AdvStDaAn, Worksheet, Week 3

Micheal Lappert

11.04.2022

Contents

Exercise 1

```
path <- file.path('Datasets', 'Synthetic.dat')
df <- read.table(path, header=TRUE)

summary(df)
```

Dataset loading and sanity check:

```
##        Y               x1              x2
##  Min.   : 1.51   Min.   :16.71   Min.   :-15.000
##  1st Qu.:16.04   1st Qu.:18.50   1st Qu.: -9.615
##  Median :21.71   Median :19.56   Median : -7.300
##  Mean   :21.54   Mean   :19.51   Mean   : -7.515
##  3rd Qu.:27.26   3rd Qu.:20.30   3rd Qu.: -5.260
##  Max.   :42.65   Max.   :22.06   Max.   :  0.610
```

```
dim(df)
```

```
## [1] 83  3
```

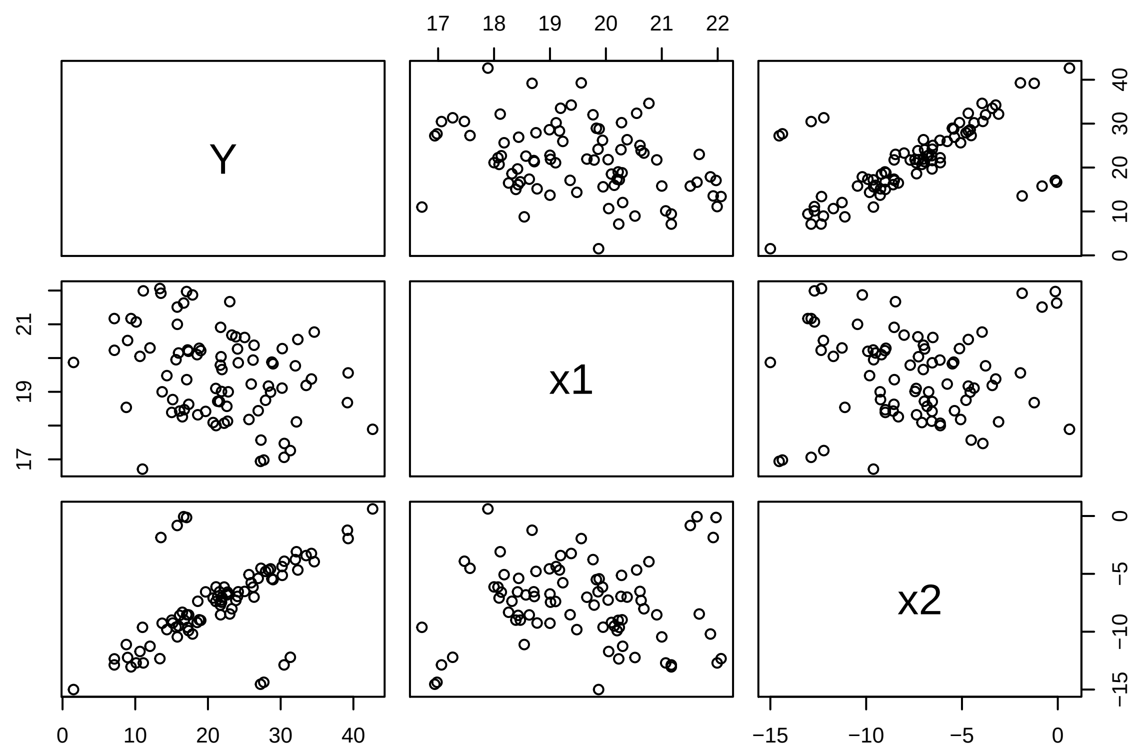
```r
head(df)
```

```
##       Y    x1     x2
## 1 33.50 19.19  -3.42
## 2 27.29 17.57  -4.52
## 3 22.60 18.57  -6.82
## 4 13.39 22.06 -12.33
## 5 20.71 18.09  -7.09
## 6 18.51 20.10  -9.20
```

```r
tail(df)
```

```
##        Y    x1     x2
## 78 21.59 18.71  -6.55
## 79 21.93 19.66  -7.02
## 80 21.80 20.04  -7.27
## 81 11.11 21.99 -12.70
## 82 21.11 18.00  -6.13
## 83 21.87 19.01  -7.45
```

```r
plot(df)
```



There seems to be some strong correlation between x2 and Y but withing x1 and x2 seems not to be a problem with multicollinearity. We fit an robust MM-Estimator model to the data.

