Solution to Series 5

1. **Model diagnostics: simulation study** Assessing model diagnostic plots requires experience. Often it is difficult to decide whether a deviation from the theoretical centre a systematic one (i.e. needing correction) or a random one (i.e. just variability in the data). Experience can be gained by performing model diagnostics on problems where it is known whether the model assumptions hold or do not hold. This allows to identify the naturally occurring variability in the results.

   Simulate 4 different models of which only one fulfils all model assumptions. The second model should include minor deviations from the constant variance assumption and the third model should include major deviations. The last model includes a systematic deviation (non-linearity), e.g., $\mathbb{E}[\epsilon_i] \neq 0$.

   R hints:

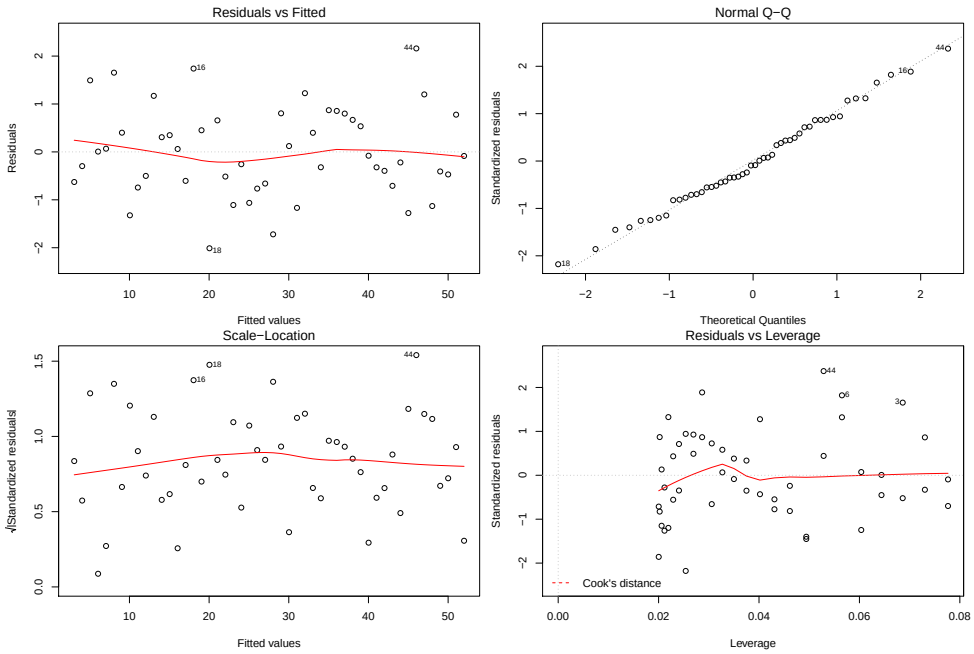
```
> n <- 50
> xx <- 1:n
> yy.a <- 2+1*xx+rnorm(n)
> yy.b <- 2+1*xx+rnorm(n)*(50*xx/n)
> yy.c <- 2+1*xx+rnorm(n)*sqrt(50*xx/n)
> yy.d <- cos(xx*pi/(n/2)) + rnorm(n)
```

   a) Decide which relation has no violation of the model assumption, minor deviation to non-constant variance, major deviation to non-constant variance and which relation is non-linear.
   From the three R formulae we can derive the following:
   .a Model assumptions valid.
   .b Model contains strong non-constant variance.
   .c Variance slightly non-constant.
   .d Non-linear model.

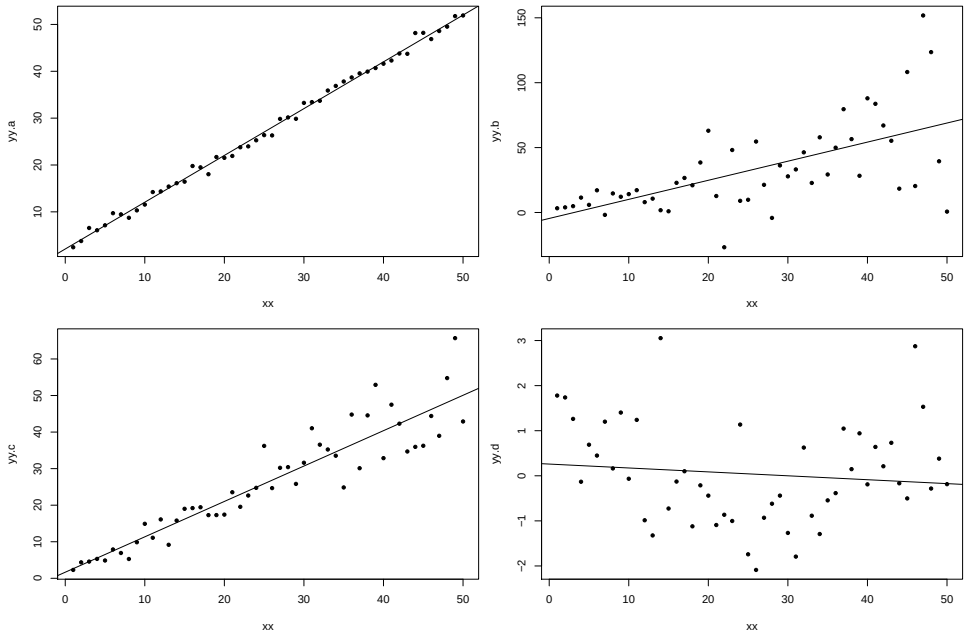
   b) Plot each response `yy.[a,b,c,d]` versus `xx`. Fit a simple linear regression and plot the regression line into the corresponding scatter plot.

