Homework_02 ®

Lilit Hovsepyan

28 09 2019

Problem 1.

c. *Simulate 100 Die Roll.*

1) Frequency Table

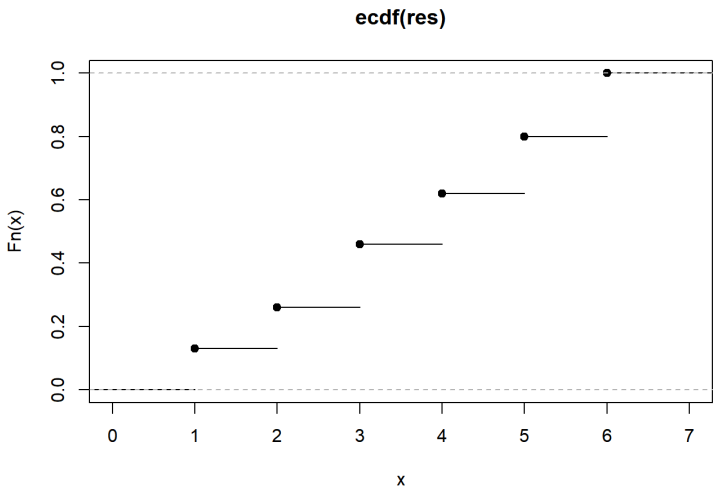
2) ECDF

3) BarPlot

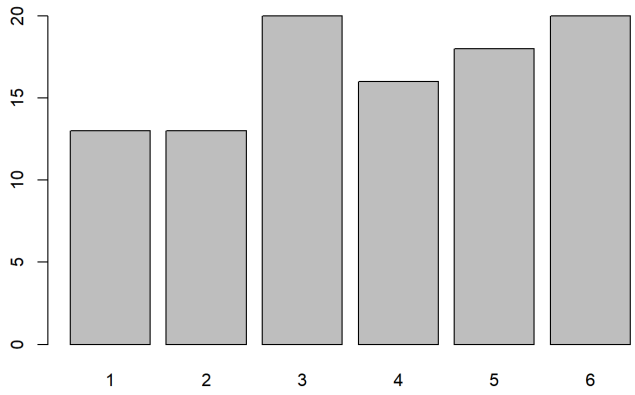
```
# 1.Frequency table
res <- c(sample(1:6, 100, replace = T))
y <- table(res)
y
```

```
## res
##  1  2  3  4  5  6
## 24 26 15 10 12 13
```

```
# 2.ECDF
res <- c(sample(1:6, 100, replace = T))
f <- ecdf(res)
plot(f)
```



