Assignment 3

Sara Salimzadeh,

Tung Nguyên Hũu Tùng,

Felipe Santos Batista de Souza

Group 18

March 12

library(multcomp)

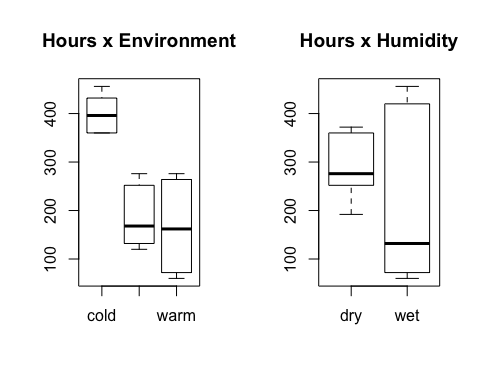
library(lme4)

### Exercise 1

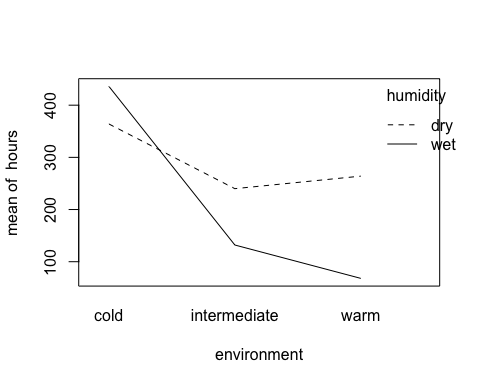
data\_bread <- read.table(file = "bread.txt", header = TRUE)  
  
# QUestion 1:  
I = 3 #levels of temperature.  
J = 2 #levels of humidity.  
N = 3 #experimental units per combination of the two factors, given that the total of units is 18.  
rbind(rep(1:I,each=N\*J),rep(1:J,N\*I),sample(1:(N\*I\*J)))

## [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]  
## [1,] 1 1 1 1 1 1 2 2 2 2 2 2 3  
## [2,] 1 2 1 2 1 2 1 2 1 2 1 2 1  
## [3,] 17 5 9 13 15 1 3 8 11 14 7 2 18  
## [,14] [,15] [,16] [,17] [,18]  
## [1,] 3 3 3 3 3  
## [2,] 2 1 2 1 2  
## [3,] 12 16 4 6 10

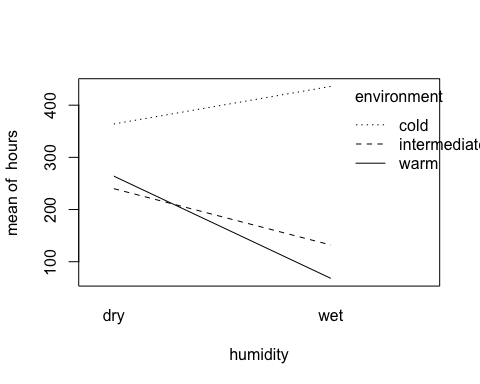
# QUestion 2:  
par(mfrow=c(1,2))



par(mfrow=c(1,1))  
attach(data\_bread)  
interaction.plot(environment,humidity,hours)



interaction.plot(humidity,environment,hours)



# QUestion 3:  
data\_bread$environment=as.factor(data\_bread$environment)  
data\_bread$humidity=as.factor(data\_bread$humidity)  
breadaov=lm(hours~environment\*humidity,data=data\_bread)

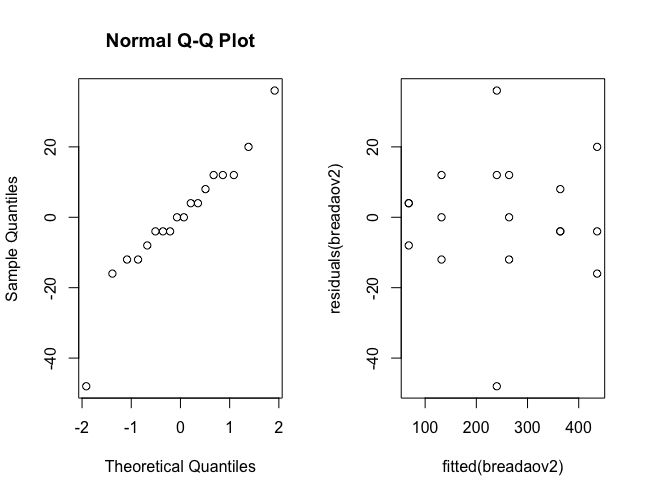
anova (breadaov)  
summary(breadaov)

## lm(formula = hours ~ environment \* humidity, data = data\_bread)  
## 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  
##   
## (Intercept) \*\*\*  
## environmentintermediate \*\*\*  
## environmentwarm \*\*\*  
## humiditywet \*\*   
## environmentintermediate:humiditywet \*\*\*  
## environmentwarm:humiditywet \*\*\*

# QUestion 4:  
contrasts(data\_bread$environment)=contr.sum  
contrasts(data\_bread$humidity)=contr.sum  
breadaov2=lm(hours~environment\*humidity,data=data\_bread)  
anova (breadaov2)  
summary(breadaov2)

## lm(formula = hours ~ environment \* humidity, data = data\_bread)  
## 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 \*

# QUestion 5:



# An extra check is also the Shapiro test.  
shapiro.test(residuals(breadaov2))

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(breadaov2)  
## W = 0.9296, p-value = 0.1911

**Exercise 1**

1. The resulting matrix shows a suggested random assignment for the different experiment units, for example:

elements of the matrix: (1,2) =1, (2,2)=2, (3,2)=7 ==> the suggestion would be then to assign the unit 7 to temperature level 1 and humidity level 2.

