EDDA ASSIGNMENT 2 - GROUP 14

Ella Smorenburg (2618639), Yoes Ywema (271544), Roel Rotteveel (271547)

08-03-2021

EDDA Assignment 2

Exercise 1: Moldy Bread

```
bread <- read.table("bread.txt", header=TRUE)
attach(bread)
```

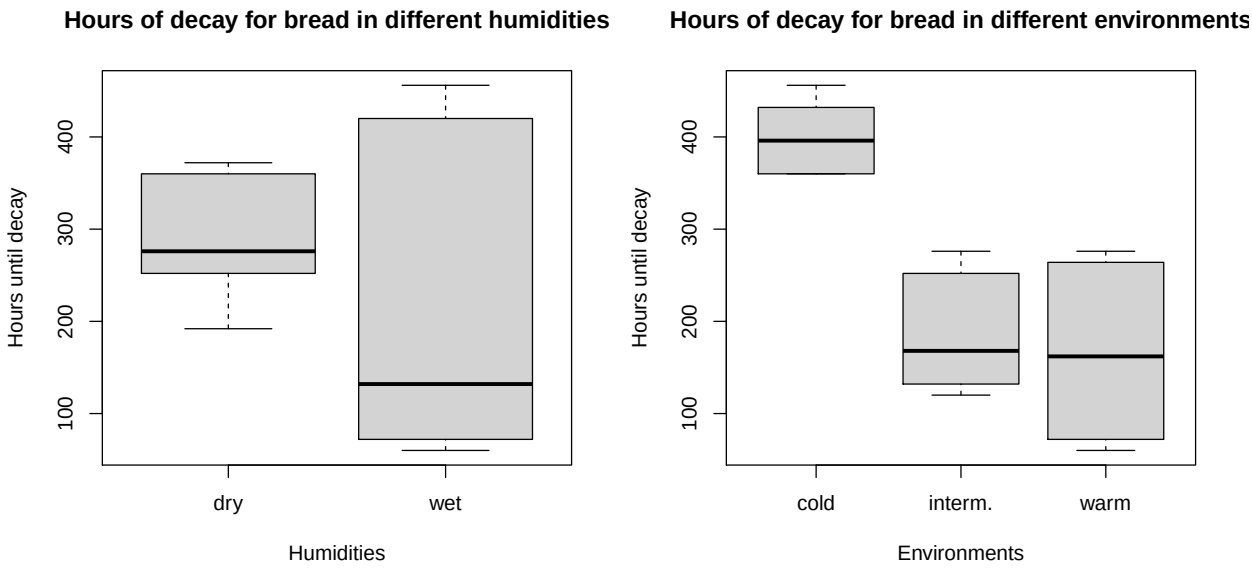
A)

```
I=3;J=2;N=3
test_division = cbind(sample(1:(N*I*J)), rep(1:I,each=N*J), rep(1:J,N*I))
colnames(test_division) = c("slice of bread", "environments", "humidity")
test_division
```

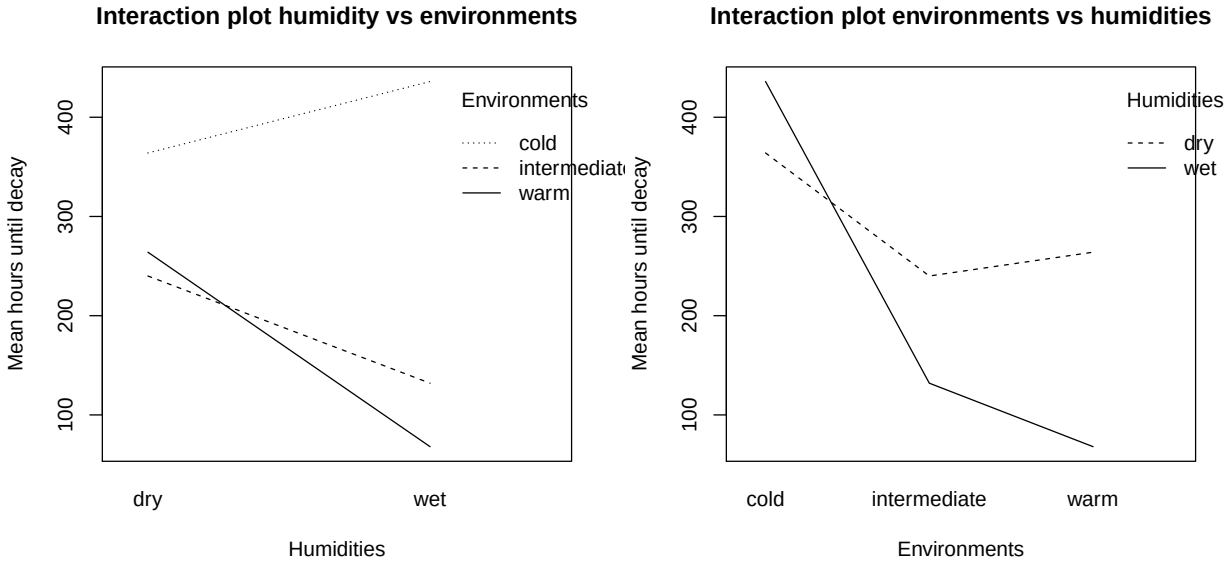
```
##       slice of bread environments humidity
##  [1,]             16            1        1
##  [2,]             14            1        2
##  [3,]             10            1        1
##  [4,]              9            1        2
##  [5,]             18            1        1
##  [6,]              2            1        2
##  [7,]              4            2        1
##  [8,]              1            2        2
##  [9,]              7            2        1
## [10,]              8            2        2
## [11,]             17            2        1
## [12,]              3            2        2
## [13,]             13            3        1
## [14,]              6            3        2
## [15,]             11            3        1
## [16,]             12            3        2
## [17,]             15            3        1
## [18,]              5            3        2
```

B)

```
par(mfrow=c(1,2))
boxplot(hours~humidity, main="Hours of decay for bread in different humidities",
        names = c("dry", "wet"), xlab = "Humidities", ylab = "Hours until decay")
boxplot(hours~environment, main="Hours of decay for bread in different environments", names = c("cold",
```




```
interaction.plot(humidity, environment, hours, ylab="Mean hours until decay", xlab="Humidities", trace.
interaction.plot(environment, humidity, hours, ylab="Mean hours until decay", xlab="Environments", trac
```



From the Boxplots we see that the variance of wet is quite large, and that the mean of 'cold' is quite high. If we look at the interaction plot we see that not all lines are parallel, and that intermediate and warm also cross each other when going from dry to wet. Especially the fact that not all lines are parallel indicates

that there is interaction between the humidity and the environment, probably strongest between 'wet' and 'intermediate'/'warm'.

