Problem Set 1

Anna Bellaver 918299, Maria Carta 917645, Roberto Di Cunto 894190, Martina Mattocci 1089750, Davide Valentini 1088760

Table of contents

Exercise 1

The data set state.x77 (package dataset) contains 8 variables recorded to the 50 states of the United States of America in year 1977.

```r
head(state.x77)
```

```
           Population Income Illiteracy Life Exp Murder HS Grad Frost   Area
Alabama          3615   3624        2.1    69.05   15.1    41.3    20  50708
Alaska            365   6315        1.5    69.31   11.3    66.7   152 566432
Arizona          2212   4530        1.8    70.55    7.8    58.1    15 113417
Arkansas         2110   3378        1.9    70.66   10.1    39.9    65  51945
California      21198   5114        1.1    71.71   10.3    62.6    20 156361
Colorado         2541   4884        0.7    72.06    6.8    63.9   166 103766
```

```r
st<-as.data.frame(state.x77)
head(st)
```

```
           Population Income Illiteracy Life Exp Murder HS Grad Frost   Area
Alabama          3615   3624        2.1    69.05   15.1    41.3    20  50708
Alaska            365   6315        1.5    69.31   11.3    66.7   152 566432
Arizona          2212   4530        1.8    70.55    7.8    58.1    15 113417
Arkansas         2110   3378        1.9    70.66   10.1    39.9    65  51945
California      21198   5114        1.1    71.71   10.3    62.6    20 156361
Colorado         2541   4884        0.7    72.06    6.8    63.9   166 103766
```

```r
names(st)[4] = "Life.Exp"
names(st)[6] = "HS.Grad"
st[,9] = st$Population * 1000 / st$Area
colnames(st)[9] = "Density"
dim(st)
```

```
[1] 50  9
```

Point 1

Compute the correlation matrix and comment on the most relevant relationships among variables (up to 10).

Now we look for the Correlation Matrix **R** which entries are defined as $r_{jk} = \frac{s_{jk}}{s_j s_k}$. In order to have a clearer view of the correlations we can put them in a matrix ordering them from the highest to the lowest ones:

```
R<-cor(st)
round(R,3)
```

```
           Population Income Illiteracy Life.Exp Murder HS.Grad  Frost   Area
Population      1.000  0.208      0.108   -0.068  0.344  -0.098 -0.332  0.023
Income          0.208  1.000     -0.437    0.340 -0.230   0.620  0.226  0.363
Illiteracy      0.108 -0.437      1.000   -0.588  0.703  -0.657 -0.672  0.077
Life.Exp       -0.068  0.340     -0.588    1.000 -0.781   0.582  0.262 -0.107
Murder          0.344 -0.230      0.703   -0.781  1.000  -0.488 -0.539  0.228
HS.Grad        -0.098  0.620     -0.657    0.582 -0.488   1.000  0.367  0.334
Frost          -0.332  0.226     -0.672    0.262 -0.539   0.367  1.000  0.059
Area            0.023  0.363      0.077   -0.107  0.228   0.334  0.059  1.000
Density         0.246  0.330      0.009    0.091 -0.185  -0.088  0.002 -0.341
           Density
Population   0.246
Income       0.330
Illiteracy   0.009
Life.Exp     0.091
Murder      -0.185
HS.Grad     -0.088
Frost        0.002
Area        -0.341
Density      1.000
```

```
R<-abs(R)
lower.tri(R)
```

```
      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9]
 [1,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [2,]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [3,]  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [4,]  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
 [5,]  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE
 [6,]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE
 [7,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE
 [8,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE
 [9,]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
```

```r
R[!lower.tri(R)]<-NA
round(R, 3)
```

```
           Population Income Illiteracy Life.Exp Murder HS.Grad Frost  Area
Population         NA     NA         NA       NA     NA      NA    NA    NA
Income          0.208     NA         NA       NA     NA      NA    NA    NA
Illiteracy      0.108  0.437         NA       NA     NA      NA    NA    NA
Life.Exp        0.068  0.340      0.588       NA     NA      NA    NA    NA
Murder          0.344  0.230      0.703    0.781     NA      NA    NA    NA
HS.Grad         0.098  0.620      0.657    0.582  0.488      NA    NA    NA
Frost           0.332  0.226      0.672    0.262  0.539   0.367    NA    NA
Area            0.023  0.363      0.077    0.107  0.228   0.334 0.059    NA
Density         0.246  0.330      0.009    0.091  0.185   0.088 0.002 0.341
           Density
Population      NA
Income          NA
Illiteracy      NA
Life.Exp        NA
Murder          NA
HS.Grad         NA
Frost           NA
Area            NA
Density         NA
```

```r
sort(R, decreasing=T, na.last=NA)
```

```
 [1] 0.780845752 0.702975199 0.671946968 0.657188609 0.619932323 0.588477926
 [7] 0.582216204 0.538883437 0.487971022 0.437075186 0.366779702 0.363315438
[13] 0.343642751 0.341388515 0.340255339 0.333541871 0.332152454 0.329968277
[19] 0.262068011 0.246227888 0.230077610 0.228390211 0.226282179 0.208227557
[25] 0.185035233 0.107622373 0.107331935 0.098489748 0.091061763 0.088367214
[31] 0.077261132 0.068051952 0.059229102 0.022543837 0.009274348 0.002276734
```

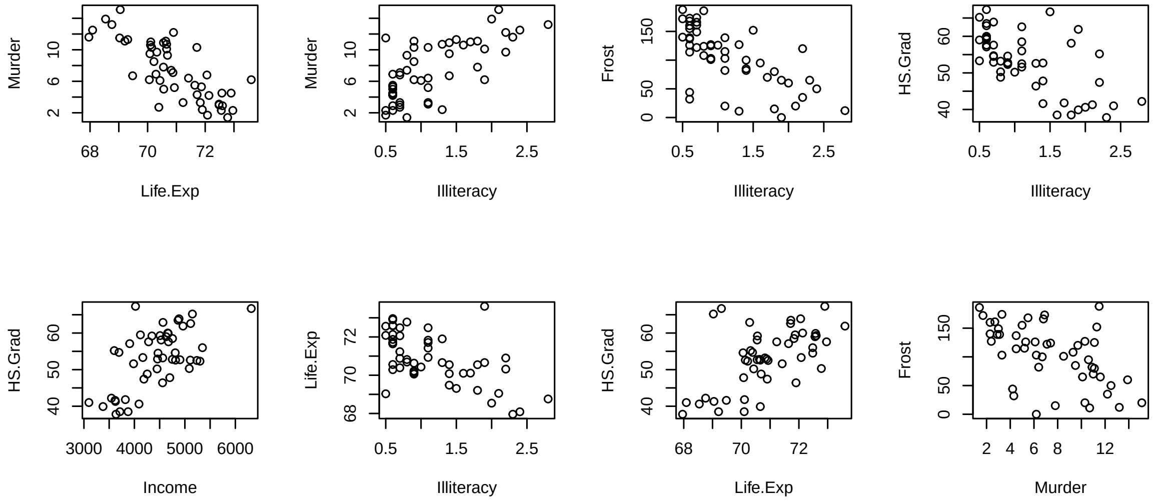
```
order(R, decreasing=T, na.last=NA)
```

```
 [1] 32 23 25 24 15 22 33 43 42 12 52 17  5 72 13 53  7 18 34  9 14 44 16  2 45
[26]  3 35  6 36 54 26  4 62  8 27 63
```

From the previous Matrix of ordered correlation we can view the higher correlations in absolute value (>0.5) and comment the less intuitive ones:

- R[32]=0.781 → negative correlation between Life.Exp and Murder as we could expect and is the highest correlation that we have.
- R[23]=0.703 → positive correlation between Illiteracy and Murder
- R[25]=0.672 → negative correlation between Illiteracy and Frost. One possible explanation of this correlation is that the variable ”Frost” could be a geographic proxy of the northern states. In general the southern states have a lower literacy rate than the northern ones.
- R[24]=0.657 → negative correlation between Illiteracy and HS.Grad
- R[15]=0.620 → positive correlation between Income and HS.Grad
- R[22]=0.588 → negative correlation between Illiteracy and Life.Exp
- R[33]=0.582 → positive correlation between Life.Exp and HS.Grad
- R[43]=0.539 → negative correlation between Murder and Frost. The ”Frost” variable is influenced by the geographical position. As known, the southern states have a higher murder rate than the northern ones.

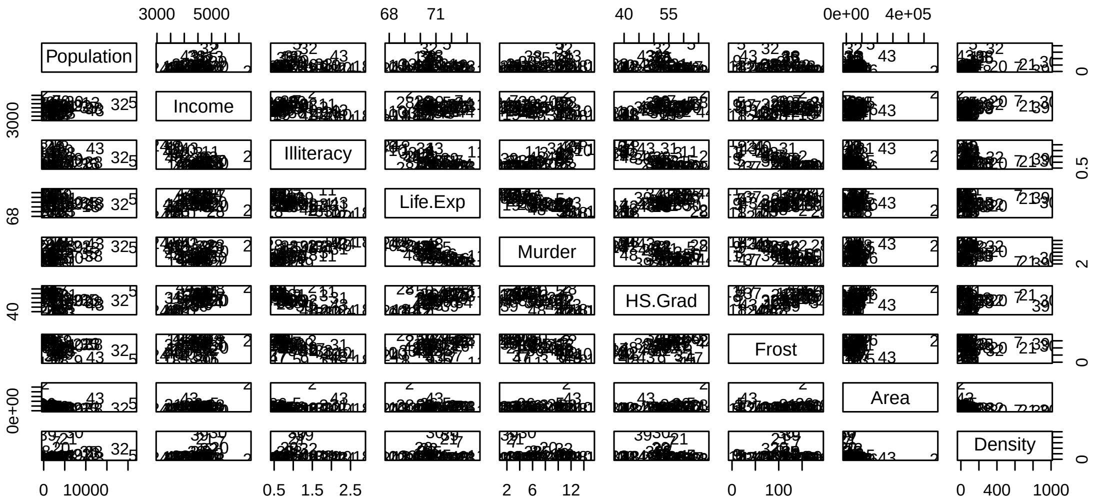
```
par(mfrow=c(2, 4))
plot(st$Life.Exp,st$Murder, xlab = "Life.Exp", ylab = "Murder")
plot(st$Illiteracy,st$Murder, xlab = "Illiteracy", ylab = "Murder")
plot(st$Illiteracy,st$Frost, xlab="Illiteracy", ylab="Frost")
plot(st$Illiteracy,st$HS.Grad, xlab="Illiteracy", ylab="HS.Grad")
plot(st$Income,st$HS.Grad, xlab="Income", ylab="HS.Grad")
plot(st$Illiteracy,st$Life.Exp, xlab="Illiteracy", ylab="Life.Exp")
plot(st$Life.Exp,st$HS.Grad, xlab="Life.Exp", ylab="HS.Grad")
plot(st$Murder,st$Frost, xlab="Murder", ylab="Frost")
```



Point 2

Find univariate outliers, up to 3 per variable, up to 10 in total.

```
j<-1
j<-j+1
x<-st[,j]
n<-nrow(st)
p<-ncol(st)
pairs(st, panel=function(x, y) text(x, y, labels=1:n))
```



From the scatterplot we can see the principal outliers but we can obtain also a list:

```
X<-scale(st)
which(abs(X)>qnorm(0.95), arr.ind=T)
```

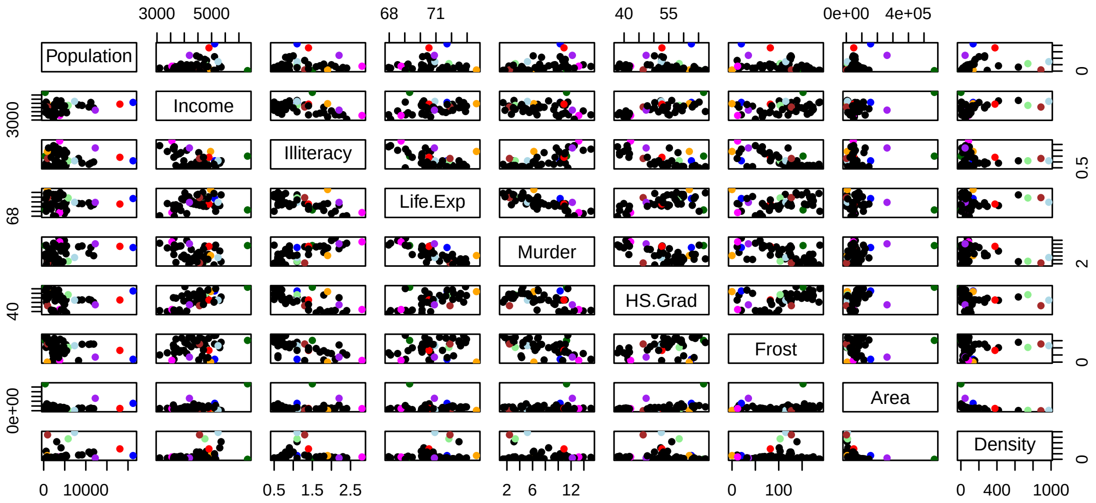
```
               row col
California       5   1
New York        32   1
Pennsylvania    38   1
Texas           43   1
Alaska           2   2
Arkansas         4   2
Mississippi     24   2
Louisiana       18   3
Mississippi     24   3
New Mexico      31   3
South Carolina  40   3
Texas           43   3
Georgia         10   4
Hawaii          11   4
Mississippi     24   4
South Carolina  40   4
Alabama          1   5
Georgia         10   5
```

```
Alaska           2   6
Kentucky        17   6
North Carolina  33   6
South Carolina  40   6
Utah            44   6
Arizona          3   7
Florida          9   7
Hawaii          11   7
Louisiana       18   7
Alaska           2   8
Texas           43   8
Connecticut      7   9
Massachusetts   21   9
New Jersey      30   9
Rhode Island    39   9
```

Comparing the scatterplot with the list we can select the principal outliers:

- Population: 5=”California”, 32=”New York”, 43=”Texas”;
- Income: 2=”Alaska”;
- Illiteracy: 18=”Louisiana”;
- Life.Exp: 11=”Hawaii”;
- Murder: no relevant;
- HS.Grad: no relevant;
- Frost: 11=”Hawaii”;
- Area: 2=”Alaska”, 43=”Texas”;
- Density: 21=”Massachusetts”, 30=”New Jersey”, 39=”Rhode Island”.

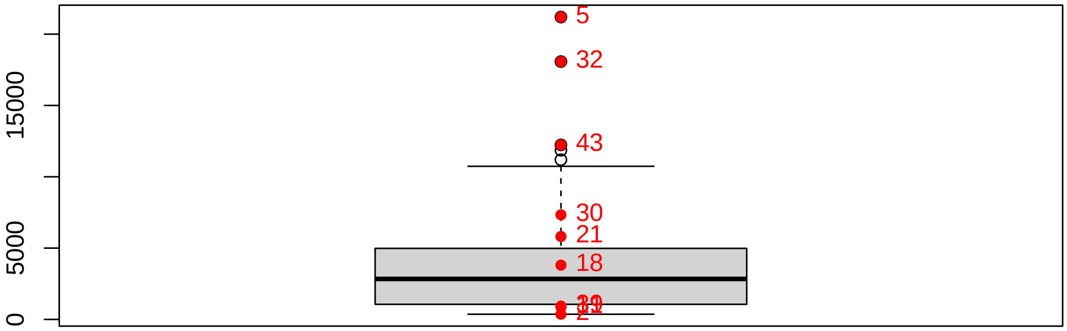
```
index<-c(2,5,11,18,21,30,32,39,43)
col.index<-rep("black", n)
col.index[2]<-"darkgreen"
col.index[5]<-"blue"
col.index[11]<-"orange"
col.index[18]<-"magenta"
col.index[21]<-"lightgreen"
col.index[30]<-"lightblue"
col.index[32]<-"red"
col.index[39]<-"brown"
col.index[43]<-"purple"
col.ind<-c("darkgreen", "blue", "orange", "magenta", "lightgreen",
           "lightblue", "red", "brown", "purple")

pairs(st, pch=16, col=col.index)
```



Point 3

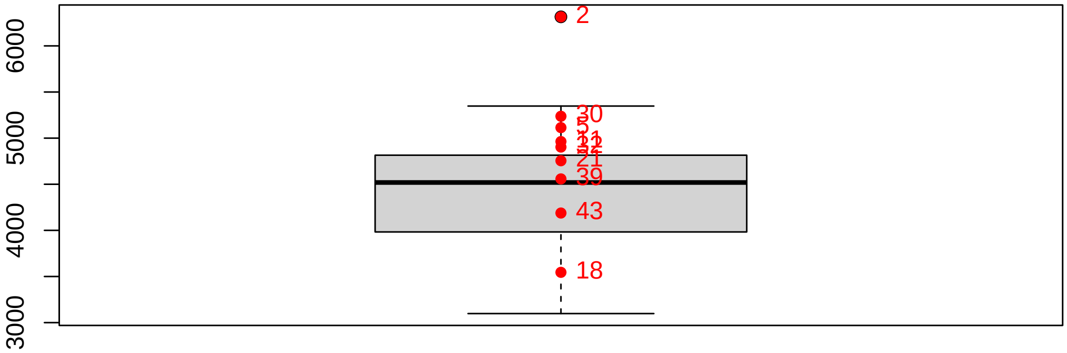
Make a boxplot of any variable plotting the corresponding outliers, if any, found in point 2 in red.

```
c1<-rep(1, 9)
ind.names<-as.character(index)
boxplot(st$Population)
points(c1,st$Population[index],col="red",pch=16,main="Population")
text(c1,st$Population[index],ind.names,pos=4,col="red")
```



