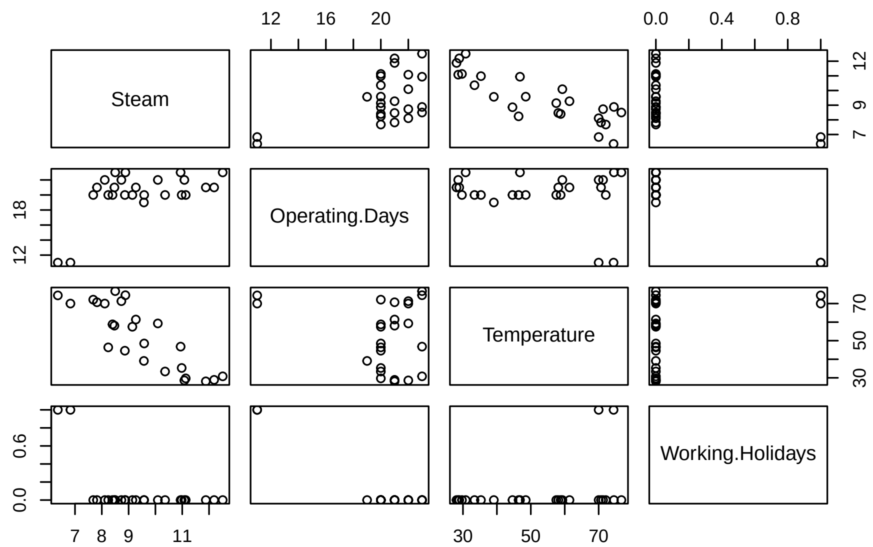
Solution to Series 2

1. Read in the data:

```
> library(MASS)
> library(robustbase)
> d.steam <- read.table("http://stat.ethz.ch/Teaching/Datasets/cas-das/dsteam.dat",
                        header = TRUE)
> d.data <- d.steam
```

a) We can see two peculiar observations in the pairs plot.

```
> plot(d.data)
```



b) We apply all the regression methods.

```
> r.ls <- lm(Steam ~ Operating.Days + Temperature, data = d.data)
> summary(r.ls)
Call:
lm(formula = Steam ~ Operating.Days + Temperature, data = d.data)

Residuals:
   Min     1Q Median     3Q    Max
-1.584 -0.438  0.119  0.482  0.948

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)      9.1269     1.1028    8.28  3.3e-08 ***
Operating.Days   0.2028     0.0458    4.43  0.00021 ***
Temperature     -0.0724     0.0080   -9.05  7.2e-09 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.662 on 22 degrees of freedom
Multiple R-squared:  0.849,       Adjusted R-squared:  0.835
F-statistic: 61.9 on 2 and 22 DF,  p-value: 9.23e-10
```

```
> r.MM <- lmrob(Steam ~ Operating.Days + Temperature, data = d.data)
> summary(r.MM)
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     3.087049   1.419653   2.175   0.0407 *
Operating.Days  0.514717   0.071214   7.228 3.05e-07 ***
Temperature    -0.082858   0.006305 -13.142 6.80e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(optimal)

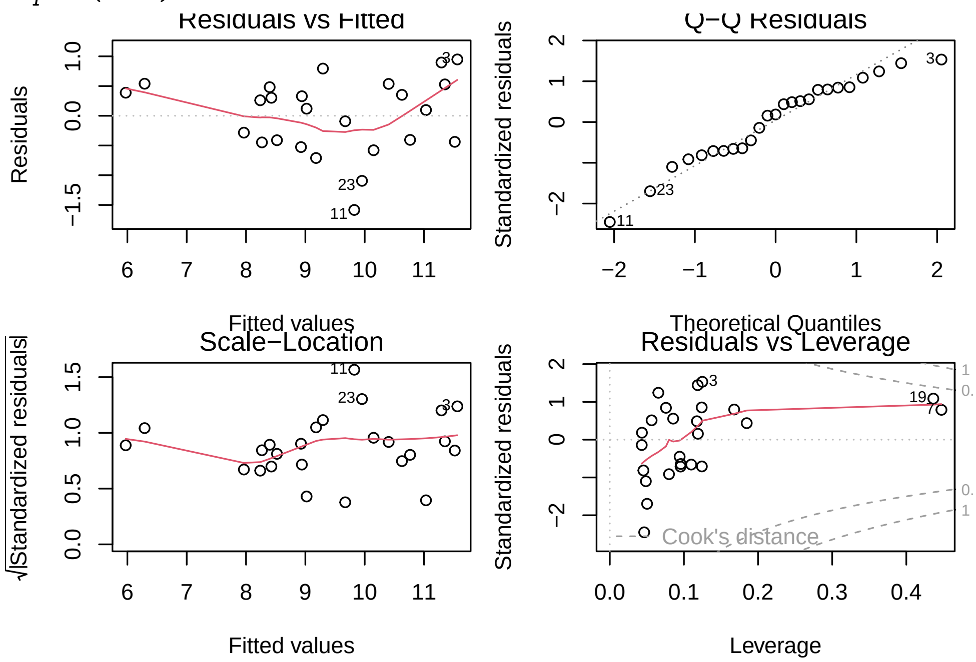
```
> r.M <- rlm(Steam ~ Operating.Days + Temperature, method = "M", data = d.data)
> summary(r.M)
Coefficients:
               Value   Std. Error t value
(Intercept)     9.2502  1.0692      8.6514
Operating.Days  0.2008  0.0444      4.5247
Temperature    -0.0733  0.0078     -9.4487
```

All the methods show highly significant explanatory variables. They differ however in the coefficients. The least squares method and the M-method yield pretty much the same results. The MM-method however shows very different results.

c) Residual analysis:

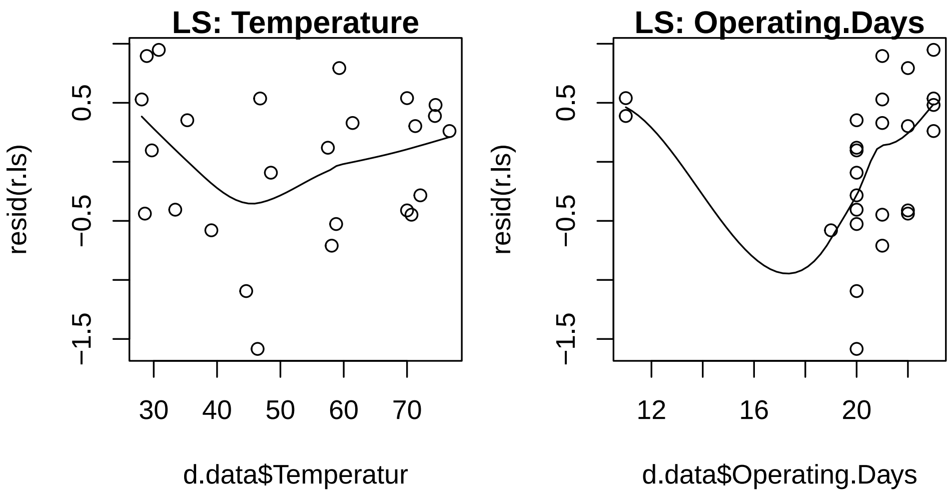
- Least Squares method: The TA-plot shows a weak systematic deviation from zero. The normal Q-Q-plot and the scale-location plot seem to be ok. In the leverage plot, observations 7 and 19 stand out. However, they cannot be identified as very influential since Cook's distance is smaller than 0.5.