1. It is interesting to see through the boxplots that under certain levels for the factors (environment and temperature) there is bigger dispersion on the sampled values for the hours variable, for example when the environment is warm or when the humidity is wet.

Now looking at the interaction plots it’s possible to see that the lines are non-parallel both when humidity or environment are fixed. This can indicate interaction between these factors.

1. Conclusion:

Considering the decomposition of the population mean as:

U (i, j) = u + a (i) + b (i) + y (i, j)

As per the results from ANOVA:

The p-value testing H0: a (i) for all i is equal to 0 is 2.461e-10 (reject H0)

The p-value testing H0: b (i) for all i is equal to 0 is 4.316e-06 (reject H0)

The p-value testing H0: y (i, j) for all ( i, j) is equal to 0 is 3.705e-07 (reject H0)

Thus, it’s possible to say that both factors have a main effect given that the null hypothesis for both of them was rejected (so they are a factor <> 0) in the equation mentioned above.

The fact that H0 is also rejected for y (i, j) means that there might be interaction between both factors.

As could be seen in the interaction plots from the item 1.2 there is an indication of interaction by the existence of non-parallel lines.

1. As can be seen on the estimations for the factors (environment1: 149.333 // environment2: -64.667), the environment has the biggest numerical influence over the time decay.

This can be a good question if it is important to know which factor could cause bigger changes in the outcome/result that is been measured.

However, is also important to notice that, as shown on the results above, that there is an interaction factor between the two factors. So, it may happen that the combination of them could lead to more significant changes in the result.

1. *Checking the normality of the population.*

The residuals seem to follow a normal distribution, there are some extreme values outliers) though.

An extra check is also the Shapiro test. In this case, we also fail to reject the null hypothesis (that the sample) follows a normal distribution (p-value = 0.1911).

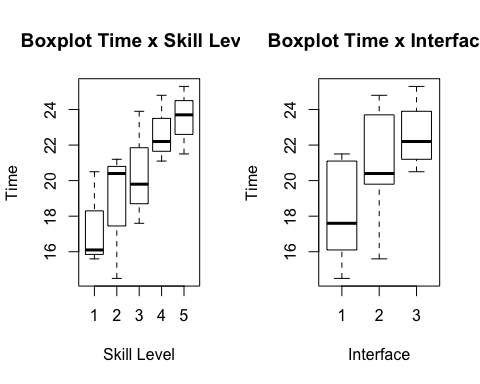
*Checking the assumption of equal population variances.*

As can be seen, there are two data-points that are extreme.

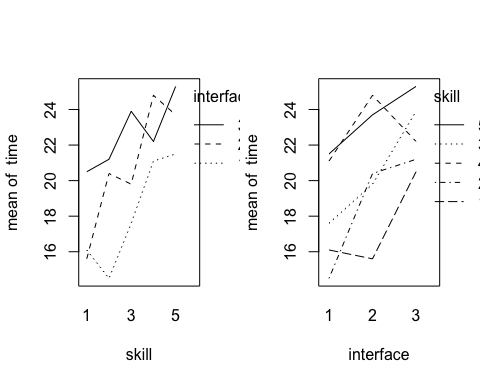
However, in general the residuals remain constant for different fitted values which is in line with the assumption of "of equal population variances".

### Exercise 2  
search = read.table("search.txt", header = TRUE)  
  
# Question 1  
B = 5  
I = 3  
N = 1  
  
rbind(rep(1:I,each=N\*B),rep(1:B,N\*I),sample(1:(N\*I\*B)))

## [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]  
## [1,] 1 1 1 1 1 2 2 2 2 2 3 3 3  
## [2,] 1 2 3 4 5 1 2 3 4 5 1 2 3  
## [3,] 9 13 11 7 10 12 1 2 3 8 15 14 4  
## [,14] [,15]  
## [1,] 3 3  
## [2,] 4 5  
## [3,] 5 6  
  
# Question 2  
attach(search)  
  
par(mfrow=c(1,2))



par(mfrow=c(1,2))  
interaction.plot(skill, interface, time); interaction.plot(interface, skill, time)



# Question 3  
search$skill <- factor(search$skill)  
search$interface <- factor(search$interface)  
aovsearch = lm(time~interface+skill, data = search)  
anova(aovsearch)

## 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

summary(aovsearch)

## lm(formula = time ~ interface + skill, data = search)  
##   
## 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 \*\*   
## F-statistic: 6.745 on 6 and 8 DF, p-value: 0.008395

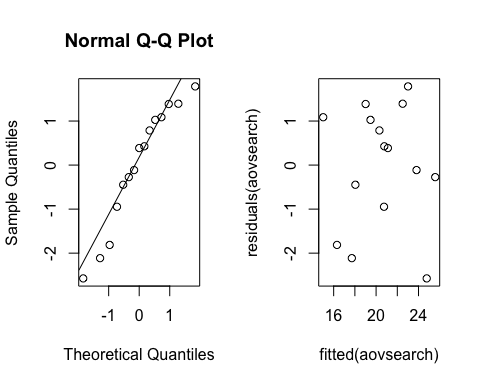
# Question 4  
contrasts(search$skill)=contr.sum  
contrasts(search$interface)=contr.sum  
aovsearch = lm(time~interface+skill, data = search)  
summary(aovsearch)