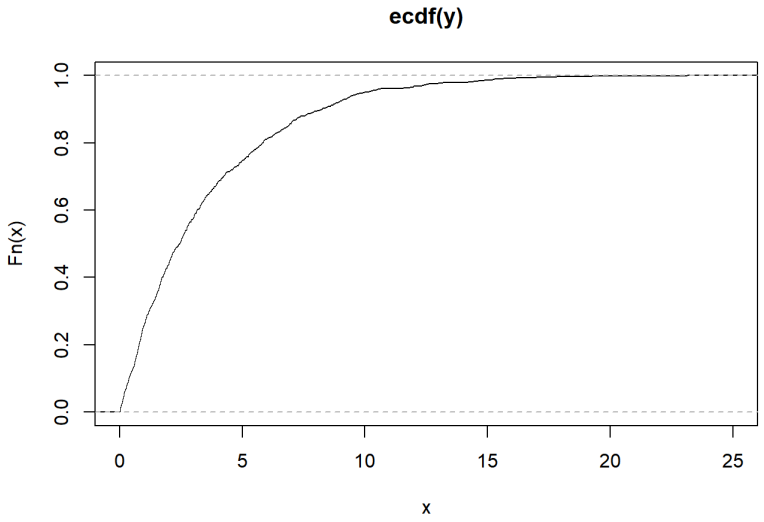
```
# 3.BarPlot
barplot(table(res))
```



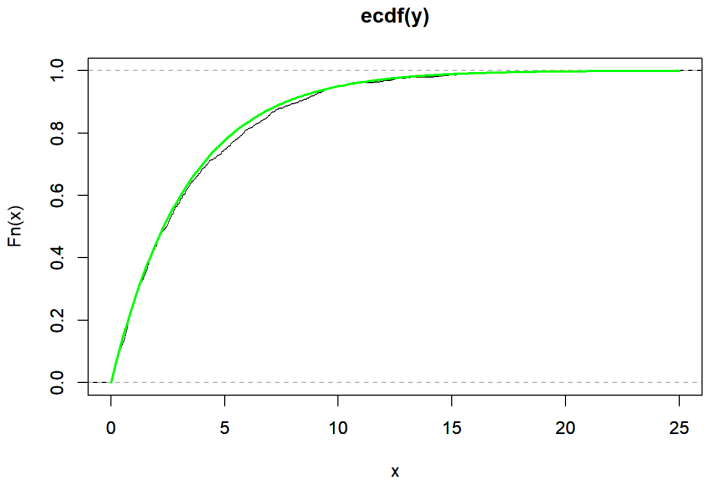
d. *Check that ECDF approximates well the CDF behind the data.*

1. Generate 1000 samples from the Exp(0.3) distribution
2. Plot the ECDF of the result
3. Plot over the previous graph the theoretical CDF of the distribution, with green color and linewidth 2 (use the lwd=2 parameter value in plot)

```
# 1.Gen. the 1000 samples from the Exp(0.3) dist.
y<-rexp(1000,rate = 0.3)
head (y)
## [1]  2.0757403  1.7478802  0.8111074  1.9869507  3.4230268 12.5701396
# 2.Plot the ECDF of the result.
yy<-ecdf(y)
plot(yy, xlim=c(0,25), ylim=c(0,1))
```




```
# 3.Plot over the previous graph the theoretical CDF.
yy<-ecdf(y)
plot(yy, xlim=c(0,25), ylim=c(0,1))
par(new=T)
range = seq(0,25,0.1)
plot(range, pexp(range, rate = 0.3), lwd = 2, col = "green", xlim=c(0,25), ylim=c(0,1), type="l", ylab = " ", xlab = " ")
```



Problem 2.

b. *Consider one of the standard Datasets in R, islands.*

1. Call the help page for this Dataset to see the description,
2. Print the structure of the Dataset,
3. Print the head of this Dataset,
4. Plot the Frequency Histogram for the islands with the area less than 200 sq miles,
5. Plot the Density Histogram for the islands with the area less than 200 sq miles,
6. Add to the previous plot the KDE (in red, with linewidth 3) for the islands with the area less than 200 sq miles

```
# 1.Call the help page for this Dataset to see the description.
help(islands)
```

```
## starting httpd help server ... done
```

```
# 2.Print the structure of the Dataset.
structure(islands)
```