```
> par(mfrow = c(2, 2))
> plot(r.ls)
```



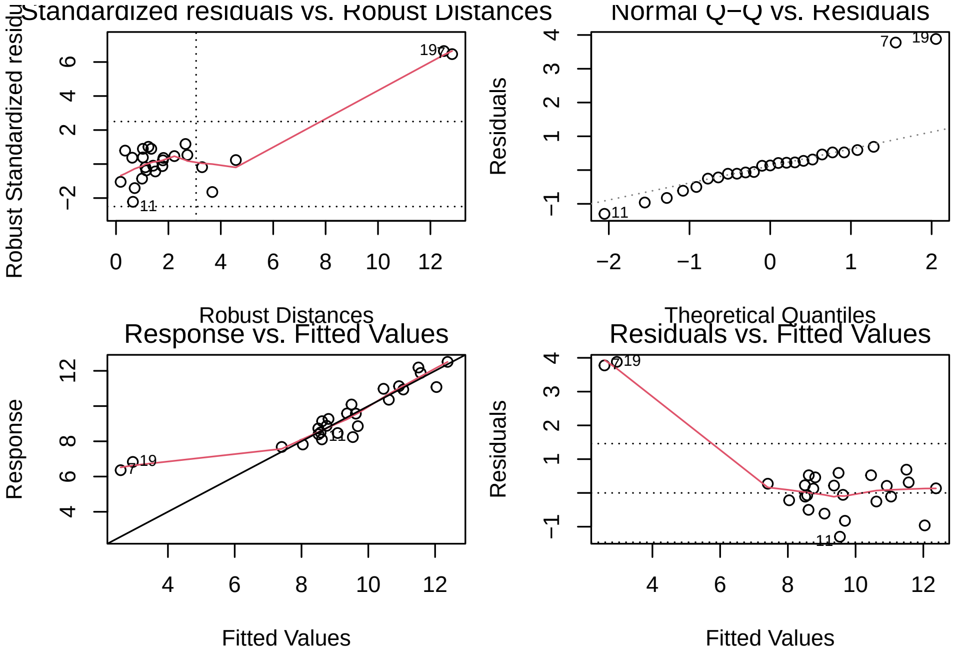
Variables vs. residuals-plots:

```
> par(mfrow = c(1, 2))
> scatter.smooth(d.data$Temperatur, resid(r.ls) ,main = "LS: Temperature")
> scatter.smooth(d.data$Operating.Days, resid(r.ls), main = "LS: Operating.Days")
```

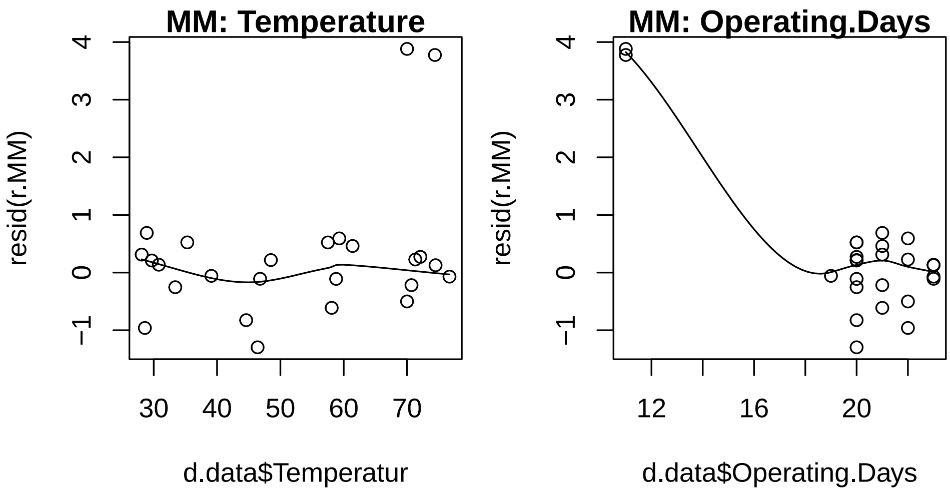


- Robust (MM-method):

```
> par(mfrow = c(2, 2))
> plot(r.MM)
```




```
> par(mfrow = c(1, 2))
> scatter.smooth(d.data$Temperatur, resid(r.MM), main = "MM: Temperature")
> scatter.smooth(d.data$Operating.Days, resid(r.MM), main = "MM: Operating.Days")
```



The Q-Q-plot shows two outliers. TA and variables-residuals-plots show similar behaviour as in the method above, except that the two outliers are not taken into account. The constant variance assumption does not hold for these two outliers.

This comparison shows that the two methods yield different results. Using the MM-method identifies the outliers clearly. The two observations with a small value in `Operating.Days` are recognized as outliers/leverage points and dealt with accordingly. The normal regression does not recognize the leverage points as being influential (="evil") at all.

d) Both regression at once:

```
> r.ls2 <- lm(Steam ~ Operating.Days + Temperature + Working.Holidays, data = d.data)
> summary(r.ls2)

Call:
lm(formula = Steam ~ Operating.Days + Temperature + Working.Holidays,
    data = d.data)

Residuals:
    Min      1Q  Median      3Q     Max
-1.2521 -0.1990  0.0541  0.3009  0.7344

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)        2.8122     1.9707    1.43   0.1683
Operating.Days     0.5247     0.0970    5.41  2.3e-05 ***
Temperature       -0.0822     0.0070  -11.74  1.1e-10 ***
Working.Holidays   3.9467     1.0994    3.59   0.0017 **
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.533 on 21 degrees of freedom
Multiple R-squared:  0.906,      Adjusted R-squared:  0.893
F-statistic: 67.9 on 3 and 21 DF,  p-value: 5.65e-11

> r.MM2 <- lmrob(Steam ~ Operating.Days + Temperature + Working.Holidays, data = d.data)
> summary(r.MM2)

