Assignment 4 & 5, EDDA 2017

Martin de la Riva(11403799) and Kieran O'Driscoll(11426438), Group 23

08 May 2017

```
library(multcomp)

## Loading required package: mvtnorm

## Loading required package: survival

## Loading required package: TH.data

## Loading required package: MASS

##
## Attaching package: 'TH.data'

## The following object is masked from 'package:MASS':
##
##     geyser

library(lme4)

## Loading required package: Matrix
```

Assignment 4

Exercise 1

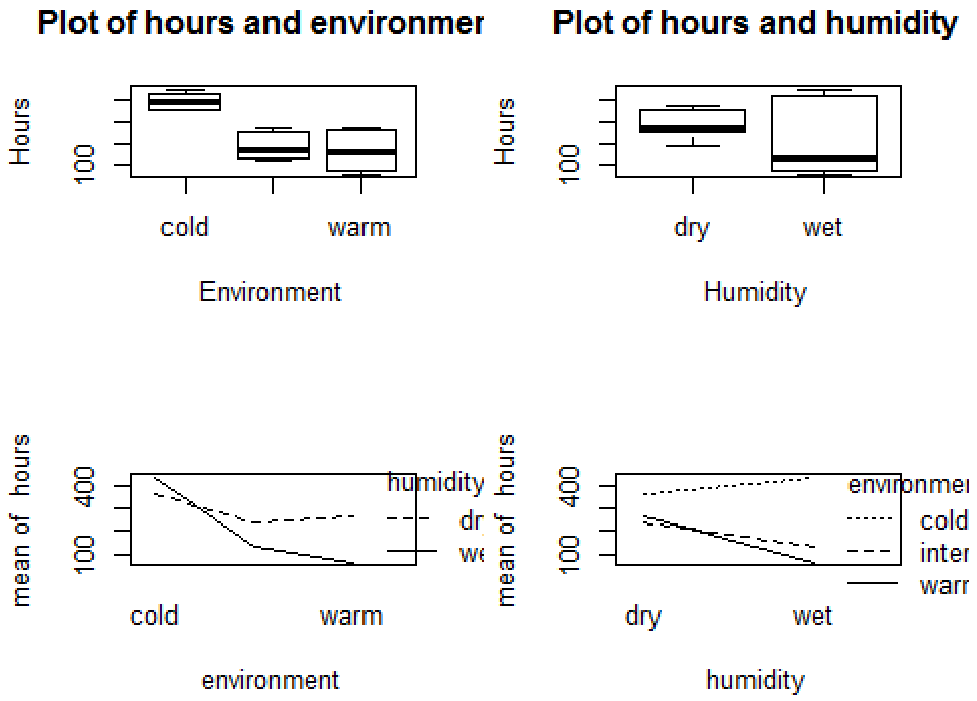
1.

For this exercise

```
bread_data=read.table("data\\bread.txt", header=TRUE)
I=nrow(unique(bread_data['environment']))
J=nrow(unique(bread_data['hours']));
N=3 #number of tests per experiment
randomization = rbind(rep(1:I,each=N*J),rep(1:J,N*I),sample(1:(N*I*J)))
#randomization code
```

2.

```
par(mfrow=c(2,2))
boxplot(hours~environment,data=bread_data, main="Plot of hours and
environment",
    xlab="Environment", ylab="Hours")
boxplot(hours~humidity,data=bread_data, main="Plot of hours and humidity",
    xlab="Humidity", ylab="Hours")
attach(bread_data)

interaction.plot(environment, humidity, hours)
interaction.plot(humidity, environment, hours)
```



3

```
#Analysisof variance
#2 way alnova
bread_data$environment=as.factor(bread_data$environment)
bread_data$humidity=as.factor(bread_data$humidity)
pvcaov=lm(hours~environment*humidity,data=bread_data)
print(anova(pvcaov))

## Analysis of Variance Table
##
## Response: hours
##                      Df Sum Sq Mean Sq F value    Pr(>F)
## environment           2 201904  100952 233.685 2.461e-10 ***
## humidity              1  26912   26912  62.296 4.316e-06 ***
## environment:humidity  2  55984   27992  64.796 3.705e-07 ***
## Residuals            12   5184     432
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Neither the effects of the humidity or environment are significantly different from 0 as they have really low p-values. This falls under the 0.05 range meaning that they have an effect.

The interaction also falls below 0.05 meaning there is evidence that the two are not independent and that their interaction has an effect.

4.

```
contrasts(bread_data$environment)=contr.sum
contrasts(bread_data$humidity)=contr.sum
pvcaov2=lm(hours~environment*humidity, data=bread_data)
print(summary(pvcaov2))
```

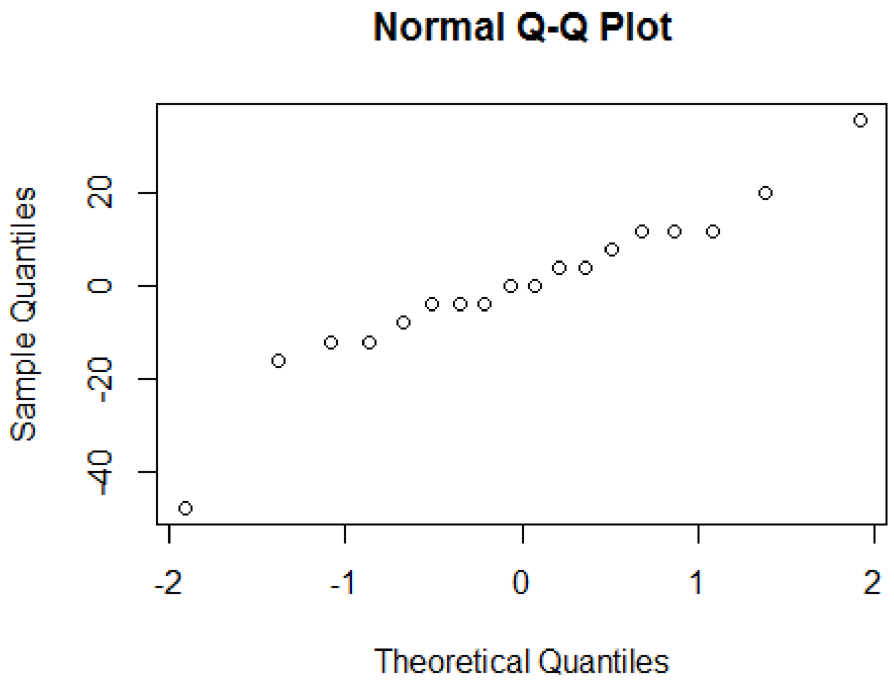
```
## 
## Call:
## lm(formula = hours ~ environment * humidity, data = bread_data)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
##    -48     -7      0     11     36 
## 
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)    
## (Intercept)             250.667      4.899  51.167 2.04e-15 ***
## environment1            149.333      6.928  21.554 5.81e-11 ***
## environment2            -64.667      6.928  -9.334 7.50e-07 ***
## humidity1                38.667      4.899   7.893 4.32e-06 ***
## environment1:humidity1  -74.667      6.928 -10.777 1.59e-07 ***
## environment2:humidity1   15.333      6.928   2.213    0.047 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 20.78 on 12 degrees of freedom
## Multiple R-squared:  0.9821, Adjusted R-squared:  0.9747 
## F-statistic: 131.9 on 5 and 12 DF,  p-value: 4.676e-10
```

Out of the two factors, environment has the greatest influence on the decay. This can be seen in by looking at the p values for the environment above and by analysing the box plots and interaction graphs earlier. The box plot inparticular show a clear difference between the types of environment and the result in the deacay with cold environments taking much longer for decay. However this is not a good question as from ealier analysis and by taking a look at the interaction graph it clear the the environment and humidity have an effect on each other. Therefore to it is hard to say which factor has the biggest effect as each factor is being influenced by the other.

```
  #print(confint(pvcaov2)) #not sure if needed
```

5.