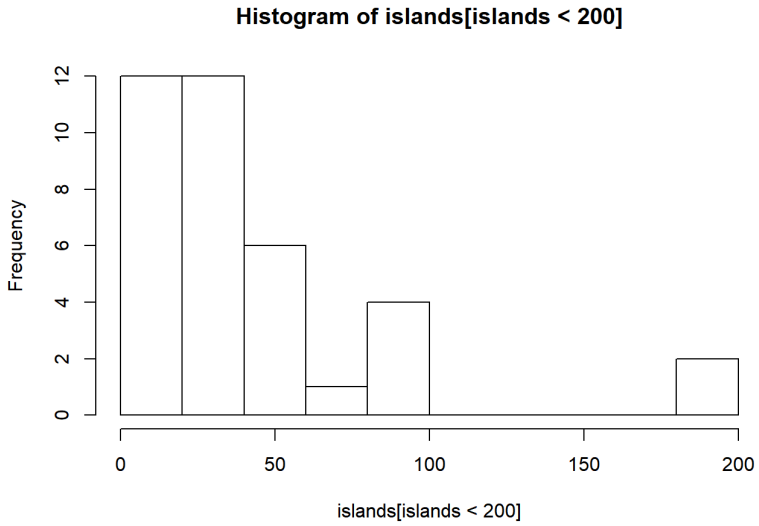
```
##         Africa      Antarctica           Asia      Australia
##          11506            5500          16988           2968
##    Axel Heiberg          Baffin          Banks         Borneo
##             16             184             23            280
##        Britain         Celebes          Celon           Cuba
##             84              73             25             43
##         Devon       Ellesmere         Europe      Greenland
##             21              82           3745            840
##        Hainan      Hispaniola        Hokkaido         Honshu
##             13              30             30             89
##       Iceland         Ireland           Java         Kyushu
##             40              33             49             14
##        Luzon       Madagascar        Melville       Mindanao
##             42             227             16             36
##      Moluccas      New Britain      New Guinea  New Zealand (N)
##             29              15            306             44
##  New Zealand (S)     Newfoundland    North America    Novaya Zemlya
##             58              43           9390             32
##  Prince of Wales        Sakhalin    South America      Southampton
##             13              29           6795             16
##     Spitsbergen        Sumatra         Taiwan        Tasmania
##             15             183             14             26
```

```
## Tierra del Fuego         Timor       Vancouver       Victoria
##           19            13            12            82
```

```
# 3.Print the head of this Dataset.
head(islands)
```

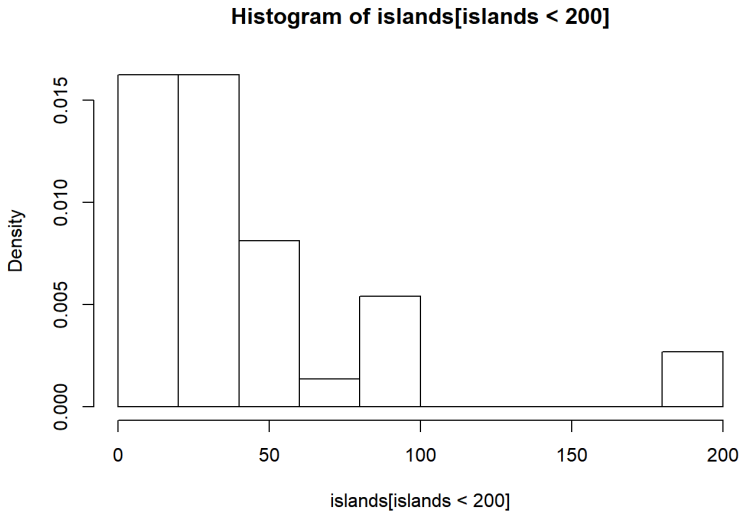
```
##      Africa   Antarctica       Asia    Australia Axel Heiberg
##       11506         5500      16988       2968         16
##      Baffin
##         184
```

```
# 4.plot the Frequency Histogram for the islands with the area less than 200 sq miles.
hist(islands[islands<200])
```

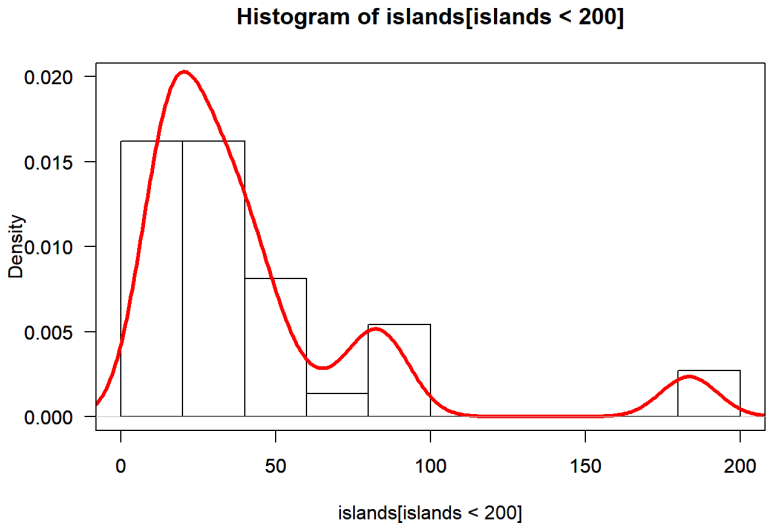



```
# 5.plot the Density Histogram for the islands with the area less than 200 sq miles.
```

```
hist(freq=F, islands[islands<200])
```



