Using R to Analyze and Visualize Education Data

Jared Knowles

Policy Research Advisor
Wisconsin Department of Public Instruction

May 5, 2012/ R Bootcamp

Outline

R

- R is an Open Source (and freely available) environment for statistical computinga nd graphics
- Available for Windows, Mac OS X, and Linux
- R is being actively developed with two major releases per year and dozens of releases of add on packages
- R can be extended with 'packages' that contain data, code, and documentation to add new functionality

R Advantages

- R is a common tool among data experts at major universities
- No need to go through procurement, R can be installed in any environment on any machine and used with no licensing or agreements needed
- R source code is very readable to increase transparency of processes
- R code is easily borrowed from and shared with others
- R is incredibly flexible and can be adapted to specific local needs
- R is under incredibly active development, improving greatly, and supported wildly by both professional and academic developers

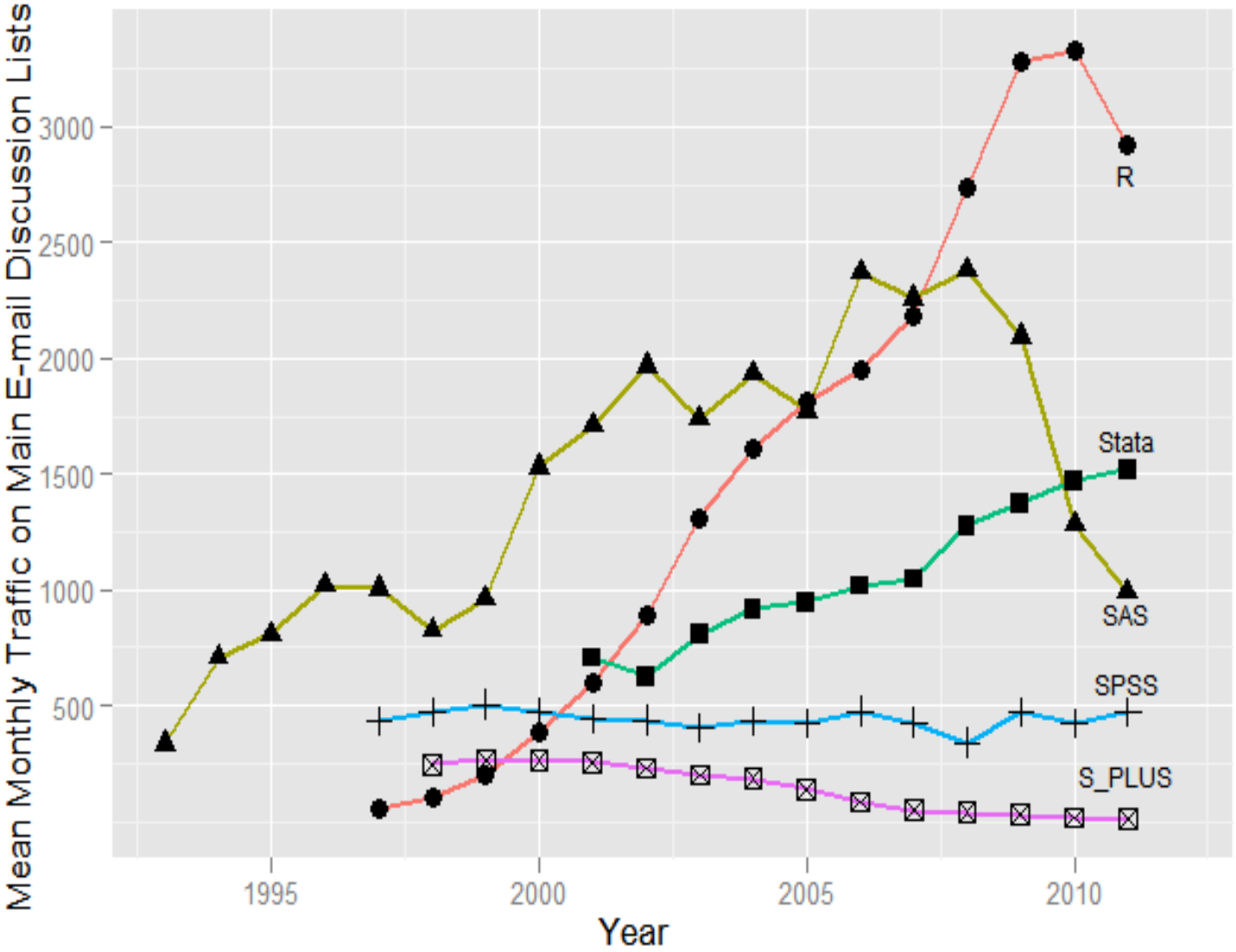
R Advantages Continued

- R is platform agnostic–Linux, Mac, PC, server, desktop, etc.
- R can output results in a variety of formats
- R can build routines straight out of a database for common and universal reporting

