# PREDICTIVE ANALYTICS

*MSC INFORMATION SYSTEMS AND SERVICES*

*SPECIALIZATION: BIG DATA AND ANALYTICS*

*R EXERCISES*

*PART 2*


*STUDENT*

*PANAGIOTAKOPOULOS GEORGIOS (ME2030)*

*SUPERVISOR*

*FILIPPAKIS MICHAEL*


Deadline: 08/07/2021

# Table of Contents

## Exercise 1

Enter the **rpart** package. Upload kyphosis data. Describe the dataset. Make the boxplot for the "Number" variable and then find the outliers (their values)
Which rows correspond to the specific data (%in% - which)
Repeat the last one with the (Identify) function

### Solution

1) The "kyphosis" data from the **rpart** package contains the following variables:

- ***Kyphosis*** = a factor with "absent, present" levels that indicates whether kyphosis (a type of deformity) was present or absent after surgery.
- ***Age*** = years (in months).
- ***Number*** = the number of vertebrae involved.
- ***Start*** = the number of the first (highest) vertebra activated.

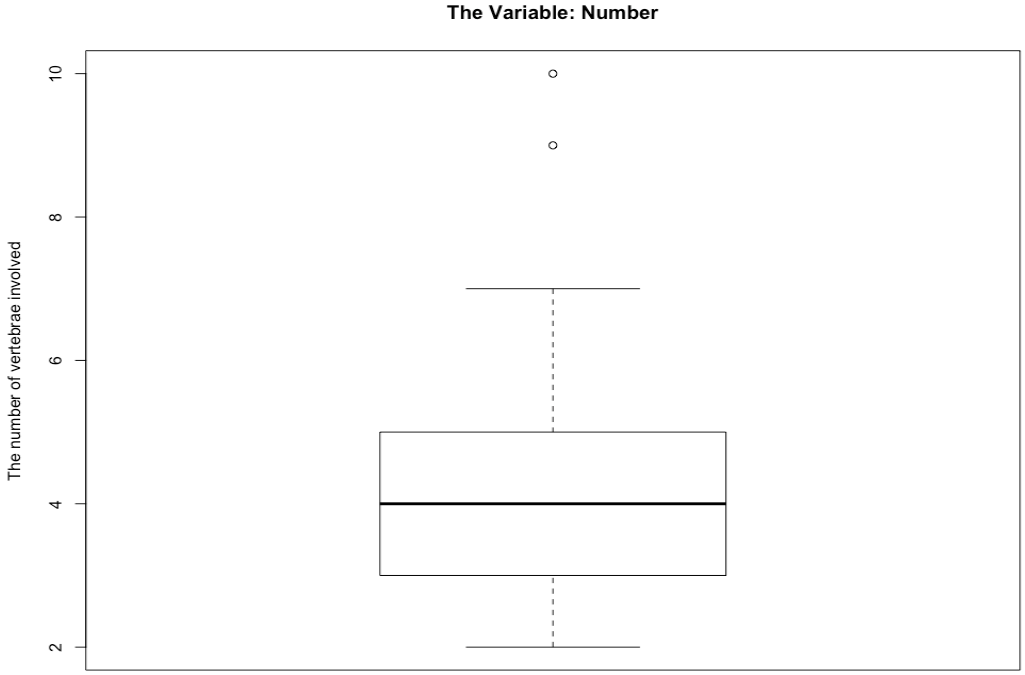
**Descriptive statistics of the dataset**

```
> summary(df)
    Kyphosis       Age              Number          Start
 absent :64   Min.   :  1.00   Min.   : 2.000   Min.   : 1.00
 present:17   1st Qu.: 26.00   1st Qu.: 3.000   1st Qu.: 9.00
              Median : 87.00   Median : 4.000   Median :13.00
              Mean   : 83.65   Mean   : 4.049   Mean   :11.49
              3rd Qu.:130.00   3rd Qu.: 5.000   3rd Qu.:16.00
              Max.   :206.00   Max.   :10.000   Max.   :18.00
```

In the descriptive statistics of the data, according to the results that appear from the above code command (summary (df)), out of a total of 81 results, after the operation 17 people show kyphosis while the remaining 64 do not show.

**Boxplot for the "Number" variable**

```
>boxplot(df$Number, main="The Variable: Number",  ylab="The number of vert
ebrae involved")
```




**Outliers**

To display the outliers we enter the following 2 commands and display the relevant result.

```
> outliers = df$Number[df$Number>8]
> outliers
[1]  9 10
```

```
> which( df$Number %in% outliers)
[1] 43 53
```

We have two outliers with the values 9 and 10 which correspond to lines 43 and 53 as shown in the above boxplot and from the **which & identity** commands.

We execute the commands:

```
> plot(df$Number, df$Age)
> identify(df$Number, df$Age)
```

And in this way we execute the identify function with values of x-axis and y-axis as arguments in the scatter plot. With the identify () function by clicking the mouse, above the points in the specific graph, the serial number for the point

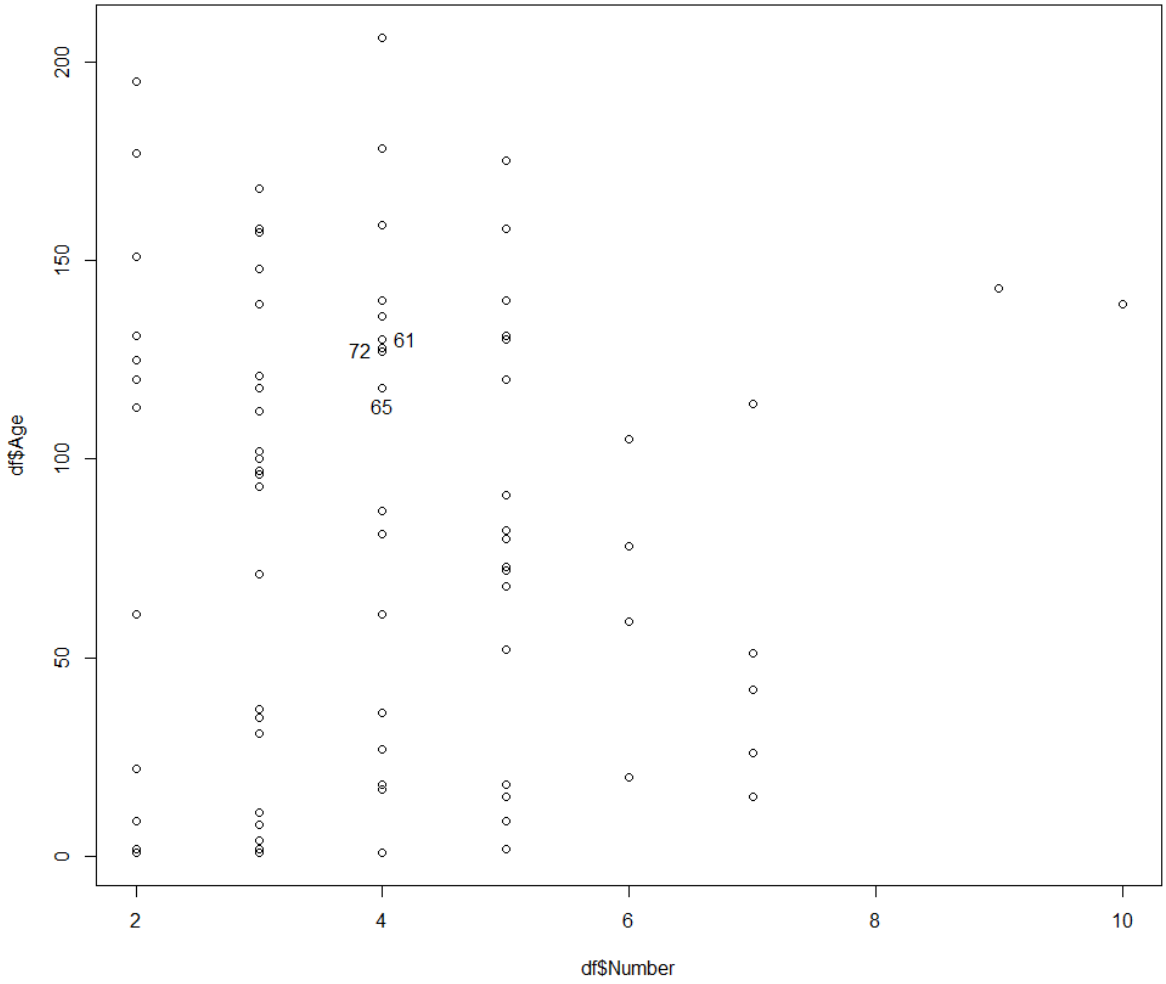
where it was clicked will be displayed. This can continue until we choose to pause.

```
> identify(df$Number, df$Age)
warning: nearest point already identified
warning: nearest point already identified
warning: nearest point already identified
```

In case of clicking on points outside the diagram, the relevant warning will be displayed:

```
warning: no point within 0.25 inches
```

The relevant results are shown in the image below.




## Exercise 2

Consider the capital.csv dataset

i. Make the graphs of Balance in relation to Gender (table of relative frequencies, bar-pie chart)
ii. The boxplot of our data and the boxplot in relation to “gender”
iii. Calculate the central trend and deviation measures
iv. Examine whether our data comes from a normal distribution (e.g. Do Q-Q-plot)

## Solutions

**dataset capital.csv.**

**Descriptive Statistics of data:**

```
> summary(df)
    balance       gender
 Min.   :  99.0   1:232
 1st Qu.: 550.8   2: 68
 Median : 737.0
 Mean   : 753.7
 3rd Qu.: 964.2
 Max.   :1493.0
```

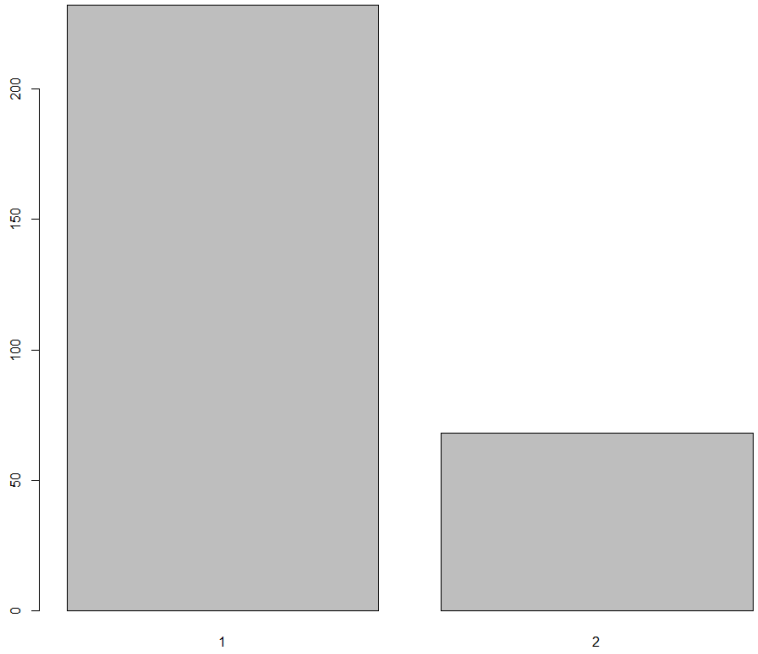
We observe that the capital.csv data file contains a total of 300 people, of which 232 are men and 68 are women..

## 2 i

table of relative frequencies of Balance in relation to Gender:

```
> xtabs(balance ~ gender , data=df)
gender
     1      2
173191  52913
```

```
> barplot( table(df$gender) )
```

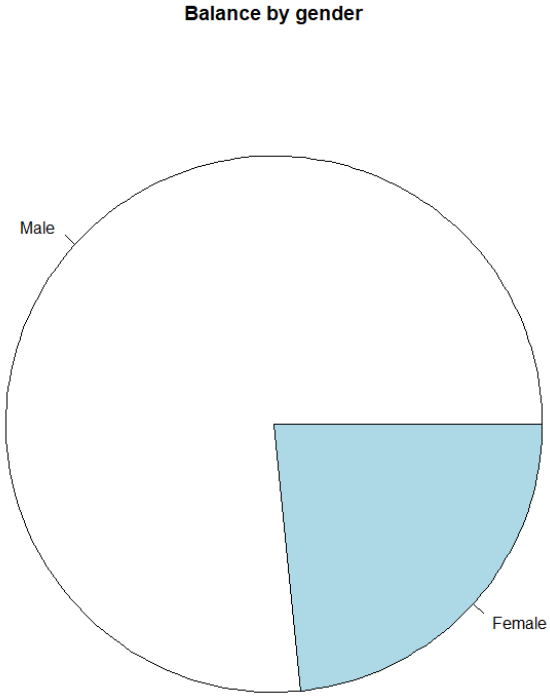



```
> prop.table(table(df$gender))

        1         2
0.7733333 0.2266667
```

From the command of the above code, we observe that men and women correspond to 77.333% and 22.666% respectively.

# Simple Pie Chart

```
> bal_gender <- c(173191, 52913)
> lbls <- c("Male", "Female")
> pie(bal_gender, labels = lbls, main="Balance by gender")
```

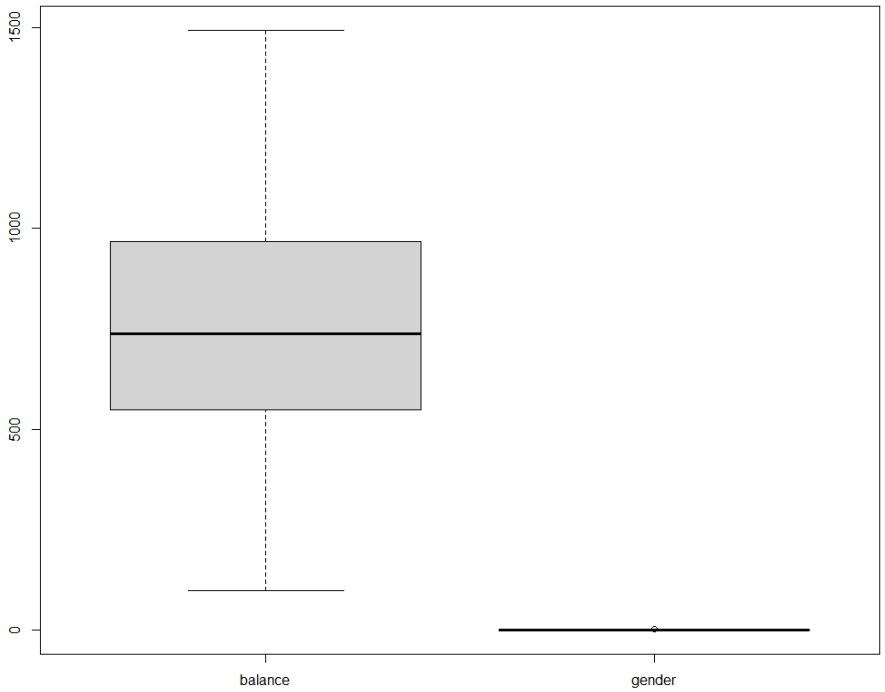



---

## 2 ii

The boxplot of our data is displayed with the following command:

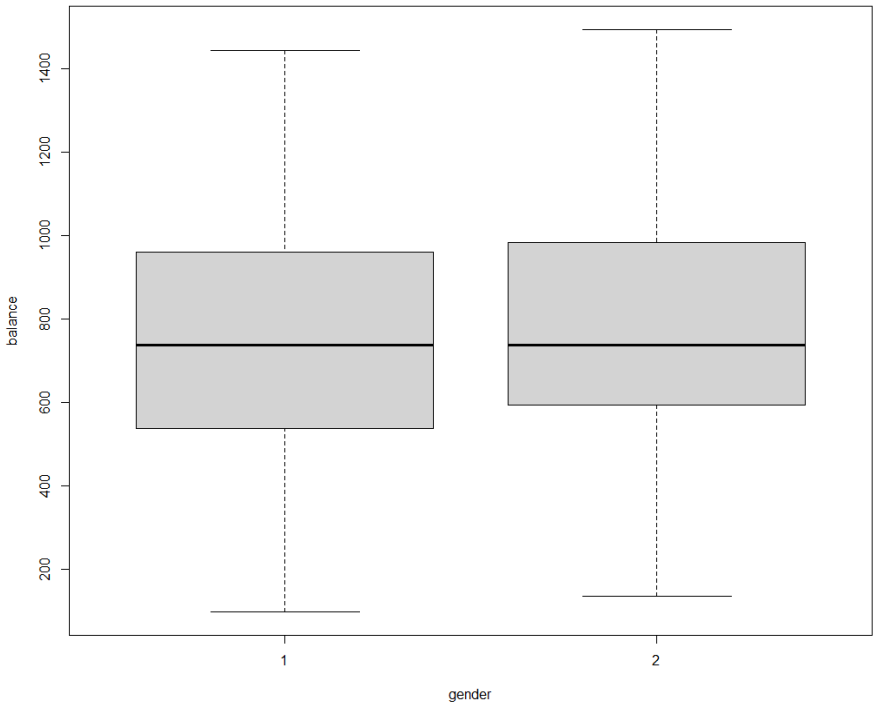
```
> boxplot(df)
```




#2 ii

The boxplot in relation to Gender is displayed by the following command:

```
> boxplot(balance ~ gender, data=df)
```



2 iii

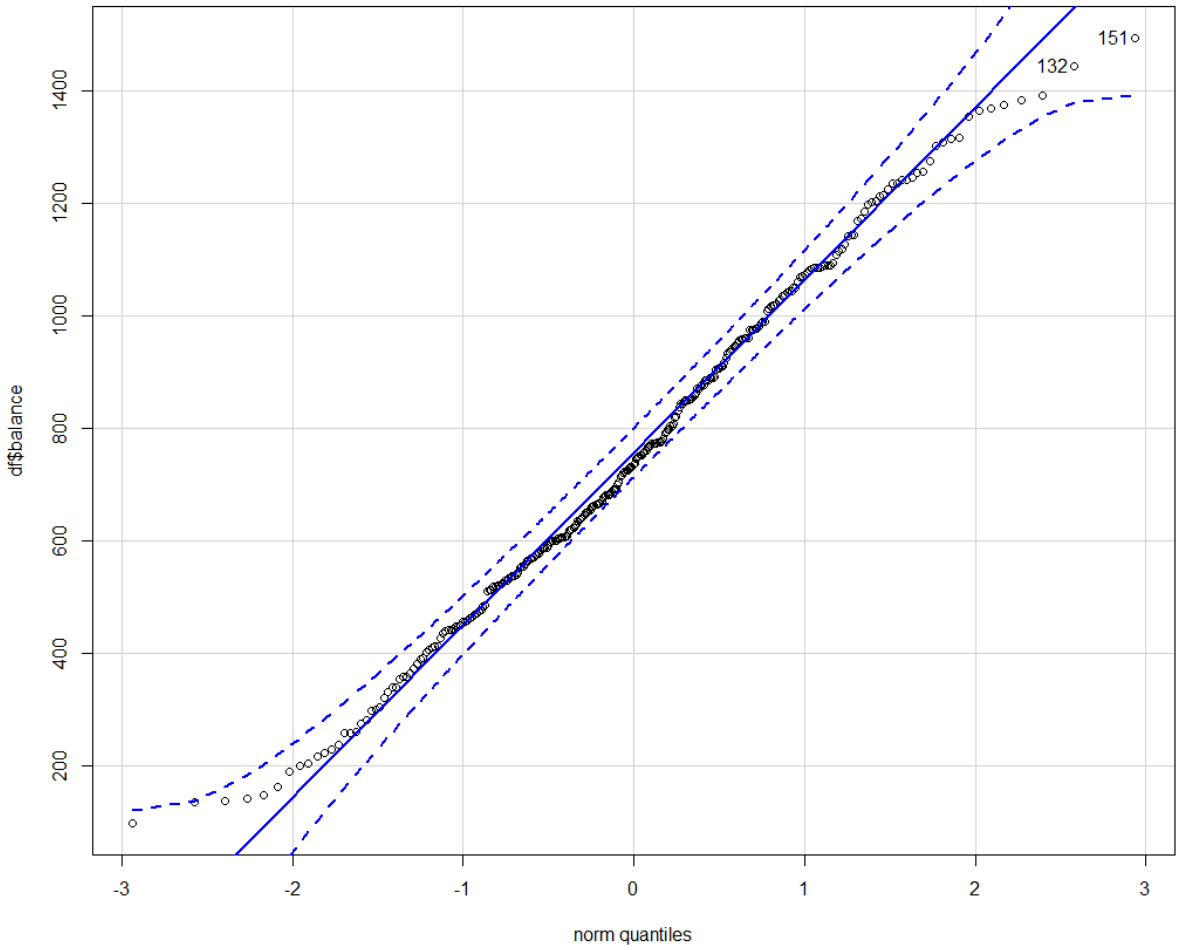
**Central Trend and Deviation Measures**

To display the results from the central trend and deviation measures, run the describeBy() command.

```
> library(psych)
> describeBy(df$balance, df$gender)

 Descriptive statistics by group
group: 1
   vars   n   mean     sd median trimmed    mad min  max range skew kurtos
is    se
X1    1 232 746.51 294.52  738.5  743.88 312.83  99 1443  1344 0.08    -0.
64 19.34
------------------------------------------------------------------------
------------------------------
group: 2
   vars  n   mean     sd median trimmed    mad min  max range skew kurtosi
s   se
X1    1 68 778.13 295.22    737  773.38 287.62 135 1493  1358 0.21    -0.3
5 35.8
> summary(df$balance)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   99.0   550.8   737.0   753.7   964.2  1493.0
```

2 iv

```
> library("car")
> qqPlot(df$balance)
[1] 151 132
```




From the above graph it appears that the distribution of the "balance" variable seems to follow faithfully the normal distribution with a very small deviation in the queues.

## Exercise 3

Consider the **mtcars** file data that follows a normal distribution.
Find the confidence interval with a confidence interval of 0.95 for the difference of the means corresponding to the fuel consumption variables for the manual and automatic car. (apply the t.test() function)

### Solution

This data set describes a number of car models and the am column explains whether the car is manual or automatic.

**Descriptive Statistics of data:**

```
> summary(mtcars)
      mpg              cyl              disp              hp              drat
wt              qsec              vs
```

```
 Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0   Min.   :2
.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000
 1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5   1st Qu.:3
.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000
 Median :19.20   Median :6.000   Median :196.3   Median :123.0   Median :3
.695   Median :3.325   Median :17.71   Median :0.0000
 Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7   Mean   :3
.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375
 3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0   3rd Qu.:3
.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000
 Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0   Max.   :4
.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000
       am              gear            carb
 Min.   :0.0000   Min.   :3.000   Min.   :1.000
 1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000
 Median :0.0000   Median :4.000   Median :2.000
 Mean   :0.4062   Mean   :3.688   Mean   :2.812
 3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000
 Max.   :1.0000   Max.   :5.000   Max.   :8.000
> str(mtcars)
'data.frame':  32 obs. of  11 variables:
 $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
 $ disp: num  160 160 108 258 360 ...
 $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
 $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
 $ qsec: num  16.5 17 18.6 19.4 17 ...
 $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
 $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
 $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
 $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

**Confidence interval with a confidence interval of 0.95 for the difference of the means corresponding to the fuel consumption variables for the manual and automatic car**

```
> t.test( mpg ~ am, data = mtcars,conf.level = 0.95)

        Welch Two Sample t-test

data:  mpg by am
t = -3.7671, df = 18.332, p-value = 0.001374
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -11.280194  -3.209684
sample estimates:
mean in group 0 mean in group 1
       17.14737        24.39231
```

From the above code it appears that the confidence interval is between -11.280194 and -3.209684.

## Exercise 4

Consider the **OctopusF.txt** dataset. Read the data, calculate the descriptive measures of the sample (mean, standard deviation). Plot the histogram. Check the normalization of the data and build the confidence interval.

### Solution

We read the **OctopusF.txt** data file with the command:

```
> df <- read.delim("OctopusF.txt")
```

**Descriptive Statistics of data:**

Finding the mean value.

```
> summary(df$weight)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   40.0   300.0   545.0   639.6   800.0  2400.0
```

According to the above results, the mean value is 639.6.

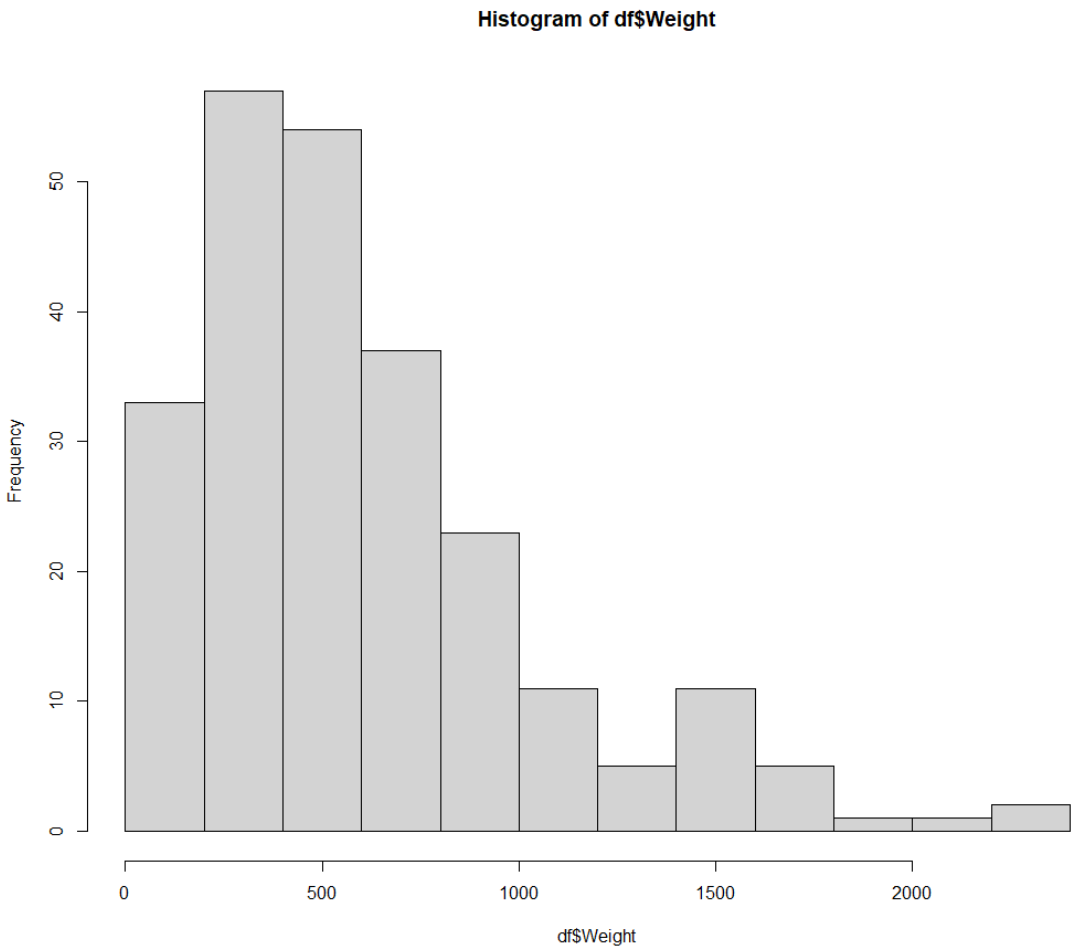
To find the standard deviation, we need to install the **pastecs** library. Then with the stat.desc() command I find the standard deviation.

```
> library(pastecs)
> stat.desc(df$weight)
     nbr.val     nbr.null       nbr.na          min          max        ra
nge          sum       median         mean      SE.mean
2.400000e+02 0.000000e+00 0.000000e+00 4.000000e+01 2.400000e+03 2.360000e
+03 1.535100e+05 5.450000e+02 6.396250e+02 2.878250e+01
CI.mean.0.95          var      std.dev     coef.var
5.669977e+01 1.988237e+05 4.458965e+02 6.971218e-01
```

According to the above results, the standard deviation is $4.458965e^{+02}$.

To build the histogram, I execute the code command:

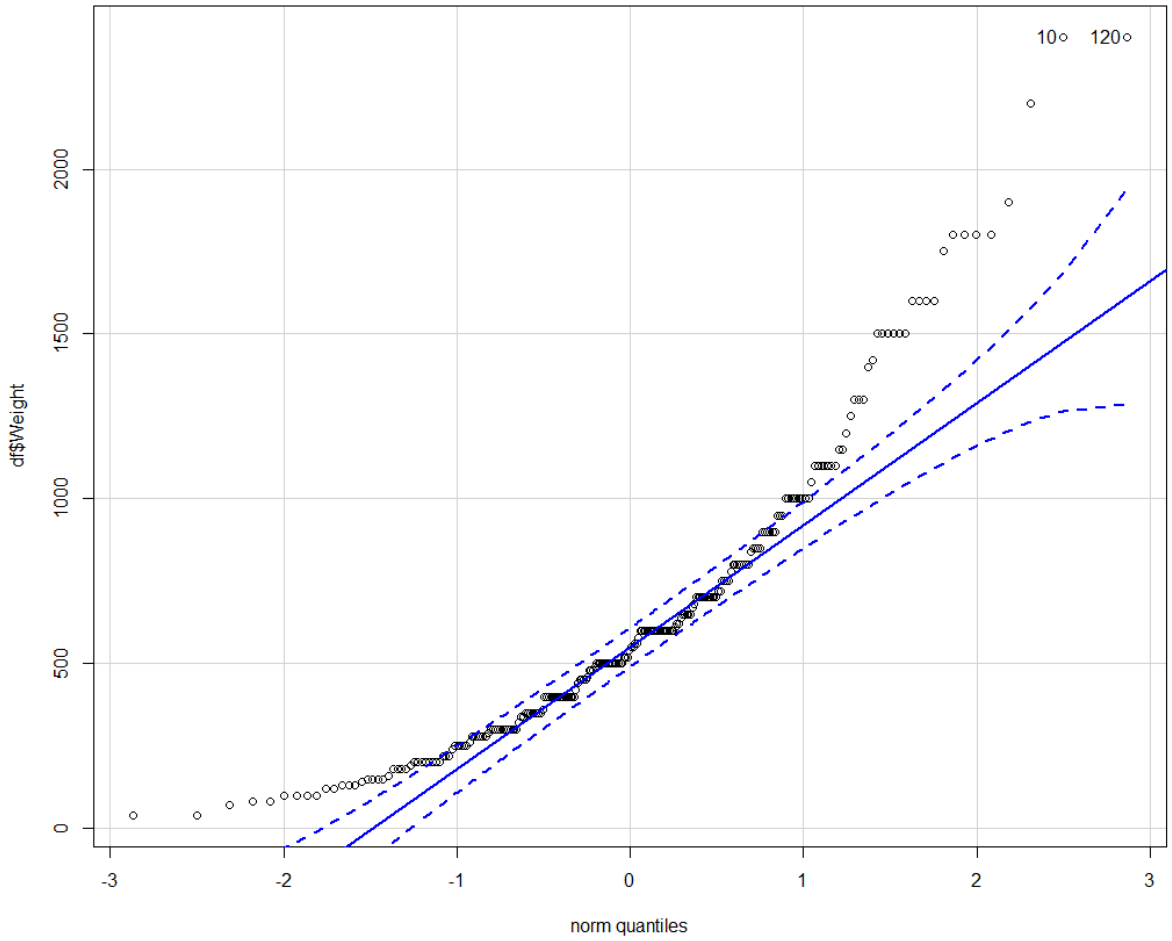
```
> hist(df$weight)
```




The distribution is oblique to the right and there are no outliers.

To check the normality of the data, we run the qqPlot command in the "car" library.

