Data Analysis with R

The basics

Sarah Teichman

What is R

R is a programming language commonly used for data analysis and statistics.

- reproducible
- free
- open-source
- large community of users and developers

Download here.

What is RStudio

RStudio is an Integrated Development Environment (IDE) for R.

- write code
- run code
- navigate files
- visualize plots
- open help files

Download here.

How to access RStudio

- locally, if you have downloaded R and RStudio on your computer
  - uses your own computer and with access to your files
  - has the computational resources and limitations of your computer
- remotely, via RStudio server
  - access via a web browser
  - often available through your institution
  - may have computational advantages
  - useful in a course setting like this one!

How to access RStudio server

- find your personal R Studio server link from this spreadsheet
- username: stamps
- password: stamps2023

RStudio organization

RStudio has a four pane layout.

- console (run single lines of code)
- editor (open and write scripts)
- environment etc. (see what objects exist in work space)
- files etc. (navigate files, view plots, open help files)

`Tools -> Global Options -> Pane Layout` to change placement of panes.

`Tools -> Global Options -> Appearance` to change design of panes.

Console

Use the **console** to run individual lines of code.

```r
5 + 384
```

```
## [1] 389
```

```r
x <- 10 # set variable with <- operator :)
y = 6 # set variable with = operator :(
x + y
```

```
## [1] 16
```

Editor

Use the **editor** for opening and writing scripts.

- for a workflow to be reproducible, all code should be written in a script (not in the console)
- in R you are working in a folder on your computer
  - `getwd()` to see (get) your working directory
  - `setwd()` to change (set) your working directory
- run code with `Run` button (and options) or `Ctrl`/`Command` + `Enter` for a single line

Environment and History

- each object saved in your working space will be in the **environment**
- **history** saves most recent lines of code
- extension: you can add a **Git** plug-in to this pane for version control through GitHub
  - here is a great resource for R and Git!

Files/Plots/Help

- use files to navigate files on your computer
- use plots to display visualizations
- use help to access help files
    - type `?` to pull up a file, for example `?sum`
    - for more extensive questions, Google is also useful!

Packages

- base functions (Base R) are automatically installed with R
  - includes mathematical operations, data manipulation, plotting, etc.
- a package is a way to store files with code, documentation, and data, and let users download and use those files
- the `tidyverse` is a suite of common data manipulation and visualization packages
  - includes `dplyr`, `ggplot`, among others

Packages

- CRAN package repository has ~20,000 packages
  - most packages available here
  - install with `install.packages("package_name")`
  - load in each R session with `library(package_name)`
- Bioconductor has ~2000 packages
- anyone can make their own package (often available to download on GitHub)

R Markdown

- in R scripts (.R), each line is evaluated unless it is a comment
  - `# this is a comment`
- in R Markdown files (.Rmd), you can combine code, output, and text
  - code in “chunks”, anything within chunk is evaluated
  - anything outside of chunk is output as text
- when compiled or “knit” .Rmd files turn into HTML, PDF, slides, webpages, etc.