Statistical Computing in R

Contents

1 Statistical computing in R

1.1 Online R learning resources

There are an overwhelming number of great resources for learning R; here are some recommendations:

- *The RStudio Education website*[1], especially:
  - *Finding your way to R*[2]
- *R for Epidemiology* (Cannell and Livingston (2024))
- *The Epidemiologist R Handbook* (Batra (2024))
- *Practical R for Epidemiologists* (Myatt (2022))
- *R for Data Science* (Wickham, Çetinkaya-Rundel, and Grolemund (2023))
- *Advanced R* (Wickham (2019))
- *R Graphics Cookbook* (Chang (2024))
- *R Packages* (Wickham and Bryan (2023))
- Nahhas (2023) (same author as Nahhas (2024))
- Myatt (2022)
- Aragon (2017) (previously Aragon (2013)): Author is State Public Health Officer and Director, California Department of Public Health, https://drtomasaragon.github.io/)

[1] https://education.rstudio.com
[2] https://education.rstudio.com/learn/

- *SAS and R* (Kleinman and Horton (2009))
- The "sassy system"[3] is "an integrated set of packages designed to make programmers more productive in R, particularly those with a background in SAS® software. The system leverages useful concepts and thought patterns to create a more efficient and satisfactory R programming experience."
  - In particular, the procs[4] package in R provides versions of common SAS procedures, such as 'proc freq', 'proc means', 'proc ttest', 'proc reg', 'proc transpose', 'proc sort', and 'proc print'
- *R for SAS and SPSS users* (Muenchen (2011))
- *Building reproducible analytical pipelines with R* (Rodrigues (2023))
- *Posit Recipes: Some tasty R code snippets*: https://posit.cloud/learn/recipes

1.2 UC Davis R programming courses

There are several dedicated UC Davis courses on R programming:

- BIS 015L[5]: Introduction to Data Science for Biologists
  - see course materials at https://jmledford3115.github.io/datascibiol/
- ENV 224[6]/ ECL 224[7]: Data Management & Visualization in R
  - see lecture videos and course materials at https://ucd-r-davis.github.io/R-DAVIS/
- ESP 106[8]: Environmental Data Science
- STA 015B[9]: Introduction to Statistical Data Science II
- STA 032[10]: Gateway to Statistical Data Science
- STA 035A[11]: Statistical Data Science
- STA 035B[12]: Statistical Data Science II
- STA 141A[13]: Fundamentals of Statistical Data Science
- STA 242[14]: Introduction to Statistical Programming
- ABG 250[15]: Mathematical Modeling in Biological Systems
- **PSC 203A**[16] "Data Cleaning & Management in the Social Sciences"
- PSC 203B[17] "Data Visualization in the Social Sciences"

DataLab[18] maintains another list of courses: https://datalab.ucdavis.edu/courses/

DataLab also provides short-form workshops on R programming and data science: https://datalab.ucdavis.edu/workshops/

1.3 Demographics tables

Demographics tables are important first steps in many data analyses and papers.

[3] https://r-sassy.org/
[4] https://cran.r-project.org/web/packages/procs/
[5] https://catalog.ucdavis.edu/search/?q=BIS+015L
[6] https://catalog.ucdavis.edu/search/?q=ENV+224
[7] https://catalog.ucdavis.edu/search/?q=ECL+224
[8] https://catalog.ucdavis.edu/search/?q=ESP+106
[9] https://statistics.ucdavis.edu/expanded-descriptions/15b
[10] https://statistics.ucdavis.edu/expanded-descriptions/32
[11] https://statistics.ucdavis.edu/expanded-descriptions/35A
[12] https://statistics.ucdavis.edu/expanded-descriptions/35B
[13] https://statistics.ucdavis.edu/expanded-descriptions/141A
[14] https://statistics.ucdavis.edu/expanded-descriptions/242
[15] https://catalog.ucdavis.edu/search/?q=ABG+250
[17] https://catalog.ucdavis.edu/search/?q=PSC+203B
[18] https://datalab.ucdavis.edu/

The `gtsummary` package is flexible and can probably provide whatever table options you're looking for, and if not, the developers are usually very welcoming of feature requests.

If `gtsummary` is really not doing what you want, other packages I've used for demographics tables include:

- https://cran.r-project.org/web/packages/procs/ (replicates common SAS commands)
- https://cran.r-project.org/web/packages/arsenal/index.html (from the Mayo Clinics)
- https://cran.r-project.org/web/packages/table1/index.html

1.4 Writing functions

- Read this ASAP: https://r4ds.hadley.nz/functions.html
- Use this as a reference: https://adv-r.hadley.nz/functions.html

1.4.1 Methods versus functions

See https://adv-r.hadley.nz/oo.html#oop-systems

1.4.2 Debugging code

- https://adv-r.hadley.nz/debugging.html
- https://www.maths.ed.ac.uk/~swood34/RCdebug/RCdebug.html

1.5 `data.frames` and `tibbles`

1.5.1 Displaying `tibbles`

See `vignette("digits", package = "tibble")`

1.6 The `tidyverse`

> The tidyverse is an opinionated collection of R packages designed for data science. All packages share an underlying design philosophy, grammar, and data structures.