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       3.210554   1.376357   2.333   0.0297 *
Operating.Days    0.510752   0.068620   7.443 2.57e-07 ***
Temperature      -0.083318   0.006296 -13.233 1.18e-11 ***
Working.Holidays  3.781709   0.781322   4.840 8.75e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

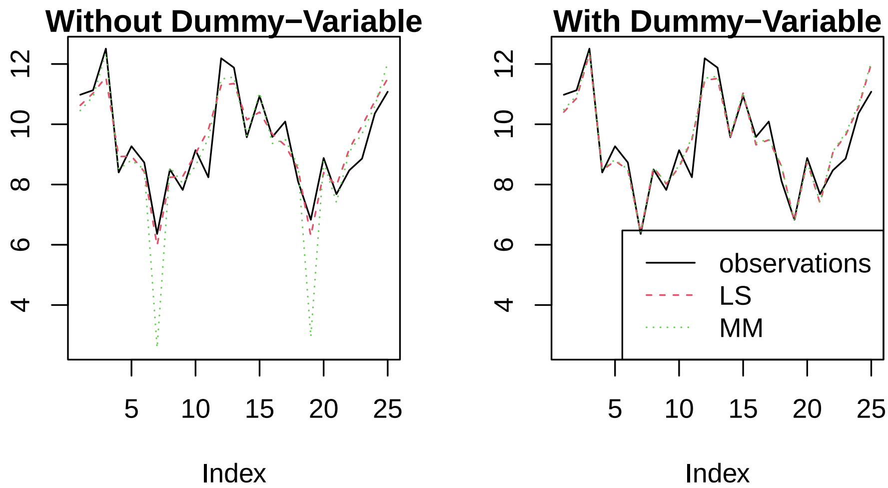
All the explanatory variables are significant. Both methods yield similar coefficients. Note that the coefficients do not differ much from the MM-coefficients in **b)**.
Residual analysis (the plots are not shown due to space limitation in the solutions):
The TA and the residual plots look the same for both methods. They look ok.
The Q-Q-plots also look ok, although they are a bit skewed to the left.
One could expect correlated residuals, but the acf and pacf (see time series analysis) do not show suspicious behaviour. However the series is too short to be sure:

```
> plot(resid(r.ls2)); acf(resid(r.ls2)); pacf(resid(r.ls2))
> plot(resid(r.MM2)); acf(resid(r.MM2)); pacf(resid(r.MM2))
```

e) We compare the results of the two regression models: the regression model without the dummy variable `Working.Holidays` (see left plot) and the regression model with the dummy variable `Working.Holidays` (see right plot).

```
> par(mfrow = c(1, 2))
> t.ylim <- range(d.data$Steam, fitted(r.ls), fitted(r.MM),
     fitted(r.ls2))
> plot(d.data$Steam, type = "l", ylim = t.ylim,
```

```
    ylab = "")
> lines(fitted(r.ls), lty = 2, col = 2)
> lines(fitted(r.MM), lty = 3, col = 3)
> title("Without Dummy-Variable")
> plot(d.data$Steam, type = "l", ylim = t.ylim,
    ylab = "")
> lines(fitted(r.ls2), lty = 2, col = 2)
> lines(fitted(r.MM2), lty = 3, col = 3)
> title("With Dummy-Variable")
> legend("bottomright", legend = c("observations",
    "LS", "MM"), lty = 1:3, col = 1:3)
```



Model without the dummy variable `Working.Holidays`: the robust fit shows that the model does not fit well at the extremely low values.
Model with the dummy variable `Working.Holidays`: the "fitted values" hardly differ from the observations.

2. Reading in the data:

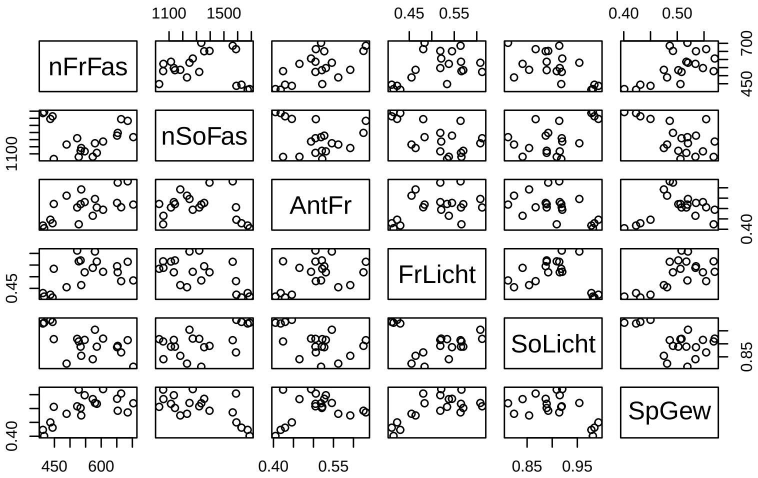
```
> library(MASS)
> library(robustbase)
> d.data.org <- read.table("http://stat.ethz.ch/Teaching/Datasets/cas-das/woodRous.dat",
                           header = TRUE)
> d.data <- d.data.org
> d.data.nichtmod <- read.table("http://stat.ethz.ch/Teaching/Datasets/cas-das/wood.dat",
                                header = TRUE)
```

a) Transformations:

```
> d.data[, 1:2] <- sqrt(d.data[, 1:2])
> d.data[, 3:5] <- asin(sqrt(d.data[, 3:5]))
```

b) We can see the four outliers in the pairs-plot as a group of four.

```
> plot(d.data.org)
```



