

Assignment 1

Oguzhan Yetkin, Oleg Litvinov, Victor Retamal, group 4

15 March 2022

Exercise 1. Post-operative nausea.

The file contains data about post-operative nausea after medication against nausea. Two different medicines were administered to patients that complained about post-operative nausea. One of the medicines, Pentobarbital, was administered in two different doses.

```
nausea_df <- read.table("data/nauseatable.txt", header=TRUE)
nausea_df
```

##	Incidence.of.no.nausea	Incidence.of.Nausea
## Chlorpromazine	100	52
## Pentobarbital(100mg)	32	35
## Pentobarbital(150mg)	48	37

a) Discuss whether a contingency table test is appropriate here. If yes, perform this test in order to test whether the different medicines work equally well against nausea. Where are the main inconsistencies?

There are two factors: presence of nausea and the medication. For each combination of factors, the number of cases are registered. Contingency table test is applicable in terms of the task to find the dependency between the factors. For that a specific condition has to be met.

```
z=chisq.test(nausea_df)
z
```

```
##
## Pearson's Chi-squared test
##
## data:  nausea_df
## X-squared = 6.6248, df = 2, p-value = 0.03643
```

There are no contraindications for the chi-square test. The test concludes that there is a dependence between row and column variables. Let's check what is that difference.

```
library(corrplot)
```

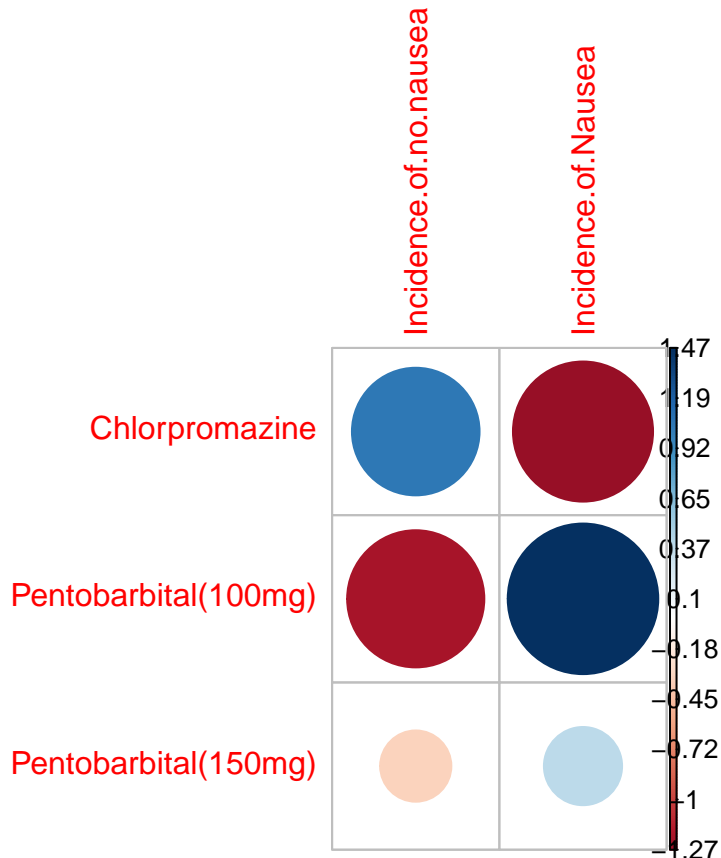
```
## corrplot 0.92 loaded
```

```
z$residuals
```

##	Incidence.of.no.nausea	Incidence.of.Nausea
----	------------------------	---------------------

```
## Chlorpromazine          1.0540926      -1.270001
## Pentobarbital(100mg)    -1.2179181      1.467383
## Pentobarbital(150mg)    -0.3282848      0.395527
```

```
corrplot(z$residuals, is.cor = FALSE)
```



Chlorpromazine is relatively more helpful in terms of fighting against nausea in comparison to both dosages of Pentobarbital. Also, 100mg of Pentobarbital has more nausea cases.

b) Perform a permutation test in order to test whether the different medicines work equally well against nausea. Permute the medicine labels for this purpose. Use as test statistic the chisquare test statistic for contingency tables, which can be extracted from the output of the command `chisq.test`. (Hint: make a data frame in R consisting of two columns. One column should contain an indicator whether or not the patient in that row suffered from nausea, and the other column should indicate the medicine.)

```
indicator_col <- c()
label_col <- c()
for(i in 1:3){
  indicator_col <- append(indicator_col, rep(0, nausea_df[i, 1]))
  indicator_col <- append(indicator_col, rep(1, nausea_df[i, 2]))
  label_col <- append(label_col, rep(rownames(nausea_df)[i], rowSums(nausea_df[i, ])))
}
```

```
nausea_two_col_df <- data.frame(indicator_col, label_col)
head(nausea_two_col_df)
```

```
##   indicator_col      label_col
## 1             0 Chlorpromazine
## 2             0 Chlorpromazine
## 3             0 Chlorpromazine
## 4             0 Chlorpromazine
## 5             0 Chlorpromazine
## 6             0 Chlorpromazine
```

```
mystat <- function(x) chisq.test(x)$statistic
B <- 1000
tstar <- numeric(B)
for(i in 1:B){
  perm_label <- sample(nausea_two_col_df$label_col) ## permuting the labels
  tstar[i] <- mystat(table(data.frame(nausea_two_col_df$indicator_col, perm_label)))
}
myt <- mystat(table(data.frame(nausea_two_col_df$indicator_col, nausea_two_col_df$label_col)))

pl <- sum(tstar<myt)/B
pr <- sum(tstar>myt)/B
p_perm <- min(pl, pr)
p_perm
```

```
## [1] 0.035
```

The permutation test rejects the null hypothesis that different medicines work equally well against nausea.

c) Compare the p-value found by the permutation test with the p-value found from the chisquare test for contingency tables. Explain the difference/equality of the two p-values.

Relaunch of the permutation test retrieves the p-value of about 0.03-0.04 while the p-value from the chi-square test is 0.036. The permutation test is completely suitable for such kind of tasks. It also reveals the same conclusion and similar p-value as the chi-square test.

Exercise 2. Airpollution.

The data were obtained to determine predictors related to air pollution. We want to investigate which explanatory variables need to be included into a linear regression model with oxidant as the response variable.

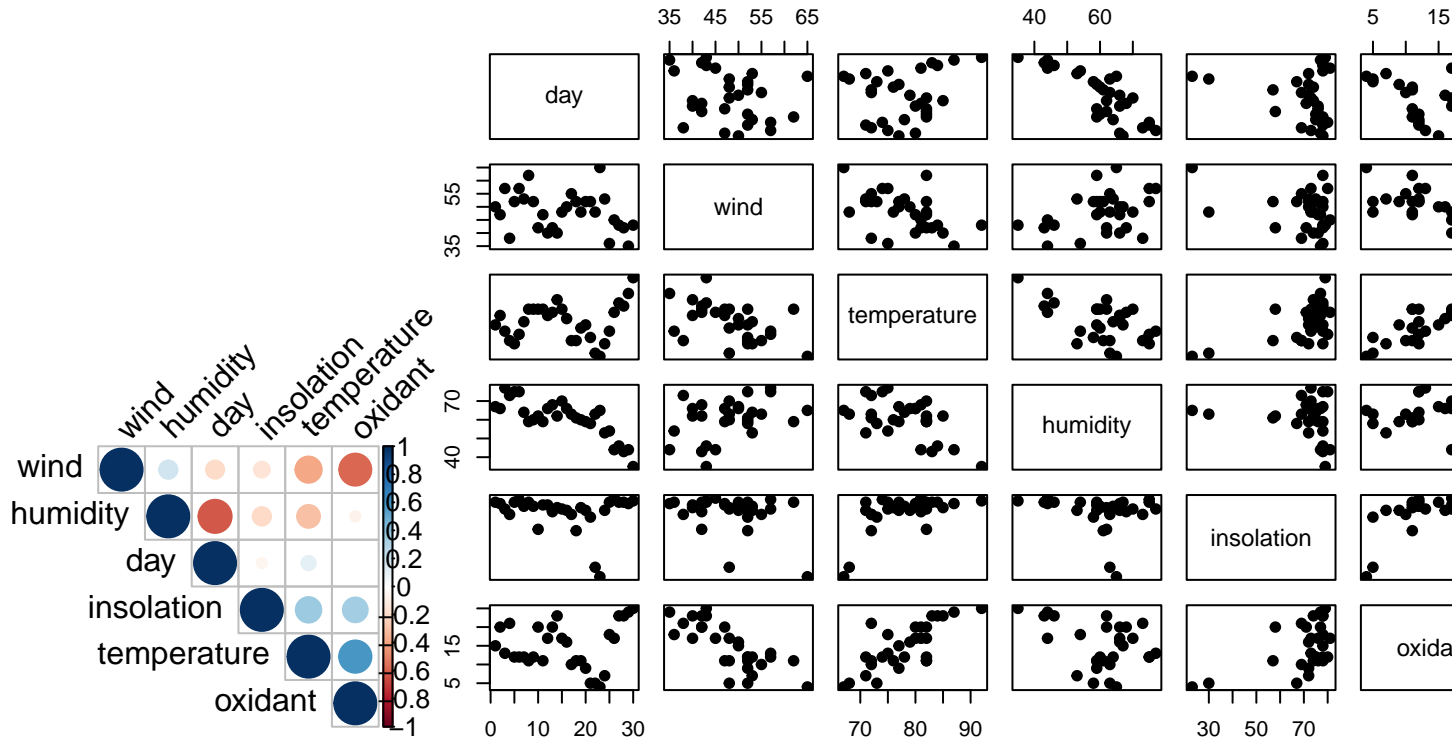
```
pollution_df <- read.table("data/airpollution.txt", header=TRUE)
head(pollution_df)
```

```
##   day wind temperature humidity insolation oxidant
## 1   1   50           77        67          78      15
## 2   2   47           80        66          77      20
```

```
## 3    3    57          75      77      73      13
## 4    4    38          72      73      69      21
## 5    5    52          71      75      78      12
## 6    6    57          74      75      80      12
```

