

# Reporting with Data in R

*Christian McDonald*

*2019-01-08*



# Contents

<b>1</b>	<b>About this class</b>	<b>5</b>
	About the author . . . . .	5
<b>2</b>	<b>Introduction to R</b>	<b>7</b>
<b>3</b>	<b>Importing data</b>	<b>9</b>
<b>4</b>	<b>Data manipulation</b>	<b>11</b>
<b>5</b>	<b>Data types</b>	<b>13</b>
<b>6</b>	<b>Aggregation</b>	<b>15</b>
<b>7</b>	<b>Tidy data</b>	<b>17</b>
<b>8</b>	<b>Graphics</b>	<b>19</b>
<b>9</b>	<b>Census</b>	<b>21</b>
<b>10</b>	<b>Joins and merges</b>	<b>23</b>
<b>11</b>	<b>Data packages</b>	<b>25</b>
<b>12</b>	<b>Maps</b>	<b>27</b>



# Chapter 1

## About this class

This collection of lessons is intended to support the class Reporting With Data, taught by me, Christian McDonald, at the School of Journalism, Moody College of Communication, University of Texas at Austin.

I'm a strong proponent of Scripted Journalism, a method of committing data-centric journalism in a programmatic, repeatable and transparent way. There are a myriad of programming languages that further this, including Python (pandas and Jupyter) and JavaScript (Observable), but we'll be using R, RMarkdown and RStudio.

R is a super powerful, open-source programming language for data that is deep with features and an awesome community of users who build upon it. No matter the challenge before you in your data storytelling, there is probably a package available to help you solve that challenge. Probably more than one.

There is always more than one way to do things in R. This course is an opinionated collection of lessons intended to teach students new to R and programming for the expressed act of committing journalism. As a beginner course, I strive to make it as simple as possible, which means I may not go into detail about alternative (and possibly better) ways to accomplish tasks.

## About the author

I'm a career journalist who most recently served as Data and Projects Editor at the Austin American-Statesman before coming to the University of Texas at Austin full-time in Fall 2018. I've taught data-related course at UT since 2013.

- UT Github: [utdata](#)
- Github: [critmcdonald](#)
- Twitter: [crit](#)
- Email: [christian.mcdonald@utexas.edu](mailto:christian.mcdonald@utexas.edu)



## Chapter 2

# Introduction to R

You can label chapter and section titles using `{#label}` after them, e.g., we can reference Chapter 2. If you do not manually label them, there will be automatic labels anyway, e.g., Chapter ??.

Testing.





## Chapter 3

# Importing data

Lesson about imports, data frames, embedded data.



## Chapter 4

# Data manipulation

About using dplyr to filter, sort and manipulate data.



## Chapter 5

# Data types

About dealing with dates and other data types.



## Chapter 6

# Aggregation

About aggregation, creating new columns, etc.





## Chapter 7

# Tidy data

About shaping data with `tidyr`.