C)

```
environment=as.factor(environment); humidity=as.factor(humidity)
breadaov=lm(hours~environment*humidity); anova(breadaov)
```

```
## 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
```

As visible in the analysis, the p-value for testing H0: $alpha_i = 0$ for all i is 2.461e-10; for H0: $beta_j=0$ for all i is 4.316e-06; for H0: $gamma_{ij}=0$ for all (i,j) is 3.705e-07. So, there is indeed evidence for interaction between the environment and the humidity, as the p-value is lower than 0.05. In other words the impact of the environment on the number of hours before decay depends on the temperature level. The warmer it is the faster the decay goes in the wet humidity in comparison with the dry humidity.

D)

```
anova(breadaov)
```

```
## 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
```

Since we concluded that the interaction between environment and humidity influences the decay, we use the interactive model to compare the influence of both factors. According to the analysis both factors (environment and humidity) have a main effect on the decay. However the influence of the environment is more significant according to its lower p-value than the influence of the humidity. This tells us that we are more certain that the influence of the environment is non-zero than that the influence of the humidity is non-zero. This could lead to thinking that the effect of the environment factor is larger than the humidity. However this is not necessarily true since the p-values are not a measure of the degree of influence, but instead only a measure of certainty that the influence is non-zero.

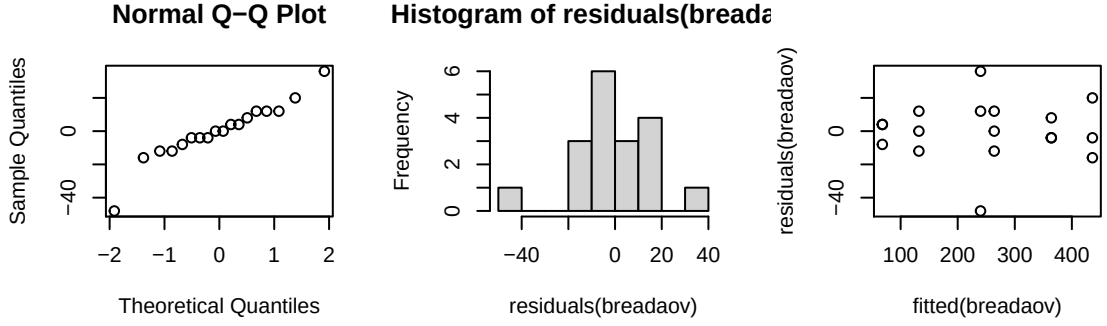
```
summary(breadaov)
```

```
##
## Call:
## lm(formula = hours ~ environment * humidity)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##    -48     -7      0     11     36
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          364.00      12.00  30.333 1.03e-12 ***
## environmentintermediate             -124.00      16.97  -7.307 9.39e-06 ***
## environmentwarm                     -100.00      16.97  -5.893 7.34e-05 ***
## humiditywet                           72.00      16.97   4.243  0.00114 **
## environmentintermediate:humiditywet -180.00      24.00  -7.500 7.23e-06 ***
## environmentwarm:humiditywet         -268.00      24.00 -11.167 1.07e-07 ***
## ---
## 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
```

Nonetheless the summary can give us more information about the influence of individual factors on the decay. Here we can see that the estimated effect of "intermediate" or "warm" temperatures shortens the decay with respectively 124 and 100 hours in comparison with the "cold" temperature. The humidity is slightly less influencing. A "wet" humidity stretches out the decay by approximately 72 hours. So these results suggest the temperature is of greater influence than the humidity. So this is a valid question to ask.

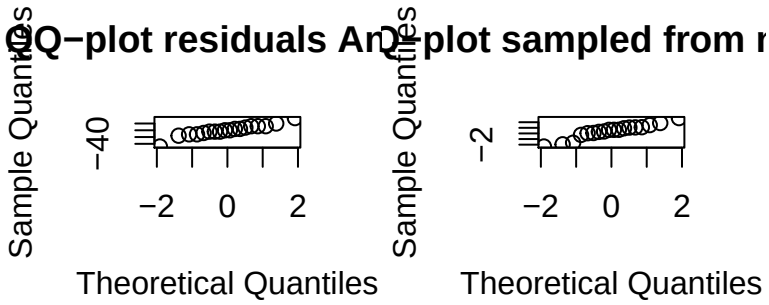
E)

```
par(mfrow=c(1,3))
qqnorm(residuals(breadaov))
hist(residuals(breadaov))
plot(fitted(breadaov),residuals(breadaov))
```



```
#"normal" is a previously sampled simulated normal distribution
{normal = c(1.85545684, 0.27601926, -2.33453968,  2.73751211, 1.11431329, -1.65630464, 1.07472446, 0.648
```

```
# compare normality for similar sized normally generated data
par(mfrow=c(1,2))
qqnorm(residuals(breadaov), main= "QQ-plot residuals Anova")
qqnorm(normal, main= "QQ-plot sampled from normal")
```



