R

BINA RAMAMURTHY

Tools for Analytics

- Elaborate tools with nifty visualizations; expensive licensing fees: Ex: Tableau, Tom Sawyer
- Software that you can buy for data analytics: Brilig, small, affordable but short-lived
- Open sources tools: Gephi, sporadic support
- Open source, freeware with excellent community involvement: R system
- Some desirable characteristics of the tools: simple, quick to apply, intuitive, useful, flat learning curve
- A demo to prove this point: **data → actions /decisions**

R Language

- R is a software package for statistical computing.
- R is an interpreted language
- It is open source with high level of contribution from the community
- “R is very good at plotting graphics, analyzing data, and fitting statistical models using data that fits in the computer’s memory.”
- “It’s not as good at storing data in complicated structures, efficiently querying data, or working with data that doesn’t fit in the computer’s memory.”

Why R?

- There are many packages available for statistical analysis such as SAS and SPSS but there are expensive (user license based) and are proprietary.
- R is open source and it can pretty much do what SAS can do but free.
- R is considered one of the best statistical tools in the world.
- For R people can submit their own packages/libraries, using the latest cutting edge techniques.
- To date R has got almost 15,000 packages in the CRAN (Comprehensive R Archive Network – The site which maintains the R project) repository.
- R is great for exploratory data analysis (EDA): for understanding the nature of your data before you launch serious analytics.
- Many tutorial vignettes are available for you to learn.

R Packages

- An R package is a set of related functions
- To use a package you need to load into R
- R offers a large number of packages for various vertical and horizontal domains:
- Horizontal: display graphics, statistical packages, machine learning
- Verticals: wide variety of industries: analyzing microarray data, modeling credit risks, social sciences, automobile data (none so far on sensor data from automobiles!)

Library

- Library→ Package→ Class→
- R considers every item as a class/object
- Thousands of Online libraries
- 150000 packages
- CRAN: Comprehensive R Archive Network
- Look at all the packages available in CRAN

http://cran.r-project.org/

- R-Forge is another source for people to collaborate on R projects

Approach to learning R

- R Basics, fundamentals
- The R language
- Working with data
- Statistics with R language

R Basics

- Obtaining the R package
- Installing it
- Install and use packages
- Quick overview and tutorial

A quick demo of R's capabilities

- See p.98 onwards till p.102 of
simpleR: Using R for introductory statistics
By J. Verzani
http://cran.r-project.org/doc/contrib/Verzani-SimpleR.pdf

- R in a nutshell, by Joseph Adler, O'reilly, 2010
Chapter 3 Basics, Ch.4 packages,
(search for this online)
Look for these resources online…and try these.
- See Rhandout.pdf linked to today's lecture

Packages

- A package is a collection of functions and data files bundled together.
- In order to use the components of a package it needs to be installed in the local library of the R environment.
- Loading packages
- Custom packages
- Building packages

More R

- R syntax
- R Control structures
- R Objects
- R formulas

R Syntax

- R language is composed of series of expression resulting in a value
- Examples of expression include assignment statements, conditional statements, and arithmetic expressions

```
> a<- 42
> b <- a% 5
> if (b == 0) " a divisible evenly by 5" else " not evenly divisible by 5"
[1] " not evenly divisible by 5"
```

- Variables in R are called symbols

R Objects

- All items vectors, lists, even functions are considered as objects in R
- Example : a vector of integers , and then floats
  - p<- c(6, 8,4,5,78)
  - p
  - q<- c(5.6, 4.5, 7.8, 9.3)
  - Q
- A list can be made of items of any type

```
> r<- list(p, q, "this demo")
> r
[[1]]
[1]  6  8  4  5 78

[[2]]
[1] 5.6 4.5 7.8 9.3

[[3]]
[1] "this demo“
```

Special Values

- > v <- c(1,2,3)
- > v
- [1] 1 2 3
- > length(v) <- 4
- > v
- [1] 1 2 3 NA

- NA : not defined or not available
- Very Large and very small numbers:
- > 2 ^ 1024
- [1] Inf
- > - 2 ^ 1024
- [1] -Inf

