

Introduction to R for Data Management and Analysis

Marcel Ramos, MPH

Session 3

Notes on the Thursday's lecture

- data classes: tabular / non-tab, vectors, lists, and matrices
- assignment `<-`
- extract vector from `data.frame` with `$`
- converting between data types (`as.numeric`)
- importing and exporting data

Lists



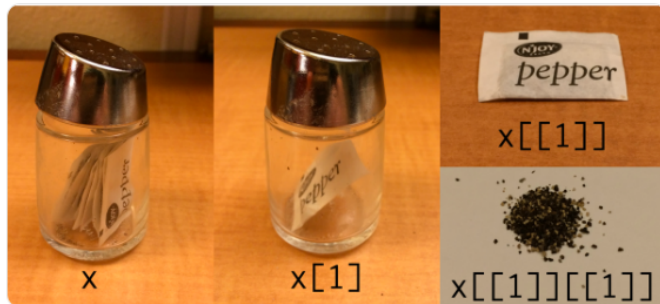
Hadley Wickham ✓

@hadleywickham

Follow



Indexing lists in [#rstats](#). Inspired by the Residence Inn



4:09 AM - 14 Sep 2015

782 Retweets 1,086 Likes



Notes on Thursday's lecture (cont.)

- generating random numbers and matrix
- Up arrow in RStudio
- more on RMarkdown

Recap

- Classes (bicycle analogy)
 - can have particular operations
 - some functions are similar across classes
 - identify a class by using the class function
- Assignment creates objects and puts them in the `.GlobalEnv` (see `ls()`)
- Factors and levels
 - Changing the levels of a factor
- `data.frames` and lists - `fruit_stand` dataset
- coercion (`as.classname`)
- Import and Export data
 - Know your paths!
 - Most export functions include keyword “write” (i.e., `write.table`)
 - Use relative paths for Rmd files
 - Put the data and file in the same folder
 - You can use absolute paths for interactive sessions (console)
- subsetting using vectors

Motivation



Everybody wants to be a
bodybuilder, but don't nobody
wanna lift no heavy ass weight.

— *Ronnie Coleman* —

AZ QUOTES

Motivation (cont.)

- Learning R requires practice
- Persistence
- Find fun exercises (Trello Board)
- Follow along with the live coding examples
- Enjoy it!

Expectations



Kim Cressman

@swmpkim

Follow



Advice I gave 2x today:

Don't feel like you have to LEARN R, like you have to know **everything** before you can do **anything**.

Just pick a thing you want to do, and learn how to do it.

It's easier to digest when you have a goal - learn the steps that get you there.

7:46 PM - 6 Jun 2018

401 Retweets 1,861 Likes



Expectations (cont.)

- Exposure to the R language
- Basics of troubleshooting and debugging
- Learning R won't happen overnight

Data Manipulation Overview

- Inspecting a `data.frame`
 - `dimension`
 - `rownames`, `colnames`
 - `head`
- Subsetting (cont.)
 - vectors and `[` with character, numeric, logical
 - lists and `[/ $`
 - double bracket extraction `[[`
 - using conditions
- Sorting and aggregating data
- removing duplicated records
- removing records with NA
- merging and binding
- transformations

Subsetting

- can be done with either the `[` bracket or tidyverse operations
- Think about dimensions before doing the subset
- Think in terms of verbs (slice, select)
- Draw it out!
- `$` extracts a vector from a `data.frame`
- `[[` extracts and reduces to a single vector where possible from a `data.frame` or list
- conditions help us specify what parts of the data we want
 - `sex == "males"`
 - `age >= 18`

Sorting and aggregating data

- `order` function - returns an index of ordered positions
- tidyverse equivalent: `arrange` - returns the arranged data

Formula notation in R

- Uses the \sim for denoting a formula
 - $y \sim m \cdot x + b$
- Good for specifying linear models
- Mainly used in base R code
- Useful for creating crosstabs!
 - `xtabs(A ~ B, data = blue)`
- Look out for formula class inputs
 - see `?xtabs`
 - see `?t.test`
- Usually requires a data input / argument for the function

Useful conditions for subsetting

- Removing duplicated rows
 - duplicated on a data.frame
- Removing records with NA
 - is.na to get a logical vector

Merging and Binding

- merge function
 - takes two data.frames as input
 - arguments tell it how to merge
 - see example
- cbind and rbind
 - concatenate by columns or rows
 - rbind: names in columns must match
 - cbind: number of rows must match
- Creating bins from continuous variables
 - Hmisc::cut2 - easy way to create categories from numeric variables
- Tidyverse
 - join construct
 - see RStudio cheatsheet

Transformations / Manipulations

- long to wide format
- dplyr and reshape2 packages
- aggregate / group_by

Discussion

- Working with data
 - Tools at your disposal
 - String together functions to reach a desired outcome
 - Add comments to code to explain steps
- Recognize how data should be represented
 - long - longitudinal data
 - wide - survey data
- Recognize what data format is best for visualization
 - ggplot2 may want long data