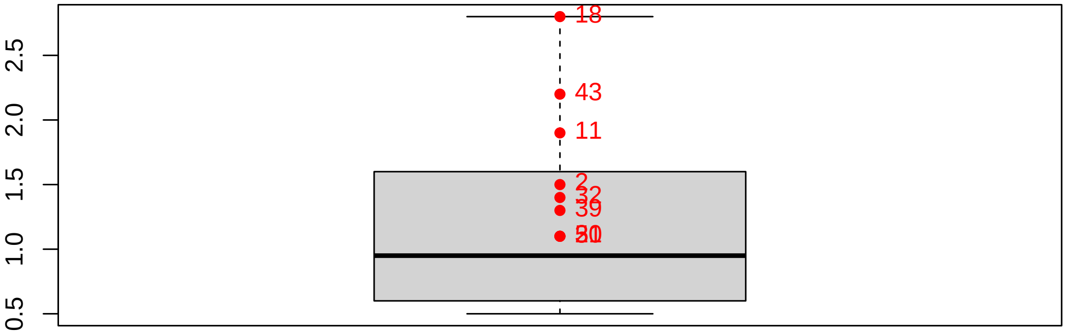
The outliers of Population founded in point 2 are 5="California", 32="New York", 43="Texas".

```
boxplot(st$Income)
points(c1,st$Income[index],col="red",pch=16, main="Income")
text(c1,st$Income[index],ind.names,pos=4,col="red")
```



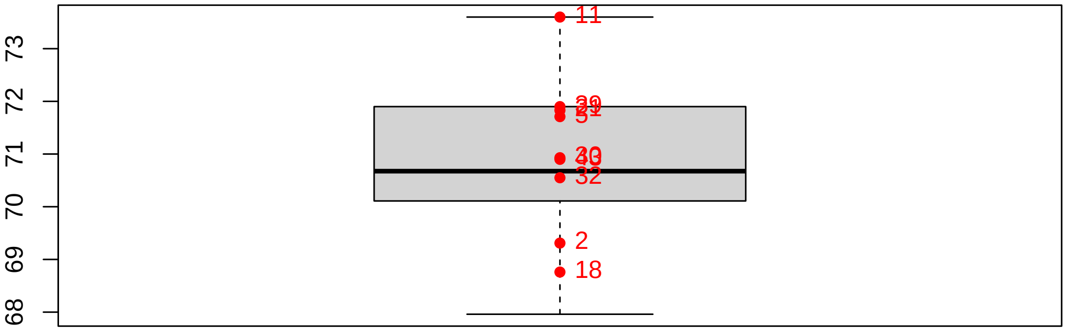
The outliers of Income founded in point 2 is 2="Alaska".

```
boxplot(st$Illiteracy)
points(c1,st$Illiteracy[index],col="red",pch=16, main="Illiteracy")
text(c1,st$Illiteracy[index],ind.names,pos=4,col="red")
```



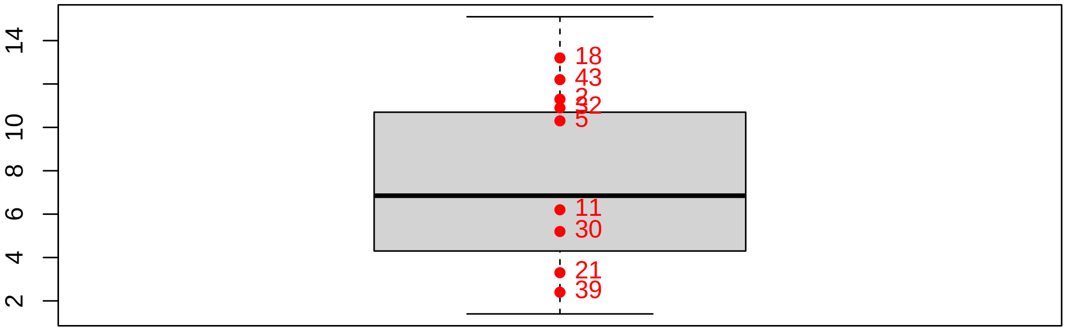
The outliers of Illiteracy founded in point 2 is 18="Louisiana".

```
boxplot(st$Life.Exp)
points(c1,st$Life.Exp[index],col="red",pch=16, main="Life.Exp")
text(c1,st$Life.Exp[index],ind.names,pos=4,col="red")
```



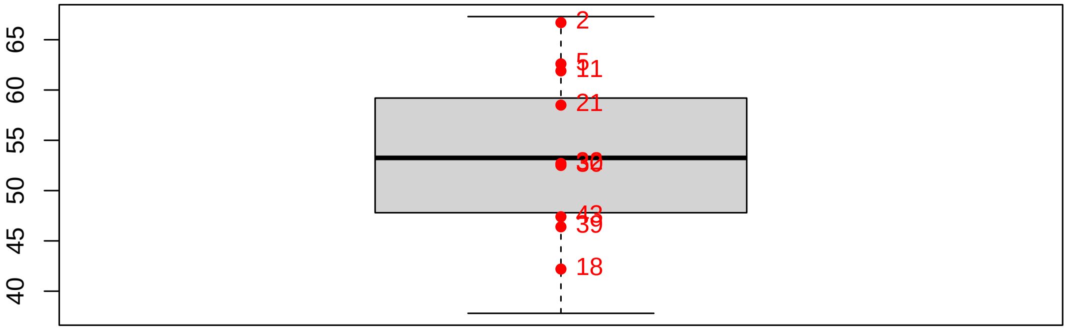
The outliers of Life.Exp founded in point 2 is 11="Hawaii".

```
boxplot(st$Murder)
points(c1,st$Murder[index],col="red",pch=16, main="Murder")
text(c1,st$Murder[index],ind.names,pos=4,col="red")
```



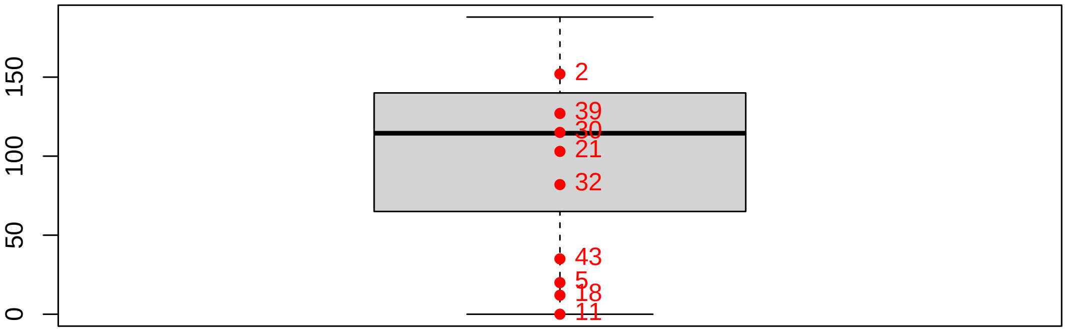
There are no relevant outliers for Murder.

```r
boxplot(st$HS.Grad)
points(c1,st$HS.Grad[index],col="red",pch=16, main="HS.Grad")
text(c1,st$HS.Grad[index],ind.names,pos=4,col="red")
```



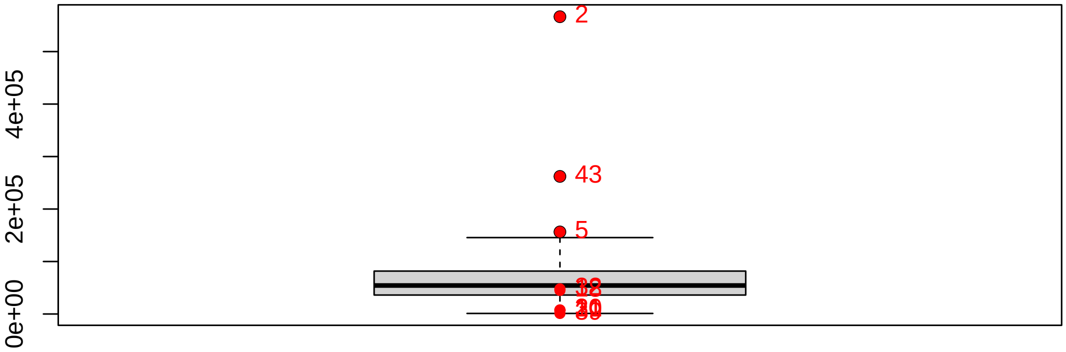
There are no relevant outliers for HS.Grad.

```r
boxplot(st$Frost)
points(c1,st$Frost[index],col="red",pch=16, main="Frost")
text(c1,st$Frost[index],ind.names,pos=4,col="red")
```



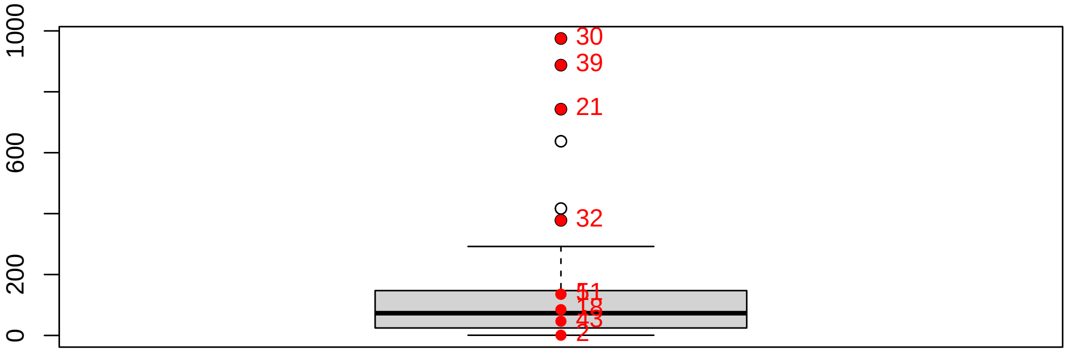
The outliers of Frost founded in point 2 is 11="Hawaii".

```
boxplot(st$Area)
points(c1,st$Area[index],col="red",pch=16, main="Area")
text(c1,st$Area[index],ind.names,pos=4,col="red")
```



The outliers of Area founded in point 2 are 2="Alaska", 43="Texas".

```
boxplot(st$Density)
points(c1,st$Density[index],col="red",pch=16, main="Density")
text(c1,st$Density[index],ind.names,pos=4,col="red")
```



The outliers of Density founded in point 2 are 21="Massachusetts", 30="New Jersey", 39="Rhode Island".

Point 4

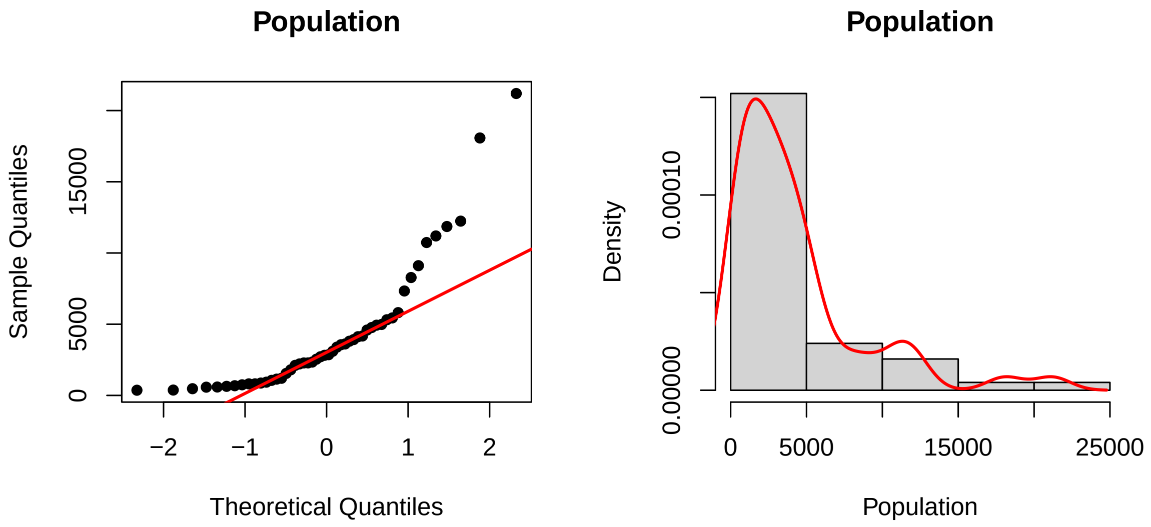
Comment about normality of each variable.

The Q-Q plot will show the relationship between the *"Theoretical Quantiles"* and the *"Sample Quantiles"* observed for each measurement ($x_{(1)} \leq ... \leq x_{(i)} \leq x_{(n)} \quad i = 1, ..., n$ where $x_{(i)}$ is the $\frac{i-0.05}{n}$ quantile). The line plotted is the so called Q-Q line that represents the equivalence between the Theoretical (Chi-square Quantiles) and Sample Quantiles.

Then we'll plot the histogram with the estimated density function (the red curve) to obtain another element for the evaluation, and finally we'll show the results of the Shapiro-Wilks normality test which for low level of the *p-value* (<0.05) reject the hypothesis of normality.

```
par(mfrow=c(1, 2))
qqnorm(st$Population, main=names(st)[1], pch=16)
```

```
qqline(st$Population, col="red", lwd=2)
hist(st$Population, main=names(st)[1], freq=F,xlab = "Population")
lines(density(st$Population), col="red", lwd=2)
```

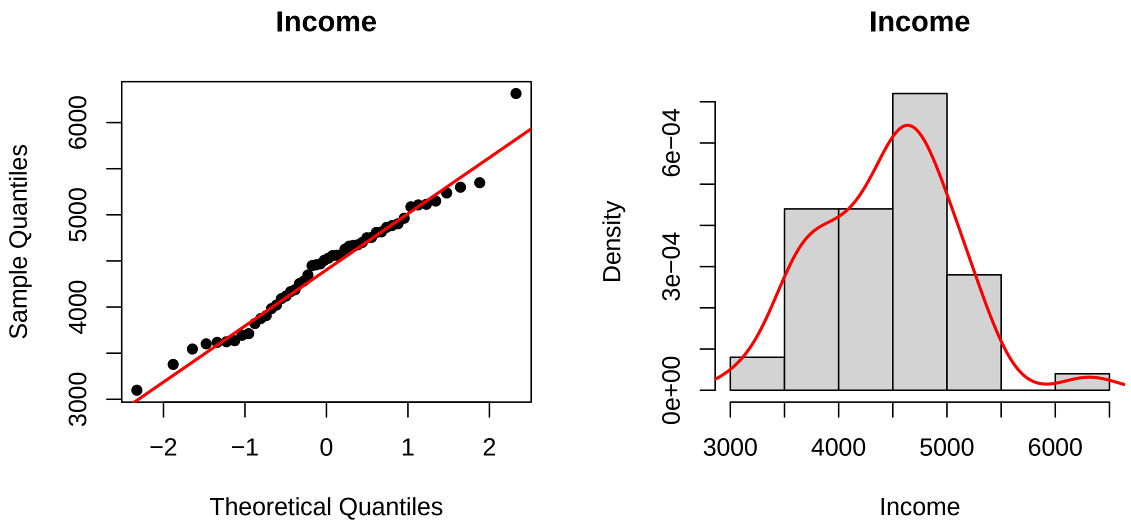


```
shapiro.test(st$Population)
```

```
	Shapiro-Wilk normality test

data:  st$Population
W = 0.76999, p-value = 1.906e-07
```

From the shapiro-test we obtain p-value = 1.906e-07 so we have to reject normality, as we can see from the qqplot and the histogram.

```
par(mfrow=c(1, 2))
qqnorm(st$Income, main=names(st)[2], pch=16)
qqline(st$Income, col="red", lwd=2)
hist(st$Income, main=names(st)[2], freq=F,xlab = "Income")
lines(density(st$Income), col="red", lwd=2)
```

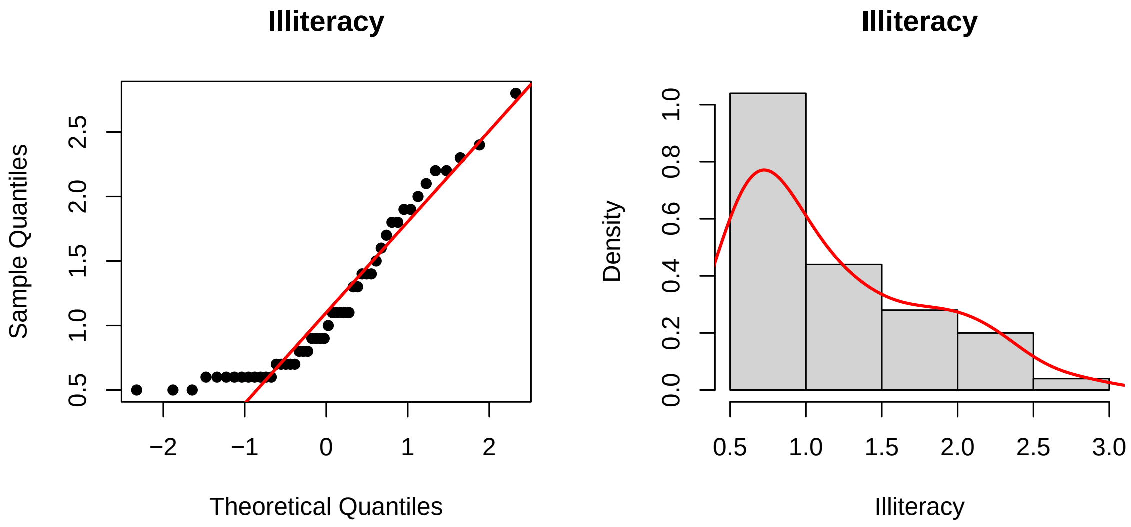


```
shapiro.test(st$Income)
```

```

	Shapiro-Wilk normality test

data:  st$Income
W = 0.9769, p-value = 0.43
```