## lm(formula = time ~ interface + skill, data = search)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 20.5467 0.4637 44.311 7.43e-11 \*\*\*  
## interface1 -2.3867 0.6558 -3.640 0.00659 \*\*   
## interface2 0.3133 0.6558 0.478 0.64556   
## skill1 -3.1467 0.9274 -3.393 0.00946 \*\*   
## skill2 -1.8467 0.9274 -1.991 0.08161 .   
## skill3 -0.1133 0.9274 -0.122 0.90575   
## skill4 2.1533 0.9274 2.322 0.04877 \*   
## F-statistic: 6.745 on 6 and 8 DF, p-value: 0.008395

estimatedTime = 20.5467 + 2.1533 + (0 - (-2.3867) - 0.3133)   
estimatedTime

## [1] 24.7734

# Question 5:  
par(mfrow=c(1,2))



# Question 6  
friedman.test(time, interface, skill, data = search)

## Friedman rank sum test  
##   
## data: time, interface and skill  
## Friedman chi-squared = 6.4, df = 2, p-value = 0.04076  
  
# Question 7:  
oneaovsearch = lm(time~interface, data = search)  
anova(oneaovsearch)

## 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

**Exercise 2**

1. As seen from randomized table, unit 3 will use levels (I=1, B=1), unit 7 will use levels (I=1, B=2), …, and unit 9 will use levels (I=3, B= 5).
2. Given the boxplot of time corresponding with skill time alone, we can observe that different levels of skill certainly affect the time. The lower the skill is (higher indicator), the more time is spent. Glancing at the boxplot of time with interface, it is fairly clear that time has large interval when using interface 3.

The interaction plots show unparalleled lines so we can suspect the interaction between two factors: interface and skill.

1. By applying anova test to our data, we get the p-value for interface factor equals 0.013, which is less than 0.05. It means that we can reject the null hypothesis that the search time is the same for all interfaces.
2. Using the summary table, we can estimate that it takes 24.773 for a typical user of skill level 4 to find the product on the website if the website uses interface 3.
3. Hereby, the data might approximately follow the normal distribution even though there is a slight curve in the QQ-Plot. The plot nearby includes points which are all over the place. In other words, the plot can be considered diagnostic.
4. The Friedman test has the p-value for testing null hypothesis of “no interface effect” is 0.04076 and therefore we can reject the null hypothesis that there is significant effect of interface.
5. The p-value of one-way anova is 0.09642, which means we cannot reject the null hypothesis that there is no interface effect. This test is only valid if we ignore the skill variable as we assumed before. However, it is too wise to apply this test because the data collected is always influenced by the skill factor. To be valid, the test should be taken by the way that skill factor is isolated, which means we should choose experiment units that have the same skill.

### Exercise 3  
  
# Question 1  
cream= read.table("cream.txt",header = TRUE)  
  
cream$batch= as.factor(cream$batch)  
cream$position= as.factor(cream$position)  
cream$starter= as.factor(cream$starter)  
  
creamaov= lm(acidity ~ starter + batch + position,data = cream)  
summary(creamaov)

## lm(formula = acidity ~ starter + batch + position, data = cream)  
## 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   
## F-statistic: 9.96 on 12 and 12 DF, p-value: 0.0001777

# Question 2  
creammult= glht(creamaov,linfct = mcp(starter="Tukey"))  
summary(creammult)

## Simultaneous Tests for General Linear Hypotheses  
## Multiple Comparisons of Means: Tukey Contrasts  
## Fit: lm(formula = acidity ~ starter + batch + position, data = cream)  
##   
## Linear Hypotheses:  
## Estimate Std. Error t value Pr(>|t|)   
## 2 - 1 == 0 -0.1500 0.4673 -0.321 0.997367   
## 3 - 1 == 0 -0.9800 0.4673 -2.097 0.282005   
## 4 - 1 == 0 2.8100 0.4673 6.013 0.000497 \*\*\*  
## 5 - 1 == 0 -0.4840 0.4673 -1.036 0.834341   
## 3 - 2 == 0 -0.8300 0.4673 -1.776 0.428894   
## 4 - 2 == 0 2.9600 0.4673 6.334 0.000290 \*\*\*  
## 5 - 2 == 0 -0.3340 0.4673 -0.715 0.949081   
## 4 - 3 == 0 3.7900 0.4673 8.110 < 1e-04 \*\*\*  
## 5 - 3 == 0 0.4960 0.4673 1.061 0.822247   
## 5 - 4 == 0 -3.2940 0.4673 -7.048 0.000109 \*\*\*  
  
  
# Question 4  
confint(creammult)

## Simultaneous Confidence Intervals  
## Multiple Comparisons of Means: Tukey Contrasts  
## Fit: lm(formula = acidity ~ starter + batch + position, data = cream)  
## Linear Hypotheses:  
## Estimate lwr upr   
## 2 - 1 == 0 -0.1500 -1.6401 1.3401  
## 3 - 1 == 0 -0.9800 -2.4701 0.5101  
## 4 - 1 == 0 2.8100 1.3199 4.3001  
## 5 - 1 == 0 -0.4840 -1.9741 1.0061  
## 3 - 2 == 0 -0.8300 -2.3201 0.6601  
## 4 - 2 == 0 2.9600 1.4699 4.4501  
## 5 - 2 == 0 -0.3340 -1.8241 1.1561  
## 4 - 3 == 0 3.7900 2.2999 5.2801  
## 5 - 3 == 0 0.4960 -0.9941 1.9861  
## 5 - 4 == 0 -3.2940 -4.7841 -1.8039

**Exercise 3**

1. The p-values produced with anova and summary commands are not simultaneous. The p-values in the lines starter2 to starter5 are for the hypothesis , and so on for the main effect of starter. This is the same for batch and position. According to the summary, starter4 has the least p-value equals to 6.10e-05. Therefore, we can conclude that the null hypothesis is strongly rejected and starter4 plays a significant role in the main model of the acidity. Similarly, bath2 and batch4 null hypothesis can be rejected as well with p-values equal to 0.0137 and 0.0127 respectively. The abstract model is: . To sum up, equals to for i = 2,3,5 (except ). For j = 2 and 4, is similar to (excluding . Finally, all in range (2,5) is the same as .
2. Here the table is based on simultaneous p-values for the null hypothesis. As it is clear from the table, starter4 has the most difference with others and significantly affects the acidity. P-values for cases, , , and, are less than 0.001 hence null hypothesis are rejected.
3. In simultaneous comparisons, the more inferences are made, the more likely erroneous inferences are to occur. Consequently, the p-value in part 2 is less trusted than in part 1 because of the error.
4. The intervals for , ,and, don’t contain 0. As we concluded in part 2, the starter4 lead to significantly different acidity. So, the confidence intervals shouldn’t include 0.

