# Handout #1

Title: FAE Spring/2015
Course: Econ 368/01 Instructor: Dr. I-Ming Chiu

The abbreviations of FIVE references are PE, MPS, BR, FCDAE, and PRA. There is additional reference about the use of R (BR).

Additional reading that accompanies this handout: Chapter 2, 3 of BG

Types of data (1)
a) Numerical data
e.g. Height, weight, stock prices, trading volume, etc.

b) Categorical data (nominal vs. ordinal)
e.g. Sex (male, female), working status (employed, unemployed), political affiliation (Dem, Rep, Ind) $\in$ nominal.
e.g. cup size at coffee shops (small, medium, large), grade (F, D, C, B, A) $\in$ ordinal.

When there are only two levels in a categorical variable, there is no need to differentiate whether it is nominal or ordinal.

Types of data (2)
In the field econometrics, in addition to the above data differentiation, the data can be categorizes as
a) Cross-section
e.g. Students' exam scores in the first exam of class FAE 368.

b) Time series
e.g. 10 year government bond's monthly rate in 2014.

c) Longitudinal/Panel
e.g. Annual GDP per capita for European countries between 1999 and 2014.

Types of data (3)
a) Experimental
e.g. Clinical Trials

b) Observational
e.g. Temperatures in NJ, crime rates in Camden, etc.

We will deal with continuous, cross-sectional, and observational data most of the time in this class. Occasionally, categorical type of data will also appear; e.g. ANOVA in Topic 3.

# Introduction to R

R is an open source, high-level, multifunction programming software. It is perfect for data analysis and graphics. In this handout, we'll learn some basic written-in functions (commands) to explore the data.

**R website**
http://cran.r-project.org/

**Preliminary**
? the name of the function you want to find

help("what you want to find")

"R" is case sensitive; a ≠ A

getwd() & setwd()

Set up a permanent working directory

R is a calculator

## I. BASIC CONCEPTS

R is an objected-oriented language. The word "Object" means that anything can be assigned to a variable. There are all kinds of objects in R and we'll focus on data objects first that includes "vector", "matrix", "data frame", "array", and "list". The element in each data object has different mode; the common used mode include "numeric", "character (i.e., text)", and "logical".

e.g. Generate vectors and retrieve its elements

(a) Using the "c" (i.e., concatenate or combine) command

```
studentID = c(1, 2, 3, 4, 5)
examscore = c(85, 77, 63, 90, 68)
studytime = c(2, 1.5, 0.9, 2.2, 1.3)
gender = c("F", "M", "M", "F", "F")
mode(studentID)
mode(gender)
```

Try the following

```
studentIDnew = c("1', "2", "3", "4", "5")
mode(studentIDnew) #What's the outcome? Make a guess.
```

```
academicrecord = data.frame(studentID, examscore, studytime, gender) #create a data frame
```

(b) Using the "seq" command or ":" to generate a sequence of numbers

```
x = seq(1,20)
y = seq(1,20, 2)
z = 1:10
```

(c) Retrieve the elements

```
x[x > 10] #using condition
x[11:20] #using location (index)
x[-(1:10)] #using the location of the complements
```

## II. READ DATA INTO R AND EXPLORE THE DATA

**Example One: Cross-sectional Data**

Most of the commonly used data files can be read into R easily, but we will use "csv" and "txt" data files only in this class. Instead of explaining the procedures step by step, we'll use an example (read pp. 138, Chapter 12, Faraway e-book) to show the whole process.

```
Case 1: Data1 = read.table("Chicago.txt",header=T) # data file is in the working directory.

Case 2: Data1 = read.table("C:/CHIU_DOCUMENT/Academics_School/Rutgers
All/2015/FAE 368/Data/Chicago.txt", header=T) # if the data file is stored somewhere.

rm(list=ls()) #clear up the console
names(Data1) #show variable names
dim(Data1) #dimension of Data1
str(Data1) #structure of Data1

head(Data1) #show the first six observations (the default) of Data1
```