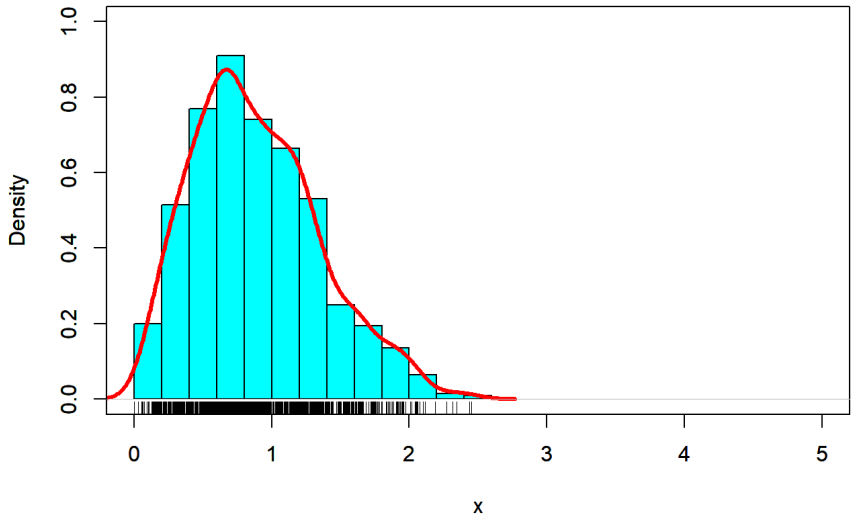

```
# 6.Add to the previous plot the KDE (in red, with linewidth 3) for the islands with the
hist(freq=F, islands[islands<200], ylim = c(0,0.02), las=1)
par(new=T)
kde<-density(islands[islands<200])
plot(kde, col="red", lwd=3, main="", ylab = "", xlab = "", xlim=c(0,200), las=1, ylim=c(0,0.02))
```



c. *Here we want to check that the Density Histogram is approximating well the PDF behind the data. To that end, we consider the Weibull distribution*

1. Take n = 1000
2. Generate a sample of size n from the Weibull distribution with the shape parameter 2
3. Plot the Density Histogram of that sample, in cyan color
4. Plot the theoretical PDF (use dweibull in R) over the previous graph, in red, and with linewidth 3

```
# 1.- 4.
n<-1000
x <- rweibull(n,2,1)
d <- density(x)
hist(x, freq = F, xlim = c(0,5), ylim = c(0,1), col = "cyan", main = " ")
rug(x)
par(new=T)
plot(d, lwd = 3, col = "red", xlim = c(0,5), ylim = c(0,1), main = " ", xlab = " ")
```



So, we can see from the graph that Density Histogram is approximating well the PDF behind the data.

d. *Now let's plot comparative Histograms. We will work with the R-s default ChickWeight Dataset.*

1. Explore the Dataset: read the description and print the first 5 rows of that Dataset
2. Separate in x the Weight variable for all Chicken with the Diet 1
3. Separate in y the Weight variable for all Chicken with the Diet 2
4. Plot the Frequency Histograms of x and y one over another.

```
# 1. Explore the Dataset: read the description and print the first 5 rows of that Dataset
help(ChickWeight)
```

```
head(ChickWeight,5)
```

```
##   weight Time Chick Diet
## 1    42    0    1    1
## 2    51    2    1    1
## 3    59    4    1    1
## 4    64    6    1    1
## 5    76    8    1    1
```