Curly braces

- Curly braces are used to group a set of operations in the body of a function:

```
> f <- function() {x <- 1; y <- 2; x + y}
> f()
[1] 3
```

Control Structures

- > i <-4
- > repeat {if (i > 25) break else {print(i); i <- i + 5;}}
- [1] 4
- [1] 9
- [1] 14
- [1] 19
- [1] 24

Demo: Exam Grade: Traditional reporting 1

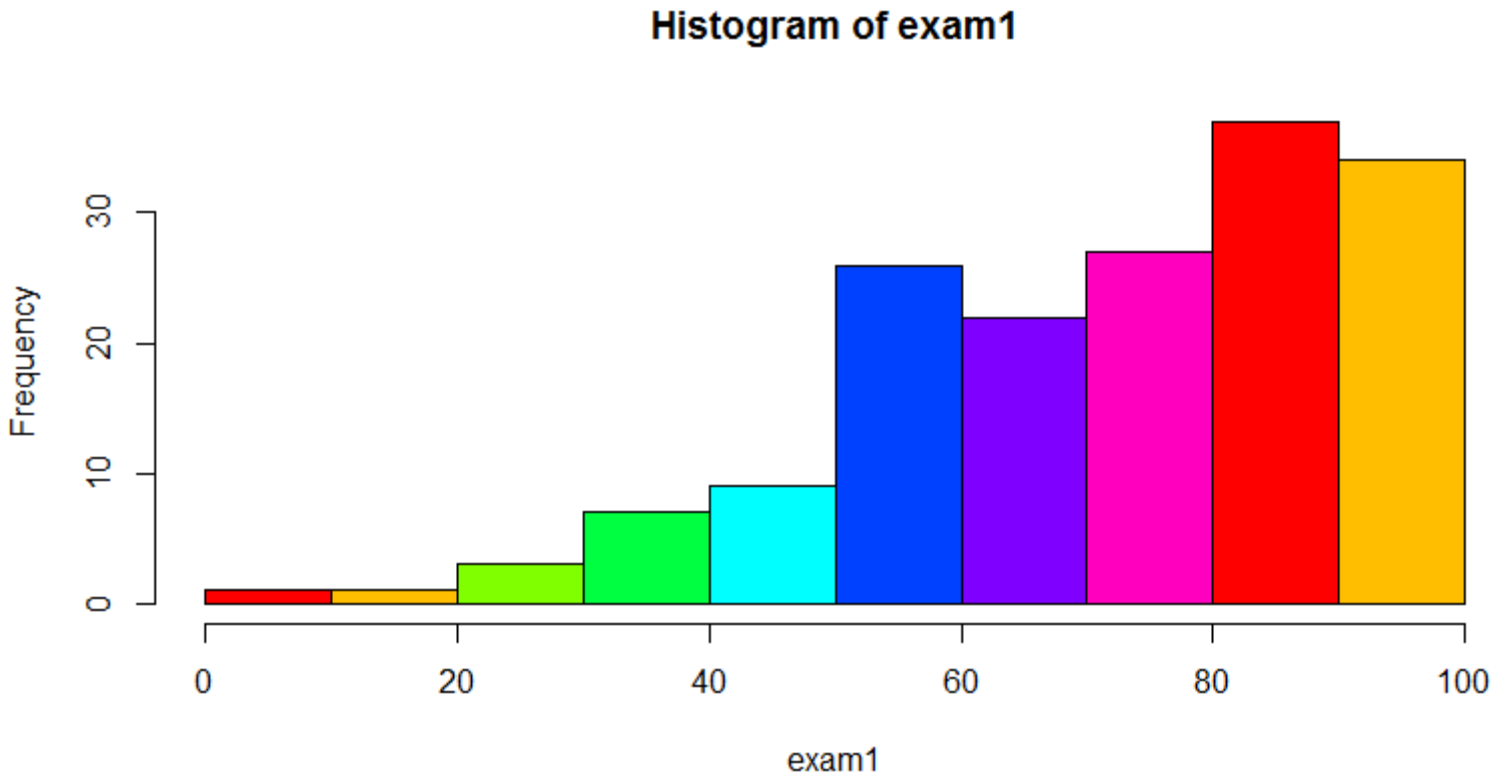
Q1	Q2	Q3	Q4	Q5	Total
16.7	13.9	9.6	18.5	13.7	72.4
20.0	16.0	9.0	19.0	17.0	76.0
20.0	20.0	15.0	25.0	20.0	90.0