### Exercise 4  
  
# Question 1  
cow = read.table("cow.txt", header = TRUE)  
cow$id = factor(cow$id)  
cow$per = factor(cow$per)  
cow$treatment = factor(cow$treatment)  
  
cowlm = lm(milk~treatment+id+per, data = cow)  
summary(cowlm)

## lm(formula = milk ~ treatment + id + per, data = cow)  
## 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   
## 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 \*   
## F-statistic: 100.6 on 10 and 7 DF, p-value: 1.349e-06

# Question 2 Result from question 1  
  
# Question 3  
cowlmer = lmer(milk~treatment+order+per+(1|id), data = cow, REML=FALSE)  
summary(cowlmer)

## Linear mixed model fit by maximum likelihood ['lmerMod']  
## Formula: milk ~ treatment + order + per + (1 | id)  
## Data: cow  
##   
## 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

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

## Data: cow  
## Models:  
## cowlmer1: milk ~ order + per + (1 | id)  
## cowlmer: milk ~ treatment + order + per + (1 | id)  
## Df AIC BIC logLik deviance Chisq Chi 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

# Question 4  
attach(cow)  
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  
## sample estimates:  
## mean of the differences   
## 0.2444444

**Exercise 4**

1. The effect of type of feedingstuffs is not significant with the p-value = 0.516 > 0.05. In other words, we cannot reject the null hypothesis of “no effect of type of feeding stuffs”
2. TreamentA is estimated to produce 0.51 more milk than TreatmentB
3. With p-value = 0.446 > 0.05 from anova test, we can draw the same conclusion as the first section that we cannot reject the null hypothesis that there is no effect of type of feeding stuffs on milk production.
4. The pair t-test is invalid since it only tests whether there is difference between the true mean of the paired samples due to 2 different treatments. Nonetheless, we cannot ignore fixed period effect and fixed sequence effect in our current experiment, which in turn leads to the less accuracy of paired-test.

The conclusion in t-test, which does not reject the null hypothesis of no effect of Treatment), is fairly compatible with the test in the question 1 (p-value equals 0.516536 and 0.8281 respectively). However, as mentioned before the pair t-test lacks of estimating other effects except “Treatment), we cannot trust the result from this test.

### Exercise 5  
  
# Question 1  
nausea.frame=data.frame("nausea" = integer(),"medicin" = character(), stringsAsFactors = FALSE)  
index = 1  
for(i in 1:100){  
 nausea.frame[i,] <- rbind(0, "Chlorpromazine")  
 index = index + 1}  
  
for(i in 1:52){  
 nausea.frame[index,] <- rbind(1, "Chlorpromazine")  
 index = index + 1}  
  
for(i in 1:32){  
 nausea.frame[index,] <- rbind(0, "Pentobarbital(100mg)")  
 index = index + 1}  
  
for(i in 1:35){  
 nausea.frame[index,] <- rbind(1, "Pentobarbital(100mg)")  
 index = index + 1}  
  
for(i in 1:48){  
 nausea.frame[index,] <- rbind(0, "Pentobarbital(150mg)")  
 index = index + 1}  
  
for(i in 1:37){  
 nausea.frame[index,] <- rbind(1, "Pentobarbital(150mg)")  
 index = index + 1}  
attach(nausea.frame)  
# Question 2  
nausea.frame$medicin=as.factor(nausea.frame$medicin)  
  
xtabs(~medicin+nausea)

## nausea  
## medicin 0 1  
## Chlorpromazine 100 52  
## Pentobarbital(100mg) 32 35  
## Pentobarbital(150mg) 48 37

# Question 3  
attach(nausea.frame)

t = chisq.test(xtabs(~medicin+nausea))[[1]]  
B=1000  
tstar=numeric(B)  
for (i in 1:B){  
 medicinstar=sample(medicin)  
 tstar[i]= chisq.test(xtabs(~medicinstar+nausea))[[1]]}  
  
t

## X-squared   
## 6.624765

pl=sum(tstar<t)/B  
pr=sum(tstar>t)/B  
p=2\*min(pl,pr)  
pl;pr;p

## [1] 0.971

## [1] 0.027

## [1] 0.054

# Question 4  
chisq.test(xtabs(~medicin+nausea))

## Pearson's Chi-squared test  
##   
## data: xtabs(~medicin + nausea)  
## X-squared = 6.6248, df = 2, p-value = 0.03643

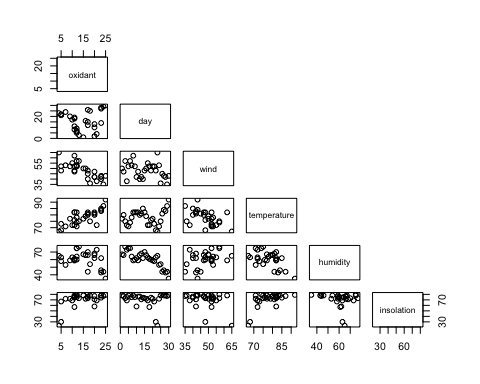
**Exercise 5**

1. R Script presented above.
2. The outcome of xtabs (~ medicine + nausea) presents the data contained in the data frame in a aggregated view in which the different factors (in this case medicines) are presented in different rows and their respective outcomes (aggregated) are presented in the columns, in this case one column with the number of patients with or without nausea.