When plotting the QQ-plot of the Anova residuals we do witness a straight line but not fully from corner to corner. When we look at the histogram it becomes clear why this is the case. There are two outliers in the dataset where the linear anova model's prediction is not very accurate. These are at -48 and 36. Apart from these outliers the data seems to look normally distributed. Especially if you compare it with a QQ-plot of 18 observations sampled from a previously simulated normal distribution with the same degrees of freedom. The variance among residuals seems not to be skewed for different decay-durations, however for the two extreme values/outliers the residuals differ more than for the other data points.

```
detach(bread)
```

Exercise 2: Search Engine

```
searchengine <- read.table("search.txt", header=TRUE)
attach(searchengine)
```

A)

```
# Make table with available information per participant
id = c(1:nrow(searchengine))
skill = searchengine$skill
treatm = NaN*c(1:nrow(searchengine))
randomized = data.frame(id, skill, treatm)
# Order participants by their skill
randomized = randomized[order(randomized$skill),]
# Create randomization table
I=3;B=5;N=1
a = matrix(0, nrow=B, ncol=N*I, byrow=TRUE)
for (i in 1:B)
```

```
    a[i,] = (sample(1:(N*I)))
# Use randomization table for filling in the treatments per participant id
for (r in 1:B)
    for (c in 1:I)
        randomized$treatm[(r-1)*I + c] = a[r,c]
# Print randomization table and assigned treatments to participant with a certain skill
a
```

```
##      [,1] [,2] [,3]
## [1,]    2    1    3
## [2,]    3    1    2
## [3,]    1    2    3
## [4,]    3    2    1
## [5,]    3    1    2
```

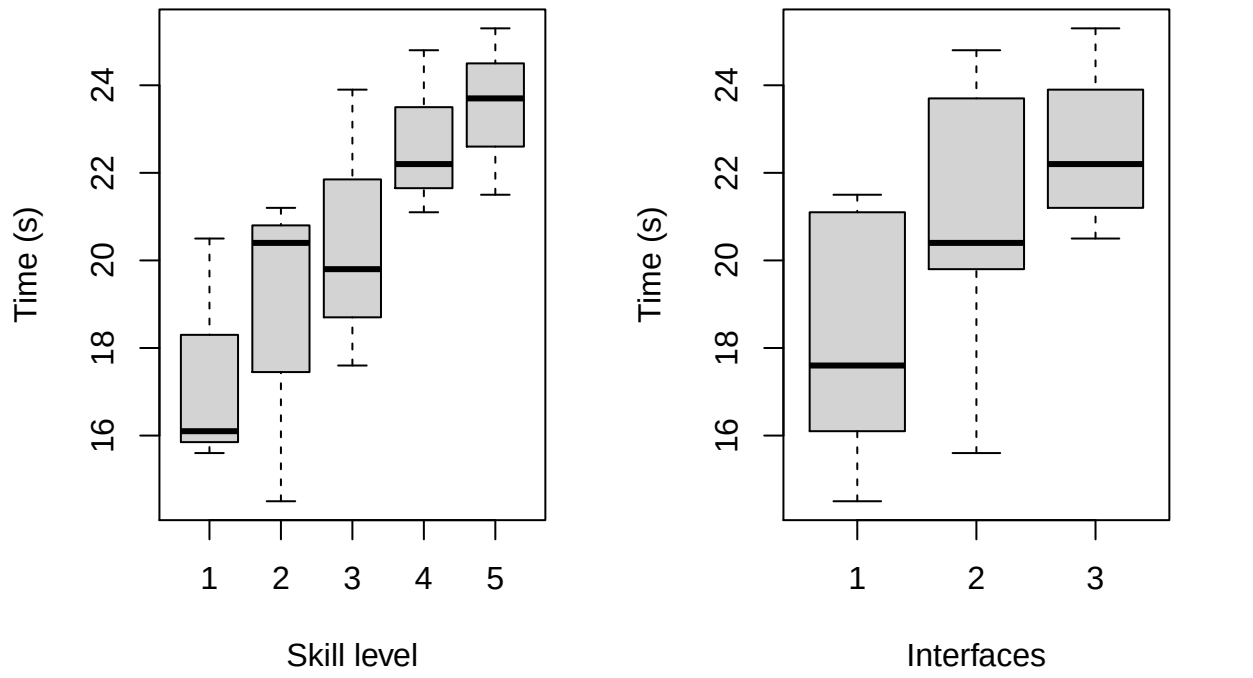
```
randomized
```

```
##    id skill treatm
## 1   1     1      2
## 6   6     1      1
## 11 11     1      3
## 2   2     2      3
## 7   7     2      1
## 12 12     2      2
## 3   3     3      1
## 8   8     3      2
## 13 13     3      3
## 4   4     4      3
## 9   9     4      2
## 14 14     4      1
## 5   5     5      3
## 10 10     5      1
## 15 15     5      2
```