```
  qqnorm(residuals(pvcaov2))
```

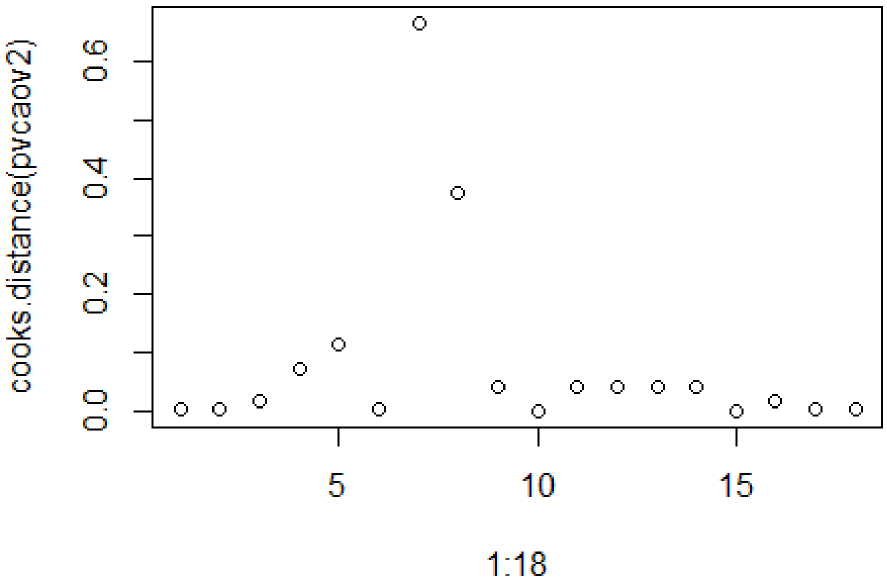


From the qplot it does not seem to be a normal distrubution so the data probably contains some outliers.

```
print(residuals(pvcaov2)) # look for residuals that are outside std

##             1             2             3             4             5
## -4.000000e+00 -4.000000e+00  8.000000e+00 -1.600000e+01  2.000000e+01
##             6             7             8             9            10
## -4.000000e+00 -4.800000e+01  3.600000e+01  1.200000e+01  3.330669e-15
##            11            12            13            14            15
## -1.200000e+01  1.200000e+01 -1.200000e+01  1.200000e+01 -1.110223e-16
##            16            17            18
## -8.000000e+00  4.000000e+00  4.000000e+00
```

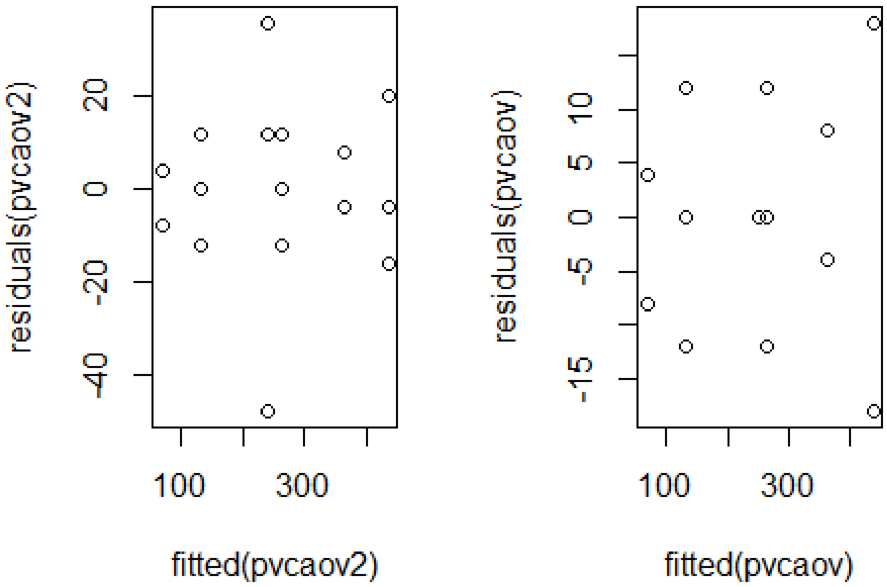
The residuals for the model could indicate some potential outliers. The extreme values for 7 and 8 could be two outliers.

```
round(cooks.distance(pvcaov2),2)

##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15 
## 0.00 0.00 0.02 0.07 0.12 0.00 0.67 0.38 0.04 0.00 0.04 0.04 0.04 0.04 0.00 
##   16   17   18 
## 0.02 0.00 0.00

plot(1:18,cooks.distance(pvcaov2))
```

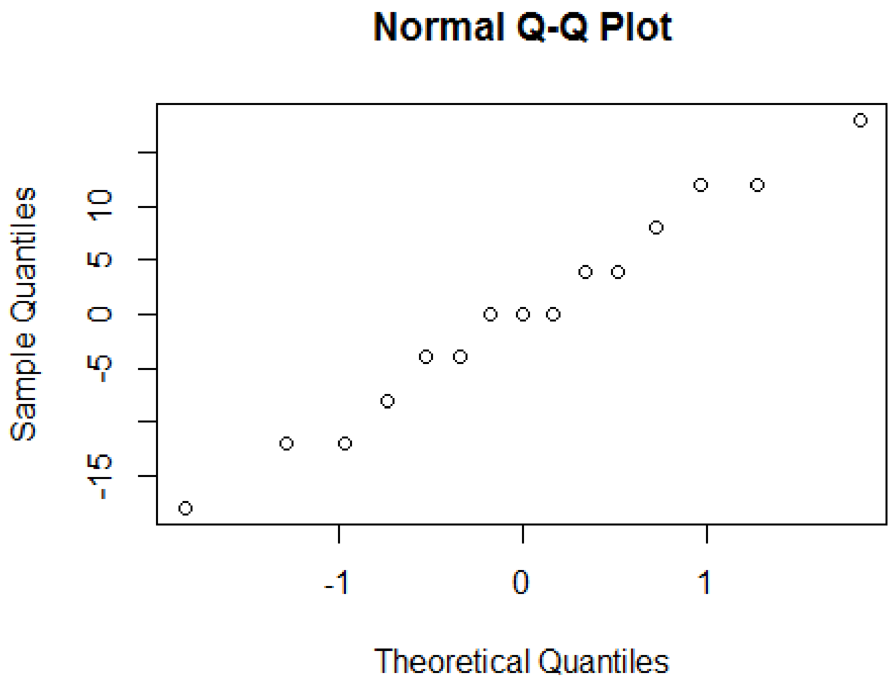


Looking at cooks distance confirms suspicion that 7 and 8 are outliers.

```
par(mfrow=c(1,2))
plot(fitted(pvcaov2),residuals(pvcaov2))
#plot(fitted(pvcaov),residuals(pvcaov))
bread_data2 = bread_data[append(append(c(1:5), c(9:16)), c(17:18)),]
pvcaov=lm(hours~environment*humidity,data=bread_data2)

plot(fitted(pvcaov),residuals(pvcaov))
```



```
 #print(anova(pvcaov))
 #cooks.distance(pvcaov2)
 #plot point on graph with cooks distance and levenes distance
 #library(car)  influencePlot(pvcaov2)

#par(mfrow=c(1,2))
#qqnorm(residuals(pvcaov2))
qqnorm(residuals(pvcaov))
```



By removing the two outliers, a qplot that better resembles a normal distrubution is displayed.

Exercise 2

1.

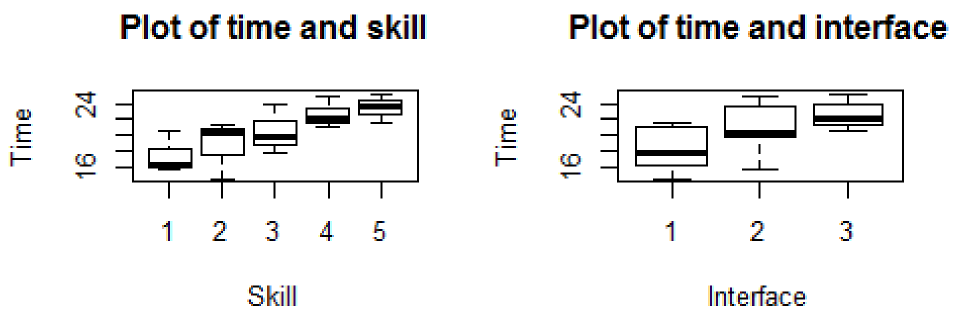
```
I=3; B=5; N=15 # 15 students
for (i in 1:B) print(sample(1:(N*I)))
```

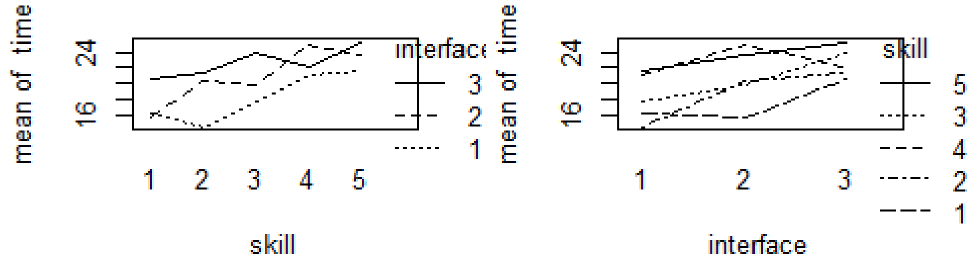
```
##  [1]  4  9  7 28 45 21 27  5 44 34 37 35 38 13 31  1  6 15 14 29 12 42 36
## [24] 33 25 18 30 11  3 26 19 20 24 22 10  8  2 32 17 16 43 41 39 23 40
##  [1] 26  4 24 36 29 22 37  3 38 18 23 16 43  7 15 45 28 44 33 25 20  1 12
## [24] 10 39 31  9  2 11 34 19 14  8 40 27 42 32  5 17  6 30 41 13 21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