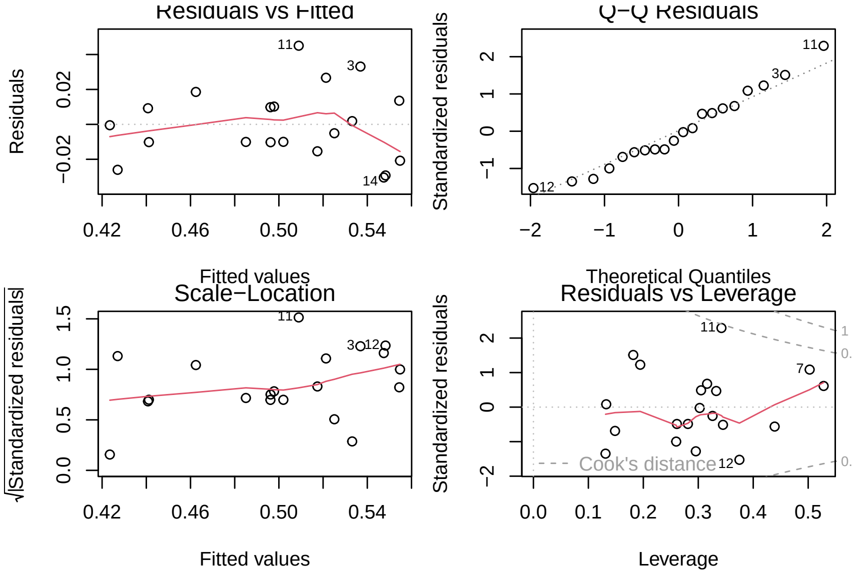
c) The variables `FrLicht` and `SoLicht` are not significant.

```
> lm1 <- lm(SpGew ~ ., data = d.data)
> summary(lm1)
```

```
Call:
lm(formula = SpGew ~ ., data = d.data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.03045 -0.01153 -0.00279  0.01103  0.04503 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.47129    0.18538    2.54   0.0235 *
nFrFas       0.01999    0.00561    3.56   0.0031 **
nSoFas      -0.00969    0.00358   -2.71   0.0170 *
AntFr       -0.26015    0.11317   -2.30   0.0374 *
FrLicht      0.05586    0.14959    0.37   0.7144
SoLicht      0.05987    0.11188    0.54   0.6009
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0242 on 14 degrees of freedom
Multiple R-squared:  0.807,	Adjusted R-squared:  0.737
F-statistic: 11.7 on 5 and 14 DF,  p-value: 0.000137
```

A residual analysis does not show anything special.

```
> par(mfrow = c(2, 2))
> plot(lm1)
```



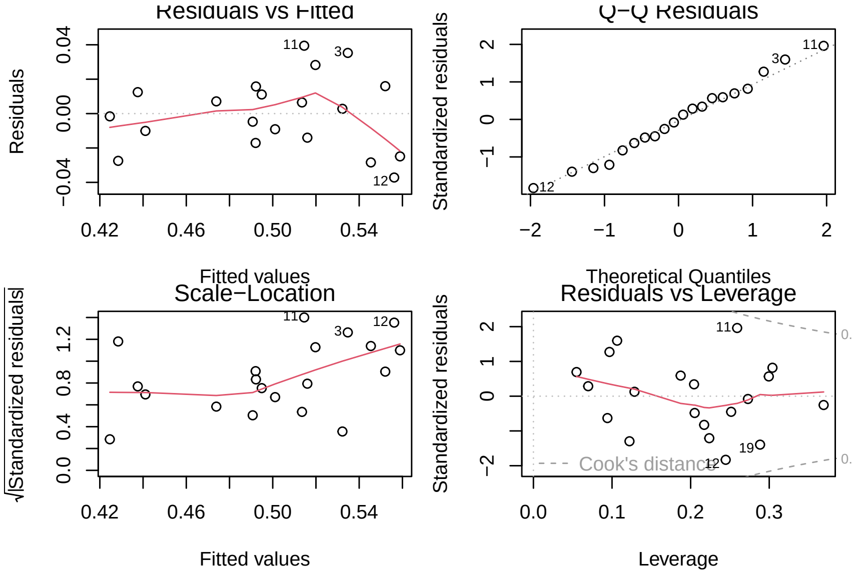
Using step(), we reduce the model to our final model. The above mentioned not significant variables are actually dropped.

```
> lm2 <- step(lm1, direction = "backward", trace = FALSE)
> summary(lm2)
```

```
Call:
lm(formula = SpGew ~ nFrFas + nSoFas + AntFr, data = d.data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.03719 -0.01478  0.00058  0.01329  0.03946 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.60886    0.10947    5.56  4.3e-05 ***
nFrFas       0.01912    0.00350    5.47  5.2e-05 ***
nSoFas      -0.00933    0.00183   -5.10  0.00011 ***
AntFr       -0.27162    0.10619   -2.56  0.02106 *  
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0234 on 16 degrees of freedom
Multiple R-squared:  0.794,	Adjusted R-squared:  0.756 
F-statistic: 20.6 on 3 and 16 DF,  p-value: 9.68e-06
```

A residual analysis again does not show anything special.

```
> par(mfrow = c(2, 2))
> plot(lm2)
```

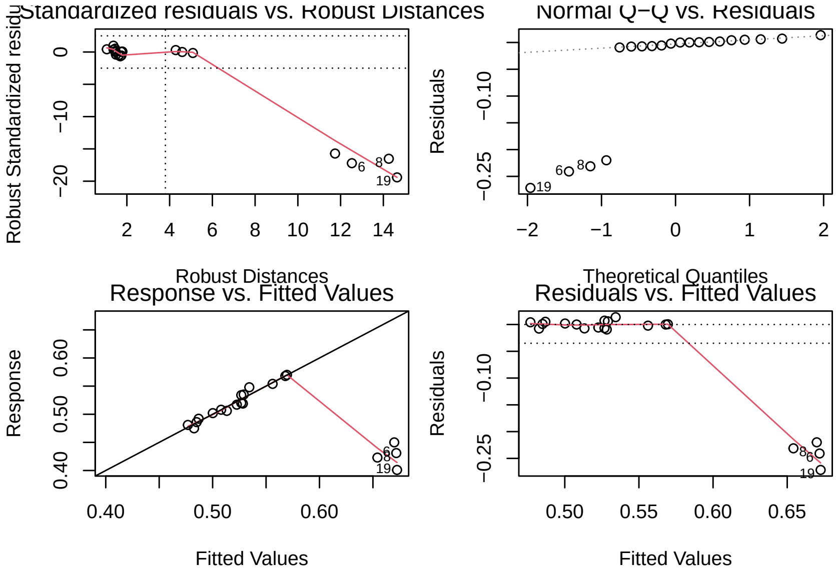


d) Robust regression: We find, that `nSoFas` is not significant. We omit this variable and compare the two models using `anova`:

```
> r1 <- lmrob(SpGew ~ ., data = d.data)
> summary(r1)
> r2 <- lmrob(SpGew ~ . -nSoFas, data = d.data)
> summary(r2)
> anova(r1, r2, test = "Wald")
Robust Wald Test Table

Model 1: SpGew ~ nFrFas + nSoFas + AntFr + FrLicht + SoLicht
Model 2: SpGew ~ (nFrFas + nSoFas + AntFr + FrLicht + SoLicht) - nSoFas
Largest model fitted by lmrob(), i.e. SM

  pseudoDf Test.Stat Df Pr(>chisq)
1       14
2       15      0.18  1       0.67
> anova(r1, r2, test = "Deviance")
Robust Deviance Table

Model 1: SpGew ~ nFrFas + nSoFas + AntFr + FrLicht + SoLicht
Model 2: SpGew ~ (nFrFas + nSoFas + AntFr + FrLicht + SoLicht) - nSoFas
Largest model fitted by lmrob(), i.e. SM

  pseudoDf Test.Stat Df Pr(>chisq)
1       14
2       15     0.126  1       0.72
```

