes and homework problems. However, in most cases we have altered their work to fit our needs. The second most important book for our work is Introductory to Probability and Statistics with R (?). Finally, we have used some examples, code, and ideas from the first addition of Prium's book Foundations and Applications of Statistics: An Introduction Using R (?).

0.1 Who is this book for?

We designed this book for study of statistics that maximizes computational ideas while minimizing algebraic symbol manipulation. Although we do discuss traditional small sample normal based inference and some of the classical probability distributions, we rely heavily on ideas such as simulation, permutations, and bootstrap. This means that students with a background in differential and integral calculus will be successful with this book.

The book makes extensive use of the R programming language. In particular we focus both on the **tidyverse** and **mosaic** packages. We include a significant amount of code in our notes and frequently demonstrate multiple ways of completing a task. We have used this book for juniors and sophomores.

0.2 Book Structure and How to Use It

The book is divided into 4 parts. Each part starts with a case study that introduces many of the main ideas of each part. Each chapter is designed to be a standalone 50 minute lesson. Within each lesson, we give exercises that can be worked in class and we have learning objectives.

This book assumes students have access to R. Finally, we keep the number of homework problems to a reasonable level and assign all problems.

The four parts of the book are:

1. Descriptive Statistical Modeling: This part introduces the student to data collection methods, summary statistics, visual summaries, and exploratory data analysis.
2. Probability: We discuss the foundation ideas of probability, counting methods, and common distributions. We use both calculus and simulation to find moments and probabilities. We introduce basic ideas of multivariate probability. We include method of moments and maximum likelihood estimators.
3. Statistical Inference: We discuss many of the basic inference ideas found

in a traditional introductory statistics class but we add ideas of bootstrap and permutation methods.

4. Statistical Prediction: The final part introduces prediction methods mainly in the form of linear regression. This part does also include inference for regression.

The learning outcomes for this course are to use computational and mathematical statistical/probabilistic concepts for:

- a. Developing probabilistic models
- b. Developing statistical models for description, inference, and prediction
- c. Advancing practical and theoretical analytic experience and skills

0.3 Prerequisites

To take this course, students are expected to have completed calculus up through and including integral calculus. We do have multivariate ideas in the course but they are easily taught and don't require calculus III. We don't assume the students have any programming experience and thus we include a great deal of code. We have historically supplemented the course with Data Camp courses. We have also used Rstudio Cloud to help students get started without the burden of loading and maintaining software.

0.4 Packages

These notes make use of the following packages in R **knitr** (?), **rmarkdown** (?), **mosaic** (?), **mosaicCalc** (?), **tidyverse** (?), **ISLR** (?), **vcd** (?), **ggplot2** (?), **MASS** (?), **openintro** (?), **broom** (?), **infer** (?), **ISLR** (?), **kableExtra** (?), **DT** (?).

0.5 Acknowledgements

We have been lucky to have numerous open sources to help facilitate this work.

This book was written using the **bookdown** package (?).



This book is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

0.6 File Creation Information

- File creation date: 2021-07-26
- R version 4.1.0 (2021-05-18)

Part I

**Descriptive Statistical
Modeling**

Chapter 1

Case Study

1.1 Objectives

- 1) Use R for basic analysis and visualization.
- 2) Compile a report using `knitr`.

1.2 Introduction to Descriptive Statistical Modeling

In this first block of material we will focus on data types, collection methods, summaries, and visualizations. We also intend to introduce computing via the R package. Programming in R requires some focus early in the course and we will supplement with some online courses. There is relatively little mathematics in this first block.

1.3 The data analytic process

Scientists seek to answer questions using rigorous methods and careful observations. These observations – collected from the likes of field notes, surveys, and experiments – form the backbone of a statistical investigation and are called **data**. Statistics is the study of how best to collect, analyze, and draw conclusions from data. It is helpful to put statistics in the context of a general process of investigation:

1. Identify a question or problem.
2. Collect relevant data on the topic.
3. Explore and understand the data.

4. Analyze the data.
5. Form a conclusion.
6. Make decisions based on the conclusion.

This is typical of an explanatory process because it starts with a research question and proceeds. However, sometimes an analysis is exploratory. There is data but not necessarily a research question. The purpose of the analysis is to find interesting features in the data and sometimes generate hypotheses. In this course we focus on the explanatory aspects of analysis but we have examples of exploratory.

Statistics as a subject focuses on making stages 2-5 objective, rigorous, and efficient. That is, statistics has three primary components:

- How best can we collect data?
- How should it be analyzed?
- And what can we infer from the analysis?

The topics scientists investigate are as diverse as the questions they ask. However, many of these investigations can be addressed with a small number of data collection techniques, analytic tools, and fundamental concepts in statistical inference. This lesson provides a glimpse into these and other themes we will encounter throughout the rest of the course.

1.4 Case study

In this lesson we will consider an experiment that studies effectiveness of stents in treating patients at risk of stroke.^{1 2} Stents are small mesh tubes that are placed inside narrow or weak arteries to assist in patient recovery after cardiac events and reduce the risk of an additional heart attack or death. Many doctors have hoped that there would be similar benefits for patients at risk of stroke. We start by writing the principal question the researchers hope to answer:

1.4.1 Research question

Does the use of stents reduce the risk of stroke?

1.4.2 Collect the relevant data

The researchers who asked this question collected data on 451 at-risk patients. Each volunteer patient was randomly assigned to one of two groups:

¹Chimowitz MI, Lynn MJ, Derdeyn CP, et al. 2011. Stenting versus Aggressive Medical Therapy for Intracranial Arterial Stenosis. *New England Journal of Medicine* 365:993-1003.

²NY Times article reporting on the study

Treatment group. Patients in the treatment group received a stent and medical management. The medical management included medications, management of risk factors, and help in lifestyle modification.

Control group. Patients in the control group received the same medical management as the treatment group but did not receive stents.

Researchers randomly assigned 224 patients to the treatment group and 227 to the control group. In this study, the control group provides a reference point against which we can measure the medical impact of stents in the treatment group.

This is an experiment and not an observational study. We will learn more about these ideas in this block.

Researchers studied the effect of stents at two time points: 30 days after enrollment and 365 days after enrollment.

1.4.3 Import data

We begin our first use of R

If you need to install a package, most likely it will be on CRAN, the Comprehensive R Archive Network. Before a package can be used, it must be installed on the computer (once per computer or account) and loaded into a session (once per R session). When you exit R, the package stays installed on the computer but will not be reloaded when R is started again.

In summary, R has packages that can be downloaded and installed from online repositories such as CRAN. When you install a package, which only needs to be done once per computer or account, in R all it is doing is placing the source code in a library folder designated during the installation of R. Packages are typically collections of functions and variables that are specific to a certain task or subject matter.

For example, to install the mosaic package, enter:

```
install.packages("mosaic") # fetch package from CRAN
```

In RStudio there is a *Packages* tab that makes it easy to add and maintain packages.

To use a package in a session, we must load it which makes it available to the current session only. When you start R again, you will have to load packages again. The command `library()` with the package name supplied as the argument is all that is needed. For this session, we will load `tidyverse` and `mosaic`. Note: the box below is executing the R commands, this is known as reproducible research since you can see the code and then you can run or modify as you need.

```
library(tidyverse)
library(mosaic)
```

Next read in the data into the working environment.

```
stent_study <- read_csv("data/stent_study.csv")
```

Let's break this code down. We are reading from a .csv file and assigning the results into an object called `stent_study`. The assignment arrow `<-` means we assign what is on the right to what is on the left. The R function we use in this case is `read_csv()`; when using R functions, you should ask yourself:

1. What do I want R to do?
2. What information must I provide for R to do this?

We want R to read in a .csv file. We can get help on this function by typing `?read_csv` at the prompt. The only required input to `read_csv()` is the file location. We have our data stored in a folder called "data" under the working directory. We can determine the working directory by typing `getwd()` at the prompt.

```
getwd()
```

Similarly, if we wish to change the working directory, we can do so by using the `setwd()` function:

```
setwd('C:/Users/Brad.Warner/Documents/Classes/Math 377/Another Folder')
```

In R if you use the `view()`, you will see the data in what looks like a standard spreadsheet.

```
view(stent_study)
```

1.4.4 Explore data

Before we attempt to answer the research question, let's look at the data. We want R to print out the first 10 rows of the data. The appropriate function is `head()` and it needs the data object. By default, R will output the first 6 rows. By using the `n=` argument, we can specify how many rows we want to view.

```
head(stent_study, n=10)
```

```
## # A tibble: 10 x 3
##   group    outcome30 outcome365
##   <chr>    <chr>      <chr>
## 1 control no_event    no_event
## 2 trmt     no_event    no_event
## 3 control no_event    no_event
## 4 trmt     no_event    no_event
## 5 trmt     no_event    no_event
## 6 control no_event    no_event
## 7 trmt     no_event    no_event
```



```
## 8 control no_event no_event
## 9 control no_event no_event
## 10 control no_event no_event
```

We also want to “inspect” the data. The function is `inspect()` and R needs the data object `stent_study`.

```
inspect(stent_study)

##
## categorical variables:
##      name      class levels  n missing
## 1    group character      2 451        0
## 2 outcome30 character      2 451        0
## 3 outcome365 character      2 451        0
##                                     distribution
## 1 control (50.3%), trmt (49.7%)
## 2 no_event (89.8%), stroke (10.2%)
## 3 no_event (83.8%), stroke (16.2%)
```

To keep things simple we will only look at the `outcome30` variable in this case study. We will summarize the data in a table. Later in the course, we will learn to do this using the `tidy` package; for now we use the `mosaic` package. This package makes use of the modeling formula that you will use extensively later in this course and in Math 378.

We want to summarize the data by making a table. In `mosaic` this is the `tally()` function. Before using this function, we have to understand the basic formula notation that `mosaic` uses. The basic format is:

```
goal( y ~ x, data = MyData, ... ) # pseudo-code for the formula template
```

We read `y ~ x` as “y tilde x” and interpret it in the equivalent forms: “y broken down by x”; “y modeled by x”; “y explained by x”; “y depends on x”; or “y accounted for by x.” For graphics, it’s reasonable to read the formula as “y vs. x”, which is exactly the convention used for coordinate axes.

For this exercise, we want to apply `tally()` to the variables `group` and `outcome30`. In this case it does not matter which we call `y` and `x`; however, it is more natural to think of `outcome30` as a dependent variable.

```
tally(outcome30~group,data=stent_study,margins = TRUE)
```

```
##           group
## outcome30 control trmt
##  no_event      214  191
##   stroke        13   33
##   Total         227  224
```

The `margins` option totals the columns.

Of the 224 patients in the treatment group, 33 had a stroke by the end of the first month. Using these two numbers, we can use R to compute the proportion of patients in the treatment group who had a stroke by the end of their first month.

```
33/(33+191)
```

```
## [1] 0.1473214
```

Exercise:

What proportion of the control group had a stroke? And why is this answer different from what `inspect()` reports?

Let's have R calculate proportions for us. Use `?` to look at the help menu for `tally()`. Note that one of the option arguments of the `tally()` function is `format=`. Setting this equal to `proportion` will output the proportions instead of the counts.

```
tally(outcome30~group,data=stent_study,format='proportion',margins = TRUE)
```

```
##           group
## outcome30 control      trmt
## no_event 0.94273128 0.85267857
## stroke   0.05726872 0.14732143
## Total    1.00000000 1.00000000
```

We can compute summary statistics from the table. A **summary statistic** is a single number summarizing a large amount of data.³ For instance, the primary results of the study after 1 month could be described by two summary statistics: the proportion of people who had a stroke in the treatment and control groups.

- Proportion who had a stroke in the treatment (stent) group: $33/224 = 0.15 = 15\%$.
- Proportion who had a stroke in the control group: $13/227 = 0.06 = 6\%$.

1.4.5 Visualize the data

It is often important to visualize the data. The table is a type of visualization but in this section we will introduce a graphical method called bar charts.

We will use the **ggformula** package to visualize. It is a wrapper to the **ggplot2** package which is becoming the industry standard for generating professional graphics. However, its interface is difficult to learn and we will ease into by using **ggformula** which makes use of the formula notation introduced above. The **ggformula** package was loaded when we loaded **mosaic**.⁴

³Formally, a summary statistic is a value computed from the data. Some summary statistics are more useful than others.

⁴<https://cran.r-project.org/web/packages/ggformula/vignettes/ggformula-blog.html>

To generate a basic graphic, we need to ask ourselves what information we are trying to see, what particular type of graph is best, what corresponding R function to use, and what information that R function needs in order to build a plot. For categorical data we want a bar chart and the R function `gf_bar()` needs the data object and the variable(s) of interest.

Here is our first attempt. In Figure ??, we leave the `y` portion of our formula blank. Doing this implies that we simply want to view the number/count of `outcome30` by type. We will see the two levels of `outcome30` on the x-axis and counts on the y-axis.

```
gf_bar(~outcome30,data=stent_study)
```

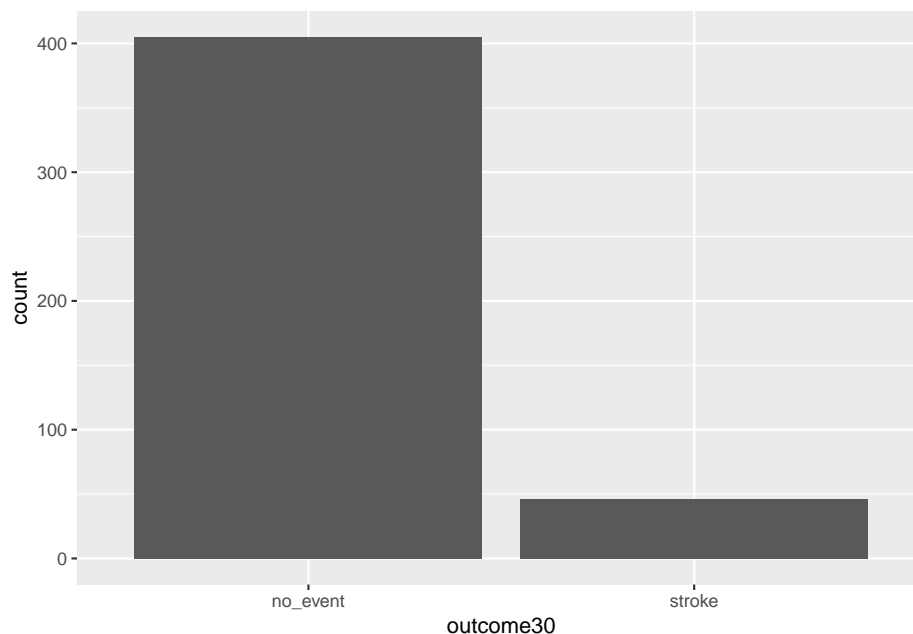


Figure 1.1: Using **ggformula** to create a bar chart.

Exercise:

Explain Figure ??..

This plot graphically shows us the total number of “stroke” and the total number of “no_event”. However, this is not what we want. We want to compare the 30-day outcomes for both treatment groups. So we need to break the data into different groups based on treatment type. In the formula language we now update it to the form:

```
goal( y ~ x|z, data = MyData, ... ) # pseudo-code for the formula template
```

We read `y ~ x|z` as “y tilde x by z” and interpret in the equivalent forms: “y

modeled by x for each z "; "y explained by x within each z "; or "y accounted for by x within z ." For graphics, it's reasonable to read the formula as "y vs. x for each z ". Figure ?? shows the results.

```
gf_bar(~outcome30|group,data = stent_study)
```

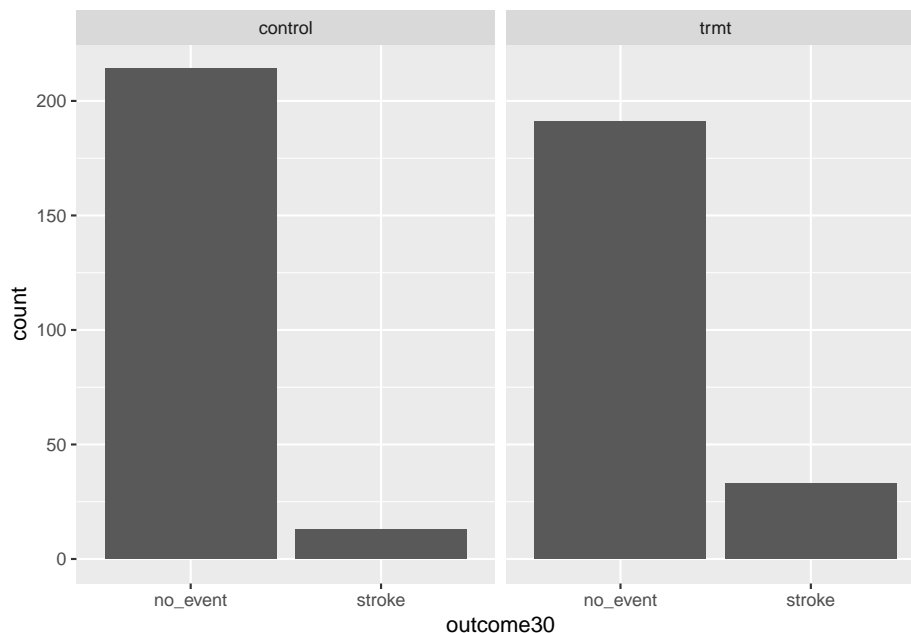


Figure 1.2: Bar charts conditioned on the `group` variable.

1.4.5.1 More advanced graphics

As a prelude for things to come, the above graphic needs work. The labels don't help; there is no title; we could add color; does it make more sense to use proportions? Here is the code and results for a better graph, see Figure ??.. Don't worry if this seems a bit advanced, but feel free to examine each new component of this code.

```
stent_study %>%
  gf_props(~group,fill=~outcome30,position='fill') %>%
  gf_labs(title="Impact of Stents of Stroke",
          subtitle='Experiment with 451 Patients',
          x="Experimental Group",
          y="Number of Events") %>%
  gf_theme(theme_bw())
```

Notice that we used the pipe operator, `%>%`. This operator allows us to string functions together in a manner that makes it easier to read the code. In

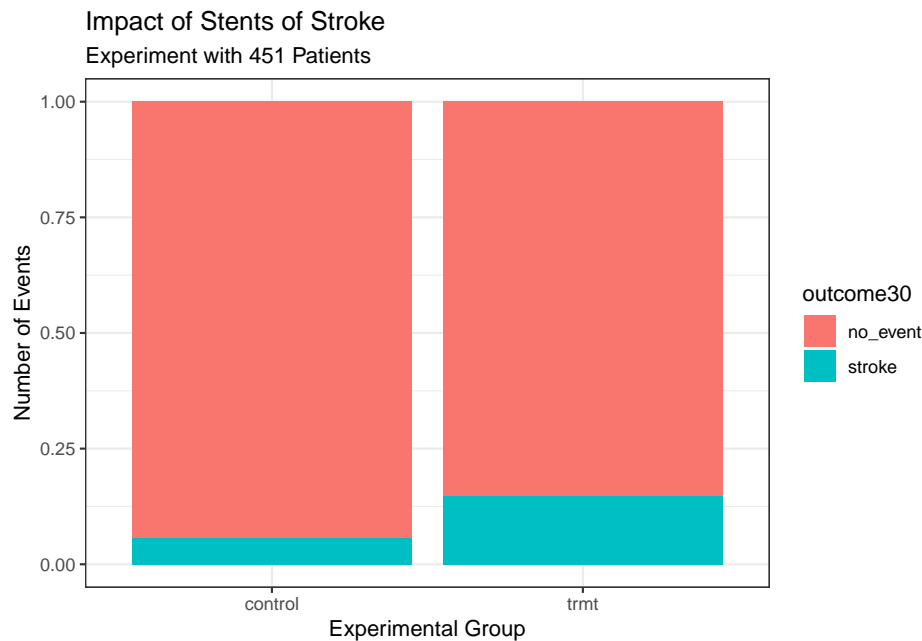


Figure 1.3: Better graph.

the above code we are sending the data object `stent_study` into the function `gf_props()` to use as data so we don't need the `data =` argument. In math, this is a composition of functions. Instead of $f(g(x))$ we could use a pipe $f(g(x)) = g(x) \%>\% f()$.

1.4.6 Conclusion

These two summary statistics are useful in looking for differences in the groups, and we are in for a surprise: an additional 9% of patients in the treatment group had a stroke! This is important for two reasons. First, it is contrary to what doctors expected, which was that stents would *reduce* the rate of strokes. Second, it leads to a statistical question: do the data show a **real** difference due to the treatment?

This second question is subtle. Suppose you flip a coin 100 times. While the chance a coin lands heads in any given coin flip is 50%, we probably won't observe exactly 50 heads. This type of fluctuation is part of almost any type of data generating process. It is possible that the 9% difference in the stent study is due to this natural variation. However, the larger the difference we observe (for a particular sample size), the less believable it is that the difference is due to chance. So what we are really asking is the following: is the difference so large that we should reject the notion that it was due to chance?

This is a preview of step 4, analyze the data, and step 5, form a conclusion, of the analysis cycle. While we haven't yet covered statistical tools to fully address these steps, we can comprehend the conclusions of the published analysis: there was compelling evidence of harm by stents in this study of stroke patients.

Be careful: do not generalize the results of this study to all patients and all stents. This study looked at patients with very specific characteristics who volunteered to be a part of this study and who may not be representative of all stroke patients. In addition, there are many types of stents and this study only considered the self-expanding Wingspan stent (Boston Scientific). However, this study does leave us with an important lesson: we should keep our eyes open for surprises.

1.5 Homework Problems

Create an Rmd file `01 Data Case Study Application.Rmd` in R, it may be provided, and start by inserting your name in the header. Code blocks below can be inserted and then you can complete the code and answer the questions. When you are done, **knit** it into a pdf file.

To create an R code chunk, use CTRL-ALT-I or on the **insert** tab of the window, use the drop down to select R. Anything between the dashes is interpreted as R code.

For more on RMarkdown see the video, <https://www.youtube.com/watch?v=DNS7i2m4sB0> This video assumes you are using R on your computer but we are using RStudio Cloud. Thus we can **knit** to a pdf since it is setup for us. You can also take the first chapter of the Data Camp course Reporting with R Markdown to learn more.

1. **Stent study continued.** Complete a similar analysis for the stent data but this time for the one year data. In particular
 - a. Read the data into your working directory.

```
stent_study <-read_csv(____)
```

- b. Complete similar steps as in the class notes. The start of code is provided below.
 - i. Use **inspect** on the data.
 - ii. Create a table of **outcome365** and **group**. Comment on the results.
 - iii. Create a barchart of the data.

Summary

```
inspect(____)
```

Table

```
tally(outcome365~____,data=stent_study,format=____,margins = TRUE)
```

Barchart

```
stent_study %>%
  gf_props(~___,fill=~___,position='fill') %>%
  gf_labs(title=___
  subtitle=___,
  x=___,
  y=___)
```

2. **Migraine and acupuncture.** A migraine is a particularly painful type of headache, which patients sometimes wish to treat with acupuncture. To determine whether acupuncture relieves migraine pain, researchers conducted a randomized controlled study where 89 females diagnosed with migraine headaches were randomly assigned to one of two groups: treatment or control. 43 patients in the treatment group received acupuncture that is specifically designed to treat migraines. 46 patients in the control group received placebo acupuncture (needle insertion at nonacupoint locations). 24 hours after patients received acupuncture, they were asked if they were pain free.⁵

The data is in the file `migraine_study.csv` in the folder `data`.

Complete the following work:

- a. Read the data an object called `migraine_study`.

```
migraine_study <- read_csv("data/___")
```

```
head(migraine_study)
```

- b. Create a table of the data.

```
tally(___)
```

- c. Report the percent of patients in the treatment group who were pain free 24 hours after receiving acupuncture.
- d. Repeat for the control group.
- e. At first glance, does acupuncture appear to be an effective treatment for migraines? Explain your reasoning.
- f. Do the data provide convincing evidence that there is a real pain reduction for those patients in the treatment group? Or do you think that the observed difference might just be due to chance?

Compile, `knit`, this report into a pdf.

⁵G. Allais et al. “Ear acupuncture in the treatment of migraine attacks: a randomized trial on the efficacy of appropriate versus inappropriate acupoints”. In: *Neurological Sci.* 32.1 (2011), pp. 173–175.

Chapter 2

Data Basics

2.1 Objectives

- 1) Define and use properly in context all new terminology to include but not limited to case, observational unit, variables, data frame, associated variables, independent, and discrete and continuous variables.
- 2) Identify and define the different types of variables.
- 3) From reading a study, explain the research question.
- 4) Create a scatterplot in R and determine the association of two numerical variables from the plot.

2.2 Data basics

Effective presentation and description of data is a first step in most analyses. This lesson introduces one structure for organizing data as well as some terminology that will be used throughout this course.

2.2.1 Observations, variables, and data matrices

For reference we will be using a data set concerning 50 emails received in 2012. These observations will be referred to as the **email50** data set, and they are a random sample from a larger data set. This data is in the **openintro** package so let's load our packages.

```
library(usdata)
```

Table ?? shows 5 rows of the **email50** data set concerning 50 emails from 2012.

The data object `email50` is a subset of `email` and we have selected to only list 5 rows and 5 variables for ease of observation.

Each row in the table represents a single email or **case**.¹ The columns represent characteristics, called **variables**, for each of the emails. For example, the first row represents email 1, which is not spam, contains 21,705 characters, 551 line breaks, is written in HTML format, and contains only small numbers.

First 5 rows of email data frame

```
spam
num_char
line_breaks
format
number
0
21.705
551
1
small
0
7.011
183
1
big
1
0.631
28
0
none
0
15.829
242
1
```

¹A case is also sometimes called a **unit of observation** or an **observational unit**

small

Let's look at the first 10 rows of data from `email50` using R. Remember to ask the two questions:

What do we want R to do? and

What must we give R for it to do this?

We want the first 10 rows so we use `head` and R needs the data object and the number of rows. The data object is called `email50` and is accessible once the `openintro` package is loaded.

```
head(email50, n=10)
```

```
## # A tibble: 10 x 21
##   spam to_multiple from cc sent_email time image attach
##   <fct> <fct>      <fct> <int> <fct>      <dtm>      <dbl> <dbl>
## 1 0 0 1 0 1 2012-01-04 13:19:16 0 0
## 2 0 0 1 0 0 2012-02-16 20:10:06 0 0
## 3 1 0 1 4 0 2012-01-04 15:36:23 0 2
## 4 0 0 1 0 0 2012-01-04 17:49:52 0 0
## 5 0 0 1 0 0 2012-01-27 09:34:45 0 0
## 6 0 0 1 0 0 2012-01-17 17:31:57 0 0
## 7 0 0 1 0 0 2012-03-18 04:18:55 0 0
## 8 0 0 1 0 1 2012-03-31 13:58:56 0 0
## 9 0 0 1 1 1 2012-01-11 01:57:54 0 0
## 10 0 0 1 0 0 2012-01-07 19:29:16 0 0
## # ... with 13 more variables: dollar <dbl>, winner <fct>, inherit <dbl>,
## #   viagra <dbl>, password <dbl>, num_char <dbl>, line_breaks <int>,
## #   format <fct>, re_subj <fct>, exclaim_subj <dbl>, urgent_subj <fct>,
## #   exclaim_mess <dbl>, number <fct>
```

In practice, it is especially important to ask clarifying questions to ensure important aspects of the data are understood. For instance, it is always important to be sure we know what each variable means and the units of measurement. Descriptions of all variables in the `email50` data set are given in its documentation which can be accessed in R by using the `?` command:

```
?email50
```

(Note that not all data sets will have associated documentation; the authors of `openintro` package included this documentation with the `email50` data set contained in the package.)

