Angela M. Zoss, Ph.D.

Visualization for Data Science with R

To my family.

I'm so grateful for your support.

Contents

List of Tables

List of Figures

Proposal

Note: This book is a work in progress, with a full draft expected in April of 2022.

This book combines instruction on writing R code with building basic graphic design skills in a way that is unusual in data science literature. The book will guide readers through a series of projects, each designed to cover both how visualizations work in R and how visualizations can be designed to have the greatest impact. Far more than a "do this, then this" checklist, this book will focus on building understanding, confidence, and the ability to transfer skills to other tools and design contexts. It will avoid technical jargon that our target audience is unlikely to have encountered before. To accommodate learners who don't have time to work through an entire book, each chapter will operate independently, covering a specific set of tasks that all make sense together as part of a visualization project. For those who would like extra practice, there will be several types of hands-on exercises, from those that are entirely prescribed to those that allow readers to apply new techniques to problems in their own areas.

The book will have solutions (in the form of completed code and sample output) for all exercises. While not a textbook, the book will also include a brief teacher's guide for courses that might want to use one or more chapters to structure lessons in a course. The book will also have a website, including links to Open Access content, solutions, and related resources like video tutorials.

The target audience of this book would be professionals who are having to learn data science techniques on the job, likely at an under-resourced organization or company. These newly minted data professionals may feel comfortable in Excel but have only just started to learn R for processing data. They have never used a programming language to build a visualization before, and even creating charts in Excel has often been a frustrating and mystifying process. They appreciate that R is freely available and are able to get started on a data science project, but the idea of creating publication-quality visualizations using only code is daunting.

Increasingly, programs of study with a focus on preparing students for professional careers in under-resourced fields, like public policy and even management, include courses on data analysis and communication using freely available software. This book, while not a textbook, could easily be used for a semester-long course, titled something like "Practical data visualization for

the modern workforce." A chapter could be covered each week, and larger projects could help learners synthesize chapters into a complete set of analyses and communication materials.

Why read this book

This book will be:

- Written for non-academics, beginning programmers
- Each chapter stands alone
- Covers pressing modern issues, like accessibility and ethics
- Focuses on freely available software
- Combines hands-on exercises with basic graphic design principles

Structure of the book

- Chapter 1: Overview of common visualizations and how to read them
- Chapter 2: Building basic visualizations with ggplot2
- Chapter 3: Working with textual data in ggplot2
- Chapter 4: Customizing the design of ggplot2 visualizations
- Chapter 5: Avoiding unethical design practices
- Chapter 6: Building ggplot2 visualizations into print publications
- Chapter 7: Basic accessibility for static visualizations
- Chapter 8: Exploring interactivity in visualizations with plotly and crosstalk
- Chapter 9: Using RMarkdown to build websites for projects
- Chapter 10: Using RMarkdown to build dashboards for projects
- Chapter 11: Basic usability for interactive visualizations
- Chapter 12: Teacher's guide

Software information and conventions

I used the **knitr** package (Xie, 2015) and the **bookdown** package (Xie, 2020) to compile my book. My R session information is shown below:

```
xfun::session_info()
```

```
## R version 4.0.3 (2020-10-10)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur 10.16
##
## Locale: en_US.UTF-8 / en_US.UTF-8 / en_US.UTF-8 / C / en_US.UTF-
8 / en_US.UTF-8
##
## Package version:
##   base64enc_0.1.3   bookdown_0.21
##   compiler_4.0.3    digest_0.6.27
##   evaluate_0.14     glue_1.4.2
##   graphics_4.0.3    grDevices_4.0.3
##   highr_0.8         htmltools_0.5.1.1
##   jsonlite_1.7.2    knitr_1.31
##   magrittr_2.0.1    markdown_1.1
##   methods_4.0.3     mime_0.9
##   rlang_0.4.10      rmarkdown_2.6
##   rstudioapi_0.13   stats_4.0.3
##   stringi_1.5.3     stringr_1.4.0
##   tinytex_0.31      tools_4.0.3
##   utils_4.0.3       xfun_0.20
##   yaml_2.2.1
```

Package names are in bold text (e.g., **rmarkdown**), and inline code and filenames are formatted in a typewriter font (e.g., `knitr::knit('foo.Rmd')`). Function names are followed by parentheses (e.g., `bookdown::render_book()`).