Exercise 1.a)

```
library(robustbase)
rlm1.1 <- lmrob(Y ~ x1 + x2, data = df)
summary(rlm1.1)
```

```
##
## Call:
## lmrob(formula = Y ~ x1 + x2, data = df)
##  \--> method = "MM"
## Residuals:
##       Min        1Q    Median        3Q       Max
## -31.00351  -0.64032   0.09996   0.70716  31.30065
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.52018    2.07332   3.145  0.00233 **
## x1           1.90838    0.11012  17.330  < 2e-16 ***
## x2           2.95177    0.04072  72.495  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Robust residual standard error: 1.111
## Multiple R-squared:  0.986,  Adjusted R-squared:  0.9856
## Convergence in 9 IRWLS iterations
##
## Robustness weights:
##  8 observations c(7,17,27,37,47,57,67,77)
##   are outliers with |weight| = 0 ( < 0.0012);
##  9 weights are ~= 1. The remaining 66 ones are summarized as
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.5546  0.9036  0.9685  0.9302  0.9898  0.9989
## Algorithmic parameters:
##        tuning.chi                bb        tuning.psi        refine.tol
##         1.548e+00         5.000e-01         4.685e+00         1.000e-07
##           rel.tol         scale.tol         solve.tol       eps.outlier
##         1.000e-07         1.000e-10         1.000e-07         1.205e-03
##             eps.x warn.limit.reject warn.limit.meanrw
##         4.013e-11         5.000e-01         5.000e-01
##      nResample         max.it       best.r.s       k.fast.s          k.max
##            500             50              2              1            200
##    maxit.scale      trace.lev            mts     compute.rd fast.s.large.n
##            200              0           1000              0           2000
##                   psi           subsampling                   cov
##            "bisquare"         "nonsingular"         ".vcov.avar1"
## compute.outlier.stats
##                  "SM"
## seed : int(0)
```

8 observations were identified as outliers from the MM-Estimator. The R^2 has a pretty good score of 0.986.

Coefficients:

```r
coef(rlm1.1)
```

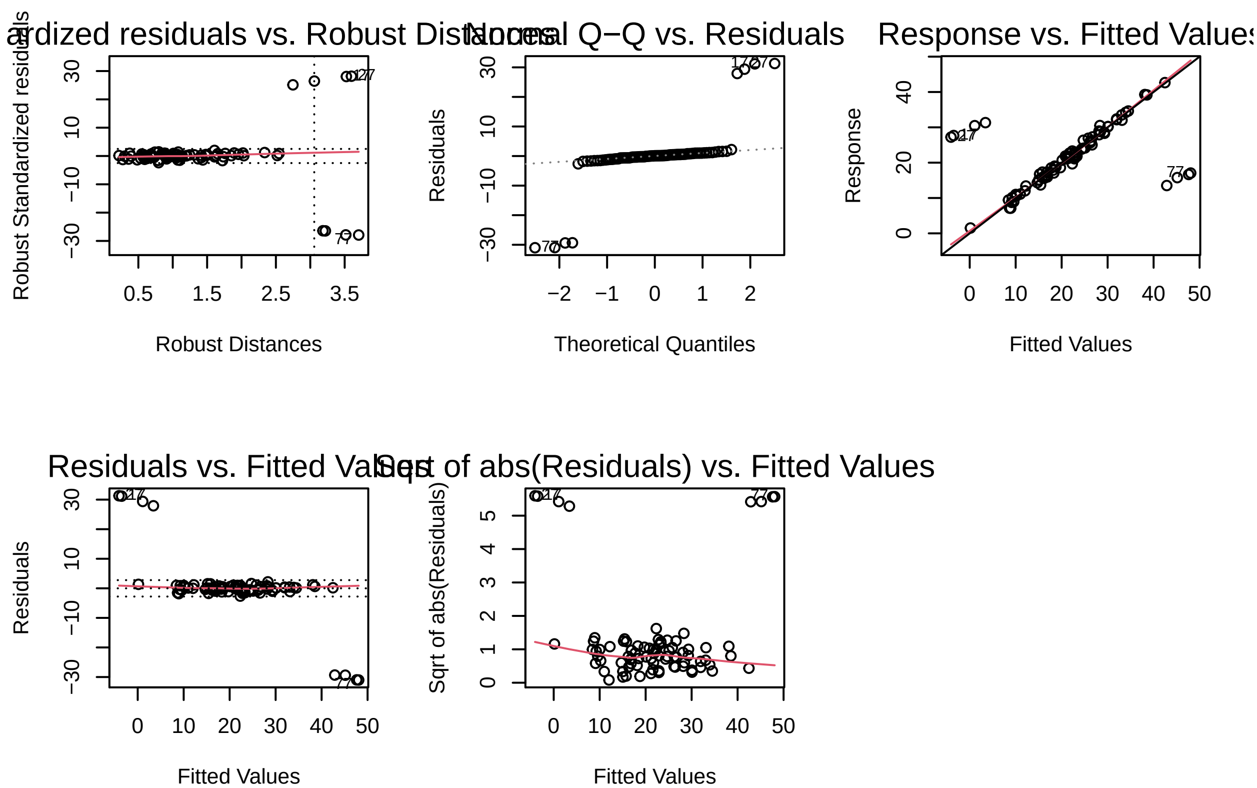
```
## (Intercept)          x1          x2
##    6.520180    1.908378    2.951771
```

The estimated standard dviation of the error is 1.111.