Q1	Q2	Q3	Q4	Q5	Total
16.0	14.2	9.6	19.4	14.0	73.2
80.1%	71.1%	64.0%	77.4%	70.2%	73.2%

Q1	Q2	Q3	Q4	Q5	Total
17.3	13.6	9.7	17.6	13.3	71.5
86.7%	67.8%	64.6%	70.3%	66.7%	71.5%

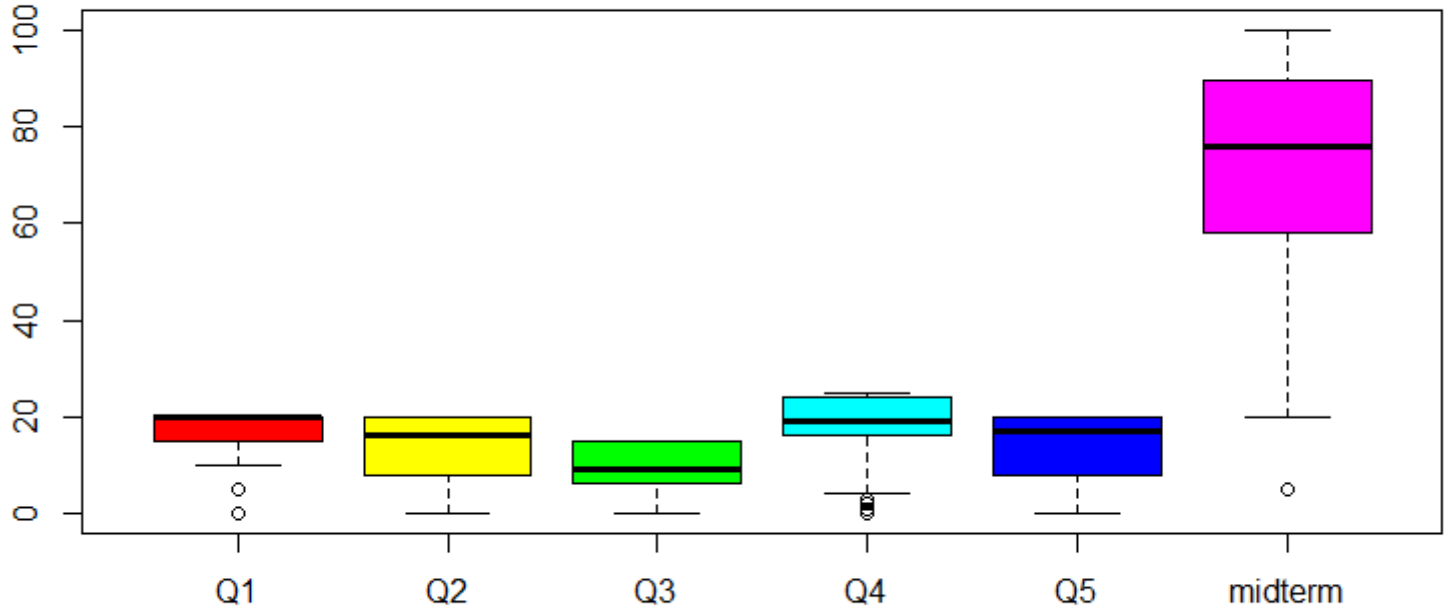
Question 1..5, total, mean, median, mode; mean ver1, mean ver2