```
> library("car")
> qqPlot(df$weight)
[1]  10 120
```




From the above graph it appears that the distribution of the "balance" variable does not follow a normal distribution with a very large deviation in the queues.

**Calculation of Confidence Interval**

```
> # Calculate the mean and standard error
> l.model <- lm(df$weight ~ 1, df)
> # Calculate the confidence interval
> confint(l.model, level=0.95)
                2.5 %    97.5 %
(Intercept) 582.9252 696.3248
```

The 95% confidence interval is in the points [582.9252,696.3248]

## Exercise 5

load the MASS library. in the survey data file, the smoke column shows the degree of smoking of the students, while the Exer column indicates their level of physical activity. Smoking levels are "Heavy", "regul", "Occas" "Never". For the Exer variable these are "Freq", "Some", "None". Examine how much smoking affects physical activity
Hint: Contigency table and chi^2 test

### Solution

**Contingency Table**

```
> contigency_table = table(df$Smoke, df$Exer)
> contigency_table
```

```
        Freq None Some
  Heavy    7    1    3
  Never   87   18   84
  Occas   12    3    4
  Regul    9    1    7
> chisq.test(contigency_table)

        Pearson's Chi-squared test

data:  contigency_table
X-squared = 5.4885, df = 6, p-value = 0.4828
```

The degree of smoking does not affect physical exercise since the above p-value is greater than 0.05 (p-value>0.05).

## Exercise 6

Our data is stored as Concrete_Data.xls. and refer to variables that affect the durability of cement. The durability of cement is a non-linear function of age variables and various components such as blast furnace slag, fly ash, water, super-plasticizer, coarse aggregate. The first 8 are independent quantitative while Concretre compressive strength is the dependent. Use some packages to train the neural network e.g., neuralnet, nnet, RSNNS.
Read the data. Then you standardize your data. Then create the training and test sets. Train your model, plot the neuron and evaluate it. (use the compute() function and see if it works differently and why from the predict() function. See what the cor() function does. Improve your model if you become and see how your model behaves if the number of hidden nodes increases

## Solution

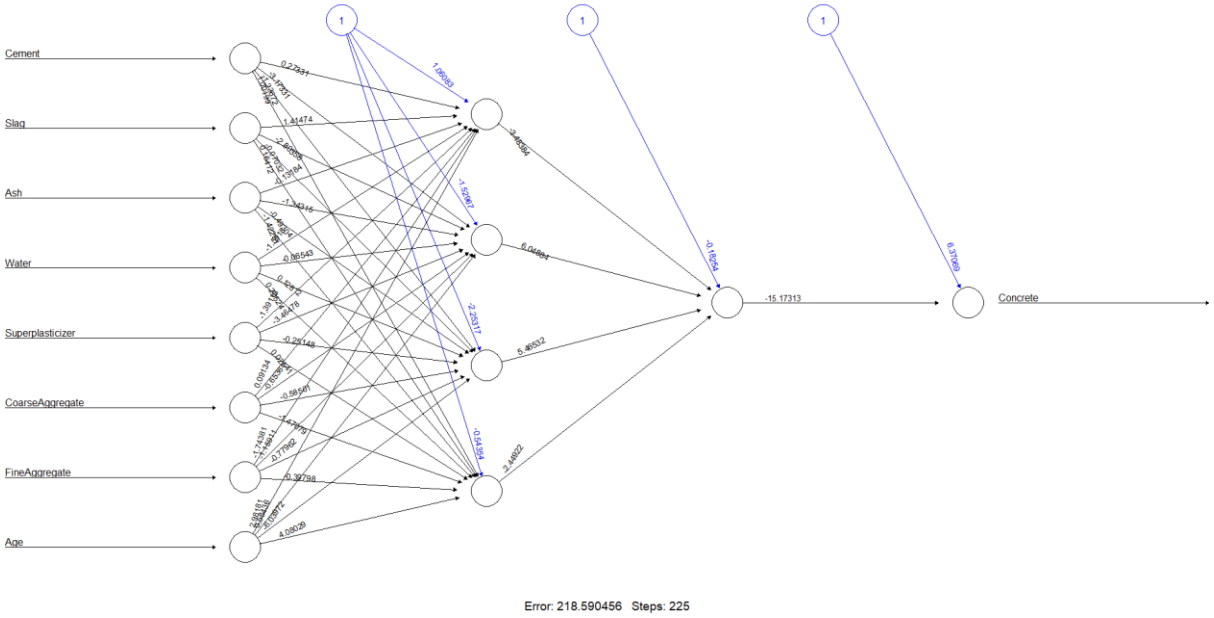
The Concrete_Data.xls data file refers to variables that affect the strength of the cement. With the str() command we read all our data.

```
> str(df)
tibble [1,030 x 9] (S3: tbl_df/tbl/data.frame)
 $ Cement          : num [1:1030] 540 540 332 332 199 ...
 $ Slag            : num [1:1030] 0 0 142 142 132 ...
 $ Ash             : num [1:1030] 0 0 0 0 0 0 0 0 0 0 ...
 $ Water           : num [1:1030] 162 162 228 228 192 228 228 228 228 228
...
 $ Superplasticizer: num [1:1030] 2.5 2.5 0 0 0 0 0 0 0 0 ...
 $ CoarseAggregate : num [1:1030] 1040 1055 932 932 978 ...
 $ FineAggregate   : num [1:1030] 676 676 594 594 826 ...
 $ Age             : num [1:1030] 28 28 270 365 360 90 365 28 28 28 ...
 $ Concrete        : num [1:1030] 80 61.9 40.3 41.1 44.3 ...
```

We notice that this file consists of 1030 rows and 9 columns.

We split the data file into **70% train set**(trainset <- df[1:721, ]) and **30% test set**(testset <- df[722:1030, ] ). Then we install the "neuralnet" library, to create a neural network, to train it and then to visualize it. The compute(),predict() and cor() functions were used.

```
library("readxl")
# xls files
df <- read_excel("Concrete_Data.xls")
str(df)
df <-scaleddata<-scale(df)
# Training and Test Data
set.seed(653)
df <- df[sample(nrow(df)), ]
trainset <- df[1:721, ]
testset <- df[722:1030, ]
#Neural Network
library(neuralnet)
nn <- neuralnet(Concrete ~ Cement + Slag + Ash + Water +
Superplasticizer + CoarseAggregate + FineAggregate + Age,
data=trainset, hidden=c(4,1), linear.output=FALSE,
threshold=0.5)
nn$result.matrix
plot(nn)
```




```
#Test the resulting output
temp_test <- subset(testset, select = c("Cement", "Slag",
"Ash", "Water", "Superplasticizer", "CoarseAggregate",
"FineAggregate", "Age"))
head(temp_test)
nn.results <- compute( nn, temp_test )
results <- data.frame( actual = testset$Concrete, prediction =
nn.results$net.result )
roundedresults<-sapply(results,round,digits=0)
roundedresultsdf=data.frame(roundedresults)
attach(roundedresultsdf)
table(actual,prediction)
```

Above we present the optimal model. If the number of hidden nodes increases more (hidden = c (4,1)) then the error of the control set increases.

## Exercise 7

Consider the **faithful** data set and estimate the next explosion if the expected time from the last explosion is 80 minutes. Find the R-Squared coefficient and the linear regression line. Check if there is a statistically significant relationship between the two variables in the linear regression model for the two variables of your data with significance level a = 0.05. Build a 95% Confidence Interval for the “eruption duration” variable for a given waiting time 80 min. Build a 95% Confidence prediction of the “eruption duration” variable given standby time 80 min. In addition to make the residual diagram of the linear regression against the

“waiting” variable. Make the diagram of the regularity for the standard residuals of the linear regression

## Solution

The faithful.txt data file contains 2 columns, which are the volcanic eruptions and the corresponding standby time. With the str() command we read all our data.

```
> str(df)
'data.frame':  272 obs. of  2 variables:
 $ eruptions: num  3.6 1.8 3.33 2.28 4.53 ...
 $ waiting  : num  79 54 74 62 85 55 88 85 51 85 ...
```

According to the above command, we observe that the specific file consists of 272 rows and 2 columns.

```
> reg.lm <- lm(eruptions ~ waiting, data = df)
> summary(reg.lm)

Call:
lm(formula = eruptions ~ waiting, data = df)

Residuals:
     Min       1Q   Median       3Q      Max
-1.29917 -0.37689  0.03508  0.34909  1.19329

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.874016   0.160143  -11.70   <2e-16 ***
waiting      0.075628   0.002219   34.09   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4965 on 270 degrees of freedom
Multiple R-squared:  0.8115,   Adjusted R-squared:  0.8108
F-statistic:  1162 on 1 and 270 DF,  p-value: < 2.2e-16
```

The regression R-Squared coefficient is 0.81 and the line of regression y:
eruptions = -1.87 + 0.0756*waiting

```
> x80.dat <- data.frame(waiting = 80)
> predict(reg.lm, newdata = x80.dat, interval = 'confidence')
      fit      lwr      upr
1 4.17622 4.104848 4.247592
```

For waiting=80 the next crisis is estimated at 4.17622

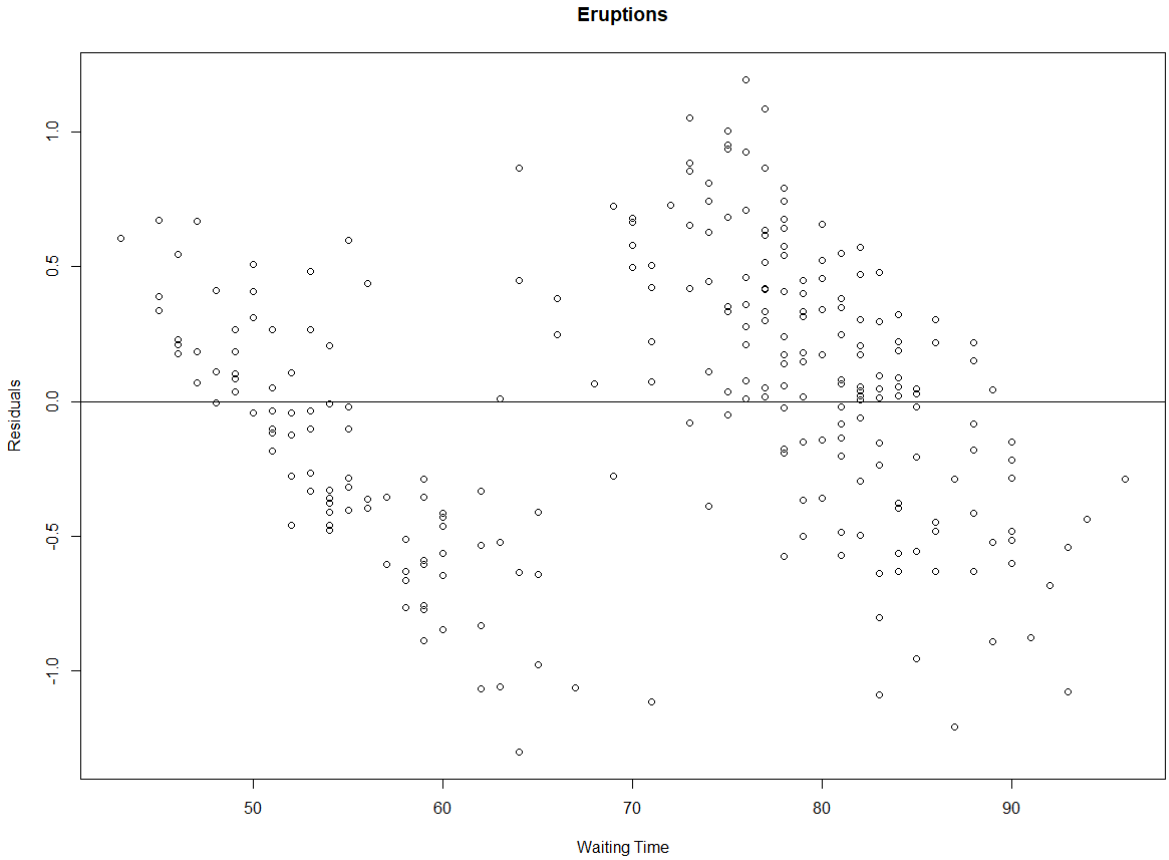
**The p-value of the t-test for the coefficient of the variable waiting is <2e$^{-16}$ *** <0.05 and therefore there is a statistically significant relationship between the two variables in the linear regression model for the two variables of your data with a significance level a=0.05**

```
> predict(reg.lm, newdata = x80.dat, interval = 'prediction')
      fit      lwr      upr
1 4.17622 3.196089 5.156351
```

The 95% Confidence Interval for the eruption duration variable for a given waiting time of 80 min is 4.104848 4.247592. A 95% Prediction Interval of the eruption duration variable given a waiting time of 80 min is 3.196089 5.156351.

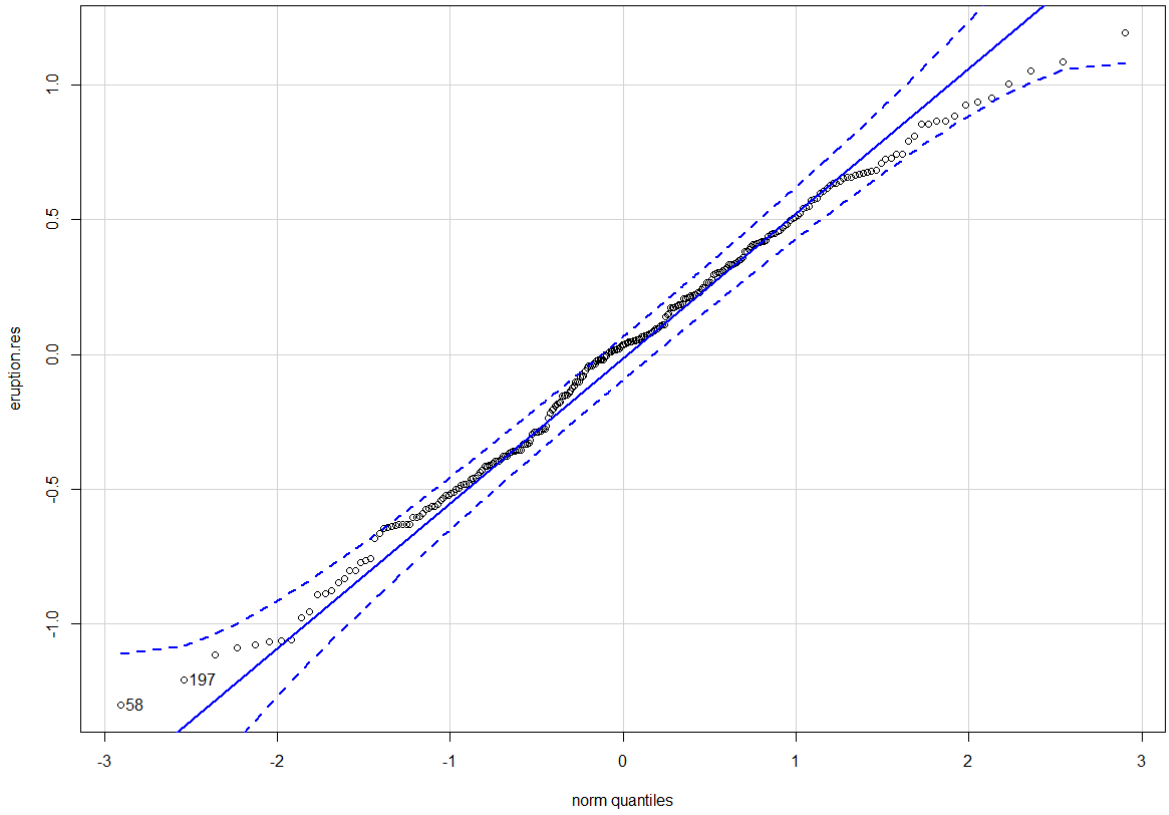
```
> eruption.res = resid(reg.lm)
> plot(df$waiting, eruption.res, ylab="Residuals", xlab="Waiting Time", ma
in="Eruptions")
> abline(0, 0)
```

**Residual diagram of linear regression versus waiting variable**




**Normality diagram for standard regression Residuals**

```
> library("car")
> qqPlot(eruption.res)
[1]  58 197
```




## Exercise 8

Consider the **stackloss** data and estimate the stack loss value if the airflow value =72 , the water temperature value =20 and the air concentration value =85. Find the R-Squared coefficient for the multiple linear model. Examine the significance of the model in our data with significance level a = 0.05. Do an 95% Confidence Interval for the mean of the dependent variable when we have values for the variables air flow =72, water temperature =20 and acid concentration =85, also to do 95% Prediction Interval for the dependent variable when we have values for the variable air flow =72, water temperature =20 and acid concentration =85

### Solution

We import the stackloss data file via the MASS library.