From the shapiro-test we obtain p-value = 0.43 so we cannot reject normality, as we can see from the qqplot and the histogram.

```
par(mfrow=c(1, 2))
qqnorm(st$Illiteracy, main=names(st)[3], pch=16)
qqline(st$Illiteracy, col="red", lwd=2)
hist(st$Illiteracy, main=names(st)[3], freq=F,xlab = "Illiteracy")
lines(density(st$Illiteracy), col="red", lwd=2)
```

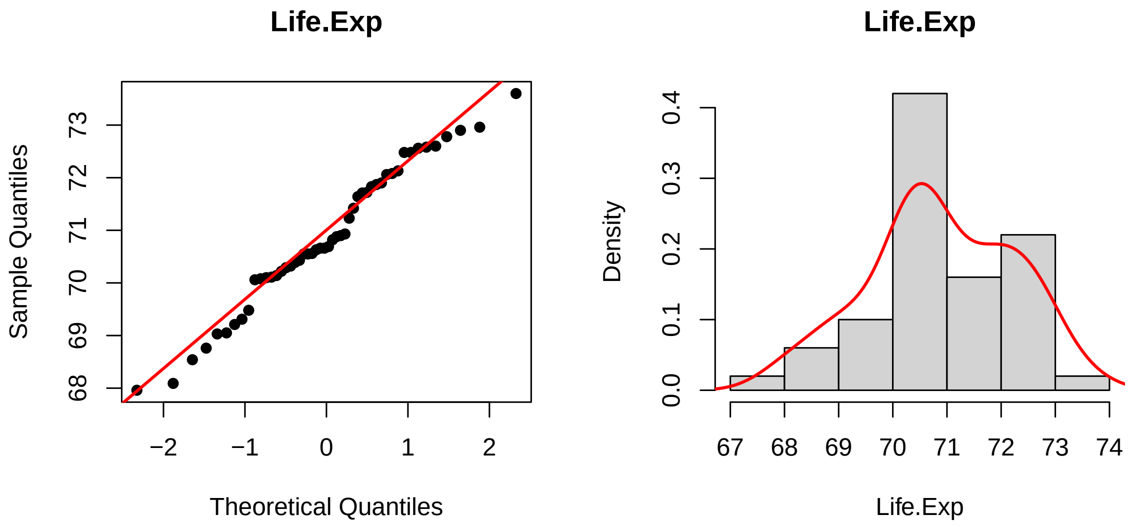


```
shapiro.test(st$Illiteracy)
```

```
	Shapiro-Wilk normality test

data:  st$Illiteracy
W = 0.88315, p-value = 0.0001396
```

From the shapiro-test we obtain p-value = 0.0001396 so we have to reject normality, as we can see from the qqplot and the histogram.

```
par(mfrow=c(1, 2))
qqnorm(st$Life.Exp, main=names(st)[4], pch=16)
qqline(st$Life.Exp, col="red", lwd=2)
hist(st$Life.Exp, main=names(st)[4], freq=F,xlab = "Life.Exp")
lines(density(st$Life.Exp), col="red", lwd=2)
```

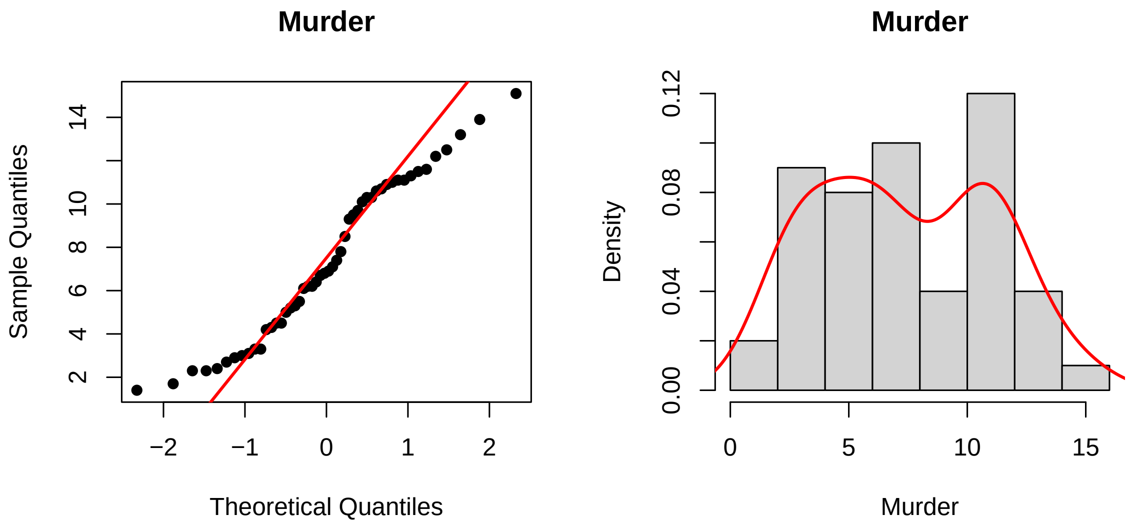


```
shapiro.test(st$Life.Exp)
```

```
	Shapiro-Wilk normality test

data:  st$Life.Exp
W = 0.97724, p-value = 0.4423
```

From the shapiro-test we obtain p-value $= 0.4423$ so we cannot reject normality, as we can see from the qqplot and the histogram.

```
par(mfrow=c(1, 2))
qqnorm(st$Murder, main=names(st)[5], pch=16)
qqline(st$Murder, col="red", lwd=2)
hist(st$Murder, main=names(st)[5], freq=F,xlab = "Murder")
lines(density(st$Murder), col="red", lwd=2)
```

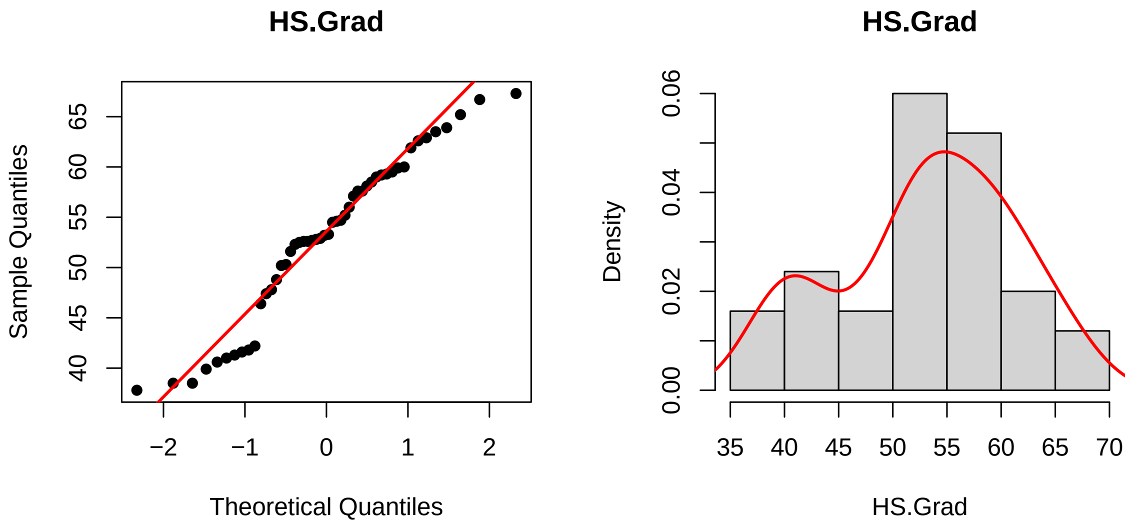


```
shapiro.test(st$Murder)
```

```
	Shapiro-Wilk normality test

data:  st$Murder
W = 0.95347, p-value = 0.04745
```

From the shapiro-test we obtain p-value = 0.04745 so we have to reject normality, as we can see from the qqplot and the histogram.

```
par(mfrow=c(1, 2))
qqnorm(st$HS.Grad, main=names(st)[6], pch=16)
qqline(st$HS.Grad, col="red", lwd=2)
hist(st$HS.Grad, main=names(st)[6], freq=F,xlab = "HS.Grad")
lines(density(st$HS.Grad), col="red", lwd=2)
```

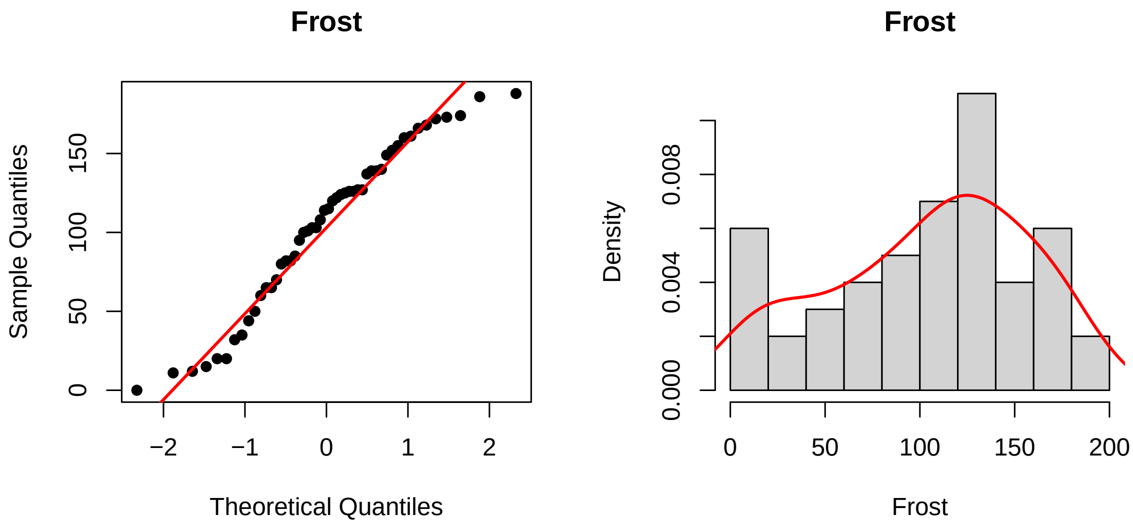


```r
shapiro.test(st$HS.Grad)
```

```

	Shapiro-Wilk normality test

data:  st$HS.Grad
W = 0.9531, p-value = 0.04582
```

From the shapiro-test we obtain p-value = 0.04582 so we have to reject normality, as we can see from the qqplot and the histogram.

```r
par(mfrow=c(1, 2))
qqnorm(st$Frost, main=names(st)[7], pch=16)
qqline(st$Frost, col="red", lwd=2)
hist(st$Frost, main=names(st)[7], freq=F,xlab = "Frost")
lines(density(st$Frost), col="red", lwd=2)
```

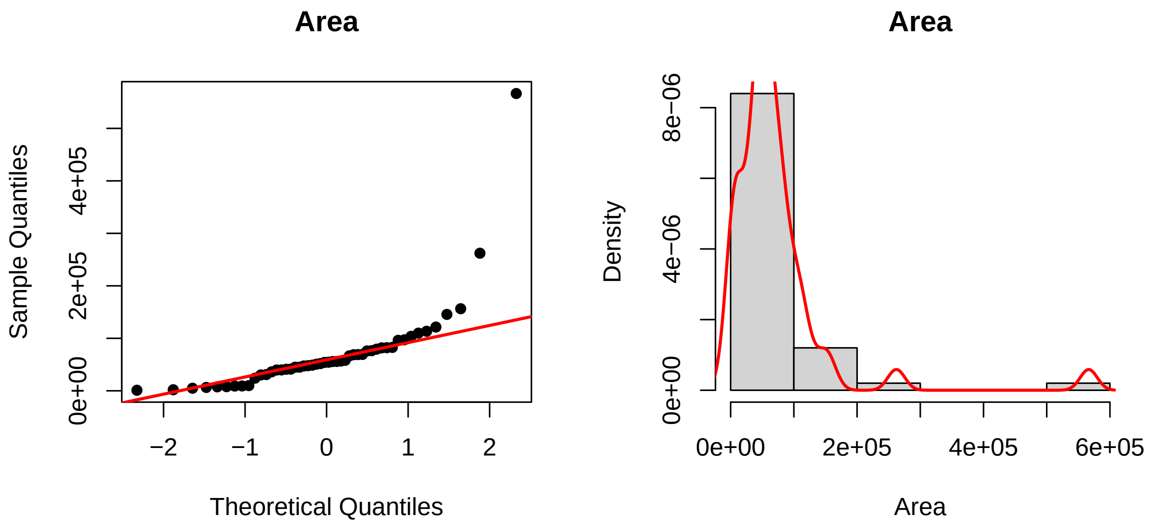


```
shapiro.test(st$Frost)
```

```
    Shapiro-Wilk normality test

data:  st$Frost
W = 0.95456, p-value = 0.05267
```

From the shapiro-test we obtain p-value = 0.05267 so we cannot reject normality, as we can see from the qqplot and the histogram.

```
par(mfrow=c(1, 2))
qqnorm(st$Area, main=names(st)[8], pch=16)
qqline(st$Area, col="red", lwd=2)
hist(st$Area, main=names(st)[8], freq=F,xlab = "Area")
lines(density(st$Area), col="red", lwd=2)
```

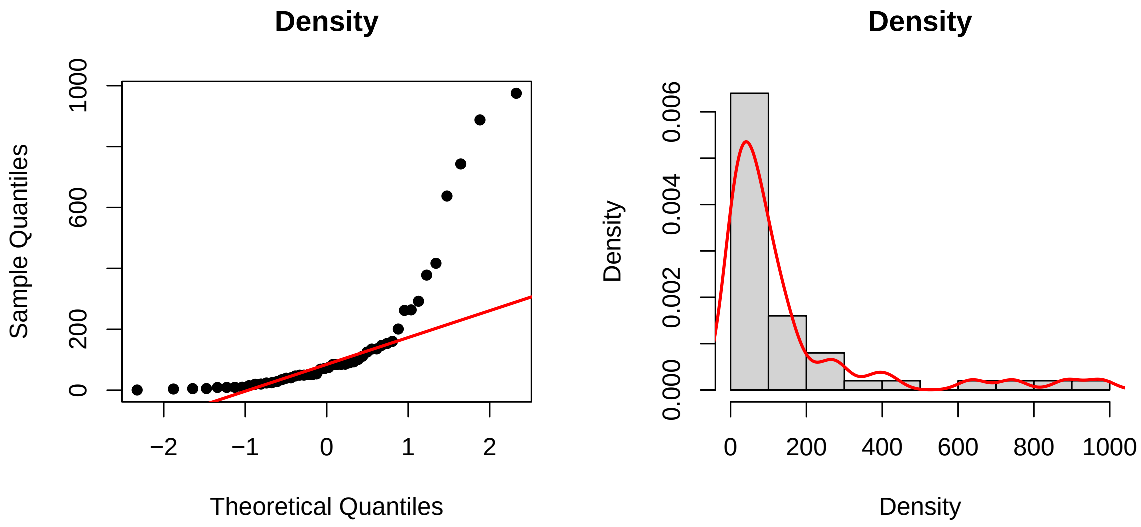


```
shapiro.test(st$Area)
```

```
	Shapiro-Wilk normality test

data:  st$Area
W = 0.57179, p-value = 7.592e-11
```

From the shapiro-test we obtain p-value = 7.592e-11 so we have to reject normality, as we can see from the qqplot and the histogram.

```
par(mfrow=c(1, 2))
qqnorm(st$Density, main=names(st)[9], pch=16)
qqline(st$Density, col="red", lwd=2)
hist(st$Density, main=names(st)[9], freq=F,xlab = "Density")
lines(density(st$Density), col="red", lwd=2)
```



```
shapiro.test(st$Density)
```

```
	Shapiro-Wilk normality test

data:  st$Density
W = 0.63727, p-value = 7.262e-10
```

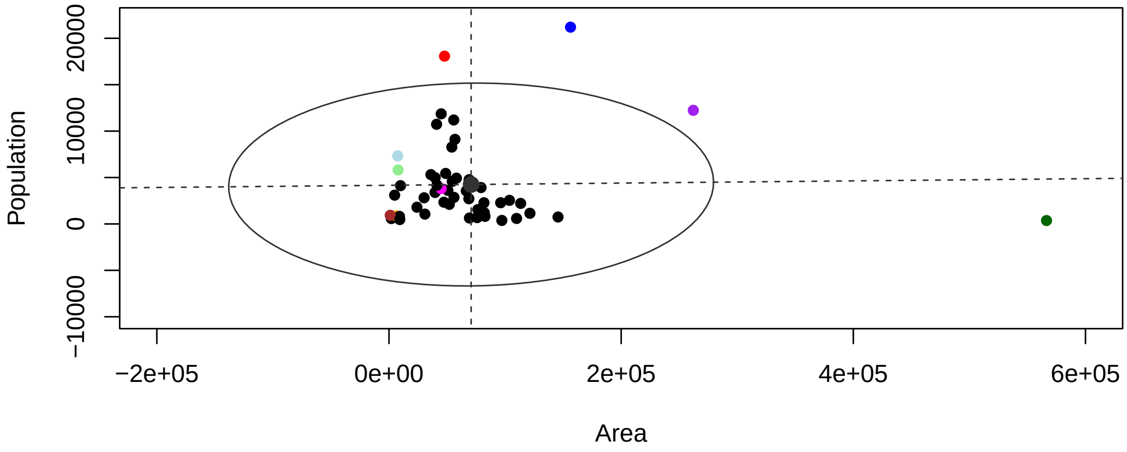
From the shapiro-test we obtain p-value = 7.262e-10 so we have to reject normality, as we can see from the qqplot and the histogram.