Traditional approach 2: points vs #students

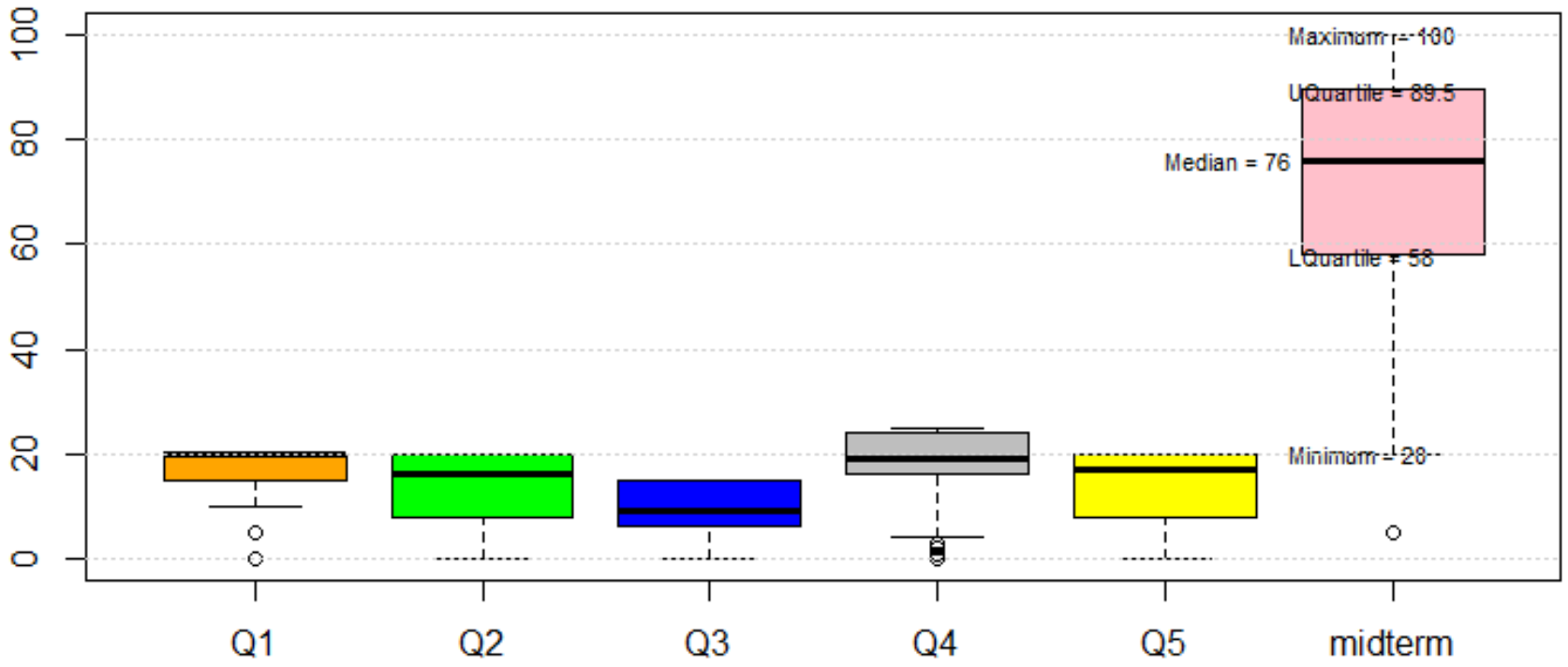


Distribution of exam1 points

Individual questions analyzed..



Interpretation and action/decisions
Do you see the difference?