2.

```
search_data=read.table("data\\search.txt", header=TRUE)
par(mfrow=c(2,2))
boxplot(time~skill,data=search_data, main="Plot of time and skill",
    xlab="Skill", ylab="Time")
boxplot(time~interface,data=search_data, main="Plot of time and interface",
    xlab="Interface", ylab="Time")
attach(search_data)
```

```
interaction.plot(skill, interface, time)
interaction.plot(interface, skill, time)
```





As expected, the increase in skill level(in this case the skill variable descends to indicate a better skill level) the increase in the time spent. What can be seen is that the skill levels are generally consistent across the interfaces with more skilled individuals generally quicker. Most usersare quicker ion interface 1 while certain skill levels are better on particular interfaces so such as interface 2 and skill level 2. This could be an anomaly due to the small sample size.

3.

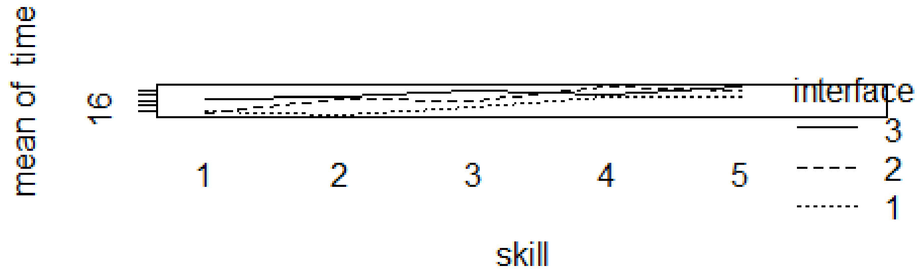
```
search_data=read.table("data\\search.txt", header=TRUE)#need for reset
search_data['skill'] = search_data$skill=as.factor(search_data$skill)
search_data['interface'] =
search_data$interface=as.factor(search_data$interface)

#temp_data = search_data
#temp_data['interface'] = paste("interface", temp_data['interface']) #change
to category
#new_data = xtabs(time~interface+skill,data=search_data)

aovpen=lm(time~interface+skill,data=search_data)
print(anova(aovpen))

## Analysis of Variance Table
##
## Response: time
##           Df Sum Sq Mean Sq F value  Pr(>F)  
## interface  2 50.465 25.2327  7.8237 0.01310 *
## skill      4 80.051 20.0127  6.2052 0.01421 *
## Residuals  8 25.801  3.2252                  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

print(summary(aovpen))

## 
## Call:
## lm(formula = time ~ interface + skill, data = search_data)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -2.5733 -0.6967  0.3867  1.0567  1.7867 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)   15.013      1.227  12.238 1.85e-06 ***
## interface2     2.700      1.136   2.377  0.04474 *  
## interface3     4.460      1.136   3.927  0.00438 ** 
## skill2         1.300      1.466   0.887  0.40118    
## skill3         3.033      1.466   2.069  0.07238 .  
## skill4         5.300      1.466   3.614  0.00684 ** 
## skill5         6.100      1.466   4.160  0.00316 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.796 on 8 degrees of freedom
## Multiple R-squared:  0.8349, Adjusted R-squared:  0.7111 
## F-statistic: 6.745 on 6 and 8 DF,  p-value: 0.008395
```

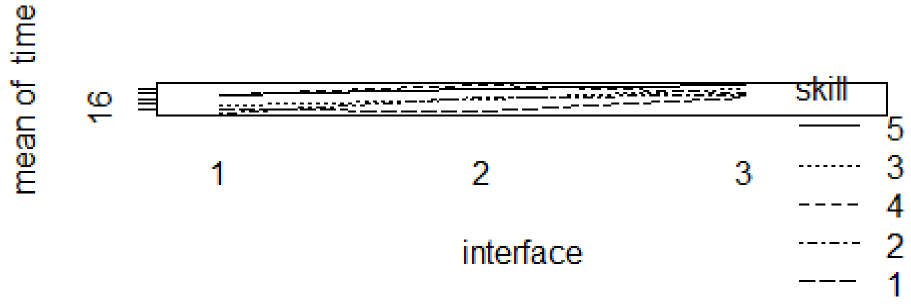
The p-value for the null hypothesis for all interfaces is 0.01310. This falls below the level of 0.05 and therefore the null hypothesis can be rejected. Therefor the search time for all interfaces is different.

4.

This can be estimated using the interaction graphs's. By looking at skill level 4 and interface 3 on the interaction graph, the mean time can be estimated to be 22.5.

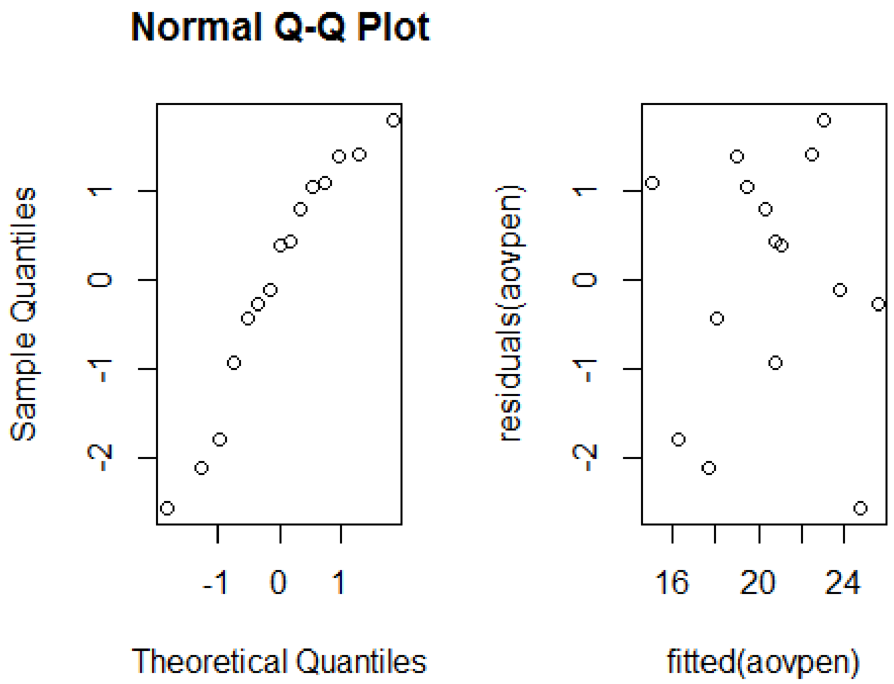
```
par(mfrow=c(2,1))
attach(search_data)

## The following objects are masked from search_data (pos = 3):
## 
##     interface, skill, time

interaction.plot(skill,interface,time)#draw line on graph
interaction.plot(interface,skill,time)
```





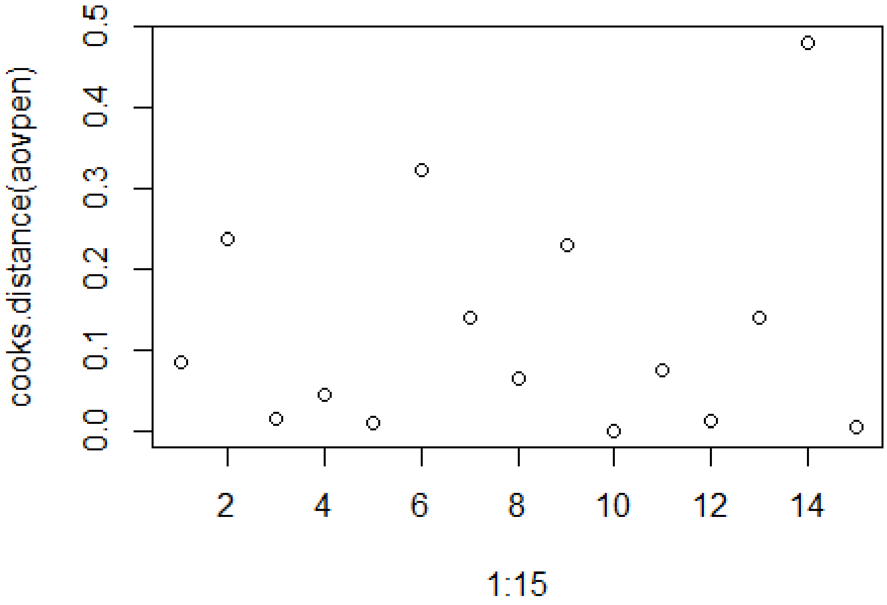
5.

```
par(mfrow=c(1,2))
qqnorm(residuals(aovpen))
plot(fitted(aovpen),residuals(aovpen))
```