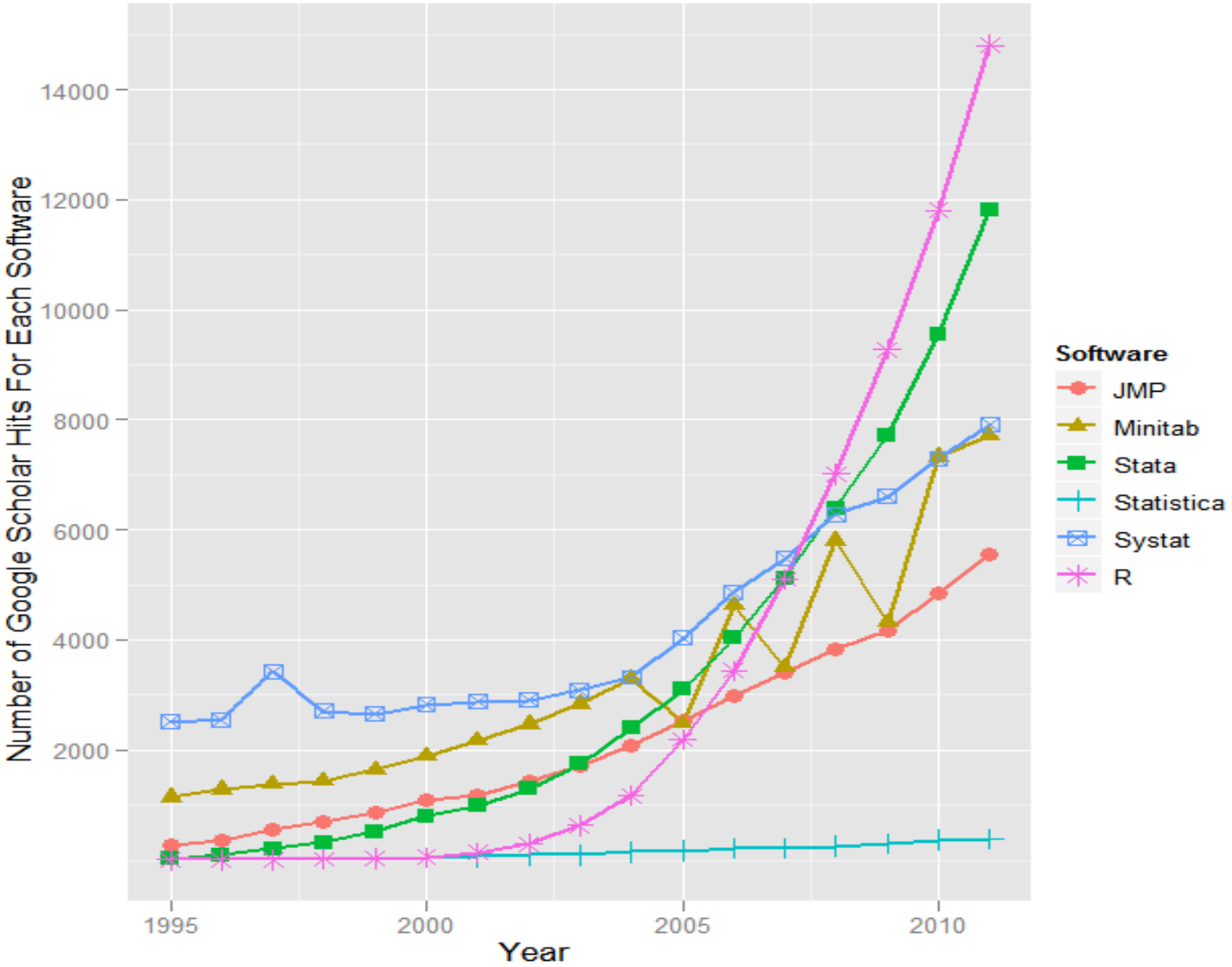
R Can Compliment

- R plays nicely with data from Stata, SPSS, SAS and others
- R can check work, produce output, visualize results from other programs
- R can do bleeding edge analysis that aren't available in proprietary packages yet
- R is becoming more prevalent in undergraduate statistics courses