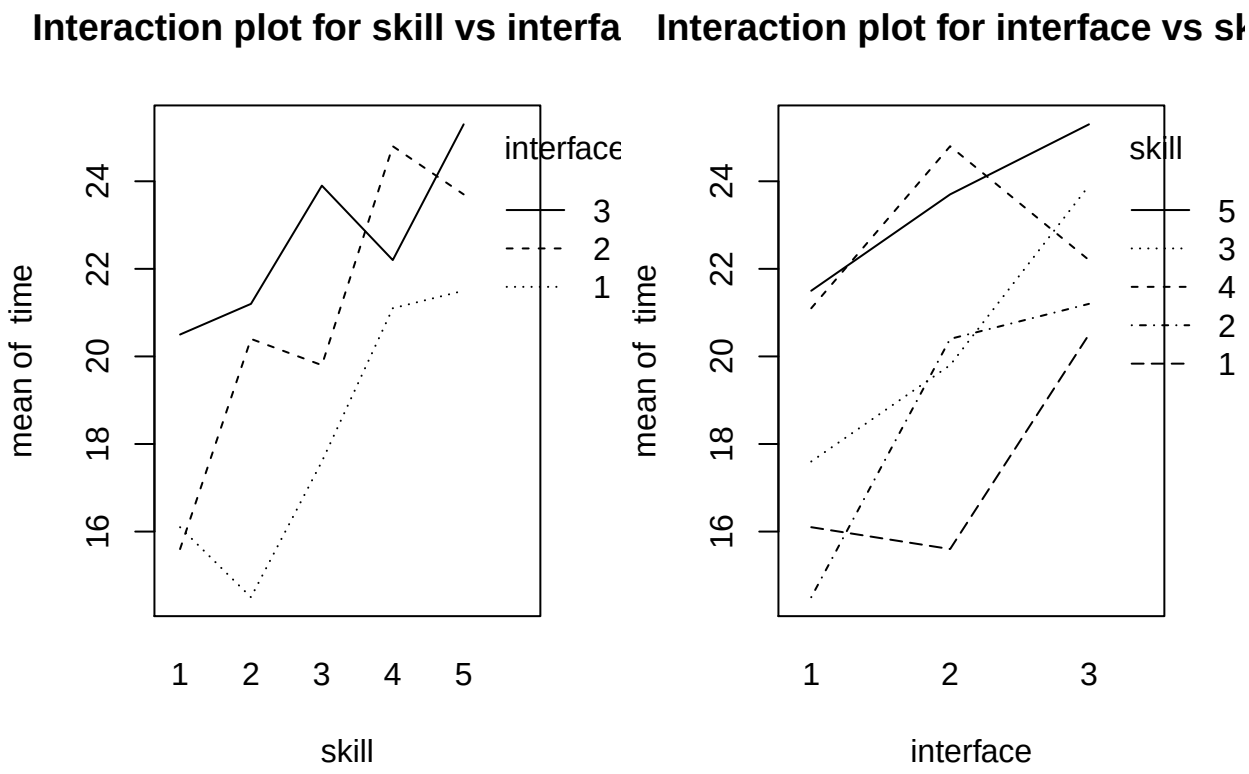
In the former table each row corresponds to one block, in other words a group with the same type of students (similar skills). Then inside these blocks we see the 3 different interface designs (treatments). In the latter table the randomization technique is applied to the 15 students. This table shows the id of the student together with its skills (which of course is fixed, since this is given and we cannot modify this) and finally which treatment the student should get.

B)

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

boxplot(time~skill, main="Time taken to find product per skill level", ylab="Time (s)", xlab="Skill leve
boxplot(time~interface, main="Time taken to find product per interface", ylab="Time (s)", xlab="Interfa
```




```
interaction.plot(skill, interface, time, main="Interaction plot for skill vs interface")
interaction.plot(interface, skill, time, main="Interaction plot for interface vs skill")
```




```
skill = as.factor(skill)
interface = as.factor(interface)
searchaov = lm(time~skill+interface)
anova(searchaov)
```

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

```
summary(searchaov)
```

```
##
## Call:
## lm(formula = time ~ skill + interface)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5733 -0.6967  0.3867  1.0567  1.7867
##
```

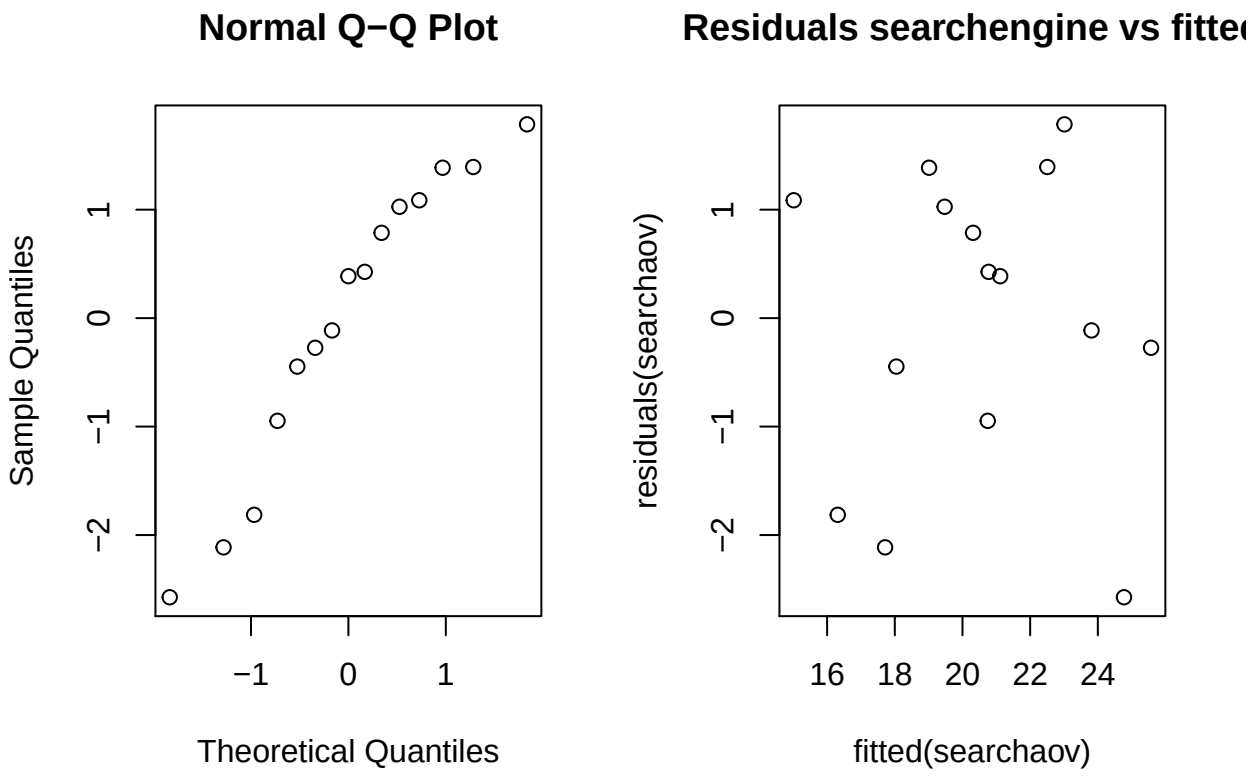
```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   15.013      1.227  12.238 1.85e-06 ***
## skill2         1.300      1.466   0.887  0.40118
## skill3         3.033      1.466   2.069  0.07238 .
## skill4         5.300      1.466   3.614  0.00684 **
## skill5         6.100      1.466   4.160  0.00316 **
## interface2     2.700      1.136   2.377  0.04474 *
## interface3     4.460      1.136   3.927  0.00438 **
## ---
## 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
```

When looking at the boxplots we see all boxes have similar variances and we see strong trends in both plots. The mean of skill level 2 stands out but is of no immediate concern. When looking at the interaction plots we see all lines are somewhat parallel, with some intersections present between the lines. Since they are parallel however we assume there is no interaction effect, and the intersections are caused by noisy data.

The analysis of the Variance Table shows us that the p-value for interface is below 0.05, meaning that it significantly influences the time it takes for students to find a certain product. So, the interface effects are significantly different from 0. The interface that requires the longest search time is interface3 as becomes clear from the summary of the anova (the time for finding an article is estimated 4.46 seconds longer than for interface1). The intercept shows the estimated search time for interface1 and skill1. All other estimates are positive meaning that students require more time finding a product for all other skill levels and interfaces. Therefore the combination skill level 1 and interface 1 give the shortest searching time. The time it takes a typical user of skill level 3 to find the product on the website if the website uses interface 3 is: the intercept + estimate skill level 3 + interface level 3 = 15.01 + 4.46 + 3.03 = 22.50 seconds.

C)

