

RenR 480/711: Review of last week (Sep 5 and 7, 2017)

Questions? david.montwe@ualberta.ca or isaacren@ualberta.ca

Concepts from Lecture:

We discussed scientific questions, data, and data management

Review questions

- What makes a good research question? There are three elements
- What are the two main types of data?
- What are the types of continuous data?
- Key terminology: review population, sample, unit, dependent variable, independent variable, data
- Data management: xls & csv in 1 folder; long-format tables

Labs

We got started in R and R Studio last week.

Through the labs, we learned about the following:

The R and R Studio environments:

- R is the "raw" program, while R studio is the easier-to use interface that overlays R
- R studio contains 4 main panels:
 - The R-console (the actual "raw" program)
 - The scripts-panel (has tabs)
 - Global environment panel (has tabs)
 - Help/plot panel (has tabs)

Scripts:

- Although you can type directly into the R-console panel, it is best to write code in a script in R Studio – this allows you to save your code, open it again in the future, and make any modifications in the future as necessary.

Working directory:

- Working directory means, generally:
 - Where R will look for your data tables when you read them in, e.g.: `read.csv()`
 - Where your output is saved, e.g.: `write.csv()`
- You can specify any folder as your working directory, but it is a good idea to save your script to the folder where your data is, so that they share a common working directory. For example, you can import data through `read.csv("data.csv")` if you set your working directory to the same folder where you save your script.
- There are two ways of setting your working directory:
 - Manually: First save your script to the folder where your data is (e.g. Lab 1 or Lab 2), In the menu, go to *Session > Set working directory > To Source file location*

- Coded: This can be coded with the `setwd()` command, where you specify your path within the brackets. For example: `setwd("C:/Users/Desktop/Lab2")`
- To confirm your working directory (wd) with the `getwd()` function

Functions:

- Functions are commands that allow you to perform actions in R or on your data.
- Functions always have a set of brackets behind them (you type in the brackets), e.g.: `read.csv("data.csv")`
- The functions we used so far come from the "base" package: they are already installed
- Other packages can be installed to be able to use specialized functions (they don't come installed; you have to install them yourself). We will do this soon
- If you google for certain functions, it will show you the package name in squiggly brackets: e.g. {base}

Coding:

- Each function (command) should be on a new line of code.
- Use the "enter" key to run a command if you are typing directly in the R console (which isn't as good as writing in a script, as discussed above)
- From a script, you use the "Ctrl+R" keys or "Run" button to send the commands to the R console, which spits out the answer for you.
- Common sources of typos/errors:
 - The wrong case (upper/lower case) - R is case-sensitive
 - Missing brackets
 - Missing quotation marks
- If R hasn't completed a command properly, the R console will spit out an error, or it might be waiting for more instructions. When it is waiting for more instructions, there will be no ">" symbol, but probably "+" symbol. Use the "Escape" key to tell R that you will start a new command, which will then cause it to show you the ">" symbol again
- If you want to fix a mistake in your code, it's best to fix it, then re-run your code to re-write over what you had done before

How to get help:

- Information and examples are provided in help files for almost all commands
- To access the help file, simply type a question mark symbol ("?",) before the command to pull up the help file (the lower-right panel – be sure the "help" tab is open)
 - E.g.: `?read.csv()`
 - E.g.: `?setwd.csv()`
 - E.g.: `?c()`

Importing data and naming objects:

- When you import data with the "`read.csv()`" function, you should save it to an object that you can call up later
 - E.g.: `dat = read.csv("data.csv")`
- Although we suggest "dat", you can actually name it whatever you want – just make sure your code is consistent.