

A 30 Minute Introduction to R

Vito

September 21, 2012

Getting Started

- Think about your task
 - Before you even open R, or even think about opening R, think about your task at hand. You have a task at hand and you have an objective
 - if you don't then go to the bar
 - When you know what it is you want to accomplish, open R.
- Relax
 - The first thing to do when you open up R and are staring at the blank, empty screen, which is called the Graphic User Interface or GUI: RELAX.
 - R is intimidating in a lot of ways, it's sort of like sitting down with a pen and a blank sheet of paper and trying to answer an essay question. You might experience coder's block, a far more serious condition than writer's block.

Loading Data

- Find the data on disk
 - Change your working directory to that location
 - Two ways to set your working directory
 - File > Change Dir
 - `setwd("location")`
- Read data into memory
 - `read.table("data.dat")`
 - Used for basic formats such as `.tsv` or `.csv`
 - Example: `"votedata.dat"`

Library

- Libraries or Packages add functions and operability to R
 - `read.dta("data.dta")`
 - `.dta` is Stata's default format
 - Example: `"votedata.dta"`
 - Add the `"foreign"` package
 - `library(foreign)`
 - Package > Install Package
 - Package > Load Package

Viewing Data

- Functions to know
 - `fix()` allows you to view all the data
 - can also use `edit()`, or `head()` to view only several rows
 - `colnames()` prints the column/variable names
 - `dim()` gives you the dimension (row by col)
 - can also use `nrow()` or `ncol()`
 - `summary()` or `describe()`
 - Describe is part of the Hmisc package
 - `plot()` is your basic plotting function

Objects

- R is an object-oriented programming language
- Object?
 - Something that can be manipulated
 - `ls()` lists the objects in the environment
 - `rm()` removes an object from the environment
- Keep track of your object classes
 - Cannot stress this enough: most errors in R come from *thinking* you have an object of one class when you actually have an object of another

Objects

- Classes of objects
 - `class()`
 - common types: `data.frame`, `logical`, `numeric`, `character`, `list`, `factor`, `matrix`, `vector`
- Changing object class
 - `as.character()`, `as.numeric()`, `as.anything()`

Common Analysis Functions

- `sum()`, `mean()`, `sd()`, `max()`, `min()`
 - All do what you think they do
- `var()`, `cov()`, `cor()`
 - `na.rm()`
 - Remove missing values?
- `lm()`
 - regression function
 - summary of which is given by... `summary()`
 - predictions given by... `predict()`

Sourcing

- Replication
 - Always, ALWAYS keep a log file
 - Just copy and paste each line into a log file
- Sourcing
 - If you have a log file or any code you can source it into R
 - `source()`
 - Reads the code in line by line
 - `random250.r`

Saving

- Saving in R can be tricky... first thing to do:
 - Think: What is it I want to save?
 - A data.frame?
 - `write.dta()`, `write.table()`, `write.delim()`
 - The entire environment?
 - `save.image()`
- Help functions for stuff on CRAN
 - `?()` for stuff you can currently use
 - `??()` for stuff you can't or don't know if you can
 - Both extremely useful... cannot underestimate their usefulness... so useful

Remember...

- UNDERSTAND THE ERROR
 - For any individual error, there will always be more than one way to solve it.
 - BUT, there will always be only one cause. If you get an error and you intend to fix it, you must first understand the source of the error.
- DO NOT RECREATE THE WHEEL!!
 - Creating something out of nothing is one of the most difficult tasks we do. Use packages when they exist, use code available on the Web, use code from your friends.
 - Basic process: use code, reuse code, repeat process.
- TALK TO THE TEDDY BEAR
 - Just like when you're editing an essay, editing code or thinking through a problem can be done by talking your way through it.

Resources

- Number One Resource: peers
 - Just ask, save yourself the lonely nights with a bottle of gin and a half gallon of fruit punch...
- Armstrong' s introduction
 - www.quantoid.net/ICPSR.php
- Vito Ricci' s regression reference card
 - <http://cran.r-project.org/doc/contrib/Ricci-refcard-regression.pdf>
- Quick R
 - <http://www.statmethods.net/>