| | race | fire | theft | age | volact | involact | income |
|---|---|---|---|---|---|---|---|
| 60626 | 10.0 | 6.2 | 29 | 60.4 | 5.3 | 0.0 | 11744 |
| 60640 | 22.2 | 9.5 | 44 | 76.5 | 3.1 | 0.1 | 9323 |
| 60613 | 19.6 | 10.5 | 36 | 73.5 | 4.8 | 1.2 | 9948 |
| 60657 | 17.3 | 7.7 | 37 | 66.9 | 5.7 | 0.5 | 10656 |
| 60614 | 24.5 | 8.6 | 53 | 81.4 | 5.9 | 0.7 | 9730 |
| 60610 | 54.0 | 34.1 | 68 | 52.6 | 4.0 | 0.3 | 8231 |

```
Data1[1:2,1:4] #show the first two rows and the first four columns of Data1
```

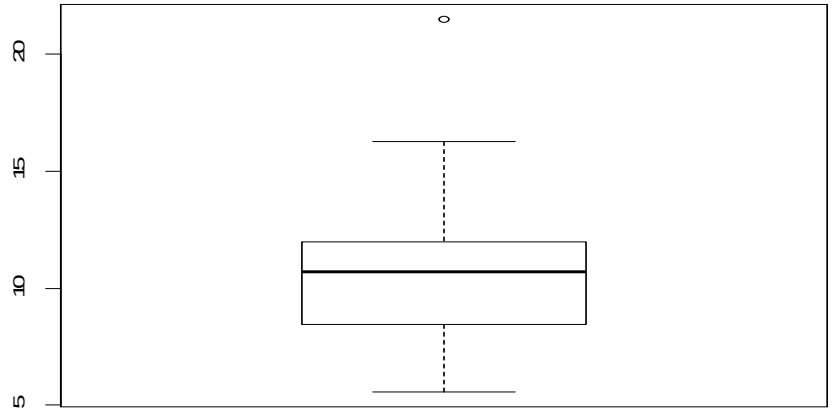
| | race | fire | theft | age |
|---|---|---|---|---|
| 60626 | 10.0 | 6.2 | 29 | 60.4 |
| 60640 | 22.2 | 9.5 | 44 | 76.5 |

Data1$income #the outcome is the whole column of "income" variable
Data1[,7] = Data1$income/1000 #rescale the "income" variable
Head(Data1) # Data1 after rescaling

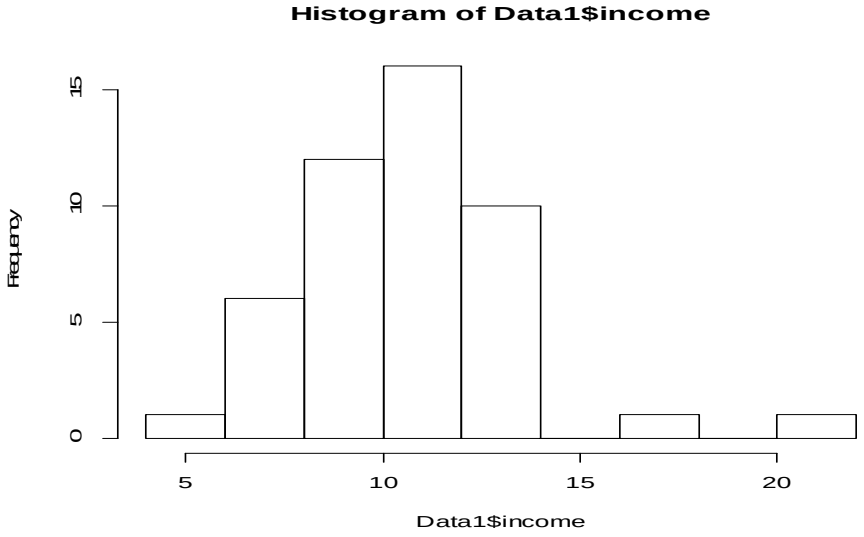
Try the following written-in functions (x: represents the variable name)

| | | | |
|---|---|---|---|
| max(x) | range(x) | min(x) | var(x) |
| sum(x) | cor(x,y) | mean(x) | median(x) |

boxplot(Data1$income)