a) Make some graphical summaries of the data. Investigate the problem of potential and influence points, and the problem of collinearity.

```
# cor(pollution_df, method = c("spearman"))
if (!require("corrplot")) install.packages("corrplot")
library(corrplot)
par(mfrow=c(2, 1))
res <- cor(pollution_df, method='kendall')
corrplot(res, type = "upper", order = "hclust",
          tl.col = "black", tl.srt = 45)
pairs(pollution_df, pch = 19)
```



The command `order(abs(residuals(model)))` gives the indices of the ordered absolute values of residuals from smallest to largest.

```
pollution_lm = lm(oxidant~insolation+humidity+wind+temperature+day, data=pollution_df)
order(abs(residuals(pollution_lm)))
```

```
## [1] 25  8 10 13 29 26 14  1 15 24 16  7  5 17 18  6 30  3  2 19 12 20 22  9 23
## [26] 27  4 28 21 11
```

The mean shift outlier model can be applied to test whether the k-th point significantly deviates from the other points in a linear regression setting.

```

u_out=rep(0, nrow(pollution_df)); u_out[21]=1
pollution_lm_outlier=lm(oxidant~insolation+humidity+wind + u_out, data=pollution_df); summary()

##
## Call:
## lm(formula = oxidant ~ insolation + humidity + wind + u_out,
##     data = pollution_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.1135 -1.7707  0.3907  2.7188  5.0021
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  33.79741    7.16856   4.715 7.81e-05 ***
## insolation    0.12672    0.05119   2.476  0.0204 *
## humidity     -0.04665    0.06837  -0.682  0.5013
## wind         -0.51577    0.09917  -5.201 2.22e-05 ***
## u_out        -7.76237    3.54111  -2.192  0.0379 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.449 on 25 degrees of freedom
## Multiple R-squared:  0.7181, Adjusted R-squared:  0.673
## F-statistic: 15.92 on 4 and 25 DF, p-value: 1.335e-06

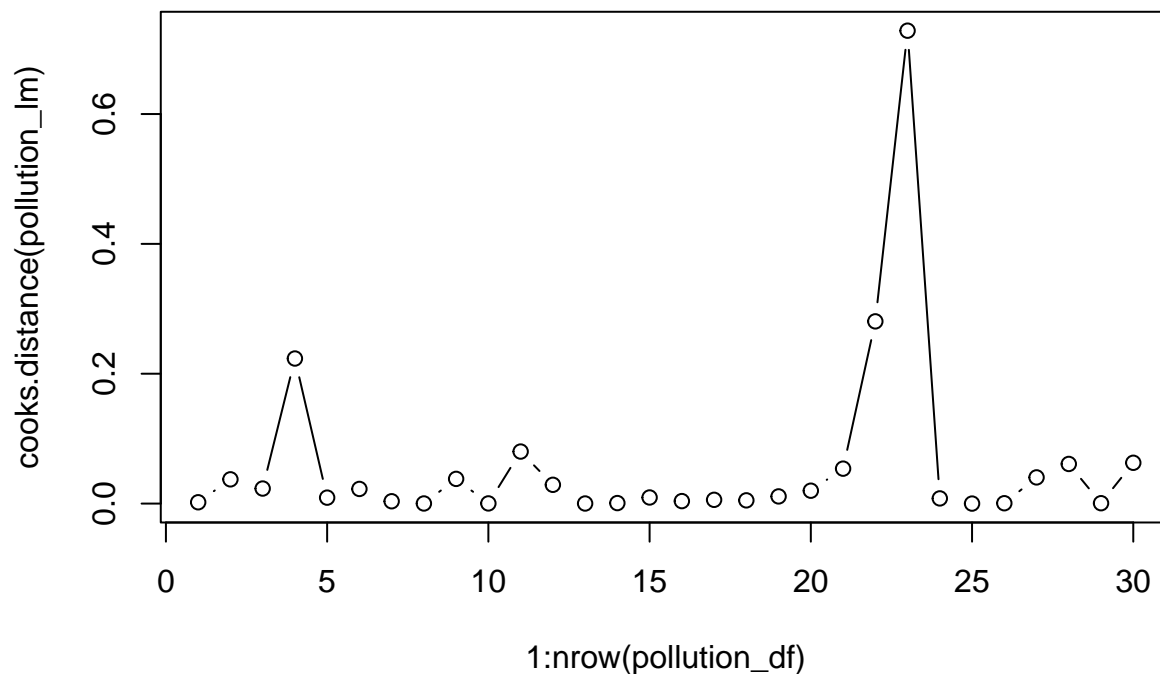
```

Only the 21 outlier is significant.

The Cook's distance D_i quantifies the influence of observation i on the predictions:

$$D_i = \frac{1}{(p+1)\hat{\sigma}^2} \sum_{j=1}^n \left(\hat{Y}_{(i),j} - \hat{Y}_j \right)^2$$

```
plot(1:nrow(pollution_df), cooks.distance(pollution_lm), type="b")
```