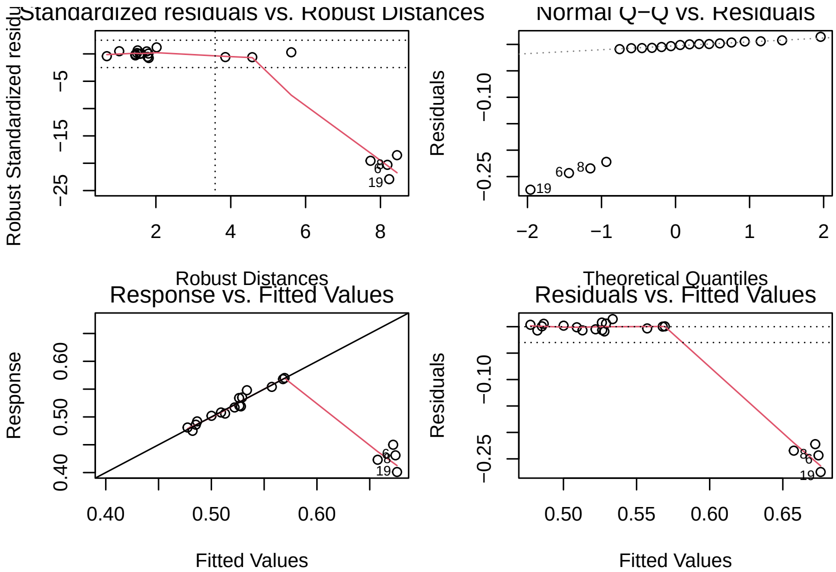
Both tests show that the variable `nSoFas` is not significant and is thus not needed in the final model.
Residual analysis for the first model: the four outliers 4, 6, 8, 19 can be seen.

```
> par(mfrow = c(2, 2))
> plot(r1, which = 1:4)
> par(mfrow = c(1, 1))
```



Residual analysis for the second model: the four outliers 4, 6, 8, 19 can also be seen.

```
> par(mfrow = c(2, 2))
> plot(r2, which = 1:4)
> par(mfrow = c(1, 1))
```



A further reduction of the model is not appropriate since all the remaining variables are highly significant and thus needed. We get a model with four explanatory variables instead of three as suggested by LS.

e) The LS suggests three explanatory variables and does not find the outliers. The robust method finds the outliers and needs four explanatory variables.

f) We check the values where the absolute value of the standartized residuals is the largest. In the LS case this are the observations 3, 11 and 12. However this residuals are not extreme. We do not call them outliers. The MM estimator suggest that 4, 6, 8 and 19 are outlieres. This are indeed the mutated observations.

```
> d.data.nichtmod[, 1:2] <- sqrt(d.data.nichtmod[, 1:2])
> d.data.nichtmod[, 3:5] <- asin(sqrt(d.data.nichtmod[, 3:5]))
> d.data.nichtmod[c(3, 11, 12), ]
```

```
   nFrFas nSoFas AntFr FrLicht SoLicht SpGew
3    24.6   35.7 0.779   0.806    1.28 0.570
11   25.8   39.8 0.791   0.766    1.20 0.554
12   26.5   36.5 0.804   0.769    1.12 0.519
```

```
> d.data[c(3, 11, 12), ]   # suggested by LS
```

```
   nFrFas nSoFas AntFr FrLicht SoLicht SpGew
3    24.6   35.7 0.779   0.806    1.28 0.570
11   25.8   39.8 0.791   0.766    1.20 0.554
12   26.5   36.5 0.804   0.769    1.12 0.519
```

```
> d.data.nichtmod[c(4, 6, 8, 19), ]
```

```
   nFrFas nSoFas AntFr FrLicht SoLicht SpGew
4    25.1   33.9 0.774   0.788    1.22 0.528
6    23.6   35.2 0.834   0.837    1.27 0.555
8    26.2   39.5 0.852   0.728    1.27 0.516
19   21.8   32.5 0.714   0.817    1.30 0.595
```

```
> d.data[c(4, 6, 8, 19), ]    # suggested by MM
```

```
   nFrFas nSoFas AntFr FrLicht SoLicht SpGew
4    20.9   39.9 0.731   0.708    1.48 0.450
6    21.1   40.3 0.714   0.696    1.44 0.431
8    20.3   40.9 0.703   0.715    1.42 0.423
19   20.4   41.1 0.690   0.700    1.43 0.401
```

g) i)

```
> lm.rm <- lm(SpGew ~ ., data = d.data, subset = -c(4, 6, 8, 19))
> lm.without <- step(lm.rm, direction = "backward", trace = FALSE)
> lm.without
```

```
Call:
lm(formula = SpGew ~ nFrFas + AntFr + FrLicht + SoLicht, data = d.data,
    subset = -c(4, 6, 8, 19))

Coefficients:
(Intercept)       nFrFas        AntFr      FrLicht
    0.60433      0.00984     -0.56662     -0.39616
    SoLicht
    0.37707
```

We get the same model as with robust regression.

ii)

```
> library(RobStatTM)
> h.cont <- lmrobdet.control(bb=0.5, efficiency=0.85, family="bisquare")
> fit.rlm1 <- lmrobdetMM(SpGew~. , data=d.data.nichtmod, control=h.cont)
> step.lmrobdetMM(fit.rlm1)
```