```
par(mfrow=c(1,2))
qqnorm(residuals(searchaov));
plot(fitted(searchaov),residuals(searchaov), main= "Residuals searchengine vs fitted")
```



The points from the data set form a roughly straight line in the QQ-plot and the spread of the residuals vs the fitted values seems homogenous. Therefore the data seems to be normally distributed and having equal population variances.

D)

```
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 Friedman test shows there is a significant treatment effect for the interface level since the p-value is below the alpha value 0.05.

E)

```
aovsearch_ow = lm(time~interface)
anova(aovsearch_ow)
```

```
## 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
```

The one way Anova test provides no evidence to reject the null hypothesis that the effect of the interface is zero. In other words the type of the interface is not significantly influencing the time for finding a product. In contrast with the two way Anova (where both factors were significant), now the factor interface on it's own is not significant. The difference is that for the one way test the factor skill is not controlled, while in the two way Anova this is the case. The one way anova test is not wrong since you're still validly checking whether the interface influences the search time. However the outcomes will predict less strong than the two way anova with a controlled factor.

Although this result gives us less information about the relevance of the interface than the two-way Anova, it does give us insight in how important it is to correctly account for blocks if these are evident. So while it is not wrong (but incomplete) it is useful for our understanding as compared to our two-way Anova.

```
detach(searchengine)
```

Exercise 3: Feedingstuffs for cows

```
cow <- read.table("cow.txt", header=TRUE)
attach(cow)
```

```
## The following object is masked _by_ .GlobalEnv:
##
##     id
```

```
per = as.factor(per); id = as.factor(cow$id)
```

A)

```
cowlm=lm(milk~id+per+treatment)
anova(cowlm)
```

```
## Analysis of Variance Table
##
## Response: milk
##           Df  Sum Sq Mean Sq  F value    Pr(>F)
## id         8 2467.47 308.434 124.4832 7.494e-07 ***
## per        1   24.50  24.500   9.8881   0.01628 *
## treatment  1    1.16   1.156   0.4666   0.51654
## Residuals  7   17.34   2.478
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(cowlm)
```

```
##
## Call:
## lm(formula = milk ~ id + per + treatment)
##
## 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 ***
## 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 ***
## per2         -2.3900     0.7466  -3.201 0.015046 *
## treatmentB   -0.5100     0.7466  -0.683 0.516536
## ---
## 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
```

The repeated measures fixed effect model shows no significant difference in milk production for the different treatments. The estimated difference in milkproduction between the two feedingstuffs is -0.510 meaning that the milkproduction for treatmentA is estimated 0.510 higher than treatmentB. It seems there is an effect of the period that the cow takes it's feedingstuff, since the p-value for per is below alpha (0.05). Also the Id of the cow is of significant influence in the milk production, which makes sense since not every cow produces the same amount of milk.

B)

```
library(lme4)
```

```
## Loading required package: Matrix
```

```
cowlmer=lmer(milk~treatment+order+per+(1|id), REML=FALSE)
cowlmer1=lmer(milk~order+per+(1|id), REML=FALSE)
anova(cowlmer1, cowlmer)
```

```
## Data: NULL
## Models:
## cowlmer1: milk ~ order + per + (1 | id)
```

```
## cowlmer: milk ~ treatment + order + per + (1 | id)
##          npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## cowlmer1    5 117.89 122.34 -53.946   107.89
## cowlmer     6 119.31 124.65 -53.656   107.31 0.5807  1      0.446
```

The code above gives the implementation of the cross-over design with the cow id's as random effects. The p-value for finding whether the treatment influences the milk production is found by refitting the model without the treatment. The comparison of the models show there is no significant effect that the models are different from each other. Thus the treatment is not influencing the milk production significantly (the p-value is higher than alpha). This leads to the same conclusion as for question a) (where the fixed effect model also did not find a significant influence for the treatment).