The data in `email50` represent a **data matrix** or in R terminology **data frame** or **tibble**², which is a common way to organize data. Each row of a data matrix corresponds to a unique case, and each column corresponds to a variable. This

²A tibble is a data frame with attributes for such things as better display and printing

is called **tidy data**.³ The data frame for the stroke study introduced in the previous lesson had patients as the cases and there were three variables recorded for each patient. If we are thinking of patients as the unit of observation, then this data is tidy.

```
## # A tibble: 10 x 3
##   group outcome30 outcome365
##   <chr>   <chr>      <chr>
## 1 control no_event  no_event
## 2 trmt    no_event  no_event
## 3 control no_event  no_event
## 4 trmt    no_event  no_event
## 5 trmt    no_event  no_event
## 6 control no_event  no_event
## 7 trmt    no_event  no_event
## 8 control no_event  no_event
## 9 control no_event  no_event
## 10 control no_event  no_event
```

If we think of an outcome as a unit of observation then it is not tidy since the two outcome columns are variable values (month or year). The tidy data for this case would be:

```
## # A tibble: 10 x 4
##   patient_id group   time result
##         <int> <chr>   <chr> <chr>
## 1           1 control month no_event
## 2           1 control year  no_event
## 3           2 trmt   month no_event
## 4           2 trmt   year  no_event
## 5           3 control month no_event
## 6           3 control year  no_event
## 7           4 trmt   month no_event
## 8           4 trmt   year  no_event
## 9           5 trmt   month no_event
## 10          5 trmt   year  no_event
```

There are three interrelated rules which make a data set tidy:

1. Each variable must have its own column.
2. Each observation must have its own row.
3. Each value must have its own cell.

Why ensure that your data is tidy? There are two main advantages:

³For more information on tidy data see the blog and the book.

1. There's a general advantage to picking one consistent way of storing data. If you have a consistent data structure, it's easier to learn the tools that work with it because they have an underlying uniformity.
2. There's a specific advantage to placing variables in columns because it allows R's vectorised nature to shine. This will be more clear as the semester progresses. Since most built-in R functions work with vectors of values, it makes transforming tidy data feel particularly natural.

Data frames are a convenient way to record and store data. If another individual or case is added to the data set, an additional row can be easily added. Similarly, another column can be added for a new variable.

Exercise:

We consider a publicly available data set that summarizes information about the 3,142 counties in the United States, and we create a data set called `county_M377` data set. This data set will include information about each county: its name, the state where it resides, its population in 2000 and 2010, per capita federal spending, poverty rate, and four additional characteristics, we create this data object next. The parent data set is part of the `usdata` library and is called `county_complete`. The variables are summarized in help menu built into the `usdata` package⁴. How might these data be organized in a data matrix? ⁵

Using R we will create our data object.

```
library(usdata)
```

We only want a subset of the columns and we will use the `select` verb in `dplyr` to select and rename columns. We also create a new variable which is federal spending per capita.

```
county_M377 <- county_complete %>%
  select(name, state, pop2000, pop2010, fed_spend=fed_spending_2009, poverty=poverty_2010,
         homeownership = homeownership_2010, multi_unit = housing_multi_unit_2010,
         income = per_capita_income_2010, med_income = median_household_income_2010) %>%
  mutate(fed_spend=fed_spend/pop2010)
```

Using R, we will display seven rows of the `county` data frame.

```
head(county_M377,n=7)
```

##		name	state	pop2000	pop2010	fed_spend	poverty	homeownership
## 1	Autauga County	Alabama	43671	54571	6.068095	10.6	77.5	
## 2	Baldwin County	Alabama	140415	182265	6.139862	12.2	76.7	

⁴These data were collected from the US Census website.

⁵Each county may be viewed as a case, and there are ten pieces of information recorded for each case. A table with 3,142 rows and 10 columns could hold these data, where each row represents a county and each column represents a particular piece of information.

```
## 3 Barbour County Alabama 29038 27457 8.752158 25.0 68.0
## 4 Bibb County Alabama 20826 22915 7.122016 12.6 82.9
## 5 Blount County Alabama 51024 57322 5.130910 13.4 82.0
## 6 Bullock County Alabama 11714 10914 9.973062 25.3 76.9
## 7 Butler County Alabama 21399 20947 9.311835 25.0 69.0
## multi_unit income med_income
## 1 7.2 24568 53255
## 2 22.6 26469 50147
## 3 11.1 15875 33219
## 4 6.6 19918 41770
## 5 3.7 21070 45549
## 6 9.9 20289 31602
## 7 13.7 16916 30659
```

2.2.2 Types of variables

Examine the `fed_spend`, `pop2010`, and `state` variables in the `county` data set. Each of these variables is inherently different from the other two yet many of them share certain characteristics.

First consider `fed_spend`, which is said to be a **numerical variable** since it can take a wide range of numerical values, and it is sensible to add, subtract, or take averages with those values. On the other hand, we would not classify a variable reporting telephone area codes as numerical; even though area codes are made up of numerical digits, their average, sum, and difference have no clear meaning.

The `pop2010` variable is also numerical; it is sensible to add, subtract, or take averages with those values, although it seems to be a little different than `fed_spend`. This variable of the population count can only be a whole non-negative number (0, 1, 2, ...). For this reason, the population variable is said to be **discrete** since it can only take specific numerical values. On the other hand, the federal spending variable is said to be **continuous**. Now technically, there are no truly continuous numerical variables since all measurements are finite up to some level of accuracy or measurement precision. However, in this course we will treat both variables types of numerical variables the same, that is as continuous variables. The only place this will be different is in probability models which we see in the next block.

The variable `state` can take up to 51 values after accounting for Washington, DC: *AL*, ..., and *WY*. Because the responses themselves are categories, `state` is called a **categorical variable**,⁶ and the possible values are called the variable's **levels**.

Finally, consider a hypothetical variable on education, which describes the highest level of education completed and takes on one of the values *noHS*, *HS*, *College*

⁶Sometimes also called a **nominal** variable.

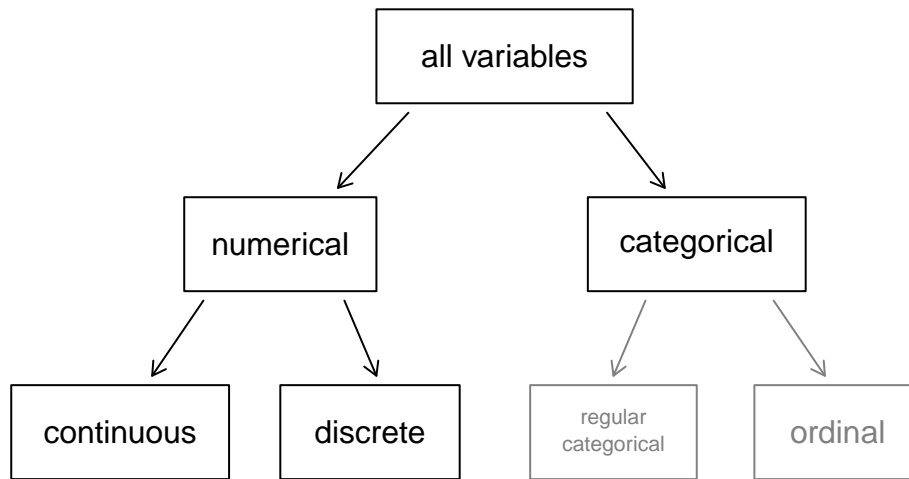


Figure 2.1: Taxonomy of Variables.

or *Graduate_school*. This variable seems to be a hybrid: it is a categorical variable but the levels have a natural ordering. A variable with these properties is called an **ordinal** variable. To simplify analyses, any ordinal variables in this course will be treated as categorical variables. In **R** categorical variables can be treated in different ways; one of the key differences is that we can leave them as character values or as factors. When **R** handles factors, it is only concerned about the *levels* of values of the factors. We will learn more about this as the semester progresses.

Figure ?? captures this classification of variables.

Exercise:

Data were collected about students in a statistics course. Three variables were recorded for each student: number of siblings, student height, and whether the student had previously taken a statistics course. Classify each of the variables as continuous numerical, discrete numerical, or categorical.

The number of siblings and student height represent numerical variables. Because the number of siblings is a count, it is discrete. Height varies continuously, so it is a continuous numerical variable. The last variable classifies students into two categories – those who have and those who have not taken a statistics course – which makes this variable categorical.

Exercise:

Consider the variables `group` and `outcome30` from the stent study in the case study lesson. Are these numerical or categorical variables?

7

⁷There are only two possible values for each variable, and in both cases they describe

2.2.3 Relationships between variables

Many analyses are motivated by a researcher looking for a relationship between two or more variables, this is the heart of statistical modeling. A social scientist may like to answer some of the following questions:

1. Is federal spending, on average, higher or lower in counties with high rates of poverty?
2. If homeownership is lower than the national average in one county, will the percent of multi-unit structures in that county likely be above or below the national average?

To answer these questions, data must be collected, such as the `county_complete` data set. Examining summary statistics could provide insights for each of the two questions about counties. Additionally, graphs can be used to visually summarize data and are useful for answering such questions as well.

Scatterplots are one type of graph used to study the relationship between two numerical variables. Figure ?? compares the variables `fed_spend` and `poverty`. Each point on the plot represents a single county. For instance, the highlighted dot corresponds to County 1088 in the `county_M377` data set: Owsley County, Kentucky, which had a poverty rate of 41.5% and federal spending of \$21.50 per capita. The dense cloud in the scatterplot suggests a relationship between the two variables: counties with a high poverty rate also tend to have slightly more federal spending. We might brainstorm as to why this relationship exists and investigate each idea to determine which is the most reasonable explanation.

Exercise:

Examine the variables in the `email150` data set. Create two questions about the relationships between these variables that are of interest to you.⁸

The `fed_spend` and `poverty` variables are said to be associated because the plot shows a discernible pattern. When two variables show some connection with one another, they are called **associated variables**. Associated variables can also be called **dependent** variables and vice-versa.

Example:

The relationship between the homeownership rate and the percent of units in multi-unit structures (e.g. apartments, condos) is visualized using a scatterplot in Figure ?. Are these variables associated?

It appears that the larger the fraction of units in multi-unit structures, the lower the homeownership rate. Since there is some relationship between the variables,

categories. Thus, each is a categorical variable.

⁸Two sample questions: (1) Intuition suggests that if there are many line breaks in an email then there would also tend to be many characters: does this hold true? (2) Is there a connection between whether an email format is plain text (versus HTML) and whether it is a spam message?

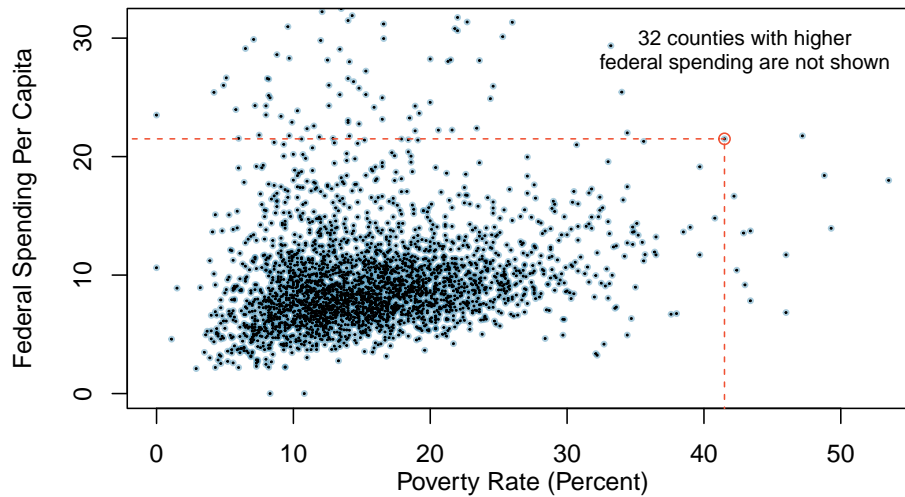


Figure 2.2: A scatterplot showing `fed_spend` against poverty. Owsley County of Kentucky, with a poverty rate of 41.5% and federal spending of \$21.50 per capita, is highlighted.

they are associated.

Because there is a downward trend in Figure ?? – counties with more units in multi-unit structures are associated with lower homeownership – these variables are said to be **negatively associated**. A **positive association** is shown in the relationship between the `poverty` and `fed_spend` variables represented in Figure ??, where counties with higher poverty rates tend to receive more federal spending per capita.

If two variables are not associated, then they are said to be **independent**. That is, two variables are independent if there is no evident relationship between the two.

A pair of variables are either related in some way (associated) or not (independent). No pair of variables is both associated and independent.

2.2.4 Creating a scatterplot

In this section we will create a simple scatterplot and then ask you to create one on your own. First we will recreate the scatterplot seen in Figure ?. This figure uses the `county_M377` data set.

Here are two questions:

What do we want R to do? and

What must we give R for it to do this?

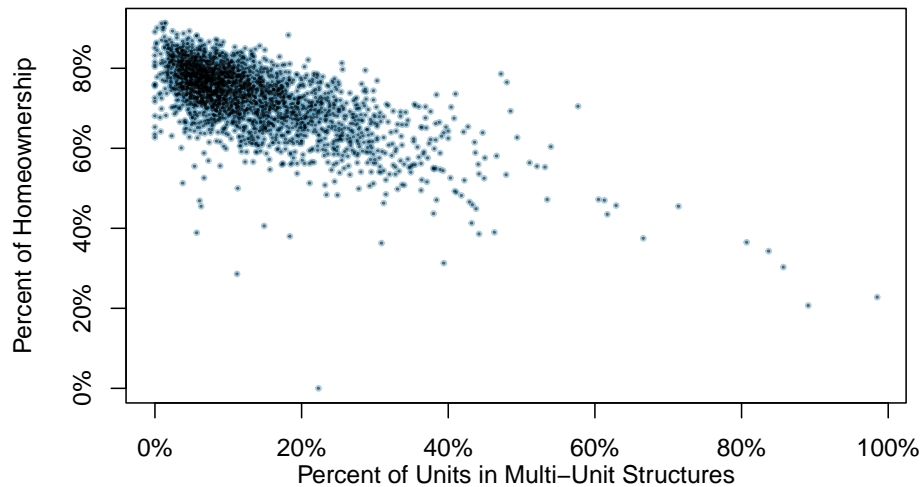


Figure 2.3: A scatterplot of the homeownership rate versus the percent of units that are in multi-unit structures for all 3,143 counties.

We want R to create a scatterplot and to do this it needs, at a minimum, the data object, what we want on the x -axis, and what we want on the y -axis. More information on `ggformula` can be found by clicking on the link.⁹

```
county_M377 %>%
  gf_point(fed_spend~poverty)
```

Figure ?? is bad, there are poor axis labels, no title, dense clustering of points, the y -axis is being driven by a couple of extreme points. We will need to clear this up. Again, try to read the code and use `help()` or `?` to determine the purpose of each command in Figure ??.

```
county_M377 %>%
  filter(fed_spend<32) %>%
  gf_point(fed_spend~poverty,
           xlab="Poverty Rate (Percent)",
           ylab="Federal Spending Per Capita",
           title="A scatterplot showing fed_spend against poverty",
           subtitle = "Owsley County of Kentucky",
           cex=1,alpha=0.2) %>%
  gf_theme(theme_classic())
```

Exercise:

Create the scatterplot in Figure ??.

⁹<https://cran.r-project.org/web/packages/ggformula/vignettes/ggformula-blog.html>

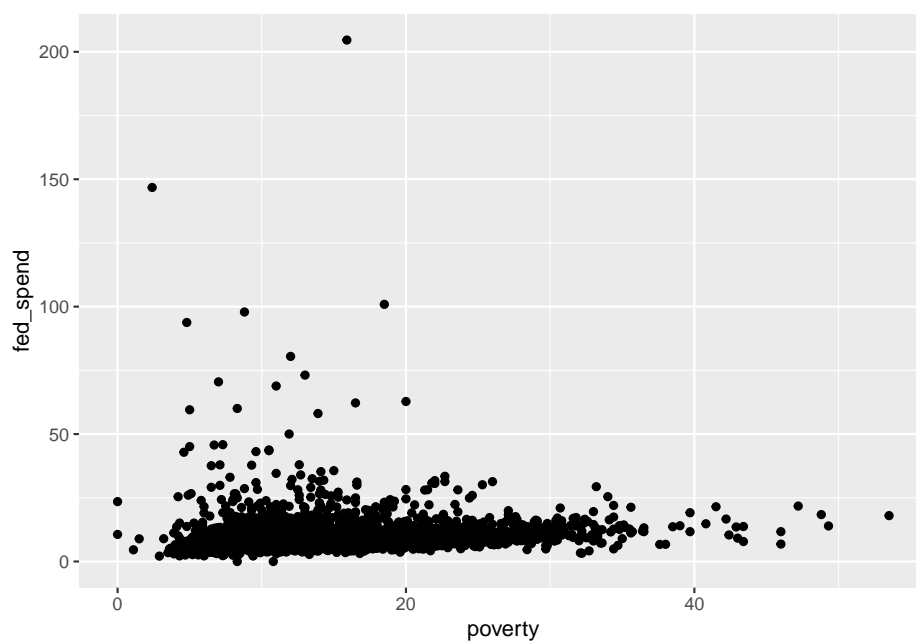
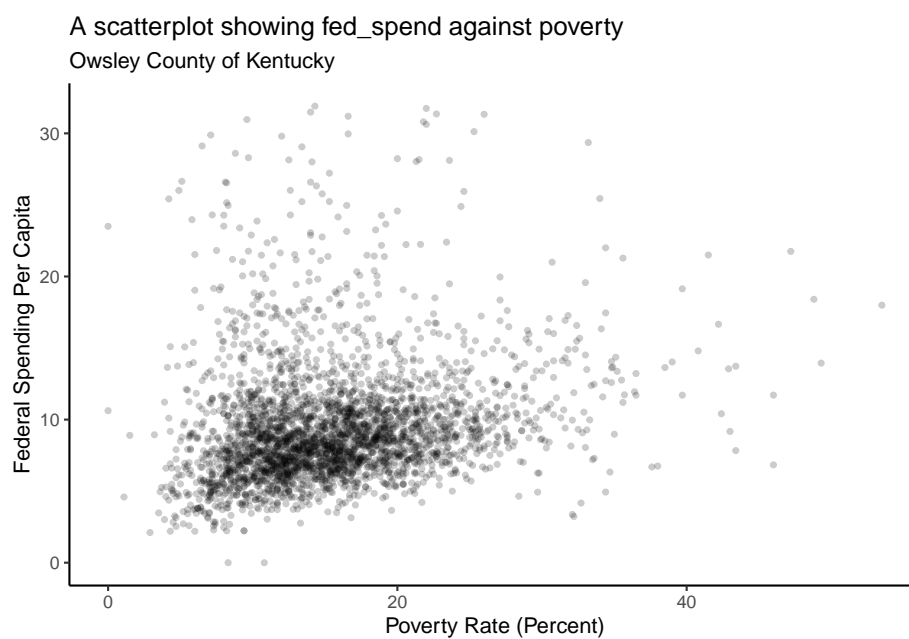
Figure 2.4: Scatterplot with `ggformula`.

Figure 2.5: Better example of a scatterplot.

2.3 Homework Problems

Identify study components

Identify (i) the cases, (ii) the variables and their types, and (iii) the main research question in the studies described below.

1. Researchers collected data to examine the relationship between pollutants and preterm births in Southern California. During the study air pollution levels were measured by air quality monitoring stations. Specifically, levels of carbon monoxide were recorded in parts per million, nitrogen dioxide and ozone in parts per hundred million, and coarse particulate matter (PM_{10}) in $\mu g/m^3$. Length of gestation data were collected on 143,196 births between the years 1989 and 1993, and air pollution exposure during gestation was calculated for each birth. The analysis suggested that increased ambient PM_{10} and, to a lesser degree, CO concentrations may be associated with the occurrence of preterm births.¹⁰
2. The Buteyko method is a shallow breathing technique developed by Konstantin Buteyko, a Russian doctor, in 1952. Anecdotal evidence suggests that the Buteyko method can reduce asthma symptoms and improve quality of life. In a scientific study to determine the effectiveness of this method, researchers recruited 600 asthma patients aged 18-69 who relied on medication for asthma treatment. These patients were split into two research groups: one practiced the Buteyko method and the other did not. Patients were scored on quality of life, activity, asthma symptoms, and medication reduction on a scale from 0 to 10. On average, the participants in the Buteyko group experienced a significant reduction in asthma symptoms and an improvement in quality of life.¹¹

¹⁰B. Ritz et al. "Effect of air pollution on preterm birth among children born in Southern California between 1989 and 1993". In: *Epidemiology* 11.5 (2000), pp. 502–511.

¹¹J. McGowan. "Health Education: Does the Buteyko Institute Method make a difference?" In: *Thorax* 58 (2003).

Chapter 3

Overview of Data Collection Principles

3.1 Objectives

- 1) Define and use properly in context all new terminology.
- 2) From a description of a research project, at a minimum be able to describe the population of interest, the generalizability of the study, the response and predictor variables, differentiate whether it is observational or experimental, and determine the type of sample.

3.2 Overview of data collection principles

The first step in conducting research is to identify topics or questions that are to be investigated. A clearly laid out research question is helpful in identifying what subjects or cases should be studied and what variables are important. It is also important to consider *how* data are collected so that they are reliable and help achieve the research goals.

3.2.1 Populations and samples

Consider the following three research questions:

1. What is the average mercury content in swordfish in the Atlantic Ocean?
2. Over the last 5 years, what is the average time to complete a degree for Duke undergraduate students?

3. Does a new drug reduce the number of deaths in patients with severe heart disease?

Each research question refers to a target **population**. In the first question, the target population is all swordfish in the Atlantic Ocean, and each fish represents a case. It is usually too expensive to collect data for every case in a population. Instead, a sample is taken. A **sample** represents a subset of the cases and is often a small fraction of the population. For instance, 60 swordfish (or some other number) in the population might be selected, and this sample data may be used to provide an estimate of the population average and answer the research question.

Exercise:

For the second and third questions above, identify the target population and what represents an individual case.¹

3.2.2 Anecdotal evidence

Consider the following possible responses to the three research questions:

1. A man on the news got mercury poisoning from eating swordfish, so the average mercury concentration in swordfish must be dangerously high.
2. I met two students who took more than 7 years to graduate from Duke, so it must take longer to graduate at Duke than at many other colleges.
3. My friend's dad had a heart attack and died after they gave him a new heart disease drug, so the drug must not work.

Each conclusion is based on data. However, there are two problems. First, the data only represent one or two cases. Second, and more importantly, it is unclear whether these cases are actually representative of the population. Data collected in this haphazard fashion are called **anecdotal evidence**.

Anecdotal evidence: Be careful of data collected haphazardly. Such evidence may be true and verifiable, but it may only represent extraordinary cases.

Anecdotal evidence typically is composed of unusual cases that we recall based on their striking characteristics. For instance, we are more likely to remember the two people we met who took 7 years to graduate than the six others who graduated in four years. Instead of looking at the most unusual cases, we should examine a sample of many cases that represent the population.

¹2) Notice that the first question is only relevant to students who complete their degree; the average cannot be computed using a student who never finished her degree. Thus, only Duke undergraduate students who have graduated in the last five years represent cases in the population under consideration. Each such student would represent an individual case. 3) A person with severe heart disease represents a case. The population includes all people with severe heart disease.



Figure 3.1: In February 2010, some media pundits cited one large snow storm as evidence against global warming. As comedian Jon Stewart pointed out, ‘*It’s one storm, in one region, of one country.*’

3.2.3 Sampling from a population

We might try to estimate the time to graduation for Duke undergraduates in the last 5 years by collecting a sample of students. All graduates in the last 5 years represent the *population*, and graduates who are selected for review are collectively called the *sample*. In general, we always seek to *randomly* select a sample from a population. The most basic type of random selection is equivalent to how raffles are conducted. For example, in selecting graduates, we could write each graduate’s name on a raffle ticket and draw 100 tickets. The selected names would represent a random sample of 100 graduates. This is illustrated in Figure ??.

Why pick a sample randomly? Why not just pick a sample by hand? Consider the following scenario.

Example:

Suppose we ask a student who happens to be majoring in nutrition to select several graduates for the study. What kind of students do you think she might collect? Do you think her sample would be representative of all graduates? ²

²Perhaps she would pick a disproportionate number of graduates from health-related fields. Or perhaps her selection would be well-representative of the population. When selecting samples by hand, we run the risk of picking a *biased* sample, even if that bias is unintentional

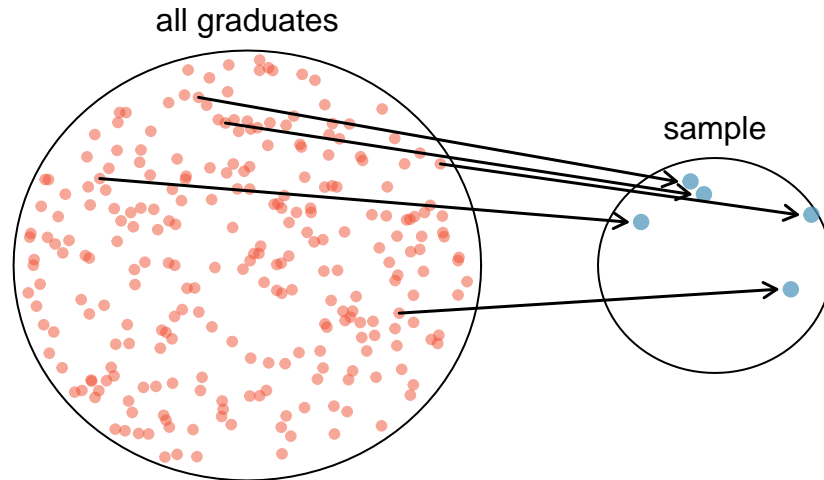


Figure 3.2: In this graphic, five graduates are randomly selected from the population to be included in the sample.