Point 5

Make a scatter plot of Area vs Population, colour-coding the outliers found in point 2 with a different colours. Choose among the following colour names. Can they be considered bivariate outliers?

```
S<-cov(st)
S_PA<-cov(st[,c(8,1)])
bar.x <- colMeans(st)
plot(st$Population~st$Area,ylim=c(-10000,22000),xlim=c(-2e+05,6e+05), pch=16,
     col=col.index,xlab="Area",ylab="Population")
lines(ellipse(x=S[c(8,1),c(8,1)], centre=bar.x[c(8,1)], level=0.95), col=grey(0.2))
```

```
eigen.v<-eigen(S[c(8,1),c(8,1)])$vectors
a1<-(eigen.v[1,2]/eigen.v[2,2])*bar.x[8]+bar.x[1]
b1<- -eigen.v[1,2]/eigen.v[2,2]
abline(a1,b1,col=grey(0.2),lty=2)
a2<-(eigen.v[1,1]/eigen.v[2,1])*bar.x[8]+bar.x[1]
b2<- -eigen.v[1,1]/eigen.v[2,1]
abline(a2,b2,col=grey(0.2),lty=2)
points(bar.x[8],bar.x[1],pch=16,cex=1.5,col=grey(0.2))
```



This method to find the bivariate outliers is not correct since we are assuming the normality, but from point 4 we have seen that “Area” and “Population” are not normally distributed and so the joint is not normally distributed.

To solve this problem we consider the Mahalanobis distance and its boxplot.

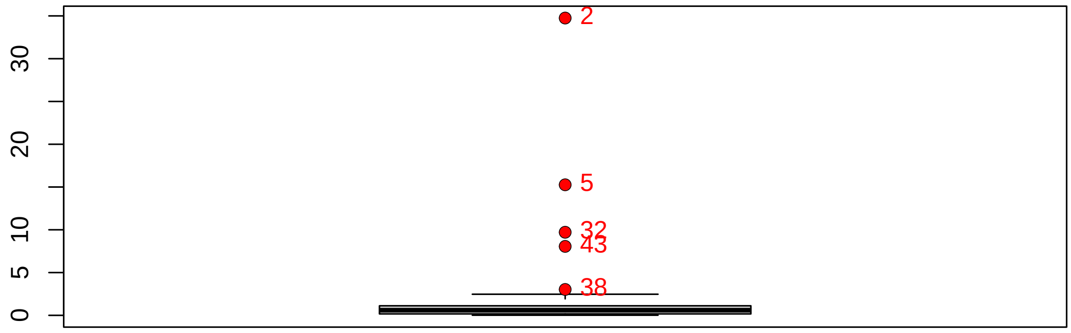
```
d_PA<-mahalanobis(st[,c(8,1)],center=bar.x[c(8,1)],cov=S_PA)
sort(d_PA, decreasing=T)
```

```
      Alaska     California       New York          Texas   Pennsylvania
 34.74977869    15.26000302     9.71012530     8.05801277     3.02423440
    Illinois           Ohio        Montana       Delaware       Michigan
  2.46824520     2.25795288     1.41599878     1.29487732     1.22250691
     Vermont   Rhode Island         Hawaii  New Hampshire     New Jersey
    1.20686781     1.19174465     1.11555181     1.09027468     1.05047047
        Nevada        Florida        Wyoming     New Mexico          Maine
    0.89871320     0.86149890     0.86035030     0.85466294     0.71311942
 Massachusetts   North Dakota    Connecticut   South Dakota          Idaho
    0.67888133     0.65362948     0.65333872     0.64406060     0.61618917
 West Virginia       Maryland           Utah        Arizona       Nebraska
    0.58640446     0.50861704     0.48677222     0.46837255     0.37296959
South Carolina       Colorado         Oregon       Arkansas    Mississippi
    0.32136746     0.30258793     0.28821971     0.27288128     0.25245869
       Indiana         Kansas       Kentucky       Virginia North Carolina
    0.22636030     0.21345760     0.16670367     0.16146283     0.14087078
          Iowa      Tennessee       Oklahoma      Louisiana        Alabama
    0.12399939     0.11885752     0.11789439     0.09990366     0.07363624
       Georgia      Wisconsin     Washington      Minnesota       Missouri
    0.04658613     0.04293609     0.02576607     0.01569832     0.01412732
```

```r
order(d_PA, decreasing = T)
```

```
 [1]  2  5 32 43 38 13 35 26  8 22 45 39 11 29 30 28  9 50 31 19 21 34  7 41 12
[26] 48 20 44  3 27 40  6 37  4 24 14 16 17 46 33 15 42 36 18  1 10 49 47 23 25
```

```r
index2<-sort(d_PA, decreasing = T)[1:5]
index2_names<-as.character(order(d_PA, decreasing = T)[1:5])
c2<-rep(1,5)
boxplot(d_PA)
points(c2,index2,col="red",pch=16)
text(c2,index2,index2_names,pos=4,col = "red")
```



So we can consider as bivariate outliers the observations: 2="Alaska", 5="California", 32="New York", 43="Texas", 38="Pennsylvania".

Point 6

Construct a chi-square Q-Q plot of the squared Mahalanobis distances and comment about multivariate normality.

We show the Q-Q plot of the observed squared Mahalanobis distances, defined as: $(X-\mu)^T\Sigma(X-\mu)$ If data X are multivariate normally distributed, we expect the squared Mahalanobis distance random variable to be distributed as a χ^2_p, and so we'll see the observed distances to be close to the lines of the Q-Q plot.

```
d<-mahalanobis(st, center=bar.x, cov=S)
d
```

```
    Alabama       Alaska      Arizona     Arkansas   California
   8.528180    39.599158     9.361704     6.947483    19.209951
   Colorado  Connecticut     Delaware      Florida      Georgia
   6.013487     9.288110     5.922128     6.323606     4.650861
     Hawaii        Idaho     Illinois      Indiana         Iowa
  23.917476     4.305755     6.466260     1.588988     2.770790
     Kansas     Kentucky    Louisiana        Maine     Maryland
   2.734947     6.030053     9.382094     9.828371     7.292981
```

```
 Massachusetts       Michigan      Minnesota    Mississippi       Missouri
     10.724466       6.035430       4.710730       8.526395       5.690828
       Montana       Nebraska         Nevada  New Hampshire     New Jersey
      4.818944       2.430131      17.433309       4.718866      14.931327
    New Mexico       New York North Carolina   North Dakota           Ohio
     15.051338      11.709485       3.752338      19.260128       3.481939
      Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
      2.385332      10.055405       7.817856      17.495955       8.482161
  South Dakota      Tennessee          Texas           Utah        Vermont
      5.384831       3.427174      15.034672      11.354809       5.634274
      Virginia     Washington  West Virginia      Wisconsin        Wyoming
      3.451621      11.979096       6.117912       3.458684       5.482185
```

```
sort(d, decreasing=T)
```

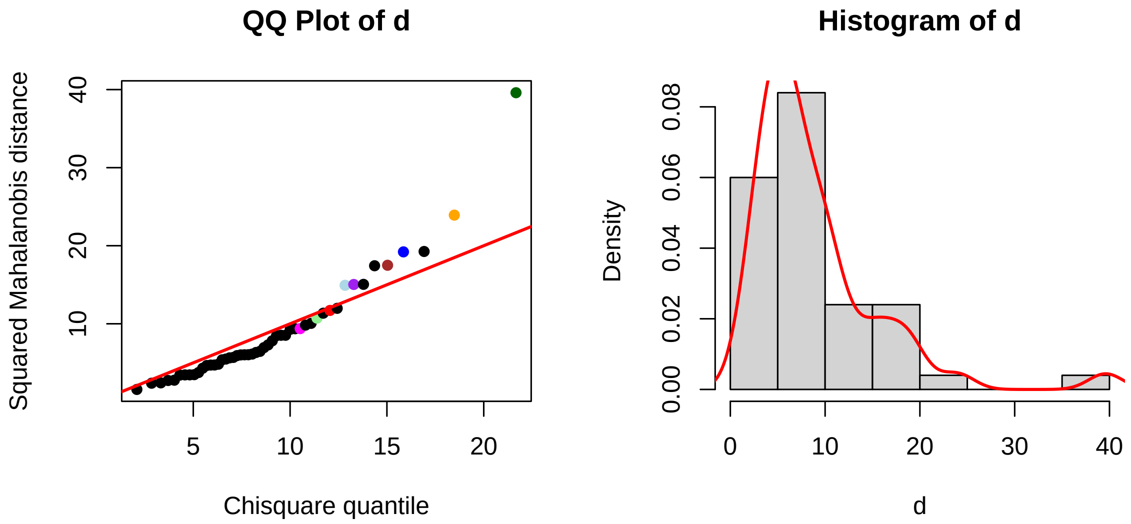
```
        Alaska         Hawaii   North Dakota     California   Rhode Island
     39.599158      23.917476      19.260128      19.209951      17.495955
        Nevada     New Mexico          Texas     New Jersey     Washington
     17.433309      15.051338      15.034672      14.931327      11.979096
      New York           Utah  Massachusetts         Oregon          Maine
     11.709485      11.354809      10.724466      10.055405       9.828371
     Louisiana        Arizona    Connecticut        Alabama    Mississippi
      9.382094       9.361704       9.288110       8.528180       8.526395
South Carolina   Pennsylvania       Maryland       Arkansas       Illinois
      8.482161       7.817856       7.292981       6.947483       6.466260
       Florida  West Virginia       Michigan       Kentucky       Colorado
      6.323606       6.117912       6.035430       6.030053       6.013487
      Delaware       Missouri        Vermont        Wyoming   South Dakota
      5.922128       5.690828       5.634274       5.482185       5.384831
       Montana  New Hampshire      Minnesota        Georgia          Idaho
      4.818944       4.718866       4.710730       4.650861       4.305755
North Carolina           Ohio      Wisconsin       Virginia      Tennessee
      3.752338       3.481939       3.458684       3.451621       3.427174
          Iowa         Kansas       Nebraska       Oklahoma        Indiana
      2.770790       2.734947       2.430131       2.385332       1.588988
```

```
order(d, decreasing = T)
```

```
 [1]  2 11 34  5 39 28 31 43 30 47 32 44 21 37 19 18  3  7  1 24 40 38 20  4 13
[26]  9 48 22 17  6  8 25 45 50 41 26 29 23 10 12 33 35 49 46 42 15 16 27 36 14
```

So the observations with the greatest Mahalanobis distances are 2="Alaska", 11="Hawaii", 34="North Dakota", 5="California".

```
par(mfrow=c(1, 2))
plot(qchisq(ppoints(d), df=p), sort(d), col=col.index[order(d)], pch=16,
     xlab="Chisquare quantile", ylab="Squared Mahalanobis distance")
abline(a=0, b=1, col="red", lwd=2)
title(main ="QQ Plot of d")
hist(d, freq=FALSE)
lines(density(d), col="red", lwd=2)
```



```
ks.test(d,rchisq(n,df=p))
```

```
    Two-sample Kolmogorov-Smirnov test

data:  d and rchisq(n, df = p)
D = 0.24, p-value = 0.1124
alternative hypothesis: two-sided
```

From the Kolmogorov Smirnov test we obtain p-value = 0.002835 so we have to reject normality, as we can see from the qqplot and the histogram. So it is not multivariate normal.

Point 7

Identify multivariate outliers, if any, and compare with the univariate outliers previously found.

```
sort(d, decreasing=T)
```

```
        Alaska         Hawaii   North Dakota     California   Rhode Island 
     39.599158      23.917476      19.260128      19.209951      17.495955 
        Nevada     New Mexico          Texas     New Jersey     Washington 
     17.433309      15.051338      15.034672      14.931327      11.979096 
      New York           Utah  Massachusetts         Oregon          Maine 
     11.709485      11.354809      10.724466      10.055405       9.828371 
     Louisiana        Arizona    Connecticut        Alabama    Mississippi 
      9.382094       9.361704       9.288110       8.528180       8.526395 
South Carolina   Pennsylvania       Maryland       Arkansas       Illinois 
      8.482161       7.817856       7.292981       6.947483       6.466260 
       Florida  West Virginia       Michigan       Kentucky       Colorado 
      6.323606       6.117912       6.035430       6.030053       6.013487 
      Delaware       Missouri        Vermont        Wyoming   South Dakota 
      5.922128       5.690828       5.634274       5.482185       5.384831 
       Montana  New Hampshire      Minnesota        Georgia          Idaho 
      4.818944       4.718866       4.710730       4.650861       4.305755 
North Carolina           Ohio      Wisconsin       Virginia      Tennessee 
      3.752338       3.481939       3.458684       3.451621       3.427174 
          Iowa         Kansas       Nebraska       Oklahoma        Indiana 
      2.770790       2.734947       2.430131       2.385332       1.588988 
```

```
order(d, decreasing = T)
```

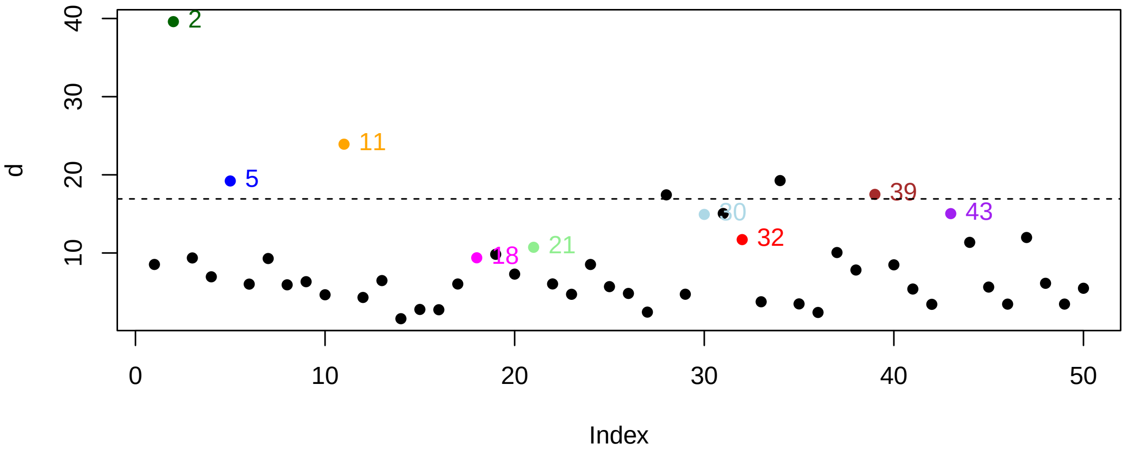
```
 [1]  2 11 34  5 39 28 31 43 30 47 32 44 21 37 19 18  3  7  1 24 40 38 20  4 13
[26]  9 48 22 17  6  8 25 45 50 41 26 29 23 10 12 33 35 49 46 42 15 16 27 36 14
```

The observations with the greatest Mahalanobis distances are 2="Alaska", 5="California", 11="Hawaii", 34="North Dakota" that correspond to univariate outliers found in point 2.

- The univariate outlier 2="Alaska" could be also a multivariate outlier since it has big Mahalanobis distance (>10).
- The univariate outlier 5="California" could be also a multivariate outlier since it has big Mahalanobis distance (>10).