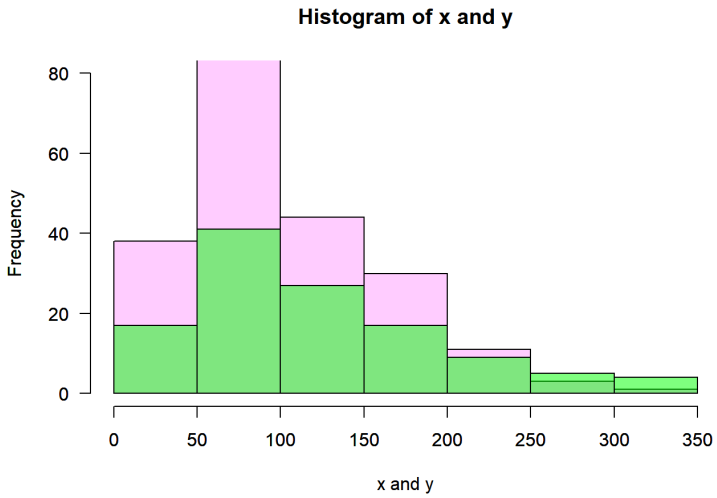
```
# 2. Separate in x the Weight variable for all Chicken with the Diet 1
x<-ChickWeight$weight[ChickWeight$Diet==1]
head(x)
```

```
## [1] 42 51 59 64 76 93
```

```
# 3. Separate in y the Weight variable for all Chicken with the Diet 1
y<-ChickWeight$weight[ChickWeight$Diet==2]
head(y)
```

```
## [1]  40  50  62  86 125 163
```

```
# 4. Plot the Frequency Histograms of x and y one over another
library(scales)
hist(x, col=alpha("magenta",0.2), las=1, ylim = c(0,80), xlab = " ", main = " ")
par(new=T)
hist(y, col=alpha("green",0.5), las=1, ylim = c(0,80), xlab = "x and y", main = "Histogram of x and y")
```



Problem 3.

c. *Consider the iris Dataset.*

1. Choose the Petal.Length variable and make its S-n-L Plot
2. Now do the same variable S-n-L Plot with the scale parameters = to 0.5, 2 and 4
3. (Supplementary) Now plot the S-n-L Plot and Histogram of our Dataset side-by-side.

```
# 1. Choose the Petal.Length variable and make its S-n-L Plot
a<-iris$Petal.Length
stem(a)
```

```
##
##   The decimal point is at the |
##
##   1 | 01223333333444444444444
##   1 | 55555555555556666666777799
##   2 |
##   2 |
##   3 | 033
##   3 | 55678999
##   4 | 000001112222334444
##   4 | 55555555666777778888899999
##   5 | 000011111111223344
##   5 | 55566666677788899
##   6 | 0011134
##   6 | 6779
```

```
# 2. Now do the same variable S-n-L Plot with the scale parameters = to 0.5, 2 and 4
a<-iris$Petal.Length
stem(a, scale=0.5)
```

```
##
##   The decimal point is at the |
##
##   1 | 01223333333444444444444455555555555556666666777799
##   2 |
##   3 | 03355678999
##   4 | 00000111222233444455555555666777778888899999
##   5 | 00001111111122334455566666677788899
##   6 | 00111346779
```

```
stem(a, scale=2)
```