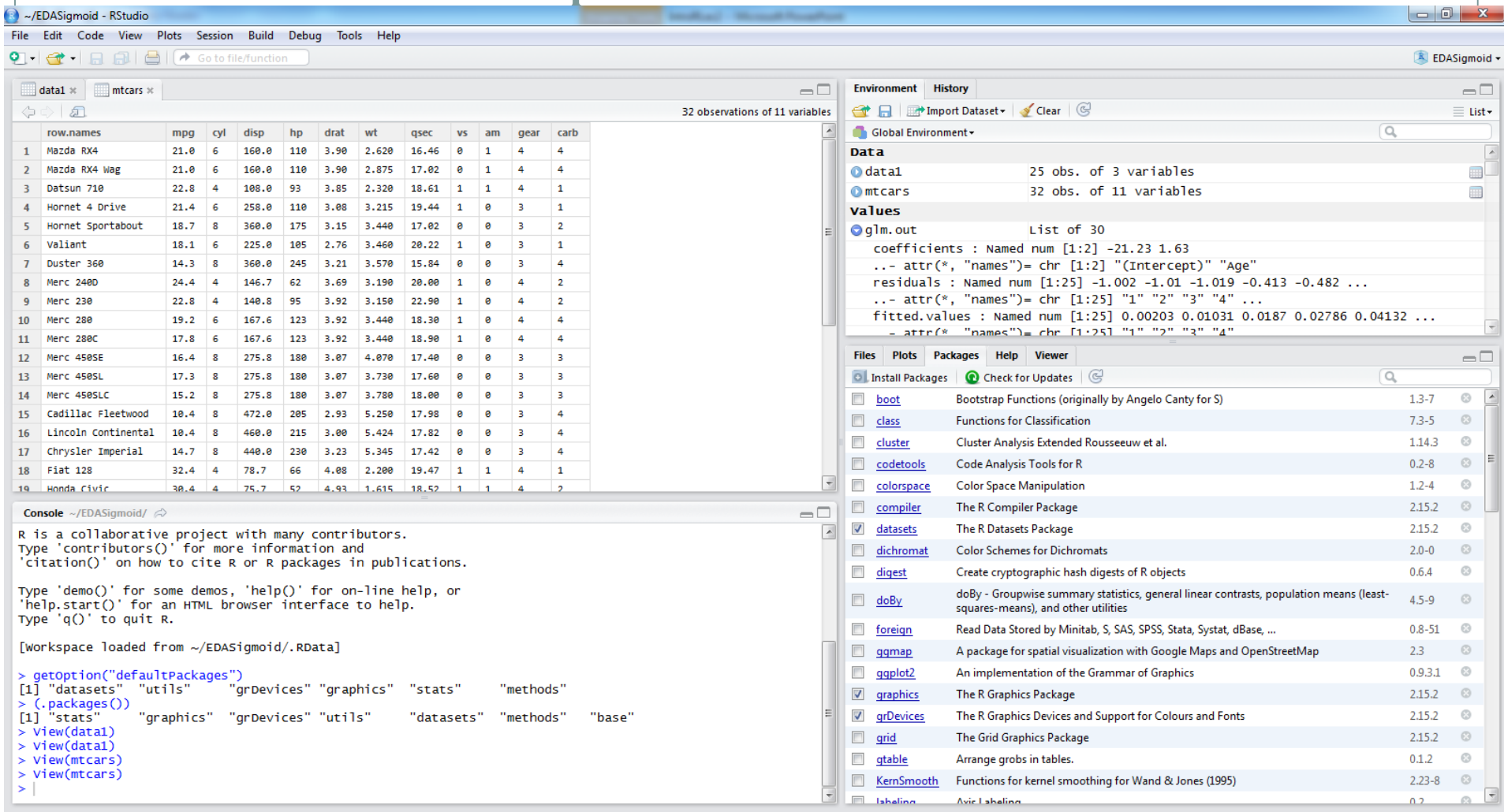
R-code


```
data2<-read.csv(file.choose())

exam1<-data2$midterm
hist(exam1, col=rainbow(8))
boxplot(data2, col=rainbow(6))

boxplot(data2,col=c("orange","green","blue","grey","yellow", "sienna"))
fn<-boxplot(data2,col=c("orange","green","blue","grey","yellow", "pink"))$stats

text(5.55, fn[1,6], paste("Minimum =", fn[1,6]), adj=0, cex=.7)
text(5.55, fn[2,6], paste("LQuartile =", fn[2,6]), adj=0, cex=.7)
text(5.0, fn[3,6], paste("Median =", fn[3,6]), adj=0, cex=.7)
text(5.55, fn[4,6], paste("UQuartile =", fn[4,6]), adj=0, cex=.7)
text(5.55, fn[5,6], paste("Maximum  =", fn[5,6]), adj=0, cex=.7)

grid(nx=NA, ny=NULL)
```