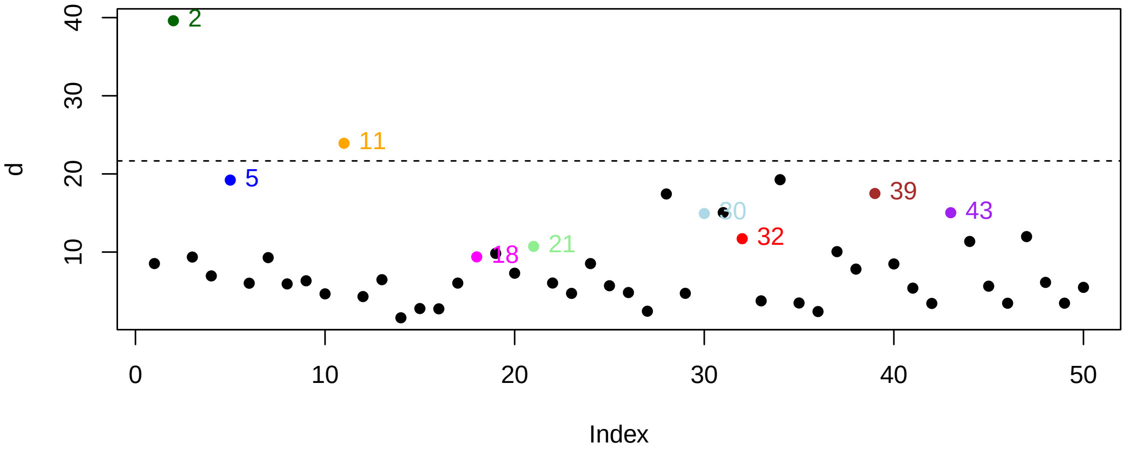
- The univariate outlier 7=”Connecticut” could not be a multivariate outlier since it has small Mahalanobis distance (<10).
- The univariate outlier 11=”Hawaii” could be also a multivariate outlier since it has big Mahalanobis distance (>10).
- The univariate outlier 18=”Louisiana” could not be a multivariate outlier since it has small Mahalanobis distance (<10).
- The univariate outlier 21=”Massachusetts” could be also a multivariate outlier since it has big Mahalanobis distance (>10).
- The univariate outlier 30=”New Jersey” could be also a multivariate outlier since it has big Mahalanobis distance (>10).
- The univariate outlier 32=”New York” could be also a multivariate outlier since it has big Mahalanobis distance (>10).
- The univariate outlier 39=”Rhode Island” could be also a multivariate outlier since it has big Mahalanobis distance (>10).
- The univariate outlier 43=”Texas” could be also a multivariate outlier since it has big Mahalanobis distance (>10).

```r
plot(d, pch=16, col=col.index)
text(index, d[index], labels = ind.names, pos=4, col=col.ind)
abline(h=qchisq(0.95, df=p), lty=2)
```



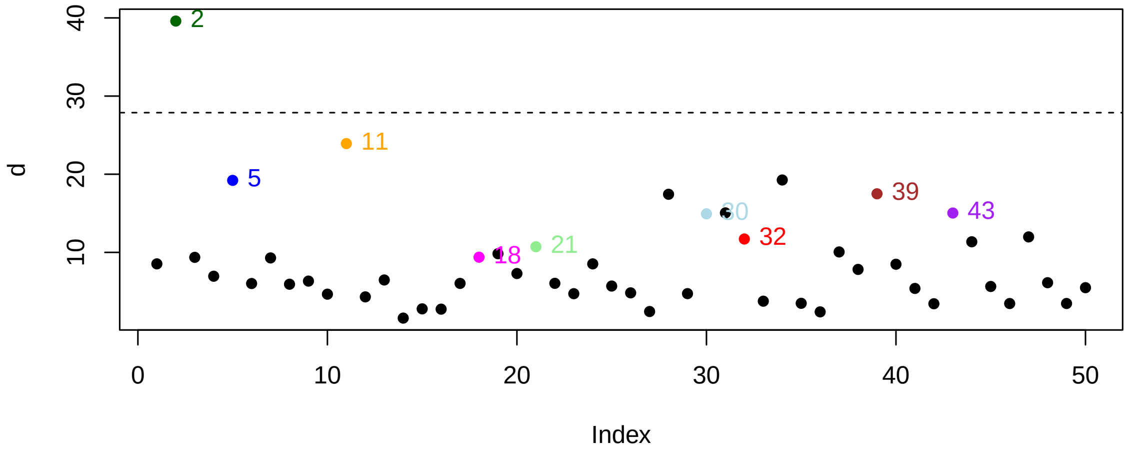
According to the upper quantile 0.95 we obtain that 2="Alaska", 5="California", 11="Hawaii", 39="Rhode Island" could be possible multivariate outliers.

```
plot(d, pch=16, col=col.index)
text(index, d[index], labels = ind.names, pos=4, col=col.ind)
abline(h=qchisq(0.99, df=p), lty=2)
```



According to the upper quantile 0.99 we obtain that 2="Alaska", 11="Hawaii" could be possible multivariate outliers.

```
plot(d, pch=16, col=col.index)
text(index, d[index], labels = ind.names, pos=4, col=col.ind)
abline(h=qchisq(0.999, df=p), lty=2)
```



According to the upper quantile 0.999 we obtain that only 2="Alaska" is over the line, so we are pretty sure that it is a multivariate outlier. We can observe that the most extreme outlier is the observation 2 that is "Alaska". Intuitively is the most different state from the others because of its geographical position.

Exercise 2

Let $Z = (X, Y_1, Y_2) \sim \mathcal{N}_3(\mu, \Sigma)$,

$$\mu = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 1 & -\rho & \rho \\ -\rho & 1 & \rho \\ \rho & \rho & 1 \end{bmatrix}, \quad -1 < \rho < 0.5, \qquad a = \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix}$$

Point 1

Find the inverse of Σ.

We can find the inverse of Σ in two different ways.

Method 1 (use $\Sigma = (1+\rho)I_3 - \rho\, a\, a^T$)

$$\Sigma^{-1} = \left[\begin{array}{c|c|c} v_1 & v_2 & v_3 \end{array} \right], \qquad v_i \in \mathbb{R}^3 \quad i = 1, 2, 3.$$

We know that $\Sigma\Sigma^{-1} = I_3$, so we have the systems $\Sigma v_i = e_i$ where e_i is the $i-th$ vector of the standard basis. We can use $\Sigma = (1+\rho)I_3 - \rho\, a\, a^T$ to solve the three systems as follows

$$\Sigma v_i = e_i \iff ((1+\rho)I_3 - \rho\, a\, a^T)v_i = e_i \iff (1+\rho)v_i - \rho\, a\, (a^T v_i) = e_i \quad (1)$$

we obtain that

$$(2) \qquad v_i = \frac{e_i + \rho a(a^T v_i)}{1+\rho}.$$

We need to find $a^T v_i$: multiply a^T on the right side of (1), the equation becomes

$$a^T((1+\rho)v_i - \rho\, a\, (a^T v_i)) = a^T e_i = a_i$$

then we use the bilinearity property on the left side, do the computations and we conclude that

$$(3) \quad a^T v_i = \frac{a_i}{1-2\rho}.$$

Finally substitute (3) in (2) , then

$$v_i = \frac{(1-2\rho)e_i + a_i \rho a}{-2\rho^2 - \rho + 1} \quad \text{remembering that} \quad a = \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix}$$

therefore

$$\Sigma^{-1} = \frac{1}{-2\rho^2 - \rho + 1} \begin{bmatrix} 1-\rho & \rho & -\rho \\ \rho & 1-\rho & -\rho \\ -\rho & -\rho & 1-\rho \end{bmatrix}.$$

Method 2 We can find the inverse of Σ using the cofactor matrix and the adjugate matrix. First of all we recall these definitions.

- Let A an $n \times n$ matrix with entries from $\mathbb{R}$. The (i,j)-minor of A, denoted $M_{i,j}$, is the determinant of the $(n-1) \times (n-1)$ matrix that results from deleting row i and column j of A.
- The **cofactor matrix** of A is the $n \times n$ matrix C whose (i,j) entry is the (i,j) cofactor of A, which is the (i,j)-minor times a sign factor: $C = ((-1)^{i+j} M_{i,j})_{1 \leq i,j \leq n}$.
- The **adjugate matrix** of A is the transpose of C, that is, the $n \times n$ matrix whose (i,j) entry is the (j,i) cofactor of A, $adj(A) = C^T$.

Important consequence [1] $A^{-1} = \frac{1}{\det(A)} adj(A)$.

Now the determinant of Σ is

$$\det(\Sigma) = \begin{vmatrix} 1 & -\rho & \rho \\ -\rho & 1 & \rho \\ \rho & \rho & 1 \end{vmatrix} = (1+\rho)(-2\rho^2 - \rho + 1)$$

and the adjugate matrix of Σ is

$$adj(\Sigma) = \begin{bmatrix} 1-\rho^2 & \rho+\rho^2 & -\rho-\rho^2 \\ \rho+\rho^2 & 1-\rho^2 & -\rho-\rho^2 \\ -\rho-\rho^2 & -\rho-\rho^2 & 1-\rho^2 \end{bmatrix}.$$

Hence we have that

$$\Sigma^{-1} = \frac{1}{\det(\Sigma)} adj(\Sigma) = \frac{1}{-2\rho^2 - \rho + 1} \begin{bmatrix} 1-\rho & \rho & -\rho \\ \rho & 1-\rho & -\rho \\ -\rho & -\rho & 1-\rho \end{bmatrix}.$$

[1] Strang, Gilbert, Linear Algebra and its Applications, 3rd edition, “Section 4.4: Applications of determinants” , pp. 231-232.

Point 2

Find the eigenvalues of Σ.

We need to find the solutions of the characteristic polynomial of the Covariance Matrix $p_\Sigma(\lambda) = 0$.

$$p_\Sigma(\lambda) = \det(\Sigma - \lambda I_3) = \begin{vmatrix} (1-\lambda) & -\rho & \rho \\ -\rho & (1-\lambda) & \rho \\ \rho & \rho & (1-\lambda) \end{vmatrix} = 0$$

The computation leads to the following solutions: $\lambda_1 = 1 - 2\rho$ and $\lambda_2 = \lambda_3 = 1 + \rho$. In alternative we can find the eigenvalues of Σ and morover the relative eigenvectors without difficult computations. Using $\Sigma = (1+\rho)I_3 - \rho\, a\, a^T$ we can observe that there is a correspondence between the eigenvalues of $a\, a^T$ and the eigenvalues of Σ *i.e.*

$$\Sigma v = \lambda v \implies (1+\rho)v - \rho\, a\, a^T\, v = \lambda v \implies a\, a^T\, v = \frac{-1-\rho+\lambda}{-\rho} v \implies \lambda = -\rho\mu + \rho + 1$$

where μ is eigenvalue of $a\, a^T$. But $a\, a^T\, v = a(a^T\, v) = \mu v$, so immediately an eigenvector is $v = a = (1, 1, -1)$ and the relative eingenvalue is $\mu = ||a||^2 = 3 \implies \lambda_1 = 1 - 2\rho$ by the previous relation. The other eigenvalue is $\mu = 0 \implies \lambda_2 = \lambda_3 = 1 + \rho$.
If $\mu = 0$ then the eigenvector v and a are orthogonal, so the other two eigenvectors can be, for instance $v_1 = (1, -1, 0)$ and $v_2 = (0, 1, 1)$.

Point 3

Let PC1 and PC2 be the first two (population) principal components of Z. Find ρ such that they account for more 80% of total variation of Z.

From the theory, we know that the first two Principal Components are two linear combinations of the original variables $Z = (X, Y_1, Y_2)$ defined as: $PC_1 = a_1^T Z$, $PC_2 = a_2^T Z$.
Where the vector of coefficients are respectively equal to the orthogonal eigenvectors of the Population Covariance Matrix Σ, related to the first two eigenvalues of Σ put in decreasing order, i.e.: $a_i = e_i$ where $\Sigma e_i = \lambda_i e_i$, $i = 1, 2, 3$ and we set $\lambda_1 \geq \lambda_2 \geq \lambda_3$.
In addiction we consider the eigenvectors for the different eigenvalues such that they're orthogonal and with unitary norm, i.e.

$$e_i e_j^T = \begin{cases} 1 & if \quad j = i \\ 0 & if \quad j \neq i \end{cases}$$

In addiction, we know that is required to determine ρ in order to give at least the 80% of the total variance to be explained by PC_1 and PC_2. The proportion of variance is computed

through the ratio $\frac{\lambda_1+...+\lambda_k}{\lambda_1+...+\lambda_p}$ since we know that the variance explained by each $j-th$ component is equal to the $j-th$ eigenvalue λ_j and the total variance of the data is $trace(\Sigma) = {s_1}^2 + ... + {s_p}^2 = \lambda_1 + ... + \lambda_p$. Since in that case $p = 3$, $k = 2$ and $trace(\Sigma) = 3$, we need to impose the following condition: $\frac{\lambda_1+\lambda_2}{3} = 0.8$? So, to find the vectors of coefficients for PC_1 and PC_2 we first need to determine λ_1 and λ_2; in **point 2** we found the eigenvalues: $\lambda_1 = 1 - 2\rho$ and $\lambda_2 = \lambda_3 = 1 + \rho$, where $-1 < \rho < 0.5$.
So we need to distinguish two cases we can face in order to extract correctly the first two eigenvalues:

- $\rho > 0$: In that case we choose the eigenvalues $\lambda_2 = 1+\rho$ and $\lambda_3 = 1+\rho$, so considering the two constraints we put on the values acceptable for ρ and on the minimum percentage of variance we want to explain with the first 2 Principal Components, we find the following range of solutions: $0.2 < \rho < 0.5$
- $\rho < 0$: In that case we choose the eigenvalues $\lambda_1 = 1 - 2\rho$ and $\lambda_2 = 1 + \rho$, so we find the following range of solutions: $-1 < \rho < -0.4$.

Point 4

Find the conditional distribution of (Y_1, Y_2) given $X = x$.

We can use the *result 4.6* from [2]. In this case we have:

$$Z = \begin{bmatrix} X \\ \hline Y_1 \\ Y_2 \end{bmatrix}, \qquad \mu = \begin{bmatrix} 1 \\ \hline 0 \\ 2 \end{bmatrix}, \qquad \Sigma = \left[\begin{array}{c|cc} 1 & -\rho & \rho \\ \hline -\rho & 1 & \rho \\ \rho & \rho & 1 \end{array}\right]$$

Then $((Y_1, Y_2)|X = x)$ has two dimensional normal distribution with

$$\text{Mean} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} + \begin{bmatrix} -\rho \\ \rho \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} x - 1 \end{bmatrix} = \begin{bmatrix} -\rho(x-1) \\ 2\rho(x-1) \end{bmatrix}$$

and

$$\text{Covariance} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} - \begin{bmatrix} -\rho \\ \rho \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} -\rho & \rho \end{bmatrix} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} - \begin{bmatrix} \rho^2 & -\rho^2 \\ -\rho^2 & \rho^2 \end{bmatrix} = \begin{bmatrix} 1-\rho^2 & \rho+\rho^2 \\ \rho+\rho^2 & 1-\rho^2 \end{bmatrix}$$

So $Y \sim \mathcal{N}_2(\mu_y, \Sigma_y)$ where

$$\mu_y = \begin{bmatrix} -\rho(x-1) \\ 2\rho(x-1) \end{bmatrix} \qquad \Sigma_y = \begin{bmatrix} 1-\rho^2 & \rho+\rho^2 \\ \rho+\rho^2 & 1-\rho^2 \end{bmatrix}$$

[2] Johnson, Wichern. Applied Multivariate Statistical Analysis, 6th ed. , Pearson, 2007, page 160.

Point 5

Let $\rho = 0.2$ and Σ_y and μ_y be the corresponding covariance matrix and the mean vector of the distribution of $Y = (Y_1, Y_2)$ given $X = 0$. Sketch the ellipse $(y - \mu_y)^T \Sigma_y^{-1} (y - \mu_y) = c^2$, in the 2 dimensional space $y = (y_1, y_2)$ by setting the constant c such that the ellipse contains 0.95 probability with respect to the conditional distribution of Y.

