

ECN 3972

Lab 1

Beginning with R on Your Computer

R is a very powerful language that is used in many disciplines across the world. It has the ability to do everything we need with data for this class (and much we will never use!). Best of all, you can get R for free because it is open source. We will be using R within R Studio to make it easier. Both are available across campus on the virtual desktop. But, if you have your own computer, you can download the products easily (no matter whether you use a PC or a Mac).

First, download R at <https://cran.rstudio.com/>

Then, download R Studio at

<https://www.rstudio.com/products/rstudio/download/#download>

You must first install R before you can install R Studio.

The first time you open R Studio, you need to add a particular package called “Mosaic” and a package called “psych.” So, type

```
install.packages("shiny")
```

```
install.packages("mosaic")
```

```
install.packages("psych")
```

```
install.packages("lme4")
```

```
install.packages("sandwich")
```

```
install.packages("AER")
```

Then, after that, you will want to call that package up each time you use R by

```
library(mosaic)
```

```
library(psych)
```

Note: You won't need every package every time.

Basic Math in R

R will compute just about anything you give it mathematically:

```
5+4
```

```
## [1] 9
sqrt(121)
## [1] 11
3^4
## [1] 81
```

Assigning Variables

Suppose we have a variable, x, that we want to set equal to 5

```
x <- 5
```

To know that x is really set at 5 we can check:

```
x
## [1] 5
```

Using a Data Set in R

Let's call up the package mosaic:

```
library(mosaic)
```

In econometrics, we will often be using sets of data with multiple observations (rows) and multiple variables (columns). R has some built-in data sets we can use, but you will eventually have to come up with your own data for your project. There will also be many data sets associated with your homework assignments.

Let's just use a pre-loaded data set for today. So, let's call it up:

```
data(women)
```

This is a data set showing the heights and weights for American women. To see what variables are in the set and what the data look like, we can ask R to show us the first 6 lines of the data:

```
head(women)
##   height weight
## 1     58    115
## 2     59    117
## 3     60    120
## 4     61    123
## 5     62    126
## 6     63    129
```

So we see that there are only two variables. If we just wanted that information, we could ask R to tell us the number of columns in the data:

```
ncol(women)
```

```
## [1] 2
```

We also might want to know how many observations there are:

```
nrow(women)
```

```
## [1] 15
```

So, our sample has 15 observations. We could see the entire data:

```
list(women)
```

```
## [[1]]  
##      height weight  
## 1      58     115  
## 2      59     117  
## 3      60     120  
## 4      61     123  
## 5      62     126  
## 6      63     129  
## 7      64     132  
## 8      65     135  
## 9      66     139  
## 10     67     142  
## 11     68     146  
## 12     69     150  
## 13     70     154  
## 14     71     159  
## 15     72     164
```

If we want specific information about a particular variable, we have to be careful how we tell R to select it. For example, suppose I want to know the average height in our sample. I must type:

```
mean(women$height)
```

```
## [1] 65
```

We must first tell R what data to use and then what variable to use, separating the two with a dollar sign.

Note that we could have also gotten the average height by summing the height column and then dividing by the number of observations (rows):

```
avgheight <- (sum(women$height))/nrow(women)  
avgheight
```

```
## [1] 65
```

You can calculate the minimum, maximum, variance, and standard deviation in the same way:

```
min(women$height)
## [1] 58

max(women$height)
## [1] 72

var(women$height)
## [1] 20

sd(women$height)
## [1] 4.472136
```

We can get important information about all of the variables at once:

```
summary(women)

##      height      weight
##  Min.   :58.0   Min.   :115.0
##  1st Qu.:61.5   1st Qu.:124.5
##  Median :65.0   Median :135.0
##  Mean   :65.0   Mean    :136.7
##  3rd Qu.:68.5   3rd Qu.:148.0
##  Max.   :72.0   Max.    :164.0
```