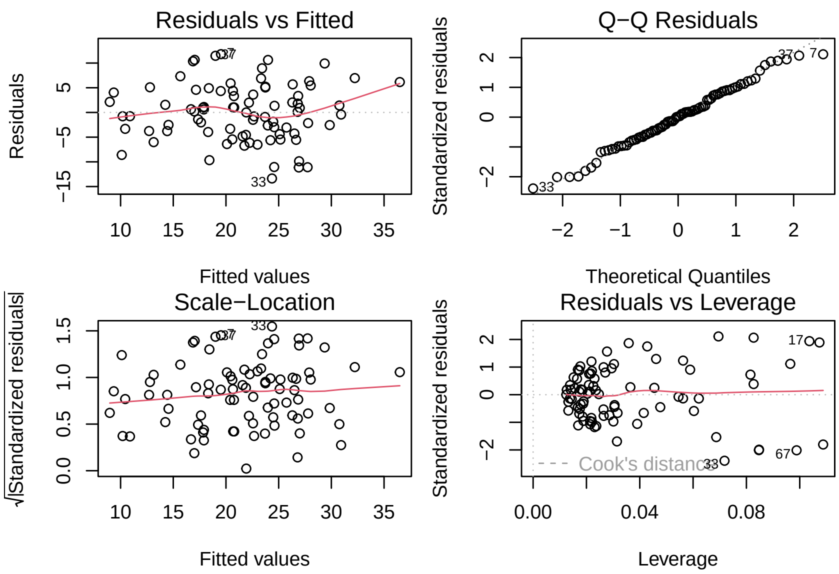
```
Start:  RFPE= 0.227
 SpGew ~ nFrFas + nSoFas + AntFr + FrLicht + SoLicht


Single term deletions

Model:
SpGew ~ nFrFas + nSoFas + AntFr + FrLicht + SoLicht

scale:  0.0141

        Df  RFPE
<none>     0.227
nFrFas   1 0.317
nSoFas   1 0.218
AntFr    1 0.982
FrLicht  1 0.316
SoLicht  1 0.369

Step:  RFPE = 0.218
 SpGew ~ nFrFas + AntFr + FrLicht + SoLicht
```

```
Single term deletions

Model:
SpGew ~ nFrFas + AntFr + FrLicht + SoLicht

scale:  0.0141

        Df  RFPE
<none>     0.218
nFrFas   1 0.413
AntFr    1
FrLicht  1 0.342
SoLicht  1 0.423

Call:
lmrobdetMM(formula = SpGew ~ nFrFas + AntFr + FrLicht + SoLicht, data = d.data.nichtmod,     c
Coefficients:
(Intercept)       nFrFas        AntFr      FrLicht
    0.61882      0.00918     -0.57810     -0.41333
    SoLicht
    0.39724
```

iii) The same variables were selected.

3. **a)**

```
> D.synt <- read.table("http://stat.ethz.ch/Teaching/Datasets/cas-das/Synthetisch.dat",
                       header = TRUE)
> D.synt.lm <- lm(y ~ x1 + x2, data = D.synt)
> summary(D.synt.lm)
Call:
lm(formula = y ~ x1 + x2, data = D.synt)

Residuals:
    Min      1Q  Median      3Q     Max
-13.367  -3.869   0.117   4.356  11.802

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   72.902      9.648    7.56  6.0e-11 ***
x1            -2.084      0.488   -4.27  5.4e-05 ***
x2             1.426      0.183    7.80  2.0e-11 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.8 on 80 degrees of freedom
Multiple R-squared:  0.496,        Adjusted R-squared:  0.484
F-statistic: 39.4 on 2 and 80 DF,  p-value: 1.23e-12
> par(mfrow = c(2, 2))
> plot(D.synt.lm)
```



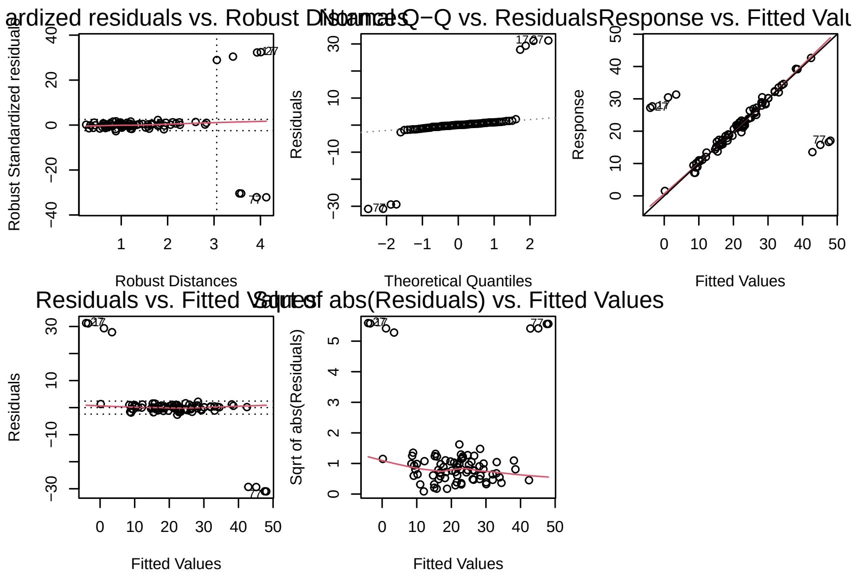
We cannot see anything special in the residual analysis. The expected value is constant and zero, the variance seems to be constant, Q-Q-plot shows no deviation of a normal distribution and we cannot see any leverage points.

b)

```
> library(robustbase)
> D.synt.rlm <- lmrob(y ~ x1 + x2, data = D.synt, setting = "KS2014")
> summary(D.synt.rlm)
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.56336    2.10307   3.121  0.00251 **
x1           1.90517    0.11410  16.698  < 2e-16 ***
x2           2.94858    0.04303  68.516  < 2e-16 ***
---
Robust residual standard error: 0.9641
```

```
> par(mfrow = c(2, 3))
> plot(D.synt.rlm)
```



We can see 8 clear outliers.

c) The two methods yield very different estimations of the parameters. We even get different signs. Also the standard errors are different.

When using the robust method, we can see 8 very prominent outliers. When we use the classical approach, there is not a slight hint that there might be problems in the data.

If we omit the outliers, the result do not differ much and are similar to the results of the robust methods with the whole dataset.

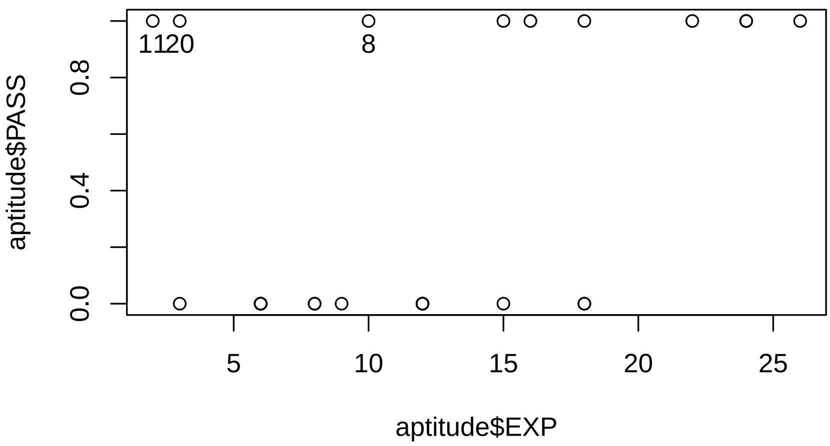
4. a) Logistic Regression

b)
```
> aptitude <- read.table("http://stat.ethz.ch/Teaching/Datasets/cas-das/aptitude.dat",
                         header = TRUE)
> fit.glm <- glm(PASS ~ ., data = aptitude, family = binomial)
> summary(fit.glm)
```

```
Call:
glm(formula = PASS ~ ., family = binomial, data = aptitude)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -3.1657     1.4245   -2.22    0.026 *
SCORE         0.2876     0.3009    0.96    0.339
EXP           0.1469     0.0727    2.02    0.043 *
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 34.646  on 25  degrees of freedom
Residual deviance: 28.879  on 23  degrees of freedom
AIC: 34.88

Number of Fisher Scoring iterations: 4
```

c)
```
> plot(aptitude$EXP, aptitude$PASS)
> identify(aptitude$EXP, aptitude$PASS)
```

```
> plot(aptitude$EXP, aptitude$PASS)
> text(aptitude[c(8, 11, 20), 2:3], labels = c("8", "11", "20"), pos = 1)
```



The observations 8, 11, and 20 (or maybe only 11 and 20) have an unexpected PASS-value (1) for their value of EXP, so they could be seenn as "outliers".