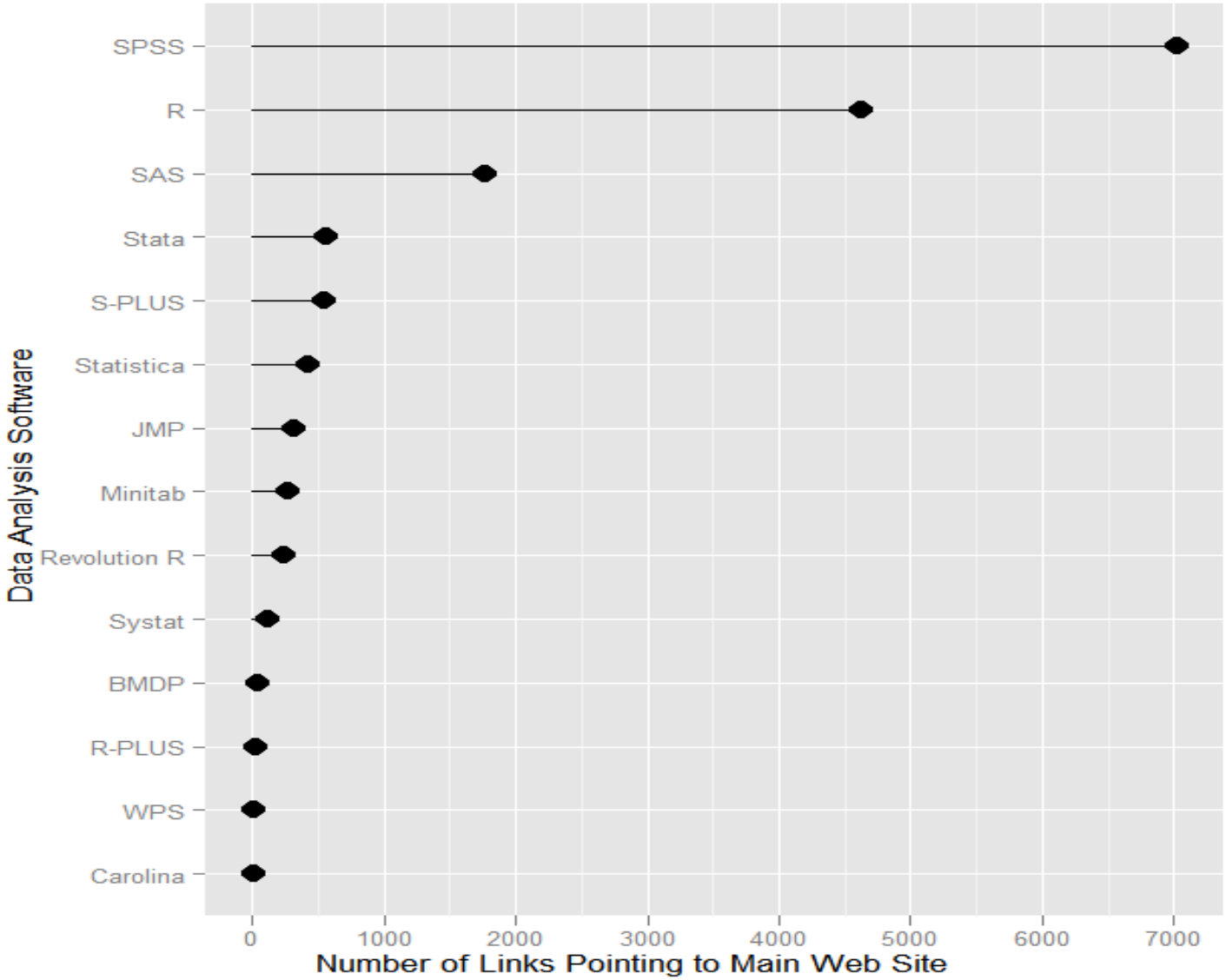
R Popularity



R Popularity II

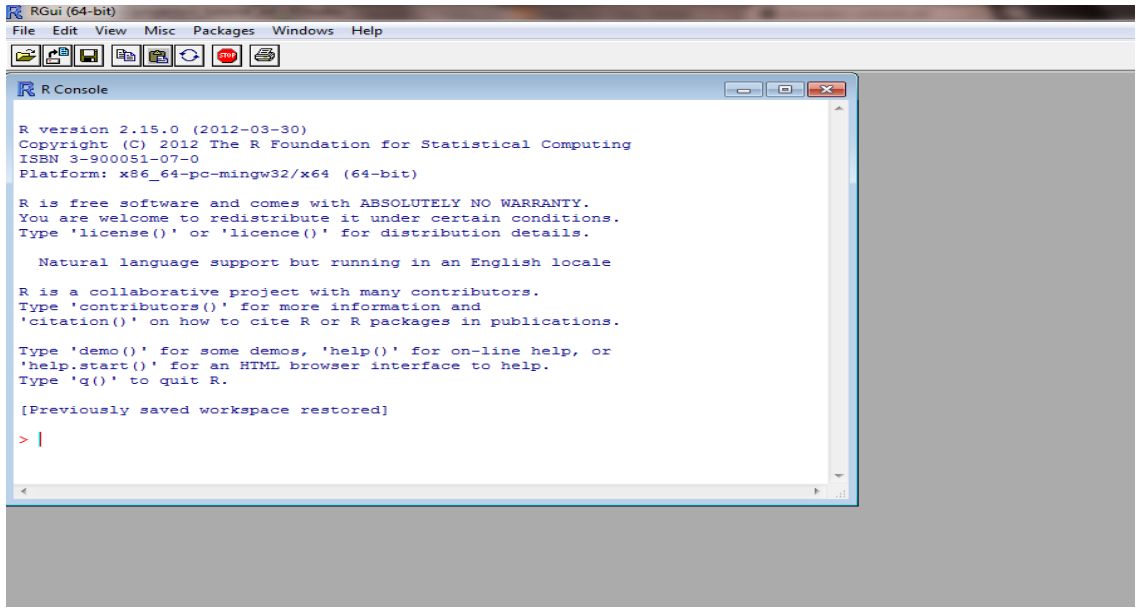


R Popularity III

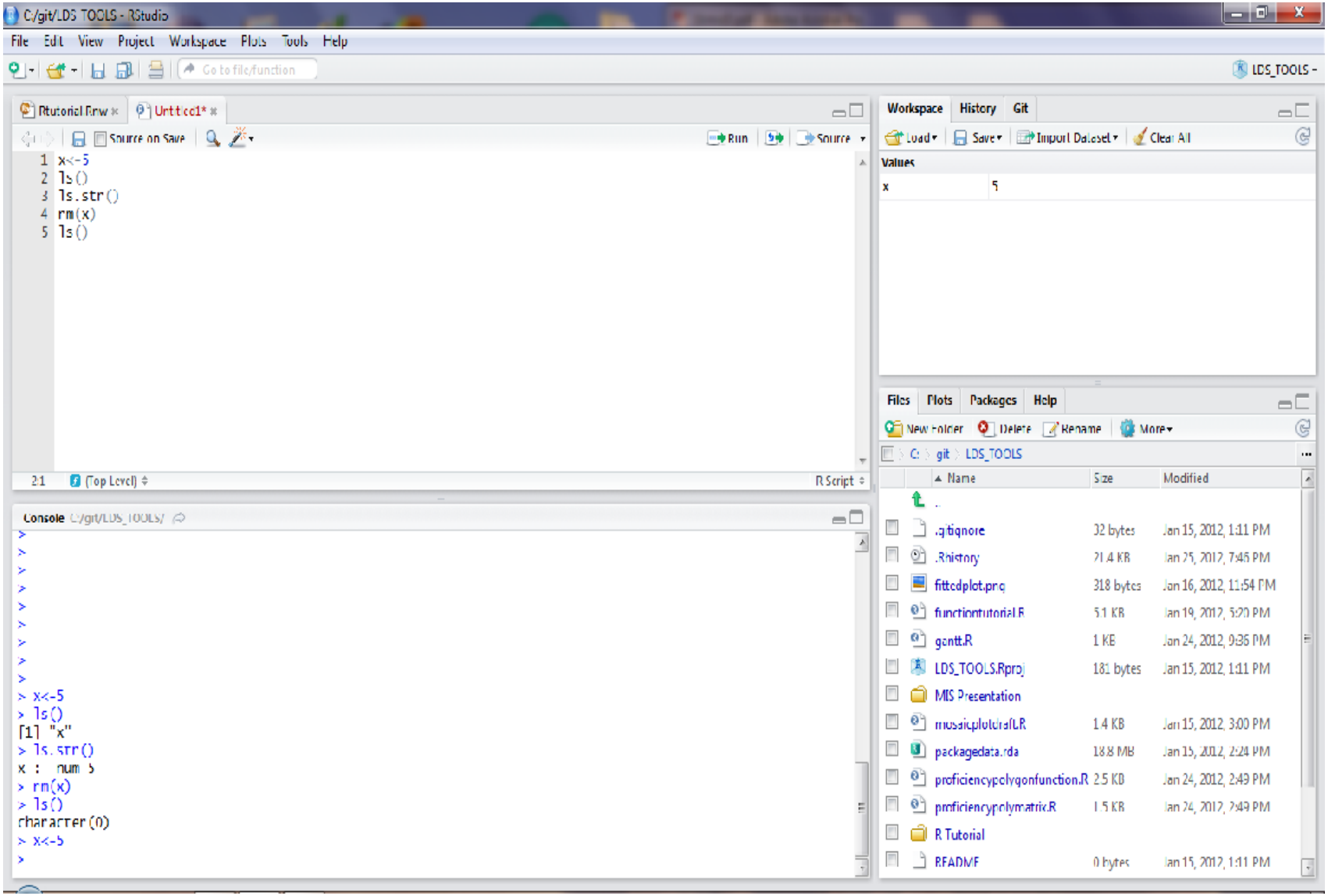


Windows

- Setting up R on Windows XP is straightforward. On Windows 7 or Vista you may need to run in administrator mode to install extensions.
- Standard R interface in Windows is pretty ugly.



Using an IDE To Make Things Easier



R as a Calculator

```
> 2+2 # add numbers
[1] 4
> 2*pi #multiply by a constant
[1] 6.283185
> 7+runif(1,min=0,max=1) #add a random variable
[1] 7.59793
> 4^4 # powers
[1] 256
> sqrt(4^4) # functions
[1] 16
```

Using the Workspace

- To do more we need to learn how to manipulate the 'workspace'.
- This includes all the scalars, vectors, datasets, and functions stored in memory.