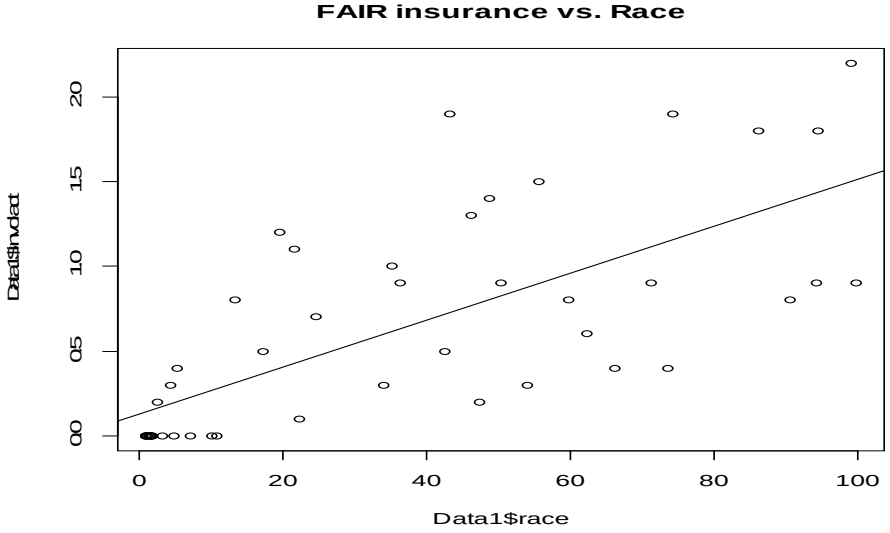



hist(Data1$income)