Angela Zoss

About the Author

FIGURE 1: Angela M. Zoss, Ph.D.

Angela is the Assessment & Data Visualization Analyst[1] in the Assessment & User Experience Department[2] in the Duke University Libraries[3]. She has many years of experience in teaching and training, predominantly focusing on teaching data visualization to university students, faculty, and staff. She is also active in several open source development projects, including FOLIO[4] and Wax[5].

[1] https://library.duke.edu/about/directory/staff/angela.zoss
[2] https://library.duke.edu/about/depts/assessment-user-experience
[3] https://library.duke.edu/
[4] https://github.com/folio-org/
[5] https://github.com/minicomp/wax

1

Overview of common visualizations and how to read them

sample text

We have a nice figure in Figure 1.1, and also a table in Table 1.1.

```
par(mar = c(4, 4, 1, .1))
plot(cars, pch = 19)
```

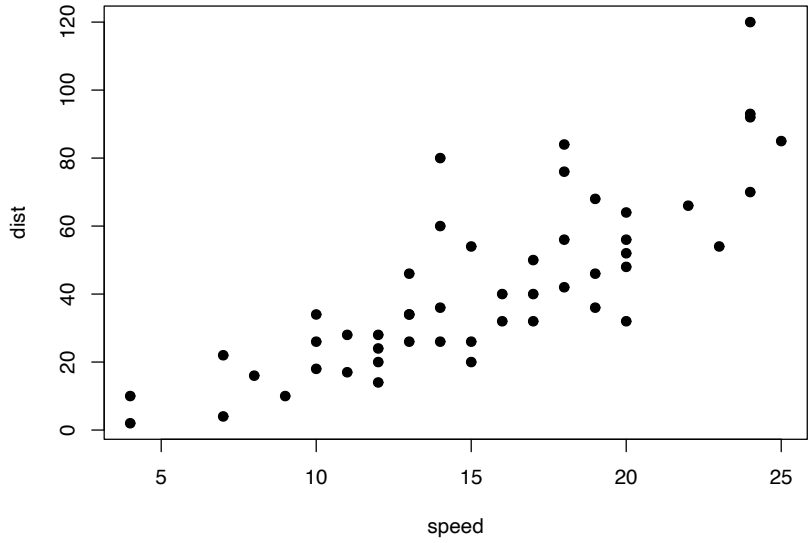


FIGURE 1.1: Hello World!

```
knitr::kable(
  head(iris), caption = 'The boring iris data.',
  booktabs = TRUE
)
```

TABLE 1.1: The boring iris data.

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

More chapters to come in `02-foo.Rmd`, `03-bar`.Rmd, …

2

Building basic visualizations with ggplot2

sample text

We talk about the *FOO* method in this chapter.

3

Working with textual data in ggplot2

sample text

Cleaning data: use duke_enrollment (either by status or school) to talk about factors. Have Semester, which is really a time-based variable. Need to combine with Year to get the real sequence of enrollment.

4

Customizing the design of ggplot2 visualizations

sample text

We talk about the *FOO* method in this chapter.

5

Avoiding unethical design practices

sample text

We talk about the *FOO* method in this chapter.

6

Building ggplot2 visualizations into print publications

sample text

We talk about the *FOO* method in this chapter.

7

Basic accessibility for static visualizations

sample text

We talk about the *FOO* method in this chapter.

8

Exploring interactivity in visualizations with plotly and crosstalk

sample text

We talk about the *FOO* method in this chapter.