It’s possible to see from xtabs that Chlorpromazine had the biggest number of samples (patients) compared to the other medicines. It also had the best performance (when considering the number of patients without nausea after the treatment).

1. Given that the value for p-value = 0.042 we reject the null hypothesis that the populations are the same, in other words, there might be a difference between the different treatments (bear in mind that the p-value can change for each computation which can lead to a different conclusion about rejecting or not the null hypothesis).
2. The p-value calculated from chi-square test is different from the value calculated by the permutation test. The p-value from chi-square would very unlikely be the same as the p-value from the permutation test, since the last is obtained from a random permutation (without replacement) of the labels which in turn leads to slightly different p-values at every computation, but the p-values could be the same if indeed the populations were exactly the same (which is the null hypothesis of this test that was rejected in the item 5.3 – above).

### Exercise 6  
  
# Question 1  
airpollution= read.table("airpollution.txt", header = TRUE)  
pairs(oxidant ~ day + wind + temperature + humidity + insolation , data= airpollution, upper.panel= NULL)



# Question 2  
oxidant\_day= lm(oxidant ~ day, data = airpollution)  
summary(oxidant\_day)

## lm(formula = oxidant ~ day, data = airpollution)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.68966 2.28580 5.989 1.89e-06 \*\*\*  
## day 0.07164 0.12876 0.556 0.582   
## Residual standard error: 6.104 on 28 degrees of freedom  
## Multiple R-squared: 0.01093, Adjusted R-squared: -0.02439

#Multiple R-squared: 0.01093  
  
oxidant\_wind= lm(oxidant ~ wind, data = airpollution)  
summary(oxidant\_wind)

## lm(formula = oxidant ~ wind, data = airpollution)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 45.3171 4.8976 9.253 5.19e-10 \*\*\*  
## wind -0.6331 0.1005 -6.300 8.20e-07 \*\*\*  
## Multiple R-squared: 0.5863, Adjusted R-squared: 0.5715

#Multiple R-squared: 0.5863  
  
oxidant\_temperature=lm(oxidant ~ temperature, data = airpollution)  
summary(oxidant\_temperature)

## lm(formula = oxidant ~ temperature, data = airpollution)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -46.4292 9.9542 -4.664 6.94e-05 \*\*\*  
## temperature 0.7850 0.1273 6.168 1.17e-06 \*\*\*  
## Multiple R-squared: 0.576, Adjusted R-squared: 0.5609

#Multiple R-squared: 0.576  
  
oxidant\_humidity=lm(oxidant ~ humidity, data = airpollution)  
summary(oxidant\_humidity)

## lm(formula = oxidant ~ humidity, data = airpollution)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 27.4446 6.4368 4.264 0.000206 \*\*\*  
## humidity -0.2088 0.1049 -1.991 0.056317 .   
## Multiple R-squared: 0.124, Adjusted R-squared: 0.09273

#Multiple R-squared: 0.124  
  
oxidant\_insolation=lm(oxidant ~ insolation, data = airpollution)  
summary(oxidant\_insolation)

## lm(formula = oxidant ~ insolation, data = airpollution)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.43279 5.32967 -0.269 0.79003   
## insolation 0.22993 0.07424 3.097 0.00441 \*\*  
## Multiple R-squared: 0.2552, Adjusted R-squared: 0.2286

#Multiple R-squared: 0.2552  
  
## step 1 - select the highest R-squared (wind)  
oxidant1= lm(oxidant ~ wind, data = airpollution)  
summary(oxidant1)

## lm(formula = oxidant ~ wind, data = airpollution)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 45.3171 4.8976 9.253 5.19e-10 \*\*\*  
## wind -0.6331 0.1005 -6.300 8.20e-07 \*\*\*  
## Multiple R-squared: 0.5863, Adjusted R-squared: 0.5715

#Multiple R-squared: 0.5863  
  
## step 2 - select the highest R-squared (temperature as second variable) among below options  
oxidant2=lm(oxidant ~ wind + temperature, data = airpollution)  
summary(oxidant2)

## lm(formula = oxidant ~ wind + temperature, data = airpollution)  
## 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 \*\*\*  
## Multiple R-squared: 0.7773, Adjusted R-squared: 0.7608

#Multiple R-squared:0.7773   
  
oxidant2=lm(oxidant ~ wind + insolation, data = airpollution)  
summary(oxidant2)

## lm(formula = oxidant ~ wind + insolation, data = airpollution)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 32.32615 6.97098 4.637 8.07e-05 \*\*\*  
## wind -0.55639 0.09778 -5.690 4.81e-06 \*\*\*  
## insolation 0.13161 0.05383 2.445 0.0213 \*   
## Multiple R-squared: 0.6613, Adjusted R-squared: 0.6362

#Multiple R-squared:0.6613   
  
oxidant2=lm(oxidant ~ wind + humidity, data = airpollution)  
summary(oxidant2)

## lm(formula = oxidant ~ wind + humidity, data = airpollution)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 46.91570 5.68573 8.251 7.38e-09 \*\*\*  
## wind -0.60955 0.10971 -5.556 6.86e-06 \*\*\*  
## humidity -0.04516 0.07866 -0.574 0.571   
## Multiple R-squared: 0.5913, Adjusted R-squared: 0.561

#Multiple R-squared:0.5913 - one variable become insignificant  
  
oxidant2=lm(oxidant ~ wind + day, data = airpollution)  
summary(oxidant2)