C)

```
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
```

The outcome of the paired t-test shows the treatment does not influence the milk production significantly. This paired t-test assumes there is no sequence/learning effect taking place. However the test setup cannot ensure that this learning effect did not happen (although the experiment tried to wash out the carry over effects). It could for example be the case that one type of food has a long term effect in producing more or less milk (which will than also be noted in the milk production measure in the second period). The conclusion for this paired t-test aligns with the conclusion in the two earlier designs. The conclusions are compatible because the order in which the treatments are dispensed is not decisive.

```
detach(cow)
```

Exercise 4: Jane Austen

```
austen <- read.table("austen.txt", header=TRUE)
attach(austen)
```

A)

Since we are looking for evidence here of whether someone has skillfully imitated Jane Austen's writing style, a contingency table test for homogeneity is most appropriate. Were we to test for independence, the null hypothesis would be that the row variable and column variable are independent, whereas, assuming Jane Austen has her own writing style, we assume that the rows are dependent upon at least the first 3 columns. When testing for homogeneity we can test whether the distributions over columns / rows are equal, which will help us with comparing the content of the fourth column to the content of the first 3.

B)

```
janeausten <- austen[,1:3] # only leave columns from real austen
ja <- chisq.test(janeausten)
ja
```

```
##
##  Pearson's Chi-squared test
##
## data:  janeausten
## X-squared = 12.271, df = 10, p-value = 0.2673
```

```
ja$expected
```

```
##             Sense      Emma     Sand1
## a       160.02950 187.76794 86.202557
## an       22.86136  26.82399 12.314651
## this     31.71091  37.20747 17.081613
## that     87.02065 102.10423 46.875123
## with     59.36578  69.65585 31.978368
## without  14.01180  16.44051  7.547689
```

```
residuals(ja)
```

```
##               Sense       Emma      Sand1
## a       -1.02997736 -0.1290203  1.5937736
## an       0.44728806 -0.1590968 -0.3746273
## this     0.05133600  0.2938669 -0.5036577
## that     0.74817619  0.2865778 -1.4423521
## with    -0.04747379  0.5205063 -0.7035205
## without  1.06544255 -1.5884103  0.8926239
```

```
style <- rowSums(abs(ja$residuals)); style
```

```
##         a        an      this      that      with   without
## 2.7527712 0.9810121 0.8488606 2.4771061 1.2715006 3.5464767
```

When looking at the expected values it definitely looks like 80% of them are over 5, so we can use the chi-squared test. We can see that the p-value suggests there is no reason to assume that the word occurrence depends on the different chapters. Therefore it looks like Austen did a pretty good job in writing stories using a consistent style in word use. The main inconsistencies are in the use of the word "a" in Sand1, where it occurs relatively more often in comparison with other stories. Another inconsistency appears for the word "without" in Emma where it relatively occurs less than in the other chapters. Finally there is an inconsistency in the use of the word "that" in Sand1 that is used relatively less in these chapters. When looking at the summed absolute values of the residuals we see that the inconsistencies (deviations) are strongest firstly for the word 'without', secondly for the word 'a', thirdly for the word 'that', followed by 'with', 'an' and 'this'.

C)

```
aus <- chisq.test(austen)
aus
```

```
##
##  Pearson's Chi-squared test
##
## data:  austen
## X-squared = 45.578, df = 15, p-value = 6.205e-05
```

```
residuals(aus)
```

```
##              Sense          Emma      Sand1       Sand2
## a       -1.0149156 -0.1120927868  1.6062866 -0.05889921
## an      -0.5906319 -1.2199545912 -1.0671306  3.72816398
## this     0.1388299  0.3904903154 -0.4436450 -0.32671736
## that     1.5943613  1.1798488360 -0.9099606 -3.04931581
## with    -0.5120944  0.0001916718 -1.0246069  1.74821745
## without  1.3919336 -1.3411962838  1.1365432 -1.06963011
```

```
fakeaus <- aus$residuals[,4]
avgausten <- rowSums(ja$residuals)/3;difference = fakeaus-avgausten

fakeaus # residuals of the admirer
```

```
##           a          an        this        that        with     without
## -0.05889921  3.72816398 -0.32671736 -3.04931581  1.74821745 -1.06963011
```

```
avgausten  # averaged residuals of Austen
```

```
##           a          an        this        that        with     without
##  0.14492530 -0.02881200 -0.05281828 -0.13586606 -0.07682935  0.12321871
```

```
difference # difference in residuals between austen and the admirer
```

```
##          a         an       this       that       with    without
## -0.2038245  3.7569760 -0.2738991 -2.9134497  1.8250468 -1.1928488
```

Adding Sand2 (the non-original text) to the data results in a different outcome of the chi-square test. Now the two factors (word use & different chapters) seem to depend significantly on one another, with a p-value of 6.205e-07. The main inconsistencies are in the use of the words "an" and "that" for Sand 2. "an" occurred significantly more often in the story written by the admirer while "that" occurred significantly less.

When comparing the residuals of the admirer with the average residuals of Austen, we can see that the inconsistencies are in descending order: "an", "that", "with", "without", "this" and "a".