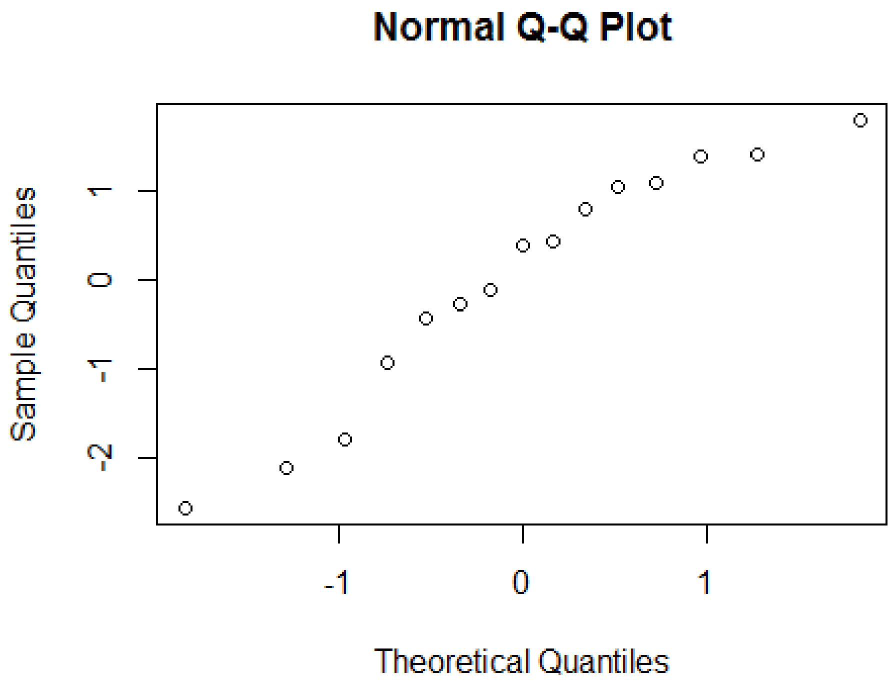
The data looks like it is distrubuted normally with the exception the top right values and the bottom ones. However by analysing the scatter there does not seem to be any outliers. To check this futher cooks distance will be used.

```
round(cooks.distance(aovpen),2)

##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15 
## 0.09 0.24 0.01 0.04 0.01 0.32 0.14 0.07 0.23 0.00 0.08 0.01 0.14 0.48 0.01

plot(1:15,cooks.distance(aovpen))
```



There is potentially two outliers, 14 and 6.

```
search_data2 = search_data[append(append(c(1:5), c(7:13)), c(15:15)),]
aovpen2=lm(time~interface+skill,data=search_data)
qqnorm(residuals(aovpen2))
```



By removing the outliers the q-plot looks more like a normal distribution.

6.

```
friedman.test(time,interface,skill)

##
##  Friedman rank sum test
##
## data:  time, interface and skill
## Friedman chi-squared = 6.4, df = 2, p-value = 0.04076
```

The p-value is 0.04076 is below 0.05 confidence level so the null hull hypothesis can be rejected. Therefore there is an effect on the interface.

7.

```
aovpen=lm(time~interface,data=search_data)
print(anova(aovpen))

## Analysis of Variance Table
##
## Response: time
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## interface  2  50.465  25.233  2.8605 0.09642 .
## Residuals 12 105.852   8.821
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
print(summary(aovpen))

##
## Call:
## lm(formula = time ~ interface, data = search_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -5.26  -1.74  -0.46   2.76   3.94
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   18.160      1.328  13.672 1.12e-08 ***
## interface2     2.700      1.878   1.437   0.1762
## interface3     4.460      1.878   2.374   0.0351 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.97 on 12 degrees of freedom
## Multiple R-squared:  0.3228, Adjusted R-squared:   0.21
## F-statistic: 2.861 on 2 and 12 DF,  p-value: 0.09642
```

P value is 0.09642 which means the null hypothesis cannot be rejected. This is not a good test to perform beacuse by ignoring the skill variable you cannot the check the effect of the interface with respect to the skill levels. it might be the case that certain skill levels perform better on different interfaces.

For this one way analysis to occur each sample should be an independent random sample, the distrubution of the target variable folows the normal distribution and that the variances in the population are equal across target values for each group level.

The sample of students can be presummed to be chosen at random from a normal population. The variance is balanced and the assumption of normaility is proven by the qplot of the residuals.

Excercise 3

1

```
cream_data=read.table("data\\cream.txt", header=TRUE)
cream_data$position = factor(cream_data$position)
cream_data$batch = factor(cream_data$batch)
cream_data$starter = factor(cream_data$starter)
model = lm(acidity~starter+batch+position, data=cream_data)
print(model)

##
## Call:
## lm(formula = acidity ~ starter + batch + position, data = cream_data)
##
## Coefficients:
```

```
## (Intercept)     starter2     starter3     starter4     starter5 
##       8.662       -0.150       -0.980        2.810       -0.484 
##      batch2       batch3       batch4       batch5    position2 
##      -1.348        0.276        1.368        0.200       -0.618 
##   position3    position4    position5 
##      -0.038       -0.764       -0.264
```

```
print(summary(model))
```

```
## 
## Call:
## lm(formula = acidity ~ starter + batch + position, data = cream_data)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -1.2836 -0.2336  0.0384  0.3584  1.0204 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)   8.6616     0.5329  16.255 1.55e-09 ***
## starter2     -0.1500     0.4673  -0.321   0.7538    
## starter3     -0.9800     0.4673  -2.097   0.0579 .  
## starter4      2.8100     0.4673   6.013 6.10e-05 ***
## starter5     -0.4840     0.4673  -1.036   0.3208    
## batch2       -1.3480     0.4673  -2.884   0.0137 *  
## batch3        0.2760     0.4673   0.591   0.5658    
## batch4        1.3680     0.4673   2.927   0.0127 *  
## batch5        0.2000     0.4673   0.428   0.6763    
## position2    -0.6180     0.4673  -1.322   0.2107    
## position3    -0.0380     0.4673  -0.081   0.9365    
## position4    -0.7640     0.4673  -1.635   0.1280    
## position5    -0.2640     0.4673  -0.565   0.5825    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.7389 on 12 degrees of freedom
## Multiple R-squared:  0.9088, Adjusted R-squared:  0.8175 
## F-statistic:  9.96 on 12 and 12 DF,  p-value: 0.0001777
```

From the model we can see starter 4 given a high coefficient along with batch 4. By taking the summary of the model we can gather that starter 4 has a the biggest effect while batch 2 and 4 have a smaller effect. The position variable does not seem to have any significant effect.

2.

```
pvcmult=glht(model,linfct=mcp(starter="Tukey"))
summary(pvcmult)
```

```
## 
##   Simultaneous Tests for General Linear Hypotheses
```

```
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lm(formula = acidity ~ starter + batch + position, data = cream_data)
##
## Linear Hypotheses:
##            Estimate Std. Error t value Pr(>|t|)
## 2 - 1 == 0  -0.1500     0.4673  -0.321 0.997365
## 3 - 1 == 0  -0.9800     0.4673  -2.097 0.282030
## 4 - 1 == 0   2.8100     0.4673   6.013 0.000453 ***
## 5 - 1 == 0  -0.4840     0.4673  -1.036 0.834309
## 3 - 2 == 0  -0.8300     0.4673  -1.776 0.428860
## 4 - 2 == 0   2.9600     0.4673   6.334 0.000289 ***
## 5 - 2 == 0  -0.3340     0.4673  -0.715 0.949082
## 4 - 3 == 0   3.7900     0.4673   8.110  < 1e-04 ***
## 5 - 3 == 0   0.4960     0.4673   1.061 0.822266
## 5 - 4 == 0  -3.2940     0.4673  -7.048 0.000137 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- single-step method)
```

In this case the starters that lead to significantly different acidity re the ones that have p-values less than 0.05. These would include starters 4 and 5.