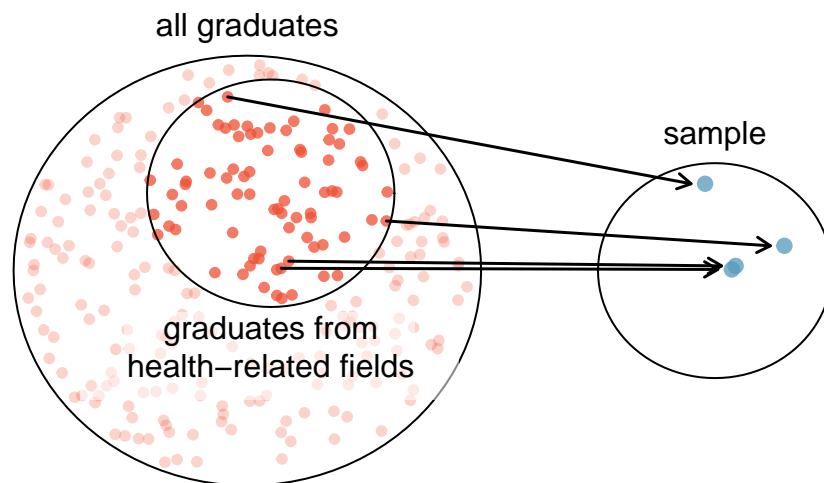


Figure 3.3: Instead of sampling from all graduates equally, a nutrition major might inadvertently pick graduates with health-related majors disproportionately often.

If someone was permitted to pick and choose exactly which graduates were included in the sample, it is entirely possible that the sample could be skewed to that person's interests, which may be entirely unintentional. This introduces **bias** into a sample, see Figure ???. Sampling randomly helps resolve this problem. The most basic random sample is called a **simple random sample**, which is equivalent to using a raffle to select cases. This means that each case in the population has an equal chance of being included and there is no implied connection between the cases in the sample.

Sometimes a simple random sample is difficult to implement and an alternative method is helpful. One such substitute is a **systematic sample**, where one case is sampled after letting a fixed number of others, say 10 other cases, pass by. Since this approach uses a mechanism that is not easily subject to personal biases, it often yields a reasonably representative sample. This course will focus on simple random samples since the use of systematic samples is uncommon and requires additional considerations of the context.

The act of taking a simple random sample helps minimize bias. However, bias can crop up in other ways. Even when people are picked at random, e.g. for surveys, caution must be exercised if the **non-response** is high. For instance, if only 30% of the people randomly sampled for a survey actually respond, and it is unclear whether the respondents are **representative** of the entire population, the survey might suffer from **non-response bias**.

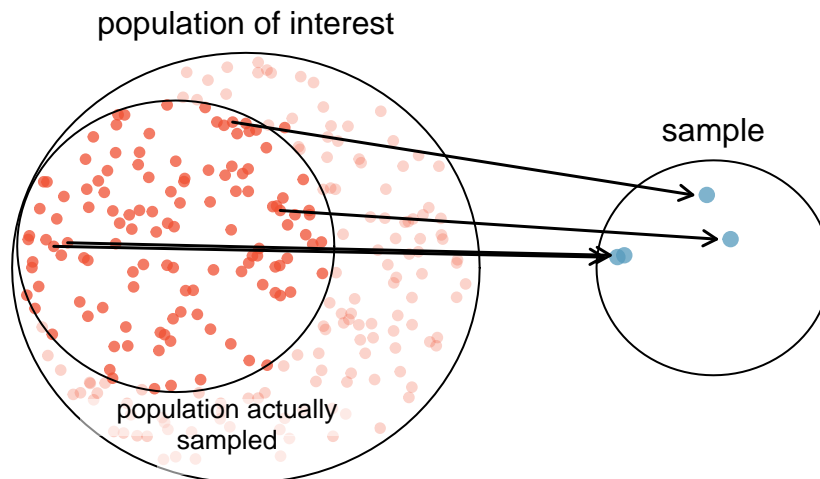


Figure 3.4: Due to the possibility of non-response, surveys studies may only reach a certain group within the population. It is difficult, and often impossible, to completely fix this problem

Another common pitfall is a **convenience sample**, where individuals who are
 or difficult to discern.

easily accessible are more likely to be included in the sample, see Figure ?? . For instance, if a political survey is done by stopping people walking in the Bronx, it will not represent all of New York City. It is often difficult to discern what sub-population a convenience sample represents.

Exercise:

We can easily access ratings for products, sellers, and companies through websites. These ratings are based only on those people who go out of their way to provide a rating. If 50% of online reviews for a product are negative, do you think this means that 50% of buyers are dissatisfied with the product?³

3.2.4 Explanatory and response variables

Consider the following question for the `county` data set:

Is federal spending, on average, higher or lower in counties with high rates of poverty?

If we suspect poverty might affect spending in a county, then poverty is the **explanatory** variable and federal spending is the **response** variable in the relationship.⁴ If there are many variables, it may be possible to consider a number of them as explanatory variables.

Explanatory and response variables

To identify the explanatory variable in a pair of variables, identify which of the two is suspected of affecting the other.

Caution: Association does not imply causation. Labeling variables as *explanatory* and *response* does not guarantee the relationship between the two is actually causal, even if there is an association identified between the two variables. We use these labels only to keep track of which variable we suspect affects the other. We also use this language to help in our use of `R` and the formula notation.

In some cases, there is no explanatory or response variable. Consider the following question:

If homeownership in a particular county is lower than the national average, will the percent of multi-unit structures in that county likely be above or below the national average?

³Answers will vary. From our own anecdotal experiences, we believe people tend to rant more about products that fell below expectations than rave about those that perform as expected. For this reason, we suspect there is a negative bias in product ratings on sites like Amazon. However, since our experiences may not be representative, we also keep an open mind.

⁴Sometimes the explanatory variable is called the **independent** variable and the response variable is called the **dependent** variable. However, this becomes confusing since a *pair* of variables might be independent or dependent, so be careful and consider the context when using or reading these words.

It is difficult to decide which of these variables should be considered the explanatory and response variable; i.e. the direction is ambiguous, so no explanatory or response labels are suggested here.

3.2.5 Introducing observational studies and experiments

There are two primary types of data collection: observational studies and experiments.

Researchers perform an **observational study** when they collect data in a way that does not directly interfere with how the data arise. For instance, researchers may collect information via surveys, review medical or company records, or follow a **cohort** of many similar individuals to study why certain diseases might develop. In each of these situations, researchers merely observe what happens. In general, observational studies can provide evidence of a naturally occurring association between variables, but by themselves, they cannot show a causal connection.

When researchers want to investigate the possibility of a causal connection, they conduct an **experiment**. Usually there will be both an explanatory and a response variable. For instance, we may suspect administering a drug will reduce mortality in heart attack patients over the following year. To check if there really is a causal connection between the explanatory variable and the response, researchers will collect a sample of individuals and split them into groups. The individuals in each group are *assigned* a treatment. When individuals are randomly assigned to a treatment group, the experiment is called a **randomized experiment**. For example, each heart attack patient in the drug trial could be randomly assigned, perhaps by flipping a coin, into one of two groups: the first group receives a **placebo** (fake treatment) and the second group receives the drug. The case study at the beginning of the semester is another example of an experiment, though that study did not employ a placebo. Math 359 is a course on the design and analysis of experimental data, DOE. In the Air Force these types of experiments are an important part of test and evaluation. Many Air Force analysts are expert practitioners of DOE. In this course we will minimize our discussion of DOE.

Association \neq Causation

Again, association does not imply causation. In a data analysis, association does not imply causation, and causation can only be inferred from a randomized experiment. Although, a hot field is the analysis of causal relationships in observational data. This is important because consider cigarette smoking, how do we know it causes lung cancer? We only have observational data and clearly cannot do an experiment. We think analysts will be charged in the near future with using causal reasoning on observational data.

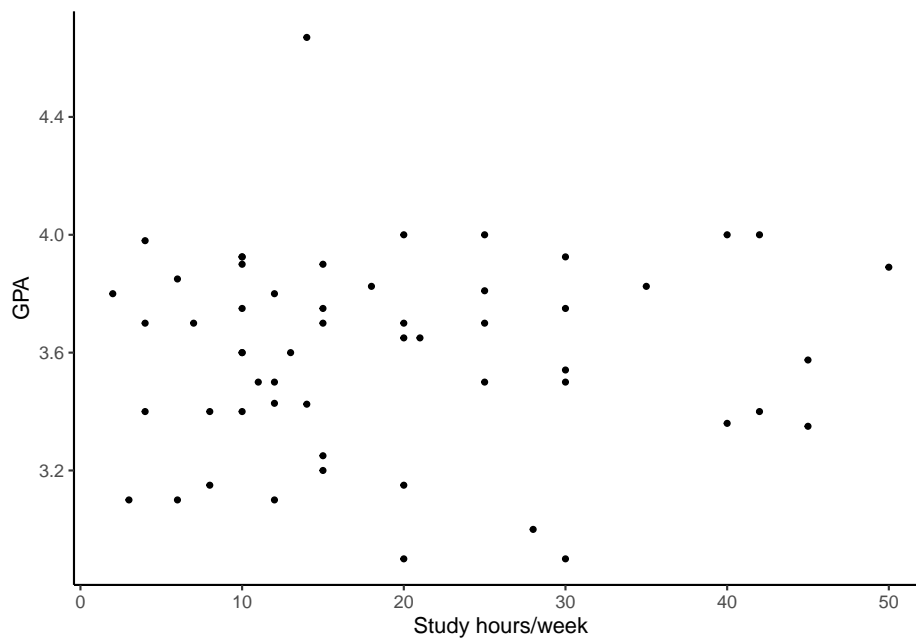
3.3 Homework Problems

1. **Generalizability and causality.** Identify the population of interest and the sample in the studies described below. These are the same studies from the previous lesson. Also comment on whether or not the results of the study can be generalized to the population and if the findings of the study can be used to establish causal relationships.
 - a. Researchers collected data to examine the relationship between pollutants and preterm births in Southern California. During the study air pollution levels were measured by air quality monitoring stations. Specifically, levels of carbon monoxide were recorded in parts per million, nitrogen dioxide and ozone in parts per hundred million, and coarse particulate matter (PM₁₀) in $\mu\text{g}/\text{m}^3$. Length of gestation data were collected on 143,196 births between the years 1989 and 1993, and air pollution exposure during gestation was calculated for each birth. The analysis suggested that increased ambient PM₁₀ and, to a lesser degree, CO concentrations may be associated with the occurrence of preterm births.⁵
 - b. The Buteyko method is a shallow breathing technique developed by Konstantin Buteyko, a Russian doctor, in 1952. Anecdotal evidence suggests that the Buteyko method can reduce asthma symptoms and improve quality of life. In a scientific study to determine the effectiveness of this method, researchers recruited 600 asthma patients aged 18-69 who relied on medication for asthma treatment. These patients were split into two research groups: one practiced the Buteyko method and the other did not. Patients were scored on quality of life, activity, asthma symptoms, and medication reduction on a scale from 0 to 10. On average, the participants in the Buteyko group experienced a significant reduction in asthma symptoms and an improvement in quality of life.⁶

⁵B. Ritz et al. "Effect of air pollution on preterm birth among children born in Southern California between 1989 and 1993". In: *Epidemiology* 11.5 (2000), pp. 502-511.

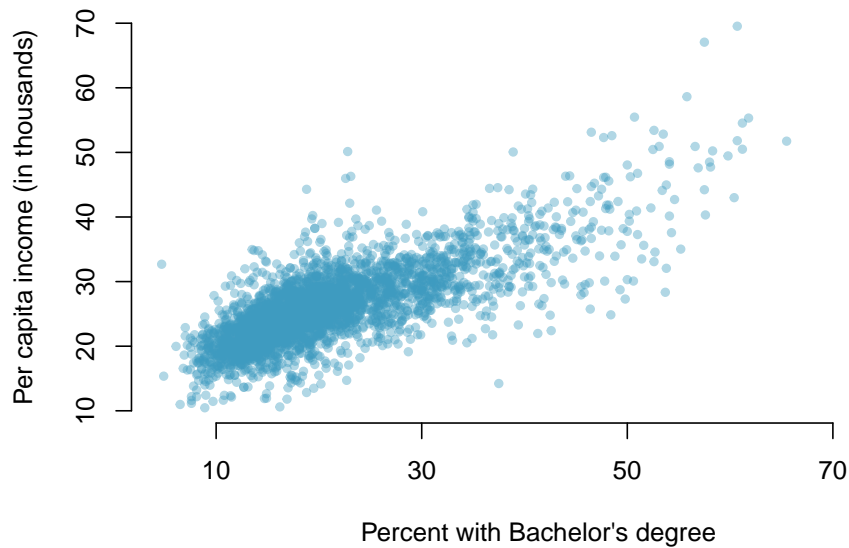
⁶J. McGowan. "Health Education: Does the Buteyko Institute Method make a difference?" In: *Thorax* 58 (2003).

2. **GPA and study time.** A survey was conducted on 55 undergraduates from Duke University who took an introductory statistics course in Spring 2012. Among many other questions, this survey asked them about their GPA and the number of hours they spent studying per week. The scatterplot below displays the relationship between these two variables.



- What is the explanatory variable and what is the response variable?
- Describe the relationship between the two variables. Make sure to discuss unusual observations, if any.
- Is this an experiment or an observational study?
- Can we conclude that studying longer hours leads to higher GPAs?

3. **Income and education** The scatterplot below shows the relationship between per capita income (in thousands of dollars) and percent of population with a bachelor's degree in 3,143 counties in the US in 2010.



- What are the explanatory and response variables?
- Describe the relationship between the two variables. Make sure to discuss unusual observations, if any.
- Can we conclude that having a bachelor's degree increases one's income?

Chapter 4

Studies

4.1 Objectives

- 1) Define and use properly in context all new terminology.
- 2) Given a study description, be able to identify and explain the study using correct terms.
- 3) Given a scenario, describe flaws in reasoning and propose study and sampling designs.

4.2 Observation studies, sampling strategies, and experiments

4.2.1 Observational studies

Generally, data in observational studies are collected only by monitoring what occurs, while experiments require the primary explanatory variable in a study be assigned for each subject by the researchers.

Making causal conclusions based on experiments is often reasonable. However, making the same causal conclusions based on observational data can be treacherous and is not recommended. Thus, observational studies are generally only sufficient to show associations.

Exercise:

Suppose an observational study tracked sunscreen use and skin cancer, and it was found that the more sunscreen someone used, the more likely the person was to have skin cancer. Does this mean

sunscreen *causes* skin cancer?¹

Some previous research² tells us that using sunscreen actually reduces skin cancer risk, so maybe there is another variable that can explain this hypothetical association between sunscreen usage and skin cancer. One important piece of information that is absent is sun exposure. If someone is out in the sun all day, she is more likely to use sunscreen *and* more likely to get skin cancer. Exposure to the sun is unaccounted for in the simple investigation.

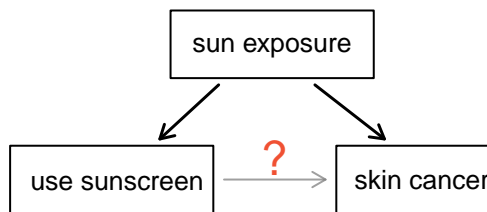


Figure 4.1: Sun exposure is a confounding variable because it is related to both response and explanatory variables.

Sun exposure is what is called a **confounding variable**,³ which is a variable that is correlated with both the explanatory and response variables, see Figure ?? . While one method to justify making causal conclusions from observational studies is to exhaust the search for confounding variables, there is no guarantee that all confounding variables can be examined or measured.

Let's look at an example of confounding visually. Using the **SAT** data from the **mosaic** package let's look at expenditure per pupil versus SAT scores. Figure ?? is a plot of the data.

Exercise:

What conclusion to you reach from the plot in Figure ??⁴

The implication that spending less might give better results is not justified. Expenditures are confounded with the proportion of students who take the exam, and scores are higher in states where fewer students take the exam.

It is interesting to look at the original plot if we place the states into two groups depending on whether more or fewer than 40% of students take the SAT. Figure ?? is a plot of the data broken down into the 2 groups.

Once we account for the fraction of students taking the SAT, the relationship between expenditures and SAT scores changes.

¹No. See the paragraph following the exercise for an explanation.

²<http://www.sciencedirect.com/science/article/pii/S0140673698121682>
<http://archderm.ama-assn.org/cgi/content/abstract/122/5/537>
 Study with a similar scenario to that described here:

<http://onlinelibrary.wiley.com/doi/10.1002/ijc.22745/full>

³Also called a **lurking variable**, **confounding factor**, or a **confounder**.

⁴It appears that average SAT score declines as expenditures per student increases.

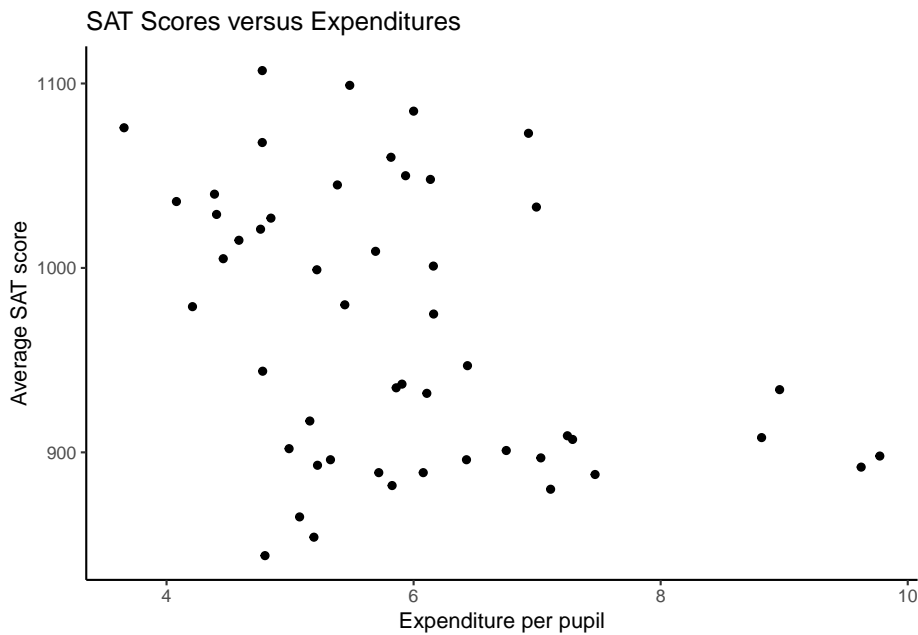


Figure 4.2: Average SAT score versus expenditure per pupil; reminder: each observation represents an individual state.

In the same way, the county data set is an observational study with confounding variables, and its data cannot easily be used to make causal conclusions.

Exercise:

Figure ?? shows a negative association between the homeownership rate and the percentage of multi-unit structures in a county. However, it is unreasonable to conclude that there is a causal relationship between the two variables. Suggest one or more other variables that might explain the relationship in the Figure ??.⁵

Observational studies come in two forms: prospective and retrospective studies. A **prospective study** identifies individuals and collects information as events unfold. For instance, medical researchers may identify and follow a group of similar individuals over many years to assess the possible influences of behavior on cancer risk. One example of such a study is The Nurses Health Study, started in 1976 and expanded in 1989.⁶ This prospective study recruits registered nurses and then collects data from them using questionnaires.

⁵Answers will vary. Population density may be important. If a county is very dense, then a larger fraction of residents may live in multi-unit structures. Additionally, the high density may contribute to increases in property value, making homeownership infeasible for many residents.

⁶<http://www.channing.harvard.edu/nhs/>

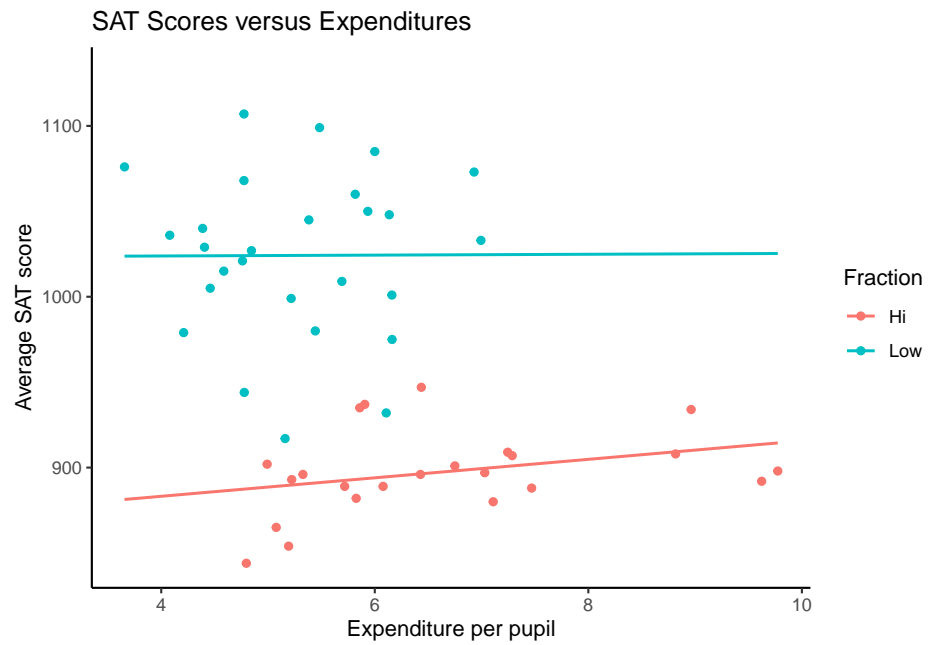


Figure 4.3: Average SAT score versus expenditure per pupil; broken down by level of participation.

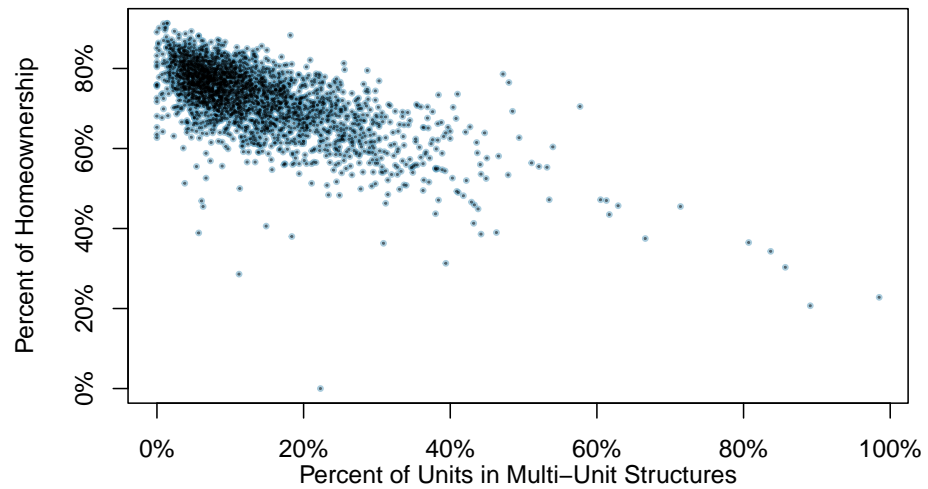


Figure 4.4: A scatterplot of the homeownership rate versus the percent of units that are in multi-unit structures for all 3,143 counties.

Retrospective studies collect data after events have taken place; e.g. researchers may review past events in medical records. Some data sets, such as `county`, may contain both prospectively- and retrospectively-collected variables. Local governments prospectively collect some variables as events unfolded (e.g. retail sales) while the federal government retrospectively collected others during the 2010 census (e.g. county population).

4.2.2 Three sampling methods

Almost all statistical methods are based on the notion of implied randomness. If observational data are not collected in a random framework from a population, results from these statistical methods are not reliable. Here we consider three random sampling techniques: simple, stratified, and cluster sampling. Figures ?? , ?? , and ?? provides a graphical representation of these techniques.

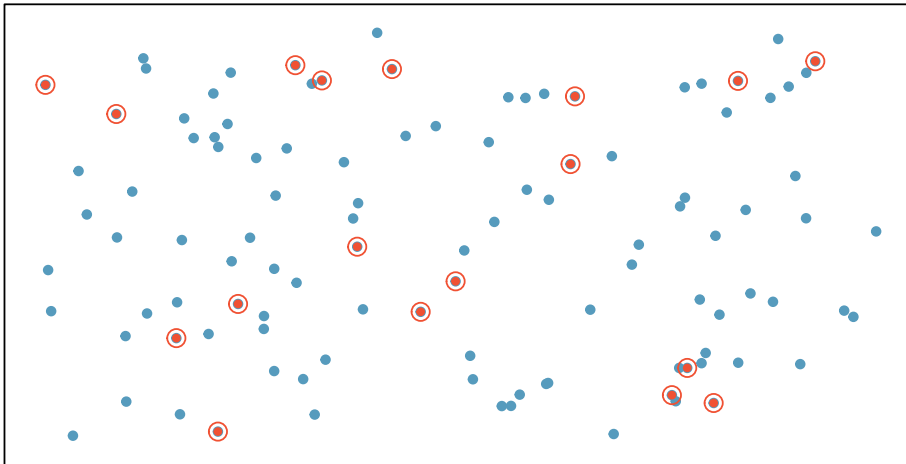


Figure 4.5: Examples of simple random sampling. In this figure, simple random sampling was used to randomly select the 18 cases.

Simple random sampling is probably the most intuitive form of random sampling. Consider the salaries of Major League Baseball (MLB) players, where each player is a member of one of the league’s 30 teams. To take a simple random sample of 120 baseball players and their salaries from the 2010 season, we could write the names of that season’s 828 players onto slips of paper, drop the slips into a bucket, shake the bucket around until we are sure the names are all mixed up, then draw out slips until we have the sample of 120 players. In general, a sample is referred to as “simple random” if each case in the population has an equal chance of being included in the final sample *and* knowing that a case is included in a sample does not provide useful information about which other cases are included.

Stratified sampling is a divide-and-conquer sampling strategy. The popula-

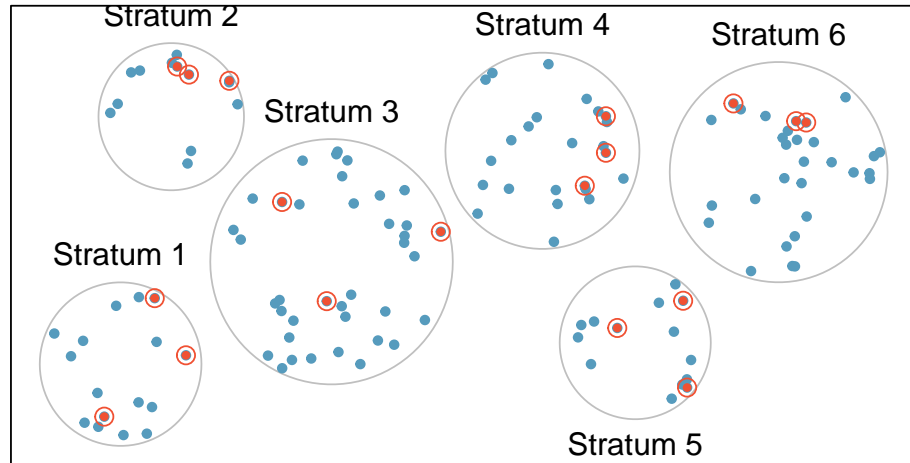


Figure 4.6: In this figure, stratified sampling was used: cases were grouped into strata, and then simple random sampling was employed within each stratum.