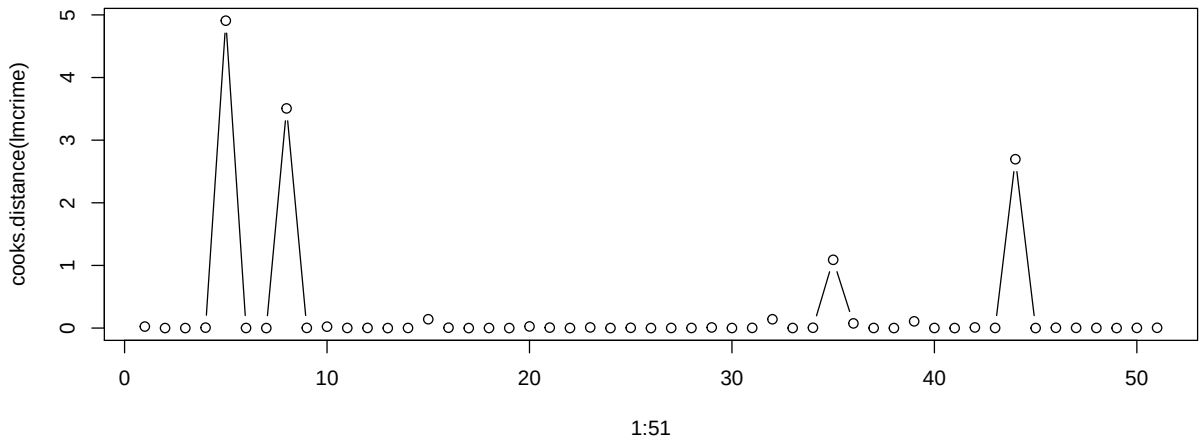
```
detach(austen)
```

Exercise 5: Expenditure on Criminal Activities

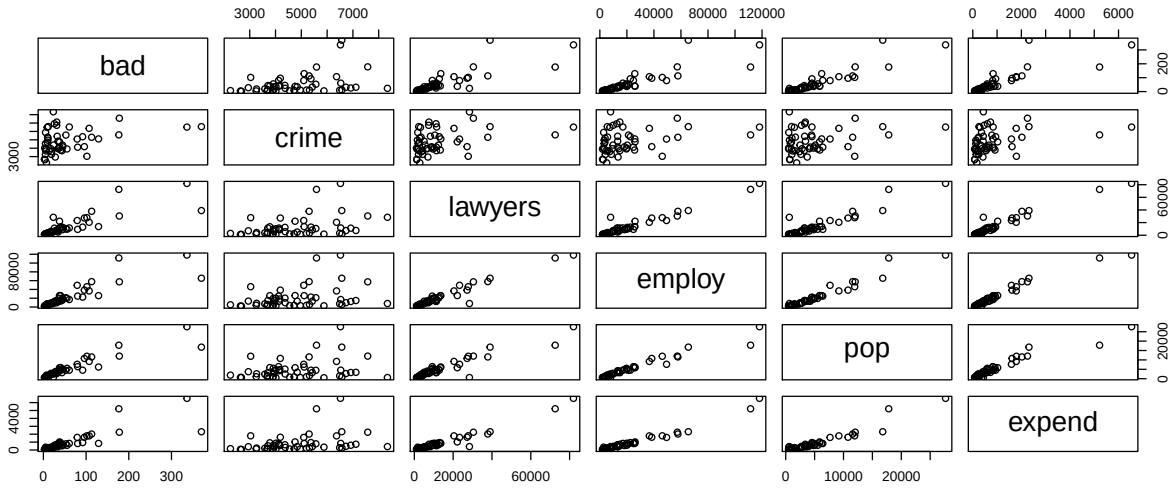
```
expensescrime <- read.table("expensescrime.txt", header = TRUE)
attach(expensescrime)
```

A) <> Roel: 6 zinnen

```
lmcrime <- lm(expend~bad+crime+lawyers+employ+pop,data=expensescrime)
plot(1:51,cooks.distance(lmcrime),type="b")
```




```
pairs(expensescrime[,c(3:7,2)])
```




```
round(cor(expensescrime[,c(3:7,2)]),2)
```

```
##          bad crime lawyers employ  pop expend
## bad     1.00  0.37    0.83   0.87 0.92   0.83
## crime   0.37  1.00    0.38   0.31 0.28   0.33
## lawyers 0.83  0.38    1.00   0.97 0.93   0.97
## employ  0.87  0.31    0.97   1.00 0.97   0.98
## pop     0.92  0.28    0.93   0.97 1.00   0.95
## expend  0.83  0.33    0.97   0.98 0.95   1.00
```

A potential point is an observation with an outlying value in an explanatory variable X_i. First off, we'll take a look at the Cook's distance for the dataframe, which is shown in the second figure. Here we find that 4 different rows of the data set (states) seem to be influence points, as Cook's distance for these is higher than (or around) 1. These rows could potentially be deleted to make the data more consistent.

We also check the scatterplot of all potential explanatory variables against each other, where we also find some potential points that could be looked at. An example would be the lawyers~bad graph with a few potential points, as well as bad~employ. If the estimated parameters change drastically by deleting the potential point, the observation is called an influence point. If the plot of two explanatory variables shows (nearly) a straight line, the two variables are collinear. All plots that have "crime" as one of the two explanatory variables are not collinear, while all other plots seem to be collinear. These collinear plots mean that the two variables explain the same influence on the outcome. We should most likely choose a model with a smaller number of explanatory variables in this particular case.

B)

First the step-up method is considered with expend as the response variable.

```
# summary(lm(expend~bad,data=expensescrime))[c(4,8)]
# summary(lm(expend~crime,data=expensescrime))[c(4,8)]
# summary(lm(expend~lawyers,data=expensescrime))[c(4,8)]
# summary(lm(expend~employ,data=expensescrime))[c(4,8)]
# summary(lm(expend~pop,data=expensescrime))[c(4,8)]
```

Then, all the possible explanatory variables are calculated as an R^2 value. The variable with the highest R^2 value is "employ" with 0.954 (which also has a p-value lower than alpha), so this one is used for the next step.

```
# summary(lm(expend~employ+bad,data=expensescrime))[c(4,8)]
# summary(lm(expend~employ+crime,data=expensescrime))[c(4,8)]
# summary(lm(expend~employ+lawyers,data=expensescrime))[c(4,8)]
# summary(lm(expend~employ+pop,data=expensescrime))[c(4,8)]
```

When adding more estimate variables, we can already see that these yield an increase in the R^2 value, but only by an extremely small amount. The highest value now is "lawyers" with a value of 0.963, which is only 0.009 higher than only using "employ" as an explanatory variable. Therefore, we should stop at the previous step, and only employ is an explanatory variable for the step-up method. The resulting model of the step-up method is: expend = -116.7052 + 0.0468*employ + error

```
# summary(lm(expend~bad+crime+lawyers+employ+pop),data=expensescrime)[c(4,8)]
```

Here we start the step-down method. We see that crime is the only variable with a p-value higher than alpha, which also indicates that it should be deleted.

```
# summary(lm(expend~bad+lawyers+employ+pop),data=expensescrime)[c(4,8)]
```

When looking at the data with crime removed, we then see that there is another variable with a p-value that is higher than the alpha of 0.05: the variable "bad." This variable is deleted next.

```
# summary(lm(expend~lawyers+employ+pop),data=expensescrime)[c(4,8)]
```

This time, we should remove "pop," which has a p-value much higher than the alpha of 0.05.