## lm(formula = oxidant ~ wind + day, data = airpollution)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 47.84224 5.62785 8.501 4.10e-09 \*\*\*  
## wind -0.65984 0.10489 -6.291 9.87e-07 \*\*\*  
## day -0.07986 0.08691 -0.919 0.366   
## Multiple R-squared: 0.5989, Adjusted R-squared: 0.5691

#Multiple R-squared:0.5989 - one variable become insignificant  
  
## step 3 - all added variable make previous variables insignificant - stop stepping up  
oxidant3=lm(oxidant ~ wind + temperature + insolation, data = airpollution)  
summary(oxidant3)

## lm(formula = oxidant ~ wind + temperature + insolation, data = airpollution)  
## 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   
## Multiple R-squared: 0.7816, Adjusted R-squared: 0.7565

#Multiple R-squared:0.7816 - one variable become insignificant  
  
oxidant3=lm(oxidant ~ wind + temperature + humidity, data = airpollution)  
summary(oxidant3)

## lm(formula = oxidant ~ wind + temperature + humidity, data = airpollution)  
## 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   
## Multiple R-squared: 0.7964, Adjusted R-squared: 0.7729

#Multiple R-squared:0.7964 - one variable become insignificant  
  
oxidant3=lm(oxidant ~ wind + temperature + day, data = airpollution)  
summary(oxidant3)

## lm(formula = oxidant ~ wind + temperature + day, data = airpollution)  
## 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   
## Multiple R-squared: 0.7958, Adjusted R-squared: 0.7722

#Multiple R-squared:0.7958 - one variable become insignificant  
  
## Best Model - step 2  
oxidant2

##   
## Call:  
## lm(formula = oxidant ~ wind + day, data = airpollution)  
##   
## Coefficients:  
## (Intercept) wind day   
## 47.84224 -0.65984 -0.07986

# Question 3  
## step 1 - full model  
oxidant5= lm(oxidant ~ wind + temperature + insolation + humidity + day, data = airpollution)  
summary(oxidant5)

## lm(formula = oxidant ~ wind + temperature + insolation + humidity +   
## day, data = airpollution)  
## 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   
## F-statistic: 19.01 on 5 and 24 DF, p-value: 1.203e-07

## step 2 - remove the highest p-value (day)  
oxidant4= lm(oxidant ~ wind + temperature + insolation + humidity , data = airpollution)  
summary(oxidant4)

## lm(formula = oxidant ~ wind + temperature + insolation + humidity,   
## data = airpollution)  
## 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   
## F-statistic: 24.69 on 4 and 25 DF, p-value: 2.279e-08

## step 3 - remove second highest p-value (insolation)  
oxidant3= lm(oxidant ~ wind + temperature + humidity , data = airpollution)  
summary(oxidant3)

## lm(formula = oxidant ~ wind + temperature + humidity, data = airpollution)  
## 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   
## F-statistic: 33.89 on 3 and 26 DF, p-value: 3.904e-09

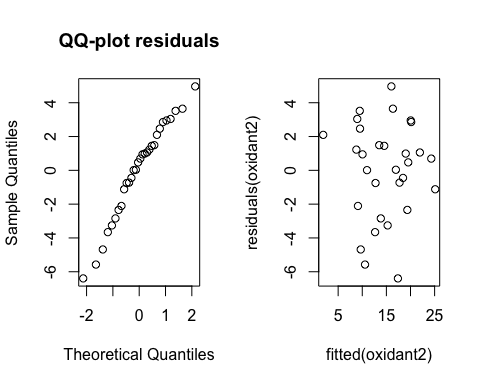
## step 4 - remove third highest p-value (humidity)  
oxidant2= lm(oxidant ~ wind + temperature , data = airpollution)  
summary(oxidant2)

## lm(formula = oxidant ~ wind + temperature, data = airpollution)  
## 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 \*\*\*  
## F-statistic: 47.12 on 2 and 27 DF, p-value: 1.563e-09

# Question 4  
## Best Model  
oxidant2

## lm(formula = oxidant ~ wind + temperature, data = airpollution)  
## Coefficients:  
## (Intercept) wind temperature   
## -5.2033 -0.4271 0.5204

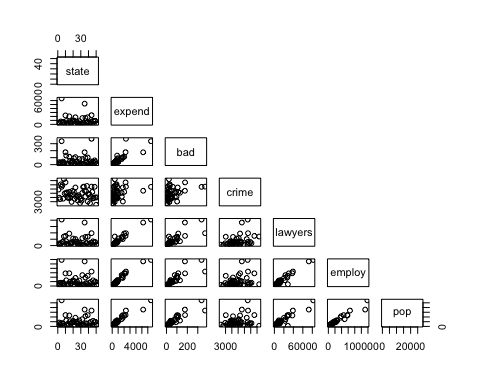
# Question 5  
par(mfrow=c(1,2))



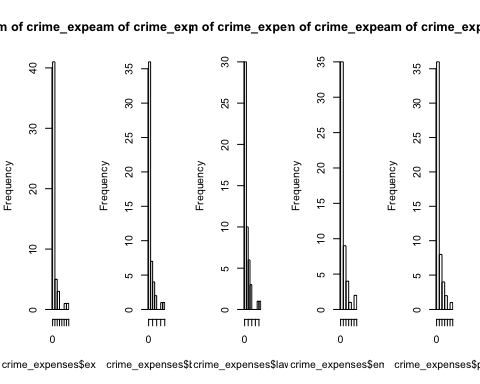
**Exercise 6**

1. According to scatterplots, wind and temperature have the most influence on oxidant. Their plots approximately show linear correlation. Afterward, insolation and humidity could affect oxidant linearly.
2. Among all explanatory variables, wind has the highest R-squared. (0.5863). Therefore, it should be selected as the basis to start to find linear regression model. In step 2, temperature is chosen to be added to the model with R-squared 0.7773 (highest among others). Adding third variable to the model in step 3 makes one variable of previous model insignificant. So, we should stop extending the model. Finally, the appropriate model consists of two explanatory variables which are wind and temperature.
3. In the first and second and third step, day, insolation and humidity are removed from the model respectively due to their p-values. In this part, similarly we reach to the same model as in previous part. This model is: oxidant ~ wind + temperature
4. Wind and temperature equal to -0.427and 0.52 respectively.
5. To investigate the normality of residuals, we consider the QQ-plot. We can assume normality based on the graph. Moreover, the samples are scattered approximately in whole area according to the fitted-residuals plot.