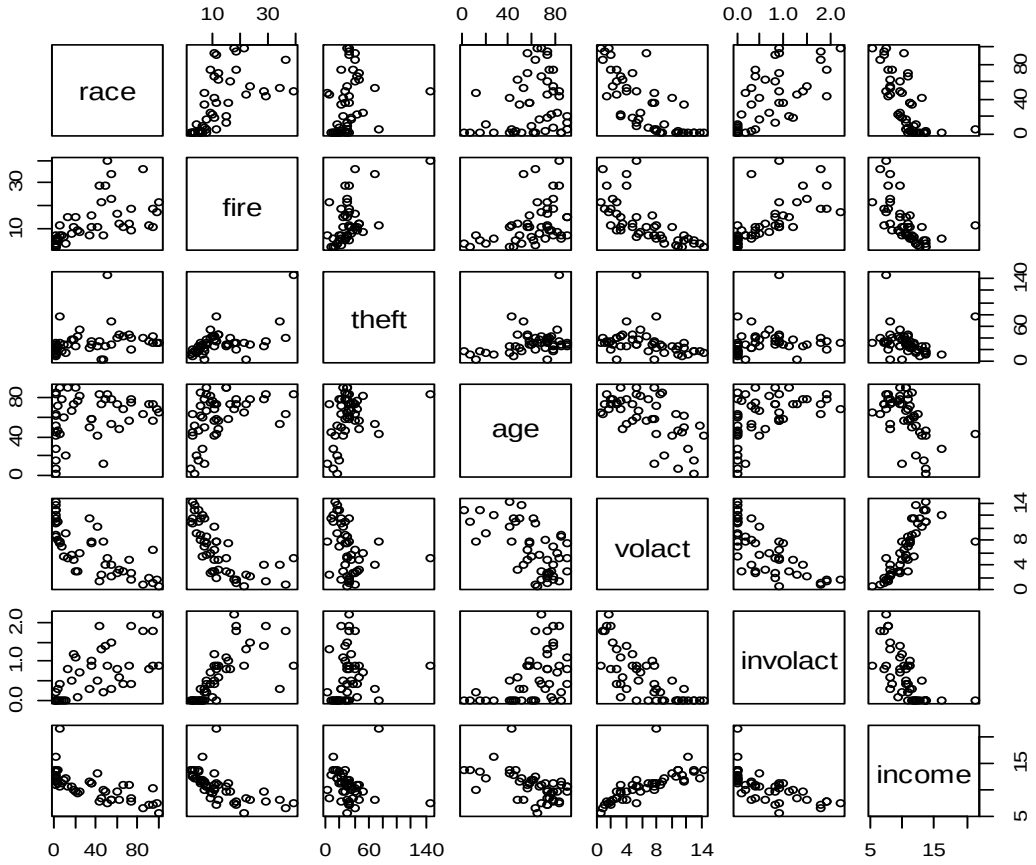



summary(Data1$income) #summarize the "income" variable

```
plot(Data1$race,Data1$involact,main="FAIR insurance vs. Race")
abline(lm(Data1$involact~Data1$race))
```




```
pairs(Data1)
```



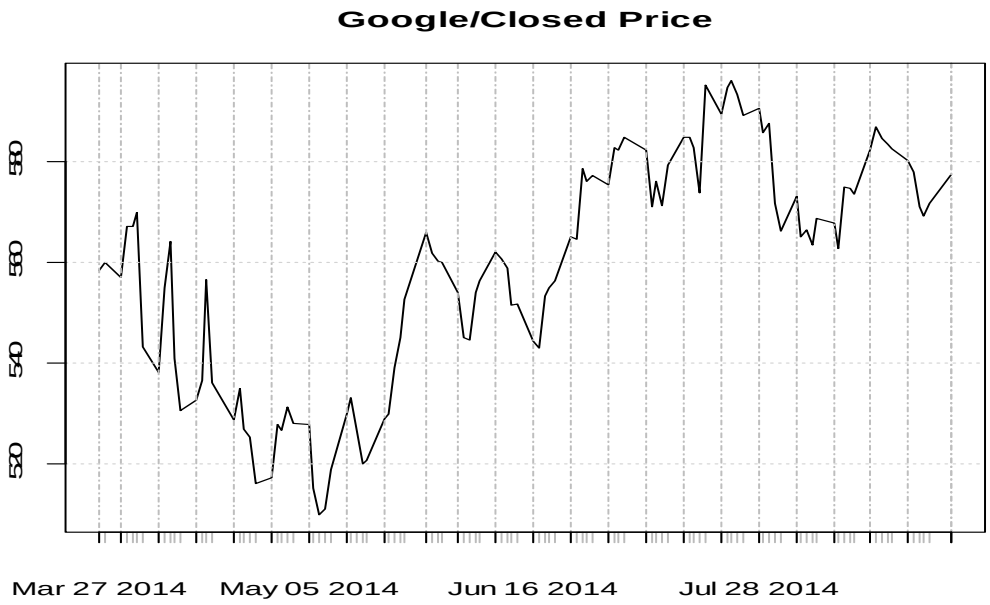
**Example Two: Time Series Data**

Very often we want to analyze economic or financial data that are publicly available at web sites. These data are time series by nature (we record them sequentially, over time). For example,

http://research.stlouisfed.org/fred2/ (Federal Reserve Bank of St. Louis)
http://www.google.com/finance (Google Finance Section)
http://finance.yahoo.com/ (Yahoo Finance Section)

Here is the R code needed to download and plot the historical stock price of Google company:

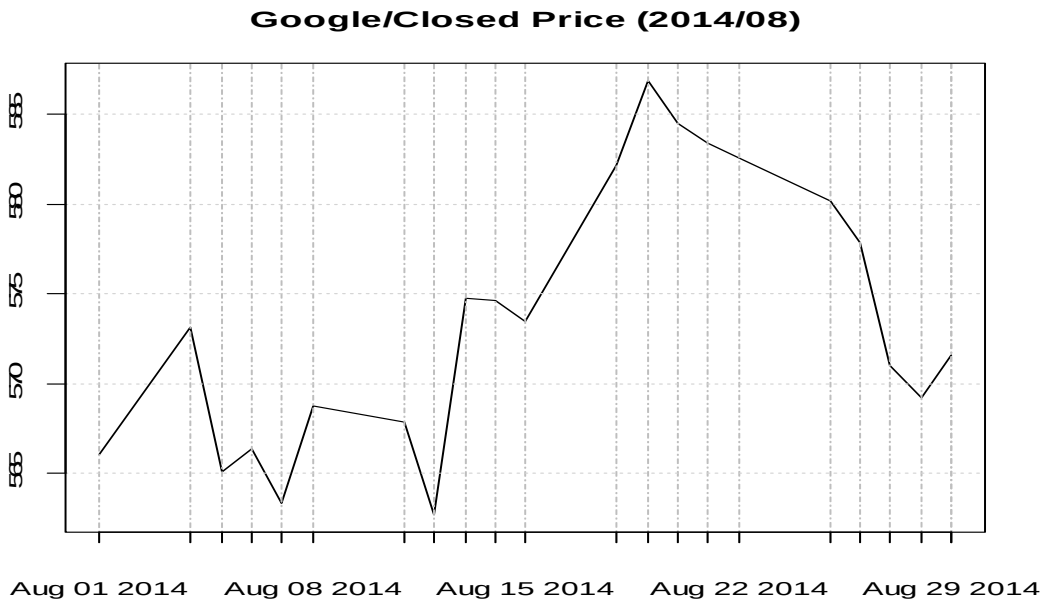
```
install.packages(quantmod) #I will explain what package is in class
library(quantmod) #use "library" to call out the package
getSymbols("GOOG", src = "yahoo") #src = source of the web site
head(GOOG)
plot(GOOG[,4],main="Google/Closed Price") #what does [,4] mean?
```




```
summary(GOOG[,4]) #summarize the data
```

| Index | | GOOG.Close | |
|---|---|---|---|
| Min. | :2014-03-27 | Min. | :510.0 |
| 1st Qu. | :2014-05-06 | 1st Qu. | :539.4 |
| Median | :2014-06-14 | Median | :562.4 |
| Mean | :2014-06-14 | Mean | :558.4 |
| 3rd Qu. | :2014-07-23 | 3rd Qu. | :576.1 |
| Max. | :2014-09-02 | Max. | :596.0 |

```
data.2014.08 = window(GOOG,start=as.Date("2014-08-01"),end=as.Date("2014-08-30"))
dim(data.2014.08)
plot(data.2014.08[,4],main="Google/Closed Price (2014/08)")
```




```
summary(data.2014.08[,4])
```

| Index | | GOOG.Close | |
|---|---|---|---|
| Min. | :2014-08-01 | Min. | :562.7 |
| 1st Qu. | :2014-08-08 | 1st Qu. | :567.9 |
| Median | :2014-08-15 | Median | :573.1 |
| Mean | :2014-08-15 | Mean | :573.6 |
| 3rd Qu. | :2014-08-22 | 3rd Qu. | :580.2 |
| Max. | :2014-08-29 | Max. | :586.9 |

## III. SIMPLE PROGRAMMING

e.g. Population Growth Forecasting

```
rm(list=ls())
N = rep(0,10) #rep(x, n); repeat x n times
N[1] = 300 #aproximate U.S. populaiton in 2014
Time = rep(0,10)
Time[1] = 2014
a = 0.007 #average popultaiton growth rate in the us between 2009 and 2013 (The World Bank data)

#start looping here
for (i in 2:10){

N[i] = (1 + a)*N[i-1] + rnorm(1,10,4)
Time[i] = 2013 + i

}
plot(Time, N, type="b", main = "Population Forescast (2014~2023)",col="red")
```

Use notepad or any text editor to type your code and save it as "R" file. You can execute your code using "source" command.

## IV. HOW TO INSTALL R ON YOUR PC/MAC

The following videos are provided by Dr. Roger Peng from Dept. of Biostatistics at Johns Hopkins University.

Install R on Mac (video at youtube.com):
https://www.youtube.com/watch?v=Icawuhf0Yqo

Install R on Windows (video at youtube.com):
https://www.youtube.com/watch?v=hxj0UG4boGU

Other videos about learning R:
https://www.youtube.com/user/rdpeng/videos