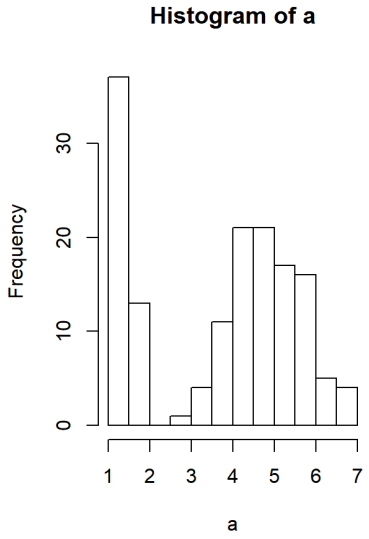
```
##
```

```
##   The decimal point is 1 digit(s) to the left of the |
##
##   10 | 00
##   12 | 000000000
##   14 | 00000000000000000000000000
##   16 | 00000000000
##   18 | 00
##   20 |
##   22 |
##   24 |
##   26 |
##   28 |
##   30 | 0
##   32 | 00
##   34 | 00
##   36 | 00
##   38 | 0000
##   40 | 00000000
##   42 | 000000
##   44 | 000000000000
##   46 | 00000000
##   48 | 000000000
##   50 | 000000000000
##   52 | 0000
##   54 | 00000
##   56 | 000000000
##   58 | 00000
##   60 | 00000
##   62 | 0
##   64 | 0
##   66 | 000
##   68 | 0
stem(a, scale=4)
##
##   The decimal point is 1 digit(s) to the left of the |
```

```
##
##  10 | 0
##  11 | 0
##  12 | 00
##  13 | 0000000
##  14 | 0000000000000
##  15 | 0000000000000
##  16 | 0000000
##  17 | 0000
##  18 |
##  19 | 00
##  20 |
##  21 |
##  22 |
##  23 |
##  24 |
##  25 |
##  26 |
##  27 |
##  28 |
##  29 |
##  30 | 0
##  31 |
##  32 |
##  33 | 00
##  34 |
##  35 | 00
##  36 | 0
##  37 | 0
##  38 | 0
##  39 | 000
##  40 | 00000
##  41 | 000
##  42 | 0000
##  43 | 00
```

```
##   44 | 0000
##   45 | 00000000
##   46 | 000
##   47 | 00000
##   48 | 0000
##   49 | 00000
##   50 | 0000
##   51 | 00000000
##   52 | 00
##   53 | 00
##   54 | 00
##   55 | 000
##   56 | 000000
##   57 | 000
##   58 | 000
##   59 | 00
##   60 | 00
##   61 | 000
##   62 |
##   63 | 0
##   64 | 0
##   65 |
##   66 | 0
##   67 | 00
##   68 |
##   69 | 0
```

```
# 3. Now plot the S-n-L Plot and Histogram of our Dataset side-by-side.
par(mfcol=c(1,2))
hist(a)
a <-iris$Petal.Length
plot.new()
out <- capture.output(stem(a))
text(0,1,paste(out, collapse = '\n'), adj=c(0,1), family='mono')
```




```
The decimal point is

1 | 0122333333344444
1 | 5555555555555666
2 |
2 |
3 | 033
3 | 55678999
4 | 0000011122223344
4 | 5555555566677777
5 | 0000111111112233
5 | 5556666667778889
6 | 0011134
6 | 6779
```

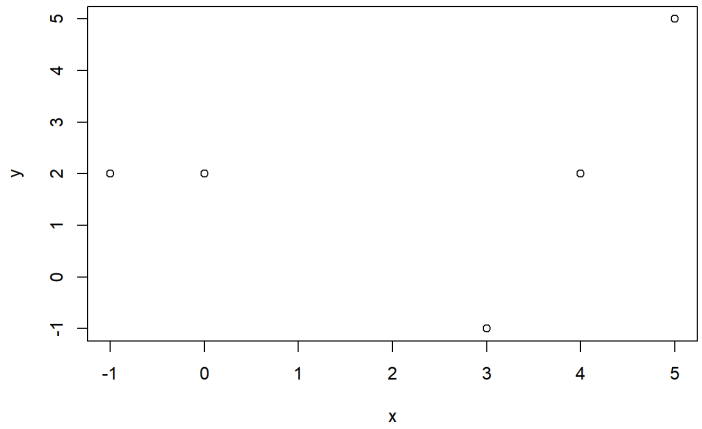
Problem 4.