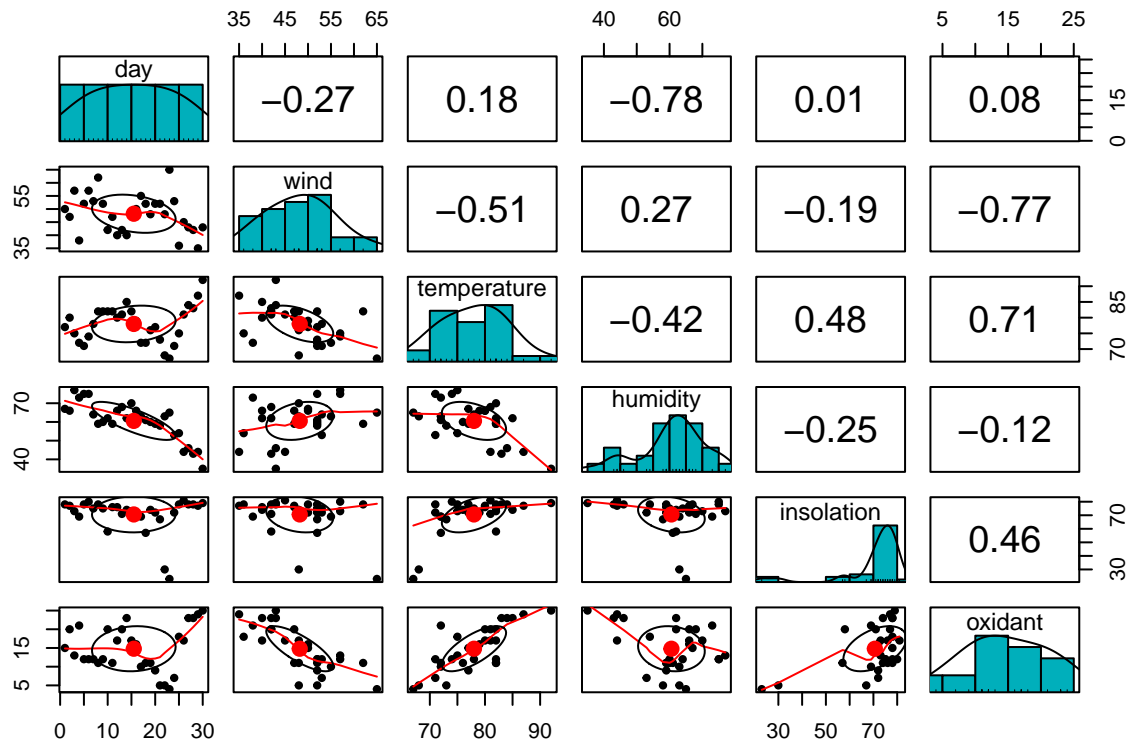


Rule of thumb: if the Cook's distance for some data point is close to or larger than 1, it is considered to be an influence point. So the point 23 is an influence here.

```
if (!require("psych")) install.packages("psych")

## Loading required package: psych

library(psych)
pairs.panels(pollution_df,
  method = "spearman", # correlation method
  hist.col = "#00AFBB",
  density = TRUE, # show density plots
  ellipses = TRUE # show correlation ellipses
)
```



b) Use the added variable plot to depict the relationship between response oxidant and predictor wind. What is the meaning of the slope of fitted regression for this scatter plot?

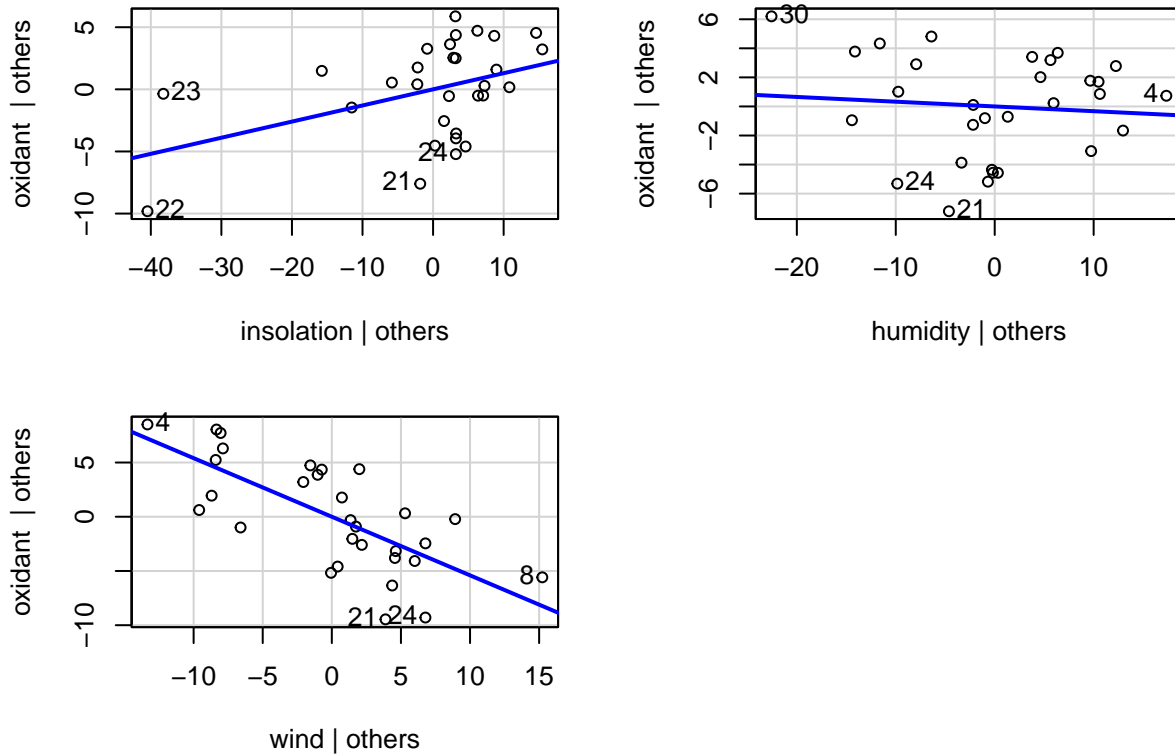
```
if (!require("car")) install.packages("car")

## Loading required package: car
## Loading required package: carData
##
## Attaching package: 'car'
## The following object is masked from 'package:psych':
##
##   logit

library(car)

attach(pollution_df)
mod = lm(oxidant~insolation+humidity+wind)
par(mfrow=c(2, 1))
avPlots(mod)
```

Added-Variable Plots



```
summary(mod)
```

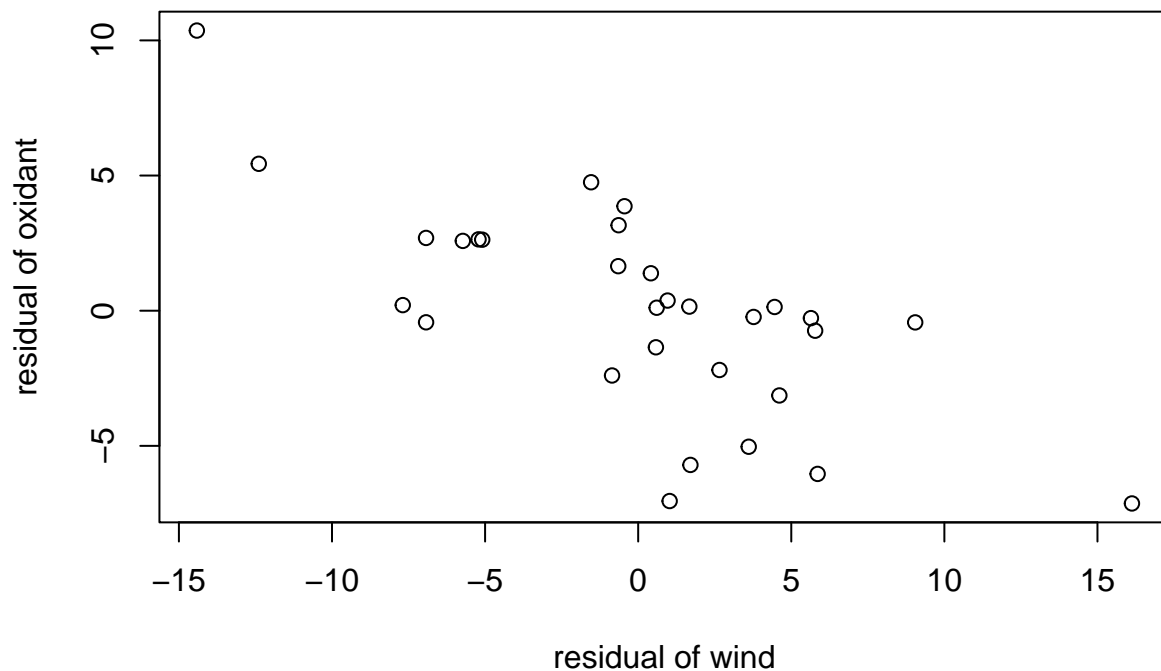
```
##
## Call:
## lm(formula = oxidant ~ insolation + humidity + wind)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7.3630 -2.4212  0.5585  3.0466  5.4644
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  33.65768    7.67492   4.385  0.00017 ***
## insolation    0.12984    0.05479   2.370  0.02550 *
## humidity     -0.03266    0.07288  -0.448  0.65775
## wind         -0.54037    0.10550  -5.122 2.44e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.693 on 26 degrees of freedom
## Multiple R-squared:  0.6639, Adjusted R-squared:  0.6251
## F-statistic: 17.12 on 3 and 26 DF, p-value: 2.423e-06
```