When $\rho = 0.2$, the Population Mean Vector and the Population Covariance Matrix are:

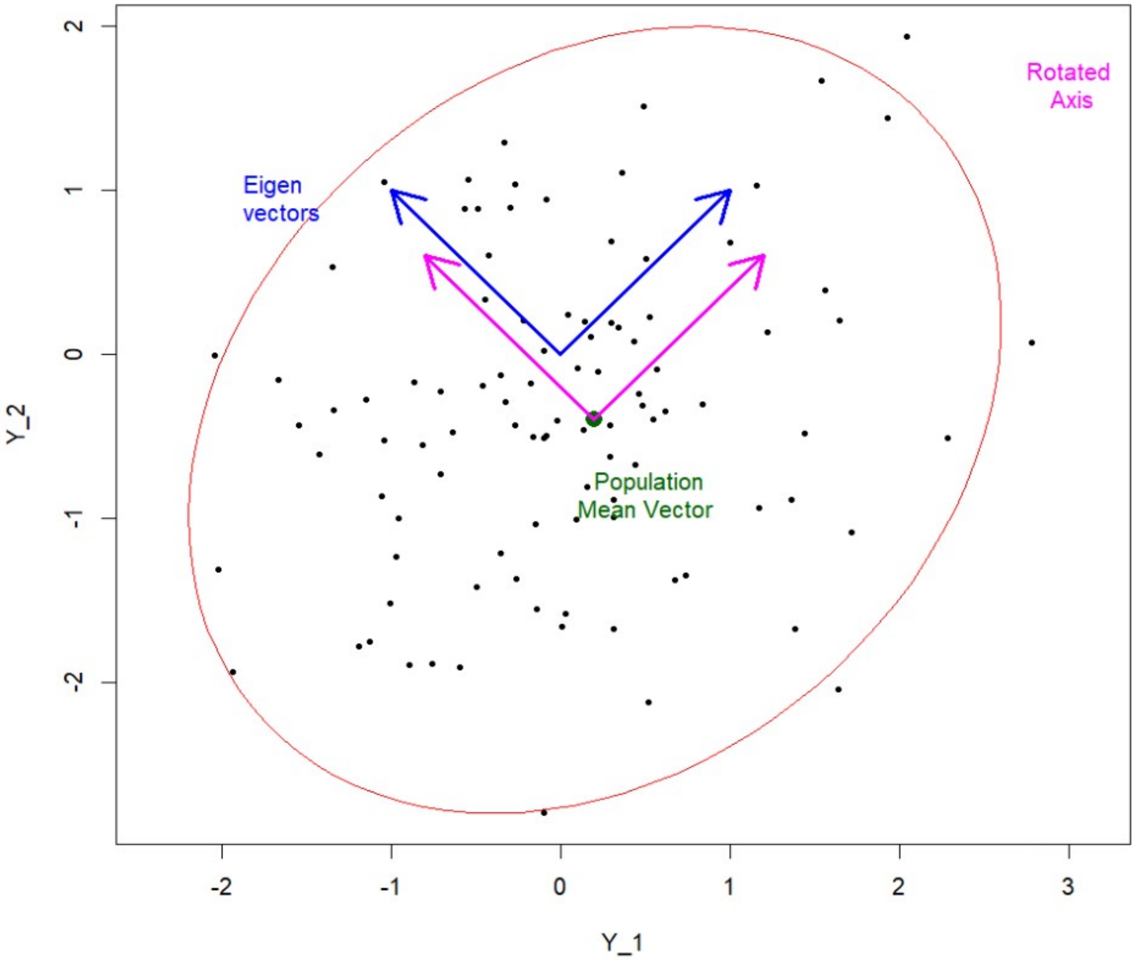
$$\mu_y = \begin{bmatrix} 0.2 \\ -0.4 \end{bmatrix} \qquad \Sigma_y = \begin{bmatrix} 0.96 & 0.24 \\ 0.24 & 0.96 \end{bmatrix}$$

If we consider the equation of the ellipse $(y - \mu_y)^T \Sigma_y^{-1} (y - \mu_y) = c^2$ we have that the constant c^2 we need in this case is Chi-squared quantile of order $0.95 \implies \alpha = 0.05$ with two degrees of freedom, which is 5.991. If we compute eigenvalues and eigenvectors of Σ_y^{-1} we get that the axes of the ellipse are in the direction of the eigenvectors and are proportional to the inverse of the square root of the eigenvalues.
In fact

$$\Sigma_y^{-1} = \begin{bmatrix} \frac{10}{9} & -\frac{5}{18} \\ -\frac{5}{18} & \frac{10}{9} \end{bmatrix}$$

Compute the eigenvalues solving $p_{\Sigma_y^{-1}}(\lambda) = 0$, the solutions are $\lambda_1 = \frac{5}{6}$ and $\lambda_2 = \frac{25}{18}$.
Compute the eigenvectors solving the system $\Sigma_y^{-1} e_i = \lambda_i e_i$ for $i = 1, 2$. The solution are $e_1 = (1, 1)$ and $e_2 = (-1, 1)$.
So the ellipse centered in the point $(0.2, -0.4)$ has one semiaxes in the direction of the vector $(1, 1)$ with lenght equal to $c\sqrt{1/\lambda_1} \approx 2,68$, and the other semiaxes in the direction of the vector $(-1, 1)$ with lenght equal to $c\sqrt{1/\lambda_2} \approx 2,08$.

Now we can plot the ellipse simulating a sample size 100 from the distribution of Y. As we can see in the next page, the ellipse contains around 95% of the points generated.


Y_2
2
1
0
-1
-2
Y_1
-2
-1
0
1
2
3
Eigen vectors
Rotated Axis
Population Mean Vector

Exercise 3

Nutritional data from 961 different food items.

```
nutritional<-read.table("data/nutritional.txt")
head(nutritional)
```

```
  fat food.energy carbohydrates protein cholesterol weight saturated.fat
1   2          25             2       0           2  15.00           0.2
2   6          60             2       0           4  16.00           1.0
3   1          90            22       4           0  28.35           0.1
4   0          90            22       3           0  28.35           0.1
5   0          10             1       1           0  33.00           0.0
6   1          70            21       4           0  28.35           0.1
```

For each food item, there are 7 variables: fat (grams), food.energy (calories), carbohydrates (grams), protein (grams), cholesterol (milligrams), weight (grams), and saturated.fat (grams)

Point 1

To equalize out the different types of servings of each food, first divide each variable by weight of the food item (which leaves us with 6 variables). Next, because of the wide variations in the different variables, standardize each variable. Perform Principal Component Analysis on the transformed data.

To equalize out the different types of servings of each food, we firstly divide each variable by the weight of the food item (which leaves us with 6 variables).

```
nutritional<- (nutritional/nutritional$weight)[,-6]
nutritional<-round(nutritional,3)
head(nutritional)
```

```
    fat food.energy carbohydrates protein cholesterol saturated.fat
1 0.133       1.667         0.133   0.000       0.133         0.013
2 0.375       3.750         0.125   0.000       0.250         0.062
3 0.035       3.175         0.776   0.141       0.000         0.004
4 0.000       3.175         0.776   0.106       0.000         0.004
5 0.000       0.303         0.030   0.030       0.000         0.000
6 0.035       2.469         0.741   0.141       0.000         0.004
```

Next, because of the wide variations in the different variables, we standardize each variable by them standard deviations

```
dt<-scale(nutritional, scale = T)
head(dt)
```

```
         fat food.energy carbohydrates    protein  cholesterol saturated.fat
1  0.1020647  -0.3028844    -0.4208835 -0.7779781 -0.182054965    -0.3657588
2  1.3528609   0.7732329    -0.4529343 -0.7779781 -0.008844096     0.3754391
3 -0.4044561   0.4761770     2.1551993  0.7898146 -0.378952791    -0.5018971
4 -0.5853564   0.4761770     2.1551993  0.4006462 -0.378952791    -0.5018971
5 -0.5853564  -1.0075526    -0.8335375 -0.4444052 -0.378952791    -0.5624031
6 -0.4044561   0.1114440     2.0149771  0.7898146 -0.378952791    -0.5018971
```

Now we can perform the Principal Component Analysis on the standardized data. Firstly, we plot the rotation matrix which contains a set of vectors that give the rotations of the principal component axes. Those vectors are the eigenvectors of the sample covariance matrix S of our standardized data.

```
food.pca<-prcomp(dt)
food.pca$rotation
```

```
                      PC1         PC2        PC3        PC4        PC5
fat           -0.55723573  0.09854969 -0.2751695  0.1302465 -0.4546910
food.energy   -0.53615250  0.35680300  0.1369426  0.0743439 -0.2731028
carbohydrates  0.02457052  0.67180502  0.5682289 -0.2862444  0.1568387
protein       -0.23529576 -0.37348488  0.6390256  0.5991772  0.1536964
cholesterol   -0.25244242 -0.52134113  0.3257728 -0.7170860 -0.2100662
saturated.fat -0.53135098 -0.01923702 -0.2610800 -0.1494902  0.7914052
                       PC6
fat            0.616725222
food.energy   -0.697399944
carbohydrates  0.344464730
protein        0.118976191
cholesterol   -0.002864321
saturated.fat  0.021536719
```

Then, we show the ordered standard deviations of each Principal Component, which corresponds to the square roots of the eigenvalues of S.

```
food.pca$sdev
```

```
[1] 1.6274675 1.1532237 1.0100762 0.8246636 0.5162718 0.2335895
```

Point 2

Decide how many components to retain in order to achieve a satisfactory lower-dimensional representation of the data. Justify your answer.

In order to achieve a satisfactory lower-dimensional representation of the data, it is possible to select only the most relevant PCs. To achieve this goal, we decide to base our selection process looking at the "Proportion of Variance" explained by the first k PCs. This is the ratio between the sum of the first k sample variances and the total variance of the original variables $\left[\frac{s_{z_1}+...+s_{z_k}}{s_1+...+s_p}\right]$ or equivalently the ratio between the sum of the first k eigenvalues of S and the trace of S $\left[\frac{\lambda_1+...+\lambda_k}{trace(S)}\right]$.

```
food.pca<-prcomp(dt)
summary(food.pca)
```

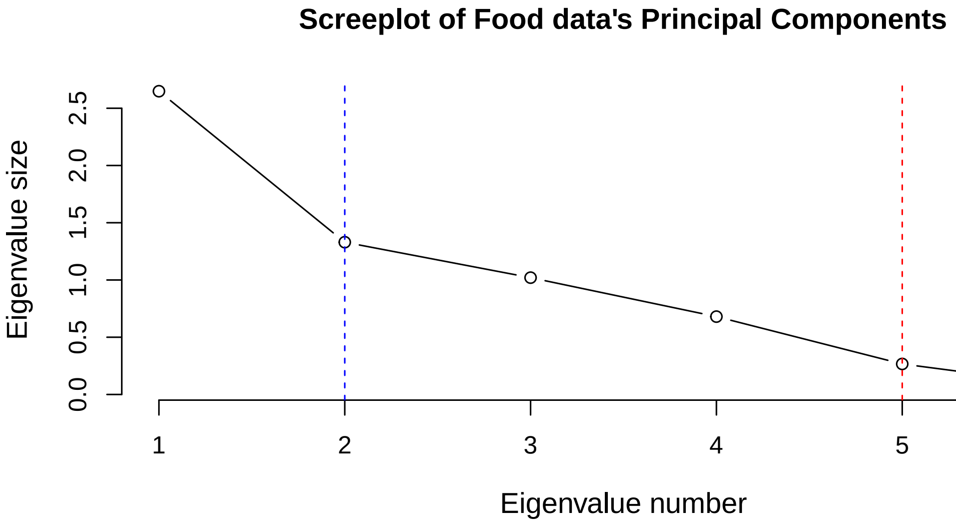
```
Importance of components:
                          PC1    PC2    PC3    PC4     PC5     PC6
Standard deviation     1.6275 1.1532 1.0101 0.8247 0.51627 0.23359
Proportion of Variance 0.4414 0.2217 0.1700 0.1134 0.04442 0.00909
Cumulative Proportion  0.4414 0.6631 0.8331 0.9465 0.99091 1.00000
```

We have decided to set the threshold value of the "Cumulative Proportion" at 80%. Since it exceeds the 80% at the 3-rd PC, we choose to retain only the first three PCs.

Another way to reach this goal is looking at the "Screeplot". This last is a graphical tool that shows the relation between the PCs' indexes (Eigenvalue numbers) and the eigenvalues' sizes. The qualitative selection procedure consists in looking at the position of an "elbow" (bend) in the curve, and associate to it an index $t + 1$, which represents the t+1-th eigenvalue. Then we retain only the first t components.

```
k<- dim(food.pca$rotation)[2]
plot(1:k,food.pca$sdev^2,type="b",axes=F,xlab="",ylab="")
title(xlab=list("Eigenvalue number",cex=1.2),
      ylab=list("Eigenvalue size",cex=1.2),
      main=list("Screeplot of Food data's Principal Components",cex=1.2))
```

```
axis(1,at=1:k,labels=1:k,tick=TRUE)
axis(2, tick=TRUE)
abline(v=2,lty=2,col="blue")
abline(v=5,lty=2,col="red")
```



This plot shows a fundamental problem of this approach, which is its subjectivity due its qualitativeness. In fact, in this case two "elbows" are observed. This makes the selection process imprecise.

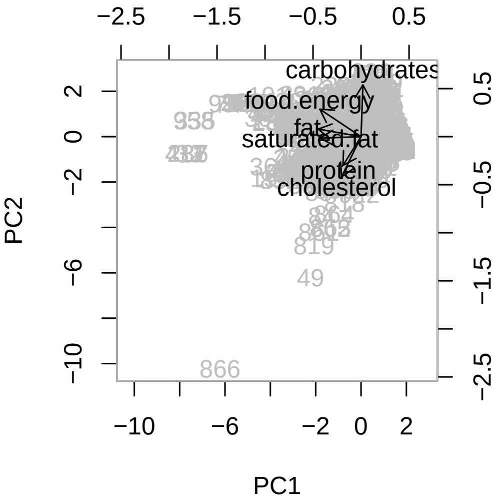
For this reason we have decided to base our decision looking at the "Proportion of variance", and so to retain only the first three Principal Components.

Point 3

Give an interpretation to the first two principal components.

To give an interpretation to the first two Principal Components we decide plot the "biplot" of the PCs to have a a clearer view.

```
biplot(food.pca, scale=0, col=c("grey","black"))
```



It is possible to check that there is an inverse relationship between the increment of fat and saturated fat (per gram of the observed foods) and the growth of the first Principal Component (PC_1). Moreover, there is an extremely high positive correlation between the carbohydrates' ratio and the second Principal Component (PC_2), which is negatively correlated with proteins and cholesterol's ratios in the foods. Also, it is possible to see that the number of calories per gram of foods is positively correlated with PC_2 and negatively with PC_1.

This is coherent with the heuristic that high calorie foods shares higher percentages of carbohydrates (simple and complex sugars) and fats, which is confirmed by the positive correlation of "food.energy" with all the three variables: "carbohydrates", "fat" and "saturated.fat".

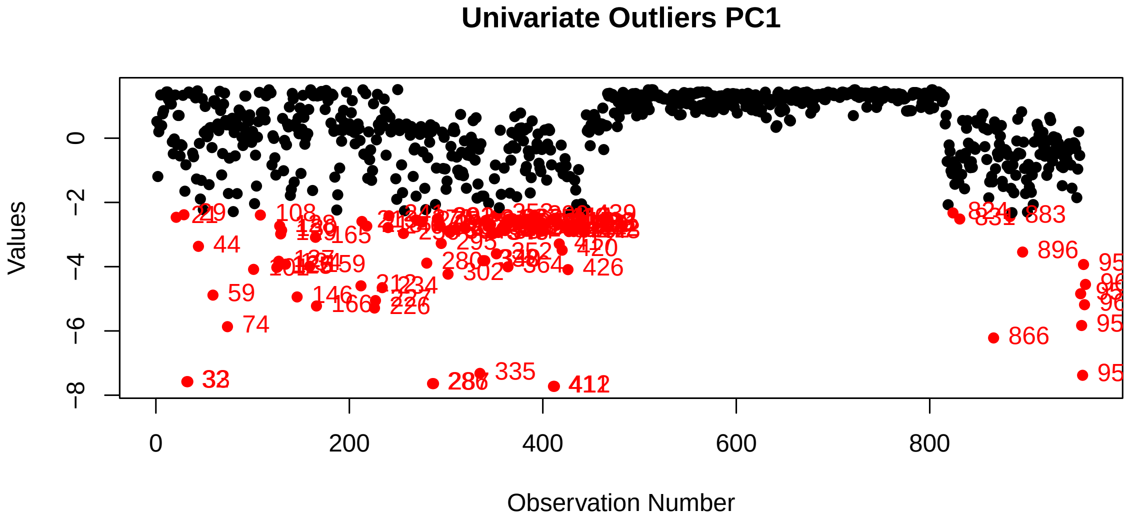
Observing the plot, we believe that the PC_1 could be interpreted as a variable that express the concentration of fat inside a given food, since it is negatively strong correlated with the variables "fat" and "saturated.fat". Whereas, the PC_2 could express the "grade of animal origin" of the food. This because it is positively high correlated with the concentration of carbohydrates and negatively strong correlated with the presence of proteins and cholesterol, which is a lipid that has an essential structural component in animal cell membranes.

Point 4

Identify univariate outliers with respect to the first three principal components, up to 3 per component. These points correspond to foods that are very high or very low in what variable (up to 2 variables per observation)?