9

Using RMarkdown to build websites for projects

sample text

We talk about the *FOO* method in this chapter.

10

Using RMarkdown to build dashboards for projects

sample text

We talk about the *FOO* method in this chapter.

11

Basic usability for interactive visualizations

sample text

We talk about the *FOO* method in this chapter.

12

Teacher's guide

sample text

We talk about the *FOO* method in this chapter.

A

Datasets

Duke Enrollment

Duke enrollment[1]

```
duke_enrollment <- read_csv('data/duke_enrollment/UberMegaMaster_70S-20F-V5.csv') %>%
  dplyr::select(-X1, -X.2, -X.1, -X)
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
##
## -- Column specification --------------------------------
## cols(
##   .default = col_double(),
##   Semester = col_character(),
##   Origin = col_character(),
##   Sex = col_character(),
##   Region = col_character()
## )
## i Use `spec()` for the full column specifications.
```

```
duke_enrollment_by_status <- duke_enrollment %>%
  dplyr::select(Year, Semester, Origin, Region, Sex, All_UG, All_Grad) %>%
  rename("Undergraduate" = "All_UG", "Graduate" = "All_Grad") %>%
  pivot_longer(cols = c(Undergraduate, Graduate),
               names_to = "Student_Status",
               values_to = "Count") %>%
  dplyr::filter(Count > 0)
```

[1] https://doi.org/10.7924/r4db82p1j

```
duke_enrollment_by_school <- duke_enrollment %>%
  dplyr::select(Year, Semester, Origin, Region, Sex, Trinity, Nursing,
                Engineering, Graduate, Divinity, Law, Business, Environment,
                Medicine, Graduate.Nursing, Allied.Health) %>%
  pivot_longer(cols = c(Trinity, Nursing, Engineering, Graduate, Divinity, Law,
                        Business, Environment, Medicine, Graduate.Nursing,
                        Allied.Health), names_to = "School", values_to = "Count") %>%
  dplyr::filter(Count > 0)

write_csv(duke_enrollment_by_status, "data/duke_enrollment/duke_enrollment_by_status.csv")
write_csv(duke_enrollment_by_school, "data/duke_enrollment/duke_enrollment_by_school.csv")
```

Coral Resilience Data

Protecting coral reefs[2]

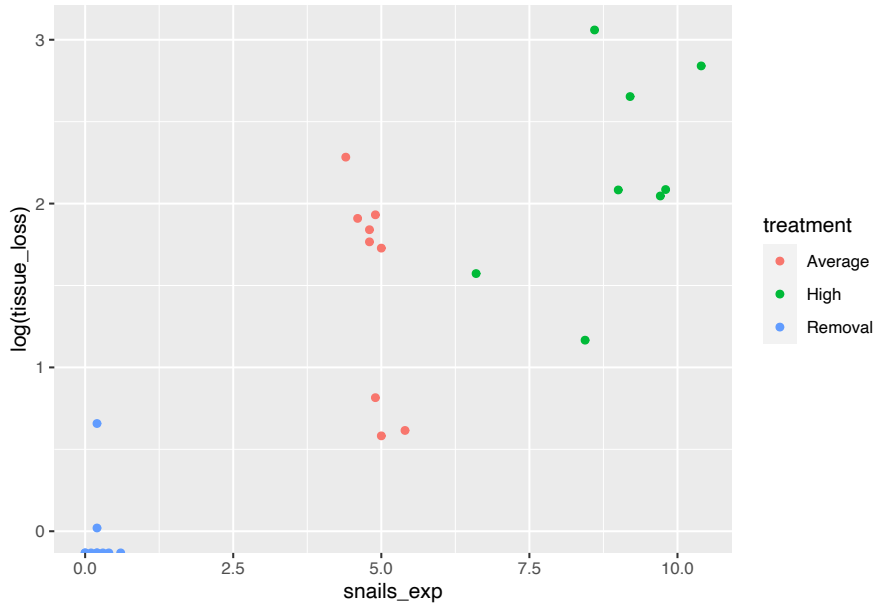
```
coral_resilience <- read_csv('data/coral_resilience/Raw_Data_Experiment.csv')
```

```
##
## -- Column specification --------------------------------
## cols(
##   coral_id = col_double(),
##   circumf_1 = col_double(),
##   circumf_2 = col_double(),
##   coral_surfarea = col_double(),
##   temp_avg = col_double(),
##   treatment = col_character(),
##   snails_exp = col_double(),
##   tissue_loss = col_double(),
##   snails_bleaching = col_double(),
##   bleaching = col_double(),
##   mort_postbleach = col_double()
## )
```