The slopes on the plots reflect the regression coefficients from the original multiple regression model mod.


```

y = residuals(lm(pollution_df$oxidant~pollution_df$temperature+pollution_df$insolation+pollution_df$wind))
x = residuals(lm(pollution_df$wind~pollution_df$temperature+pollution_df$insolation+pollution_df$oxidant))
plot(x, y, xlab='residual of wind', ylab='residual of oxidant')

```



c) Fit a linear regression model to the data. Use both the step-up and step-down methods to find the best model. If step-up and step-down yield two different models, choose one and motivate your choice.

```

for(i in names(pollution_df)){
  if(i == 'oxidant'){next}
  # summary(lm(oxidant~i))
  print(summary(lm(paste('pollution_df$oxidant', '~pollution_df$', i))))
}

```

Step-up

```

##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$",
##   i))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.3373  -3.8537   0.1298   5.5403   9.1613
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    13.68966     2.28580   5.989 1.89e-06 ***

```

```

## pollution_df$day 0.07164 0.12876 0.556 0.582
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.104 on 28 degrees of freedom
## Multiple R-squared: 0.01093, Adjusted R-squared: -0.02439
## F-statistic: 0.3095 on 1 and 28 DF, p-value: 0.5824
##
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$",
## i))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9.9266 -2.5923  0.2065  2.6636  6.9077
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    45.3171     4.8976   9.253 5.19e-10 ***
## pollution_df$wind -0.6331     0.1005  -6.300 8.20e-07 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.948 on 28 degrees of freedom
## Multiple R-squared: 0.5863, Adjusted R-squared: 0.5715
## F-statistic: 39.68 on 1 and 28 DF, p-value: 8.205e-07
##
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$",
## i))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.9400 -2.2138  0.3775  2.5550 10.9099
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -46.4292     9.9542  -4.664 6.94e-05 ***
## pollution_df$temperature  0.7850     0.1273   6.168 1.17e-06 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.997 on 28 degrees of freedom
## Multiple R-squared: 0.576, Adjusted R-squared: 0.5609
## F-statistic: 38.04 on 1 and 28 DF, p-value: 1.167e-06
##

```

```
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$",
##     i))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.3358  -4.0749   0.8782   4.7800   8.7957
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      27.4446      6.4368   4.264 0.000206 ***
## pollution_df$humidity -0.2088      0.1049  -1.991 0.056317 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.745 on 28 degrees of freedom
## Multiple R-squared:  0.124, Adjusted R-squared:  0.09273
## F-statistic: 3.964 on 1 and 28 DF,  p-value: 0.05632
##
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$",
##     i))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
##  -8.9723  -4.4841  -0.3281   4.7631   8.2686
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -1.43279      5.32967  -0.269  0.79003
## pollution_df$insolation  0.22993      0.07424   3.097  0.00441 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.297 on 28 degrees of freedom
## Multiple R-squared:  0.2552, Adjusted R-squared:  0.2286
## F-statistic: 9.592 on 1 and 28 DF,  p-value: 0.004411
```

Wind variable gives maximum increase in the R^2 . The variable is significant. Therefore, we can continue.

```
for(i in names(pollution_df)){
  if(i %in% c('oxidant', 'wind')){next}
  print(summary(lm(paste('pollution_df$oxidant', '~pollution_df$wind+pollution_df$', i))))
}
```

```
##
```

```

## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$wind+pollution_df$",
##      i))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9.4129 -2.5621  0.4498  2.3827  7.9267
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    47.84224     5.62785   8.501 4.10e-09 ***
## pollution_df$wind -0.65984     0.10489  -6.291 9.87e-07 ***
## pollution_df$day -0.07986     0.08691  -0.919  0.366
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.959 on 27 degrees of freedom
## Multiple R-squared:  0.5989, Adjusted R-squared:  0.5691
## F-statistic: 20.15 on 2 and 27 DF,  p-value: 4.411e-06
##
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$wind+pollution_df$",
##      i))
##
## 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
## pollution_df$wind    -0.42706     0.08645  -4.940 3.58e-05 ***
## pollution_df$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
##
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$wind+pollution_df$",
##      i))
##
## Residuals:
##      Min       1Q   Median       3Q      Max

```

```
## -9.8120 -2.2808 0.3433 3.0476 5.8757
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    46.91570     5.68573   8.251 7.38e-09 ***
## pollution_df$wind    -0.60955     0.10971  -5.556 6.86e-06 ***
## pollution_df$humidity -0.04516     0.07866  -0.574  0.571
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.996 on 27 degrees of freedom
## Multiple R-squared:  0.5913, Adjusted R-squared:  0.561
## F-statistic: 19.53 on 2 and 27 DF, p-value: 5.674e-06
##
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$wind+pollution_df$",
##      i))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7.2119 -2.7198  0.4815  2.8733  6.2012
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    32.32615     6.97098   4.637 8.07e-05 ***
## pollution_df$wind    -0.55639     0.09778  -5.690 4.81e-06 ***
## pollution_df$insolation 0.13161     0.05383   2.445  0.0213 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.638 on 27 degrees of freedom
## Multiple R-squared:  0.6613, Adjusted R-squared:  0.6362
## F-statistic: 26.36 on 2 and 27 DF, p-value: 4.491e-07
```

Temperature variable works in the same way as the previous choice. Continue.

```
for(i in names(pollution_df)){
  if(i %in% c('oxidant', 'wind', 'temperature')){next}
  print(summary(lm(paste('pollution_df$oxidant',
                        '~pollution_df$wind',
                        '+pollution_df$temperature',
                        '+pollution_df$',
                        i))))
}
```

```
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$wind",
```

```
##      "+pollution_df$temperature", "+pollution_df$", i))
##
## 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
## pollution_df$wind    -0.45604     0.08644  -5.276 1.63e-05 ***
## pollution_df$temperature  0.52918     0.10568   5.008 3.29e-05 ***
## pollution_df$day    -0.09711     0.06328  -1.535    0.137
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.878 on 26 degrees of freedom
## Multiple R-squared:  0.7958, Adjusted R-squared:  0.7722
## F-statistic: 33.78 on 3 and 26 DF,  p-value: 4.042e-09
##
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$wind",
##      "+pollution_df$temperature", "+pollution_df$", i))
##
## Residuals:
##      Min        1Q    Median        3Q        Max
## -6.5887 -1.1686  0.1978   1.9004   4.1544
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -16.60697    13.07154  -1.270    0.215
## pollution_df$wind    -0.44620     0.08513  -5.241 1.78e-05 ***
## pollution_df$temperature  0.60190     0.11764   5.117 2.47e-05 ***
## pollution_df$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
##
##
## Call:
## lm(formula = paste("pollution_df$oxidant", "~pollution_df$wind",
##      "+pollution_df$temperature", "+pollution_df$", i))
##
## 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
## pollution_df$wind   -0.42353     0.08737  -4.848 5.02e-05 ***
## pollution_df$temperature  0.47558     0.12564   3.785 0.000816 ***
## pollution_df$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
```

Humidity has the highest R-squared increase but the variable is not significant. Therefore, we don't add it to the model. Resulting model is:

```
print(summary(lm(pollution_df$oxidant~pollution_df$wind+pollution_df$temperature)))

##
## Call:
## lm(formula = pollution_df$oxidant ~ pollution_df$wind + pollution_df$temperature)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.3939 -1.8608  0.5826  1.9461  4.9661
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -5.20334    11.11810  -0.468   0.644
## pollution_df$wind   -0.42706     0.08645  -4.940 3.58e-05 ***
## pollution_df$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
```

oxidant = $-5.2 - 0.4wind + 0.5temperature$ + error, with R-squared = 0.8.

```
summary(lm(pollution_df$oxidant~ pollution_df$wind + pollution_df$temperature + pollution_df$d
```

