STAT 234: Data Science

Matt Higham

2021-05-12

Contents

Chapter 1

Course Description

Describe course, maybe give information on `R Studio` server.

This is a *sample* book written in **Markdown**. You can use anything that Pandoc's Markdown supports, e.g., a math equation $a^2 + b^2 = c^2$.

The **bookdown** package can be installed from CRAN or Github:

Remember each Rmd file contains one and only one chapter, and a chapter is defined by the first-level heading `#`.

Chapter 2

Introduction: Getting Started with R and R Studio

Figures and tables with captions will be placed in `figure` and `table` environments, respectively.

```
par(mar = c(4, 4, .1, .1))
plot(pressure, type = 'b', pch = 19)
```

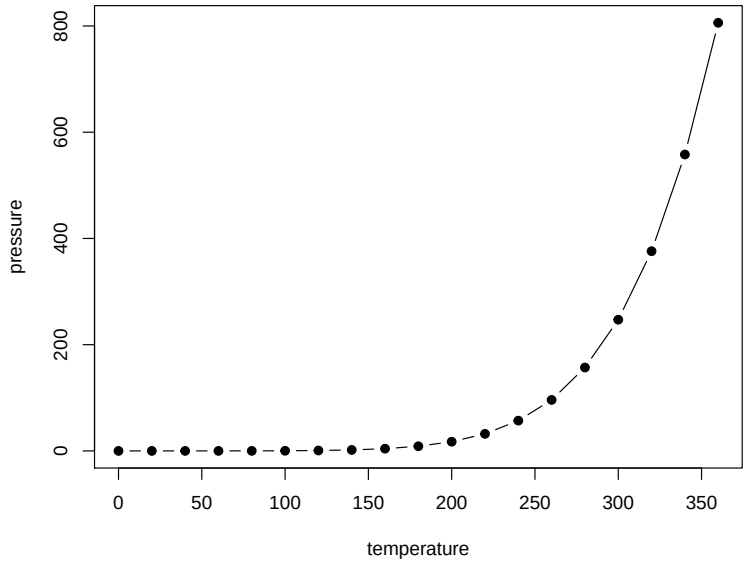

Figure 2.1: Here is a nice figure!

Goals:

1. Use `R Studio` on the server
2. Use `R Markdown` and code chunks
3. Load in data to `R Studio`
4. Run code and change a few things within that code
5. Correct some common errors when running code in `R`

Note: If you are using a downloaded version of `R Studio` because you already have it on your computer from Spring 2020, then you'll need to install the `rmdformats` package. If you're using the server, then ignore this note :)

2.1 Intro to `R` and `R Studio`

`R` is a statistical computing software used by many statisticians as well as professionals in other fields, such as biology, ecology, business, and psychology. The goal of Week 0 is to provide basic familiarity with `R` and `R Markdown`, which we will be using for the entire semester.

Open `R Studio` on the SLU `R Studio` server at http://rstudio.stlawu.local:8787 and create a folder called STAT_234 or some other meaningful title to you. Note that you must be on campus to use the `R Studio` server, unless you use a VPN. Directions on how to set-up VPN are https://infotech.stlawu.edu/support/content/11269 for Macs and https://stlawu.teamdynamix.com/TDClient/1805/Portal/KB/ArticleDet?ID=55118 for Windows.

Next, create a subfolder within your STAT_234 folder. Title it *Notes* (or whatever you want really).

Then, create an `R Project` by Clicking File -> New Project -> New Directory. Name the new project Week0_R_Intro (or some other meaningful title to you) and put it in the Notes folder. We will make a new project for each topic of class, for each homework assignment that requires the use of `R`, and for each project we have in class.

After the project is created, download the *Week0_IntroR.zip* file found on Sakai (in Resources) to your folder with the new project. Then, you can upload that file in to the server by clicking "Upload" in the bottom right panel. In the dialog box that appears, you can click "Choose File" and navigate to the folder where you saved the .Rmd file (probably Downloads by default). The zip file will automatically expand once uploaded. It includes a .Rmd file, a .html file, and two .csv data sets.

Once you upload the file, click on it to open it and scroll down to this line.

From this point forward, I recommend following along with the .Rmd file that you just opened. Reading this type of file takes a little bit of getting used to, but I think it's easier than trying to jump back and forth between the .Rmd and .html file.

Before moving on, click the **Knit** button in the top-left window at the top of the menu bar (look for the knitting needle icon). Make sure that the file knits to a pretty-looking .html file. This .html file is the same as the one that you've been reading along with so far. The newly knitted .html file can now be found in your folder with your R project.

2.2 What are R, R Studio, and R Markdown?

The distinction between the 3 will become more clear later on. For now,

- R is a statistical coding software used heavily for data analysis and statistical procedures.
- R Studio is a nice IDE (Integrated Development Environment) for R that has a lot of convenient features. Think of this as just a convenient User Interface.
- R Mardkown allows users to mix regular Microsoft-Word-style text with code. The .Rmd file ending denotes an R Mardkown file. R Markdown has many options that we will use heavily throughout the semester, but there's no need to worry about these now.

2.2.0.0.1 R Packages and the tidyverse You can think of R packages as add-ons to R that let you do things that R on its own would not be able to do. If you're in to video games, you can think of R packages as extra Downloadable Content (DLC). But, unlike most gaming DLC, R packages are always free and we will make very heavy use of R packages.