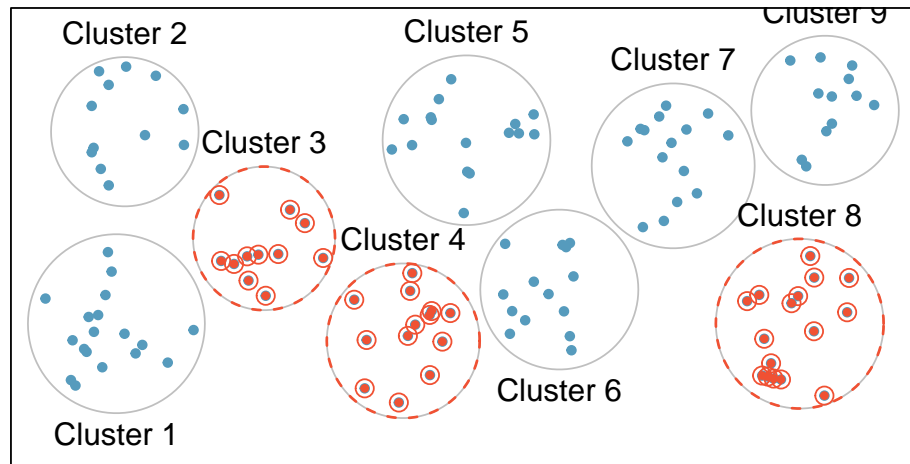


Figure 4.7: In this figure, cluster sampling was used, where data were binned into nine clusters, and three of the clusters were randomly selected.

tion is divided into groups called **strata**. The strata are chosen so that similar cases are grouped together, then a second sampling method, usually simple random sampling, is employed within each stratum. In the baseball salary example, the teams could represent the strata; some teams have a lot more money (we're looking at you, Yankees). Then we might randomly sample 4 players from each team for a total of 120 players.

Stratified sampling is especially useful when the cases in each stratum are very similar with respect to the outcome of interest. The downside is that analyzing data from a stratified sample is a more complex task than analyzing data from a simple random sample. The analysis methods introduced in this course would need to be extended to analyze data collected using stratified sampling.

Example:

Why would it be good for cases within each stratum to be very similar?⁷

In **cluster sampling**, we group observations into clusters, then randomly sample some of the clusters. Sometimes cluster sampling can be a more economical technique than the alternatives. Also, unlike stratified sampling, cluster sampling is most helpful when there is a lot of case-to-case variability within a cluster but the clusters themselves don't look very different from one another. For example, if neighborhoods represented clusters, then this sampling method works best when the neighborhoods are very diverse. A downside of cluster sampling is that more advanced analysis techniques are typically required, though the methods in this course can be extended to handle such data.

Example:

Suppose we are interested in estimating the malaria rate in a densely tropical portion of rural Indonesia. We learn that there are 30 villages in that part of the Indonesian jungle, each more or less similar to the next. What sampling method should be employed?⁸

Another technique called **multistage sampling** is similar to cluster sampling, except that we take a simple random sample within each selected cluster. For instance, if we sampled neighborhoods using cluster sampling, we would next sample a subset of homes within each selected neighborhood if we were using multistage sampling.

⁷We might get a more stable estimate for the subpopulation in a stratum if the cases are very similar. These improved estimates for each subpopulation will help us build a reliable estimate for the full population.

⁸A simple random sample would likely draw individuals from all 30 villages, which could make data collection extremely expensive. Stratified sampling would be a challenge since it is unclear how we would build strata of similar individuals. However, cluster sampling seems like a very good idea. We might randomly select a small number of villages. This would probably reduce our data collection costs substantially in comparison to a simple random sample and would still give us helpful information.

4.2.3 Experiments

Studies where the researchers assign treatments to cases are called **experiments**. When this assignment includes randomization, e.g. using a coin flip to decide which treatment a patient receives, it is called a **randomized experiment**. Randomized experiments are fundamentally important when trying to show a causal connection between two variables.

4.2.3.1 Principles of experimental design

Randomized experiments are generally built on four principles.

1. **Controlling.** Researchers assign treatments to cases, and they do their best to **control** any other differences in the groups. For example, when patients take a drug in pill form, some patients take the pill with only a sip of water while others may have it with an entire glass of water. To control for the effect of water consumption, a doctor may ask all patients to drink a 12 ounce glass of water with the pill.
2. **Randomization.** Researchers randomize patients into treatment groups to account for variables that cannot be controlled. For example, some patients may be more susceptible to a disease than others due to their dietary habits. Randomizing patients into the treatment or control group helps even out such differences, and it also prevents accidental bias from entering the study.
3. **Replication.** The more cases researchers observe, the more accurately they can estimate the effect of the explanatory variable on the response. In a single study, we **replicate** by collecting a sufficiently large sample. Additionally, a group of scientists may replicate an entire study to verify an earlier finding. You replicate to the level of variability you want to estimate. For example, in flight test, we can run the same flight conditions again to get a replicate; however, if the same plane and pilot are being used, the replicate is not getting the pilot-to-pilot or the plane-to-plane variability.
4. **Blocking.** Researchers sometimes know or suspect that variables, other than the treatment, influence the response. Under these circumstances, they may first group individuals based on this variable and then randomize cases within each block to the treatment groups. This strategy is often referred to as **blocking**. For instance, if we are looking at the effect of a drug on heart attacks, we might first split patients into low-risk and high-risk **blocks**, then randomly assign half the patients from each block to the control group and the other half to the treatment group, as shown in Figure ???. This strategy ensures each treatment group has an equal number of low-risk and high-risk patients.

It is important to incorporate the first three experimental design principles into any study, and this course describes methods for analyzing data from such

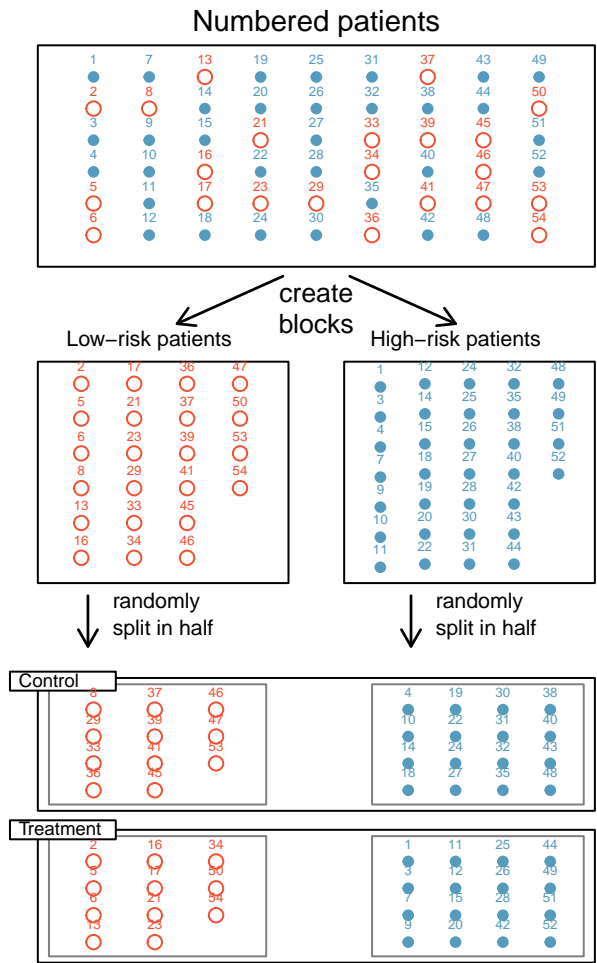


Figure 4.8: Blocking using a variable depicting patient risk. Patients are first divided into low-risk and high-risk blocks, then each block is evenly divided into the treatment groups using randomization. This strategy ensures an equal representation of patients in each treatment group from both the low-risk and high-risk categories.

experiments. Blocking is a slightly more advanced technique, and statistical methods in this course may be extended to analyze data collected using blocking. Math 359 is an entire course devoted to the design and analysis of experiments.

4.2.3.2 Reducing bias in human experiments

Randomized experiments are the gold standard for data collection, but they do not ensure an unbiased perspective into the cause and effect relationships in all cases. Human studies are perfect examples where bias can unintentionally arise. Here we reconsider a study where a new drug was used to treat heart attack patients.⁹ In particular, researchers wanted to know if the drug reduced deaths in patients.

These researchers designed a randomized experiment because they wanted to draw causal conclusions about the drug's effect. Study volunteers¹⁰ were randomly placed into two study groups. One group, the **treatment group**, received the drug. The other group, called the **control group**, did not receive any drug treatment.

Put yourself in the place of a person in the study. If you are in the treatment group, you are given a fancy new drug that you anticipate will help you. On the other hand, a person in the other group doesn't receive the drug and sits idly, hoping her participation doesn't increase her risk of death. These perspectives suggest there are actually two effects: the one of interest is the effectiveness of the drug, and the second is an emotional effect that is difficult to quantify.

Researchers aren't usually interested in the emotional effect, which might bias the study. To circumvent this problem, researchers do not want patients to know which group they are in. When researchers keep the patients uninformed about their treatment, the study is said to be **blind**. But there is one problem: if a patient doesn't receive a treatment, she will know she is in the control group. The solution to this problem is to give fake treatments to patients in the control group. A fake treatment is called a **placebo**, and an effective placebo is the key to making a study truly blind. A classic example of a placebo is a sugar pill that is made to look like the actual treatment pill. Often times, a placebo results in a slight but real improvement in patients. This effect has been dubbed the **placebo effect**.

The patients are not the only ones who should be blinded: doctors and researchers can accidentally bias a study. When a doctor knows a patient has been given the real treatment, she might inadvertently give that patient more attention or care than a patient that she knows is on the placebo. To guard against this bias, which again has been found to have a measurable effect in some instances, most modern studies employ a **double-blind** setup where doc-

⁹Anturane Reinfarction Trial Research Group. 1980. Sulfipyrazone in the prevention of sudden death after myocardial infarction. *New England Journal of Medicine* 302(5):250-256.

¹⁰Human subjects are often called **patients**, **volunteers**, or **study participants**.

tors or researchers who interact with patients are, just like the patients, unaware of who is or is not receiving the treatment.¹¹

Exercise:

Look back to the stent study in the first lesson where researchers were testing whether stents were effective at reducing strokes in at-risk patients. Is this an experiment? Was the study blinded? Was it double-blinded?¹²

4.3 Homework Problems

1. **Propose a sampling strategy.** A large college class has 160 students. All 160 students attend the lectures together, but the students are divided into 4 groups, each of 40 students, for lab sections administered by different teaching assistants. The professor wants to conduct a survey about how satisfied the students are with the course, and he believes that the lab section a student is in might affect the student's overall satisfaction with the course.
 - a. What type of study is this?
 - b. Suggest a sampling strategy for carrying out this study.
2. **Flawed reasoning.** Identify the flaw in reasoning in the following scenarios. Explain what the individuals in the study should have done differently if they wanted to make such strong conclusions.
 - a. Students at an elementary school are given a questionnaire that they are required to return after their parents have completed it. One of the questions asked is, *Do you find that your work schedule makes it difficult for you to spend time with your kids after school?* Of the parents who replied, 85% said *no*. Based on these results, the school officials conclude that a great majority of the parents have no difficulty spending time with their kids after school.
 - b. A survey is conducted on a simple random sample of 1,000 women who recently gave birth, asking them about whether or not they smoked during pregnancy. A follow-up survey asking if the children have respiratory problems is conducted 3 years later, however, only 567 of these women are reached at the same address. The researcher reports that these 567 women are representative of all mothers.

¹¹There are always some researchers in the study who do know which patients are receiving which treatment. However, they do not interact with the study's patients and do not tell the blinded health care professionals who is receiving which treatment.

¹²The researchers assigned the patients into their treatment groups, so this study was an experiment. However, the patients could distinguish what treatment they received, so this study was not blind. The study could not be double-blind since it was not blind.

3. **Sampling strategies.** A Math 377 student who is curious about the relationship between the amount of time students spend on social networking sites and their performance at school decides to conduct a survey. Four research strategies for collecting data are described below. In each, name the sampling method proposed and any bias you might expect.

- a. He randomly samples 40 students from the study's population, gives them the survey, asks them to fill it out and bring it back the next day.
- b. He gives out the survey only to his friends, and makes sure each one of them fills out the survey.
- c. He posts a link to an online survey on his Facebook wall and asks his friends to fill out the survey.
- d. He stands outside the QRC and asks every third person that walks out the door to fill out the survey.

4. **Vitamin supplements.** In order to assess the effectiveness of taking large doses of vitamin C in reducing the duration of the common cold, researchers recruited 400 healthy volunteers from staff and students at a university. A quarter of the patients were assigned a placebo, and the rest were evenly divided between 1g Vitamin C, 3g Vitamin C, or 3g Vitamin C plus additives to be taken at onset of a cold for the following two days. All tablets had identical appearance and packaging. The nurses who handed the prescribed pills to the patients knew which patient received which treatment, but the researchers assessing the patients when they were sick did not. No significant differences were observed in any measure of cold duration or severity between the four medication groups, and the placebo group had the shortest duration of symptoms.
- Was this an experiment or an observational study? Why?
 - What are the explanatory and response variables in this study?
 - Were the patients blinded to their treatment?
 - Was this study double-blind?
 - Participants are ultimately able to choose whether or not to use the pills prescribed to them. We might expect that not all of them will adhere and take their pills. Does this introduce a confounding variable to the study? Explain your reasoning.
5. **Exercise and mental health.** A researcher is interested in the effects of exercise on mental health and she proposes the following study: Use stratified random sampling to ensure representative proportions of 18-30, 31-40 and 41-55 year olds from the population. Next, randomly assign half the subjects from each age group to exercise twice a week, and instruct the rest not to exercise. Conduct a mental health exam at the beginning and at the end of the study, and compare the results.
- What type of study is this?
 - What are the treatment and control groups in this study?
 - Does this study make use of blocking? If so, what is the blocking variable?
 - Does this study make use of blinding?
 - Comment on whether or not the results of the study can be used to establish a causal relationship between exercise and mental health, and indicate whether or not the conclusions can be generalized to the population at large.

- f. Suppose you are given the task of determining if this proposed study should get funding. Would you have any reservations about the study proposal?

Chapter 5

Numerical Data

5.1 Objectives

- 1) Define and use properly in context all new terminology.
- 2) Generate in R summary statistics for a numeric variable including breaking down by cases.
- 3) Generate in R appropriate graphical summaries of numerical variables.
- 4) Be able to interpret and explain output both graphically and numerically.

5.2 Numerical Data

This lesson introduces techniques for exploring and summarizing numerical variables, and the `email50` and `mlb` data sets from the **openintro** package and a subset of `county_complete` from `usdata` provide rich opportunities for examples. Recall that outcomes of numerical variables are numbers on which it is reasonable to perform basic arithmetic operations. For example, the `pop2010` variable, which represents the populations of counties in 2010, is numerical since we can sensibly discuss the difference or ratio of the populations in two counties. On the other hand, area codes and zip codes are not numerical.

5.2.1 Scatterplots for paired data

A **scatterplot** provides a case-by-case view of data for two numerical variables. In Figure ??, we again present a scatterplot used to examine how federal spending and poverty were related in the `county` data set.

Another scatterplot is shown in Figure ??, comparing the number of line breaks

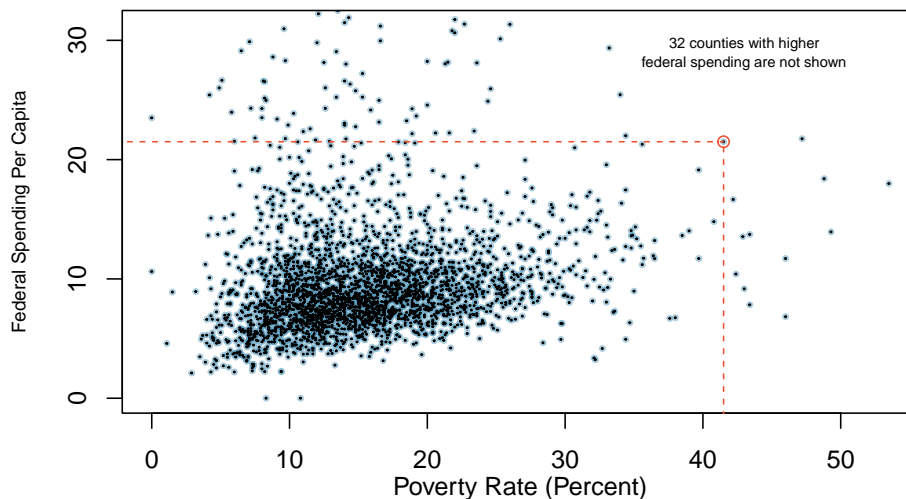


Figure 5.1: A scatterplot showing `fed_spend` against poverty. Owsley County of Kentucky, with a poverty rate of 41.5% and federal spending of \$21.50 per capita, is highlighted.

`line_breaks` and number of characters `num_char` in emails for the `email50` data set. In any scatterplot, each point represents a single case. Since there are 50 cases in `email50`, there are 50 points in Figure ??.

To put the number of characters in perspective, this paragraph has 357 characters. Looking at Figure ??, it seems that some emails are incredibly long! Upon further investigation, we would actually find that most of the long emails use the HTML format, which means most of the characters in those emails are used to format the email rather than provide text.

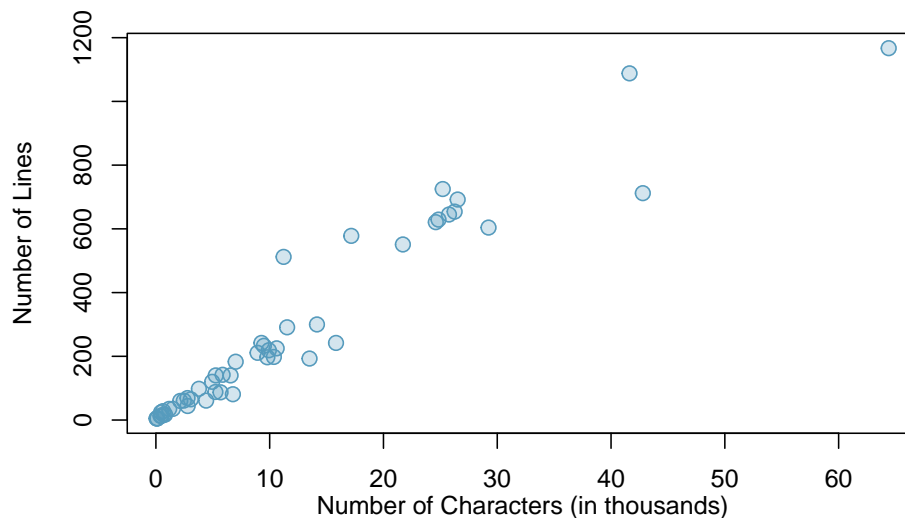


Figure 5.2: A scatterplot of `line_breaks` versus `num_char` for the `email50` data.

Exercise:

What do scatterplots reveal about the data, and how might they be useful?¹

Example:

Consider a new data set of 54 cars with two variables: vehicle price and weight.² A scatterplot of vehicle price versus weight is shown in Figure ???. What can be said about the relationship between these variables?

The relationship is evidently nonlinear, as highlighted by the dashed line. This is different from previous scatterplots we’ve seen which show relationships that are very linear.

Exercise:

Describe two variables that would have a horseshoe shaped association in a scatterplot.³

¹Answers may vary. Scatterplots are helpful in quickly spotting associations between variables, whether those associations represent simple or more complex relationships.

²Subset of data from <http://www.amstat.org/publications/jse/v1n1/datasets.lock.html>

³Consider the case where your vertical axis represents something “good” and your horizontal axis represents something that is only good in moderation. Health and water consumption fit this description since water becomes toxic when consumed in excessive quantities.

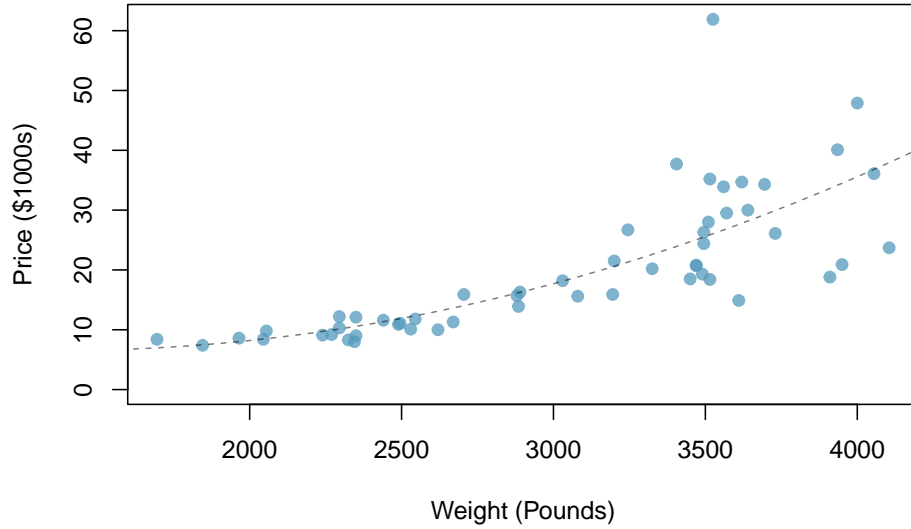


Figure 5.3: A scatterplot of *price* versus *weight* for 54 cars.

5.2.2 Dot plots and the mean

Sometimes two variables are one too many: only one variable may be of interest. In these cases, a dot plot provides the most basic of displays. A **dot plot** is a one-variable scatterplot; an example using the number of characters from 50 emails is shown in Figure ??.

The **mean**, sometimes called the average, is a common way to measure the center of a **distribution** of data. To find the mean number of characters in the 50 emails, we add up all the character counts and divide by the number of emails. For computational convenience, the number of characters is listed in the thousands and rounded to the first decimal.

$$\bar{x} = \frac{21.7 + 7.0 + \cdots + 15.8}{50} = 11.6$$

The sample mean is often labeled \bar{x} , and the letter x is being used as a generic placeholder for the variable of interest, `num_char`.

Mean

The sample mean of a numerical variable is the sum of all of the observations divided by the number of observations, Equation (??).

$$\bar{x} = \frac{x_1 + x_2 + \cdots + x_n}{n} \quad (5.1)$$

where x_1, x_2, \dots, x_n represent the n observed values.

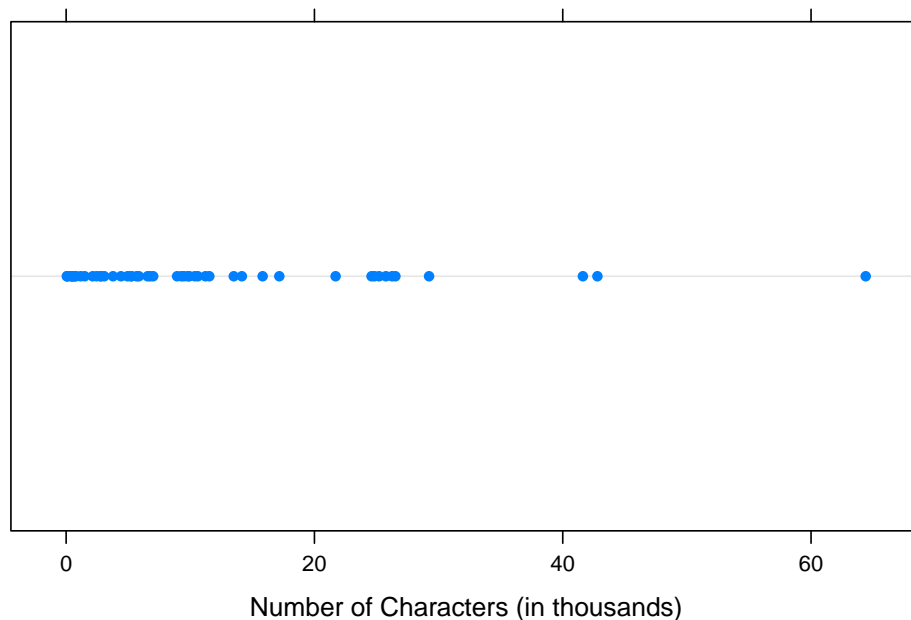


Figure 5.4: A dot plot of `num_char` for the `email150` data set.

Exercise:

Examine the two equations above. What does x_1 correspond to? And x_2 ? Can you infer a general meaning to what x_i might represent?⁴

Exercise:

What was n in this sample of emails?⁵

The `email150` data set is a sample from a larger population of emails that were received in January and March. We could compute a mean for this population in the same way as the sample mean. However, there is a difference in notation: the population mean has a special label: μ . The symbol μ is the Greek letter *mu* and represents the average of all observations in the population. Sometimes a subscript, such as x , is used to represent which variable the population mean refers to, e.g. μ_x .

Example: The average number of characters across all emails can be estimated using the sample data. Based on the sample of 50 emails, what would be a reasonable estimate of μ_x , the mean number of characters in all emails in the `email` data set? (Recall that `email150`

⁴ x_1 corresponds to the number of characters in the first email in the sample (21.7, in thousands), x_2 to the number of characters in the second email (7.0, in thousands), and x_i corresponds to the number of characters in the i^{th} email in the data set.

⁵The sample size was $n = 50$.

is a sample from `email`.)

The sample mean, 11.6, may provide a reasonable estimate of μ_x . While this number will not be perfect, it provides a **point estimate** of the population mean. Later in the semester, we will develop tools to characterize the accuracy of point estimates, and we will find that point estimates based on larger samples tend to be more accurate than those based on smaller samples.

Example:

We might like to compute the average income per person in the US. To do so, we might first think to take the mean of the per capita incomes from the 3,143 counties in the `county` data set. What would be a better approach?

The `county` data set is special in that each county actually represents many individual people. If we were to simply average across the `income` variable, we would be treating counties with 5,000 and 5,000,000 residents equally in the calculations. Instead, we should compute the total income for each county, add up all the counties' totals, and then divide by the number of people in all the counties. If we completed these steps with the `county` data, we would find that the per capita income for the US is \$27,348.43. Had we computed the *simple* mean of per capita income across counties, the result would have been just \$22,504.70!

This previous example used what is called a **weighted mean**, which will be a key topic in the probability section. As a look ahead, the probability mass function gives the population proportions of each value and thus to find the population mean μ , we will use a weighted mean.

5.2.3 Histograms and shape

Dot plots show the exact value of each observation. This is useful for small data sets, but they can become hard to read with larger samples. Rather than showing the value of each observation, think of the value as belonging to a *bin*. For example, in the `email50` data set, we create a table of counts for the number of cases with character counts between 0 and 5,000, then the number of cases between 5,000 and 10,000, and so on. Observations that fall on the boundary of a bin (e.g. 5,000) are allocated to the lower bin. This tabulation is shown below.

##										
##	(0,5]	(5,10]	(10,15]	(15,20]	(20,25]	(25,30]	(30,35]	(35,40]	(40,45]	(45,50]
##	19	12	6	2	3	5	0	0	2	0
##	(50,55]	(55,60]	(60,65]							
##	0	0	1							

These binned counts are plotted as bars in Figure ?? into what is called a **histogram**.