### Exercise 7  
  
# Question 1  
crime\_expenses <- read.table(file = "expensescrime.txt", header = TRUE)  
  
#Identifying potential correlations  
pairs(crime\_expenses, upper.panel=NULL)



# It's possible to see a potential correlation between expend and bad, lawyers, employ and pop.  
# Verifying the histogram for these variables:  
par(mfrow=c(1,5))  
hist(crime\_expenses$expend)  
hist(crime\_expenses$bad)  
hist(crime\_expenses$lawyers)  
hist(crime\_expenses$employ)  
hist(crime\_expenses$pop)



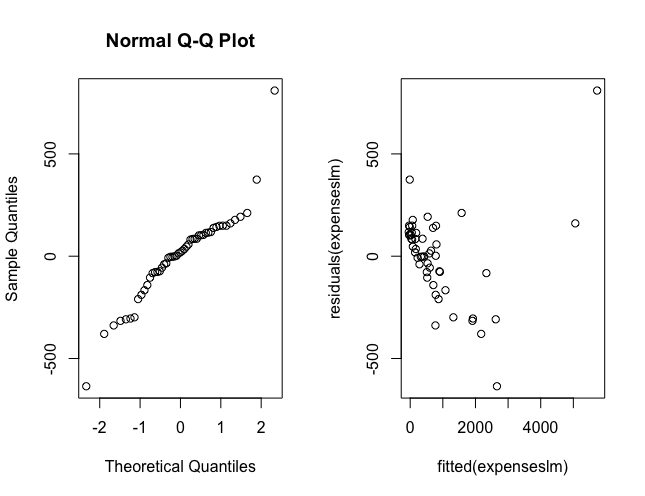
# Thus a first attempt to calculate the linear model will be made considering these factors.  
attach(crime\_expenses)  
expenseslm = lm(expend~bad+lawyers+employ+pop, data=crime\_expenses)  
summary(expenseslm)

## lm(formula = expend ~ bad + lawyers + employ + pop, data = crime\_expenses)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.464e+02 4.541e+01 -3.224 0.00232 \*\*  
## bad -2.241e+00 1.133e+00 -1.977 0.05402 .   
## lawyers 2.646e-02 7.571e-03 3.495 0.00106 \*\*  
## employ 2.283e-02 7.487e-03 3.049 0.00380 \*\*  
## pop 6.368e-02 3.304e-02 1.927 0.06012 .   
## F-statistic: 332.5 on 4 and 46 DF, p-value: < 2.2e-16

confint(expenseslm)

## 2.5 % 97.5 %  
## (Intercept) -2.378296e+02 -55.01809137  
## bad -4.521680e+00 0.04037352  
## lawyers 1.122149e-02 0.04169976  
## employ 7.761195e-03 0.03790378  
## pop -2.824611e-03 0.13018861

#Assessing the current model  
par(mfrow=c(1,1))  
qqnorm(residuals(expenseslm))



shapiro.test(residuals(expenseslm)) #Extract check using Shapiro's normality test.

## Shapiro-Wilk normality test  
## data: residuals(expenseslm)  
## W = 0.91203, p-value = 0.001087

#2nd Iteration:  
expenseslm2 = lm(expend~bad\*lawyers\*employ, data=crime\_expenses)  
summary(expenseslm2)

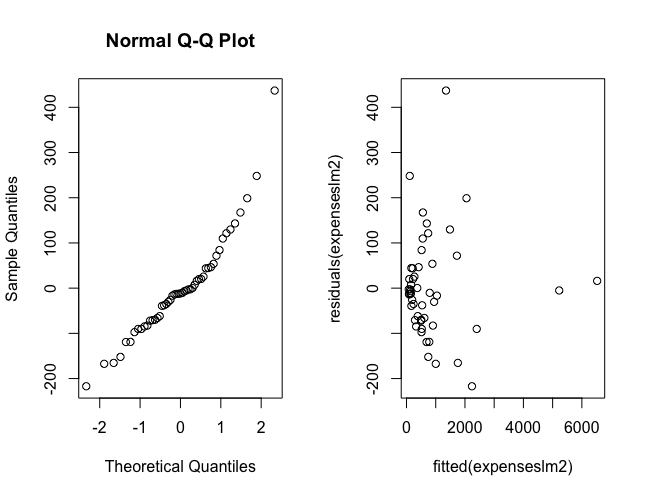
## lm(formula = expend ~ bad \* lawyers \* employ, data = crime\_expenses)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 9.750e-01 3.511e+01 0.028 0.9780   
## bad 2.649e+00 1.714e+00 1.545 0.1297   
## lawyers 7.365e-03 7.442e-03 0.990 0.3279   
## employ 3.052e-02 5.006e-03 6.097 2.64e-07 \*\*\*  
## bad:lawyers -3.770e-05 1.682e-04 -0.224 0.8237   
## bad:employ -7.931e-05 8.661e-05 -0.916 0.3649   
## lawyers:employ 1.044e-07 8.609e-08 1.213 0.2319   
## bad:lawyers:employ 1.412e-09 5.284e-10 2.673 0.0106 \*   
## F-statistic: 680.2 on 7 and 43 DF, p-value: < 2.2e-16

# Assessing the new regression parameters:

qqnorm(residuals(expenseslm2))

plot(fitted(expenseslm2),residuals(expenseslm2))

shapiro.test(residuals(expenseslm2)) #Extract check using Shapiro's normality test.