[2] https://doi.org/10.7924/G8348HFP

```
# chart from p. 68 of https://hdl.handle.net/10161/16819
#   (also published in https://doi.org/10.1038/s41559-018-0589-0)
ggplot(coral_resilience, aes(x=snails_exp, y=log(tissue_loss), color=treatment)) + geom_point()
```

```
## Warning: Removed 1 rows containing missing values
## (geom_point).
```




```
# Could make a map with reef coordinates from the thesis (p.46):

#Site Name,Protection Status,Florida Keys Region,GPS Location
#Molasses Reef,SPA/ No-take,Upper Keys,N 25 0.579, W 80 22.471
#Conch Reef,SPA/ No-take,Upper Keys,N 24 57.110, W 80 27.564
#Coffins Patch,SPA/ No-take,Middle Keys,N 24 41.400, W 80 57.850
#Pickles Reef,Fishing allowed,Upper Keys,N 24 59.170, W 80 24.940
#Horseshoe Reef,Fishing allowed,Middle Keys,N 24 39.670, W 80 59.650
#American Shoals,Fishing allowed,Lower Keys,N 24 31.568, W 81 31.383

# but all of the experiment data comes from multiple colonies in one reef.
# p. 46 of thesis:

# "This experiment was conducted at Pickles Reef in Key Largo, Florida,
```

```
# USA from mid-June to mid-August 2014. Within the experimental site, I selected
# 30 similarly-sized colonies (~159.1 ± 15.5 cm 2 surface area) that were already
# harboring C. abbreviata."
```

Git Experience

A Behavioral Approach to Understanding the Git Experience[3]

```
git_experience <- read_csv('data/git_experience/2020-07-12_processed-survey.csv') %>%
  dplyr::select(-X1) %>%
  mutate(year_vcs = as.numeric(year_vcs),
         across(.cols = c(first_bazaar, first_cvs, first_git, first_hg,
                          first_monotone, first_svn, first_other, use_bitbucket,
                          use_github, use_gitlab, use_sourceforge, use_selfhost,
                          use_nothing, use_other_platform, use_local_gui,
                          use_local_term, use_local_other, how_learn_books,
                          how_learn_credit_course, how_learn_online_course,
                          how_learn_rtfm, how_learn_accel, how_learn_webinar,
                          how_learn_workshop, how_learn_other, teach_inperson,
                          teach_vasync, teach_vsync, use_ci, use_annotation,
                          use_fork_pr, use_issues, use_pages, use_boards,
                          use_wikis, use_other_feat, private_fund, public_fund,
                          dontknow_fund, no_funds, other_fund, scholexp_collab,
                          scholexp_edu, scholexp_method, scholexp_peerprod,
                          scholexp_peer_review, scholexp_pub, scholexp_qa,
                          scholexp_repro, scholexp_other, archive_figshare,
                          archive_ir, archive_osf, archive_sh, archive_zenodo,
                          archive_other),
                as.logical))
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
##
## -- Column specification --------------------------------
## cols(
##   .default = col_double(),
```