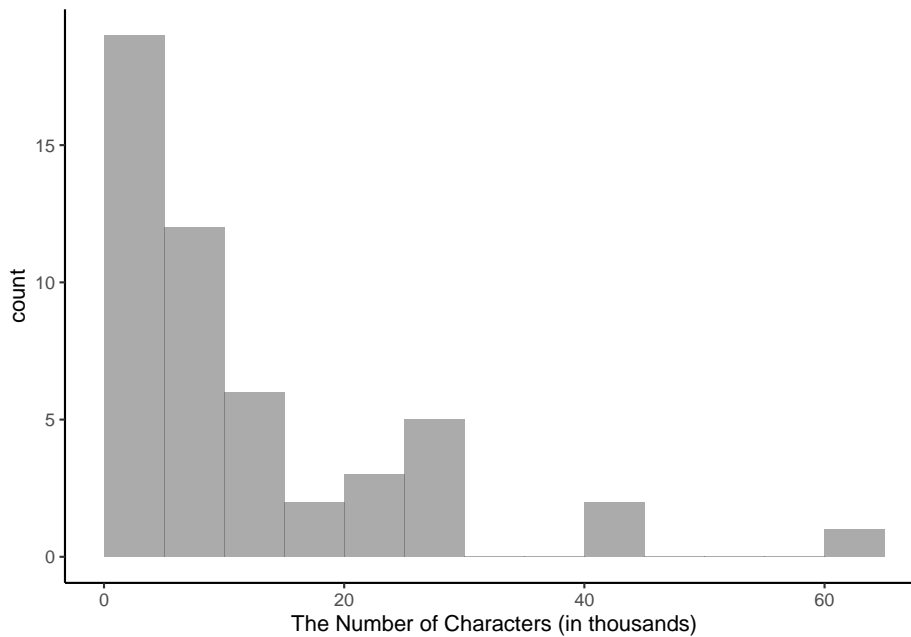


Figure 5.5: A histogram of `num_char`. This distribution is very strongly skewed to the right.

Histograms provide a view of the **data density**. Higher bars represent where the data are relatively more dense. For instance, there are many more emails between 0 and 10,000 characters than emails between 10,000 and 20,000 characters in the data set. The bars make it easy to see how the density of the data changes relative to the number of characters.

Histograms are especially convenient for describing the shape of the data distribution. Figure ?? shows that most emails have a relatively small number of characters, while fewer emails have a very large number of characters. When data trail off to the right in this way and have a longer right **tail**, the shape is said to be **right skewed**.⁶

Data sets with the reverse characteristic – a long, thin tail to the left – are said to be **left skewed**. We also say that such a distribution has a long left tail. Data sets that show roughly equal trailing off in both directions are called **symmetric**.

Long tails to identify skew

When data trail off in one direction, the distribution has a **long tail**.

If a distribution has a long left tail, it is left skewed. If a distribution

⁶Other ways to describe data that are skewed to the right: **skewed to the right**, **skewed to the high end**, or **skewed to the positive end**.

has a long right tail, it is right skewed.

Exercise:

Take a look at the dot plot above, Figure ?? . Can you see the skew in the data? Is it easier to see the skew in this histogram or the dot plots?⁷

Exercise:

Besides the mean, what can you see in the dot plot that you cannot see in the histogram?⁸

5.2.3.1 Making our own histogram

Let's take some time to make a simple histogram. We will use the **ggformula** package which is a wrapper for the **ggplot** package.

Here are two questions:

What do we want R to do? and

What must we give R for it to do this?

We want R to make a histogram. In **ggformula** the plots have the form **gf_XXXX** so we will use the **gf_histogram**. To find options and more information type:

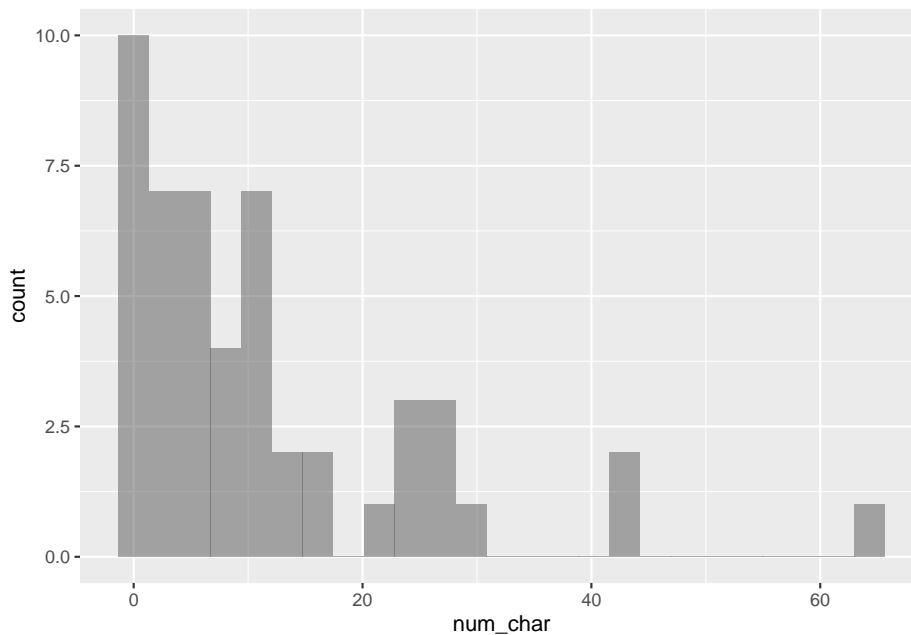
```
?gf_histogram
```

To start we just have to give the formulas and data to R.

```
gf_histogram(~num_char, data=email50)
```

⁷The skew is visible in all both plots, though the dot plot is the least useful.

⁸Character counts for individual emails.

**Exercise:**

Look at the help menu for `gf_histogram` and change the x-axis label, change the bin width to 5, and have the left bin start at 0.

Here is the code for the exercise

```
email50 %>%
  gf_histogram(~num_char, binwidth = 5, boundary=0,
    xlab="The Number of Characters (in thousands)") %>%
  gf_theme(theme_classic())
```

In addition to looking at whether a distribution is skewed or symmetric, histograms can be used to identify modes. A **mode** is represented by a prominent peak in the distribution.⁹ There is only one prominent peak in the histogram of `num_char`.

Figure ?? show histograms that have one, two, or three prominent peaks. Such distributions are called **unimodal**, **bimodal**, and **multimodal**, respectively. Any distribution with more than 2 prominent peaks is called multimodal. Notice that there was one prominent peak in the unimodal distribution with a second less prominent peak that was not counted since it only differs from its neighboring bins by a few observations.

Exercise:

Height measurements of young students and adult teachers at a K-3

⁹Another definition of mode, which is not typically used in statistics, is the value with the most occurrences. It is common to have *no* observations with the same value in a data set, which makes this other definition useless for many real data sets.

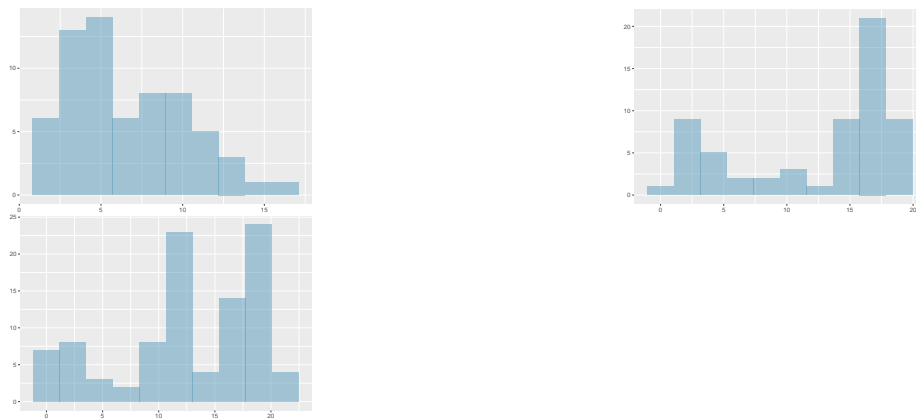


Figure 5.6: Histograms that demonstrate unimodal, bimodal, and multimodal data.

elementary school were taken. How many modes would you anticipate in this height data set?¹⁰

Looking for modes

Looking for modes isn't about finding a clear and correct answer about the number of modes in a distribution, which is why **prominent** is not rigorously defined in these notes. The important part of this examination is to better understand your data and how it might be structured.

5.2.4 Variance and standard deviation

The mean is used to describe the center of a data set, but the *variability* in the data is also important. Here, we introduce two measures of variability: the **variance** and the **standard deviation**. Both of these are very useful in data analysis, even though the formulas are a bit tedious to calculate by hand. The standard deviation is the easier of the two to conceptually understand, and it roughly describes how far away the typical observation is from the mean. Equation (??) is the equation for sample variance. We will demonstrate it with data so that the notation is easier to understand.

$$s^2 = \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{n-1} = \frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + \cdots + (x_n - \bar{x})^2}{n-1} \quad (5.2)$$

¹⁰There might be two height groups visible in the data set: one of the students and one of the adults. That is, the data are probably bimodal. But it could be multimodal because within each group we may be able to see a difference in males and females.

where x_1, x_2, \dots, x_n represent the n observed values.

We call the distance of an observation from its mean its **deviation**. Below are the deviations for the 1st, 2nd, 3rd, and 50th observations in the `num_char` variable. For computational convenience, the number of characters is listed in the thousands and rounded to the first decimal.

$$\begin{aligned}x_1 - \bar{x} &= 21.7 - 11.6 = 10.1 \\x_2 - \bar{x} &= 7.0 - 11.6 = -4.6 \\x_3 - \bar{x} &= 0.6 - 11.6 = -11.0 \\&\vdots \\x_{50} - \bar{x} &= 15.8 - 11.6 = 4.2\end{aligned}$$

If we square these deviations and then take an average, the result is equal to the **sample variance**, denoted by s^2 :

$$\begin{aligned}s^2 &= \frac{10.1^2 + (-4.6)^2 + (-11.0)^2 + \dots + 4.2^2}{50 - 1} \\&= \frac{102.01 + 21.16 + 121.00 + \dots + 17.64}{49} \\&= 172.44\end{aligned}$$

We divide by $n - 1$, rather than dividing by n , when computing the variance; you need not worry about this mathematical nuance yet. Notice that squaring the deviations does two things. First, it makes large values much larger, seen by comparing 10.1^2 , $(-4.6)^2$, $(-11.0)^2$, and 4.2^2 . Second, it gets rid of any negative signs.

The sample **standard deviation** s is the square root of the variance:

$$s = \sqrt{172.44} = 13.13$$

The sample standard deviation of the number of characters in an email is 13.13 thousand. A subscript of $_x$ may be added to the variance and standard deviation, i.e. s_x^2 and s_x , as a reminder that these are the variance and standard deviation of the observations represented by x_1, x_2, \dots, x_n . The $_x$ subscript is usually omitted when it is clear which data the variance or standard deviation is referencing.

Variance and standard deviation

The variance is roughly the average squared distance from the mean.

The standard deviation is the square root of the variance and describes how close the data are to the mean.

Formulas and methods used to compute the variance and standard deviation for a population are similar to those used for a sample.¹¹ However, like the mean,

¹¹The only difference is that the population variance has a division by n instead of $n - 1$.

the population values have special symbols: σ^2 for the variance and σ for the standard deviation. The symbol σ is the Greek letter *sigma*.

Tip: standard deviation describes variability

Focus on the conceptual meaning of the standard deviation as a descriptor of variability rather than the formulas. Usually 70% of the data will be within one standard deviation of the mean and about 95% will be within two standard deviations. However, as we have seen, these percentages are not strict rules.

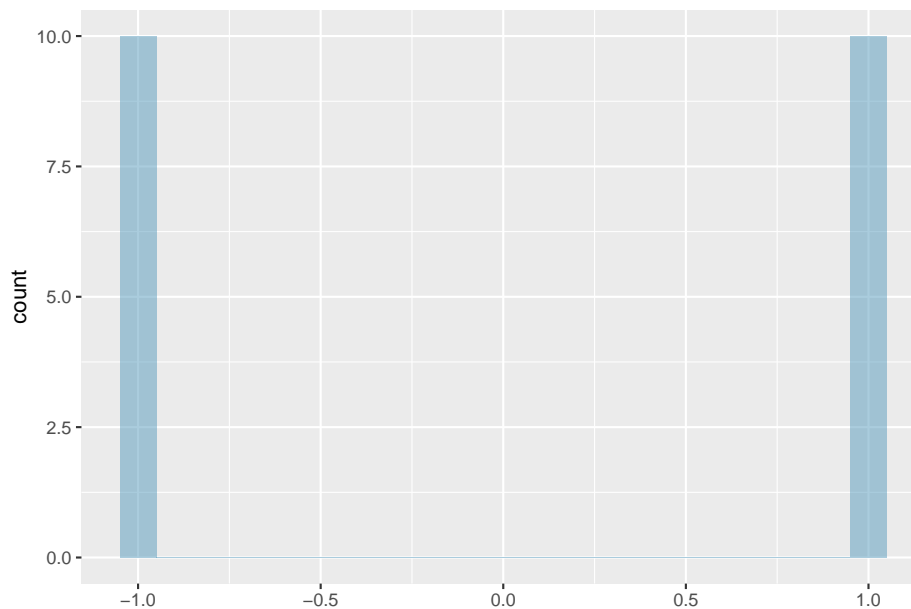


Figure 5.7: The first of three very different population distributions with the same mean, 0, and standard deviation, 1.

Exercise:

Earlier the concept of shape of a distribution was introduced. A good description of the shape of a distribution should include modality and whether the distribution is symmetric or skewed to one side. Using the three figures, Figures ??, ??, and ?? as an example, explain why such a description is important.¹²

Example:

Describe the distribution of the `num_char` variable using the his-

¹²Starting with Figure ??, the three figures show three distributions that look quite different, but all have the same mean, variance, and standard deviation. Using modality, we can distinguish between the first plot (bimodal) and the last two (unimodal). Using skewness, we can distinguish between the last plot (right skewed) and the first two. While a picture, like a histogram, tells a more complete story, we can use modality and shape (symmetry/skew) to characterize basic information about a distribution.

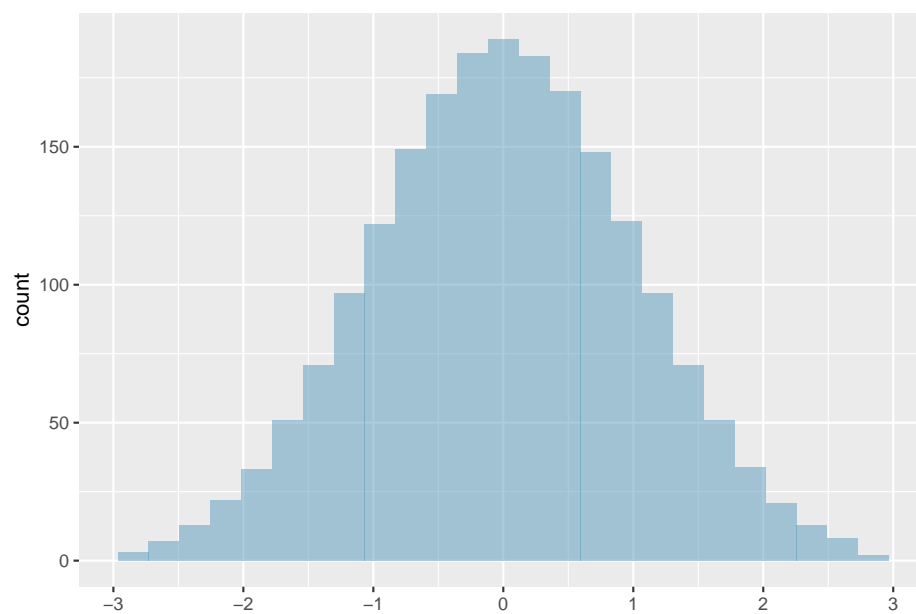


Figure 5.8: The second plot with mean 0 and standard deviation 1.

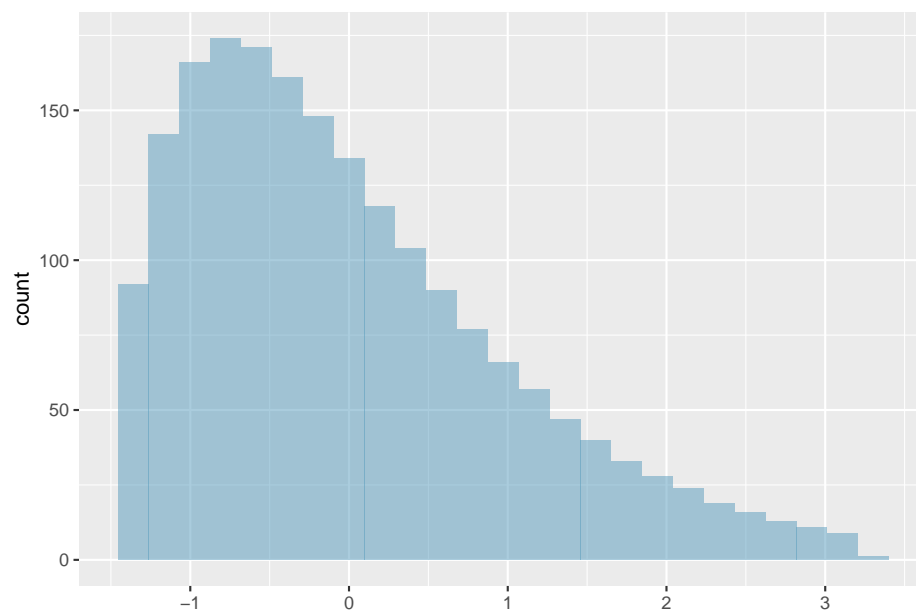


Figure 5.9: The final plot with mean 0 and standard deviation 1.

togram in Figure ?? . The description should incorporate the center, variability, and shape of the distribution, and it should also be placed in context: the number of characters in emails. Also note any especially unusual cases.¹³

In practice, the variance and standard deviation are sometimes used as a means to an end, where the *end* is being able to accurately estimate the uncertainty associated with a sample statistic. For example, later in the course we will use the variance and standard deviation to assess how close the sample mean is to the population mean.

5.2.5 Box plots, quartiles, and the median

A **box plot** summarizes a data set using five statistics while also plotting unusual observations. Figure ?? provides a vertical dot plot alongside a box plot of the `num_char` variable from the `email50` data set.

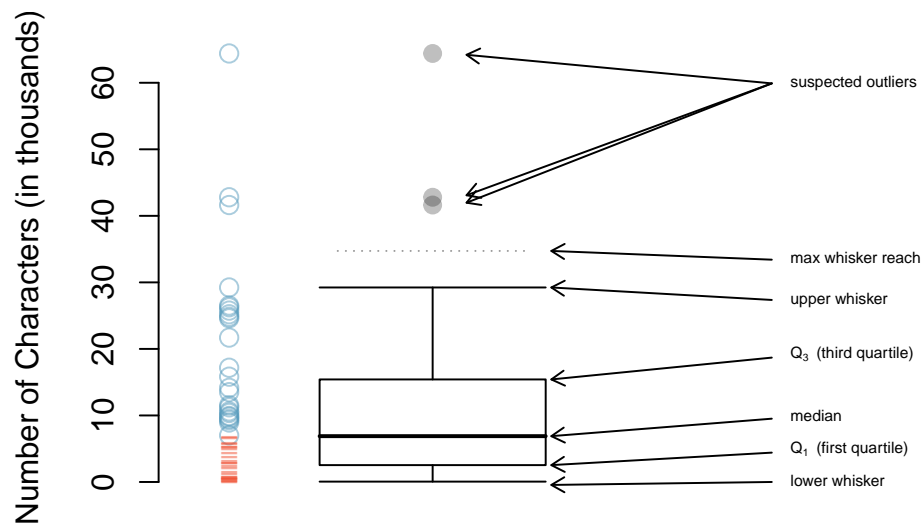


Figure 5.10: A vertical dot plot next to a labeled box plot for the number of characters in 50 emails. The median (6,890), splits the data into the bottom 50% and the top 50%, marked in the dot plot by horizontal dashes and open circles, respectively.

The first step in building a box plot is drawing a dark line denoting the **median**, which splits the data in half. Figure ?? shows 50% of the data falling below the median (red dashes) and the other 50% falling above the median (blue open

¹³The distribution of email character counts is unimodal and very strongly skewed to the high end. Many of the counts fall near the mean at 11,600, and most fall within one standard deviation (13,130) of the mean. There is one exceptionally long email with about 65,000 characters.

circles). There are 50 character counts in the data set (an even number) so the data are perfectly split into two groups of 25. We take the median in this case to be the average of the two observations closest to the 50th percentile: $(6,768 + 7,012)/2 = 6,890$. When there are an odd number of observations, there will be exactly one observation that splits the data into two halves, and in this case that observation is the median (no average needed).

Median: the number in the middle

If the data are ordered from smallest to largest, the **median** is the observation right in the middle. If there are an even number of observations, there will be two values in the middle, and the median is taken as their average.

The second step in building a box plot is drawing a rectangle to represent the middle 50% of the data. The total length of the box, shown vertically in Figure ??, is called the **interquartile range** (IQR, for short). It, like the standard deviation, is a measure of variability in data. The more variable the data, the larger the standard deviation and IQR. The two boundaries of the box are called the **first quartile** (the 25th percentile, i.e. 25% of the data fall below this value) and the **third quartile** (the 75th percentile), and these are often labeled Q_1 and Q_3 , respectively.

Interquartile range (IQR)

The IQR is the length of the box in a box plot. It is computed as

$$IQR = Q_3 - Q_1$$

where Q_1 and Q_3 are the 25th and 75th percentiles.

Exercise:

What percent of the data fall between Q_1 and the median? What percent is between the median and Q_3 ?¹⁴

Extending out from the box, the **whiskers** attempt to capture the data outside of the box, however, their reach is never allowed to be more than $1.5 \times IQR$.¹⁵ They capture everything within this reach. In Figure ??, the upper whisker does not extend to the last three points, which are beyond $Q_3 + 1.5 \times IQR$, and so it extends only to the last point below this limit. The lower whisker stops at the lowest value, 33, since there is no additional data to reach; the lower whisker's limit is not shown in the figure because the plot does not extend down to $Q_1 - 1.5 \times IQR$. In a sense, the box is like the body of the box plot and the whiskers are like its arms trying to reach the rest of the data.

Any observation that lies beyond the whiskers is labeled with a dot. The purpose of labeling these points – instead of just extending the whiskers to the minimum

¹⁴Since Q_1 and Q_3 capture the middle 50% of the data and the median splits the data in the middle, 25% of the data fall between Q_1 and the median, and another 25% falls between the median and Q_3 .

¹⁵While the choice of exactly 1.5 is arbitrary, it is the most commonly used value for box plots.

and maximum observed values – is to help identify any observations that appear to be unusually distant from the rest of the data. Unusually distant observations are called **outliers**. In this case, it would be reasonable to classify the emails with character counts of 41,623, 42,793, and 64,401 as outliers since they are numerically distant from most of the data.

Outliers are extreme

An **outlier** is an observation that is extreme relative to the rest of the data.

Why it is important to look for outliers

Examination of data for possible outliers serves many useful purposes, including

1. Identifying **strong skew** in the distribution.
2. Identifying data collection or entry errors. For instance, we re-examined the email purported to have 64,401 characters to ensure this value was accurate.
3. Providing insight into interesting properties of the data.

Exercise:

The observation with value 64,401, an outlier, was found to be an accurate observation. What would such an observation suggest about the nature of character counts in emails?¹⁶

Exercise:

Using Figure ??, estimate the following values for `num_char` in the `email50` data set:

- (a) Q_1 ,
- (b) Q_3 , and
- (c) IQR.¹⁷

Of course R can calculate these summary statistics for us. First we will do these calculations individually and then in one function call. Remember to ask what you want R to do and what it needs.

```
mean(~num_char,data=email50)
```

```
## [1] 11.59822
```

```
sd(~num_char,data=email50)
```

```
## [1] 13.12526
```

```
quantile(~num_char,data=email50)
```

```
##      0%      25%      50%      75%     100%
## 0.05700 2.53550 6.88950 15.41075 64.40100
```

¹⁶That occasionally there may be very long emails.

¹⁷These visual estimates will vary a little from one person to the next: $Q_1 \sim 3,000$, $Q_3 \sim 15,000$, $\text{IQR} = Q_3 - Q_1 \sim 12,000$. (The true values: $Q_1 = 2,536$, $Q_3 = 15,411$, $\text{IQR} = 12,875$.)

```
iqr(~num_char,data=email50)

## [1] 12.87525

favstats(~num_char,data=email50)

##      min      Q1 median      Q3      max      mean      sd  n missing
## 0.057 2.5355 6.8895 15.41075 64.401 11.59822 13.12526 50      0
```

5.2.6 Robust statistics

How are the *sample statistics* of the `num_char` data set affected by the observation with value 64,401? What would have happened if this email wasn't observed? What would happen to these *summary statistics* if the observation at 64,401 had been even larger, say 150,000? These scenarios are plotted alongside the original data in Figure ??, and sample statistics are computed in R.

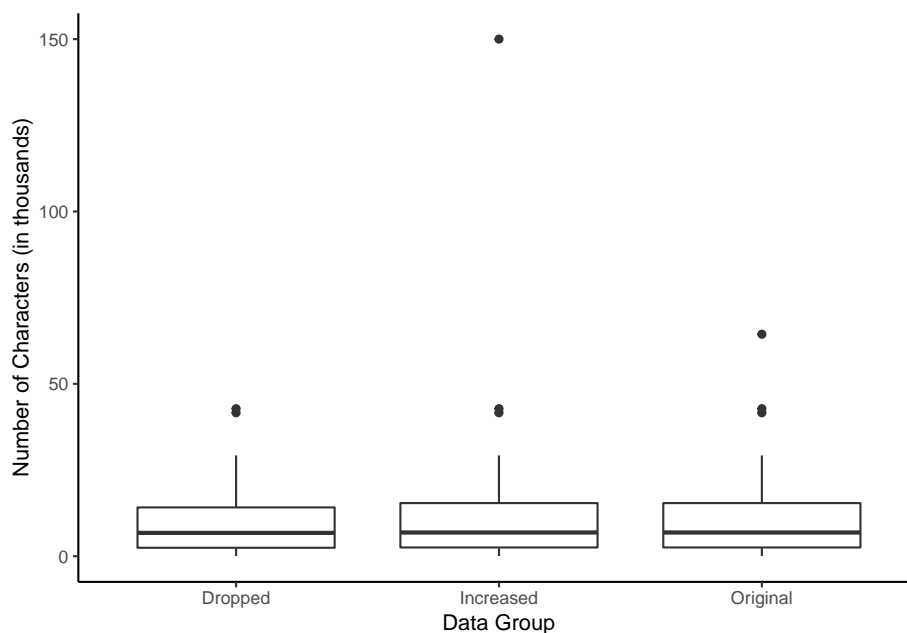


Figure 5.11: Box plots of the original character count data and two modified data sets.

```
##      group  min      Q1 median      Q3      max      mean      sd  n missing
## 1  Dropped 0.057 2.4540 6.7680 14.15600 42.793 10.52061 10.79768 49      0
## 2 Increased 0.057 2.5355 6.8895 15.41075 150.000 13.31020 22.43436 50      0
## 3 Original 0.057 2.5355 6.8895 15.41075 64.401 11.59822 13.12526 50      0
```

The code used to generate this table is

```

p1 <- email50$num_char
p2 <- p1[-which.max(p1)]
p3 <- p1
p3[which.max(p1)] <- 150

robust <- data.frame(value= c(p1,p2,p3),group=c(rep("Original",50),rep("Dropped",49),r

favstats(value~group,data=robust)

```

Notice by using the formula notation, we were able to calculate the summary statistics for each group.

