R Scripts

David Gerard

2019-02-06

Learning Objectives

- R scripts
- sourcing R scripts
- Chapter 6 of RDS

Background

- We've already learned that using the same code over and over again in the form of functions is efficient and less prone to errors.
- What if we want to share code between multiple R Markdown files?

R scripts

- An R Script is a file that contains just R code (no plain text like in an R Markdown file).
- All R scripts end in ".R" (community standard) or ".r" (less standard).
- Open up an R script with CONTROL + SHIFT + N (or by using the File menu).
- You can then type R code in the R script and send it to the Console like in an R chunk (using CONTROL/COMMAND + ENTER).
- Save R scripts in the "code" folder of your project.

Using R Scripts to Reuse Code

- Suppose we want to calculate the geometric mean over and over again, in many files of our analysis.
- So far, we would copy and paste the function in every new Rmd. But we can avoid this copying and pasting by coding the geometric mean function in its own R script.
- In the code folder, my R script "geo_mean.R" looks like this:

```
# Calculates the geometric mean of a vector of numerics.
#
# x: A vector of numerics
#
# returns: The geometric mean of x
geo_mean <- function(x) {
  x %>%
    log() %>%
    mean() %>%
    exp()
}
```

- You can **source** this code (make R run the R script) by

```
source("../code/geo_mean.R")
```

- You then have access to the `geo_mean()` function in your R Markdown file:

```
library(magrittr)
geo_mean(c(1, 6, 2, 5))
```

```
## [1] 2.783
```

- **Exercise**: Write a function that takes a vector of numerics and returns `TRUE` if that vector is between 0.1 and 0.5, and `FALSE` otherwise. Save this function in an R script in your code folder. Source it in an R Markdown file in your analysis folder. Then evaluate that function with `c(0, 0.2, 0.4, 0.7)`.