```
> library(MASS)
> data(stackloss)
> str(stackloss)
'data.frame':  21 obs. of  4 variables:
 $ Air.Flow  : num  80 80 75 62 62 62 62 62 58 58 ...
 $ Water.Temp: num  27 27 25 24 22 23 24 24 23 18 ...
 $ Acid.Conc.: num  89 88 90 87 87 87 93 93 87 80 ...
 $ stack.loss: num  42 37 37 28 18 18 19 20 15 14 ...
```

Through the str() command we notice that this data set contains 21 rows and 4 columns.

```
> stackloss.lm <- lm(stack.loss ~ Air.Flow + Water.Temp + Acid.Conc., data
= stackloss)
> summary(stackloss.lm)

Call:
lm(formula = stack.loss ~ Air.Flow + Water.Temp + Acid.Conc.,
    data = stackloss)

Residuals:
    Min      1Q  Median      3Q     Max
-7.2377 -1.7117 -0.4551  2.3614  5.6978

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -39.9197    11.8960  -3.356  0.00375 **
Air.Flow      0.7156     0.1349   5.307  5.8e-05 ***
Water.Temp    1.2953     0.3680   3.520  0.00263 **
Acid.Conc.   -0.1521     0.1563  -0.973  0.34405
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.243 on 17 degrees of freedom
Multiple R-squared:  0.9136,   Adjusted R-squared:  0.8983
F-statistic:  59.9 on 3 and 17 DF,  p-value: 3.016e-09
```

The R-Squared for the multiple linear model is:
**R-squared: 0.9136.**

The significance of the model in our data with significance level a=0.05 is strong since the p-value of the total F-test is **p-value: $3.016e^{-09}$.**

```
> new_data = data.frame(Air.Flow=72, Water.Temp=20, Acid.Conc.=85)
> predict(stackloss.lm, new_data)
       1
24.58173
```

The value of stack loss if the value of airflow=72 and the value of water temperature=20 and the value of air concentration=85 is **24.58173**

.

```
> predict(stackloss.lm, newdata = new_data, interval = 'confidence')
       fit      lwr    upr
1 24.58173 20.21846 28.945
```

95% of Confidence Interval for the mean of the dependent variable when we have values for the variables air flow=72, water temperature=20 and acid concentration=85:
**20.218 28.945**

```
> predict(stackloss.lm, newdata = new_data, interval = 'prediction')
       fit     lwr      upr
1 24.58173 16.4661 32.69736
```

95% of Confidence Interval forecast for the dependent variable when we have values for the variables air flow=72, water temperature=20 and acid concentration=85:
**16.466 32.69736**

## Exercise 9

The sales representative of a product is not satisfied with the sales of the product in his area of responsibility. He finds that the products sold differ from store to store in a price range of 921 to 2604 pieces with an mean price of 1846.8. He wants to know what the reason for this difference is. For that, he randomly selects 37 stores of the same size, observing the sales of the product, the price of the product, the sales costs and the number of arrivals in these stores for a certain time period. The following research should answer the question of whether the sales of product sets are affected by the other sizes chosen.

Examine if

i. Your model is statistically significant
ii. If the sizes are statistically significant
iii. If there is interdependence between the sizes
iv. Calculate the standardized residuals and find the lowest and highest standardized residual
v. Examine the distribution of the residuals
vi. Write the prediction for sales. Where are the best predictions made?
vii. Write the theory of homoscedasticity and test if there are such in your model

### Solution

We read the xls data file through the readxl library.

```
> library("readxl")
> # xls files
> df <- read_xlsx("market.xlsx")
> market.lm <- lm(Sales ~ Preis + Costs + Arrivals, data = df)
> summary(market.lm)

Call:
lm(formula = Sales ~ Preis + Costs + Arrivals, data = df)

Residuals:
```

```
    Min      1Q  Median      3Q     Max
-261.04 -127.20  -38.65  144.85  229.20

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 730.01652  225.03056   3.244   0.0027 **
Preis       -42.09271   16.17993  -2.602   0.0138 *
Costs         0.55206    0.05069  10.891 1.83e-12 ***
Arrivals      9.64889    1.66599   5.792 1.78e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 156.1 on 33 degrees of freedom
Multiple R-squared:  0.8447,   Adjusted R-squared:  0.8306
F-statistic: 59.83 on 3 and 33 DF,  p-value: 1.953e-13
```

I)

The model is statistically significant since the p-value of the total F-test is <0.05.

```
F-statistic: 59.83 on 3 and 33 DF,  p-value: 1.953e-13
```

II)

The p-value of the t-test for the coefficient of the «Preis», «Cost», «Arrivals» variable is 0.0138 * < 0.05, $1.83^{e-12}$ *** < 0.05 and $1.78^{e-06}$ *** < 0.05 respectively, and therefore there is a statistically significant relationship between the two variables in the linear regression model for the two variables of your data with significance level a=0.05

III)

From the above correlation coefficients we conclude that there is no interdependence between the quantities since the coefficients are very small.

```
> # x-correlation
> cor(df[,2:4])
              Preis      Costs   Arrivals
Preis    1.00000000 0.01400399 0.04289388
Costs    0.01400399 1.00000000 0.14847016
Arrivals 0.04289388 0.14847016 1.00000000
```

```
> market_inter.lm <- lm(Sales ~ Preis + Costs + Arrivals + Preis*Costs + P
reis*Arrivals + Costs*Arrivals, data = df)
> summary(market_inter.lm)

Call:
lm(formula = Sales ~ Preis + Costs + Arrivals + Preis * Costs +
    Preis * Arrivals + Costs * Arrivals, data = df)
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-285.36 -118.59  -26.14  130.36  220.76

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    -1.992e+02  1.301e+03  -0.153    0.879
Preis           7.139e+01  1.081e+02   0.660    0.514
Costs           2.353e-01  4.712e-01   0.499    0.621
Arrivals        2.134e+01  1.341e+01   1.592    0.122
Preis:Costs     1.059e-02  3.137e-02   0.338    0.738
Preis:Arrivals -1.413e+00  1.144e+00  -1.235    0.226
Costs:Arrivals  2.462e-03  3.637e-03   0.677    0.504

Residual standard error: 158.4 on 30 degrees of freedom
Multiple R-squared:  0.8546,   Adjusted R-squared:  0.8256
F-statistic:  29.4 on 6 and 30 DF,  p-value: 2.771e-11
```

There is no interaction between the values since the p-values of the following t-tests are >0.05.