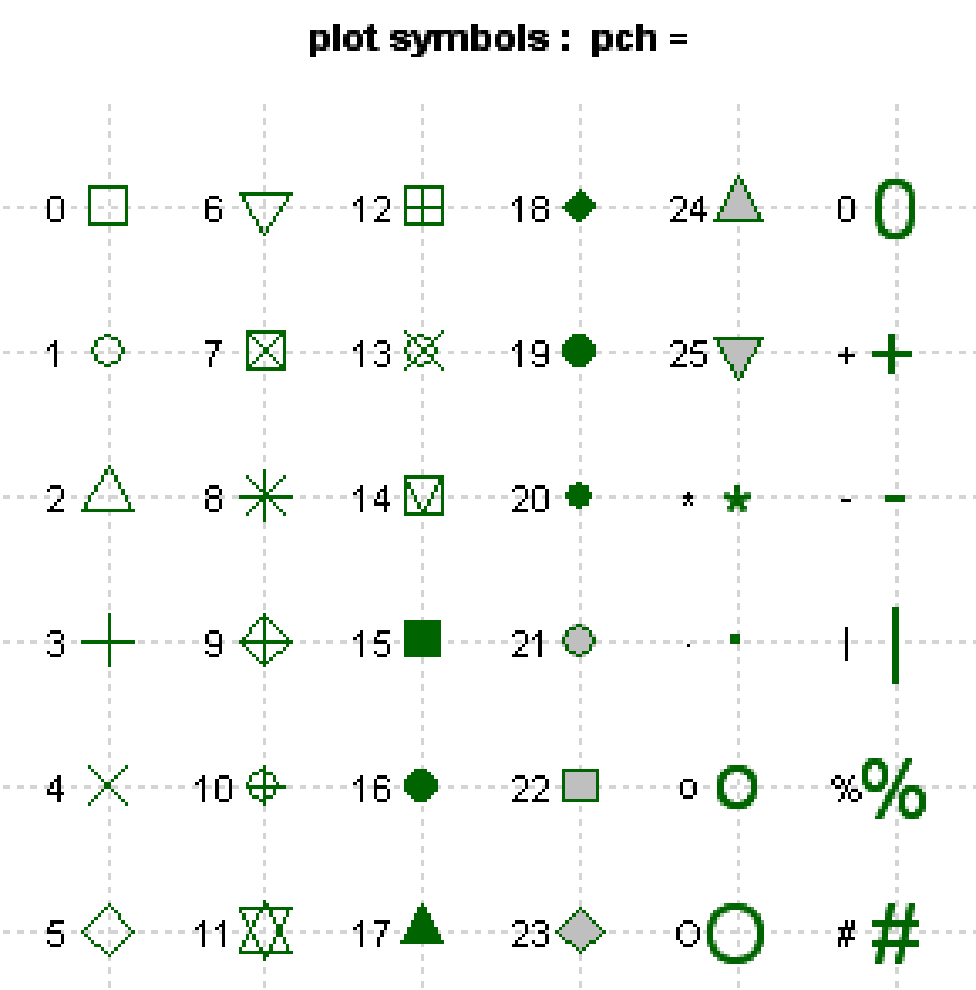
One last example

- survey.results <- factor(c("Disagree", "Neutral", "Strongly Disagree", "Neutral", "Agree", "Strongly Agree","Disagree", "Strongly Agree", "Neutral","Strongly Disagree", "Neutral", "Agree"),levels=c("Strongly Disagree", "Disagree", "Neutral", "Agree", "Strongly Agree"),ordered=TRUE)
- survey.results
- R will automatically compute the numbers in each category!
- There are many more functions and operations available in R that are related to data.

Lets explore RStudio



What is in pch?



Let do a EDA of cars data

- Look at the tutorial in handout#1
- R is good for Exploratory Data Analytics
- It is really good for most statistical computing you will you in your domain.
- You can repeat the same on Jupyter.
- We will also look at real “click” data from Nytimes from Oneill and Schutt’s text. See the data in today’s notes.