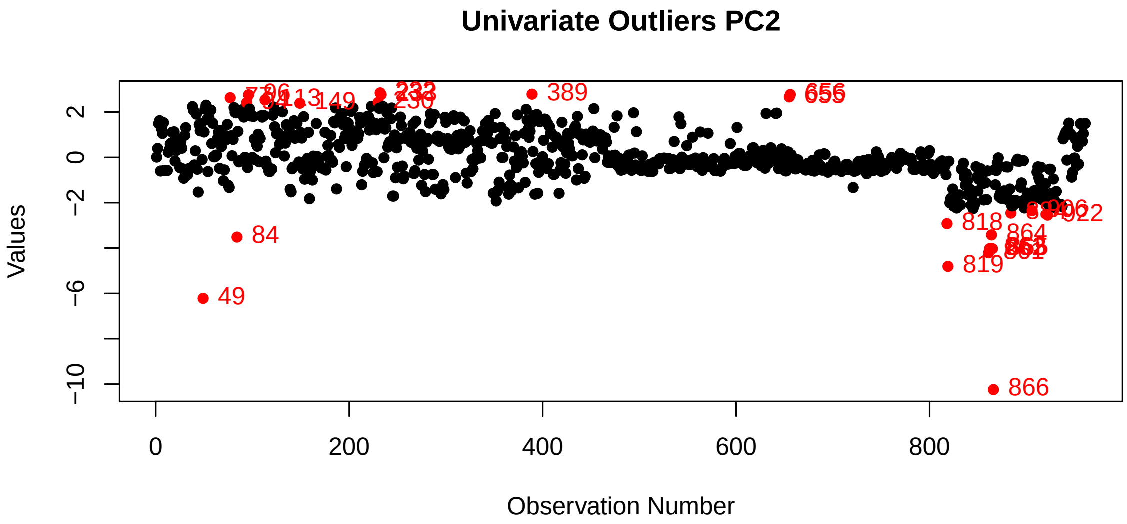
To identify the univariate outliers with respect to the first three principal components, firstly we plot the absolute values of the observations associated to all three of them, and we color in red the ones that are located over the 99-th percentile of a Standard Gaussian distribution. This to understand the approximate amount of univariate outliers.

```
w<-which(abs(food.pca$x)>qnorm(0.99), arr.ind=T)

filter1<-w[which(w[,2]==1)]
filter2<-w[which(w[,2]==2)]
filter3<-w[which(w[,2]==3)]

n=length(food.pca$x[,1])

col.ind<- rep("black",n)
col.ind[filter1]<- "red"
plot(food.pca$x[,1], pch=16,  col=col.ind, xlab = "Observation Number",
     ylab="Values", main = "Univariate Outliers PC1")
text(filter1,food.pca$x[filter1,1], labels= as.character(filter1), pos=4, col="red")
```



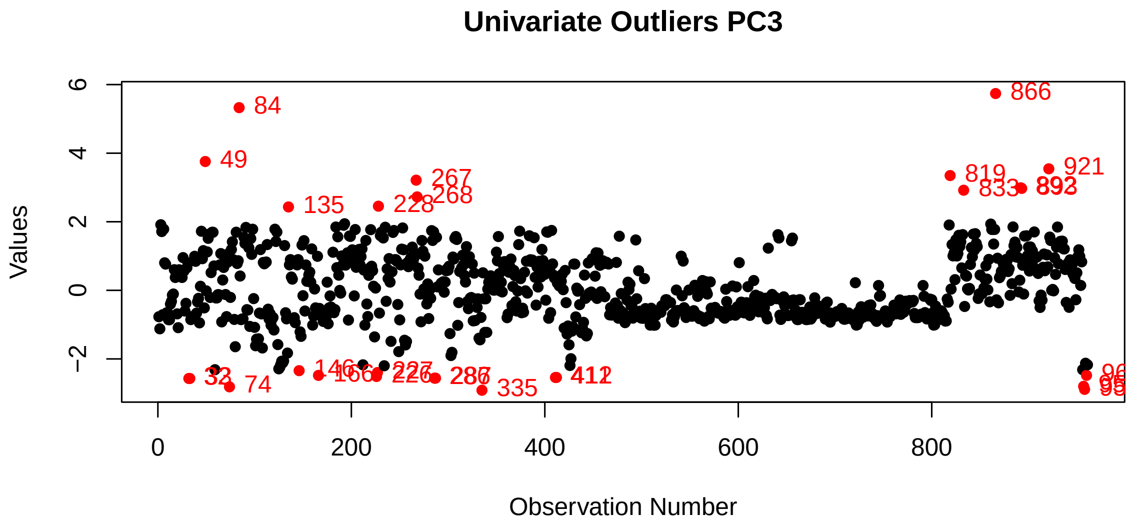
Despite the restrictive threshold to select the univariate outliers, it is already clear from the first plot a consistent presence of them. For this reason we decide to look at the chart to select the most 3 extreme observations, which are the numbers 286,411 and 412 with respect to the 1-st Principal Component. Now we plot the observations associated to the PC_2.

```
col.ind<- rep("black",n)
col.ind[filter2]<- "red"
plot(food.pca$x[,2], pch=16,  col=col.ind, xlab = "Observation Number",
     ylab="Values", main = "Univariate Outliers PC2")
text(filter2,food.pca$x[filter2,2], labels= as.character(filter2), pos=4, col="red")
```



Because of the same procedure, we select as the three univariate outliers with respect to the 2-nd Principal Component the observations 49,819 and 866. Plotting those related to the PC_3 we have:

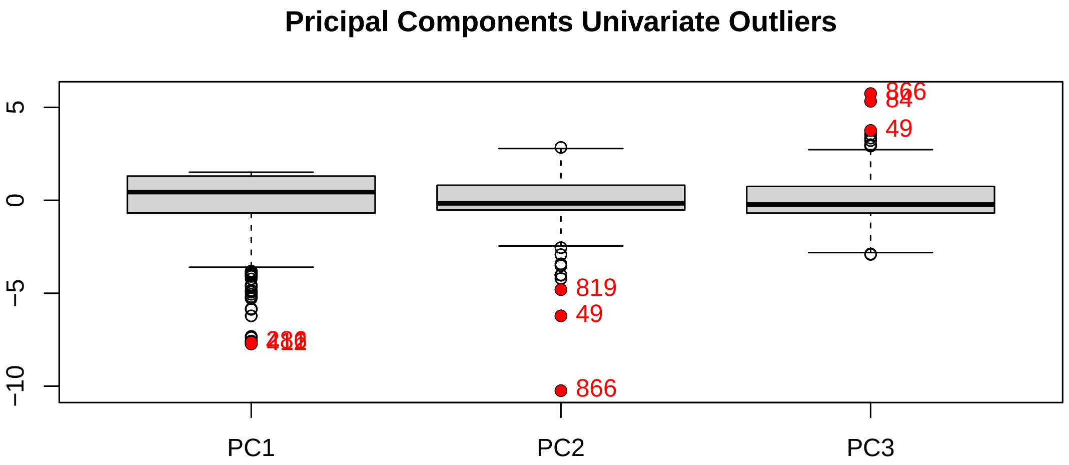
```
col.ind<- rep("black",n)
col.ind[filter3]<- "red"
plot(food.pca$x[,3], pch=16,  col=col.ind, xlab = "Observation Number",
     ylab="Values", main = "Univariate Outliers PC3")
text(filter3,food.pca$x[filter3,3], labels= as.character(filter3), pos=4, col="red")
```



Instead, here we select the observations 49, 84 and 866.

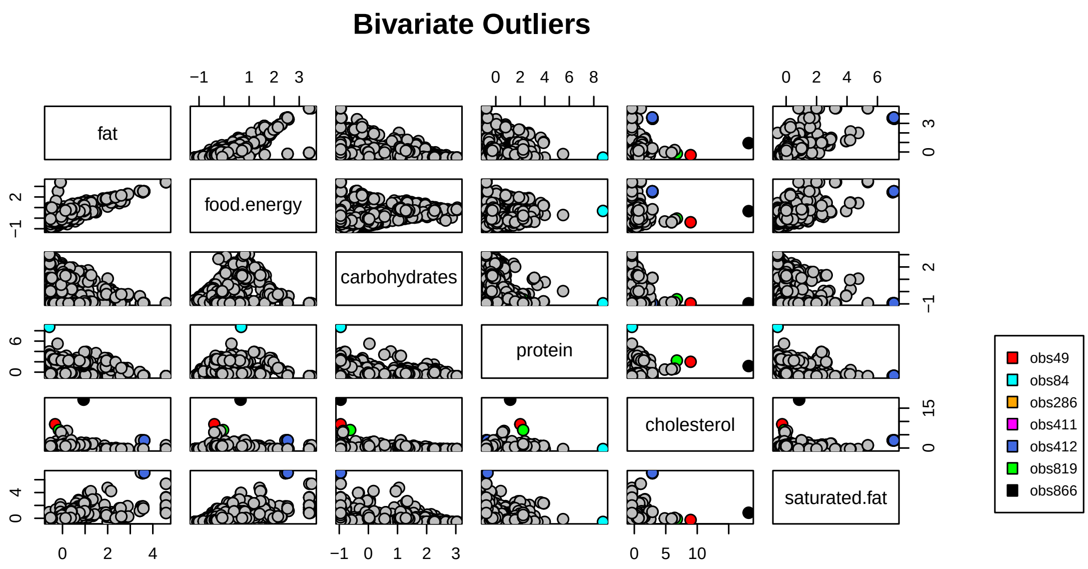
To check the appropriateness of the selection we plot the boxplots of the first three Principal Components with the univariate outliers highlighted in red.

```
boxplot(food.pca$x[,-4:-6], main="Pricipal Components Univariate Outliers")

points(rep(1, 3), food.pca$x[c(286,411,412),1], pch=16, col="red")
points(rep(2, 3), food.pca$x[c(49,819,866),2], pch=16, col="red")
points(rep(3, 3), food.pca$x[c(49,84,866),3], pch=16, col="red")

text(rep(1, 3),food.pca$x[c(286,411,412),1],as.character(c(286,411,412)),pos=4,
     col="red")
text(rep(2, 3), food.pca$x[c(49,819,866),2],as.character(c(49,819,866)),pos=4,
     col="red")
text(rep(3, 3), food.pca$x[c(49,84,866),3],as.character(c(49,84,866)),pos=4,
     col="red")
```



Now for each extreme observation (food) selected, we want to understand with respect to which variables they are considered outliers. To do this we start making a “scatter plot” that shows the “bivariate outliers” with respect to the original food’s variables.

```r
outliers<- c(49,84,286,411,412,819,866)

index<-rep("grey",n)
index[outliers[1]]<- "red"
index[outliers[2]]<-"cyan"
index[outliers[3]]<- "orange"
index[outliers[4]]<- "magenta1"
index[outliers[5]]<- "royalblue"
index[outliers[6]]<- "green"
index[outliers[7]]<- "black"

ott<-c("obs49", "obs84", "obs286", "obs411", "obs412", "obs819", "obs866")
pairs(dt,pch=21, bg=index, cex=1.5,oma=c(3,3,5,13),  main="Bivariate Outliers")
par(xpd=TRUE)
legend("bottomright", legend = ott, fill = c("red", "cyan", "orange", "magenta1",
                                             "royalblue", "green", "black"), cex=0.55)
```



It is possible to check that both observations 286 and 411 seems missing. We checked that this does not depends by the presence of an error in the code, or in the compilation. In fact, checking the composition of the “index” object, it is possible to verify the presence of the values “orange” and “magenta1”, associated to the twin of outliers we are talking of. Probably, this point out the presence of an overlap between the observations 286 and 411 with at least one other observation that we have not selected before. This because we had to pick only three outliers per variable. This assertion is probably confirmed by the previous charts in which, for example, the observations 411 and 412 are overlapped. As additional support, we will use the 3-D scatter plot in point 3.5 .

Because of this problem, we create a table that helps us to understand with respect to which original variables they are considered extreme values, always using the threshold of the 99-th percentile. The index associated to the original variables is ordered with respect to the columns of the original table: 1 = “fat”, 2 = “food.energy”, 3 = “carbohydrates”, 4 = “protein”, 5 = “cholesterol”, 6 = “saturated.fat”.

```
ww<-which(abs(dt)>qnorm(0.99), arr.ind=T)
frame<- list(ww[ww[,1]==49],ww[ww[,1]==84], ww[ww[,1]==286], ww[ww[,1]==411],
             ww[ww[,1]==412], ww[ww[,1]==819],ww[ww[,1]==866] )
frame
```

```
[[1]]
[1] 49  5
```

```
[[2]]
[1] 84  4

[[3]]
[1] 286 286 286 286   1   2   5   6

[[4]]
[1] 411 411 411 411   1   2   5   6

[[5]]
[1] 412 412 412 412   1   2   5   6

[[6]]
[1] 819   5

[[7]]
[1] 866   5
```

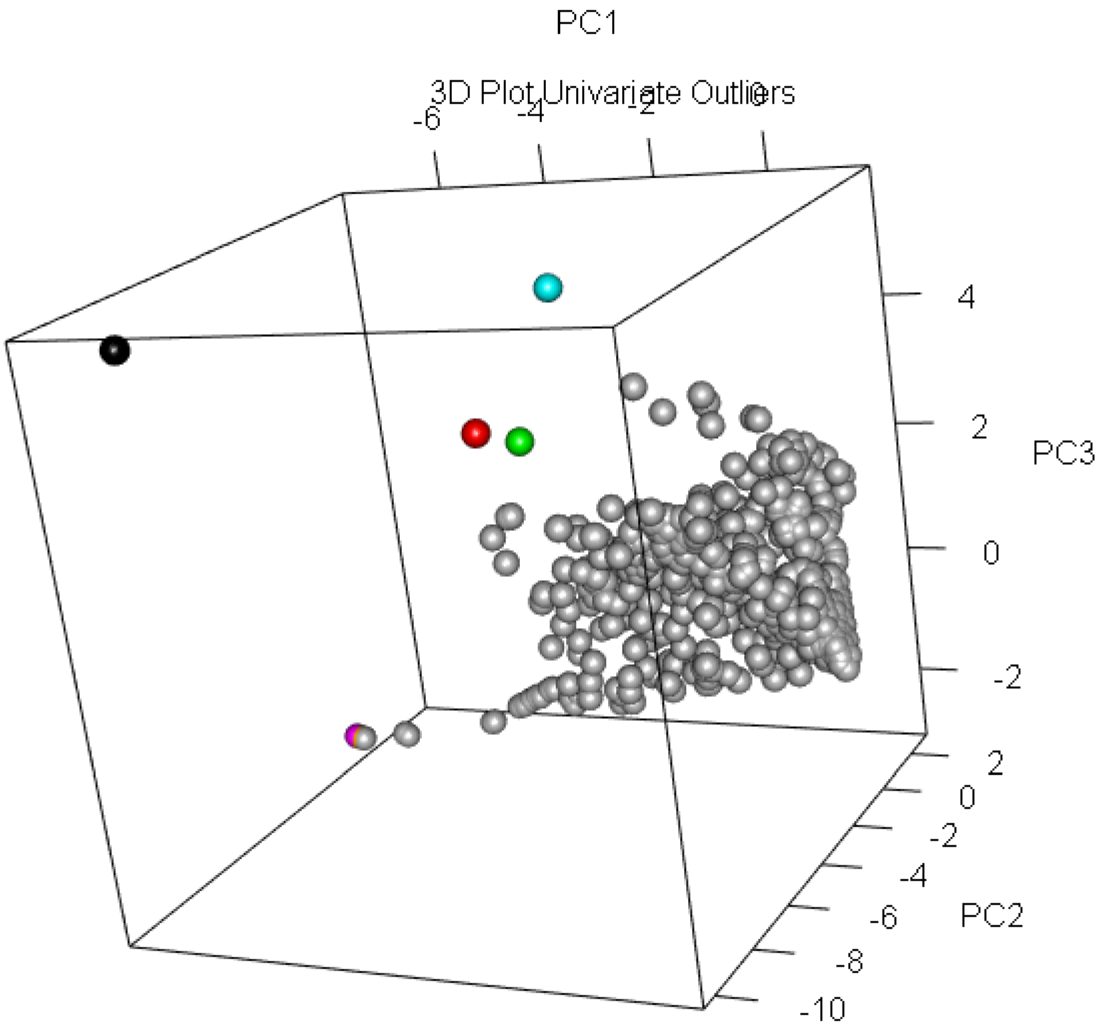
To make clearer the interpretation, for example, the 3-rd block of the list, means that the observed food 286 shares extreme values with respect to four original variables: "fat", "food.energy", "cholesterol" and "saturated.fat" (1,2,5,6 respectively)

Point 5

Make a 3-d scatter plot with the first three principal components, while color coding these outliers.

Now we make a 3-d scatter plot with the first three principal components, coloring the previously selected outliers.

```
pca3<- food.pca$x[,1:3]

plot3d(
  x=pca3[,1], y=pca3[,2], z=pca3[,3],
  col = index,
  type = 's',
  radius = .25,
  main="3D Plot Univariate Outliers",
  xlab="PC1", ylab="PC2", zlab="PC3")
```



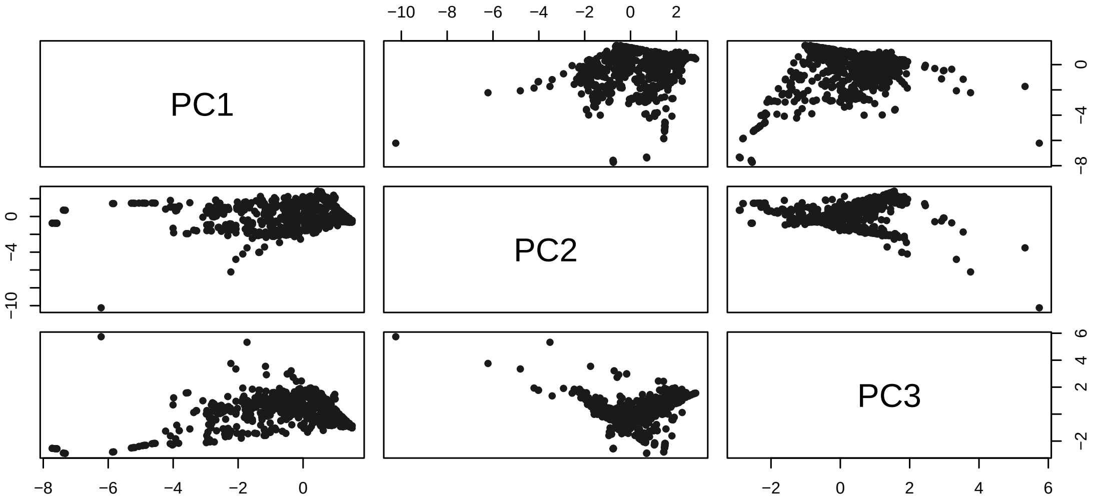
Recovering the comment above, it is possible to observe how the outliers 286 and 411 overlap each other with an additional observation that is not classified as an outlier, which is coherent with the hypothesis given before.

Point 6

Investigate multivariate normality through the first three principal components.

To investigate the multivariate normality through the first three principal components, we have firstly decided to check the normality of the marginal variables (PC_1, PC_2, PC_3), since a necessary condition of the joint multivariate normality is that each marginal distribution is also Gaussian. Firstly, we produce a scatter plot to have a qualitative impression of the spatial disposition of the bivariate data, searching for non-spread ellipsoidal shapes.