## Shapiro-Wilk normality test  
## data: residuals(expenseslm2)  
## W = 0. 91853, p-value = 0.001858

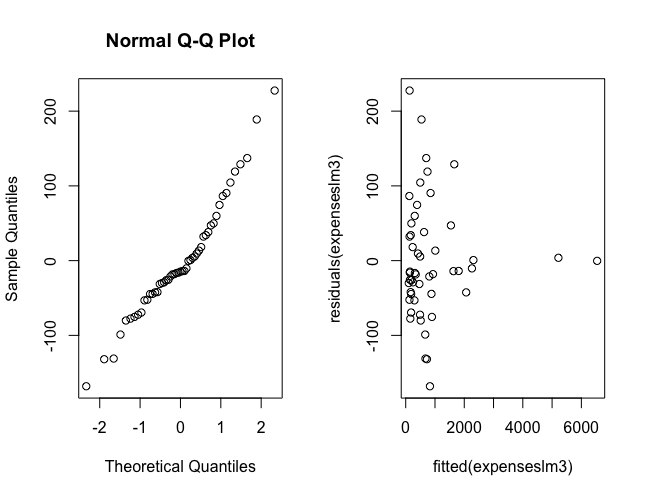
#3rd Iteration:  
  
crime\_expenses$bad2 = crime\_expenses$bad^2  
expenseslm3 = lm(expend~bad\*lawyers\*employ\*bad2, data=crime\_expenses)  
summary(expenseslm3)

qqnorm(residuals(expenseslm3))

plot(fitted(expenseslm3),residuals(expenseslm3))

shapiro.test(residuals(expenseslm3)) #Extract check using Shapiro's normality test.

## Call:  
## lm(formula = expend ~ bad \* lawyers \* employ \* bad2, data = crime\_expenses)  
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.363e+02 5.737e+01 2.376 0.02311 \*   
## bad -8.186e+00 8.960e+00 -0.914 0.36720   
## lawyers -1.297e-01 4.309e-02 -3.010 0.00482 \*\*  
## employ 8.236e-02 3.199e-02 2.574 0.01444 \*   
## bad2 -1.440e-01 2.315e-01 -0.622 0.53783   
## bad:lawyers 9.096e-03 2.766e-03 3.288 0.00230 \*\*  
## bad:employ -2.846e-03 1.390e-03 -2.048 0.04814 \*   
## lawyers:employ 2.403e-07 1.658e-06 0.145 0.88559   
## bad:bad2 4.209e-04 1.268e-03 0.332 0.74201   
## lawyers:bad2 -1.318e-04 4.415e-05 -2.984 0.00515 \*\*  
## employ:bad2 6.072e-05 2.342e-05 2.593 0.01380 \*   
## bad:lawyers:employ -8.522e-08 3.972e-08 -2.145 0.03894 \*   
## bad:lawyers:bad2 6.559e-07 2.397e-07 2.736 0.00969 \*\*  
## bad:employ:bad2 -3.253e-07 1.241e-07 -2.622 0.01285 \*   
## lawyers:employ:bad2 8.008e-10 2.751e-10 2.911 0.00624 \*\*  
## bad:lawyers:employ:bad2 -2.450e-12 8.576e-13 -2.856 0.00717 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 92.9 on 35 degrees of freedom  
## Multiple R-squared: 0.9957, Adjusted R-squared: 0.9939   
## F-statistic: 542.4 on 15 and 35 DF, p-value: < 2.2e-16



## Shapiro-Wilk normality test  
## data: residuals(expenseslm3)  
## W = 0.95998, p-value = 0.08324

**Exercise 7**

Using the function pairs it’s possible to see a potential correlation between expend and bad, lawyers, employ and pop.

Thus, a first attempt to calculate the linear model will be made considering these factors.

**1st Iteration**

The model was assessed after the first iteration using: qqnorm, the fitted-residuals plot and the Schapiro test.

To start, it is already possible to see via the qqnorm graph that the distribution is far from normal. The Shapiro test with a calculated p-value of 0.001 also go on that direction (rejecting the null hypothesis – that is a normal distribution).

It is also possible to see a concentration on the fitted-residuals plot in the region between 0 and 1000 (fitted values) for the different fitted re is concentrated in a region around smaller value of fitted expenses.

**2nd Iteration**

Considering that in the previous iteration, looking at the calculated p-values, lawyers and employ reject the null hypothesis the 2nd iteration will consider these factors (bad will also be considered since it has the biggest estimated coeficient).

Additionally, now we will calculate the regression considering interaction between the variables.

After the calculations is possible to see that the qqnorm presents a curved shape with some points far from the line, the fitted-residuals plot again shows a concentration around certain fitted values (<1000) and the shapiro test still rejects the null hypothesis (indicating a non-normal distribution for the residuals) p-value = 0.0018.

**3rd Iteration**

Given that bad is the variable with highest coeficient we will try to have better results by elevating bad to the power of 2 (still taking interaction in consideration).

After the third iteration the qqnorm graph presents a better slope and distance between the points. The fitted-residuals plot shows more sparse points. Finally the Shapiro test presents a p-value = 0.08324 in which we fail to reject the null hypothesis.

After the 3rd iteration is possible to present a reasonable model:

expenses = 136.3 -8.186\*bad -0.1297\*lawyers + 0.08236\*employ - 0.1440\*bad^2