Statistical Computing

Arni Magnusson

Statistical Modeling in R
Universidad de Concepción
19–23 January 2026

Outline

Statistical Computing and R
features, history

Open Science
software, scientific method, repeatability, reviewability

First Steps in R
calculator, objects, plots, help

Statistical computing

GUI	Excel, JMP, LibreOffice, Minitab, SPSS, Stata
Interpreted	BUGS, Julia, Matlab, Python, **R**, SAS
Compiled	C/C++, Fortran, Java
Related	Bash, HTML, LaTeX, Make, SQL
Editor	Emacs, Positron, **RStudio**, VS Code

Spreadsheets

Excel and LibreOffice Calc can be useful for:

- Taking notes in an environment where you can calculate summary statistics
  family trip, project management
- Simple statistical analysis, if spreadsheet is the only tool you know

Compared to statistical software, spreadsheets are:

- Limited, with few features
- Unreliable, can produce the wrong result
- Error-prone, easy to make mistakes without noticing
- Unwieldy, difficult to work with many tables and sheet references
- Can be difficult to review or repeat analysis for another dataset

Use only `+ - / *`, `sum`, `average`, and statistical software for everything else

Using the right tool

Imagine writing a 20-page text document in Excel

$\Rightarrow$ inferior quality, hard to modify, prone to errors

Likewise, R is not always the right tool in statistical computing:

Databases for large amounts of data

C++ for computationally intensive subtasks

R has good support for connecting to databases and running compiled C++ code

R features

Large collection of tools for statistical analysis, constantly updated by a large user commity, including leading authors in statistical fields

Graphics for exploratory analysis and publications

Language for expressing statistical models, object-oriented and extensible by users

Used by university statistics departments and research institutes around the world

R early history

S

Programming language, first released in 1976
Created by John Chambers et al., Bell Laboratories

S-Plus

Statistical software based on S, first released in 1988
Created by R Douglas Martin, Univ Washington

R

Statistical software based on S, first released in 1993
Created by Ross Ihaka and Robert Gentleman, Univ Auckland
Maintained by R Core Development Team since 1997
R became better than S-Plus around 2002

RStudio and tidyverse

ggplot2

Plotting package, first released in 2007
Created by Hadley Wickham

RStudio

Development environment, first released in 2011
Created by J.J. Allaire and Joe Cheng

dplyr

Data manipulation package, first released in 2014
Created by Hadley Wickham and Romain François

R for Data Science

Influential textbook, first published in 2017
Written by Hadley Wickham and Garrett Grolemund

R Project website

https://www.r-project.org

Download R, manuals, etc.


OPEN
SCIENCE

Open source

Most R functions are written in the R language, and the full code is shown if you type the name of the function

Low-level functions are written in C, and the full code can be browsed at `https://svn.r-project.org/R/trunk/`

This access to the source code is of critical value for complex statistical models

Open source principles (making a thorough description of methods publicly available) have been a foundation of scientific research for centuries

Repeatable and reviewable research

An Excel file containing a complex analysis may not allow other scientists to apply the method to another dataset, or to understand the different steps in the analysis

R makes a clear separation between data and calculations, and the computations follow a logical order: first this, then that, etc.

We try to write code that is easy for others to understand

Scientific method

Open source statistical software has become a cornerstone of scientific inference, and is a modern element of the scientific method: medical research, astronomy, everywhere.

The software development process is a collaborative effort of scientists worldwide, and relies on users contributing code, documentation, bug reports, etc.

You can contribute to open source software

First steps in R

Install R

`https://www.r-project.org`

Install RStudio (or some other editor)

`https://posit.co`

Calculator with functions

```
2 + 2
```

```
sqrt(10)
```

```
log(10)
```

```
# try the up arrow
```

Objects in workspace

```
x <- 2

10 * x


ls()

rm(x)

rm(list=ls())
```

Data objects

Vectors

```
rivers
month.abb
```

Data frames

```
BOD
mtcars
```

Select column

```
mtcars$hp
```

Plots

```
x <- 1:10

y <- 3 * x


plot(x, y)


y <- 100 * x
```

The plot is not “alive”, so the y coordinates are not updated unless `plot(x, y)` is called again

Help

```
help(log)
```

```
?log
```

```
args(log)
```

Help

If you get an error message:

- press the up arrow and try to rewrite
- the error message sometimes describes the problem

If R doesn't respond to user input:

- press the `Esc` key

Outline

Statistical Computing and R
features, history

Open Science
software, scientific method, repeatability, reviewability

First Steps in R
calculator, objects, plots, help