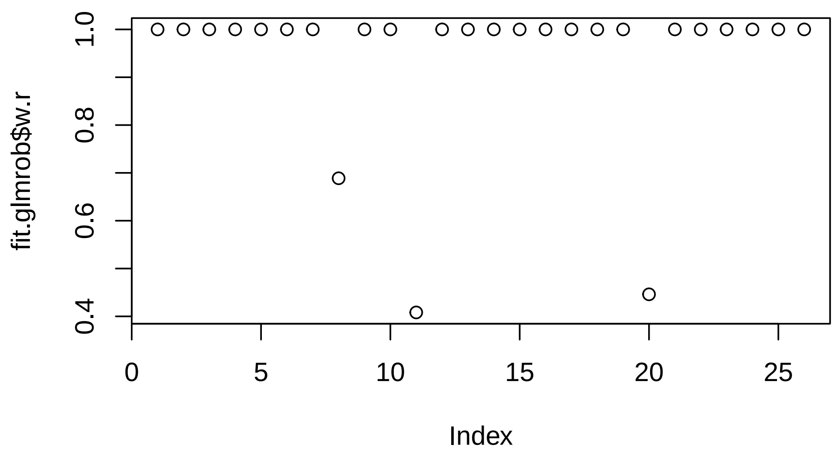
d)
```
> library(robustbase)
> fit.glmrob <- glmrob(PASS ~ ., family = binomial, method = "Mqle", data = aptitude)
> summary(fit.glmrob)
```

```
Call:  glmrob(formula = PASS ~ ., family = binomial, data = aptitude,      method = "Mqle")


Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -3.2051     1.4837   -2.16    0.031 *
SCORE         0.0931     0.3279    0.28    0.776
EXP           0.1773     0.0820    2.16    0.030 *
---
```

```
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Robustness weights w.r * w.x:
 23 weights are ~= 1. The remaining 3 ones are
    8    11    20
0.689 0.408 0.446

Number of observations: 26
Fitted by method 'Mqle'  (in 16 iterations)

(Dispersion parameter for binomial family taken to be 1)

No deviance values available
Algorithmic parameters:
   acc    tcc
0.0001 1.3450
maxit
   50
test.acc
  "coef"
```

The estimate for SCORE got smaller, the one for EXP got larger and "more significant".

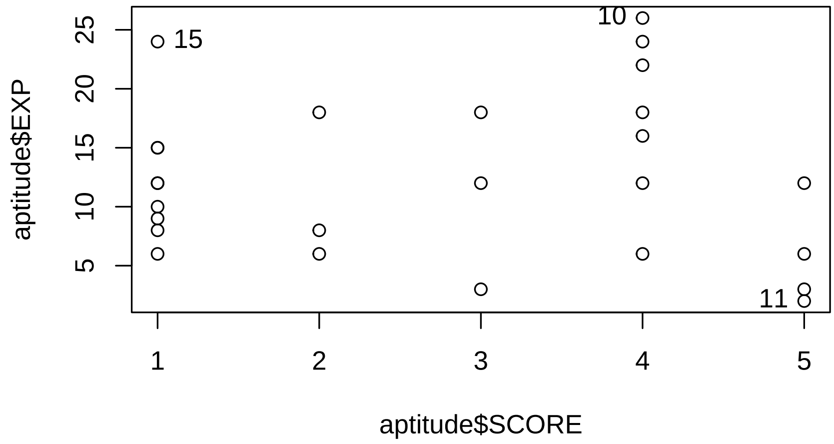
e) *> plot(fit.glmrob$w.r)*



The observations 11, 20 and 8 really got the smallest weights (which makes sense when looking at the plot in **c)**, as these will have the largest residuals when fitting regression curve).

f) When $w(x_i) = 1$ for all observations i, then the influence of a leverage point is not bounded. This is a problem if there are leverage points (points with extreme x-values resp. an unusual combination of x-values). Since there are only two explanatory variables we can check this by plotting them against each other:

> plot(aptitude$SCORE, aptitude$EXP)
> identify(aptitude$SCORE, aptitude$EXP)



The most "extreme" values are 10, 11 and 15, but they don't seem very "unusual" compared to the other points.

5. **a)** Poisson Regression

b)
- The interaction `Age10:Trt` could be present if the treatment effect on the total sum of attacks is different for different ages.
- The interaction `Base4:Trt` could also be present if the effect of the treatment on the total sum of attacks is different depending on how "severe" your epilepsy was before.

We thus include both interactions.

```
> library(robustbase)
> data(epilepsy)
> fit1 <- glm(Ysum ~ Age10 + Base4 + Trt + Age10:Trt + Base4:Trt, family = poisson,
              data = epilepsy)
> summary(fit1)
```

```
Call:
glm(formula = Ysum ~ Age10 + Base4 + Trt + Age10:Trt + Base4:Trt,
    family = poisson, data = epilepsy)

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.76751    0.17818    9.92  < 2e-16 ***
Age10               0.31020    0.05589    5.55  2.9e-08 ***
Base4               0.08544    0.00367   23.30  < 2e-16 ***
Trtprogabide        0.18176    0.25978    0.70    0.484
Age10:Trtprogabide -0.14561    0.08280   -1.76    0.079 .
Base4:Trtprogabide  0.00567    0.00452    1.25    0.210
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 2122.73  on 58  degrees of freedom
Residual deviance:  553.42  on 53  degrees of freedom
AIC: 848.7

Number of Fisher Scoring iterations: 5
```

Only `Age10` and `Base4` seem to have a significant influence.