Residual ans sensitivity analysis:

```r
par(mfrow=c(2,3))
plot(rlm1.1)
```

```
## recomputing robust Mahalanobis distances

## saving the robust distances 'MD' as part of 'rlm1.1'
```



The graphic top left replaces the classical graphic "Residuals against leverage". Robust distances measures the outlyingness of observations in the x-space. It replaces the classical measure of leverage, H_ii, and is not distorted by outliers. The two dotted horizontal lines is the band 0 +/- 2.5 sigma^. Most residuals should be within this band. All residuals right of the dotted vertical line are leverage points; i.e. they are too far from the bulk of the data.

In all of the five graphics, 8 distinct outliers are visible. Hence the residuals are not Gaussian distributed. The is a slight decreasing trend visible in the last graphic. Hence, it might be that the variance is not constant. But the hint is weak. There is no evidence that the expectation is not constant. Conclusion: There are 8 distinct outliers. Inferential results must be based on robust estimation. Least squares estimation will not deliver reliable results.

Exercise 1.b)

Fit the above model again but with the lest squares method.

```
lm1.1 <- lm(Y ~ x1 + x2, data = df)
summary(lm1.1)
```

```
##
## Call:
## lm(formula = Y ~ x1 + x2, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.3668  -3.8685   0.1167   4.3564  11.8021
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  72.9020     9.6482   7.556 5.96e-11 ***
## x1           -2.0837     0.4882  -4.268 5.37e-05 ***
## x2            1.4258     0.1828   7.802 1.98e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.799 on 80 degrees of freedom
## Multiple R-squared:  0.4963, Adjusted R-squared:  0.4837
## F-statistic: 39.41 on 2 and 80 DF,  p-value: 1.226e-12
```

The R^2 crashes to the half of the value than with the robust MM-estimator and the residual standard error increases to 5.799

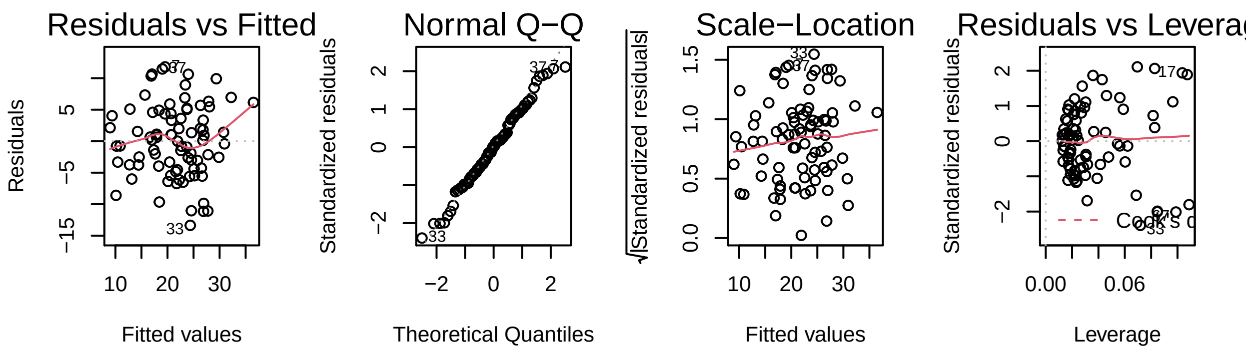
Coefficients:

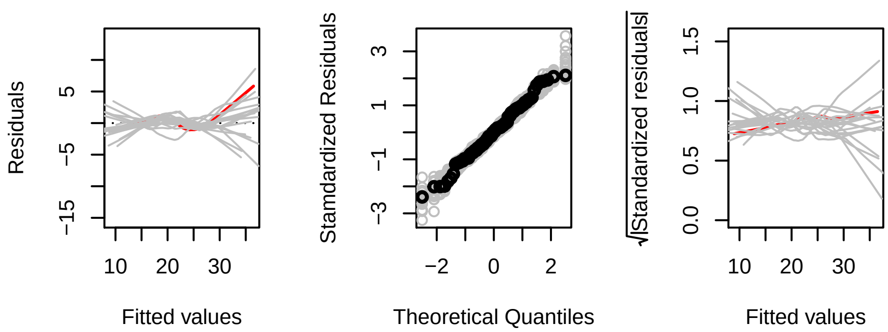
```
coef(lm1.1)
```

```
## (Intercept)          x1          x2
##   72.902036   -2.083705    1.425830
```

The intercept is way higher which shows a much flater line. Also the estmators for β_1 and β_2 are very different than from the robust estimator. Lets perform a residual and sensitivity analysis:

```
par(mfrow=c(2,4))
plot(lm1.1)
plot.lmSim(lm1.1, SEED = 1)
```





Surprisingly there is no evidence that any of the model assumptions (constant expactation of residual, gaussien distributed residuals and constant residual variance) is violated. Also there are no too influential residuals with Cook's distance > 1.