- https://www.tidyverse.org/

These packages are being actively developed by Hadley Wickham[19] and his colleagues at posit[20][21].

Details:

- Wickham et al. (2019)
- Wickham, Çetinkaya-Rundel, and Grolemund (2023)
- Kuhn and Silge (2022)

1.7 Piping

See Wickham, Çetinkaya-Rundel, and Grolemund (2023)[23] for details.

There are currently (2025) two commonly-used pipe operators in R:

- `%>%`: the "`magrittr` pipe", from the `magrittr`[24] package (Bache and Wickham (2022); re-exported[25] by `dplyr`[26] and others) .
- `|>`: the "native pipe", from base R ($\geq$ 4.1.0)

See https://www.tidyverse.org/blog/2023/04/base-vs-magrittr-pipe for a comparison of their behavior.

[19] https://hadley.nz/
[20] https://posit.co/
[21] the company formerly known as RStudio[22]
[23] https://r4ds.hadley.nz/data-transform.html#sec-the-pipe
[24] https://cran.r-project.org/web/packages/magrittr/index.html
[25] https://r-pkgs.org/dependencies-in-practice.html#re-exporting
[26] https://cran.r-project.org/web/packages/dplyr/index.html

1.7.1 Which pipe should I use?

Wickham, Çetinkaya-Rundel, and Grolemund (2023) recommends the native pipe[27]:

> For simple cases, |> and %>% behave identically. So why do we recommend the base pipe? Firstly, because it's part of base R, it's always available for you to use, even when you're not using the tidyverse. Secondly, |> is quite a bit simpler than %>%: in the time between the invention of %>% in 2014 and the inclusion of |> in R 4.1.0 in 2021, we gained a better understanding of the pipe. This allowed the base implementation to jettison infrequently used and less important features.

1.7.2 Why doesn't `ggplot2` use piping?

Here's `tidyverse` creator Hadley Wickham's answer (from 2018):

> I think it's worth unpacking this question into a few smaller pieces:
>
> - Should ggplot2 use the pipe? IMO, yes.
> - Could ggplot2 support both the pipe and plus? No
> - Would it be worth it to create a ggplot3 that uses the pipe? No.

https://forum.posit.co/t/why-cant-ggplot2-use/4372/7

1.8 Grouping operations in `dplyr`

The `dplyr` package provides two approaches for grouping data:

- **Persistent grouping** with `group_by()`: Creates a grouped data frame that remains grouped for subsequent operations until explicitly ungrouped
- **Per-operation grouping** with the `.by` argument: Applies grouping for a single operation only, without modifying the data frame structure

Recommendation: Default to using per-operation grouping with the `.by` argument, as it is more explicit, reduces the risk of accidentally operating on grouped data, and eliminates the need to remember to `ungroup()`.

For a detailed comparison of these approaches, see `?dplyr_by`[28].

1.9 Quarto

Quarto is a system for writing documents with embedded R code and/or results:

- Read this ASAP: https://r4ds.hadley.nz/communicate
- Then use this for reference: https://quarto.org/docs/reference/
- Learn LaTeX in 30 minutes (not everything in here is relevant to Quarto): https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes
- LaTeX symbol reference guide: https://oeis.org/wiki/List_of_LaTeX_mathematical_symbols
- LaTeX commands: https://www.overleaf.com/learn/latex/Commands

To compile Quarto documents to pdf, run these commands first:

```
install.packages("tinytex")
tinytex::install_tinytex()
```

See Knuth (1984) for additional discussion of literate programming.

1.10 One source file, multiple outputs

One of quarto's excellent features is the ability to convert the same source file into multiple output formats; in particular, I am using the same set of source files to generate an html website, a pdf document, and a set of revealjs slide decks.

[27] https://r4ds.hadley.nz/data-transform.html#sec-the-pipe:~:text=So%20why%20do%20we%20recommend%20the%20base%20pipe%3F

[28] https://dplyr.tidyverse.org/reference/dplyr_by.html

I use `::: notes` divs to mark text chunks to omit from the revealjs format but include in the website and pdf format.

1.11 Packages

> This book espouses our philosophy of package development: anything that can be automated, should be automated. Do as little as possible by hand. Do as much as possible with functions. The goal is to spend your time thinking about what you want your package to do rather than thinking about the minutiae of package structure.