[3] https://osf.io/57tb8/

```
##   year_vcs = col_character(),
##   first_vcs = col_character(),
##   first_vcs_other = col_character(),
##   first_other_text = col_character(),
##   use_selfhost_text = col_character(),
##   use_other_platform_text = col_character(),
##   why_no_platform = col_character(),
##   freq_git = col_character(),
##   freq_git_text = col_character(),
##   freq_platform = col_character(),
##   freq_platform_text = col_character(),
##   use_local_other_text = col_character(),
##   why_vcs = col_character(),
##   why_vcs_text = col_character(),
##   how_learn_other_text = col_character(),
##   who_taught_git = col_character(),
##   when_learn_git = col_character(),
##   when_learn_git_other = col_character(),
##   freq_reteach = col_character(),
##   freq_reteach_text = col_character()
##   # ... with 27 more columns
## )
## i Use `spec()` for the full column specifications.

## Warning in mask$eval_all_mutate(quo): NAs introduced by
## coercion
```

Inclusiveness Index

Inclusiveness Index[4]

```
library(readxl)
```

```
inclusiveness_index <- read_excel('data/inclusiveness_index/global_data_for_website_2020.xlsx', na
```

[4] https://belonging.berkeley.edu/inclusivenessindex/data-and-resources

Candidate Demographics

Candidate Demographics[5]

```
#https://wholeads.us/research/rising-tide-ballot-demographics/ - includes State, Candidate Name, C

candidate_demographics <- bind_rows(
  read_excel('data/candidate_demographics/RD-Candidate-Analysis-2012-8.xlsx', sheet=1) %>%
    mutate(year = 2018),
  read_excel('data/candidate_demographics/RD-Candidate-Analysis-2012-8.xlsx', sheet=2) %>%
    mutate(year = 2016),
  read_excel('data/candidate_demographics/RD-Candidate-Analysis-2012-8.xlsx', sheet=3) %>%
    mutate(year = 2014),
  read_excel('data/candidate_demographics/RD-Candidate-Analysis-2012-8.xlsx', sheet=4) %>%
    mutate(year = 2012)
    )
```

Affinity Spending

Affinity Spending[6]

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

[5] https://wholeads.us/research/rising-tide-ballot-demographics/
[6] https://github.com/OpportunityInsights/EconomicTracker

```
affinity_spending <- read_csv('data/affinity_spending/Affinity - National - Daily.csv', na=".") %>%
  mutate(date = ymd(paste(year, month, day, sep="-"))) %>%
  dplyr::select(-starts_with("spend_s_")) %>%
  pivot_longer(cols =  starts_with("spend_"), names_to = "spending_category", values_to = "spendin
  mutate(spending_category = str_replace(str_remove(spending_category, "spend_"), "_q", "|q")) %>%
  separate(spending_category, into=c("spending_category", "income_quartile"), sep="[|]") %>%
  dplyr::filter(!is.na(spending_proportion) &
                  !is.na(income_quartile) &
                  spending_category %in% c("retail_w_grocery", "retail_no_grocery",
                                           "durables", "nondurables",
                                           "remoteservices", "inpersonmisc")) %>%
  dplyr::select(date, income_quartile, spending_category,
                spending_proportion, freq, provisional)
```

```
##
## -- Column specification --------------------------------
## cols(
##   .default = col_double(),
##   freq = col_character()
## )
## i Use `spec()` for the full column specifications.

## Warning: Expected 2 pieces. Missing pieces filled with
## `NA` in 15894 rows [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,
## 12, 13, 14, 15, 16, 17, 18, 87, 88, ...].
```

```
ggplot(affinity_spending, aes(x=date, y=spending_proportion, color=income_quartile)) +
  geom_line() +
  facet_wrap(vars(spending_category))
```


durables
inpersonmisc
nondurables
remoteservices
retail_no_grocery
retail_w_grocery
spending_proportion
0.6
0.3
0.0
-0.3
2020-01
2020-07
2021-01
2021-07
date
income_quartile
q1
q2
q3
q4

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 0.21.

Index