3.

```
summary(model)
```

```
##
## Call:
## lm(formula = acidity ~ starter + batch + position, data = cream_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2836 -0.2336  0.0384  0.3584  1.0204
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.6616     0.5329  16.255 1.55e-09 ***
## starter2     -0.1500     0.4673  -0.321   0.7538
## starter3     -0.9800     0.4673  -2.097   0.0579 .
## starter4      2.8100     0.4673   6.013 6.10e-05 ***
## starter5     -0.4840     0.4673  -1.036   0.3208
## batch2       -1.3480     0.4673  -2.884   0.0137 *
## batch3        0.2760     0.4673   0.591   0.5658
## batch4        1.3680     0.4673   2.927   0.0127 *
## batch5        0.2000     0.4673   0.428   0.6763
## position2    -0.6180     0.4673  -1.322   0.2107
## position3    -0.0380     0.4673  -0.081   0.9365
## position4    -0.7640     0.4673  -1.635   0.1280
## position5     -0.2640     0.4673  -0.565   0.5825
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7389 on 12 degrees of freedom
## Multiple R-squared:  0.9088, Adjusted R-squared:  0.8175
## F-statistic:  9.96 on 12 and 12 DF,  p-value: 0.0001777
```

These are found in the p-values. Starters 4 6.10e-05. No as only on comparison is done where as in the simultaneous p-value, checks are done with regards to each of the other starters.

4.

```
confint(pvcmult)

##
##   Simultaneous Confidence Intervals
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lm(formula = acidity ~ starter + batch + position, data = cream_data)
##
## Quantile = 3.1869
## 95% family-wise confidence level
##
##
## Linear Hypotheses:
##            Estimate lwr     upr
## 2 - 1 == 0 -0.1500  -1.6394  1.3394
## 3 - 1 == 0 -0.9800  -2.4694  0.5094
## 4 - 1 == 0  2.8100   1.3206  4.2994
## 5 - 1 == 0 -0.4840  -1.9734  1.0054
## 3 - 2 == 0 -0.8300  -2.3194  0.6594
## 4 - 2 == 0  2.9600   1.4706  4.4494
## 5 - 2 == 0 -0.3340  -1.8234  1.1554
## 4 - 3 == 0  3.7900   2.3006  5.2794
## 5 - 3 == 0  0.4960  -0.9934  1.9854
## 5 - 4 == 0 -3.2940  -4.7834 -1.8046
```

In 95% of all experiments all intervals will cover the true difference with the exception of starter 3.

Excercise 4

1.

```
cow_data=read.table("data\\cow.txt", header=TRUE)
cow_data$id=factor(cow_data$id)
cow_data$per=factor(cow_data$per)
```

```
model=lm(milk~treatment+per+id,data=cow_data)
model

##
## Call:
## lm(formula = milk ~ treatment + per + id, data = cow_data)
##
## Coefficients:
## (Intercept)   treatmentB         per2          id2          id3
##       30.30        -0.51        -2.39        23.00        11.15
##         id4          id5          id6          id7          id8
##       -1.35        -7.05        23.45        13.55         4.90
##         id9
##      -11.20
```

2.

```
summary(model)

##
## Call:
## lm(formula = milk ~ treatment + per + id, data = cow_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2600 -0.4375  0.0000  0.4375  2.2600
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  30.3000     1.2444  24.349 5.02e-08 ***
## treatmentB   -0.5100     0.7466  -0.683 0.516536
## per2         -2.3900     0.7466  -3.201 0.015046 *
## id2          23.0000     1.5741  14.612 1.68e-06 ***
## id3          11.1500     1.5741   7.084 0.000196 ***
## id4          -1.3500     1.5741  -0.858 0.419480
## id5          -7.0500     1.5741  -4.479 0.002870 **
## id6          23.4500     1.5741  14.898 1.47e-06 ***
## id7          13.5500     1.5741   8.608 5.69e-05 ***
## id8           4.9000     1.5741   3.113 0.017011 *
## id9         -11.2000     1.5741  -7.115 0.000191 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.574 on 7 degrees of freedom
## Multiple R-squared:  0.9931, Adjusted R-squared:  0.9832
## F-statistic: 100.6 on 10 and 7 DF,  p-value: 1.349e-06
```

There is no significant effect in the treatment but there is an effect in the period. It is estimated you would get -0.51 less milk prodcution.

3.

```
mixed_model=lmer(milk~treatment+order+per+(1|id),data=cow_data,REML=FALSE)
print(summary(mixed_model))

## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: milk ~ treatment + order + per + (1 | id)
##    Data: cow_data
## 
##      AIC      BIC   logLik deviance df.resid 
##    119.3    124.7    -53.7    107.3       12 
## 
## Scaled residuals: 
##      Min       1Q   Median       3Q      Max 
## -1.53111 -0.37104  0.02686  0.26747  1.72489 
## 
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 133.145  11.539  
##  Residual               1.927   1.388  
## Number of obs: 18, groups:  id, 9
## 
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  38.5000     5.8110   6.625
## treatmentB   -0.5100     0.6585  -0.775
## orderBA      -3.4700     7.7685  -0.447
## per2         -2.3900     0.6585  -3.630
## 
## Correlation of Fixed Effects:
##            (Intr) trtmnB ordrBA
## treatmentB -0.063              
## orderBA    -0.743  0.000       
## per2       -0.063  0.111  0.000
```

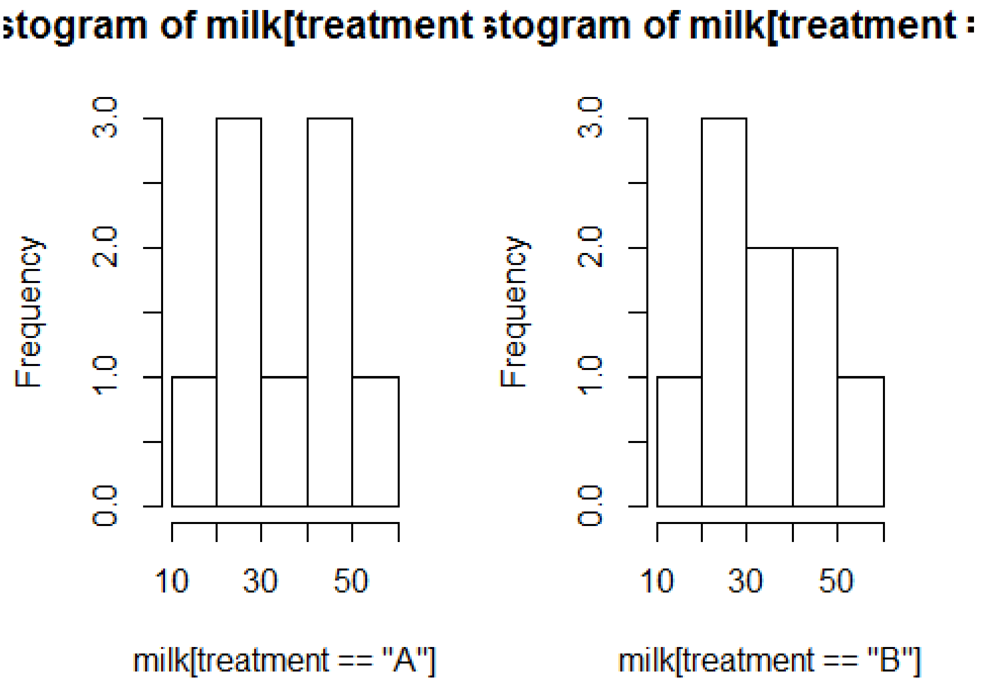
It is estimated you would get -0.51 less milk prodcution which is the same as the fixed results model. There is an estimated variance of 133.145of the normal population of the "individual effects".

```
mixed_model2=lmer(milk~order+per+(1|id),data=cow_data,REML=FALSE)
anova(mixed_model2,mixed_model)

## Data: cow_data
## Models:
## mixed_model2: milk ~ order + per + (1 | id)
## mixed_model: milk ~ treatment + order + per + (1 | id)
##              Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## mixed_model2  5 117.89 122.34 -53.946   107.89                         
## mixed_model   6 119.31 124.65 -53.656   107.31 0.5807      1      0.446
```

The results are the same with no significat difference in the treatment.