Exercise 1.c)

The residual and sensitivity analysis shows no model violations of both model (robust estmator as well as the least squares fit). The only 2 ways to know, that the fit of least squares is not adequat is by identifying the outliers in the robust method and the rather low R^2 in the summary of the least squares fit. This is crucial to find out when modeling and one should therfore always use at least for adequacy checking of the linear model as well fit a robust estimator in the end.

Exercise 2

```r
path <- file.path('Datasets', 'ExpressDS.dat')
df <- read.table(path, header=TRUE)

summary(df)
```

Dataset loading and sanity check:

```
##      weight          distance           cost
```

```
##  Min.   :0.300   Min.   : 45.00   Min.   : 1.000
##  1st Qu.:2.075   1st Qu.: 93.75   1st Qu.: 1.975
##  Median :4.250   Median :160.00   Median : 4.700
##  Mean   :4.058   Mean   :156.05   Mean   : 6.335
##  3rd Qu.:6.275   3rd Qu.:216.75   3rd Qu.: 9.650
##  Max.   :8.100   Max.   :280.00   Max.   :15.500
```

```
str(df)
```

```
## 'data.frame':    20 obs. of  3 variables:
##  $ weight  : num  5.9 3.2 4.4 6.6 0.75 0.7 6.5 4.5 0.6 7.5 ...
##  $ distance: int  47 145 202 160 280 80 240 53 100 190 ...
##  $ cost    : num  2.6 3.9 8 9.2 4.4 1.5 14.5 1.9 1 14 ...
```

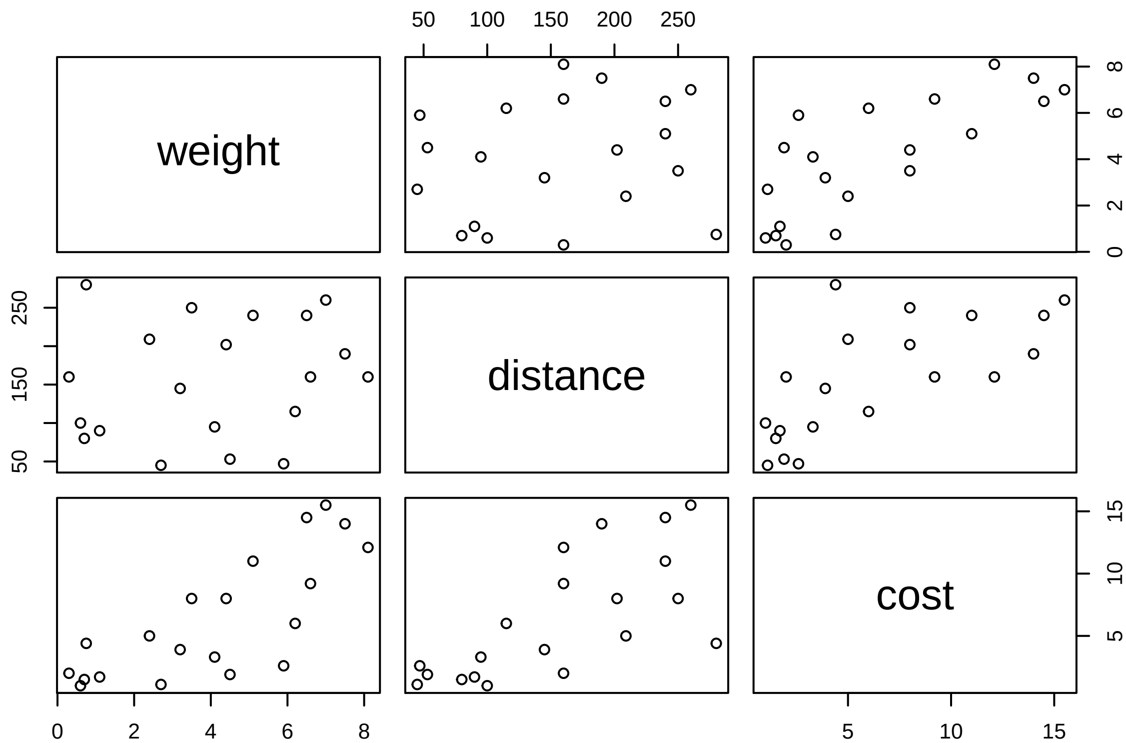
```
head(df)
```

```
##   weight distance cost
## 1   5.90       47  2.6
## 2   3.20      145  3.9
## 3   4.40      202  8.0
## 4   6.60      160  9.2
## 5   0.75      280  4.4
## 6   0.70       80  1.5
```

```
tail(df)
```

```
##    weight distance cost
## 15    2.7       45  1.1
## 16    3.5      250  8.0
## 17    4.1       95  3.3
## 18    8.1      160 12.1
## 19    7.0      260 15.5
## 20    1.1       90  1.7
```

```
plot(df)
```



Data looks alright.

