(Re)Introduction to R

Kirk Bansak

January 26th, 2023

R vs. RStudio vs. R Markdown

- `R`: a programming language and software environment for statistical computing and graphics

- `RStudio`: an integrated development environment (IDE) for `R`, providing facilities and features to enhance your ability to program and execute analyses in `R`

- `R Markdown`: a file format that allows you to create well-formatted documents embedding code chunks and analyses executed in `R`

- We will use `R`, `RStudio`, and `R Markdown` altogether to complete the data exercises in this class

- Installation sequence: `R`, then `RStudio`, and then `R Markdown`

R

- R: a programming language and software environment for statistical computing and graphics

    – first distributed version dates back to 1995

    – a top 10 programming language according to common rankings (e.g. TIOBE, IEEE Spectrum, PYPL Index)

    – one of the two "premier" data science programming languages (along with Python), used widely in both academia and industry

    – Python is a general-purpose programming language, whereas R was created to be and continues to be geared toward data science, statistical analysis, and visualization

    – presence of R among top-10 programming languages, given its specificity, is a testament to how powerful it is for the data science tasks it was built for

Installing R

- Go to https://cran.r-project.org/

- Choose appropriate selection (macOS or Windows) in the "Download and Install R" section at the top. Then:
  - For macOS: Choose the release that is appropriate with your particular macOS version.
  - For Windows: Choose the "base" selection.

- In general, follow default installation recommendations; only deviate from these if you know what you're doing!

Updating R

- If you already have R and RStudio installed, it is a good idea to update them if you have not done so recently

- On Mac, run R directly (i.e. not with RStudio): "Check for R Updates" in "R" menu

- On Windows, install "installr" package, load up the library, and run the code "updateR()"

R vs. Python: Flagging Syntax Differences

- Assignment operator
    - R: `<- or =`
    - Python: `=`
- Vectors
    - R: `rvec <- c(10,20,30,40,50)`
    - Python: `pythonvec = [10,20,30,40,50]`
- Dot operator (.)
    - R: Can be used in names (for variables, functions)
    - Python: Used to access data or behaviors for a particular object type
- Extraction
    - R: `df$var`
    - Python: `df.var`
- Using functions
    - R: `function(x)`
    - Python: `x.function()`
- Basic function names: some same, many different
   - R: `dim(df)`
    - Python: `df.shape`

To R!

RStudio

- RStudio: an integrated development environment (IDE) for R, providing facilities and features to enhance your ability to program and execute analyses in R

    – includes a console, syntax-highlighting editor that supports direct code execution, and tools for plotting, history, debugging, and workspace management

    – in simple terms, it makes working in R more efficient, more organized, and more aesthetically pleasing

    – to download/install and learn more about RStudio: http://www.rstudio.com

RStudio panes (quadrants)

1. **Console** (lower left): where code is evaluated and executed
2. **Source** (upper left): notepad for writing and saving code
3. **Environment/History** (upper right): for tracking progress
4. **Files/Plots/Etc.** (lower right): other stuff, including graphics
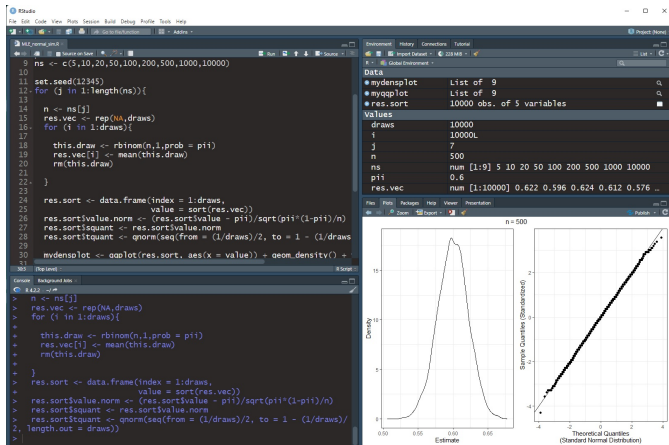
Getting Started with `RStudio`

- Settings

- Writing and saving source code (`R` scripts)

- Installing and loading packages

- Working directory

To R Studio!

R Markdown

- R Markdown: a file format that allows you to create well-formatted documents embedding code chunks and analyses executed in R

    - useful for creating high quality documents, reports, presentations, and content for websites, as well as to output and more easily inspect results and analyses from data science projects

    - requires following specialized conventions in how you write and organize your R Markdown files, some of which is a bit somewhat idiosyncratic but quick to learn

    - if using RStudio, rendering R Markdown files into beautiful final documents is as easy as clicking a button

    - requires installing the rmarkdown and knitr packages, i.e. in R, run:
      install.packages("rmarkdown"); install.packages("knitr")

    - for more information, see:
      http://rmarkdown.rstudio.com