4.

```r
attach(cow_data)
t.test(milk[treatment=="A"],milk[treatment=="B"],paired=TRUE)

##
##  Paired t-test
##
## data:  milk[treatment == "A"] and milk[treatment == "B"]
## t = 0.22437, df = 8, p-value = 0.8281
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.267910  2.756799
## sample estimates:
## mean of the differences
##               0.2444444

par(mfrow=c(1,2))
hist(milk[treatment=="A"])
hist(milk[treatment=="B"])
```



The distrubution of treatment a does not look to have been drawn from a normal distribution but this could be a lack of data. If one of the populations from A or B are not drawn froma normal distrubution the paired t-test should not be used.

The paired t-test does not take into the account the order in which treatment was applied or the period is was applied in. The p-value does not reject null hypothesis whhich is the same conclusion as the fixed effects model. However the fixed effects model is a much better test as it analyses the treatements and the periods along with the id of each cow. In the fixed effects model it is implied that cow itself is has a big effect if a treatement works or not.

Assignment 5

Exercise 1

1.

```
nauseatable=read.table('data\\nauseatable.txt', header=TRUE)

nausea=c()
medicin=c()
for(i in 1:nrow(nauseatable)){
  for(j in 1:ncol(nauseatable)){
    medicin=append(medicin, rep(row.names(nauseatable)[i], nauseatable[i,j]))
    nausea=append(nausea, rep(j-1, nauseatable[i,j]))
  }
}

nausea.frame=data.frame(nausea,medicin)
#print(nausea.frame)
```

With this code, an appropiate data frame is created from any table of the same characteristics.

2.

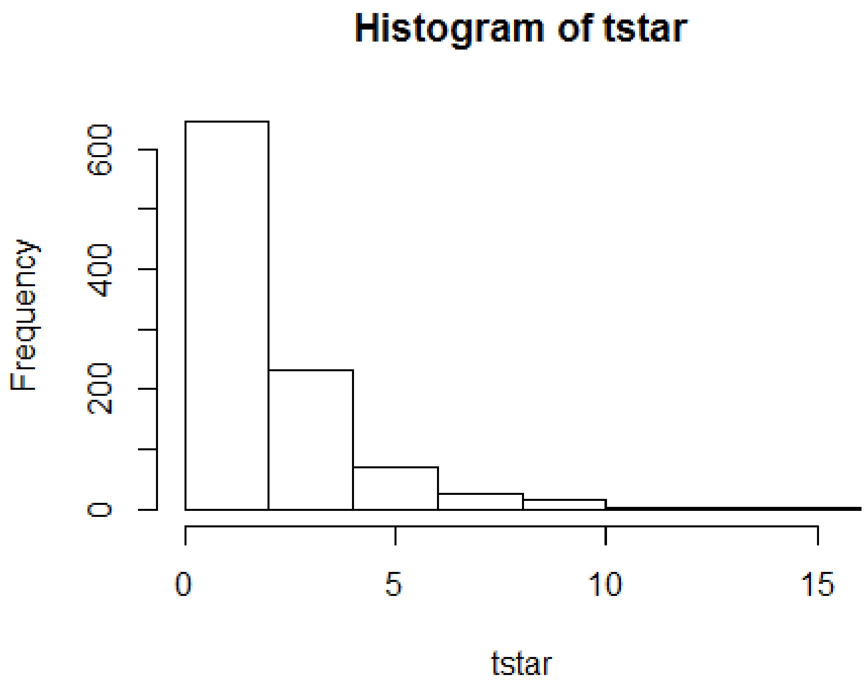
```
nauseatable
##                        Incidence.of.no.nausea Incidence.of.Nausea
## Chlorpromazine                            100                  52
## Pentobarbital(100mg)                       32                  35
## Pentobarbital(150mg)                       48                  37
xtabs(~medicin+nausea)
##                        nausea
## medicin                  0   1
##   Chlorpromazine       100  52
##   Pentobarbital(100mg)  32  35
##   Pentobarbital(150mg)  48  37
```

We can see that the xtabs code makes a table out of a data frame of 2 vectors. With this outcome we can confirm that the transformation of the data.frame is correct.

3.

```
attach(nausea.frame)

## The following objects are masked _by_ .GlobalEnv:
##
##     medicin, nausea
```

```
B=1000
tstar=pstar=numeric(B)
for (i in 1:B){
    nausstar=sample(nausea) ## permuting the labels
    tstar[i]=chisq.test(xtabs(~medicin+nausstar))[[1]]
}
myt=chisq.test(xtabs(~medicin+nausea))[[1]]
hist(tstar)
```



```
pl=sum(tstar<myt)/B
pr=sum(tstar>myt)/B
pr

## [1] 0.042
```

Having as H_0=The different medicins work equally well against nausea.

Really low chi-square values are registered (mostly less than 2) when permuting the labels. We get a (bootstrap fashion) p-value of around 0.042. Therefore, we can reject the hypothesis. In other words, the different medicins work differently against nausea.

4.

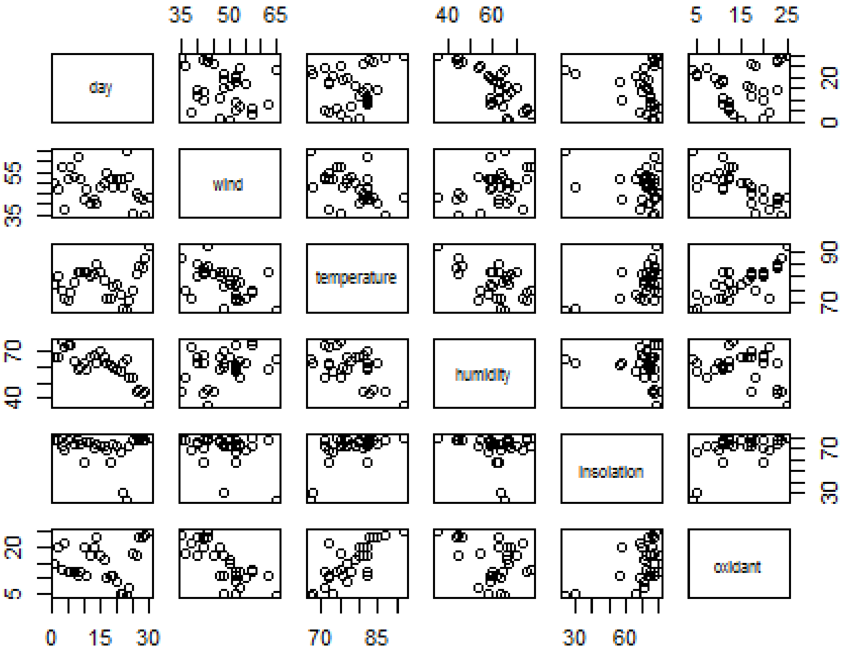
```
chisq.test(xtabs(~medicin+nausea))[[3]]

## [1] 0.03642928
```

Chi-Square test returns a really similar value as the one in the permutation test.

Exercise 2

1.

```
airpollution=read.table('data\\airpollution.txt', header=TRUE)
pairs(airpollution)
```



There seems to be clear relation between oxidant with temperature, insolation and wind. There also seems to be related with humidity and day (although not that clear). A linear model looks useful for this data.

2.

```
summary(lm(oxidant~insolation, data=airpollution))$r.squared#0.2551683
## [1] 0.2551683
summary(lm(oxidant~humidity, data=airpollution))$r.squared#0.12402
## [1] 0.12402
summary(lm(oxidant~temperature, data=airpollution))$r.squared#0.5760164
## [1] 0.5760164
summary(lm(oxidant~wind, data=airpollution))$r.squared#0.5863157
## [1] 0.5863157
summary(lm(oxidant~day, data=airpollution))$r.squared#0.01093407
## [1] 0.01093407
```

As said before, the most relevant variables for the linear model are (in order of importance): wind, temperature, insolation, humidity and day.