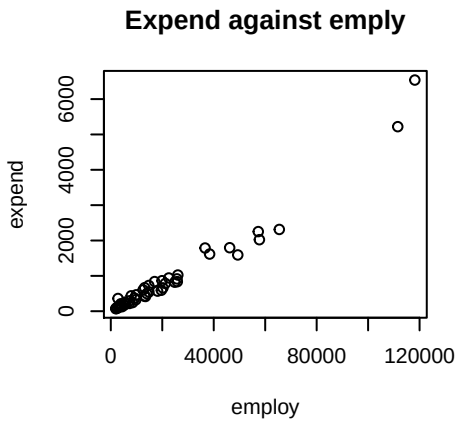
```
# summary(lm(expend~lawyers+employ),data=expensescrime)[c(4,8)]
```

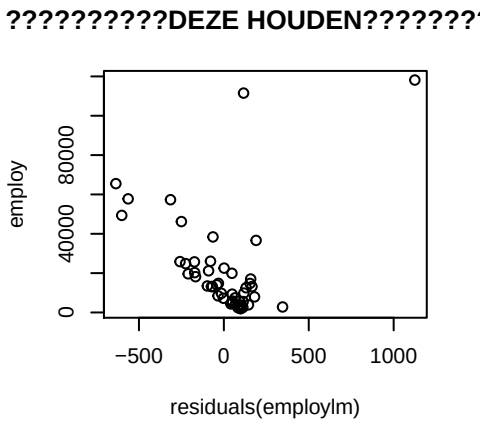
Now all p-values are below alpha, meaning that we don't need to delete any more explanatory variables any more. The resulting model of the step-down method is: expand =`{r} -110.6588+0.0269*lawyers+0.0297*employ+error`

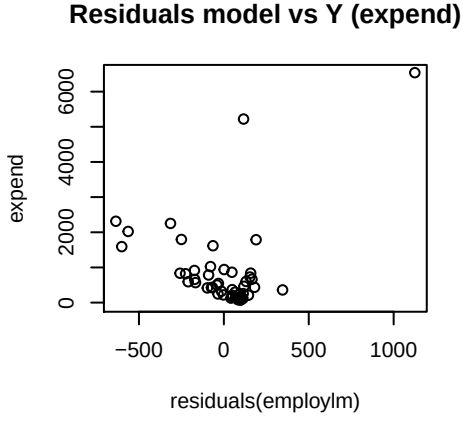
The models for step-up and step-down are not the same in this case, meaning that we have to pick one which is the better fit. Because the R^2 value for step-up is 0.954 with just 1 variable, and the R^2 value for step-down is 0.963 with 2 variables, the step-up model is preferred, since the difference is too small to really make much of a difference in estimating the expend, meaning that we base our decision on the smallest number of variables.

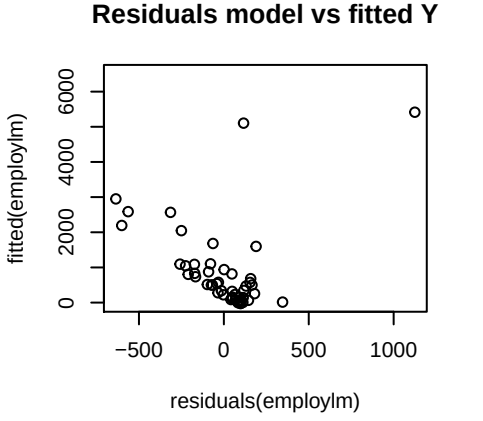
C) <> ROEL

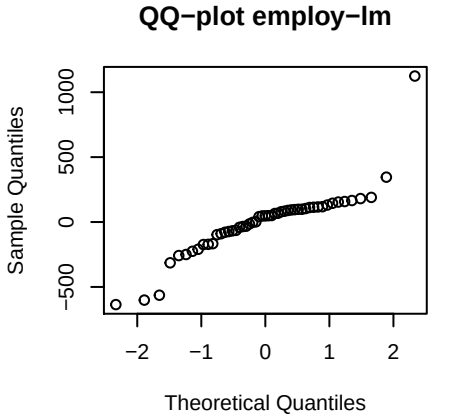
```
par(mfrow=c(3,2))
# Step-up model:
# Since we only have one explanatory variable we also do not need to check VIF-values
employlm = lm(expend~employ, data=expensescrime)

plot(employ, expend, main="Expend against emply")
plot(residuals(employlm),employ, main="??????????DEZE HOUDEN??????????")

plot(residuals(employlm),expend, main="Residuals model vs Y (expend)", ylim=c(0,6500))
plot(residuals(employlm),fitted(employlm), main="Residuals model vs fitted Y", ylim=c(0,6500))
qqnorm(residuals(employlm),main="QQ-plot employ-lm")
```











ROEL: plot waar 'DEZE HOUDEN' bij staat weet ik niet goed mee wat we er mee moeten

The model assumptions are: the linearity of the relation and the normality of the errors. The linearity of the relation can be tested by plotting Y (expend) against X (employ), as we only have one explanatory variable in our model. In the first graph above we indeed see a linear pattern in the data.

To check the normality we can compare the residuals plotted against Y (expend) and fitted Y (outcome of the model), as well as the QQ-plot of the residuals. When looking at the residuals vs Y and residuals vs fitted Y we see that the spread is quite similar, although the data points seem to be drawn slightly towards

each other (towards the line y=3000). Given that the spread is similar (and the values close to) we can assume the model is a good predictor for expend. When looking at the qq-plot however we see the data is far from normal. We can therefore not assume normality.