Step-down

```
##
## Call:
## lm(formula = pollution_df$oxidant ~ pollution_df$wind + pollution_df$temperature +
```

```
##      pollution_df$day + pollution_df$humidity + pollution_df$insolation)
##
## Residuals:
##      Min        1Q    Median        3Q        Max
## -6.6920 -1.1675  0.2582   1.8289   4.0773
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -12.04010     21.20961   -0.568  0.57553
## pollution_df$wind    -0.44749      0.09103   -4.916 5.14e-05 ***
## pollution_df$temperature  0.55714      0.15347    3.630  0.00133 **
## pollution_df$day     -0.02997      0.13995   -0.214  0.83227
## pollution_df$humidity  0.06818      0.13336    0.511  0.61384
## pollution_df$insolation  0.01822      0.05583    0.326  0.74694
## ---
## 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
```

Day has the largest p-value and the value is larger than 0.05. Removing it.

```
summary(lm(pollution_df$oxidant ~ pollution_df$wind + pollution_df$temperature + pollution_df$
```

```
##
## Call:
## lm(formula = pollution_df$oxidant ~ pollution_df$wind + pollution_df$temperature +
##      pollution_df$humidity + pollution_df$insolation)
##
## Residuals:
##      Min        1Q    Median        3Q        Max
## -6.5861 -1.0961  0.3512   1.7570   4.0712
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -15.49370     13.50647   -1.147  0.26219
## pollution_df$wind    -0.44291      0.08678   -5.104 2.85e-05 ***
## pollution_df$temperature  0.56933      0.13977    4.073  0.00041 ***
## pollution_df$humidity  0.09292      0.06535    1.422  0.16743
## pollution_df$insolation  0.02275      0.05067    0.449  0.65728
## ---
## 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
```

Insolation is the largers from the insignificant. Removing it.


```
summary(lm(pollution_df$oxidant~ pollution_df$wind + pollution_df$temperature + pollution_df$humidity))
```

```
##
## Call:
## lm(formula = pollution_df$oxidant ~ pollution_df$wind + pollution_df$temperature +
##     pollution_df$humidity)
##
## 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
## pollution_df$wind   -0.44620     0.08513   -5.241 1.78e-05 ***
## pollution_df$temperature  0.60190     0.11764    5.117 2.47e-05 ***
## pollution_df$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
```

Humidity is the only insignificant. Removing.

```
summary(lm(pollution_df$oxidant~ pollution_df$wind + pollution_df$temperature))
```

```
##
## Call:
## lm(formula = pollution_df$oxidant ~ pollution_df$wind + pollution_df$temperature)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.3939 -1.8608  0.5826  1.9461  4.9661
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -5.20334    11.11810   -0.468    0.644
## pollution_df$wind   -0.42706     0.08645   -4.940 3.58e-05 ***
## pollution_df$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
```

All remaining variables are significant. Resulting model: $\text{oxidant} = -5.2 - 0.4\text{wind} + 0.5\text{temperature}$

+ error, with R-squared = 0.8. The model is the same as obtained with the step-up approach.

- d) Determine 95% confidence and prediction intervals for oxidant using the model you preferred in c) for wind=33, temperature=54, humidity=77 and insolation=21.

```
x1 <- pollution_df$wind
x2 <- pollution_df$temperature

mod = lm(pollution_df$oxidant ~ x1 + x2)

newxdata = data.frame(x1=33, x2=54)

predict(mod, newxdata, interval='prediction', level=0.95)

##          fit          lwr          upr
## 1 8.80281 -0.5617877 18.16741

predict(mod, newxdata, interval='confidence', level=0.95)

##          fit          lwr          upr
## 1 8.80281 1.656548 15.94907
```

Exercise 3. Fruit flies.

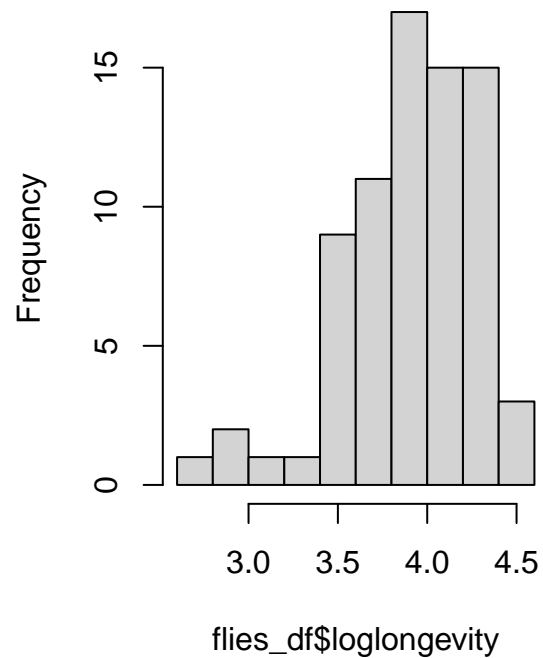
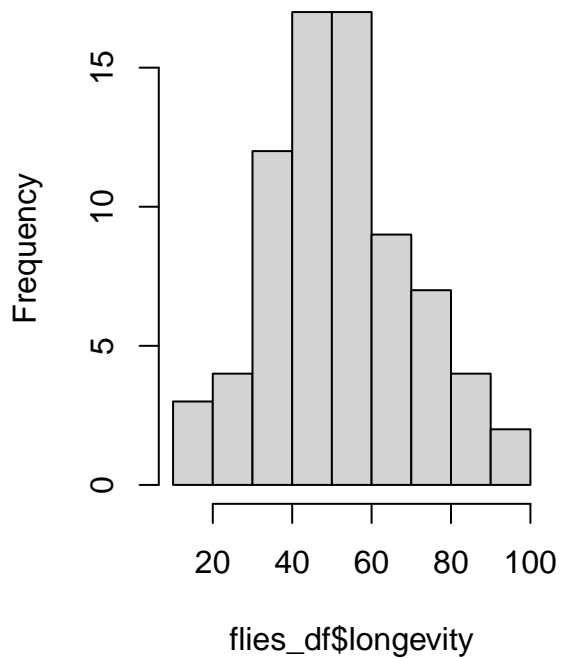
To investigate the effect of sexual activity on longevity of fruit flies, 75 male fruit flies were divided randomly in three groups of 25. The fruit flies in the first group were kept solitary, those in the second were kept together with one virgin female fruit fly per day, and those in the third group were kept together with eight virgin female fruit flies a day. In the data-file three groups are labelled isolated, low and high. The number of days until death (longevity) was measured for all flies. Later, it was decided to measure also the length of their thorax. Add a column loglongevity to the data-frame, containing the logarithm of the number of days until death. *Use this as the response variable in the following.*

```
flies_df <- read.table("data/fruitflies.txt", header=TRUE)
flies_df$loglongevity=log(flies_df$longevity)
head(flies_df)

##   thorax longevity activity loglongevity
## 1   0.64         40 isolated    3.688879
## 2   0.70         37 isolated    3.610918
## 3   0.72         44 isolated    3.784190
## 4   0.72         47 isolated    3.850148
## 5   0.72         47 isolated    3.850148
## 6   0.76         47 isolated    3.850148

par(mfrow=c(1, 2))
hist(flies_df$longevity)
hist(flies_df$loglongevity)
```

Histogram of flies_df\$longevity Histogram of flies_df\$loglongevity



```
shapiro.test(flies_df$longevity)
```