```
Preis:Costs     1.059e-02  3.137e-02   0.338    0.738
Preis:Arrivals -1.413e+00  1.144e+00  -1.235    0.226
Costs:Arrivals  2.462e-03  3.637e-03   0.677    0.504
```

IV)

Find the standardized residuals with the rstandard() command.).

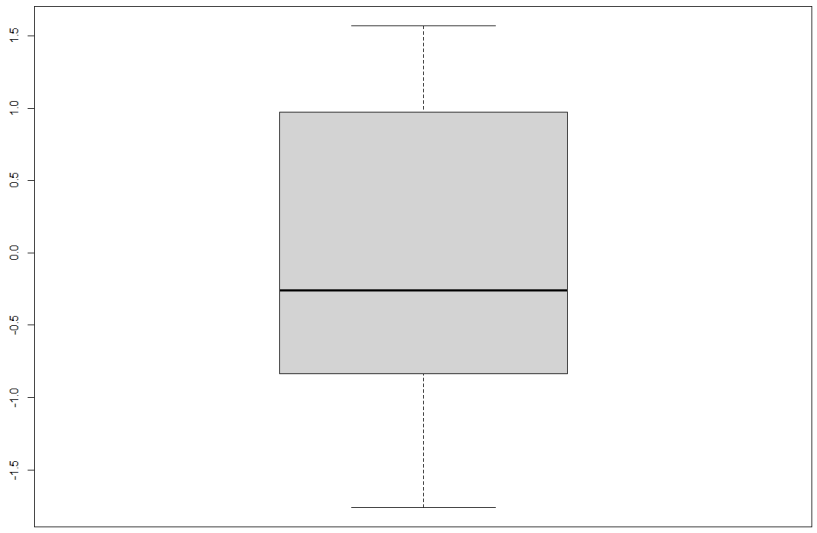
```
> # Standardized Residuals
> market.stdres = rstandard(market.lm)
> summary(market.stdres)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
-1.759975 -0.836211 -0.261024 -0.001757  0.976235  1.570400
```

According to the results of the above code, the smallest and largest standardized residual amounts to -1.759975 and 1.5704 respectively.

V)

To examine the distribution of the residuals, we apply the following code command to visualize our data.

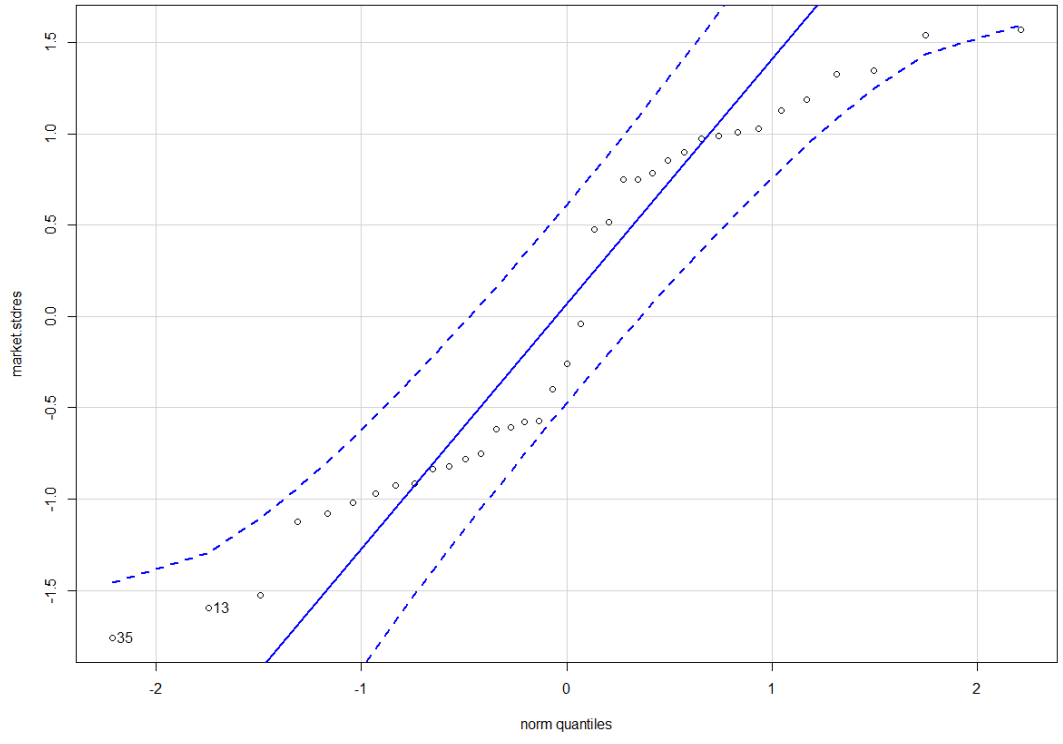
```
> boxplot(market.stdres, data=df)
```

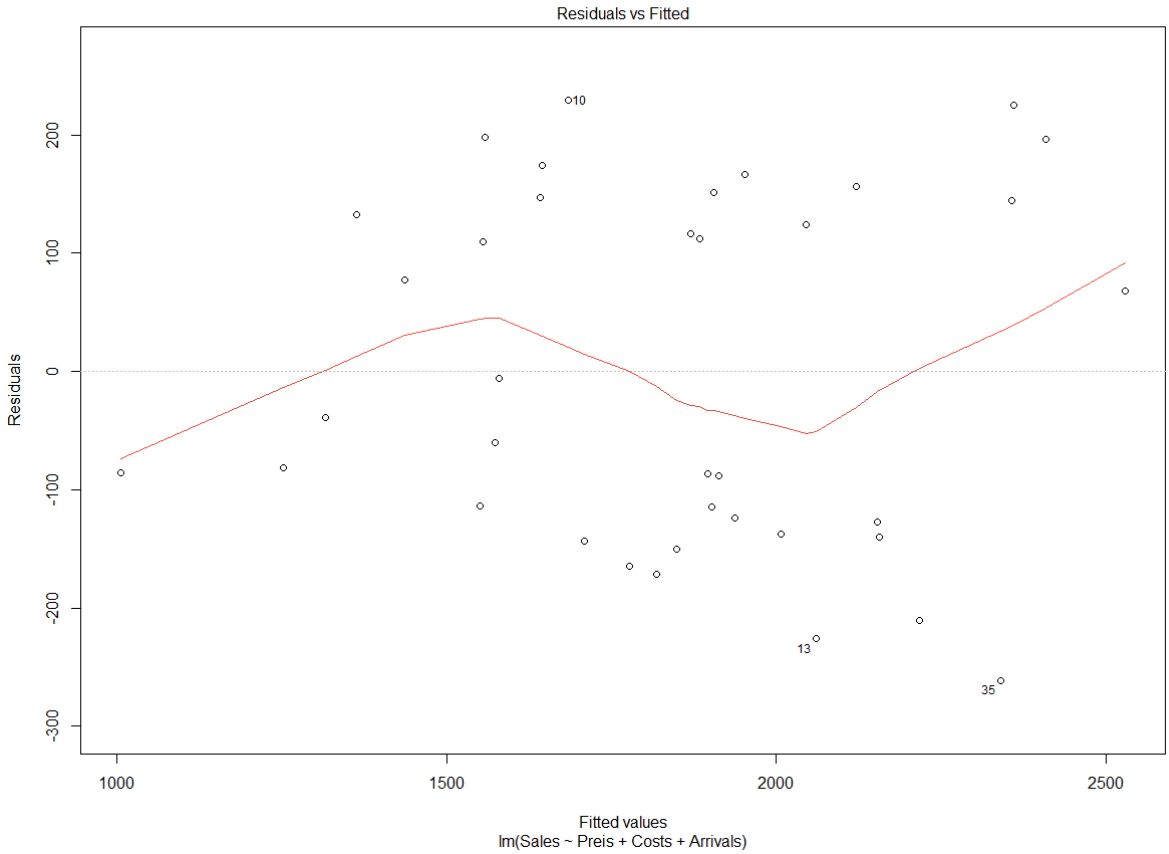


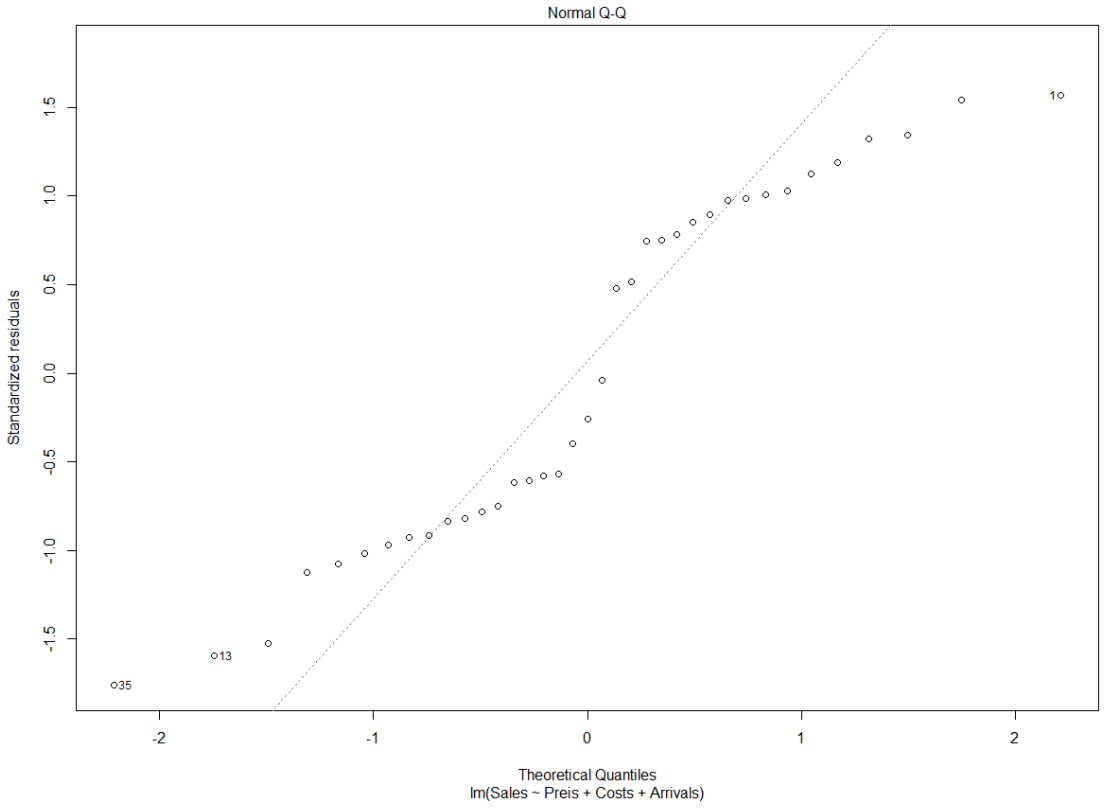

According to the above pictogram, the distribution is oblique to the right and there are no outliers.

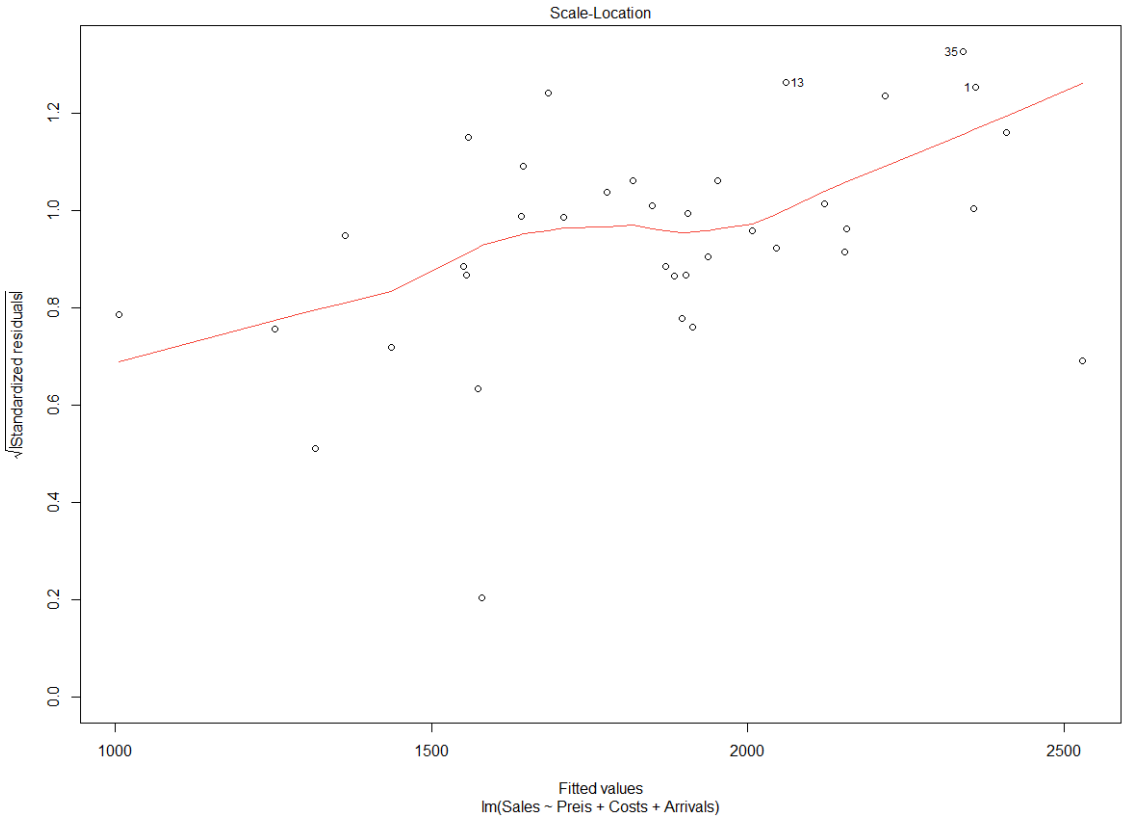
```
> qqPlot(market.stdres)
[1] 35 13
 > plot(market.lm)

 Hit <Return> to see next plot:
```

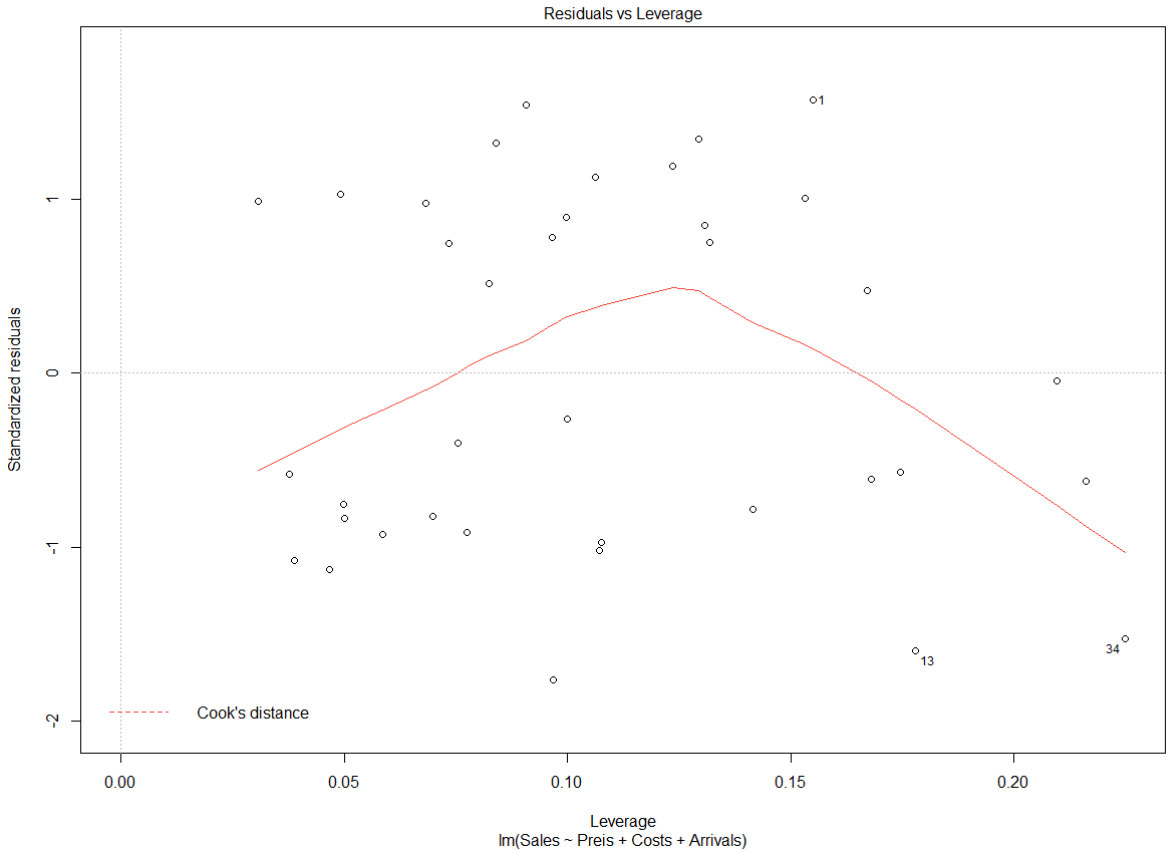
Pressing the Enter button shows the following relevant results.




Residuals vs Fitted
Residuals
Fitted values
lm(Sales ~ Preis + Costs + Arrivals)
10
13
35
-300
-200
-100
0
100
200
1000
1500
2000
2500


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Sales ~ Preis + Costs + Arrivals)
1
13
35
-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5
-2
-1
0
1
2


Scale-Location
√|Standardized residuals|
0.0
0.2
0.4
0.6
0.8
1.0
1.2
35
13
1
1000
1500
2000
2500
Fitted values
lm(Sales ~ Preis + Costs + Arrivals)




VI)

The forecast for sales is as follows:
Sales=730-42.09*Preis + 0.55*Costs + 9.65*Arrivals

The best predictions are made in the observations with the smallest residuals.
The most important variable is "costs", then "Arrivals" and finally "Preis" based on t-test values.

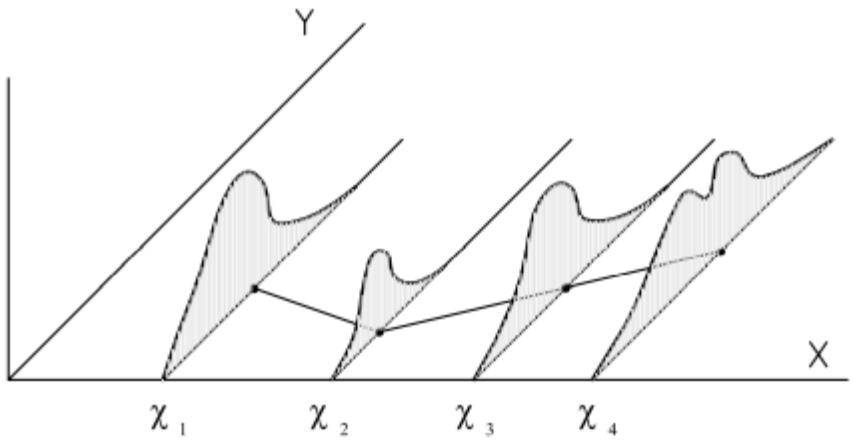
VII)