Exercise:

- (a) Which is more affected by extreme observations, the mean or median? The data summary may be helpful.
- (b) Is the standard deviation or IQR more affected by extreme observations?¹⁸

The median and IQR are called **robust estimates** because extreme observations have little effect on their values. The mean and standard deviation are much more affected by changes in extreme observations.

Example:

The median and IQR do not change much under the three scenarios above. Why might this be the case?¹⁹

Exercise:

The distribution of vehicle prices tends to be right skewed, with a few luxury and sports cars lingering out into the right tail. If you were searching for a new car and cared about price, should you be more interested in the mean or median price of vehicles sold, assuming you are in the market for a regular car?²⁰

5.2.7 Transforming data

When data are very strongly skewed, we sometimes transform them so they are easier to model. Consider the histogram of salaries for Major League Baseball players' salaries from 2010, which is shown in Figure ??.

Example:

The histogram of MLB player salaries is useful in that we can see the

¹⁸(a) Mean is affected more. (b) Standard deviation is affected more.

¹⁹The median and IQR are only sensitive to numbers near Q_1 , the median, and Q_3 . Since values in these regions are relatively stable – there aren't large jumps between observations – the median and IQR estimates are also quite stable.

²⁰Buyers of a *regular car* should be concerned about the median price. High-end car sales can drastically inflate the mean price while the median will be more robust to the influence of those sales.

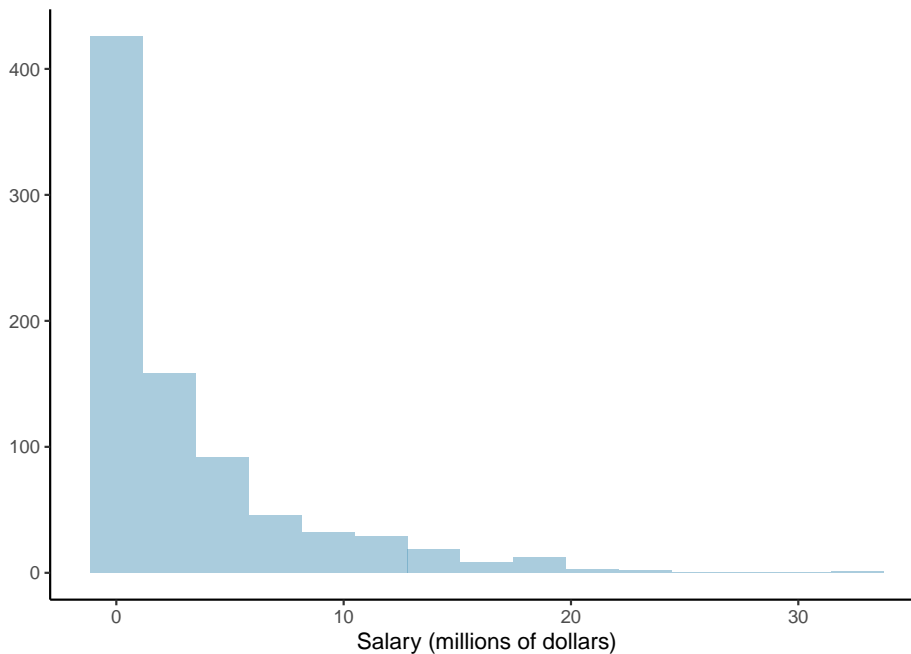


Figure 5.12: Histogram of MLB player salaries for 2010, in millions of dollars.

data are extremely skewed and centered (as gauged by the median) at about \$1 million. What isn't useful about this plot?²¹

There are some standard transformations that are often applied when much of the data cluster near zero (relative to the larger values in the data set) and all observations are positive. A **transformation** is a rescaling of the data using a function. For instance, a plot of the natural logarithm²² of player salaries results in a new histogram in Figure ???. Transformed data are sometimes easier to work with when applying statistical models because the transformed data are much less skewed and outliers are usually less extreme.

Transformations can also be applied to one or both variables in a scatterplot. A scatterplot of the `line_breaks` and `num_char` variables is shown in Figure ??? above. We can see a positive association between the variables and that many observations are clustered near zero. Later in this course, we might want to use a straight line to model the data. However, we'll find that the data in their current state cannot be modeled very well. Figure ??? shows a scatterplot where both the `line_breaks` and `num_char` variables have been transformed using a log (base e) transformation. While there is a positive association in each plot,

²¹Most of the data are collected into one bin in the histogram and the data are so strongly skewed that many details in the data are obscured.

²²Statisticians often write the natural logarithm as `log`. You might be more familiar with it being written as `ln`.

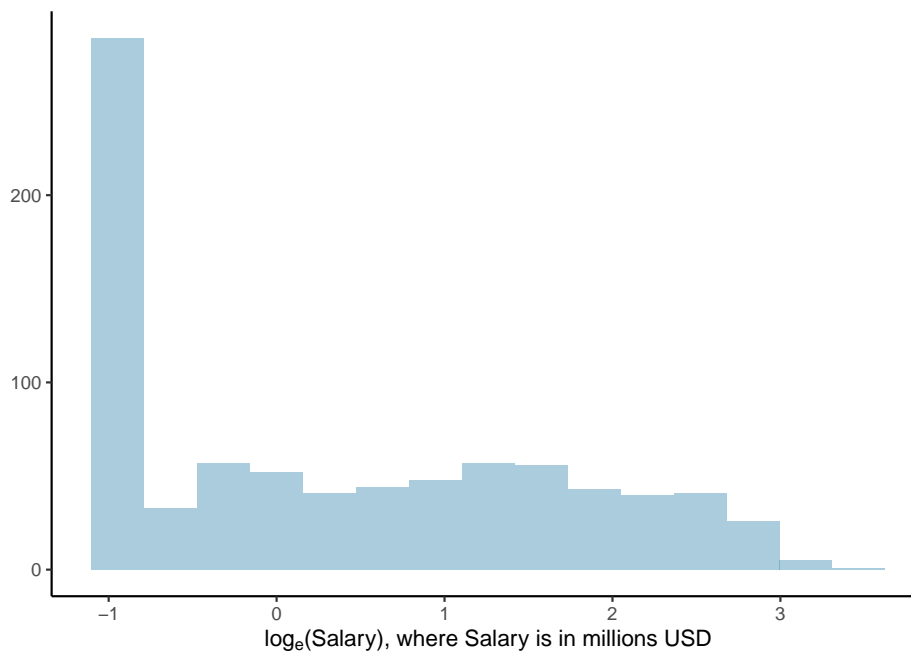


Figure 5.13: Histogram of the log-transformed MLB player salaries for 2010.

the transformed data show a steadier trend, which is easier to model than the untransformed data.

Transformations other than the logarithm can be useful, too. For instance, the square root ($\sqrt{\text{original observation}}$) and inverse ($\frac{1}{\text{original observation}}$) are used by statisticians. Common goals in transforming data are to see the data structure differently, reduce skew, assist in modeling, or straighten a nonlinear relationship in a scatterplot.

5.3 Homework Problems

Create an Rmd file for the work including headers, file creation data, and explanation of your work. Make sure your plots have a title and the axes are labeled. We are asking you to do more in this application to get ready for your Oral Board.

1. Mammals exploratory

Data were collected on 39 species of mammals distributed over 13 orders. The data is in the **openintro** package as **mammals**

- a. Using help, report the units for the variable **brain_Wt**.

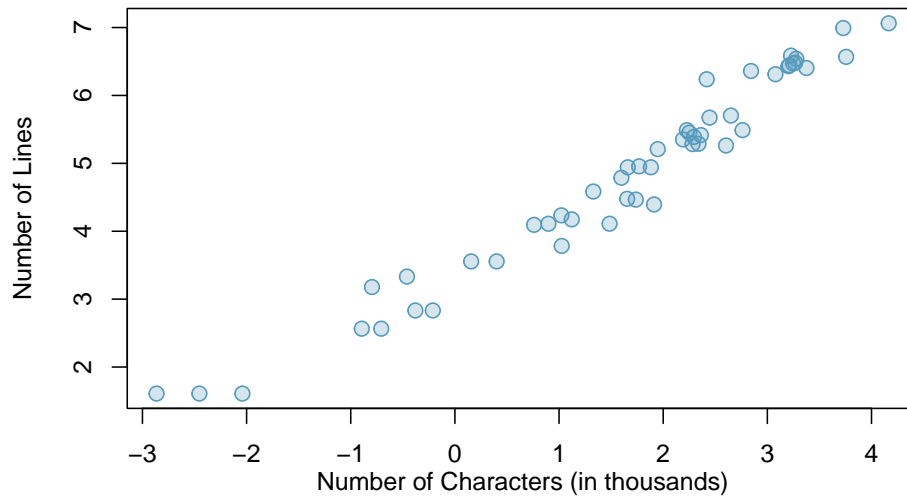


Figure 5.14: A scatterplot of `line_breaks` versus `num_char` for the `email50` data but where each variable has been log-transformed.

- b. Using `inspect` how many variables are numeric?
- c. What type of variable is `danger`?
- d. Create a histogram of `total_sleep` and describe the distribution.
- e. Create a boxplot of `life_span` and describe the distribution.
- f. Report the mean and median life span of a mammal.
- g. Calculate the summary statistics for `life_span` broken down by `danger`. What is the standard deviation of life span in danger outcome 5?

2. Mammals life spans

Continue using the `mammals` data set.

- a. Create side-by-side boxplots for `life_span` broken down by `exposure`. Note: you will have to change `exposure` to a `factor()`. Report on any findings.
- b. What happened to the median and third quartile in exposure group 4?
- c. Create faceted histograms. What are the shortcomings of this plot?
- d. Create a new variable `exposed` that is a factor with level `Low` if exposure is 1 or 2 and `High` otherwise.
- e. Repeat part c with the new variable. Explain what you see in the plot.

3. Mammals life spans continued

- a. Create a scatterplot of life span versus length of gestation.
- b. What type of an association is apparent between life span and length of gestation?
- c. What type of an association would you expect to see if the axes of the plot were reversed, i.e. if we plotted length of gestation versus life span?
- d. Create the new scatterplot suggested in c.
- e. Are life span and length of gestation independent? Explain your reasoning.

Chapter 6

Categorical Data

6.1 Objectives

- 1) Define and use properly in context all new terminology.
- 2) Generate in R tables for categorical variable(s).
- 3) Generate in R appropriate graphical summaries of categorical and numerical variables.
- 4) Be able to interpret and explain output both graphically and numerically.

6.2 Categorical data

Like numerical data, categorical data can also be organized and analyzed. This section introduces tables and other basic tools for categorical data. Remember at the beginning of this block of material, our case study had categorical data so we have seen some of the ideas in this lesson.

The `email50` data set represents a sample from a larger email data set called `email`. This larger data set contains information on 3,921 emails. In this section we will use the email data set to examine whether the presence of numbers, small or large, in an email provides any useful value in classifying email as spam or not spam.

6.2.1 Contingency tables and bar plots

In the `email` data set we have two variables: `spam` and `number` that we want to summarize. Let's use `inspect()` to get information and insight about the two variables. We can also type `?email` to learn more about the data. First load the `openintro` library.

Table 6.1: A contingency table for the ‘email’ data.

Spam	Number			Total
	none	small	big	
0	400	2659	495	3554
1	149	168	50	367
Total	549	2827	545	3921

```
library(openintro)

email %>%
  select(spam,number) %>%
  inspect()

##
## categorical variables:
##   name class levels   n missing
## 1  spam factor      2 3921      0
## 2 number factor      3 3921      0
##                                     distribution
## 1 0 (90.6%), 1 (9.4%)
## 2 small (72.1%), none (14%) ...
```

Notice the use of the `pipe` operator and how it adds to the ease of reading the code. The `select()` function allows us to narrow the variables down to the two of interest. Then `inspect()` gives us information about those variables. We read from top line; we start with the data set `email`, input it into `select()` and select variables from it, and then use `inspect()` to summarize the variables.

As is indicated `number` is a categorical variable that describes whether an email contains no numbers, only small numbers (values under 1 million), or at least one big number (a value of 1 million or more). The variable `spam` is a numeric variable where 1 indicates the email is spam. To treat it as categorical we will want to change it to a **factor** but first we will build a table that summarizes data for the two variables, see Table ???. This table is called a **contingency table**. Each value in the table represents the number of times a particular combination of variable outcomes occurred. We will show you the code to generate the contingency table.

```
tally(~spam+number,data=email,margins = TRUE)
```

```
##           number
## spam   none small  big Total
## 0      400 2659 495 3554
## 1      149  168  50  367
```

```
## Total 549 2827 545 3921
```

The value 149 corresponds to the number of emails in the data set that are spam *and* had no number listed in the email. Row and column totals are also included. The **row totals** provide the total counts across each row (e.g. $149 + 168 + 50 = 367$), and **column totals** are total counts down each column. The row and column totals are known as **marginal** counts and the values in the table, such as 149, as **joint** counts.

Let's turn `spam` into a factor and update the `email` data object. We will use `mutate()` to do this.

```
email <- email %>%
  mutate(spam = factor(email$spam, levels=c(1,0), labels=c("spam", "not spam")))
```

Now checking the data again.

```
email %>%
  select(spam, number) %>%
  inspect()
```

```
##
## categorical variables:
##   name class levels   n missing
## 1  spam factor      2 3921      0
## 2 number factor      3 3921      0
##
##                                distribution
## 1 not spam (90.6%), spam (9.4%)
## 2 small (72.1%), none (14%) ...
```

Let's generate the table again.

```
tally(~spam+number, data=email, margins = TRUE)
```

```
##           number
## spam      none small  big Total
##  spam      149  168   50   367
## not spam  400 2659  495 3554
## Total    549 2827  545 3921
```

A table for a single variable is called a **frequency table**. The table below is a frequency table for the `number` variable.

```
tally(~number, data=email)
```

```
## number
## none small  big
##  549 2827  545
```

If we replaced the counts with percentages or proportions, the table would be called a **relative frequency table**.

```
tally(~number,data=email,format='proportion')

## number
##      none      small      big
## 0.1400153 0.7209895 0.1389952

round(tally(~number,data=email,format='percent'),2)

## number
##  none small   big
##  14.0  72.1  13.9
```

A bar plot is a common way to display a single categorical variable. Figure ?? shows a **bar plot** for the **number** variable.

```
email %>%
  gf_bar(~number) %>%
  gf_theme(theme_bw()) %>%
  gf_labs(x="Size of Number",y="Count")
```

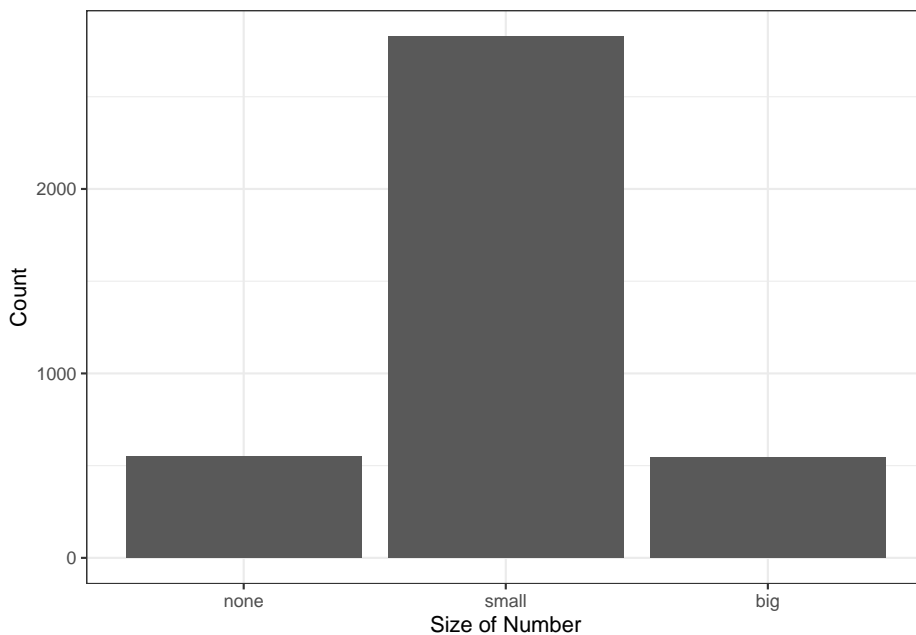


Figure 6.1: Bar chart of the **number** variable.

Next the counts are converted into proportions (e.g. $549/3921 = 0.140$ for **none**) in Figure ??.

```
email %>%  
  gf_props(~number) %>%  
  gf_theme(theme_bw()) %>%  
  gf_labs(x="Size of Number",y="Proportion")
```

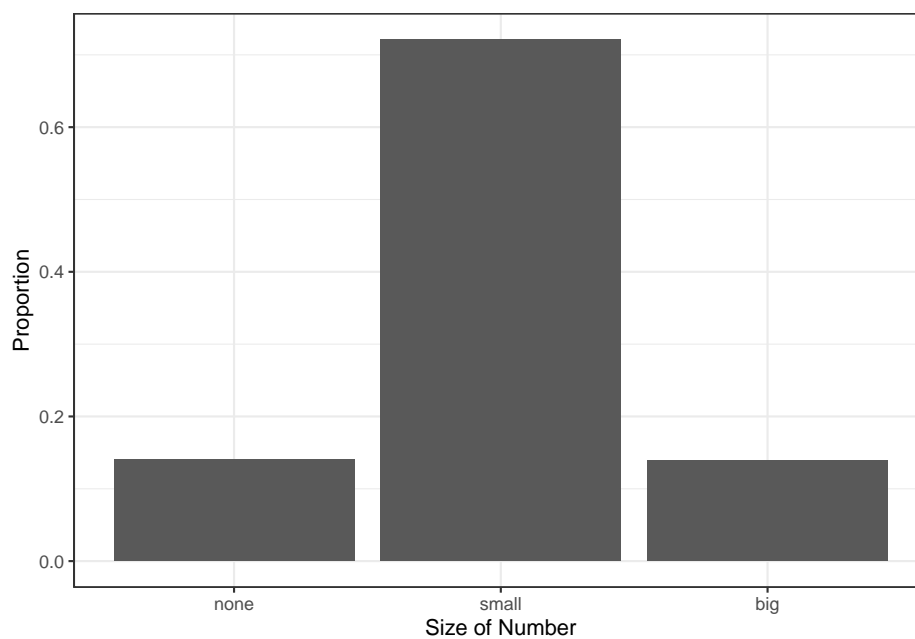
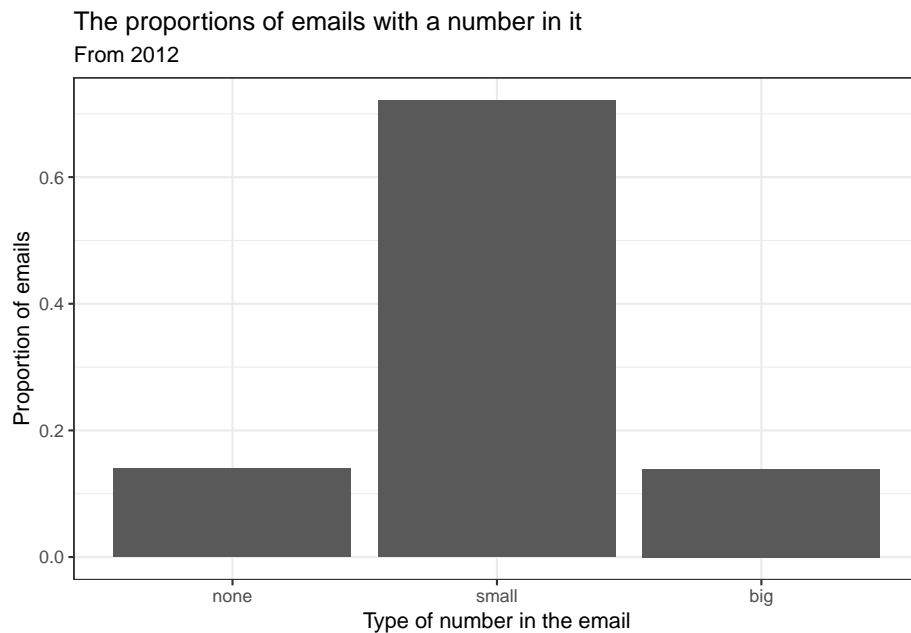


Figure 6.2: (ref:quote62)

Again, let's clean up the plot into a style that we could use in a report.

```
email %>%  
  gf_props(~number,title="The proportions of emails with a number in it",  
           subtitle="From 2012",xlab="Type of number in the email",  
           ylab="Proportion of emails") %>%  
  gf_theme(theme_bw())
```



6.2.2 Column proportions

The table below shows the column proportions. The **column proportions** are computed as the counts divided by their column totals. The value 149 at the intersection of *spam* and *none* is replaced by $149/549 = 0.271$, i.e. 149 divided by its row total, 549. So what does 0.271 represent? It corresponds to the proportion of emails in the sample with no numbers that are spam. We are **conditioning**, restricting, on emails with no number. This rate of spam is much higher than emails with only small numbers (5.9%) or big numbers (9.2%). Because these spam rates vary between the three levels of **number** (*none*, *small*, *big*), this provides evidence that the **spam** and **number** variables are associated.

```
tally(spam~number,data=email,margins = TRUE,format='proportion')
```

```
##           number
## spam      none      small      big
## spam    0.27140255 0.05942695 0.09174312
## not spam 0.72859745 0.94057305 0.90825688
## Total   1.00000000 1.00000000 1.00000000
```

The `tally()` function will always condition on the variable on the right hand side of the tilde, `~`, when calculating proportions and thus only generate column proportions. The more general `table()` function of R will allow either column or row proportions.

Exercise:

Create a table of column proportions where the variable **spam** is the

column variable.

```
tally(number~spam,data=email,margins = TRUE,format='proportion')
```

```
##          spam
## number      spam  not spam
##   none  0.4059946 0.1125492
##   small 0.4577657 0.7481711
##   big   0.1362398 0.1392797
##   Total 1.0000000 1.0000000
```

Exercise:

In the table you just created, what does 0.748 represent?¹

Example:

Data scientists use statistics to filter spam from incoming email messages. By noting specific characteristics of an email, a data scientist may be able to classify some emails as spam or not spam with high accuracy. One of those characteristics is whether the email contains no numbers, small numbers, or big numbers. Another characteristic is whether or not an email has any HTML content. A contingency table for the `spam` and `format` variables is needed.

1 Make `format` into a categorical factor variable. The levels should be “text” and “HTML”.²

2 Create a contingency table from the `email` data set with `format` in the columns and `spam` in the rows.

In deciding which variable to use as a column, the data scientist would be interested in how the proportion of spam changes within each email format. This corresponds to column proportions based on `format`: the proportion of spam in plain text emails and the proportion of spam in HTML emails.

```
email <- email %>%
  mutate(format = factor(email$format,levels=c(1,0),labels=c("HTML","text")))

tally(spam~format,data=email,margins = TRUE,format="proportion")
```

```
##          format
## spam      HTML      text
##   spam  0.05796038 0.17489540
##   not spam 0.94203962 0.82510460
##   Total   1.00000000 1.00000000
```

In generating the column proportions, we can see that a higher fraction of plain text emails are spam ($209/1195 = 17.5\%$) than compared to HTML emails ($158/2726 = 5.8\%$). This information on its own is insufficient to classify an email as spam or not spam, as over 80% of plain text emails are not spam. Yet,

¹0.748 represents the proportions of emails with no spam that had a small number in it.

²From the help menu on the data HTML is coded as a 1

when we carefully combine this information with many other characteristics, such as **number** and other variables, we stand a reasonable chance of being able to classify some email as spam or not spam.

In constructing a table, we need to think about which variable we want in the column and which in the row. The formula format in some way makes us think about the response and predictor variables. However in some cases, it is not clear which variable should be in the column and row and the analyst must decide the point to be made with the table. Before settling on one form for a table, it is important to consider the audience and the message they are to receive from the table.

Exercise:

Create two tables with **number** and **spam** where each are in the column, so two table where you change which variable is in the column. Which would be more useful to someone hoping to identify spam emails using the **number** variable?³

```
tally(spam~number,email,format='proportion',margin=TRUE)
```

```
##           number
## spam           none      small      big
##  spam      0.27140255 0.05942695 0.09174312
##  not spam  0.72859745 0.94057305 0.90825688
##  Total      1.00000000 1.00000000 1.00000000
```

```
tally(number~spam,email,format='proportion',margin=TRUE)
```

```
##           spam
## number      spam not spam
##  none  0.4059946 0.1125492
##  small 0.4577657 0.7481711
##  big   0.1362398 0.1392797
##  Total 1.0000000 1.0000000
```

6.2.3 Segmented bar and mosaic plots

Contingency tables using column proportions are especially useful for examining how two categorical variables are related. Segmented bar and mosaic plots provide a way to visualize the information in these tables.

A **segmented bar plot** is a graphical display of contingency table information. For example, a segmented bar plot representing the table with **number** in the column is shown in Figure ??, where we have first created a bar plot using the **number** variable and then separated each group by the levels of **spam**.

³The column proportions with **number** in the columns will probably be most useful, which makes it easier to see that emails with small numbers are spam about 5.9% of the time (relatively rare). We would also see that about 27.1% of emails with no numbers are spam, and 9.2% of emails with big numbers are spam.

```
email %>%
  gf_bar(~number, fill=~spam) %>%
  gf_theme(theme_bw()) %>%
  gf_labs(x="Size of Number", y="Count")
```

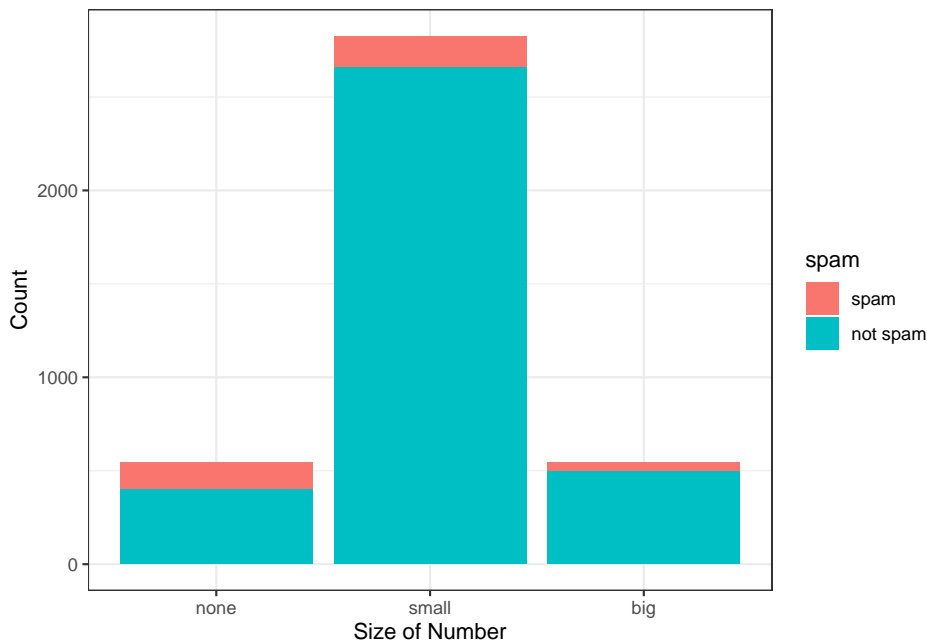


Figure 6.3: Segmented bar plot for numbers found in `emails`, where the counts have been further broken down by `spam`.