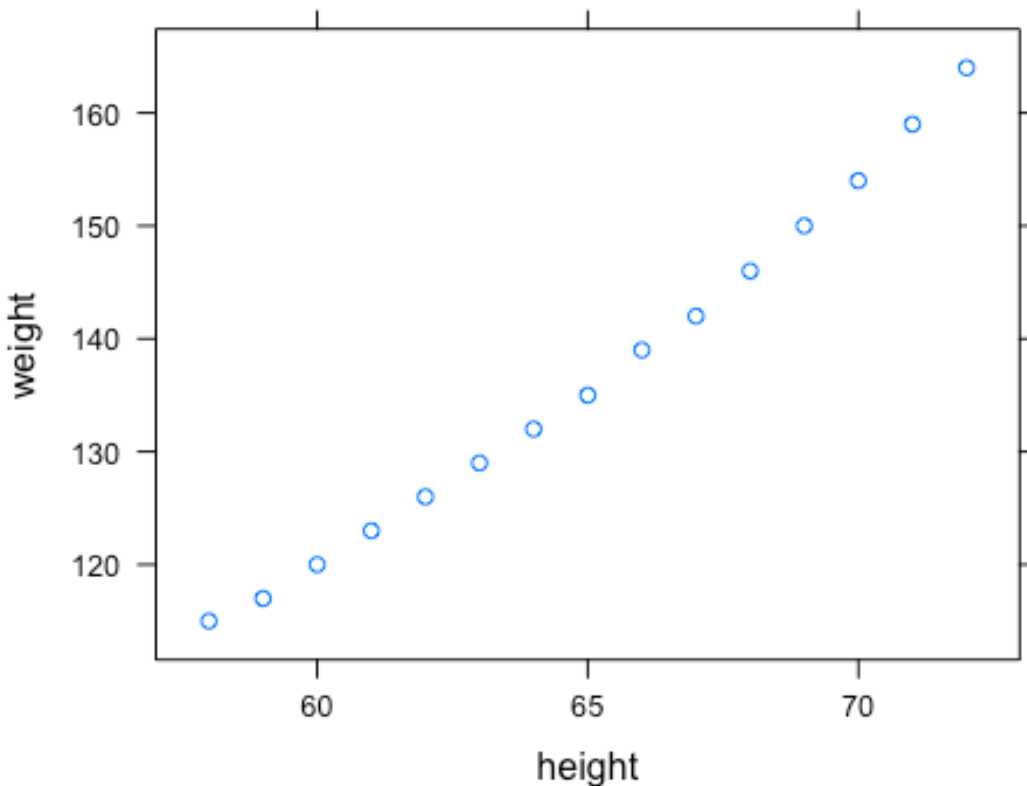
Or another way to get a description is:

```
describe(women)

##      vars  n   mean    sd median trimmed   mad min max range skew
## height   1 15  65.00  4.47    65   65.00   5.93  58  72    14  0.00
## weight   2 15 136.73 15.50   135  136.31  17.79 115 164    49  0.23
##      kurtosis   se
## height   -1.44  1.15
## weight   -1.34  4.00
```

Last, you can plot the data:

```
xyplot(weight~height, data = women)
```



Sampling

You can take a large data set, and sample any number of observations from it.

Let's begin by loading a data set. I have preloaded several data sets in the “thaw space” on the lab computers. Import `life.xls`. Note that this is an Excel file. You need to choose that option in R.

After it loads, you should see that it has 119 observations of 4 variables. Let's find out what the four variables are:

```
head(life)
```

```
## # A tibble: 6 x 4
##   LifeEx   TV PopDoc   GDP
##   <dbl> <dbl> <dbl> <dbl>
## 1  65.6   7.4  2330  2870
## 2  71.1  22.2   330  5120
## 3  76.7  49     440 16680
## 4  75.7  47     230 17690
## 5  71     41.4   930 11536
## 6  52.2   0.5   6670  1160
```

So we have Life Expectancy, % of homes with a TV, population, and GDP. Let's generate descriptive statistics for these 119 countries:

```
describe(life)

##      vars   n   mean      sd median trimmed   mad   min   max
## LifeEx    1 119   64.61   10.53   67.2   65.35  12.01  42.4   78.6
## TV        2 119   17.55   18.56    9.7   14.94  13.34   0.1   81.0
## PopDoc    3 119 7949.33 13959.14 1610.0 4645.98 1838.42 210.0 71430.0
## GDP       4 119 6433.60 6156.83 3683.0 5629.63 3948.16 370.0 22130.0
##           range skew kurtosis      se
## LifeEx    36.2 -0.57   -0.99    0.97
## TV        80.9  1.18    0.70    1.70
## PopDoc 71220.0  2.63    7.16 1279.63
## GDP     21760.0  1.07   -0.19  564.40
```

Now, let's use only 50 of the observations. To draw a random sample from "life" and name it "newsample," we use the following:

```
newsample <- life[sample(1:nrow(life), 50, replace=FALSE),]
```

Now, let's look at our newsample and see what it looks like:

```
nrow(newsample)

## [1] 50

ncol(newsample)

## [1] 4
```

So, we have a new data set with 50 observations and 4 variables. Now, we can run descriptive statistics for this data:

```
describe(newsample)

##      vars   n   mean      sd median trimmed   mad   min   max
## LifeEx    1 50   63.12   12.09   67.55   63.66  13.71  42.6   77.7
## TV        2 50   17.43   19.85    7.30   14.48  10.08   0.1   81.0
## PopDoc    3 50 9863.20 15148.27 1855.00 6457.25 2231.31 210.0 66670.0
## GDP       4 50 6619.70 6748.82 3419.00 5833.68 3944.46 480.0 22130.0
##           range skew kurtosis      se
## LifeEx    35.1 -0.32   -1.55    1.71
## TV        80.9  1.16    0.57    2.81
## PopDoc 66460.0  1.95    3.39 2142.29
## GDP     21650.0  0.85   -0.86  954.43
```