```
> lmtest::bptest(market.lm)  # Breusch-Pagan test

        studentized Breusch-Pagan test

data:  market.lm
BP = 10.574, df = 3, p-value = 0.01427
```

Based on the Breusch-Pagan test p-value which is <0.05, there is heteroscedasticity.

**Conditions-assumptions for the application of the Simple Linear Model**
**$Y = \alpha + \beta \cdot X + \varepsilon$**

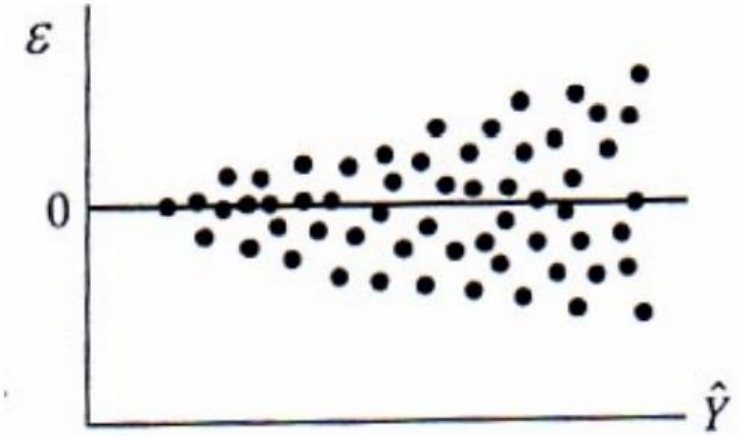
The general assumption we make for a regression model (linear or not) is that the variable X is measured without error and that Y, for every level i x of X, is a random variable with finite mean and scatter (Πανάρετος, 2003).




For the simple linear model we also make the following assumptions:

1. Linearity.
2. Homoscedasticity - Variance Stability.
3. Independence.
4. Normality.

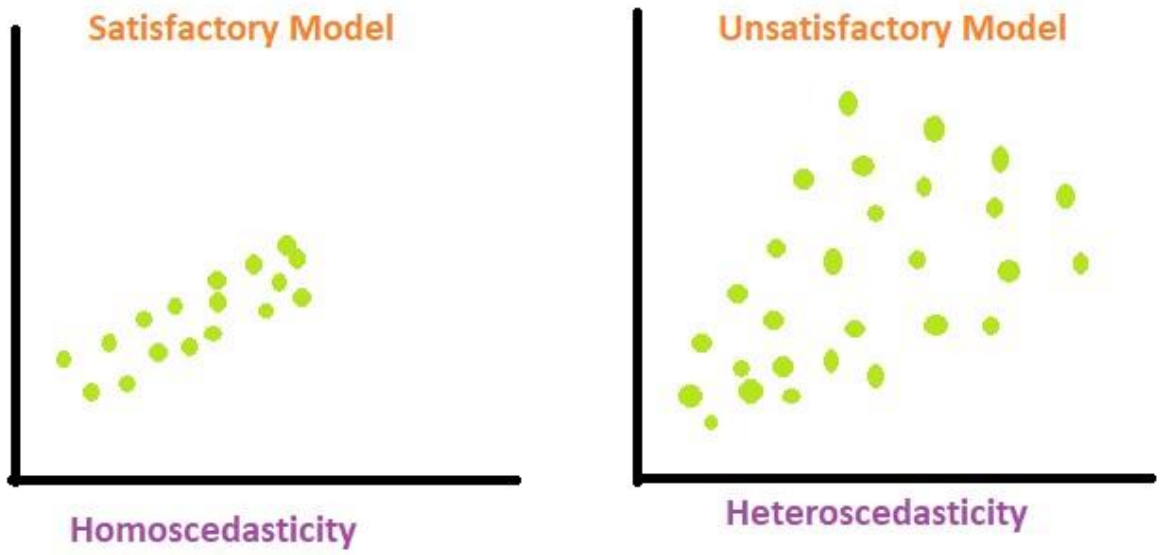
A first check of the stability or non-dispersion of Y (or e) for the various levels of X can be made with the dispersion diagram and the rest diagrams. If, for example, the residual diagram is in the form of a trapezoid (open fan), as follows, the most probable cause of this disturbance7 is the instability of the scattering of random errors e. X or Y gives diagrams of trapezoidal remainders (increasing by X or Y, increases by 2 σ or vice versa). This is because such applications follow multiplication models where 2 2 2 σ Y = [E (Y)] ·σ and 2 σ the scatter of errors ε (why?) 8. Also, corresponding balance diagrams give variables that measure the number of events in the unit of time, space, length, etc., ie variables that follow a Poisson distribution.




If the rest diagrams suggest that we do not have constant dispersions, we can statistically check whether there is a significant difference in the dispersions or not if we have more than one observation for the different levels of X. We can also classify the observations in ascending order of X, divide them into two or more groups, and statistically check whether

the groups have a significant difference in dispersions or not. When dispersion instability is detected we can, in several cases, address the problem with appropriate transformations in the variables.

One of the basic assumptions of the classical linear model is that the perturbation term $\boldsymbol{\varepsilon t}$ (of the population) is a random variable having mean $\mathbf{E}$ ($\boldsymbol{\varepsilon t}$) = $\mathbf{0}$ and constant variance for all values of the dependent variable $\boldsymbol{\Upsilon t}$. When the variance of the disturbing term is constant, then the pattern is characterized by homoskedasticity:
$\mathbf{Var}$ ($\boldsymbol{\varepsilon t}$) = $\boldsymbol{\sigma\varepsilon}$ $\mathbf{2}$ = constant, for $\mathbf{t}$ = $\mathbf{1}$,…, $\mathbf{T}$
Otherwise, when the variance of the disturbing term is not constant, then there is heteroskedasticity in the model:
$\mathbf{Var}$ ($\boldsymbol{\varepsilon t}$) = $\boldsymbol{\sigma t}$ $\mathbf{2}$, for $\mathbf{t}$ = $\mathbf{1}$,…, $\mathbf{T}$
where all the fluctuations $\boldsymbol{\sigma t}$. are not equal to each other, so we say that there is heterosexuality in the disturbing terms. Heteroscedasticity refers to the case where in different observations of the independent variable $\mathbf{Xt}$, the variance of the disturbing term $\boldsymbol{\varepsilon t}$ is not constant.




## Consequences of heteroscedasticity

Suppose the model $\boldsymbol{\Upsilon t}$ = $\boldsymbol{\beta 0}$ + $\boldsymbol{\beta 1 X t}$ + $\mathbf{u}\boldsymbol{t}$ and assume that all the basic assumptions of the classical linear model are valid, except homoscedasticity (Stock & Watson, 2017), (Χρήστου, 2007). That is, we assume that: $\mathbf{Var}$ ($\boldsymbol{\varepsilon t}$) = $\mathbf{2}$ $\mathbf{2}$, for $\mathbf{t}$ = $\mathbf{1}$,…, $\mathbf{T}$.
The estimators of the coefficients of the above model, which result from the least squares method, when the disturbing term is heterosexual, are still linear and unbiased. The problem that arises is mainly related to the estimates of their fluctuations and their effectiveness.
That is, least squares estimators do not have the slightest variation (usually fluctuations are underestimated). Due to the underestimation of the fluctuations, higher values of the statistics $\boldsymbol{t}$ and F are estimated.

- Confidence Intervals are unreliable and model predictions are ineffective.
- Our conclusions about the parameters of the population will be unreliable.

- Appraisers, however, remain impartial and consistent. This is because none of the interpretive variables is associated with the error term. Thus, the values of the estimated coefficients will be very close to the actual parameters.

### Ways to determine heteroscedasticity

There is a large number of tests to determine the problem of heteroscedasticity, depending on the available statistics. If for each value of the interpretive variable there are several observations for the dependent variable, or if the sample size is large enough to be divided into groups, the heterosexuality test can be done with the standard homogeneity test. Many tests are based on the analysis of residues resulting from the least squares method. Some basic heteroscedasticity tests are as follows:

a. Barlett criterion.
b. Spearman correlation coefficient.
c. Goldfeld-Quandt control.
d. Breusch-Pagan-Godfrey control.
e. White control.

## Exercise 10

Consider the insurance.csv data. Our purpose is to study how these variables affect expenses

Do:

i. Reading the data
ii. Control of variables (use the "psych" package)
iii. Create the appropriate model to answer the above question

### Solution

#### I)

We read the insurance.csv file with the command

```
> df = read.csv("insurance.csv", sep = ',')
```

Then with the str() command we observe the data file data.

```
> str(df)
'data.frame':  1338 obs. of  7 variables:
 $ age     : int  19 18 28 33 32 31 46 37 37 60 ...
 $ sex     : chr  "female" "male" "male" "male" ...
 $ bmi     : num  27.9 33.8 33 22.7 28.9 ...
 $ children: int  0 1 3 0 0 0 1 3 2 0 ...
 $ smoker  : chr  "yes" "no" "no" "no" ...
 $ region  : chr  "southwest" "southeast" "southeast" "northwest" ...
 $ charges : num  16885 1726 4449 21984 3867 ...
```

This data set consists of 1338 rows and 7 column.

```
> df$sex = as.factor(df$sex)
> df$sex
Levels: female male
```

With the above command it shows me in total how many different types of sexes there are in this column.

```
> df$smoker = as.factor(df$smoker)
> df$smoker
Levels: no yes
```

With the above command it shows me if there are smokers in the specific column.

```
> df$region = as.factor(df$region)
> df$region
Levels: northeast northwest southeast southwest
```

```
>library(psych)
>library(psychTools)  #additional tools and data are here
> describe(df)  #basic descriptive statistics
          vars    n     mean       sd  median  trimmed     mad     min
max    range  skew kurtosis     se
age          1 1338    39.21    14.05   39.00    39.01   17.79   18.00    6
4.00    46.00  0.06    -1.25   0.38
sex          2 1338     1.51     0.50    2.00     1.51    0.00    1.00
2.00     1.00 -0.02    -2.00   0.01
bmi          3 1338    30.66     6.10   30.40    30.50    6.20   15.96    5
3.13    37.17  0.28    -0.06   0.17
children     4 1338     1.09     1.21    1.00     0.94    1.48    0.00
5.00     5.00  0.94     0.19   0.03
smoker       5 1338     1.20     0.40    1.00     1.13    0.00    1.00
2.00     1.00  1.46     0.14   0.01
region       6 1338     2.52     1.10    3.00     2.52    1.48    1.00
4.00     3.00 -0.04    -1.33   0.03
charges      7 1338 13270.42 12110.01 9382.03 11076.02 7440.81 1121.87 6377
0.43 62648.55  1.51     1.59 331.07
```

II)

```
> corr.test(df)
Call:corr.test(x = df)
Correlation matrix
            age   sex  bmi children smoker region charges
age        1.00 -0.02 0.11     0.04  -0.03   0.00    0.30
sex       -0.02  1.00 0.05     0.02   0.08   0.00    0.06
bmi        0.11  0.05 1.00     0.01   0.00   0.16    0.20
children   0.04  0.02 0.01     1.00   0.01   0.02    0.07
smoker    -0.03  0.08 0.00     0.01   1.00   0.00    0.79
region     0.00  0.00 0.16     0.02   0.00   1.00   -0.01
charges    0.30  0.06 0.20     0.07   0.79  -0.01    1.00
Sample Size
[1] 1338
Probability values (Entries above the diagonal are adjusted for multiple t
ests.)
            age  sex  bmi children smoker region charges
age        0.00 1.00 0.00     1.00   1.00   1.00    0.00
```

```
sex      0.45 0.00 1.00     1.00   0.08   1.00    0.51
bmi      0.00 0.09 0.00     1.00   1.00   0.00    0.00
children 0.12 0.53 0.64     0.00   1.00   1.00    0.19
smoker   0.36 0.01 0.89     0.78   0.00   1.00    0.00
region   0.94 0.87 0.00     0.54   0.94   0.00    1.00
charges  0.00 0.04 0.00     0.01   0.00   0.82    0.00
```

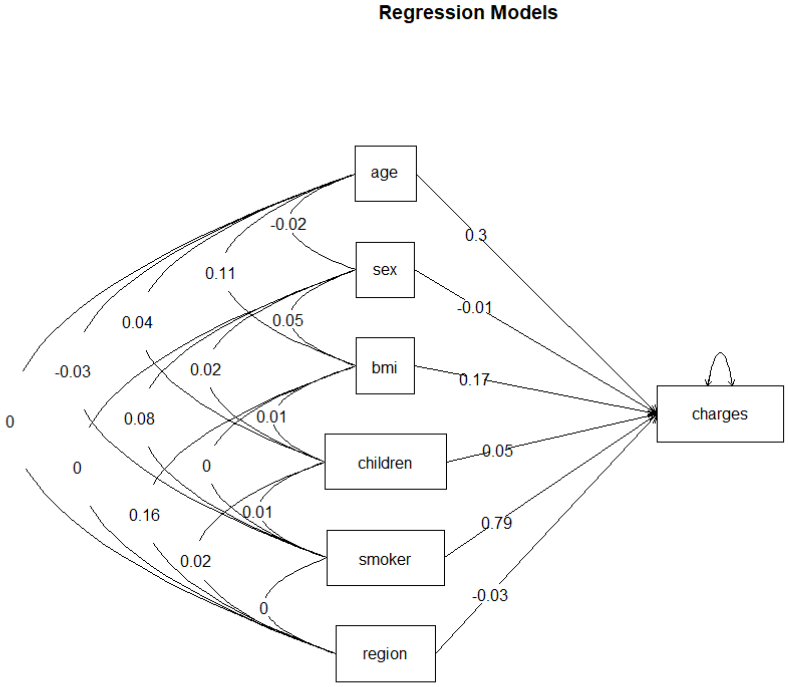
From the p-values of the tests for the correlation coefficients, we see that all the variables are significantly correlated with the expenses except the "region". In the regression model, however, all variables are statistically significant, which is shown in question III.

III)

```
> setCor(y = 7,x=1:6, data = df)
Call: setCor(y = 7, x = 1:6, data = df)

Multiple Regression from raw data

 DV =  charges
            slope   se     t       p lower.ci upper.ci  VIF
(Intercept)  0.00 0.01  0.00 1.0e+00    -0.03     0.03 1.00
age          0.30 0.01 21.65 2.8e-89     0.27     0.33 1.02
sex         -0.01 0.01 -0.39 6.9e-01    -0.03     0.02 1.01
bmi          0.17 0.01 12.00 1.5e-31     0.14     0.19 1.04
children     0.05 0.01  3.48 5.1e-04     0.02     0.07 1.00
smoker       0.79 0.01 57.84 0.0e+00     0.77     0.82 1.01
region      -0.03 0.01 -2.33 2.0e-02    -0.06    -0.01 1.03

Residual Standard Error =  0.5  with  1331  degrees of freedom

 Multiple Regression
           R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1  df2 p
charges 0.87 0.75 0.57 0.32        0.75     0.01    668.12   6 1331 0
           R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1  df2 p
charges 0.87 0.75 0.57 0.32        0.75     0.01    668.12   6 1331 0
```

The important variables that affect costs are reported in the following model:
Charges = 0.3*age – 0.01*sex + 0.17*bmi +0.05 *children + 0.79*smoker – 0.03*region

, as shown in the diagram below.




## Exercise 11

MF is an electronics company. If the company has a good quality assurance system of its products there are cases of return. During its operation it created a database. In the first column there are the compensations for each return, in the second the shifts construction of the goods, in the third the types of complaints and in the fourth the part of production. The company's quality manager studied a random sample of 110 returns (mf.xls)

i. Present the data with contingency tables looking at the sizes in pairs (compensation for the rest as well as type of complaints against a shift or place of production). Comment on the data.

ii. The person in charge wants to give an answer to his suspicion that there is a relationship between the types of complaints and the place of production. Calculate the expected values in each cell. Indicate a way to have an expected value =5 in each cell. Is the suspect's suspicion valid at a significance level of 0.01?

iii. The sales representative wants to know if there is a difference in the compensation between production sites (Boise and Salt Lake City). Check it at significance level a = 0.02

### Solution

```
> library("readxl")
> # xls files
> df <- read_excel("mf.xls")
```

```
> # I)
> df$amount_cut = cut(df$`Dollar Claim Amount`,3)
```

I)

With the table() command calculates the Contigency tables of the following variables of each category.

```
> tb1 = table(df$amount_cut, df$Shift)
> tb1

             1  2  3
  (146,228] 15  8  0
  (228,311] 41 16  3
  (311,393] 19  6  2
> chi2_1 = chisq.test(tb1)
> chi2_1$p.value
[1] 0.6727148
```

```
> tb2 = table(df$amount_cut, df$`Complaint Code`)
> tb2

             1  2  3  4
  (146,228]  7  8  5  3
  (228,311] 20 23 13  4
  (311,393]  8 14  5  0
> chi2_2 = chisq.test(tb2)
> chi2_2$p.value
[1] 0.5915626
```

```
> tb3 = table(df$amount_cut, df$`Manufacturing Plant`)
> tb3

             1  2  3
  (146,228] 18  5  0
  (228,311] 43 11  6
  (311,393] 17  8  2
> chi2_3 = chisq.test(tb3)
> chi2_3$p.value
[1] 0.4342091
```

```
> tb4 = table(df$`Complaint Code`, df$Shift)
> tb4

    1  2  3
  1 23 11  1
  2 32 12  1
  3 16  4  3
  4  4  3  0
> chi2_4 = chisq.test(tb4)
> chi2_4$p.value
[1] 0.3574079
```

```
> tb5 = table(df$`Complaint Code`, df$`Manufacturing Plant`)
```

```
> tb5

    1  2  3
  1 30  3  2
  2 31 11  3
  3 13  8  2
  4  4  2  1
> chi2_5 = chisq.test(tb5)
> chi2_5$p.value
[1] 0.2755304
```

II)

Based on the following π-value of the chi2-test (π-value = 0.2755> 0.05) there is no relationship between the types of complaints and the place of production.