The column proportions of the table have been translated into a standardized segmented bar plot in Figure ??, which is a helpful visualization of the fraction of spam emails in each level of `number`.

```
email %>%
  gf_props(~number, fill=~spam, position='fill') %>%
  gf_theme(theme_bw()) %>%
  gf_labs(x="Size of Number", y="Proportion")
```

Example:

Examine both of the segmented bar plots. Which is more useful?

Figure ?? contains more information, but Figure ?? presents the information more clearly. This second plot makes it clear that emails with no number have a relatively high rate of spam email – about 27%! On the other hand, less than 10% of email with small or big numbers are spam.

Since the proportion of spam changes across the groups in Figure ??, we can

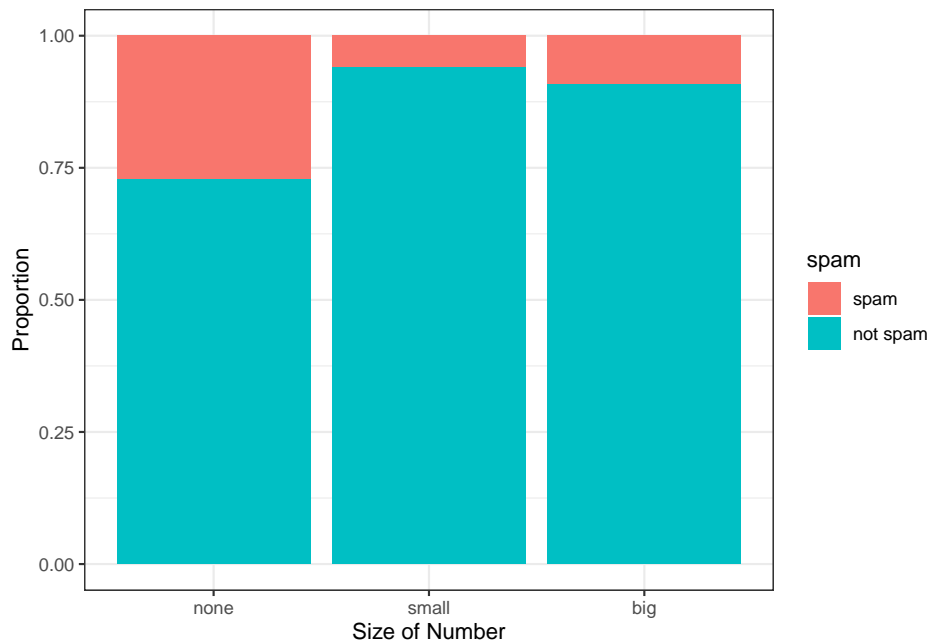


Figure 6.4: Standardized version of Figure ??.

conclude the variables are dependent, which is something we were also able to discern using table proportions. Because both the `none` and `big` groups have relatively few observations compared to the `small` group, the association is more difficult to see in Figure ??.

In some other cases, a segmented bar plot that is not standardized will be more useful in communicating important information. Before settling on a particular segmented bar plot, create standardized and non-standardized forms and decide which is more effective at communicating features of the data.

A **mosaic plot** is a graphical display of contingency table information that is similar to a bar plot for one variable or a segmented bar plot when using two variables. It seems strange, but mosaic plots are not part of the **mosaic** package. We must load another set of packages called **vcd** and **vcdExtra**. Mosaic plot displays help to visualize the pattern of associations among variables in two-way and larger tables. Mosaic plots are controversial since they rely on the perception of area. Human vision is not good at distinguishing areas.

We will introduce mosaic plots because it is another way to visualize contingency tables. Figure ?? shows a mosaic plot for the `number` variable. Each row represents a level of `number`, and the row heights correspond to the proportion of emails of each number type. For instance, there are fewer emails with no numbers than emails with only small numbers, so the `none` outcome row is shorter

in height. In general, mosaic plots use box *areas* to represent the number of observations. Since there is only one variable, the widths are all constant. Thus area is simply related to row height making this visual easy to read.

```
library(vcd)
```

```
mosaic(~number, data=email)
```

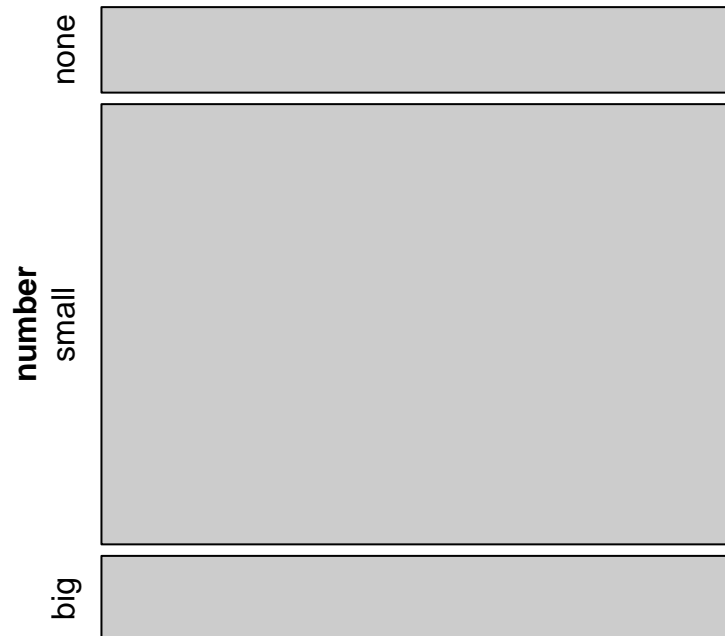


Figure 6.5: Mosaic plot where emails are grouped by the **number** variable.

This one-variable mosaic plot can be further divided into pieces as in Figure ?? using the **spam** variable. The first variable in the formula is used to determine row height. That is, each row is split proportionally according to the fraction of emails in each number category, these heights are similar to Figure ?. Next each row is split horizontally according to the proportion of emails that were spam in that number group. For example, the second row, representing emails with only small numbers, was divided into emails that were spam (left) and not spam (right). The area of the rectangles is proportional to the proportions in the table where each cell count is divided by the total count. First we will generate the table and then represent it as a mosaic plot.

```
tally(~number+spam, data=email, format='proportion')
```

```
##          spam
## number      spam  not spam
##   none  0.03800051 0.10201479
```

```
##   small 0.04284621 0.67814333
##   big   0.01275185 0.12624331
mosaic(~number+spam,data=email)
```

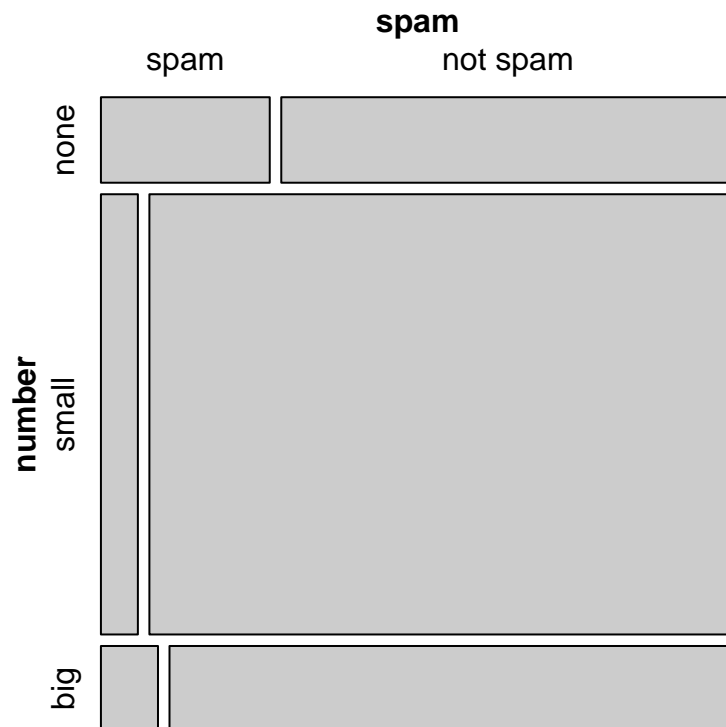


Figure 6.6: Mosaic plot with **number** as the first variable.

These plots are hard to use in a visual comparison of area. For example, is the area for *small* number *spam* emails different from *none* number *spam* emails? The rectangles have different shapes but from the table we can tell the areas are close.

An important use of the mosaic plot is to determine if an association between variables may be present. The bottom of the first column represents spam emails that had big numbers, and the bottom row of the second column represents regular emails that had big numbers. We can again use this plot to see that the **spam** and **number** variables are associated since some rows are divided in different vertical locations than others, which was the same technique used for checking an association in the standardized version of the segmented bar plot.

In a similar way, a mosaic plot representing column proportions where *spam* is in the column could be constructed. To completely understand the mosaic plot as shown in Figure ?? let's first find the proportions of **spam**.

```
tally(~spam,data=email,format="proportion")
```

```
## spam
##      spam    not spam
## 0.09359857 0.90640143
```

So the row heights will be split 90-10. Next let's find the proportions of number within each value of spam. In the spam row, *none* will be 41%, *small* will be 46%, and *big* will be 13%.

```
tally(number~spam,data=email,margins = TRUE,format="proportion")
```

```
##      spam
## number      spam    not spam
##   none 0.4059946 0.1125492
##   small 0.4577657 0.7481711
##    big  0.1362398 0.1392797
##   Total 1.0000000 1.0000000
```

However, because it is more insightful for this application to consider the fraction of spam in each category of the **number** variable, we prefer Figure ??.

```
mosaic(~spam+number,data=email)
```

6.2.4 The only pie chart you will see in this course, hopefully

While pie charts are well known, they are not typically as useful as other charts in a data analysis. A **pie chart** is shown in Figure ?. It is generally more difficult to compare group sizes in a pie chart than in a bar plot, especially when categories have nearly identical counts or proportions. In the case of the *none* and *big* categories, the difference is so slight you may be unable to distinguish any difference in group sizes.

```
pie(table(email$number), col=COL[c(3,1,2)], radius=0.75)
```

Pie charts are popular in the Air Force due to the ease of generating them in Excel and PowerPoint. However, the values for each slice are often printed on top of the chart making the chart irrelevant. We recommend a minimum use of pie charts in your work.

6.2.5 Comparing numerical data across groups

Some of the more interesting investigations can be considered by examining numerical data across groups. This is the case where one variable is categorical and the other is numerical. The methods required here aren't really new. All that is required is to make a numerical plot for each group. Here two convenient methods are introduced: side-by-side box plots and density plots.

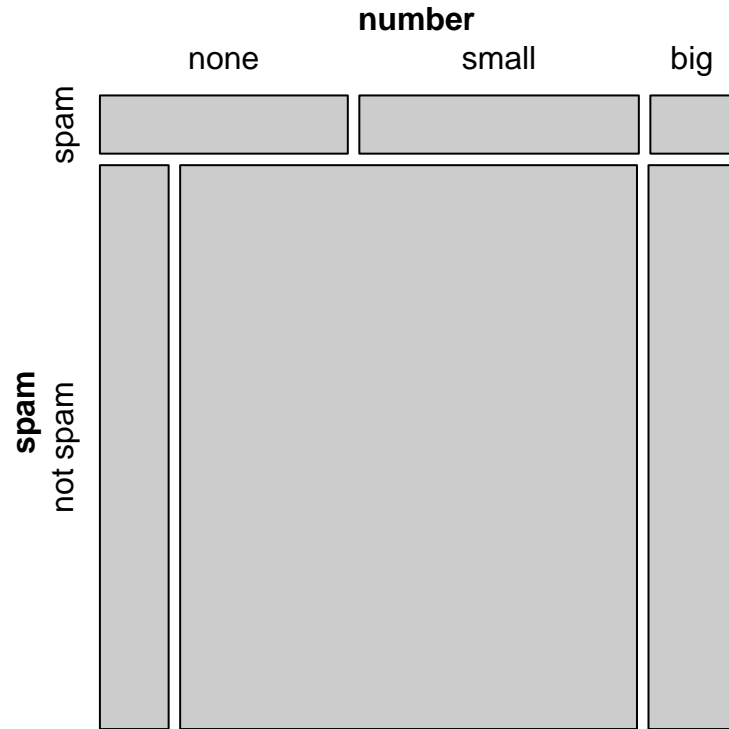
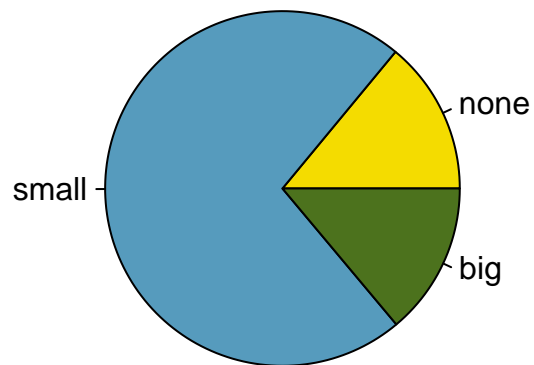
Figure 6.7: Mosaic plot with **spam** as the first variable

Figure 6.8: A pie chart number for the email data set.

We will take a look again at the subset of `county_complete` data set and compare the median household income for counties that gained population from 2000 to 2010 versus counties that had no gain. While we might like to make a causal connection here, remember that these are observational data and so such an interpretation would be unjustified.

This section will give us a chance to perform some data wrangling. We will be using the `tidyverse` verbs in the process. Data wrangling is an important part of analysis work and typically takes a significant portion of the analysis work.

Here is the code to generate the data we need.

```
library(usdata)

county_M377 <- county_complete %>%
  select(name, state, pop2000, pop2010, fed_spend=fed_spending_2009, poverty=poverty_2010,
         homeownership = homeownership_2010, multi_unit = housing_multi_unit_2010,
         income = per_capita_income_2010, med_income = median_household_income_2010) %>%
  mutate(fed_spend=fed_spend/pop2010)
```

First, as a reminder, let's look at the data.

What do we want R to do? We want to select the variables `pop2000`, `pop2010`, and `med_income`.

What does R need? It needs the data object, and variable names.

We will use the `select()` and `inspect()` functions.

```
county_M377 %>%
  select(pop2000, pop2010, med_income) %>%
  inspect()

##
## quantitative variables:
##      name  class  min      Q1 median   Q3     max    mean      sd
## ...1  pop2000 numeric    67 11223.50 24621 61775 9519338 89649.99 292547.67
## ...2  pop2010 numeric    82 11114.50 25872 66780 9818605 98262.04 312946.70
## ...3 med_income numeric 19351 36956.25 42450 49144 115574 44274.12 11547.49
##      n missing
## ...1 3139      3
## ...2 3142      0
## ...3 3142      0
```

Notice that three counties are missing population values, reported as `NA`. Let's remove them and find which counties increased population by creating a new variable.

```
cc_reduced <- county_M377 %>%
  drop_na(pop2000) %>%
```

```
select(pop2000,pop2010,med_income) %>%
mutate(pop_gain = sign(pop2010-pop2000))
```

```
tally(~pop_gain,data=cc_reduced)
```

```
## pop_gain
##   -1    0    1
## 1097    1 2041
```

There were 2,041 counties where the population increased from 2000 to 2010, and there were 1,098 counties with no gain, only 1 county had a net of zero, or a loss. Let's just look at the counties with a gain or loss in side-by-side boxplot. Again, we will use `filter()` to select the two groups and then make the variable `pop_gain` into a categorical variable, more data wrangling.

```
cc_reduced <- cc_reduced %>%
  filter(pop_gain != 0) %>%
  mutate(pop_gain = factor(pop_gain,levels=c(-1,1),labels=c("Loss","Gain")))
```

```
inspect(cc_reduced)
```

```
##
## categorical variables:
##      name  class levels    n missing
## 1 pop_gain factor      2 3138        0
##                                     distribution
## 1 Gain (65%), Loss (35%)
##
## quantitative variables:
##      name  class  min      Q1  median      Q3      max      mean
## ...1  pop2000 numeric    67 11217.25 24608.0 61783.5 9519338 89669.37
## ...2  pop2010 numeric    82 11127.00 25872.0 66972.0 9818605 98359.23
## ...3 med_income numeric 19351 36950.00 42443.5 49120.0 115574 44253.24
##      sd      n missing
## ...1 292592.28 3138      0
## ...2 313133.28 3138      0
## ...3 11528.95 3138      0
```

The **side-by-side box plot** is a traditional tool for comparing across groups. An example is shown in Figure ?? where there are two box plots, one for each group and drawn on the same scale.

```
cc_reduced %>%
  gf_boxplot(med_income~pop_gain,
             subtitle="The income data were collected between 2006 and 2010.",
             xlab="Population change from 2000 to 2010",
             ylab="Median Household Income") %>%
  gf_theme(theme_bw())
```

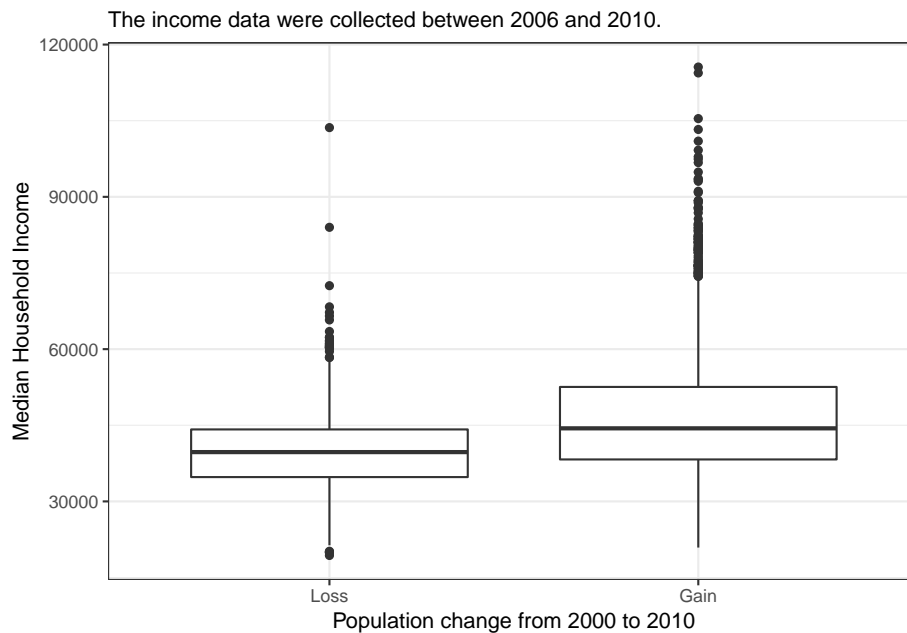


Figure 6.9: Side-by-side box plot for median household income, where the counties are split by whether there was a population gain or loss from 2000 to 2010.

Another useful plotting method uses **density plots** to compare numerical data across groups. A histogram bins data but is highly dependent on the number and boundary of the bins. A density plot also estimates the distribution of a numerical variable but does this by estimating the density of data points in a small window around each data point. The overall curve is the sum of this small density estimate. A density plot can be thought of as a smooth version of the histogram. Several options go into a density estimate such as the width of the window and type of smoothing function. These ideas are beyond the point here and we will just use the default options. Figure ?? is a plot of the two density curves.

```
cc_reduced %>%
  gf_dens(~med_income,color=~pop_gain,lwd=1) %>%
  gf_theme(theme_bw()) %>%
  gf_labs(x="Median household income",y="",col="Population \nChange")
```

Exercise:

Use the box plots and density plots to compare the incomes for counties across the two groups. What do you notice about the approximate center of each group? What do you notice about the variability between groups? Is the shape relatively consistent between groups?

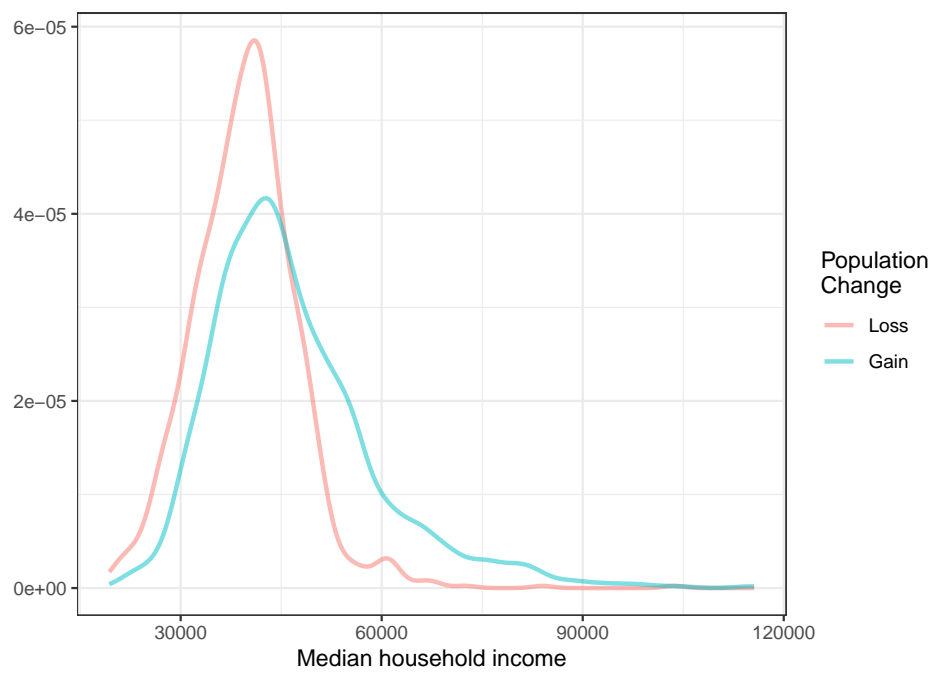


Figure 6.10: Density plots of median household income for counties with population gain versus population loss

How many *prominent* modes are there for each group?⁴

Exercise:

What components of each plot in Figures 8 and 9 do you find most useful?⁵

6.3 Homework Problems

Create an Rmd file for the work including headers, file creation data, and explanation of your work. Make sure your plots have a title and the axes are labeled.

1. Views on immigration

910 randomly sampled registered voters from Tampa, FL were asked if they thought workers who have illegally entered the US should be (i) allowed to keep their jobs and apply for US citizenship, (ii) allowed to keep their jobs as temporary guest workers but not allowed to apply for US citizenship, or (iii) lose their jobs and have to leave the country.

The data is in the **openintro** package in the **immigration** data object.

- How many levels of *political* are there?
- Create a table using **tally**.
- What percent of these Tampa, FL voters identify themselves as conservatives?
- What percent of these Tampa, FL voters are in favor of the citizenship option?
- What percent of these Tampa, FL voters identify themselves as conservatives and are in favor of the citizenship option?
- What percent of these Tampa, FL voters who identify themselves as conservatives are also in favor of the citizenship option? What percent of moderates and liberal share this view?
- Create a stacked bar chart.
- Using your plot, do political ideology and views on immigration appear to be independent? Explain your reasoning.

⁴Answers may vary a little. The counties with population gains tend to have higher income (median of about \$45,000) versus counties without a gain (median of about \$40,000). The variability is also slightly larger for the population gain group. This is evident in the IQR, which is about 50% bigger in the *gain* group. Both distributions show slight to moderate right skew and are unimodal. There is a secondary small bump at about \$60,000 for the *no gain* group, visible in the density plot, that seems out of place. (Looking into the data set, we would find that 8 of these 15 counties are in Alaska and Texas.) The box plots indicate there are many observations far above the median in each group, though we should anticipate that many observations will fall beyond the whiskers when using such a large data set.

⁵The side-by-side box plots are especially useful for comparing centers and spreads, while the density plots are more useful for seeing distribution shape, skew, and groups of anomalies.

2. **Views on the DREAM Act** The same survey from Exercise 1 also asked respondents if they support the DREAM Act, a proposed law which would provide a path to citizenship for people brought illegally to the US as children.

The data is in the **openintro** package in the **dream** data object.

- a. Create a **mosaic** plot.
- b. Based on the mosaic plot, are views on the DREAM Act and political ideology independent?

3. Heart transplants

The Stanford University Heart Transplant Study was conducted to determine whether an experimental heart transplant program increased lifespan. Each patient entering the program was designated an official heart transplant candidate, meaning that he was gravely ill and would most likely benefit from a new heart. Some patients got a transplant and some did not. The variable *transplant* indicates which group the patients were in; patients in the treatment group got a transplant and those in the control group did not. Another variable called *survived* was used to indicate whether or not the patient was alive at the end of the study.

The data is in the **openintro** package and is called **heart_transplant**.

- a. Create a **mosaic** plot.
- b. Based on the mosaic plot, is survival independent of whether or not the patient got a transplant? Explain your reasoning.
- c. Using the variable **survtime**, create side-by-side boxplots for the control and treatment groups.
- d. What do the box plots suggest about the efficacy (effectiveness) of transplants?

Part II

Probability Modeling

Chapter 7

Case Study

7.1 Objectives

- 1) Use R to simulate a probabilistic model.
- 2) Use basic counting methods.

7.2 Introduction to probability models

In this second block of material we will focus on probability models. We will take two approaches, one is mathematical and the other is computational. In some cases we can use both methods on a problem and in others only the computational approach is feasible. The mathematical approach to probability modeling allows us insight into the problem and the ability to understand the process. Simulation has a much greater ability to generalize but can be time intensive to run and often requires the writing of custom functions.

This case study is extensive and may seem overwhelming, do not worry we will discuss these ideas again in the many lessons we have coming up this block.

7.3 Probability models

Probability models are an important tool for data analysts. They are used to explain variation in outcomes that cannot be explained by other variables. We will use these ideas in block 3 to help us make decisions about our statistical models.

Often probability models are used to answer a question of the form “What is the chance that?” This means that we typically have an experiment or trial where multiple outcomes are possible and we only have an idea of the frequency

of those outcomes. We use this frequency as a measure of the probability of a particular outcome.

For this block we will focus just on probability models. To apply a probability model we will need to

1. Select the experiment and its possible values.
2. Have probability values for the outcomes which may include **parameters** that determine the probabilities.
3. Understand the assumptions behind the model

7.4 Case study

There is a famous example of a probability question that we will attack in this case study. The question we want to answer is “In a room of n people what is the chance that at least two people have the same birthday?”

Exercise:

The typical classroom at USAFA has 18 students in it. What do you think the chance that at least two students have the same birthday?¹

7.4.1 Break down the question

The first action we should take is to understand what is being asked.

1. What is the experiment or trial?
2. What does it mean to have the same birthday?
3. What about leap years?
4. What about the frequency of births? Are some days less likely than others?

Exercise:

Discuss these questions and others that you think are relevant.²

The best first step is to make a simple model, often these are the only ones that will have a mathematical solution. For our problem this means we answer the above questions.