- https://r-pkgs.org/introduction.html#:~:text=This%20book%20espouses,of%20package%20structure.
- Read this ASAP: https://r-pkgs.org/whole-game.html
- Use the rest of Wickham and Bryan (2023) as a reference

1.12 Submitting packages to CRAN

- Read this first: https://r-pkgs.org/release.html
- A problems-and-solutions book is under construction: https://contributor.r-project.org/cran-cookbook/

1.13 Git

94% of respondents to a 2022 Stack Overflow survey[29] reported using git for version control.

More details[30]

- *Happy Git with R* https://happygitwithr.com/
- https://usethis.r-lib.org/articles/pr-functions.html
- *Git Magic* http://www-cs-students.stanford.edu/~blynn/gitmagic/
- https://ohshitgit.com/
- https://maelle.github.io/saperlipopette/

1.14 Spatial data science

- Pebesma and Bivand (2023)

1.15 Shiny apps

- Read Wickham (2021) first
- Use Fay et al. (2021) as a reference

1.16 Making the most of RStudio

Over time, explore all the tabs and menus; there are a lot of great quality-of-life features.

- use the `History` tab to view past commands; you can rerun them or copy them into a source code file in one click! (up-arrow in the Console also enables this process, but less easily).

1.17 Contributing to R

Many modern R packages are developed on Github, and welcome bug reports and pull requests (suggested edits to source code) through the Github interface.

To contribute to "base R" (the core systems), see https://contributor.r-project.org/

[29] https://survey.stackoverflow.co/2022/#section-version-control-version-control-systems
[30] https://r-pkgs.org/software-development-practices.html#sec-sw-dev-practices-git-github

Aragon, Tomas J. 2013. *Applied Epidemiology Using R*. Online. https://tbrieder.org/epidata/course_reading/e_aragon.pdf.

———. 2017. *Population Health Data Science with R: Transforming Data into Actionable Knowledge*. Online. https://bookdown.org/medepi/phds/.

Bache, Stefan Milton, and Hadley Wickham. 2022. *Magrittr: A Forward-Pipe Operator for r*. https://CRAN.R-project.org/package=magrittr.

Batra, Neale, ed. 2024. *The Epidemiologist R Handbook*. Online. https://www.epirhandbook.com/.

Cannell, Brad, and Melvin Livingston. 2024. *R for Epidemiology*. Online. https://www.r4epi.com/.

Chang, Winston. 2024. *R Graphics Cookbook: Practical Recipes for Visualizing Data*. O'Reilly Media. https://r-graphics.org/.

Fay, Colin, Sébastien Rochette, Vincent Guyader, and Cervan Girard. 2021. *Engineering Production-Grade Shiny Apps*. Chapman; Hall/CRC. https://engineering-shiny.org/.

Kleinman, Ken, and Nicholas J Horton. 2009. *SAS and r: Data Management, Statistical Analysis, and Graphics*. Chapman; Hall/CRC. https://www.routledge.com/SAS-and-R-Data-Management-Statistical-Analysis-and-Graphics-Second-Edition/Kleinman-Horton/p/book/9781466584495.

Knuth, Donald E. 1984. "Literate Programming." *Comput. J.* 27 (2): 97–111. https://doi.org/10.1093/comjnl/27.2.97.

Kuhn, Max, and Julia Silge. 2022. *Tidy Modeling with r*. " O'Reilly Media, Inc.". https://www.tmwr.org/.

Muenchen, Robert A. 2011. *R for SAS and SPSS Users*. Springer Science & Business Media. https://link.springer.com/book/10.1007/978-1-4614-0685-3.

Myatt, Mark. 2022. *Practical R for Epidemiologists*. Online. https://practical-r.org/index.html.

Nahhas, Ramzi W. 2023. *An Introduction to r for Research*. https://bookdown.org/rwnahhas/IntroToR/.

———. 2024. *Introduction to Regression Methods for Public Health Using R*. CRC Press. https://www.bookdown.org/rwnahhas/RMPH/.

Pebesma, Edzer, and Roger Bivand. 2023. *Spatial Data Science: With Applications in R*. Boca Raton: Chapman; Hall/CRC. https://doi.org/10.1201/9780429459016.

Rodrigues, Bruno. 2023. *Building Reproducible Analytical Pipelines with r*. Online. https://raps-with-r.dev/.

Wickham, Hadley. 2019. *Advanced r*. Chapman; Hall/CRC. https://adv-r.hadley.nz/index.html.

———. 2021. *Mastering Shiny*. " O'Reilly Media, Inc.". https://mastering-shiny.org/.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Wickham, Hadley, and Jennifer Bryan. 2023. *R Packages*. O'Reilly Media, Inc. https://r-pkgs.org/.

Wickham, Hadley, Mine Çetinkaya-Rundel, and Garrett Grolemund. 2023. *R for Data Science*. " O'Reilly Media, Inc.". https://r4ds.hadley.nz/.