- All R objects are stored in the memory of the computer, limiting the available space for calculation to the size of the RAM on your machine.
- R makes organizing the workspace easy.

```
> x<-5 #store a variable with <-
> x    #print the variable
[1] 5
> z<-3
> ls() #list all variables
[1] "x" "z"
> ls.str() #list and describe variables
x :  num 5
z :  num 3
> rm(x)    # delete a variable
```

Reading Data

- To read data in we have to tell R where it currently is on the filesystem
- Then we have to tell it where to look for the dataset and how to read it
- CSV files are simplest for beginning use cases, but R is flexible

```
> # Set working directory to the tutorial director
> # In RStudio can do this in "Tools" tab
> setwd('~/r_tutorial_ed')
> #Load some data
> df<-read.csv('data/smalldata.csv')
> # Note if we don't assign data to 'df'
> # R just prints contents of table
```

Objects

- Everything in R is an object–even functions
- Objects can be manipulated many ways
- A common example is applying the 'summary' function to a variety of object types and seeing how it adapts

```
> summary(df[,28:31]) #summary look at df object
   schoollow          readSS          mathSS
 Min.   :0.0000   Min.   :251.5   Min.   :210.2
 1st Qu.:0.0000   1st Qu.:430.0   1st Qu.:418.3
 Median :0.0000   Median :495.3   Median :480.1
 Mean   :0.2422   Mean   :496.2   Mean   :483.4
 3rd Qu.:0.0000   3rd Qu.:562.5   3rd Qu.:543.2
 Max.   :1.0000   Max.   :833.2   Max.   :828.4
        proflvl
 advanced   : 788
 basic      : 523
 below basic: 210
 proficient :1179
> summary(df$readSS) #summary of a single column
```