a. *Plot the following points: (0, 2), (3,-1), (4, 2), (5, 5), (-1, 2)*

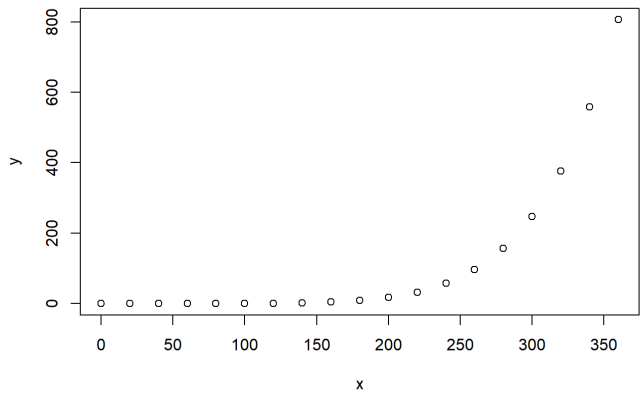
b. *R-s pressure Dataset consists of 2 Variables. Give the ScatterPlot of these Variables.*

```
# a.
```

```
x<-c(0,3,4,5,-1)
y<-c(2,-1,2,5,2)
plot(x,y)
```




```
# b.
help(pressure)
x<-pressure$temperature
y<-pressure$pressure
plot(x,y)
```



Problem 5.

```
aapl<-read.csv(file.choose())
Adj_close <- aapl$Adj.Close
head(aapl)
```

```
##         Date    Open    High     Low   Close Adj.Close    Volume
```

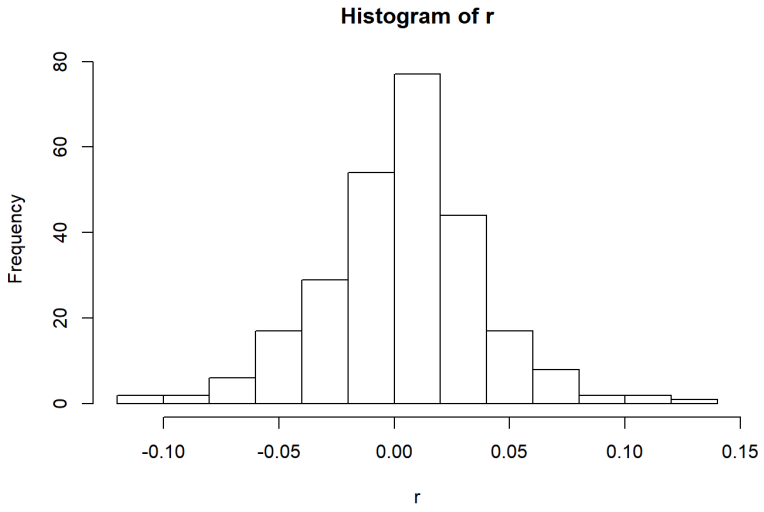
```
## 1 2014-09-22 101.80 102.94  97.72 100.75  92.44788 338824900
## 2 2014-09-29  98.65 101.54  98.04  99.62  91.41103 247749100
## 3 2014-10-06  99.95 102.38  98.31 100.73  92.42956 280258200
## 4 2014-10-13 101.33 101.78  95.18  97.67  89.62170 358539800
## 5 2014-10-20  98.32 105.49  98.22 105.22  96.54954 358532900
## 6 2014-10-27 104.85 108.04 104.70 108.00  99.10047 220230600
```

```
r<-(Adj_close[2:length(Adj_close)]-Adj_close[-length(Adj_close)])/Adj_close[-len
gth(Adj_close)]
```

```
head(r)
```

```
## [1] -0.011215509  0.011142332 -0.030378399  0.077301025  0.026420901
## [6]  0.009351863
```

```
hist(r)
```



Here we see that the returns are almost closer to 0 both form the left and from the right sides. And most of the datapoints are in the center.