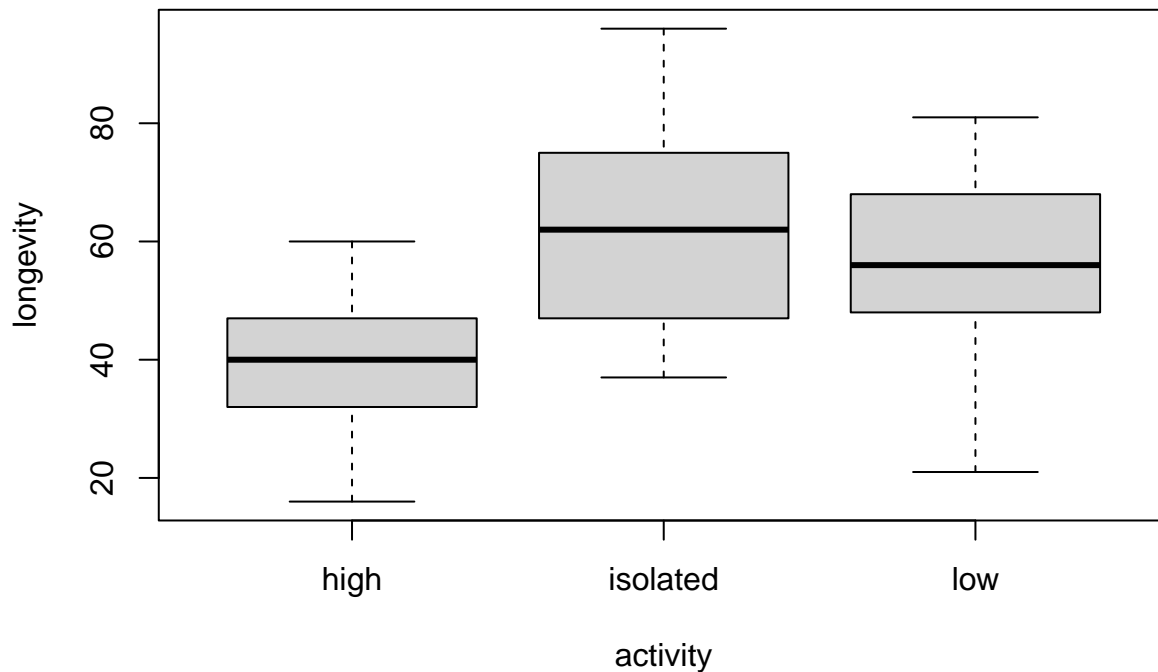
```
##
##  Shapiro-Wilk normality test
##
## data:  flies_df$longevity
## W = 0.98686, p-value = 0.6333
```

```
shapiro.test(flies_df$loglongevity)
```

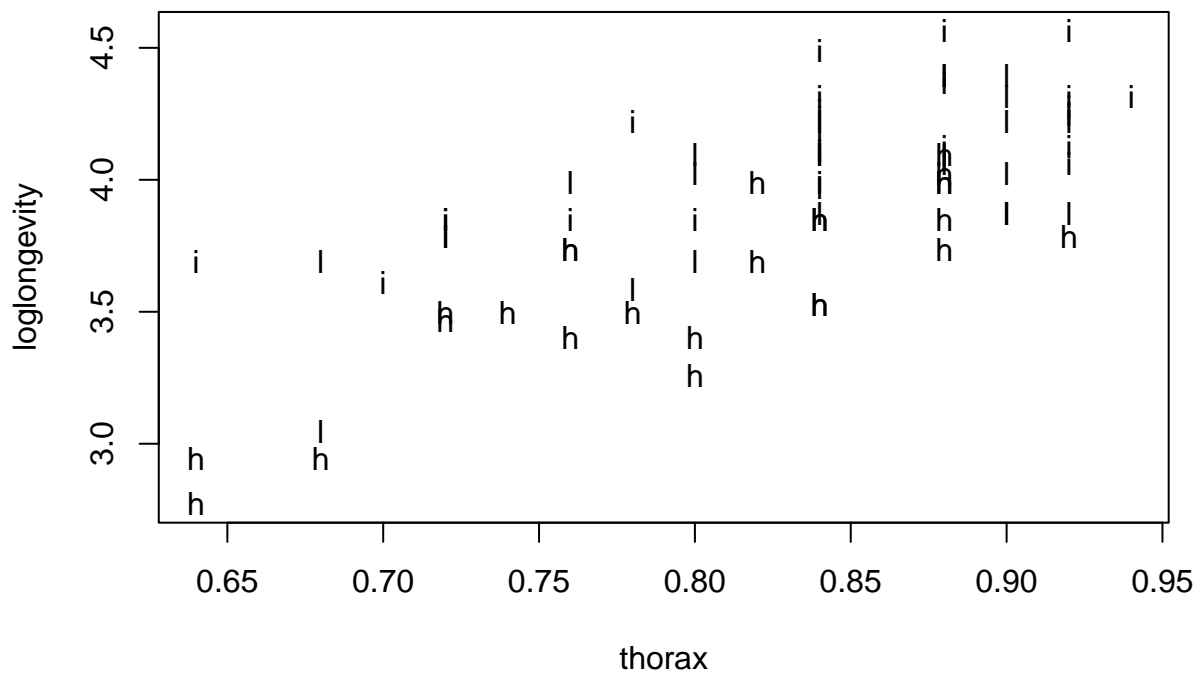
```
##
##  Shapiro-Wilk normality test
##
## data:  flies_df$loglongevity
## W = 0.95437, p-value = 0.008728
```

The original column is normal while the log of it is not.

```
boxplot(longevity~activity, data=flies_df)
```



```
plot(loglongevity~thorax, pch=as.character(activity), data=flies_df)
```



a) Make an informative plot of the data. Investigate whether sexual activity influences longevity by performing a statistical test, without taking the thorax length into account. What are the estimated longevitys for the three conditions? Comment.

```
if (!require("dplyr")) install.packages("dplyr")
```

```
## Loading required package: dplyr
```

```
##
## Attaching package: 'dplyr'

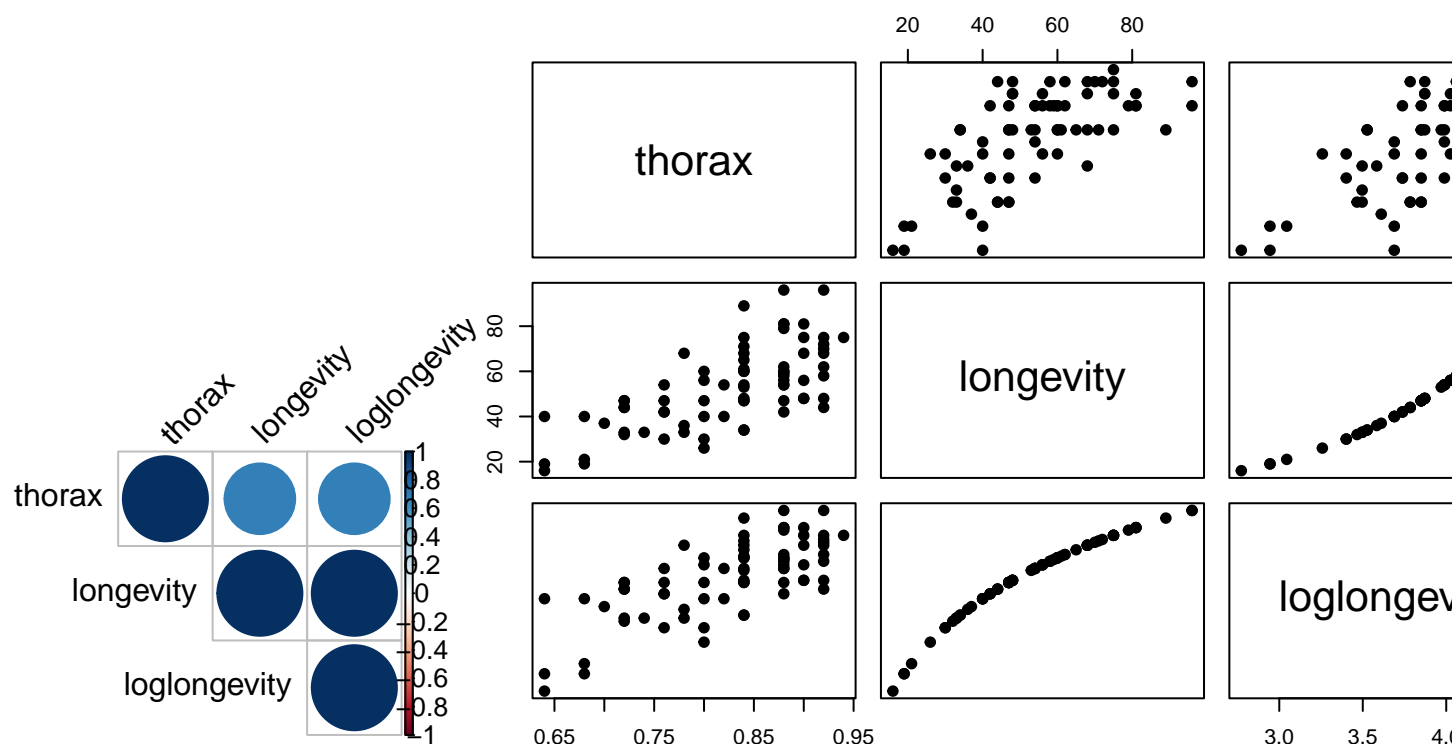
## The following object is masked from 'package:car':
##
##      recode

## The following objects are masked from 'package:stats':
##
##      filter, lag

## The following objects are masked from 'package:base':
##
##      intersect, setdiff, setequal, union

library(dplyr)

par(mfrow=c(2, 1))
res <- cor(select_if(flies_df, is.numeric), method='spearman')
corrplot(res, type = "upper", order = "hclust",
          tl.col = "black", tl.srt = 45)
pairs(select_if(flies_df, is.numeric), pch = 19)
```



```
flies_df$activity=as.factor(flies_df$activity)

flies_lm = lm(loglongevity ~ activity, data=flies_df)
anova(flies_lm)
```

```
## Analysis of Variance Table
```

```
##
## Response: loglongevity
##           Df Sum Sq Mean Sq F value    Pr(>F)
## activity   2 3.6665   1.8333   19.421 1.798e-07 ***
## Residuals 72 6.7966   0.0944
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

H0 is rejected. Sexual activity influences longevity.