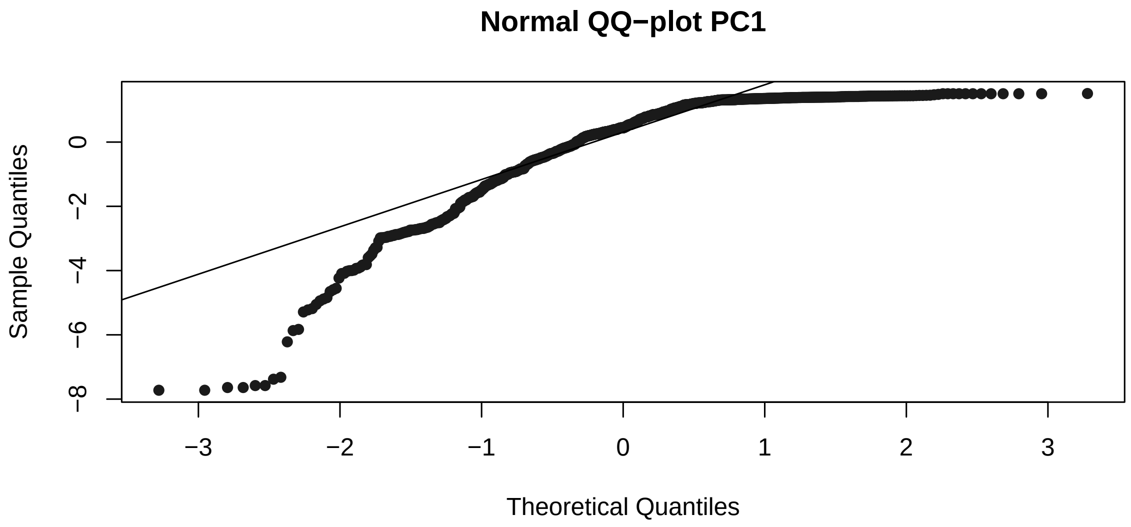
```
pairs(food.pca$x[,1:3], pch=16, col=gray(0.1))
```



This gives us a first clue of the presence of non-normally distributed data.

Trying to test this initial hypothesis, we decide to perform the Q-Q plot for each PC, which is a common graphical technique to evaluate the normality of marginal distributions of the sample observations. In fact, it plots the relationship between the sample quantile and the theoretical quantile expected in the presence of normally distributed observations (data). If there is a consistent number of observations above or below the straight line, it means that the distribution of the data is not shaped as a Gaussian one. To be as sure as possible, we also perform for each pricipal component the "Shapiro-Wilk normality test" with an alpha level of 0.05 ($\alpha = 0.05$)

```
qqnorm(food.pca$x[,1],pch=16, col=gray(0.1),main="Normal QQ-plot PC1")
qqline(food.pca$x[,1])
```



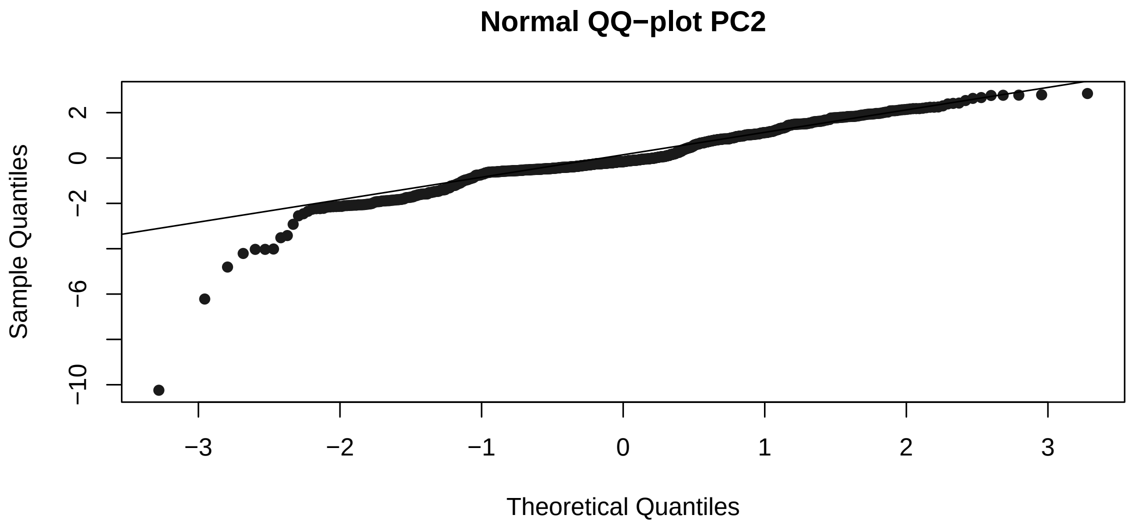
```
shapiro.test(food.pca$x[,1])
```

```
    Shapiro-Wilk normality test

data:  food.pca$x[, 1]
W = 0.8133, p-value < 2.2e-16
```

Since $p - value << \alpha = 0.05$ and by the shape of the Q-Q plot, which identify a distribution with very heavy tails, we can reject the null hypothesis of normality of the variable PC_1.

We do the same for PC_2.

```
qqnorm(food.pca$x[,2],pch=16, col=gray(0.1),main="Normal QQ-plot PC2")
qqline(food.pca$x[,2])
```



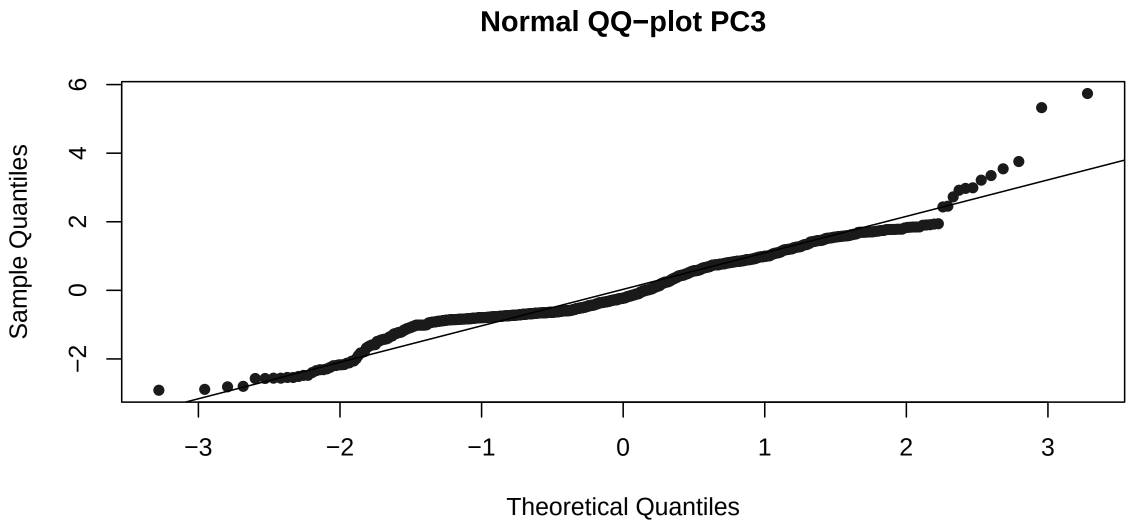
```
shapiro.test(food.pca$x[,2])
```

```
    Shapiro-Wilk normality test

data:  food.pca$x[, 2]
W = 0.93186, p-value < 2.2e-16
```

Also in this case the Q-Q plot shows a non-normality configuration, but less obvious than before. For this reason, we look at the test which tells us that is possible to reject the null hypothesis of normality with respect to the marginal of PC_2, since $p - value << \alpha = 0.05$.

Plotting for PC_3 we have:

```
qqnorm(food.pca$x[,3],pch=16, col=gray(0.1),main="Normal QQ-plot PC3")
qqline(food.pca$x[,3])
```



```
shapiro.test(food.pca$x[,3])
```

```
	Shapiro-Wilk normality test

data:  food.pca$x[, 3]
W = 0.95587, p-value < 2.2e-16
```

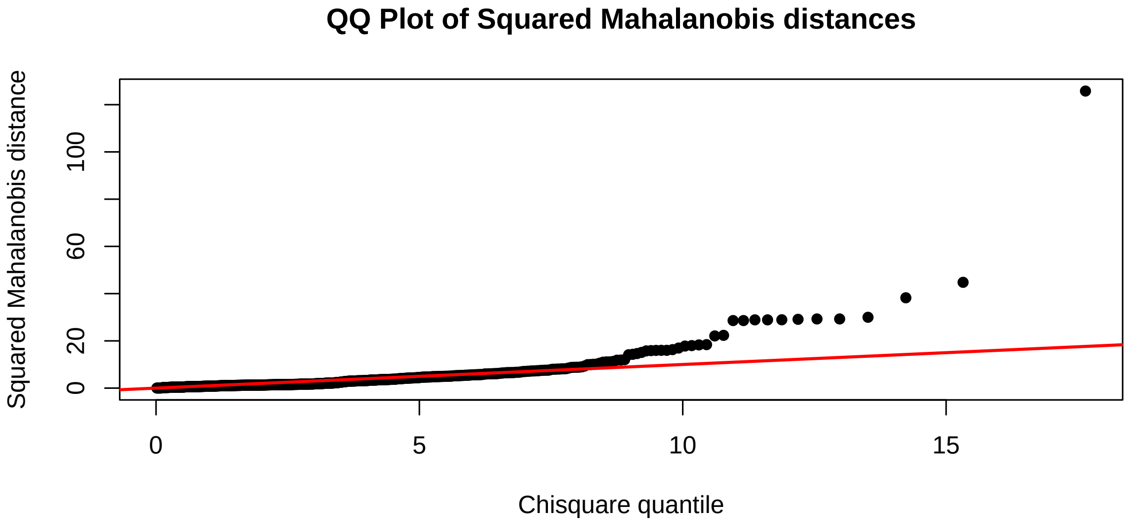
Also here we can do the same considerations about the just above results for PC_2.

However, there is another method to investigate directly the multivariate normality of our observations' distribution, which consists in producing a "Gamma plot" for all the first three PCs. This last is a QQ plot of squared Mahalanobis distances and so if the data is multivariate normal, it is expected that most of the points should fall on the diagonal line, but in this case it does not happen.

```
p<-ncol(food.pca$x[,1:3])
mdist<- mahalanobis(food.pca$x[,1:3],food.pca$center,cov(food.pca$x[,1:3]))
```

```
Warning in sweep(x, 2L, center): STATS è più lungo dell'estensione di 'dim(x)
[MARGIN]'
```

```
plot(qchisq(ppoints(mdist), df=p), sort(mdist), pch=16, xlab="Chisquare quantile",
     ylab="Squared Mahalanobis distance")
abline(a=0, b=1, col="red", lwd=2)
title(main ="QQ Plot of Squared Mahalanobis distances")
```



The results of both methods allow us to conclude that the considered data set (observations) is not jointly distributed as a multivariate Gaussian.

Point 7

Find multivariate outliers through the first three principal components, up to 5 in total. Are they the most extreme observations with respect to the 6 original variables?

To detect the multivariate outliers with respect to the first three principal components we use the "squared Mahalanobis distance", defined as $\Delta^2 = (x-\mu)^T S^{-1}(x-\mu)$. Since we have checked that our multidimensional data are not distributed as a multivariate Gaussian, and we don't know from which distribution those are picked, we decide to select as multivariate outliers the observations with the five biggest squared Mahalanobis distances. In the plot, we have also inserted a line that characterizes the value associated to a chi-square distribution with 3 degrees of freedom and an alpha-value of 0.01, which is $\chi^2_{3,0.01} = 16.266$.

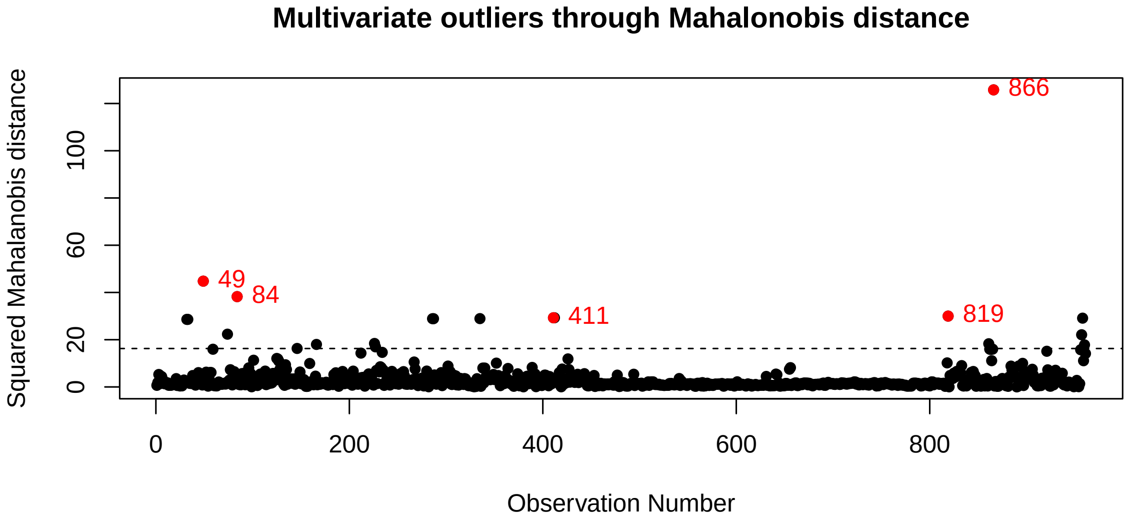
```
mdist_sort<- sort(mdist,decreasing = TRUE)
head(mdist_sort)
```

```
      866        49        84       819       411       412 
125.77298  44.78476  38.24189  29.98765  29.28049  29.28049 
```

```
multiout<- c(866, 49, 84, 819, 411)

plot(mdist,pch=16, xlab="Observation Number", ylab="Squared Mahalanobis distance",
     main="Multivariate outliers through Mahalonobis distance")
points(multiout, mdist[multiout], pch=16, col="red")
text(multiout,mdist[multiout],as.character(multiout),pos=4, col="red")
a<-0.999; qchisq(a,df=3)
```

```
[1] 16.26624
```

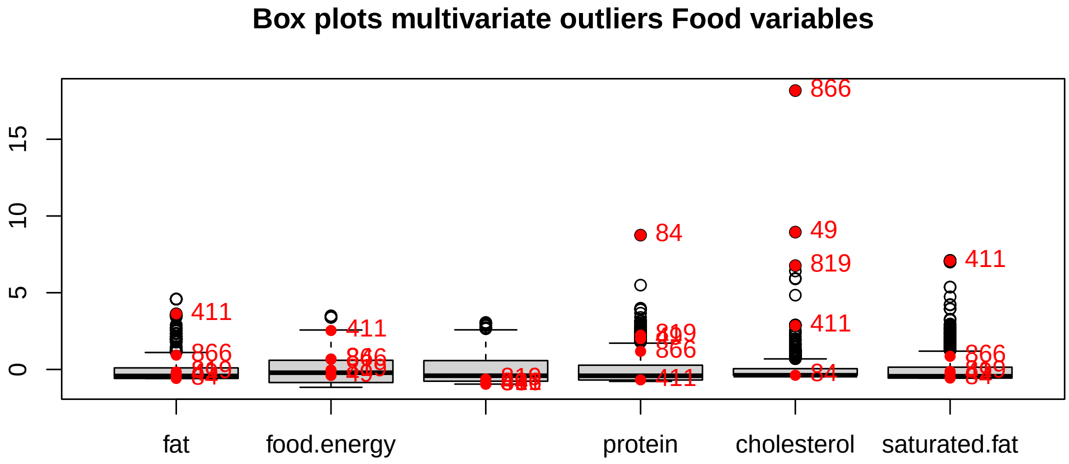
```
abline(h=qchisq(a,df=3), lty=2)
```



(Remark: we have selected as outlier the observation 411 instead of 412 completely arbitrarily since they are overlapped and so they share the same Mahalanobis distance).

To check in a clear way if they are the most extreme observations with respect to the 6 original variables, we produce the box plots of each original food variable and point out in red the previously selected multivariate outliers.

```
boxplot(dt, main="Box plots multivariate outliers Food variables")

points(rep(1,5), dt[multiout,1], pch=16, col="red")
text(rep(1,5),dt[multiout,1],as.character(multiout),pos=4, col="red")

points(rep(2,5), dt[multiout,2], pch=16, col="red")
text(rep(2,5),dt[multiout,2],as.character(multiout),pos=4, col="red")

points(rep(3,5), dt[multiout,3], pch=16, col="red")
text(rep(3,5),dt[multiout,3],as.character(multiout),pos=4, col="red")

points(rep(4,5), dt[multiout,4], pch=16, col="red")
text(rep(4,5),dt[multiout,4],as.character(multiout),pos=4, col="red")

points(rep(5,5), dt[multiout,5], pch=16, col="red")
text(rep(5,5),dt[multiout,5],as.character(multiout),pos=4, col="red")

points(rep(6,5), dt[multiout,6], pch=16, col="red")
text(rep(6,5),dt[multiout,6],as.character(multiout),pos=4, col="red")
```



From the box plots it is possible to observe that only some of them are the most extreme values for the original variables: the observation 84 represents the most extreme value with respect the variable “protein” and the foods 866 and 411 respectively to “cholesterol” and “saturated.fat”.