

Introduction to R for Data Management and Analysis

Marcel Ramos

Tuesday, June 4, 2019

Welcome!

Introductions

- Name
- Program/Concentration
- Familiarity with R
- Expectations
- Fun fact?

Class structure

- Lecture and short exercises: 90 minutes
- Break: 10 minutes
- Exercise: 40 minutes
- Discussion: 10 minutes

Grading

- Attendance: 15%
- Quizzes: 20%
- Exercises: 65%

Course Outline

Date	Topics	Assessment
Tues. June 4	Introductions About this class Features of the R language Getting help and troubleshooting	-
Thurs. June 6	Classes and data types data.frame Import/Export subsets	Exercise

Announcements

- Slack available at <https://cunysphcode.slack.com>
- In-person attendance is required
- Syllabus available on Blackboard

Today's class

- Installation
- R and the RStudio interface
- Basic features of the language
- Getting started
- Tips
- Getting help and troubleshooting
- External resources
- R as a calculator

What is R?

- R is a programming language and environment for statistical computing and data visualization
- “Base R” refers to the standalone suite of pre-packaged functions that allow R to function as a language
- Extensions of the R language are what are called “packages”
- A **package** is a container of functions that give R additional flexibility

What is RStudio?

- IDE Interactive Development Environment
- Console + Help + Figures + Project Management
- Let's have a look at it!

What is Git / GitHub?

- GitHub is a public repository of user generated code / analyses
- Provides a foundation for reproducible reports
- Versioning is done using git
- The git software takes care of versioning of text files

How do I get started?

- First download the latest R version from [r-project.org](https://www.r-project.org)
- Install R with all the default settings
- Download RStudio from [RStudio.com](https://www.rstudio.com)
- RStudio allow you to select the R version installed in your system.

Recommendations for RStudio setup

- Tools > Global Options
- Don't restore .RData into workspace
- Never save workspace to .RData on exit

Features of the R Language

- case sensitive!!
- Spaces are ignored (except in names)
- works with functions
- vectorized operations
- objects
- help pages
- ?reserved

Recognizing a function

- a name followed by parentheses `help()`
- arguments (e.g., `functionname(argument1 = "default")`)
- input / output

Useful tips for learning R

Pseudo code	Example code
<code>install.packages(packagename)</code>	<code>install.packages("dplyr")</code>
<code>?functionname</code>	<code>?select</code>
<code>?package::functionname</code>	<code>?dplyr::select</code>
<code>? 'Reserved keyword or symbol' (or backticks)</code>	<code>? '%>%'</code>
<code>??searchforpossiblyexistingfunctionandortopic</code>	<code>??simulate</code>
<code>help(package = "loadedpackage")</code>	<code>help("dplyr")</code>
<code>browseVignettes("packagename")</code>	<code>browseVignettes("dplyr")</code>

General tips for learning R

- Learning R will be frustrating
- Learning a language
- Practice promotes familiarity

R Housekeeping tips

- Maintain a clean R “global” environment
- Save your scripts rather than outputs
- Use object names that are descriptive
- Improve readability with clean formatting

Getting help and troubleshooting

- Critically important
- “Debugging” your script
- Step by step, line by line process

First contact with R

- R as a calculator exercise

Motivating examples

- <http://shiny.rstudio.com/gallery/google-charts.html>
- <https://mramos.shinyapps.io/PowerCalc/>

External Resources

- Coursera
- edX
- RStudio
- Quick-R - Mostly for basic and base functions
- RStudio Cheatsheets

VIM / Emacs (*Extra*)

- Tools for 'efficient' typing
- (Optional) Vim game
- Emacs download