```
summary(flies_lm)
```

```
##
## Call:
## lm(formula = loglongevity ~ activity, data = flies_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.95531 -0.13338  0.02552  0.20891  0.49222
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.60212    0.06145  58.621 < 2e-16 ***
## activityisolated 0.51722    0.08690   5.952 8.82e-08 ***
## activitylow     0.39771    0.08690   4.577 1.93e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3072 on 72 degrees of freedom
## Multiple R-squared:  0.3504, Adjusted R-squared:  0.3324
## F-statistic: 19.42 on 2 and 72 DF,  p-value: 1.798e-07
```

estimated longevitys for the three conditions:

```
high = 3.60212
isolated = high + 0.51722
low = high + 0.39711
high; isolated; low
```

```
## [1] 3.60212
## [1] 4.11934
## [1] 3.99923
```

b) Investigate whether sexual activity influences longevity by performing a statistical test, now including thorax length as an explanatory variable into the analysis. Does sexual activity increase or decrease longevity? What are the estimated longevitys for the three groups, for flies with the minimal and maximal thorax lengths?

```
ancova_lm = lm(loglongevity ~ activity + thorax, data=flies_df) ## order matters but we use dr
drop1(ancova_lm, test="F")
```

```
## Single term deletions
##
## Model:
## loglongevity ~ activity + thorax
##      Df Sum of Sq    RSS      AIC F value    Pr(>F)
## <none>                2.9180 -235.50
## activity  2      2.1129  5.0309 -198.64  25.705 4.000e-09 ***
## thorax    1      3.8786  6.7966 -174.08  94.374 1.139e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(ancova_lm)
```

```
##
## Call:
## lm(formula = loglongevity ~ activity + thorax, data = flies_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4858 -0.1612  0.0104  0.1510  0.3574
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.21893    0.24865   4.902 5.79e-06 ***
## activityisolated 0.40998    0.05839   7.021 1.07e-09 ***
## activitylow      0.28570    0.05849   4.885 6.18e-06 ***
## thorax          2.97899    0.30665   9.715 1.14e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2027 on 71 degrees of freedom
## Multiple R-squared:  0.7211, Adjusted R-squared:  0.7093
## F-statistic: 61.2 on 3 and 71 DF, p-value: < 2.2e-16
```

From the coefficients we can conclude that more sex - shorter life.

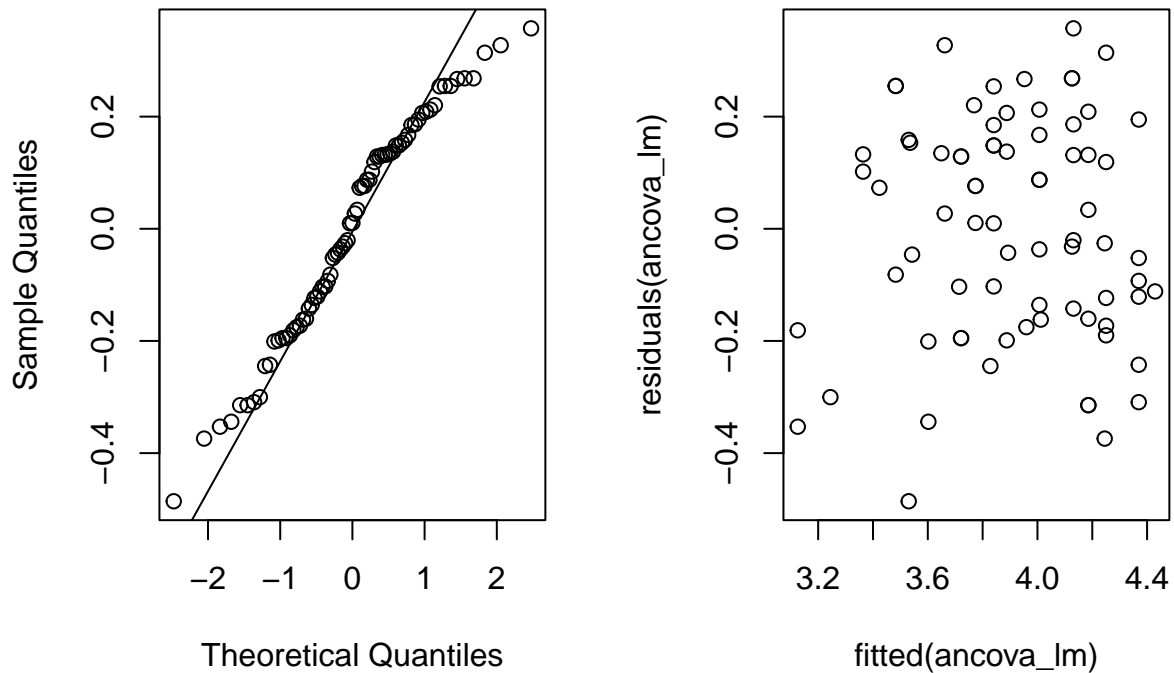
```
shapiro.test(residuals(ancova_lm))
```

```
##
## Shapiro-Wilk normality test
##
## data:  residuals(ancova_lm)
```

```
## W = 0.96838, p-value = 0.05748
```

```
par(mfrow=c(1, 2))
qqnorm(residuals(ancova_lm)); qqline(residuals(ancova_lm))
plot(fitted(ancova_lm), residuals(ancova_lm))
```

Normal Q-Q Plot



```
ancova_lm_int = lm(loglongevity ~ activity * thorax, data=flies_df)
anova(ancova_lm_int)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: loglongevity
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
activity	2	3.6665	1.8332	45.7687	2.228e-13 ***
thorax	1	3.8786	3.8786	96.8327	9.020e-15 ***
activity:thorax	2	0.1542	0.0771	1.9251	0.1536
Residuals	69	2.7638	0.0401		

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

There is no significant interaction between factor activity and predictor thorax.

```
summary(ancova_lm_int)
```

```
##
```

```
## Call:
```

```
## lm(formula = loglongevity ~ activity * thorax, data = flies_df)
```

```
##
```



```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49803 -0.15920 -0.00031  0.14624  0.35984
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.5978     0.4192   1.426   0.1584
## activityisolated    1.5465     0.5845   2.646   0.0101 *
## activitylow         0.9717     0.6423   1.513   0.1349
## thorax             3.7554     0.5216   7.199 5.78e-10 ***
## activityisolated:thorax -1.3929     0.7122  -1.956   0.0545 .
## activitylow:thorax    -0.8539     0.7794  -1.096   0.2771
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2001 on 69 degrees of freedom
## Multiple R-squared:  0.7359, Adjusted R-squared:  0.7167
## F-statistic: 38.44 on 5 and 69 DF,  p-value: < 2.2e-16
```