Therefore, we will start with a linear model with wind as it's first explanatory variable, and we will add the variables that increase the determinant coefficient until it does not increase anymore (i.e. a step-up method).

```
summary(lm(oxidant~wind+humidity, data=airpollution))$r.squared
## [1] 0.5913056
summary(lm(oxidant~wind+insolation, data=airpollution))$r.squared
## [1] 0.66131
summary(lm(oxidant~wind+day, data=airpollution))$r.squared
## [1] 0.5988604
summary(lm(oxidant~wind+temperature, data=airpollution))$r.squared
## [1] 0.7773065
# therefore we add temperature
summary(lm(oxidant~wind+temperature+humidity, data=airpollution))
## 
## Call:
## lm(formula = oxidant ~ wind + temperature + humidity, data = airpollution)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -6.5887 -1.1686  0.1978  1.9004  4.1544 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -16.60697   13.07154  -1.270    0.215    
## wind         -0.44620    0.08513  -5.241 1.78e-05 ***
## temperature   0.60190    0.11764   5.117 2.47e-05 ***
## humidity      0.09850    0.06316   1.559    0.131    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.874 on 26 degrees of freedom
## Multiple R-squared:  0.7964, Adjusted R-squared:  0.7729 
## F-statistic: 33.89 on 3 and 26 DF,  p-value: 3.904e-09
summary(lm(oxidant~wind+temperature+insolation, data=airpollution))
## 
## Call:
## lm(formula = oxidant ~ wind + temperature + insolation, data =
airpollution)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
## -6.407 -2.056  1.012  1.760  4.792 
## 
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -4.45496   11.26714  -0.395 0.695778    
## wind        -0.42353    0.08737  -4.848 5.02e-05 ***
## temperature  0.47558    0.12564   3.785 0.000816 ***
## insolation   0.03646    0.05071   0.719 0.478636    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.976 on 26 degrees of freedom
## Multiple R-squared:  0.7816, Adjusted R-squared:  0.7565
## F-statistic: 31.02 on 3 and 26 DF,  p-value: 9.583e-09
summary(lm(oxidant~wind+temperature+day, data=airpollution))
##
## Call:
## lm(formula = oxidant ~ wind + temperature + day, data = airpollution)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.9010 -1.3477  0.1596  1.7766  3.9405
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.98987   10.94466  -0.273    0.787
## wind        -0.45604    0.08644  -5.276 1.63e-05 ***
## temperature  0.52918    0.10568   5.008 3.29e-05 ***
## day         -0.09711    0.06328  -1.535    0.137
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.878 on 26 degrees of freedom
## Multiple R-squared:  0.7958, Adjusted R-squared:  0.7722
## F-statistic: 33.78 on 3 and 26 DF,  p-value: 4.042e-09
# Adding none of the variables yields significance. Therefore we stop at the
previous model.
```

Looking at the R^2, it increases when adding more variables, although insignificantly when having added the most relevant ones. Investigating the summary and the p-value (using hypothesis $H_0: \beta_i = 0$) using the full linear model, we can see that insolation, humidity and day do not apport much information.

From those 3 variables, humidity seems to be the most relevant one, reaching a p-value of 0.131 when having a wind+temperature+humidity model. Besides this, the p-value is above 0.05 and the increase in R^2 is still not significant. Therefore, we do not add any more variables to the model, finishing with a oxidant~wind+temperature model.

3.

```
summary(lm(oxidant~wind+temperature+insolation+humidity+day,
data=airpollution))
##
## Call:
## lm(formula = oxidant ~ wind + temperature + insolation + humidity +
##     day, data = airpollution)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -6.6920 -1.1675  0.2582  1.8289  4.0773
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.04010   21.20961  -0.568  0.57553
## wind         -0.44749    0.09103  -4.916 5.14e-05 ***
## temperature   0.55714    0.15347   3.630  0.00133 **
## insolation    0.01822    0.05583   0.326  0.74694
## humidity      0.06818    0.13336   0.511  0.61384
## day          -0.02997    0.13995  -0.214  0.83227
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.977 on 24 degrees of freedom
## Multiple R-squared:  0.7984, Adjusted R-squared:  0.7564
## F-statistic: 19.01 on 5 and 24 DF,  p-value: 1.203e-07
summary(lm(oxidant~wind+temperature+insolation+humidity, data=airpollution))
##
## Call:
## lm(formula = oxidant ~ wind + temperature + insolation + humidity,
##     data = airpollution)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5861 -1.0961  0.3512  1.7570  4.0712
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -15.49370   13.50647  -1.147  0.26219
## wind         -0.44291    0.08678  -5.104 2.85e-05 ***
## temperature   0.56933    0.13977   4.073  0.00041 ***
## insolation    0.02275    0.05067   0.449  0.65728
## humidity      0.09292    0.06535   1.422  0.16743
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.92 on 25 degrees of freedom
## Multiple R-squared:  0.798,  Adjusted R-squared:  0.7657
## F-statistic: 24.69 on 4 and 25 DF,  p-value: 2.279e-08
summary(lm(oxidant~wind+temperature+humidity, data=airpollution))
##
## Call:
## lm(formula = oxidant ~ wind + temperature + humidity, data = airpollution)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5887 -1.1686  0.1978  1.9004  4.1544
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -16.60697   13.07154  -1.270    0.215
## wind         -0.44620    0.08513  -5.241 1.78e-05 ***
## temperature   0.60190    0.11764   5.117 2.47e-05 ***
## humidity      0.09850    0.06316   1.559    0.131
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.874 on 26 degrees of freedom
## Multiple R-squared:  0.7964, Adjusted R-squared:  0.7729
## F-statistic: 33.89 on 3 and 26 DF,  p-value: 3.904e-09
summary(lm(oxidant~wind+temperature, data=airpollution))
##
## Call:
## lm(formula = oxidant ~ wind + temperature, data = airpollution)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.3939 -1.8608  0.5826  1.9461  4.9661
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.20334   11.11810  -0.468    0.644
## wind        -0.42706    0.08645  -4.940 3.58e-05 ***
## temperature  0.52035    0.10813   4.812 5.05e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.95 on 27 degrees of freedom
## Multiple R-squared:  0.7773, Adjusted R-squared:  0.7608
## F-statistic: 47.12 on 2 and 27 DF,  p-value: 1.563e-09
```

Using a step-down process using the hypothesis $H_0: \beta_i = 0$, we get that day, insolation and humidity get extracted from the model. This is as all of them have a p-value higher than 0.05 on every step, as it can be seen above. This leaves a linear model with wind+temperature.

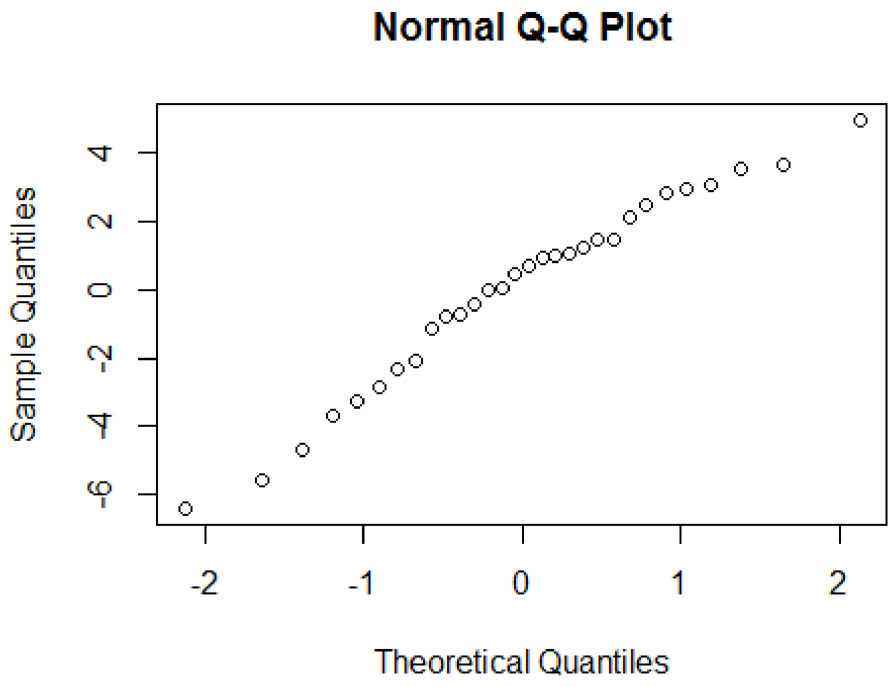
4.

Final model:

$$-5.20334 - 0.42706 * wind + 0.52035 * temperature + error$$

5.

```
lm1 = summary(lm(oxidant~wind+temperature, data=airpollution))
qqnorm(residuals(lm1))
```



```
shapiro.test(residuals(lm1))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(lm1)
## W = 0.96591, p-value = 0.4342
```

The normality assumption of the residuals seems correct, and it therefore seems like a correct model.