Exercise 2.a)

Apply Tukey's firs-aid transformations to the data and checking if the transformations are suitable with an additive model.

```
df$lWeight <- log(df$weight)
df$lDistance <- log(df$distance)
df$lCost <- log(df$cost)

library(gam)
```

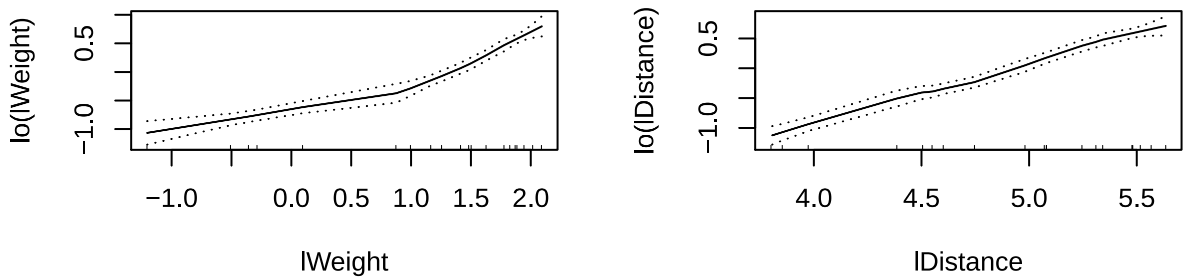
```
## Loading required package: splines

## Loading required package: foreach

## Loaded gam 1.20
```

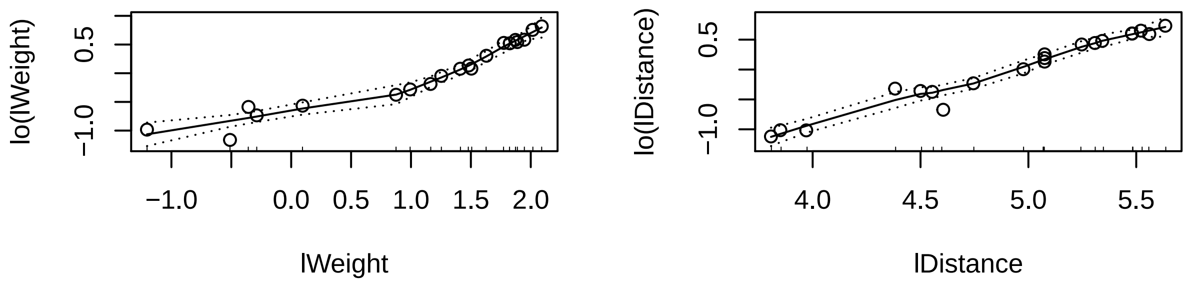
```
gam1.1 <- gam(lCost ~lo(lWeight) + lo(lDistance), data = df)
summary(gam1.1)
```

```
##
## Call: gam(formula = lCost ~ lo(lWeight) + lo(lDistance), data = df)
```

```
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33168 -0.01125  0.01272  0.03079  0.19185
##
## (Dispersion Parameter for gaussian family taken to be 0.0159)
##
##     Null Deviance: 15.452 on 19 degrees of freedom
## Residual Deviance: 0.1759 on 11.0929 degrees of freedom
## AIC: -18.0979
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##                   Df Sum Sq Mean Sq F value    Pr(>F)
## lo(lWeight)    1.000 7.4209  7.4209  467.94 2.012e-10 ***
## lo(lDistance)  1.000 6.6557  6.6557  419.70 3.630e-10 ***
## Residuals     11.093 0.1759  0.0159
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##               Npar Df Npar F    Pr(F)
## (Intercept)
## lo(lWeight)       3.2 8.6409 0.002783 **
## lo(lDistance)     2.7 1.0921 0.387041
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow=c(2,2))
plot(gam1.1, se = TRUE)
plot(gam1.1, se = TRUE, residuals = TRUE)
```





Rule of Thumb: If a straight line fits between the confidence band the variable fits and does not need any further transformation.

According to the rule of thumb lDist does not need any further transformation, but lWeight does.