Problem 6.

f. *ChickWeight Dataset from R.*

1. Calculate the Mean of Wights for chicken fed with the first diet
2. Calculate the Mean of Wights for chicken fed with the second diet

3. Compare the results: can the difference between the means be the result of just randomness, or we can state that one of the diets is better than the other one?

```
# 1. Calculate the Mean of Wights for chicken fed with the first diet
w1<-ChickWeight$weight[ChickWeight$Diet==1]
mean(w1)
## [1] 102.6455
# 2. Calculate the Mean of Wights for chicken fed with the second diet
w2<-ChickWeight$weight[ChickWeight$Diet==2]
mean(w2)
## [1] 122.6167
122.6167-102.6455
## [1] 19.9712
```

3. Here we see that the chicken mean weight in case of the first type of experimental diet is 102.6455, for the second type it is higher by 19.9712. Maybe the second type of diet was better than the first one, and on average chicks started to get higher weight.

Problem 7.

d. Calculate and compare the Sample Standard Deviations and Variances for the mpg variable from the Dataset mtcars for different cylinder type cars. For example, compare 6 cylinder cars mpg-s SD with the 4 cylinder cars mpg-s SD.

```
a <- mtcars$mpg[which(mtcars$cyl==6)]
var(a)
## [1] 2.112857
sd(a)
## [1] 1.453567
b <- mtcars$mpg[which(mtcars$cyl==4)]
var(b)
## [1] 20.33855
sd(b)
## [1] 4.509828
sd(b)-sd(a)
## [1] 3.056261
```

e. We consider the iris Dataset. For which type of the flower (for which Species) the variability in Petal.Width is maximal, and for which is minimal.

```
x <- iris$Petal.Width
y <- x[which(var(x) == max(var(x)))]
iris$Species[which(iris$species == y)]
## factor(0)
## Levels: setosa versicolor virginica
z <- x[which(var(x) == min(var(x)))]
iris$Species[which(iris$species == z)]
## factor(0)
## Levels: setosa versicolor virginica
```

f.1) Calculate the Median Absolute Deviation from the Median for the dist variable of the cars Dataset

```
mad(cars$dist)
## [1] 23.7216
```

f.2) Calculate the Median Absolute Deviation from the Mean for the dist variable of the cars Dataset.

```
mad(cars$dist, center = mean(cars$dist))
## [1] 25.17455
```

f.3) Write a function mad1 which will calculate the Mean Absolute Deviation from the Mean.

```
y <- cars$dist - mean(cars$dist)
```

```
mad1 <- mean(abs(y))
mad1
```

```
## [1] 20.6968
```

f.4) Write a function mad2 which will calculate the Mean Absolute Deviation from the Median.

```
x <- cars$dist - median(cars$dist)
mad2 <- mean(abs(x))
mad2
```

```
## [1] 20.14
```

Problem 8.

c.

1. Calculate the Quartiles of x and y form part a. by using the quantile function.

```r
# 1. Calculate the Quartiles of x and y form part a. by using the quantile function.
x <- c(-6, 15, 0, 5, 17, -4, 1, -9, -9, 13)
y <- c(0.0, 3.6, 2.7, -1.5, 5.7, 1.5, -3.0, 4.5, 6.0)

q1_x<-quantile(x, 0.25)
q2_x<-median(x)
q3_x<-quantile(x, 0.75)
q1_x
##  25%
## -5.5
q2_x
## [1] 0.5
q3_x
## 75%
##  11
q1_y<-quantile(y, 0.25)
q2_y<-median(y)
q3_y<-quantile(y, 0.75)
q1_y
## 25%
##   0
q2_y
## [1] 2.7
q3_y
## 75%
## 4.5
```

2. Write an R function quartile(x) which will return the Quartiles of the input vector x just like we have defined.

```r
quartile <- function(x){
 q2 <- median(x)
 q1 <- median(x[which(x<=q2)])
 q3 <- median(x[which(x>=q2)])
 q <- c(q1, q2, q3)
 return(q)
```

```
}


quartile(x)
```

```
## [1] -6.0  0.5 13.0
```