c)
```
> fit2 <- glmrob(Ysum ~ Age10 + Base4 + Trt + Age10:Trt + Base4:Trt, family = poisson,
                 data = epilepsy, method = "Mqle", weights.on.x = "hat")
> summary(fit2)
```

```
Call:  glmrob(formula = Ysum ~ Age10 + Base4 + Trt + Age10:Trt + Base4:Trt,      family = pois


Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.93939    0.19423    9.99   <2e-16 ***
Age10               0.17455    0.06203    2.81   0.0049 **
Base4               0.09750    0.00398   24.52   <2e-16 ***
Trtprogabide       -0.37522    0.28908   -1.30   0.1943
Age10:Trtprogabide  0.02937    0.09254    0.32   0.7510
Base4:Trtprogabide  0.00559    0.00515    1.09   0.2772
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Robustness weights w.r * w.x:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.047   0.393   0.710   0.605   0.865   0.928

Number of observations: 59
Fitted by method ‘Mqle’  (in 34 iterations)

(Dispersion parameter for poisson family taken to be 1)

No deviance values available
Algorithmic parameters:
   acc    tcc
0.0001 1.3450
maxit
   50
test.acc
  "coef"
```

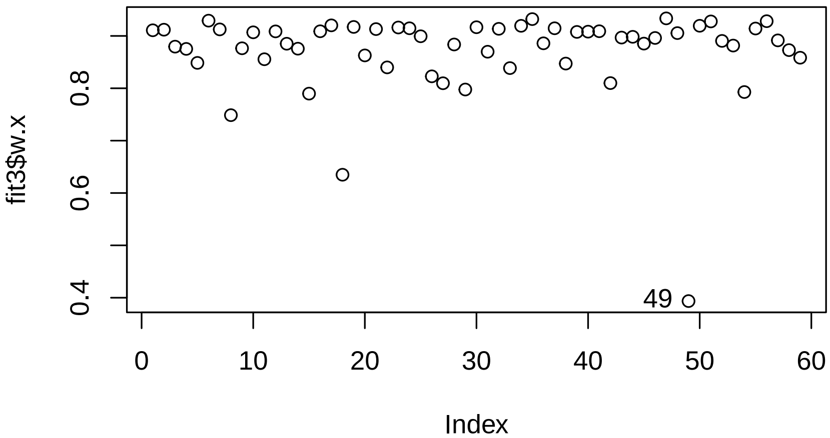
d) Interaction `Age10:Trt`:

```
> fit3 <- glmrob(Ysum~ Age10 + Base4 + Trt, family = poisson,
                 data = epilepsy, method = "Mqle", weights.on.x = "hat")
> anova(fit2, fit3, test = "QD")
Robust Quasi-Deviance Table

Model 1: Ysum ~ Age10 + Base4 + Trt + Age10:Trt + Base4:Trt
Model 2: Ysum ~ Age10 + Base4 + Trt
Models fitted by method 'Mqle'

  pseudoDf Test.Stat Df Pr(>chisq)
1       53
2       55    -0.983 -2       0.61
> summary(fit3)
Call:  glmrob(formula = Ysum ~ Age10 + Base4 + Trt, family = poisson,      data = epilepsy, me


Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   2.00608    0.14961   13.41  < 2e-16 ***
Age10         0.14983    0.04495    3.33  0.00086 ***
Base4         0.09557    0.00223   42.92  < 2e-16 ***
Trtprogabide -0.22575    0.05336   -4.23  2.3e-05 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Robustness weights w.r * w.x:
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.080   0.420   0.779   0.647   0.898   0.929

Number of observations: 59
Fitted by method 'Mqle'  (in 18 iterations)

(Dispersion parameter for poisson family taken to be 1)

No deviance values available
Algorithmic parameters:
   acc    tcc
0.0001 1.3450
maxit
   50
test.acc
  "coef"
```

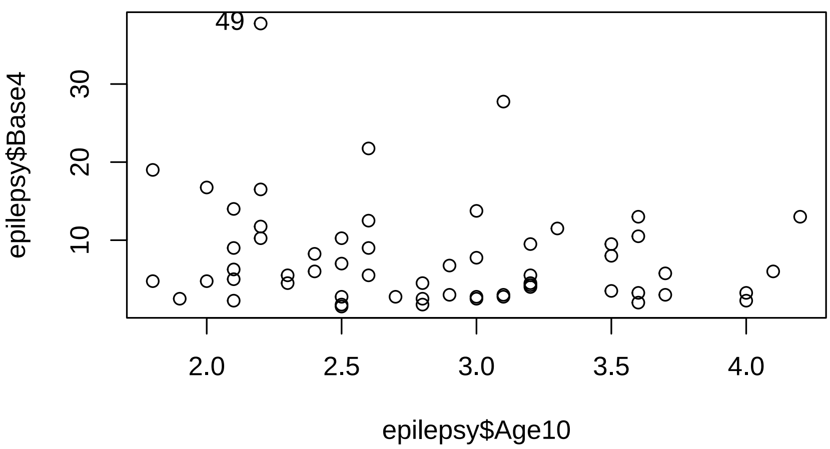
The interactions don't have to be included in the model. Without the interactions `Trt` gets highly significant.

e) *> plot(fit3$w.x)*
> identify(fit3$w.x)

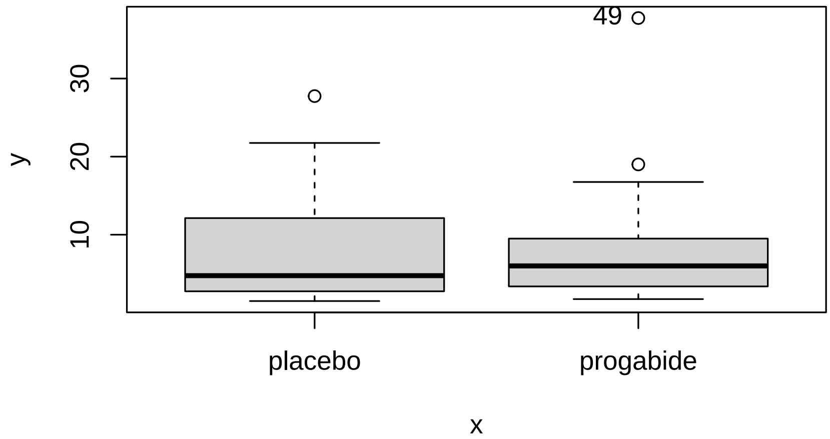


Observation 49 gets the smallest weight because of its explanatory variables.

> plot(epilepsy$Age10, epilepsy$Base4)
> identify(epilepsy$Age10, epilepsy$Base4)



> plot(epilepsy$Trt, epilepsy$Base4)
> identify(epilepsy$Trt, epilepsy$Base4)



It can be clearly seen the observation 49 has an extreme `Base4` value and thus gets the small weight.