The interaction term is not significant. So, no indication that the initial analysis (ancova_lm without interaction) is in trouble.

```
# predict(ancova_lm, flies_df[which.min(flies_df$thorax), c('thorax', 'activity')])
# predict(ancova_lm, flies_df[which.max(flies_df$thorax), c('thorax', 'activity')])

min_thorax = min(flies_df$thorax)
max_thorax = max(flies_df$thorax)

# min_high = predict(ancova_lm, data.frame(thorax = min_thorax, activity = "high"))
min_high = 1.21893 + 2.97899 * min_thorax
max_high = 1.21893 + 2.97899 * max_thorax

min_iso = 1.21893 + 0.40998 + 2.97899 * min_thorax
max_iso = 1.21893 + 0.40998 + 2.97899 * max_thorax

min_low = 1.21893 + 0.28570 + 2.97899 * min_thorax
max_low = 1.21893 + 0.28570 + 2.97899 * max_thorax

exp(min_high); exp(max_high)

## [1] 22.7709
## [1] 55.65548
exp(min_iso); exp(max_iso)

## [1] 34.31092
## [1] 83.86099
```

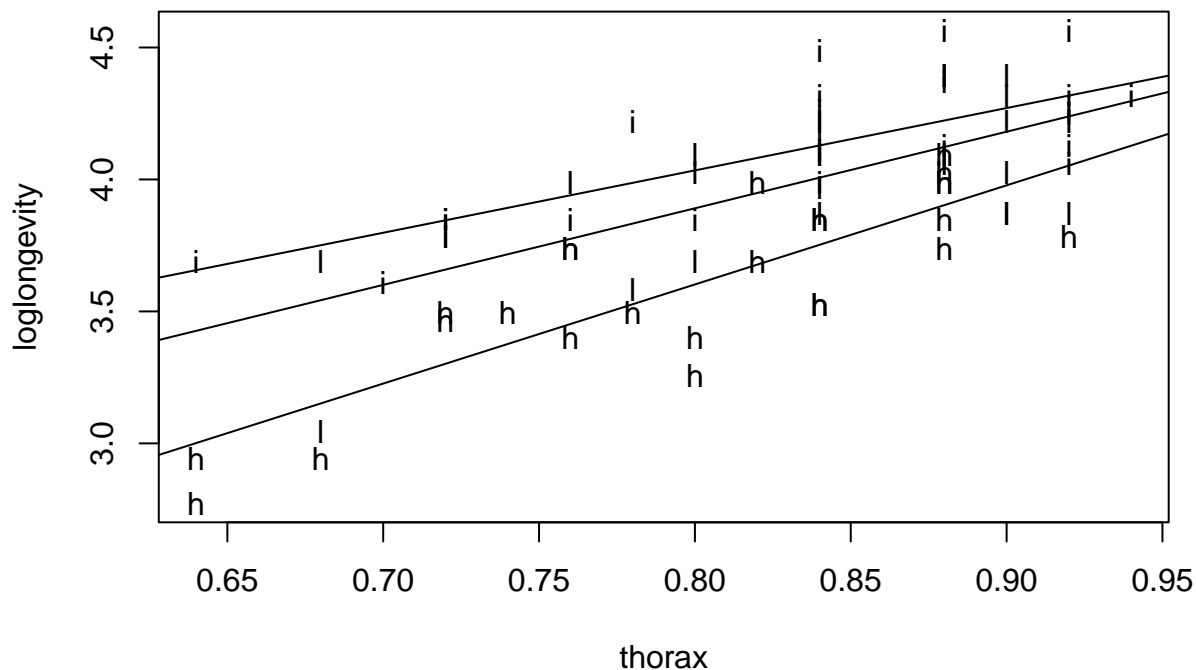
```
exp(min_low); exp(max_low)
```

```
## [1] 30.30109
```

```
## [1] 74.06037
```

c) How does thorax length influence longevity? Investigate graphically and by using an appropriate test whether this dependence is similar under all three conditions of sexual activity.

```
plot(loglongevity~thorax, pch=as.character(activity), data=flies_df)
for (i in unique(flies_df$activity)) abline(lm(loglongevity~thorax, data=flies_df[flies_df$activity==i,]))
```



```
ancova_lm_int = lm(loglongevity ~ activity * thorax, data=flies_df)
anova(ancova_lm_int)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: loglongevity
```

```
##          Df Sum Sq Mean Sq F value    Pr(>F)
## activity    2  3.6665   1.8332  45.7687 2.228e-13 ***
## thorax      1  3.8786   3.8786  96.8327 9.020e-15 ***
## activity:thorax 2  0.1542   0.0771   1.9251  0.1536
## Residuals  69  2.7638   0.0401
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Only the last p-value is relevant which always concerns interaction for models with interaction. We conclude from it that $H_0: \beta_1 = \beta_2$ is not rejected, i.e., there is no interaction between factor activity and predictor thorax.

d) Which of the two analyses, without or with thorax length, do you prefer? Is one of the analyses wrong?

Both analyses case to the conclusion that sexual activity influences longevity, more sex - shorter life.

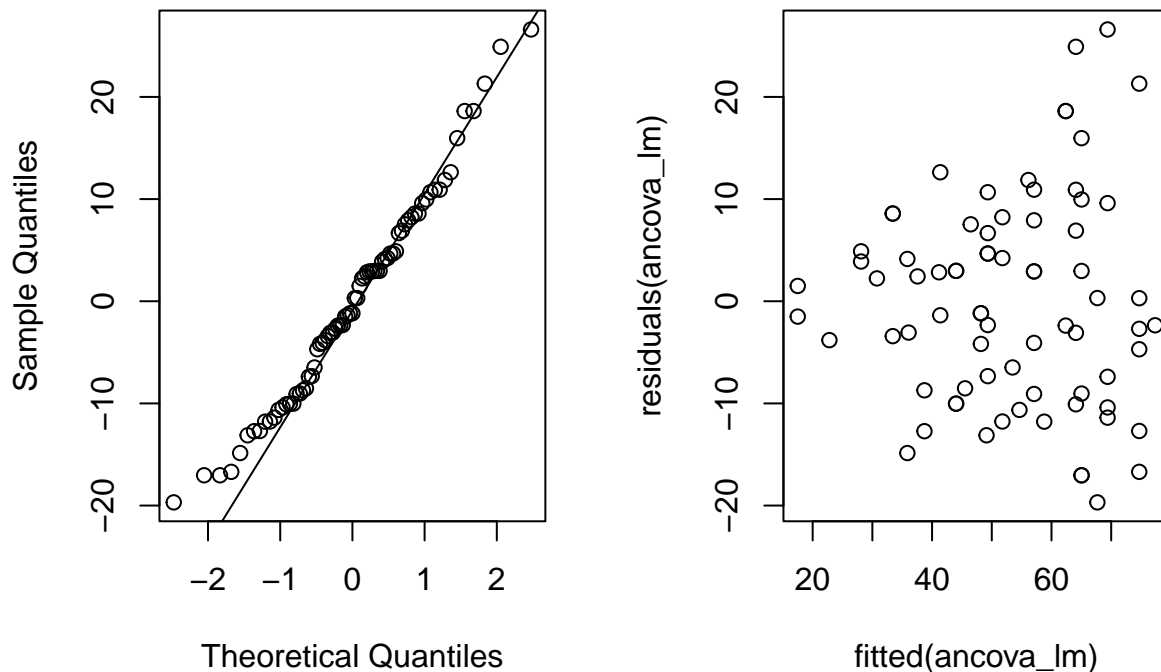
e) Perform the ancova analysis with the number of days as the response, rather than its logarithm. Was it wise to use the logarithm as response?

```
ancova_lm = lm(longevity ~ activity + thorax, data=flies_df) ## order matters but we use drop 1
drop1(ancova_lm, test="F")
```

```
## Single term deletions
##
## Model:
## longevity ~ activity + thorax
##           Df Sum of Sq  RSS   AIC F value    Pr(>F)
## <none>                 7673 355.10
## activity  2      4966.7 12640 388.53  22.979 2.016e-08 ***
## thorax    1      7686.8 15360 405.15  71.127 2.624e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow=c(1, 2))
qqnorm(residuals(ancova_lm)); qqline(residuals(ancova_lm))
plot(fitted(ancova_lm), residuals(ancova_lm))
```

Normal Q-Q Plot



```
shapiro.test(residuals(ancova_lm))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(ancova_lm)  
## W = 0.98091, p-value = 0.3176
```

Residuals are normal.

It's wise. For both columns the residuals are normal. If they were not, ANCOVA would not be relevant. For non-log model the residuals look non-homogenous.