The tidyverse is a series of R packages that are useful for data science. In the order that we will encounter them in this class, the core tidyverse packages are:

1. ggplot2 for plotting data
2. dplyr for data wrangling and summarizing
3. tidyr for data tidying and reshaping
4. readr for data import
5. tibble for how data is stored
6. stringr for text data

7. `forcats` for factor (categorical) data
8. `purrr`, for functional programming, the only one of these core 8 that we won't get to use

We will use packages outside of the core `tidyverse` as well, but the `tidyverse` is the main focus.

2.2.0.0.2 Putting Code in a `.Rmd` File We are going to change one option before proceeding. In the top file menu, click Tools -> Global Options -> R Markdown and then uncheck the box that says "Show output inline for all R Markdown documents". Don't worry about this for now, but changing this option just means that code results will appear in the bottom-left window and graphs will appear in the bottom-right window of `R Studio`.

Up to this point, we have only had text in our .Rmd file. The first thing that we will do that involves code is to load a package into `R` with the `library()` function. A package is just an `R` add-on that lets you do more than you could with just `R` on its own. Load the `tidyverse` package into `R` by typing and running the `library(tidyverse)` line. You can run code by placing your cursor in the line of code and

1. Clicking the "Run" button in the menu bar of the top-left window of `R Studio` or
2. (Recommended) Clicking "Command + Enter" on a Mac or "Control + Enter" on a PC.

Note that all code (like what appears in the following lines) appears in grey boxes surrounded by three backticks while normal text (like what you are reading now) has a white background with no backticks.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.1     v dplyr   1.0.4
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

When you run the following line, some text will appear in the bottom-left window. We won't worry too much about what this text means now, but we also won't ignore it completely. You should be able to spot the 8 core `tidyverse` packages listed above as well as some numbers that follow each package. The

numbers correspond to the package version. There's some other things too, but as long as this text does not start with "Error:", you're good to go!

Congrats on running your first line of code for this class! This particular code isn't particularly exciting because it doesn't really do anything that we can see.

We have run `R` code using an `R` chunk. While most of this document is text (not `R` code), the text inside the three opening quotation marks is `R` code. Try making your own `R` chunk below by clicking **Insert** (the green button in the top left window) -> R. In your `R` chunk, perform the calculation `99 + 10` by typing `99 + 10` into your code chunk and pressing Command + Enter or Control + Enter.

So, that still wasn't super exciting. `R` can perform basic calculations, but you could just use a calculator or Excel for that. In order to look at things that are a bit more interesting, we need some data.

2.3 Alcohol Data

We will be looking at two data sets just to get a little bit of a preview of things we will be working on for the rest of the semester. **Important**: Do not worry about understanding what all of this code is doing at this point. There will be plenty of time to understand this in the weeks ahead. The purpose of this section is just to get used to using `R`: there will be more detailed videos, explanations, and exercises about the functions used and various options in the coming weeks. In particular, the following code uses the `ggplot2`, `dplyr`, and `tidyr` packages, which we will cover in detail throughout the first ~ 4 weeks of this course.

Data for this first part was obtained from fivethirtyeight at the following link: https://github.com/fivethirtyeight/data/tree/master/alcohol-consumption

The first step is to read the data set into `R`. Though you have already downloaded alcohol.csv, we still need to load it into `R`. Check to make sure the alcohol.csv is in the Files pane of your bottom-right hand window. Then, run the following line by placing your cursor in the line below and clicking Cmd + Enter or Control + Enter:

```
read_csv("alcohol.csv")
```

Note that we do not need the full file extension **if** we have the data set in an `R` project.

Did something show up in your console window? If so, great! If not, make sure that the data set is in your `R` project folder.

We would like to name our data set something so that we could easily reference it later, so name your data set using the `<-` operator, as in

```
alcohol_data <- read_csv("alcohol.csv")
head(alcohol_data)
```

```
## # A tibble: 6 x 5
##   country     beer_servings spirit_servings wine_servings total_litres_of_pure~
##   <chr>               <dbl>           <dbl>         <dbl>                 <dbl>
## 1 Afghanistan             0               0             0                   0
## 2 Albania                89             132            54                   4.9
## 3 Algeria                25               0            14                   0.7
## 4 Andorra               245             138           312                  12.4
## 5 Angola                217              57            45                   5.9
## 6 Antigua & B~          102             128            45                   4.9
```

Reference a figure by its code chunk label with the `fig:` prefix, e.g., see Figure 2.1. Similarly, you can reference tables generated from `knitr::kable()`, e.g., see Table 2.1.

```
knitr::kable(
  head(iris, 20), caption = 'Here is a nice table!',
  booktabs = TRUE
)
```

You can write citations, too. For example, we are using the **bookdown** package (Xie, 2020) in this sample book, which was built on top of R Markdown and **knitr** (Xie, 2015).

Table 2.1: Here is a nice table!

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa

Chapter 3

Literature

Here is a review of existing methods.

Chapter 4

Methods

We describe our methods in this chapter.

Chapter 5

Applications

Some *significant* applications are demonstrated in this chapter.

5.1 Example one

5.2 Example two

Chapter 6

Final Words

We have finished a nice book.

Bibliography

Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.

Xie, Y. (2020). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.20.