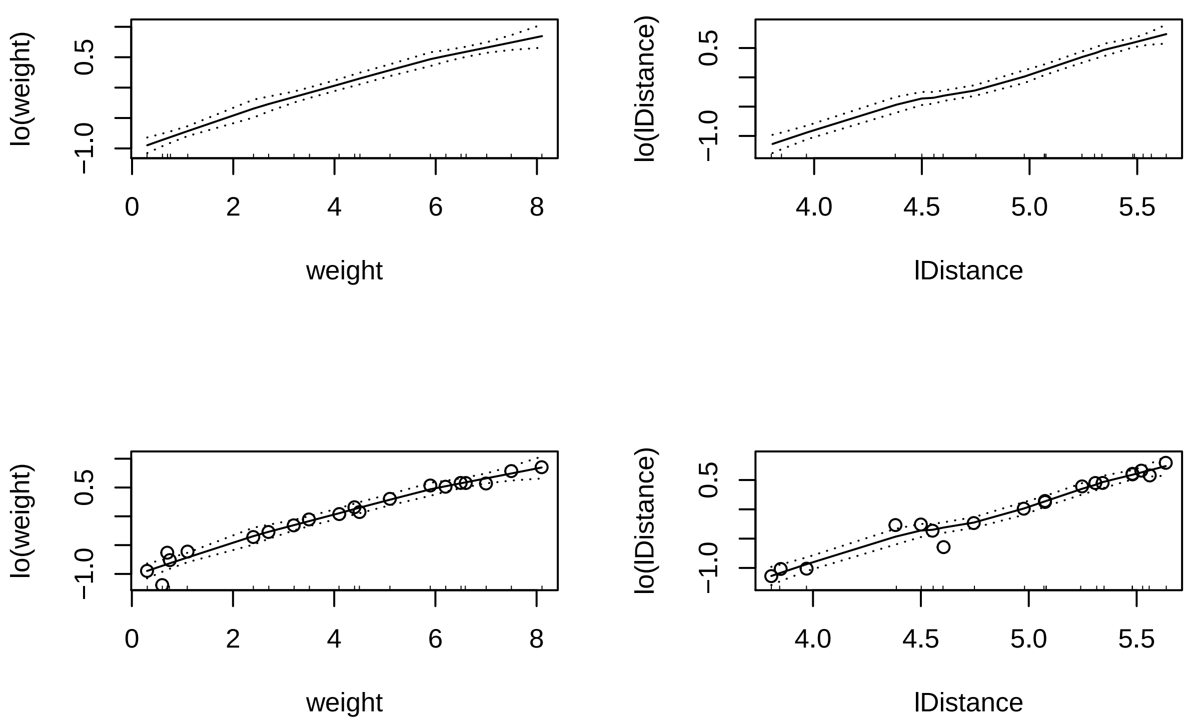
Lets try how the untransformed explanatory variable weight looks in the model:

```
gam1.2 <- gam(lCost ~ lo(weight) + lo(lDistance), data = df)
summary(gam1.2)
```

```
##
## Call: gam(formula = lCost ~ lo(weight) + lo(lDistance), data = df)
## Deviance Residuals:
##       Min        1Q    Median        3Q       Max
## -0.334017 -0.012240  0.003017  0.034002  0.198488
##
## (Dispersion Parameter for gaussian family taken to be 0.016)
##
##     Null Deviance: 15.452 on 19 degrees of freedom
## Residual Deviance: 0.187 on 11.6702 degrees of freedom
## AIC: -18.0286
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##                  Df Sum Sq Mean Sq F value    Pr(>F)
## lo(weight)     1.00 9.2388  9.2388  576.52 2.685e-11 ***
```

```
## lo(lDistance)  1.00 6.3857  6.3857  398.48 2.209e-10 ***
## Residuals     11.67 0.1870  0.0160
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##                 Npar Df Npar F  Pr(F)
## (Intercept)
## lo(weight)          2.6 1.1812 0.3537
## lo(lDistance)       2.7 1.2727 0.3262
```

```r
par(mfrow=c(2,2))
plot(gam1.2, se = TRUE)
plot(gam1.2, se = TRUE, residuals = TRUE)
```



Like that a straight line fits also between the confidence bands of weight and therefore weight gets untransformed into the model.