```
> tab1 = table(df$`Complaint Code`, df$`Manufacturing Plant`)
> chi2 = chisq.test(tab1)
> chi2$expected

           1        2         3
  1 24.818182 7.636364 2.5454545
  2 31.909091 9.818182 3.2727273
  3 16.309091 5.018182 1.6727273
  4  4.963636 1.527273 0.5090909
> chi2$p.value
[1] 0.2755304
```

**Expected values:**

In the Complaint Code variable, I apply merge in groups 3 & 4 and in the Manufacturing Plant variable I apply merge in groups 2 & 3

To have expected values> 5, we join rows of columns with low frequencies to achieve this goal.

III)

```
> df1 = df[,c(1,4)]
> df1 = df1[ df1$`Manufacturing Plant` < 3,  ]
> t.test(df1$`Dollar Claim Amount` ~ df1$`Manufacturing Plant`, data = df1
)

        Welch Two Sample t-test

data:  df1$`Dollar Claim Amount` by df1$`Manufacturing Plant`
t = -0.63437, df = 32.887, p-value = 0.5302
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -37.43634  19.64147
```

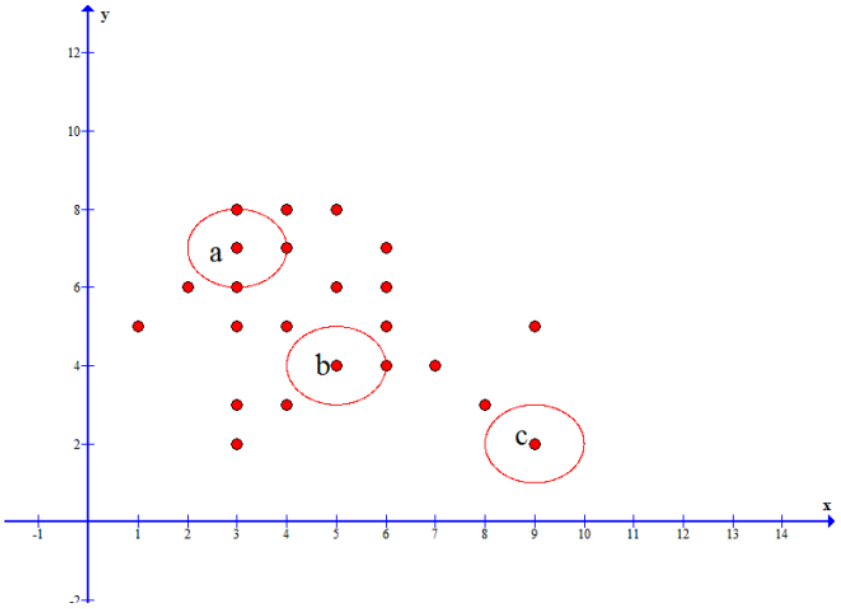
```
sample estimates:
mean in group 1 mean in group 2
       268.4359        277.3333
```

We apply Two Sample t-test and based on the p-value=0.53 >0.02 we find that there is no difference in the amount of compensation between the production sites (Boise and Salt Lake City) with a level of significance a=0.02

## Exercise 12

Apply the DBSCAN algorithm with ε = 1 and MinPts = 3. To which category do points a, b, c belong?

Apply K-means and find the best possible number of k clusters




### Solution

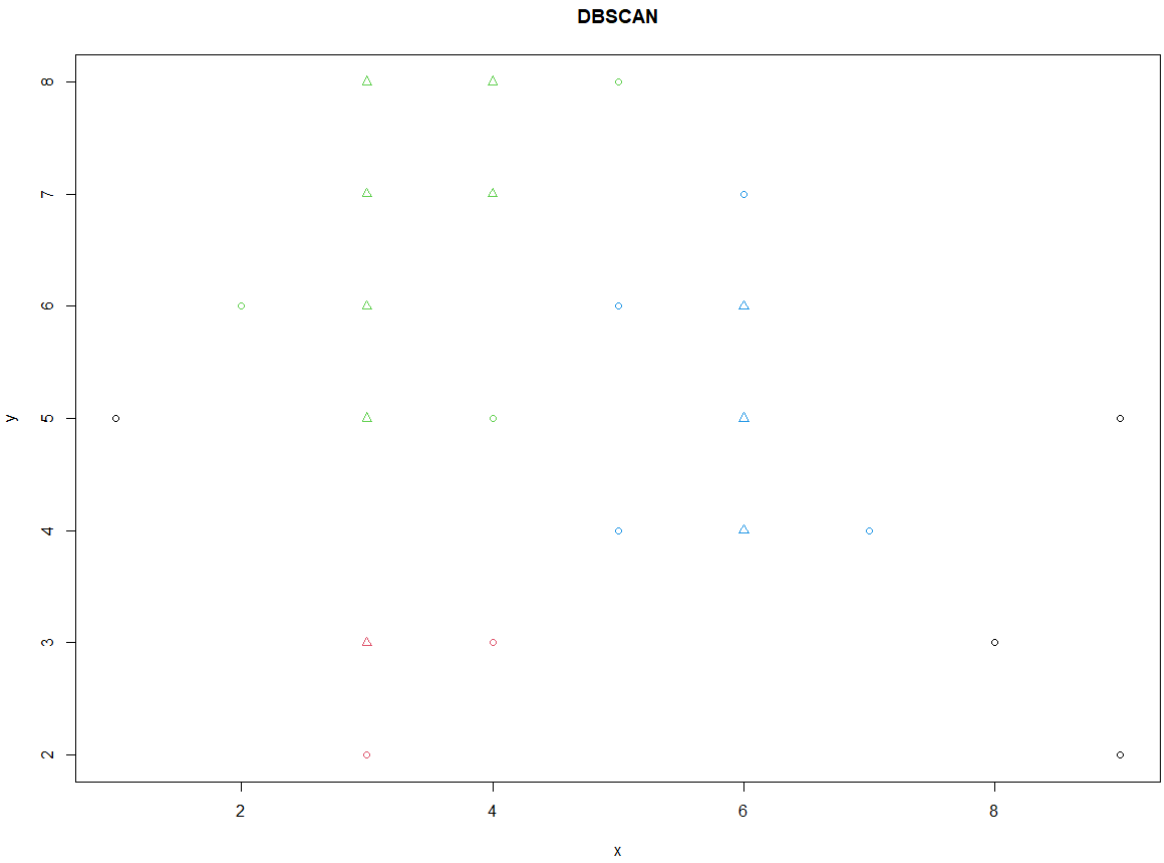
#question a

Point **a** is a central point, because in its neighborhood there are 3 points. Point **b** has in its neighborhood 1 point which is accessible from the point with coordinates (6,4). Finally, point **c** is extreme since there is no other point in its neighborhood.

#question b

```
> library("fpc")
> library("dbscan")
> x <- c(1, 2, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 6, 6, 6, 6, 7, 8, 9,
9)
> y <- c(5, 6, 2, 3, 5, 6, 7, 8, 3, 5, 7, 8, 4, 6, 8, 4, 5, 6, 7, 4, 3, 2,
5)
```

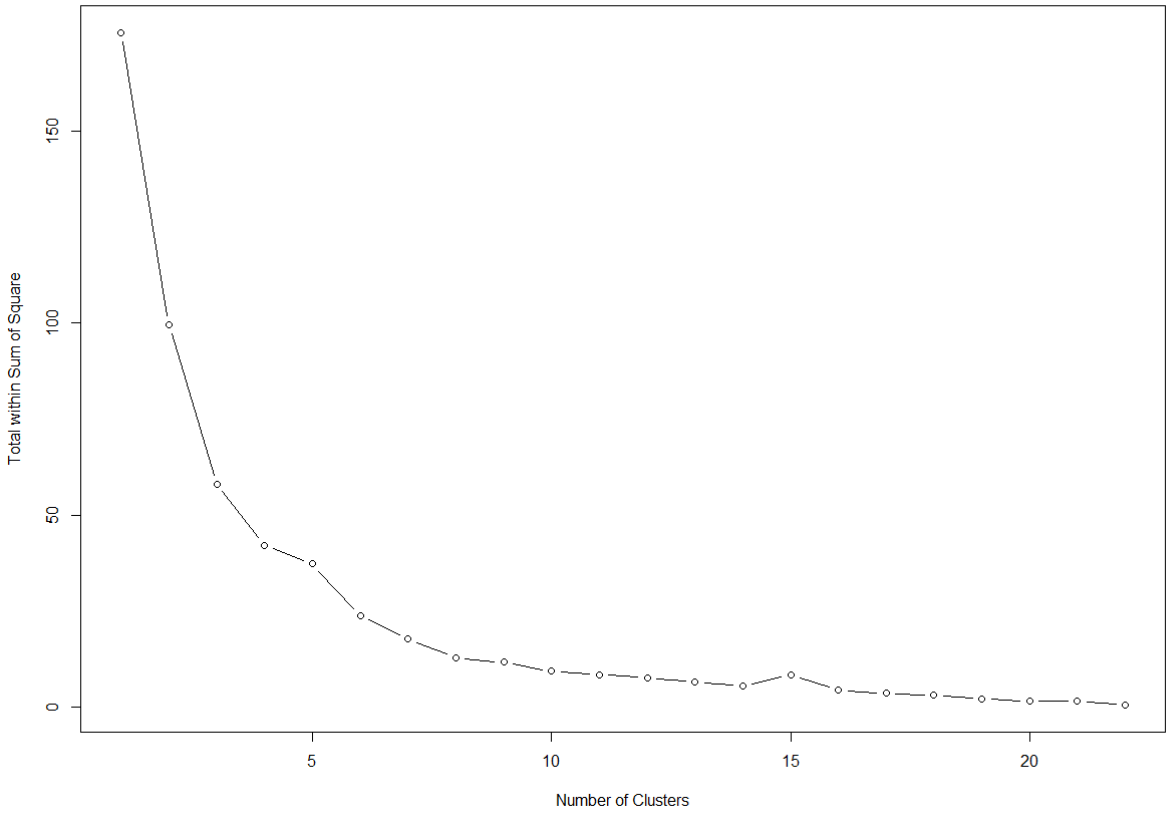
```
> data <- data.frame(x=x, y=y)
> db <- fpc::dbscan(data , eps = 1, MinPts = 3)
> plot(db, data , main = "DBSCAN")
```




#question c

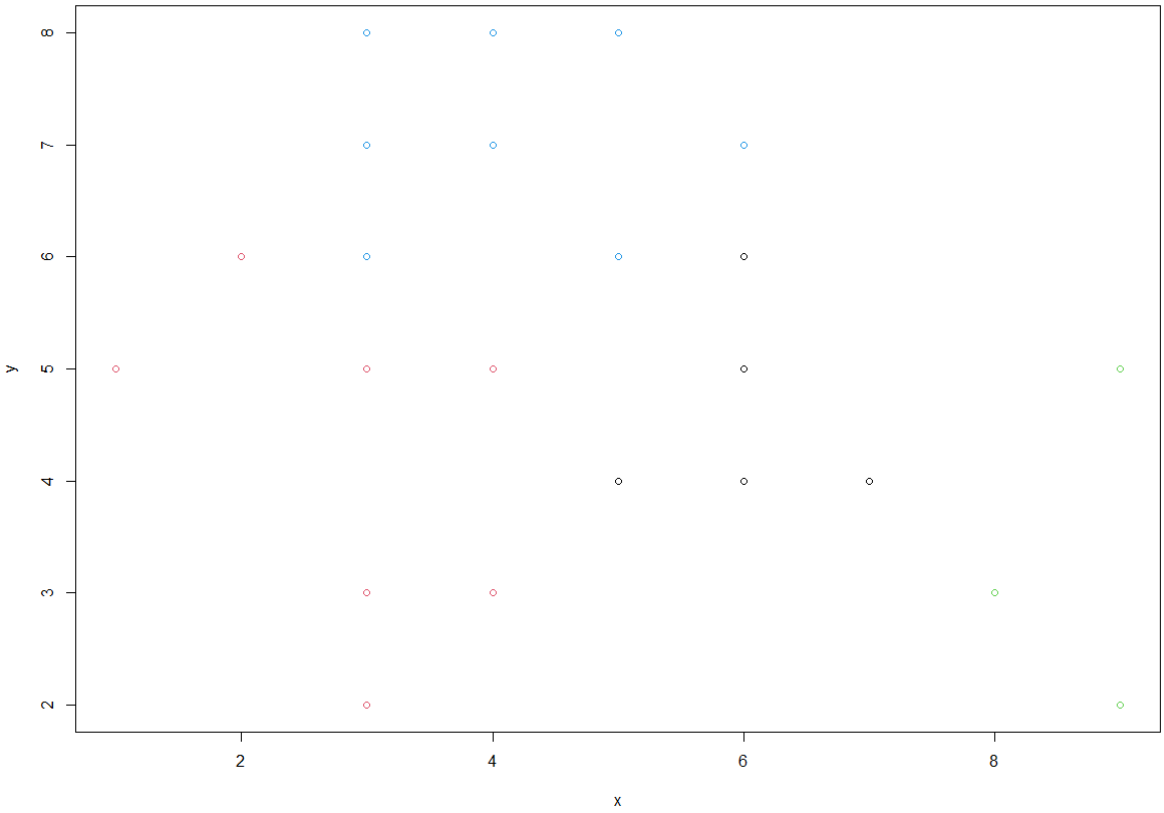
```
# We will try with the elbow rule to find the ideal number of clusters
```

```
> plot_fun <- function(data, nc){
+   wss <- (nrow(data)-1) * sum(apply(data,2,var))
+   for(i in 2:nc){wss[i] <- sum(kmeans(data,centers=i)$withinss)}
+   plot(1:nc, wss, type = "b" ,xlab="Number of Clusters",ylab="Total with in Sum of Square")}
> plot_fun(data, nrow(data)-1)
```




Where the elbow bends is the best number for k, that is where k = 4 will come out

```
> kc <- kmeans(data, 4)
> plot(data, col=kc$cluster)
```




we take out the same clusters with kmeans

## Exercise 13

The table is given:

| | X | Y |
|---|---|---|
| 1 | 0.4005 | 0.5306 |
| 2 | 0.2148 | 0.3854 |
| 3 | 0.3457 | 0.3156 |
| 4 | 0.2652 | 0.1875 |
| 5 | 0.0789 | 0.4139 |
| 6 | 0.4548 | 0.3022 |

Find the dendrogram and the exact values of the distances at which the corresponding clusters are created during the cumulative hierarchical clustering, such as resulting from the application of Euclidean distance in combination with its technique: a) single link, and b) full link

### Solution

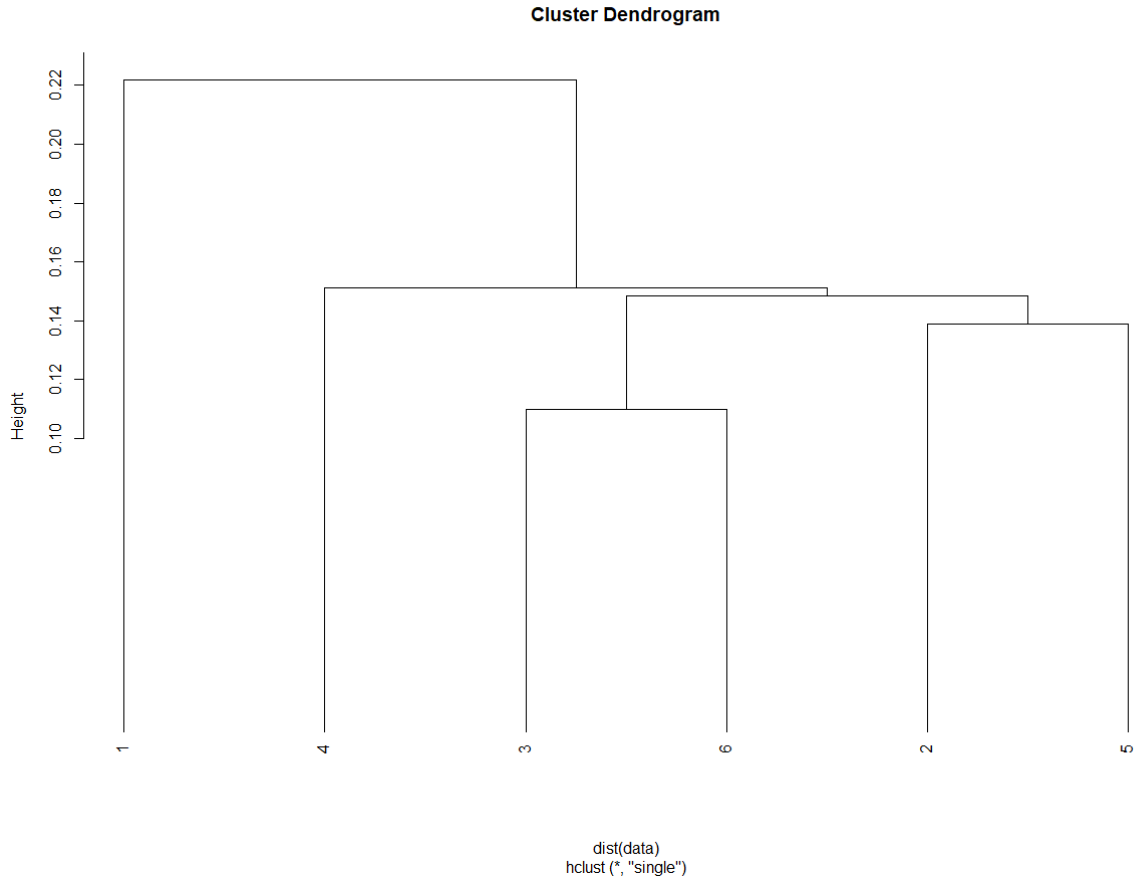
First we make the table for Euclidean distance with the points x and y.