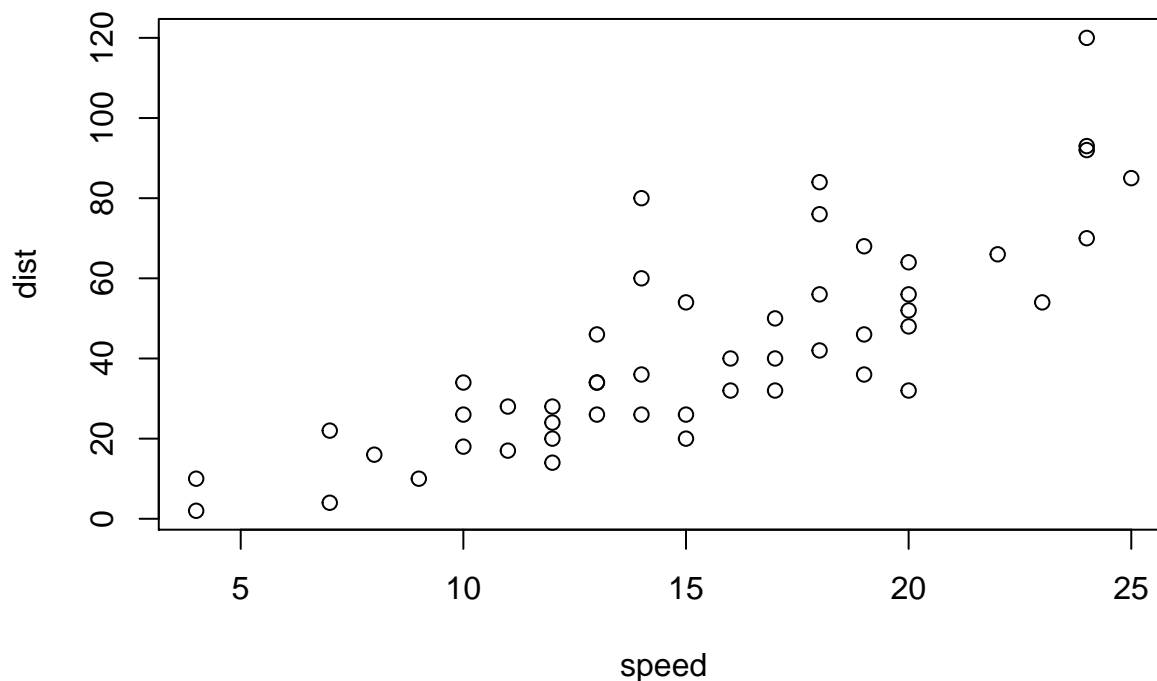
## Chapter 8

# Graphics

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

```
plot(cars)
```



Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Cmd+Option+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Cmd+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.



## Chapter 9

# Census

A mini project using census data.



## Chapter 10

# Joins and merges

About joins, merges and the like.





## Chapter 11

# Data packages

About various data packages and such.



## Chapter 12

# Maps

About making maps.

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)
```

Reference a figure by its code chunk label with the **fig:** prefix, e.g., see Figure 12.1. Similarly, you can reference tables generated from `knitr::kable()`, e.g., see Table 12.1.

```
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, 2018) in this sample book, which was built on top of R Markdown and **knitr** (Xie, 2015).

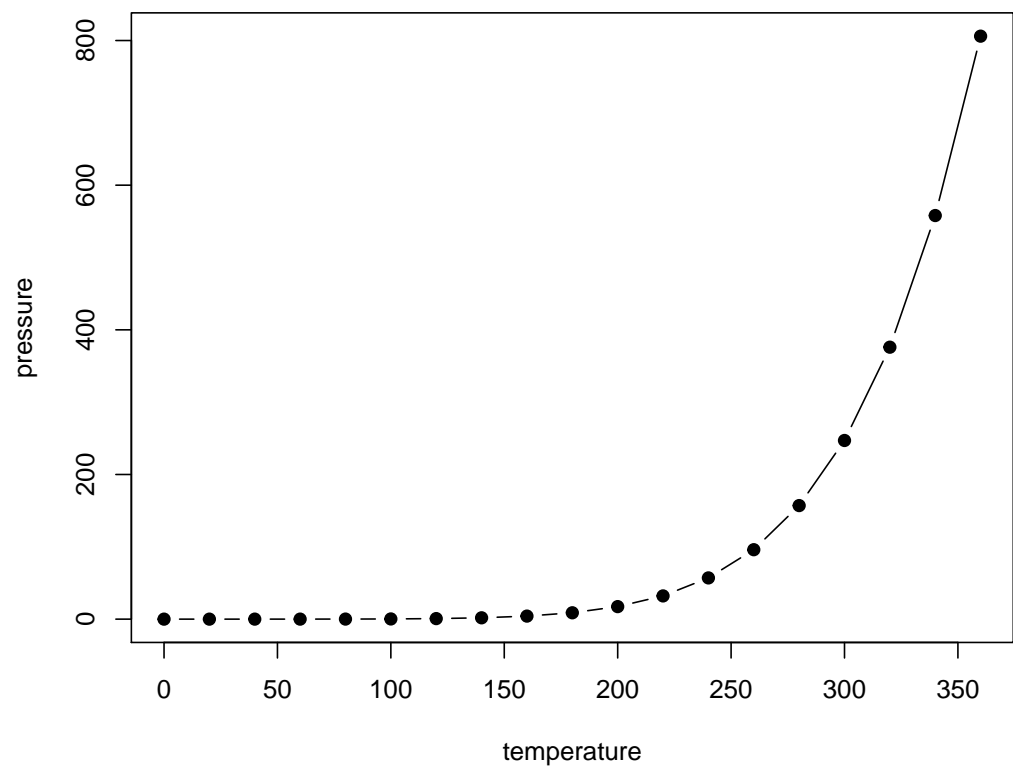


Figure 12.1: Here is a nice figure!

Table 12.1: 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

# Bibliography

- Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.
- Xie, Y. (2018). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.9.