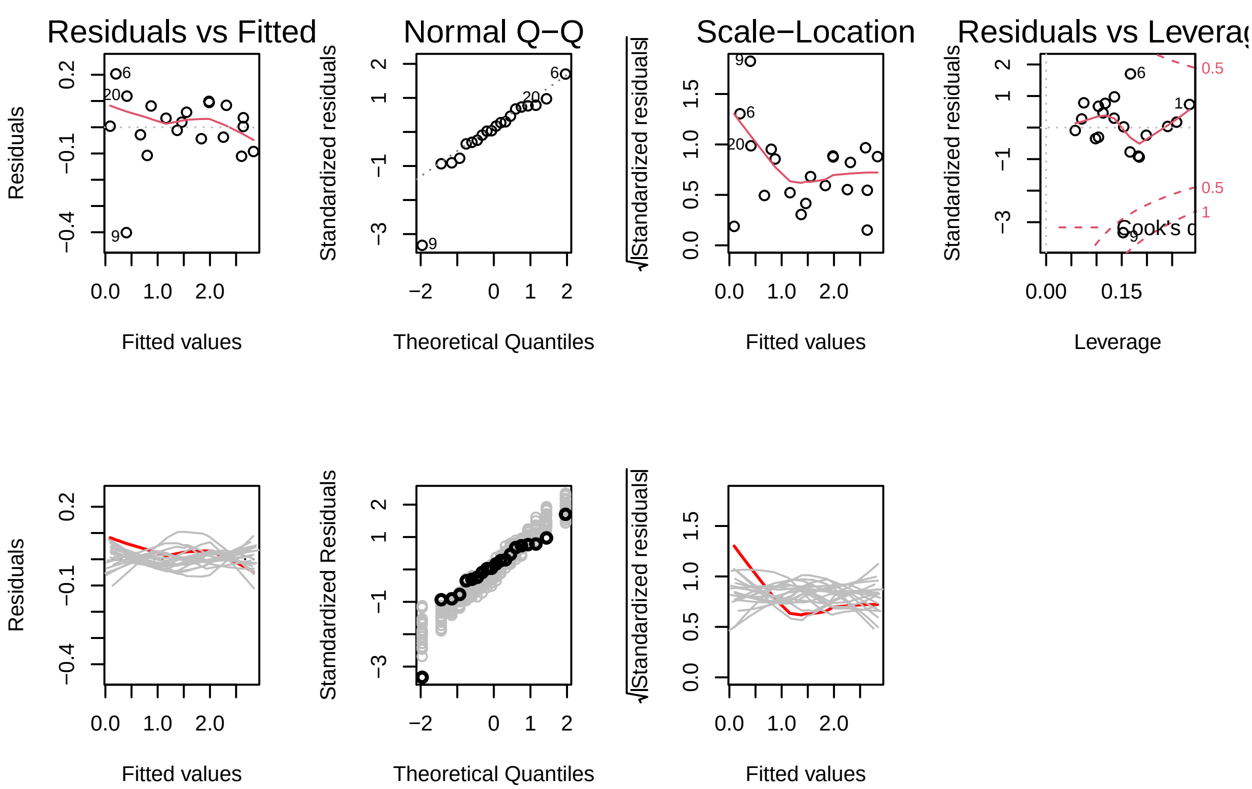
Exercise 2.b)

```r
lm2.1 <- lm(lCost ~ weight + lDistance, data = df)
summary(lm2.1)
```

```
##
```

```
## Call:
## lm(formula = lCost ~ weight + lDistance, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.40148 -0.03926  0.01173  0.08173  0.20323
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.32627    0.25512  -16.96 4.36e-12 ***
## weight       0.23119    0.01208   19.14 6.14e-13 ***
## lDistance    0.99650    0.05245   19.00 6.91e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.131 on 17 degrees of freedom
## Multiple R-squared:  0.9811, Adjusted R-squared:  0.9789
## F-statistic:   442 on 2 and 17 DF,  p-value: 2.206e-15
```

```
par(mfrow=c(2,4))
plot(lm2.1)
plot.lmSim(lm2.1, SEED = 1)
```



Interpretation:

1. Tukey-Anscombe plot: The smoother shows a decreasing trend which is, however in the stochastic fluctuation. Observation i=9 seems to be an oulier.
   => The assumption of constant expactation is not violated.
2. Q-Q plot: The data scatters nicely around the straight line (except i=9) and seems to be within the stochastic fluctuation.
   => The assumption of Gaussian distributed errors seems not violated.
3. Scale-location plot: The smoother has a strong decrease in the first half and levels out afterwards and stays almost within the stochastic fluctuation.
   => There is no (real) evidence against the assumption of constant variance of the residuals.
4. Residuals vs. Leverage: All observations have Cook's Distance <1 and therewith no too influential points.
   => No too influential (dangerous) observations

CONCLUSION: The model does fit but there might be an outlier (obs i=9). Lets try to remedy this with an robust estimator.