```
> x <- c(0.4005, 0.2148, 0.3457, 0.2652, 0.0789, 0.4548)
> y <- c(0.5306, 0.3854, 0.3156, 0.1875, 0.4139, 0.3022)
```

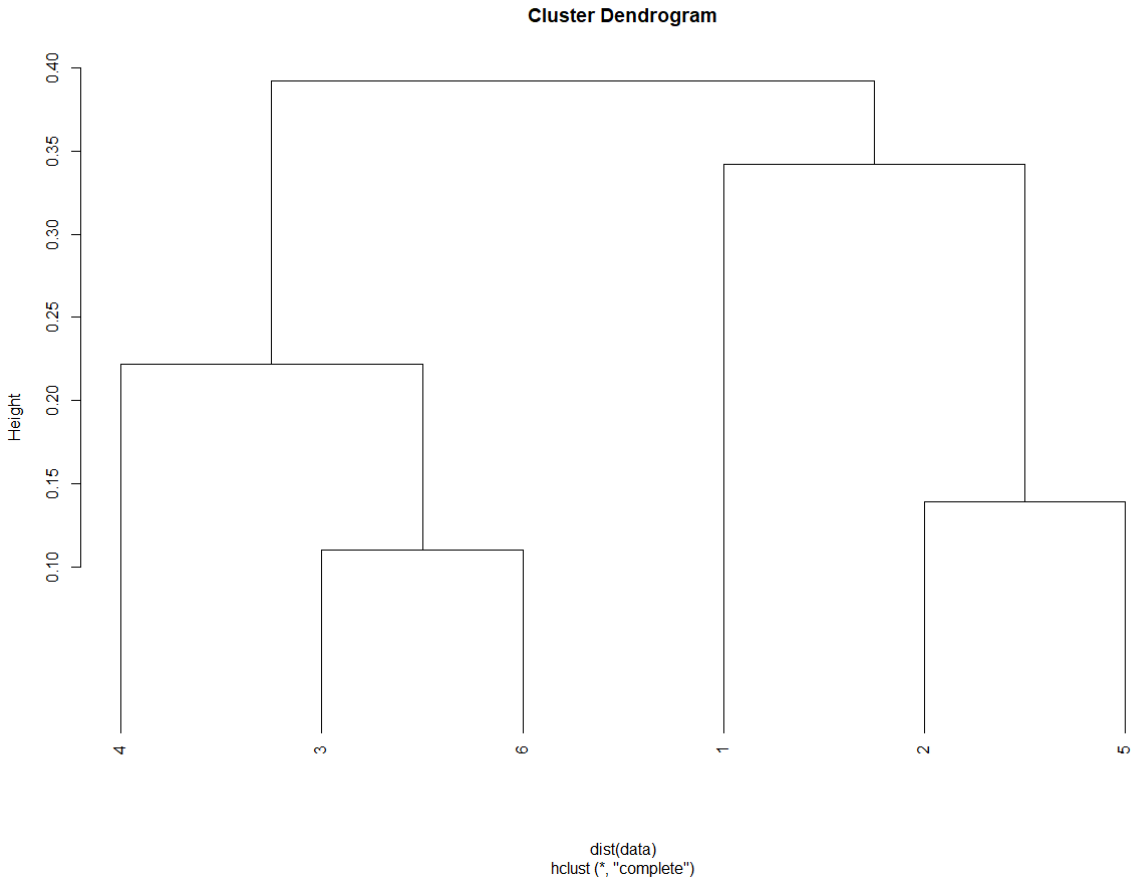
```
> data <- cbind(x,y)
> dist(data, method = "euclidean", diag = TRUE, upper = TRUE)
          1         2         3         4         5         6
1 0.0000000 0.2357277 0.2218739 0.3688139 0.3421191 0.2347659
2 0.2357277 0.0000000 0.1483471 0.2042170 0.1388563 0.2540123
3 0.2218739 0.1483471 0.0000000 0.1512940 0.2843328 0.1099198
4 0.3688139 0.2042170 0.1512940 0.0000000 0.2931973 0.2215948
5 0.3421191 0.1388563 0.2843328 0.2931973 0.0000000 0.3921450
6 0.2347659 0.2540123 0.1099198 0.2215948 0.3921450 0.0000000
```

a)

```
> hc <- hclust(dist(data ), method = "single" )
> plot(hc,hang = -1)
```



b)




# Bibliography

Stock, J. H., & Watson, M. W. (2017). *Εισαγωγή στην Οικονομετρία.* Γ. ΔΑΡΔΑΝΟΣ - Κ. ΔΑΡΔΑΝΟΣ Ο.Ε.

Πανάρετος, Ι. (2003). *Γραμμικά μοντέλα με έμφαση στις εφαρμογές.* Οικονομικό Πανεπιστήμιο Αθηνών.

Χρήστου, Γ. Κ. (2007). *Εισαγωγή στην Οικονομετρία, Τόμος 1.* Γ. ΔΑΡΔΑΝΟΣ - Κ. ΔΑΡΔΑΝΟΣ Ο.Ε.

## Appendix of R codes

```
setwd('C:/Users/Giorgos/Desktop/ergasies_metaptxiakwn/ergasia
filippaki_(predictive
analytics)/ergasia_examinou/ergasia_mfilip_2021/ergasia_mfilip
_2021/dataset_merous_1_kai_2')


# 1)
# install.packages("rpart")
library("rpart")
# Kyphosis
# a factor with levels absent present indicating if a kyphosis
(a type of deformation) was present after the operation.
# Age
# in months
# Number
# the number of vertebrae involved
# Start
# the number of the first (topmost) vertebra operated on.
df = kyphosis
summary(df)
# Boxplot of the variable NUMBER
boxplot(df$Number, main="The Variable: Number",  ylab="The
number of vertebrae involved")

outliers = df$Number[df$Number>8]
outliers
which( df$Number %in% outliers)

plot(df$Number, df$Age)
identify(df$Number, df$Age)


# 2)
df = read.csv("capital.csv", sep = ';')
summary(df)
#df$gender <- as.factor(df$gender)

# 2i) πίνακας σχετικών συχνοτήτων
xtabs(balance ~ gender , data=df)
with(df, discretePlot(gender, scale="frequency"))
barplot( table(df$gender) )
prop.table(table(df$gender))

# Simple Pie Chart
bal_gender <- c(173191, 52913)
lbls <- c("Male", "Female")
pie(bal_gender, labels = lbls, main="Balance by gender")
```

```
# 2ii) Boxplot by Group
boxplot(balance ~ gender, data=df)

# 2iii)
library(psych)
describeBy(df$balance, df$gender)

summary(df$balance)
library(pastecs)
stat.desc(df$balance)

# 2iv)
library("car")
qqPlot(df$balance)


# 3)
data(mtcars)
head(mtcars)
summary(mtcars)
str(mtcars)
#df = read.csv("mtcars.csv", sep = ',')

t.test( mpg ~ am, data = mtcars,conf.level = 0.95)


# 4)
df <- read.delim("OctopusF.txt")
summary(df$Weight)
library(pastecs)
stat.desc(df$Weight)

hist(df$Weight)

library("car")
qqPlot(df$Weight)

# Calculate the mean and standard error
l.model <- lm(df$Weight ~ 1, df)
# Calculate the confidence interval
confint(l.model, level=0.95)


# 5)
library(MASS)
df = survey
contigency_table = table(df$Smoke, df$Exer)
contigency_table
chisq.test(contigency_table)
```

```
# 6)
# Loading
library("readxl")
# xls files
df <- read_excel("Concrete_Data.xls")
str(df)
df <-scaleddata<-scale(df)
# Training and Test Data
set.seed(653)
df <- df[sample(nrow(df)), ]
trainset <- df[1:721, ]
testset <- df[722:1030, ]
#Neural Network
library(neuralnet)
nn <- neuralnet(Concrete ~ Cement + Slag + Ash + Water +
Superplasticizer + CoarseAggregate + FineAggregate + Age,
data=trainset, hidden=c(4,1), linear.output=FALSE,
threshold=0.5)
nn$result.matrix
plot(nn)
#Test the resulting output
temp_test <- subset(testset, select = c("Cement", "Slag",
"Ash", "Water", "Superplasticizer", "CoarseAggregate",
"FineAggregate", "Age"))
head(temp_test)
nn.results <- compute( nn, temp_test )
results <- data.frame( actual = testset$Concrete, prediction =
nn.results$net.result )
roundedresults<-sapply(results,round,digits=0)
roundedresultsdf=data.frame(roundedresults)
attach(roundedresultsdf)
table(actual,prediction)


# 7)
library(ggplot2)

df <- read.delim("faithfull.txt")
str(df)
summary(df$Weight)

reg.lm <- lm(eruptions ~ waiting, data = df)
summary(reg.lm)

x80.dat <- data.frame(waiting = 80)
predict(reg.lm, newdata = x80.dat, interval = 'confidence')
predict(reg.lm, newdata = x80.dat, interval = 'prediction')

eruption.res = resid(reg.lm)
```

```
plot(df$waiting, eruption.res, ylab="Residuals", xlab="Waiting
Time", main="Eruptions")
abline(0, 0)

library("car")
qqPlot(eruption.res)


# 8)
library(MASS)
data(stackloss)
str(stackloss)
stackloss.lm <- lm(stack.loss ~ Air.Flow + Water.Temp +
Acid.Conc., data = stackloss)
summary(stackloss.lm)
new_data = data.frame(Air.Flow=72, Water.Temp=20,
Acid.Conc.=85)
predict(stackloss.lm, new_data)
predict(stackloss.lm, newdata = new_data, interval =
'confidence')
predict(stackloss.lm, newdata = new_data, interval =
'prediction')


# 9)
# Loading
library("readxl")
# xls files
df <- read_xlsx("market.xlsx")
market.lm <- lm(Sales ~ Preis + Costs + Arrivals, data = df)
summary(market.lm)
# x-correlation
cor(df[,2:4])
# Interactions
market_inter.lm <- lm(Sales ~ Preis + Costs + Arrivals +
Preis*Costs + Preis*Arrivals + Costs*Arrivals, data = df)
summary(market_inter.lm)
# Standardized Residuals
market.stdres = rstandard(market.lm)
summary(market.stdres)
# Residuals
market.res = residuals(market.lm)
summary(market.res)
# Boxplot()
# 2ii) Boxplot by Group
boxplot(market.stdres, data=df)
# QQ-plot
qqPlot(market.stdres)
plot(market.lm)
```

```
lmtest::bptest(market.lm)  # Breusch-Pagan test


# 10)
# I)
df = read.csv("insurance.csv", sep = ',')
df$sex = as.factor(df$sex)
df$smoker = as.factor(df$smoker)
df$region = as.factor(df$region)
df <- sapply(df, unclass)
# II)
library(psych)
library(psychTools)  #additional tools and data are here
describe(df)  #basic descriptive statistics
corr.test(df)
# III)
setCor(y = 7,x=1:6, data = df)


#11)
# Loading
library("readxl")
# xls files
df <- read_excel("mf.xls")

# I)
df$amount_cut = cut(df$`Dollar Claim Amount`,3)

tb1 = table(df$amount_cut, df$Shift)
chi2_1 = chisq.test(tb1)
chi2_1$p.value

tb2 = table(df$amount_cut, df$`Complaint Code`)
chi2_2 = chisq.test(tb2)
chi2_2$p.value

tb3 = table(df$amount_cut, df$`Manufacturing Plant`)
chi2_3 = chisq.test(tb3)
chi2_3$p.value

tb4 = table(df$`Complaint Code`, df$Shift)
chi2_4 = chisq.test(tb4)
chi2_4$p.value

tb5 = table(df$`Complaint Code`, df$`Manufacturing Plant`)
chi2_5 = chisq.test(tb5)
chi2_5$p.value


# II)
tab1 = table(df$`Complaint Code`, df$`Manufacturing Plant`)
```

```
chi2 = chisq.test(tab1)
chi2$expected
chi2$p.value

# III)
df1 = df[,c(1,4)]
df1 = df1[ df1$`Manufacturing Plant` < 3,  ]
t.test(df1$`Dollar Claim Amount` ~ df1$`Manufacturing Plant`,
data = df1)

#12)
#II)
library("fpc")
library("dbscan")
x <- c(1, 2, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 6, 6, 6,
6, 7, 8, 9, 9)
y <- c(5, 6, 2, 3, 5, 6, 7, 8, 3, 5, 7, 8, 4, 6, 8, 4, 5, 6,
7, 4, 3, 2, 5)
data <- data.frame(x=x, y=y)
db <- fpc::dbscan(data , eps = 1, MinPts = 3)
plot(db, data , main = "DBSCAN")

#III)
plot_fun <- function(data, nc){
  wss <- (nrow(data)-1) * sum(apply(data,2,var))
  for(i in 2:nc){wss[i] <-
sum(kmeans(data,centers=i)$withinss)}
  plot(1:nc, wss, type = "b" ,xlab="Number of
Clusters",ylab="Total within Sum of Square")}

plot_fun(data, nrow(data)-1)

kc <- kmeans(data, 4)
plot(data, col=kc$cluster)


#13)
x <- c(0.4005, 0.2148, 0.3457, 0.2652, 0.0789, 0.4548)
y <- c(0.5306, 0.3854, 0.3156, 0.1875, 0.4139, 0.3022)
data <- cbind(x,y)
#πίνακας για ευκλειδεια αποσταση
dist(data, method = "euclidean", diag = TRUE, upper = TRUE)
#α ερωτημα
hc <- hclust(dist(data ), method = "single" )
plot(hc,hang = -1)

#β ερωτημα
hc <- hclust(dist(data), method = "complete")
plot(hc, hang = -1)
```