1. We have a room of 18 people and we look at their birthdays. We either have two or more birthdays matching or not; thus there are two outcomes.
2. We don't care about the year, only the day and month. Thus two people born on May 16th are a match.
3. We will ignore leap years.
4. We will assume that a person has equal probability of being born on any of the 365 days of the year.
5. At least two means we could have multiple matches on the same day or several different days where multiple people have matching birthdays.

¹The answer is around 34.7%, how close were you?

²Another question may be What does it mean at least two people have matching birthdays?

7.4.2 Simulate (computational)

Now that we have an idea about the structure of the problem, we next need to think about how we would simulate a single classroom. We have 18 students in the classroom and they all could have any of the 365 days of the year as a birthday. What we need to do is sample birthdays for each of the 18 students. But how do we code the days of the year?

An easy solution is to just label the days from 1 to 365. The function `seq()` does this for us.

```
days <- seq(1,365)
```

Next we need to pick one of the days using the sample function. Note that we set the seed to get repeatable results, this is not required.

```
set.seed(2022)
sample(days,1)
```

```
## [1] 228
```

The first person was born on the 228th day of the year.

Since R works on vectors, we don't have to write a loop to select 18 days, we just have `sample()` do it for us.

```
class <- sample(days,size=18,replace = TRUE)
class
```

```
## [1] 206 311 331 196 262 191 206 123 233 270 248 7 349 112 1 307 288 354
```

What do we want R to do? Sample from the numbers 1 to 365 with replacement, which means a number can be picked more than once.

Notice in our sample we have at least one match, although it is difficult to look at this list and see the match. Let's sort them to make it easier for us to see.

```
sort(class)
```

```
## [1] 1 7 112 123 191 196 206 206 233 248 262 270 288 307 311 331 349 354
```

The next step is to find a way in R for the code to detect that there is a match.

Exercise:

What idea(s) can we use to determine if a match exists?

We could sort the data and look at differences in sequential values and then check if the set of differences contains a zero. This seems to be computationally expensive. Instead we will use the function `unique()` which gives a vector of unique values in an object. The function `length()` gives the number of elements in the vector.

```
length(unique(class))
```

```
## [1] 17
```

Since we only have 17 unique values in a vector of size 18, we have a match. Now let's put this all together to generate another classroom of size 18.

```
length(unique(sample(days,size=18,replace = TRUE)))
```

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

The next problem that needs to be solved is how to repeat the classrooms and keep track of those that have a match. There are several functions we could use to include `replicate()` but we will use `do()` from the **mosaic** package because it returns a data frame so we can use **tidyverse** verbs to wrangle the data.

The `do()` function allows us to repeat an operation many times. The following template

```
do(n) * {stuff to do}                # pseudo-code
```

where `{stuff to do}` is typically a single R command, but may be something more complicated.

Load the libraries.

```
library(mosaic)
library(tidyverse)
```

```
do(5)*length(unique(sample(days,size=18,replace = TRUE)))
```

```
##   length
## 1     18
## 2     17
## 3     17
## 4     17
## 5     18
```

Let's repeat for a larger number of simulated classroom, remember you should be asking yourself:

What do I want R to do?

What does R need to do this?

```
(do(1000)*length(unique(sample(days,size=18,replace = TRUE)))) %>%
  mutate(match=if_else(length==18,0,1)) %>%
  summarise(prob=mean(match))
```

```
##   prob
## 1 0.36
```

This is within 2 decimal places of the mathematical solution we develop shortly.

How many classrooms do we need to simulate to get an accurate estimate of the probability of a match? That is a statistical modeling question and it depends

on how much variability we can accept. We will discuss these ideas later in the semester. For now, you can run the code multiple times and see how the estimate varies. If computational power is cheap, you can increase the number of simulations.

```
(do(10000)*length(unique(sample(days,size=18,replace = TRUE)))) %>%
  mutate(match=if_else(length==18,0,1)) %>%
  summarise(prob=mean(match))
```

```
##      prob
## 1 0.3442
```

7.4.3 Plotting

By the way, the method we have used to create the data allows us to summarize the number of unique birthdays using a table or bar chart. Let's do that now. Note that since the first argument in `tally()` is not data then the **pipe** operator will not work without some extra effort. We must tell R that the data is the previous argument in the pipeline and thus use the symbol `.` to denote this.

```
(do(1000)*length(unique(sample(days,size=18,replace = TRUE)))) %>%
  tally(~length,data=.)
```

```
## length
## 14 15 16 17 18
## 1 7 52 253 687
```

Figure ?? is a plot of the number of unique birthdays in our sample.

```
(do(1000)*length(unique(sample(days,size=18,replace = TRUE)))) %>%
  gf_bar(~length) %>%
  gf_theme(theme_bw()) %>%
  gf_labs(x="Number of unique birthdays",y="Count")
```

Exercise:

What does it mean if the length of unique birthdays is 16, in terms of matches?³

7.4.4 Mathematical solution

To solve this problem mathematically, we will step through the logic one step at a time. One of the key ideas that we will see many times is the idea of the **multiplication** rule. This idea is the foundation for **permutation** and **combinations** which are counting methods frequently used in probability calculations.

³It is possible that 3 people all have the same birthday or two sets of 2 people have the same birthday but different from the other pair.

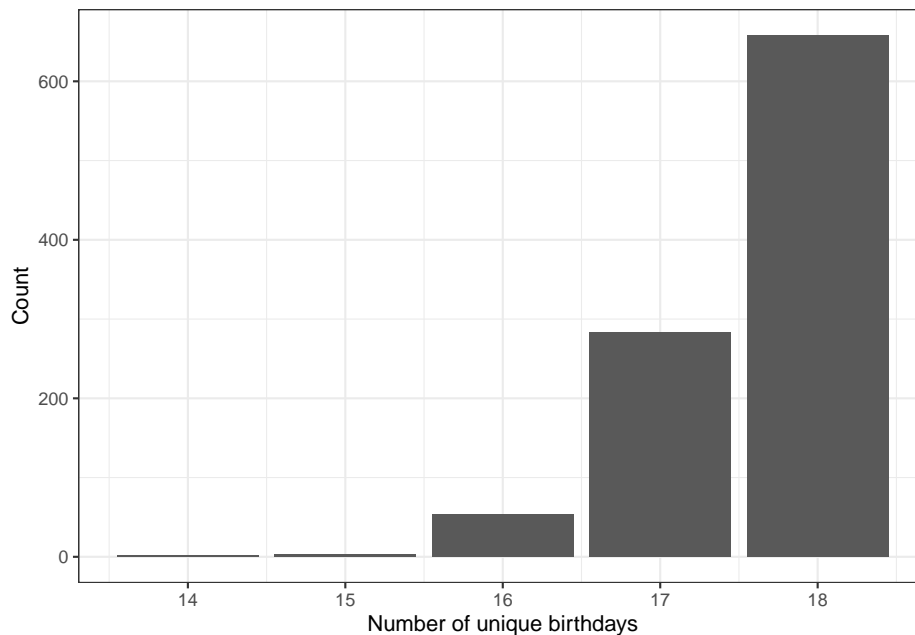


Figure 7.1: Bar chart of the number of unique birthdays in the sample.

The first step that we take is to understand the idea of 2 or more people with the same birthday. With 18 people, there are a great deal of possibilities for 2 or more birthdays. We could have exactly 2 people with the same birthday. We could have 18 people with the same birthday, We could have 3 people with the same birthday and another 2 people with the same birthday but different from the other 3. Accounting for all these possibilities is too large a counting process. Instead, we will take the approach of finding the probability of no one having a matching birthday. Then the probability of at least 2 people having a matching birthday is 1 minus the probability that no one has a matching birthday. This is known as a **complementary** probability. A simpler example is to think about rolling a single die. The probability of rolling a 6 is equivalent to 1 minus the probability of not rolling a 6.

We first need to think about all the different ways we could get 18 birthdays. This is going to be our denominator in the probability calculation. First let's just look at 2 people. The first person could have 365 different days for their birthday. The second person could also have 365 different birthdays. So for each birthday of the first person there could be 365 birthdays for the second. Thus for 2 people there are 365^2 possible sets of birthdays. This is an example of the *multiplication rule*. For 18 people there are 365^{18} sets of birthdays. That is a large number. Again, this will be our denominator in calculating the probability.

The numerator is the number of sets of birthdays with no matches. Again, let's

consider 2 people. The first person can have a birthday on any day of the year, so 365 possibilities. Since we don't want a match, the second person can only have 364 possibilities for a birthday. Thus we have 365×364 possibilities for two people to have different birthdays.

Exercise:

What is the number of possibilities for 18 people so that no one has the same birthday.

The answer for 18 people is $365 \times 364 \times 363 \dots \times 349 \times 348$. This looks like a truncated factorial. Remember a factorial, written as $n!$ with an explanation point, is the product of successive positive integers. As an example $3!$ is $3 \times 2 \times 1$ or 6. We could write the multiplication for the numerator as

$$\frac{365!}{(365 - n)!}$$

As we will learn, the multiplication rule for the numerator is known as a **permutation**.

We are ready to put it all together. For 18 people, the probability of 2 or more people with the same birthday is 1 minus the probability that no one has the same birthday, which is

$$1 - \frac{\frac{365!}{(365-18)!}}{365^{18}}$$

or

$$1 - \frac{\frac{365!}{347!}}{365^{18}}$$

In R there is a function called `factorial()` but factorials get large fast and we will **overflow** the memory. Try `factorial(365)` in R to see what happens.

```
factorial(365)
```

```
## [1] Inf
```

It is returning *infinity* because the number is too large for the buffer. As is often the case we will have when using a computational method, we must be clever about our approach. Instead of using factorials we can make use of R's ability to work on vectors. If we provide R with a vector of values, the `prod()` will perform a product of all the elements.

```
365*364
```

```
## [1] 132860
```

```
prod(365:364)

## [1] 132860
1- prod(365:348)/(365^18)

## [1] 0.3469114
```

7.4.5 General solution

We now have the mathematics to understand the problem. We can easily generalize this to any number of people. To do this, we have to write a function in R. As with everything in R, we save a function as an object. The general format for creating a function is

```
my_function <- function(parameters){
  code for function
}
```

For this problem we will call the function `birthday_prob()`. The only parameter we need is the number of people in the room, `n`. Let's write this function.

```
birthday_prob <- function(n=20){
  1- prod(365:(365-(n-1)))/(365^n)
}
```

Notice we assigned the function to the name `birthday_prob`, we told R to expect one argument to the function, which we are calling `n`, and then we provide R with the code to find the probability. We set a default value for `n` in case one is not provided to prevent an error when the function is run. We will learn more about writing functions over this and the next semester.

Test the code with a know answer.

```
birthday_prob(18)
```

```
## [1] 0.3469114
```

Now we can determine the probability for any size room. You may have heard that it only takes about 23 people in a room to have a 50% probability of at least 2 people matching birthdays.

```
birthday_prob(23)
```

```
## [1] 0.5072972
```

Let's create a plot of the probability versus number of people in the room. To do this, we need to apply the function to a vector of values. The function `sapply()` will work or we can also use `Vectorize()` to alter our existing function. We choose the latter option.

First notice what happens if we input a vector into our function.

```
birthday_prob(1:20)
```

```
## Warning in 365:(365 - (n - 1)): numerical expression has 20 elements: only the
## first used
```

```
## [1] 0.0000000 0.9972603 0.9999925 1.0000000 1.0000000 1.0000000 1.0000000
## [8] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
## [15] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
```

It only uses the first value. Let's vectorize it.

```
birthday_prob <- Vectorize(birthday_prob)
```

Now notice what happens.

```
birthday_prob(1:20)
```

```
## [1] 0.000000000 0.002739726 0.008204166 0.016355912 0.027135574 0.040462484
## [7] 0.056235703 0.074335292 0.094623834 0.116948178 0.141141378 0.167024789
## [13] 0.194410275 0.223102512 0.252901320 0.283604005 0.315007665 0.346911418
## [19] 0.379118526 0.411438384
```

We are good to go. Let's create our line plot, Figure ??.

```
gf_line(birthday_prob(1:100)~ seq(1,100),
        xlab="Number of People",
        ylab="Probability of Match",
        title="Probability of at least 2 people with matching birthdays") %>%
  gf_theme(theme_bw())
```

Is this what you expected the curve to look like? We, the authors, did not expect this. It has a sigmoidal shape with a large increase in the middle range and flatten in the tails.

7.4.6 Data science approach

The final approach we will take is one based on data, a data science approach. In the **mosaicData** package is a data set called **Births** that contains the number of births in the US from 1969 to 1988. This data will allow us to estimate the number of births on any day of the year. This allows us to eliminate the reliance on the assumption that each day is equally likely. Let's first **inspect()** the data object.

```
inspect(Births)
```

```
##
## categorical variables:
##   name    class levels    n missing
## 1 wday ordered      7 7305      0
```

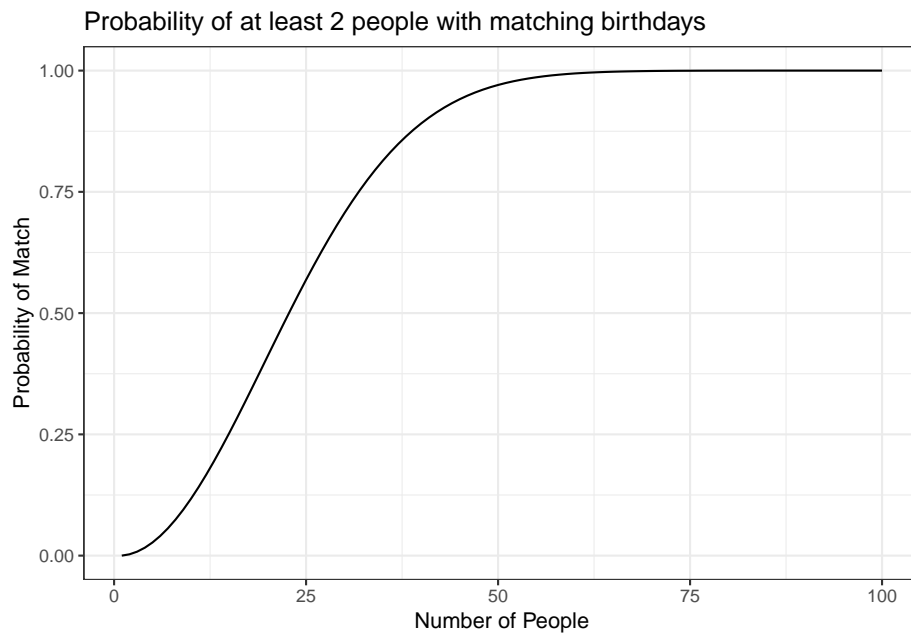


Figure 7.2: The probability of at least 2 people having matching birthdays

```
##                                     distribution
## 1 Wed (14.3%), Thu (14.3%), Fri (14.3%) ...
##
## Date variables:
##   name class      first      last min_diff max_diff  n missing
## 1 date  Date 1969-01-01 1988-12-31   1 days   1 days 7305      0
##
## quantitative variables:
##       name  class  min  Q1 median  Q3  max      mean      sd
## ...1   births integer 6675 8792  9622 10510 12851 9648.940178 1127.315229
## ...2    year integer 1969 1974  1979  1984  1988 1978.501027   5.766735
## ...3   month integer    1    4    7    10    12   6.522930   3.448939
## ...4 day_of_year integer    1   93   184   275   366 183.753593 105.621885
## ...5 day_of_month integer    1    8    16    23    31  15.729637   8.800694
## ...6 day_of_week integer    1    2    4    6    7   4.000274   1.999795
##
##       n missing
## ...1 7305      0
## ...2 7305      0
## ...3 7305      0
## ...4 7305      0
## ...5 7305      0
## ...6 7305      0
```

It could be argued that we could randomly pick one year and use it. Let's see what happens if we just used 1969. Figure ?? is a scatter plot of the number of births in 1969 for each day of the year.

```
Births %>%
  filter(year == 1969) %>%
  gf_point(births~day_of_year) %>%
  gf_theme(theme_bw()) %>%
  gf_labs(x="Day of the Year",y="Number of Births")
```

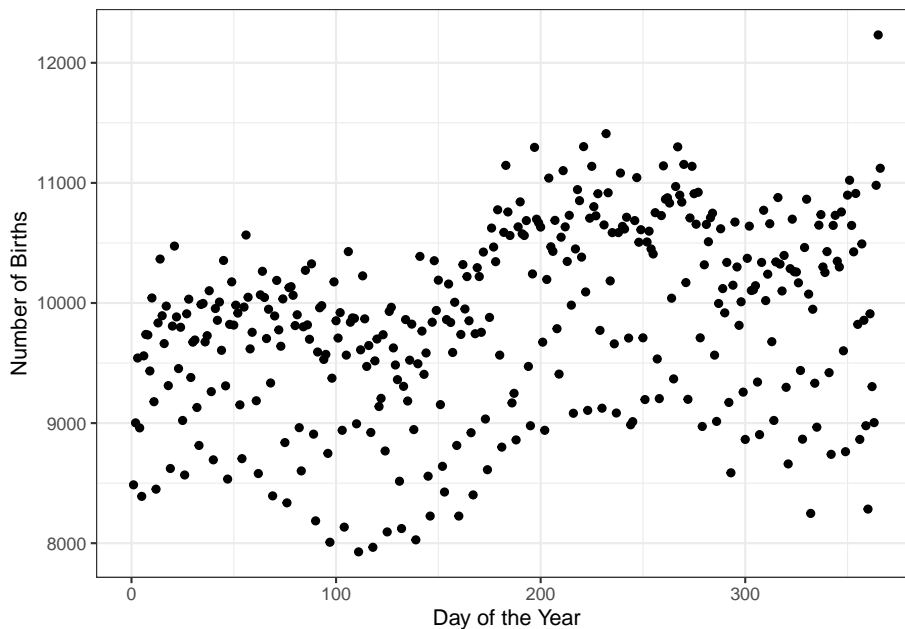


Figure 7.3: The number of births for each day of the year in 1969

Exercise:

What patterns do you see in Figure ??? What might explain them?

There are definitely bands appearing in the data which could be the day of the week; there are less birthdays on the weekend. There is also seasonality with more birthdays in the summer and fall. There is also probably an impact from holidays.

Quickly, let's look at the impact of day of the week by using color for day of the week. Figure ?? makes it clear that the weekends have less number of births as compared to the work week.

```
Births %>%
  filter(year == 1969) %>%
  gf_point(births~day_of_year,color=~factor(day_of_week)) %>%
```

```
gf_labs(x="Day of the Year",col="Day of Week") %>%
gf_theme(theme_bw())
```

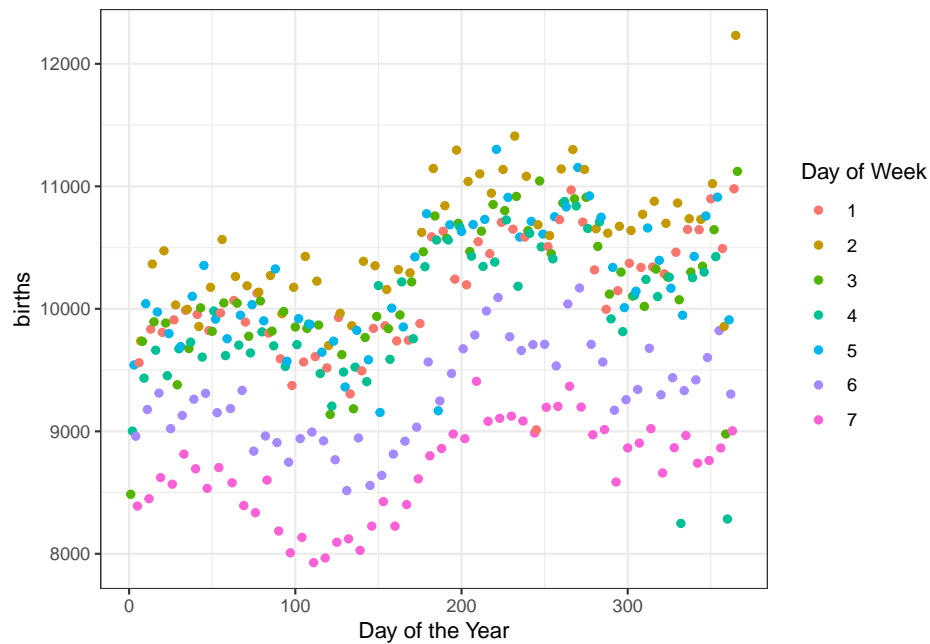


Figure 7.4: The number of births for each day of the year in 1969 broken down by day of the week

By only using one year, this data might give poor results since holidays will fall on certain days of the week and the weekends will also be impacted. Note that we also still have the problem of leap years.

```
Births %>%
  group_by(year) %>%
  summarise(n=n())
```

```
## # A tibble: 20 x 2
##   year      n
##   <int> <int>
## 1 1969   365
## 2 1970   365
## 3 1971   365
## 4 1972   366
## 5 1973   365
## 6 1974   365
## 7 1975   365
## 8 1976   366
```

```
## 9 1977 365
## 10 1978 365
## 11 1979 365
## 12 1980 366
## 13 1981 365
## 14 1982 365
## 15 1983 365
## 16 1984 366
## 17 1985 365
## 18 1986 365
## 19 1987 365
## 20 1988 366
```

The years 1972, 1976, 1980, 1984, and 1988 are all leap years. At this point, to make the analysis easier, we will drop those years.

```
Births %>%
  filter(!(year %in% c(1972,1976,1980,1984,1988))) %>%
  group_by(year) %>%
  summarise(n=n())
```

```
## # A tibble: 15 x 2
##   year      n
##   <int> <int>
## 1 1969 365
## 2 1970 365
## 3 1971 365
## 4 1973 365
## 5 1974 365
## 6 1975 365
## 7 1977 365
## 8 1978 365
## 9 1979 365
## 10 1981 365
## 11 1982 365
## 12 1983 365
## 13 1985 365
## 14 1986 365
## 15 1987 365
```

Notice in `filter()` we used the `%in%` argument. This is a **logical** argument checking if `year` is one of the values. The `!` at the front negates this in a sense requiring `year` not to be one of those values.

We are almost ready to simulate. We need to get the count of `births` on each day of the year for the non-leap years.

```
birth_data <- Births %>%
  filter(!(year %in% c(1972,1976,1980,1984,1988))) %>%
  group_by(day_of_year) %>%
  summarise(n=sum(births))
```

```
head(birth_data)
```

```
## # A tibble: 6 x 2
##   day_of_year      n
##       <int>  <int>
## 1         1 120635
## 2         2 129042
## 3         3 135901
## 4         4 136298
## 5         5 137319
## 6         6 140044
```

Let's look at a plot of the number of births versus day of the year. We combined years in Figure ??.

```
birth_data %>%
  gf_point(n~day_of_year,
           xlab="Day of the year",
           ylab="Number of births") %>%
  gf_theme(theme_bw())
```

This curve has the seasonal cycling we would expect. The smaller scale cycling is unexpected. Maybe because we are dropping the leap years, we are getting some days appearing in our time interval more frequently on weekends. We leave it to you to investigate this phenomenon.

We use these counts as weights in a sampling process. Days with more births will have a higher probability of being selected. Days such as Christmas and Christmas Eve have a lower probability of being selected. Let's save the weights in an object to use in the `sample()` function.

```
birth_data_weights <- birth_data %>%
  select(n) %>%
  pull()
```

The `pull()` function pulls the vectors of values out of the data frame format into a vector format which the `sample()` needs.

Now let's simulate the problem. The probability of a match should change slightly, maybe go down slightly?, but not much since most of the days have about the same probability or number of occurrences.

```
set.seed(20)
(do(1000)*length(unique(sample(days,size=18,replace = TRUE,prob=birth_data_weights))))
```

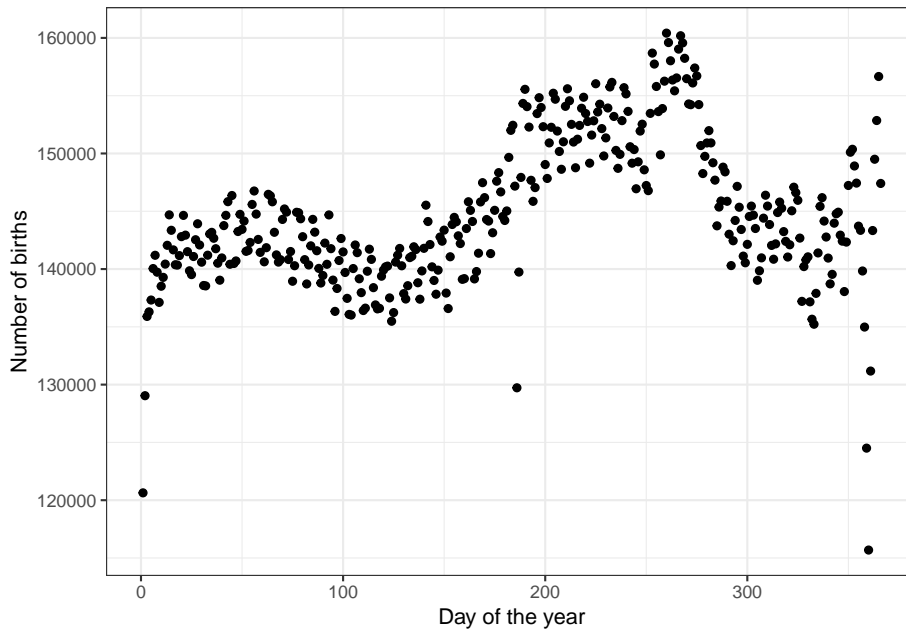



Figure 7.5: Number of births by day of the year for all years.

```
mutate(match=if_else(length==18,0,1)) %>%
summarise(prob=mean(match))
```

```
##      prob
## 1 0.352
```

We could not solve this problem of varying frequency of birth days using mathematics, at least as far as we know.

Cool stuff, let's get to learning more about probability models in the next chapters.

7.5 Homework Problems

1. **Exactly 2 people with the same birthday - Simulation.** Complete a similar analysis for case where exactly 2 people in a room of 23 people have the same birthday. In this exercise you will use a computational simulation.
 - a. Create a new R Markdown file and create a report. Yes, we know you could use this file but we want you to practice generating your own report.

- b. Simulate having 23 people in the class with each day of the year equally likely. Find the cases where exactly 2 people have the same birthday, you will have to alter the code from the Notes more than changing 18 to 23.
- c. Plot the frequency of occurrences as a bar chart.
- d. Estimate the probability of exactly two people having the same birthday.
- 2. **Exactly 2 people with the same birthday - Mathematical.** Repeat problem 1 but do it mathematically. As a big hint, you will need to use the `choose()` function. The idea is that with 23 people we need to choose 2 of them to match. We thus need to multiply, the multiplication rule again, by `choose(23,2)`. If you are having trouble, work with a total of 3 people in the room first.
 - a. Find a formula to determine the exact probability of exactly 2 people in a room of 23 having the same birthday.
 - b. Generalize your solution to any number `n` people in the room and create a function.
 - c. Vectorize the function.
 - d. Plot the probability of exactly 2 people having the same birthday versus number of people in the room.
 - e. Comment on the shape of the curve and explain it.