Exercise 2.c)

```
library(robustbase)
rlm2.1 <- lmrob(lCost ~ weight + lDistance, data = df)
summary(rlm2.1)
```

```
##
## Call:
## lmrob(formula = lCost ~ weight + lDistance, data = df)
##  \--> method = "MM"
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.483629 -0.053521 -0.009431  0.054175  0.120967
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.20096    0.14760  -28.46 8.79e-16 ***
## weight       0.21585    0.01145   18.85 7.86e-13 ***
## lDistance    0.98912    0.03183   31.07  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Robust residual standard error: 0.0817
## Multiple R-squared:  0.9919, Adjusted R-squared:  0.9909
## Convergence in 11 IRWLS iterations
##
## Robustness weights:
##  observation 9 is an outlier with |weight| = 0 ( < 0.005);
##  2 weights are ~= 1. The remaining 17 ones are summarized as
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.5655  0.9253  0.9582  0.9229  0.9647  0.9919
## Algorithmic parameters:
##        tuning.chi                bb        tuning.psi        refine.tol
##         1.548e+00         5.000e-01         4.685e+00         1.000e-07
##           rel.tol         scale.tol         solve.tol       eps.outlier
##         1.000e-07         1.000e-10         1.000e-07         5.000e-03
```

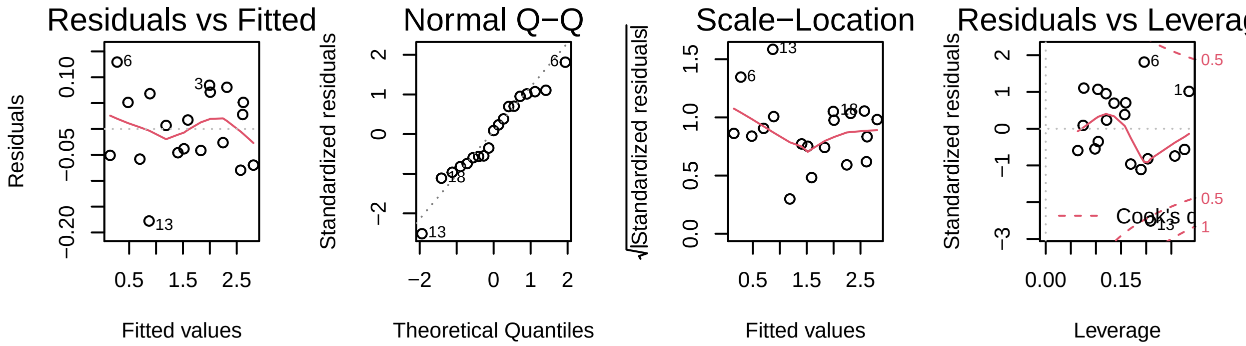
```
##             eps.x warn.limit.reject warn.limit.meanrw
##         1.473e-11         5.000e-01         5.000e-01
##      nResample         max.it       best.r.s       k.fast.s          k.max
##            500             50              2              1            200
##    maxit.scale      trace.lev            mts     compute.rd fast.s.large.n
##            200              0           1000              0           2000
##                   psi           subsampling                   cov
##            "bisquare"         "nonsingular"         ".vcov.avar1"
## compute.outlier.stats
##                  "SM"
## seed : int(0)
```

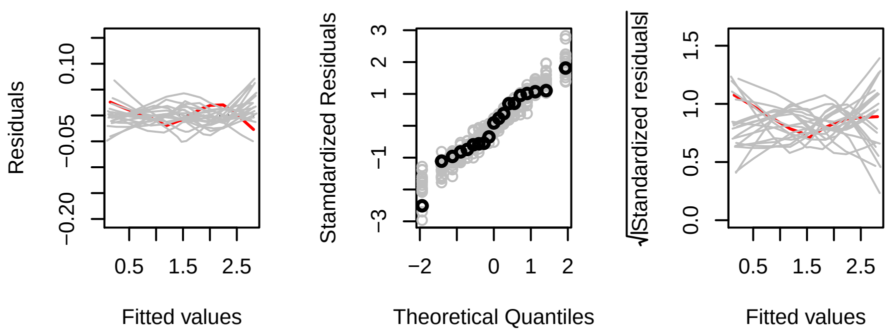
Indeed observation i=9 is a strong outlier. So we fit the linear model again without it and check the model assumptions again.

```
lm2.2 <- lm(lCost ~ weight + lDistance, data = df[-9,])
summary(lm2.2)
```

```
##
## Call:
## lm(formula = lCost ~ weight + lDistance, data = df[-9, ])
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.177810 -0.048428  0.006828  0.059783  0.129351
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.188943   0.157014  -26.68 1.08e-14 ***
## weight       0.218195   0.007713   28.29 4.32e-15 ***
## lDistance    0.984093   0.031945   30.80 1.13e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07956 on 16 degrees of freedom
## Multiple R-squared:  0.9923, Adjusted R-squared:  0.9913
## F-statistic:  1024 on 2 and 16 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,4))
plot(lm2.2)
plot.lmSim(lm2.2, SEED = 1)
```





Like that none of the model assumptions is violated and no outlier is visible. So this model fits the data more adequately.