Exercise 3

Finding a model

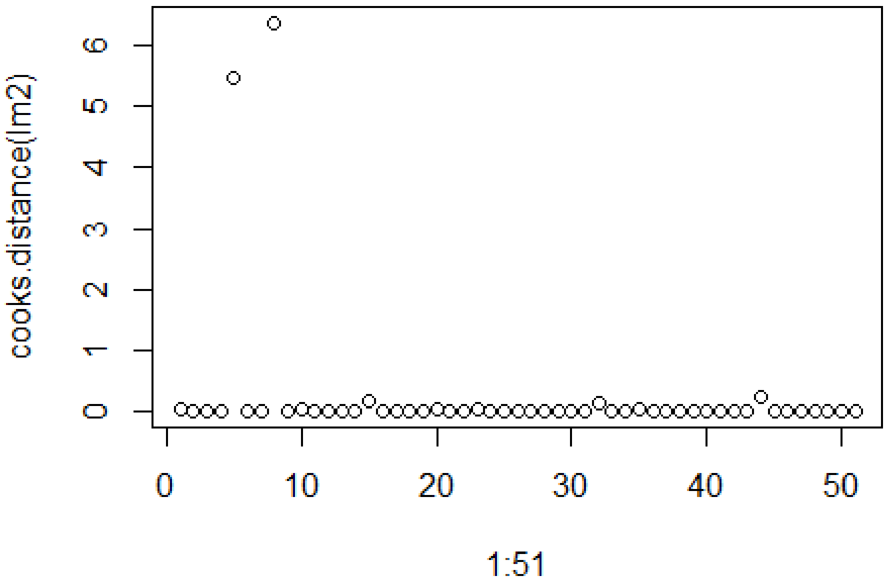
```
expensescrime=read.table('data\\expensescrime.txt', header=TRUE)

### step-down model
summary(lm(expend~employ+lawyers+pop+bad+crime, data=expensescrime))
##
## Call:
## lm(formula = expend ~ employ + lawyers + pop + bad + crime, data =
expensescrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -638.41  -87.42   22.15  114.96  804.98
```

```
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.991e+02  1.401e+02  -2.136  0.03817 *
## employ       2.297e-02  7.462e-03   3.078  0.00354 **
## lawyers      2.324e-02  8.044e-03   2.890  0.00592 **
## pop          7.787e-02  3.515e-02   2.215  0.03184 *
## bad         -2.832e+00  1.240e+00  -2.283  0.02719 *
## crime        3.241e-02  2.813e-02   1.152  0.25534
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 225.6 on 45 degrees of freedom
## Multiple R-squared:  0.9675, Adjusted R-squared:  0.9639
## F-statistic: 268.2 on 5 and 45 DF,  p-value: < 2.2e-16
# delete crime
summary(lm(expend~employ+lawyers+pop+bad, data=expensescrime))
##
## Call:
## lm(formula = expend ~ employ + lawyers + pop + bad, data = expensescrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -635.62  -80.18   18.77  114.54  809.66
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.464e+02  4.541e+01  -3.224  0.00232 **
## employ       2.283e-02  7.487e-03   3.049  0.00380 **
## lawyers      2.646e-02  7.571e-03   3.495  0.00106 **
## pop          6.368e-02  3.304e-02   1.927  0.06012 .
## bad         -2.241e+00  1.133e+00  -1.977  0.05402 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 226.4 on 46 degrees of freedom
## Multiple R-squared:  0.9666, Adjusted R-squared:  0.9637
## F-statistic: 332.5 on 4 and 46 DF,  p-value: < 2.2e-16
# delete pop
summary(lm(expend~employ+lawyers+bad, data=expensescrime))
##
## Call:
## lm(formula = expend ~ employ + lawyers + bad, data = expensescrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -631.75  -93.69   30.34   89.68  963.67
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.106e+02  4.261e+01  -2.595  0.01257 *
## employ       3.232e-02  5.803e-03   5.569  1.2e-06 ***
## lawyers      2.631e-02  7.786e-03   3.379  0.00147 **
## bad         -8.627e-01  9.042e-01  -0.954  0.34496
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 232.8 on 47 degrees of freedom
## Multiple R-squared:  0.9639, Adjusted R-squared:  0.9616
## F-statistic:   418 on 3 and 47 DF,  p-value: < 2.2e-16
# delete bad
summary(lm(expend~employ+lawyers, data=expensescrime))
##
## Call:
## lm(formula = expend ~ employ + lawyers, data = expensescrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -599.47  -94.43   36.01   91.98  936.55
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.107e+02  4.257e+01  -2.600  0.01236 *
## employ       2.971e-02  5.114e-03   5.810 4.89e-07 ***
## lawyers      2.686e-02  7.757e-03   3.463  0.00113 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 232.6 on 48 degrees of freedom
## Multiple R-squared:  0.9632, Adjusted R-squared:  0.9616
## F-statistic: 627.7 on 2 and 48 DF,  p-value: < 2.2e-16
# done
```

Using a step-down approach to choose the variables of the linear model, we end up with expend~employ+lawyers. Step-up approach was also tested, leading to the same result.

Influence points

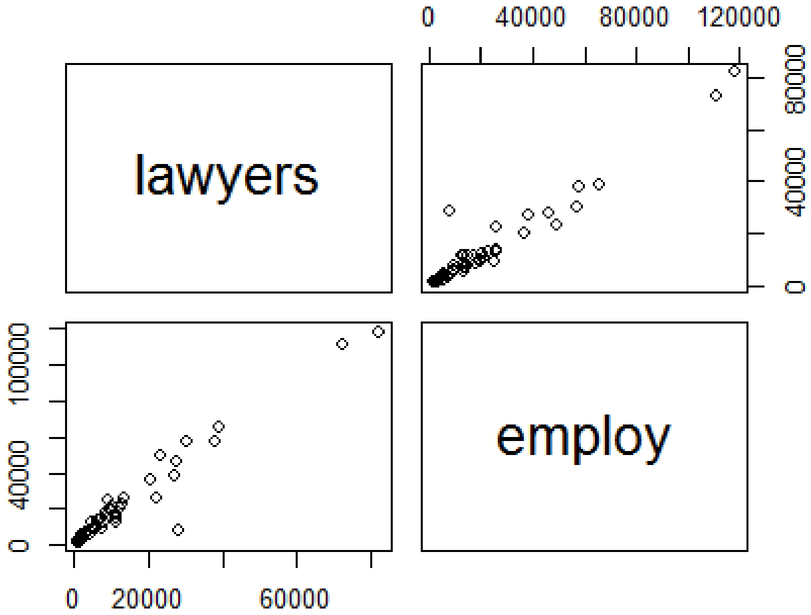
```
lm2 = lm(expend~employ+lawyers, data=expensescrime)
round(cooks.distance(lm2),2)

##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15
## 0.02 0.00 0.00 0.01 5.47 0.00 0.00 6.38 0.00 0.02 0.01 0.00 0.00 0.00 0.17
##   16   17   18   19   20   21   22   23   24   25   26   27   28   29   30
## 0.00 0.00 0.00 0.00 0.02 0.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0.00 0.00
##   31   32   33   34   35   36   37   38   39   40   41   42   43   44   45
## 0.00 0.14 0.00 0.00 0.02 0.01 0.00 0.00 0.01 0.00 0.00 0.00 0.00 0.22 0.00
##   46   47   48   49   50   51
## 0.00 0.00 0.00 0.00 0.00 0.00

plot(1:51, cooks.distance(lm2))
```



It can be clearly seen that the model has 2 influence points: point 5 and 8, with Cook's distances 5.47 and 6.38 respectively.

Collinearity

```
pairs(lawyers~employ, data=expensescrime)
```



Graphically, a clear collinearity between the variables lawyers and employ can be seen.

```
round(cor(expensescrime[,5:6]),2)

##          lawyers employ
## lawyers     1.00   0.97
## employ      0.97   1.00
```

Numerically we confirm their collinearity (0.97). Therefore, we should remove one of the variables. Checking the models with both variables, we can decide to keep employ, as it has a higher determination coefficient. We end with a model of the form expend~employ.

```
summary(lm(expend~lawyers, data=expensescrime))

##
## Call:
## lm(formula = expend ~ lawyers, data = expensescrime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1504.24   -26.77    36.66    95.05   827.01
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept) -59.611979  53.799425  -1.108    0.273
## lawyers       0.070385   0.002601  27.060   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 300.4 on 49 degrees of freedom
## Multiple R-squared:  0.9373, Adjusted R-squared:  0.936
## F-statistic: 732.2 on 1 and 49 DF,  p-value: < 2.2e-16
```

```
summary(lm(expend~employ, data=expensescrime))
```

```
##
## Call:
## lm(formula = expend ~ employ, data = expensescrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -636.04  -84.35   47.60  107.99 1124.70
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.167e+02  4.706e+01   -2.48   0.0166 *
## employ       4.681e-02  1.469e-03   31.87   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 257.4 on 49 degrees of freedom
## Multiple R-squared:  0.954,  Adjusted R-squared:  0.953
## F-statistic:  1016 on 1 and 49 DF,  p-value: < 2.2e-16
```