   c) Perform model diagnostics and have a look at the diagnostic plots. Where can we see the deviations? How large is the random variation within these plots?

```
> par(mfrow=c(2,2))
> plot(fit.a)
```

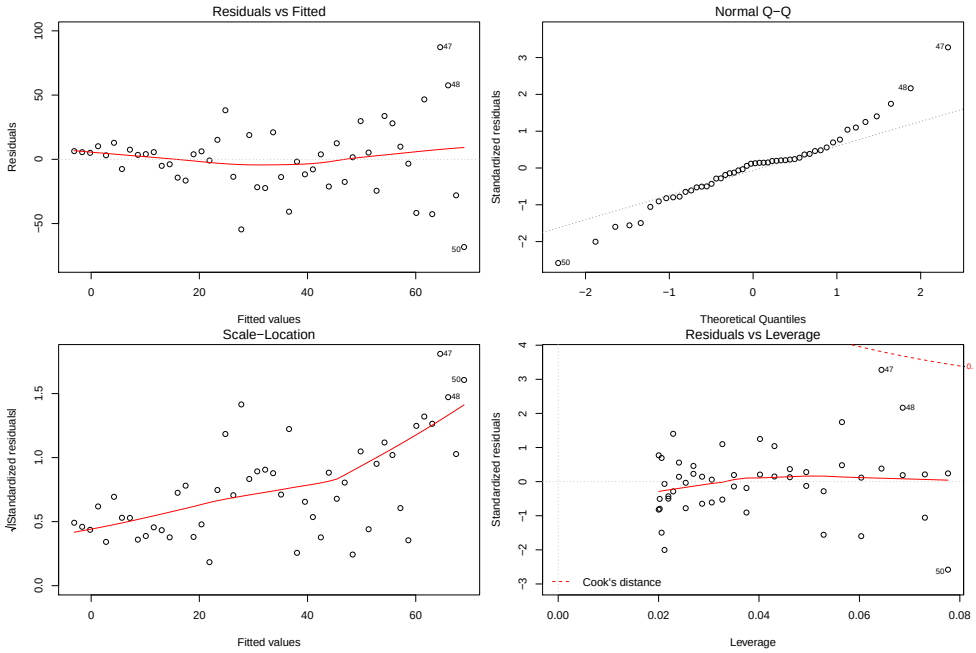


```
> set.seed(123)
> n <- 50
> xx <- 1:n
> yy.a <- 2+1*xx+rnorm(n)
> yy.b <- 2+1*xx+rnorm(n)*(50*xx/n)
> yy.c <- 2+1*xx+rnorm(n)*sqrt(50*xx/n)
> yy.d <- cos(xx*pi/(n/2)) + rnorm(n)
> par(mfrow=c(2,2))
> fit.a <- lm(yy.a ~ xx)
> plot(xx, yy.a, pch=20)
> abline(fit.a)
> fit.b <- lm(yy.b ~ xx)
> plot(xx, yy.b, pch=20)
> abline(fit.b)
> fit.c <- lm(yy.c ~ xx)
> plot(xx, yy.c, pch=20)
> abline(fit.c)
> fit.d <- lm(yy.d ~ xx)
> plot(xx, yy.d, pch=20)
> abline(fit.d)
```



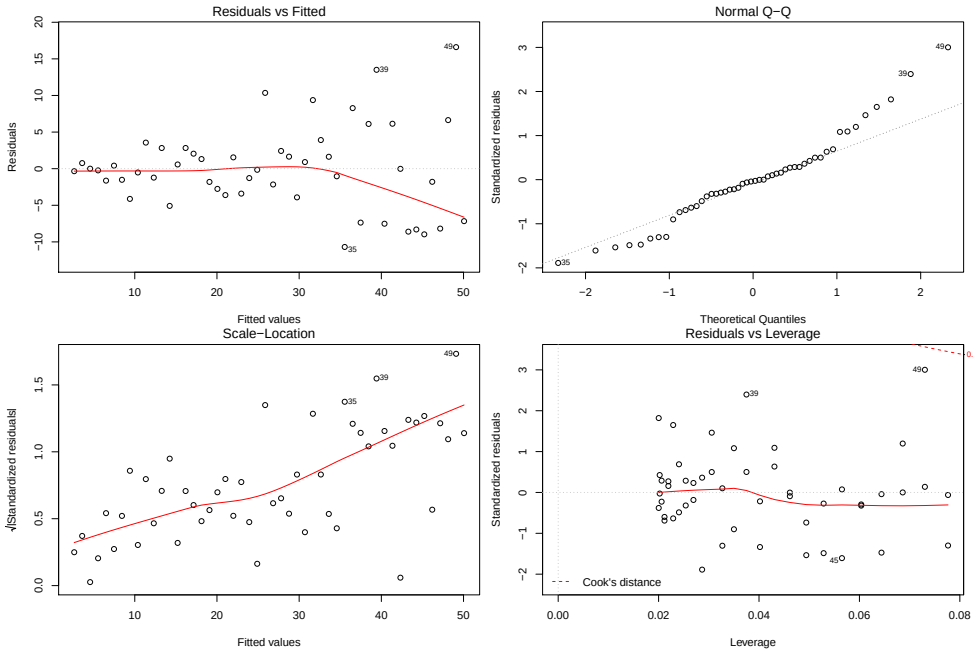
The Tukey-Anscombe as well as the scale-location plot show residuals with strong non-constant variance. The size of the residuals increase with larger fitted values.

```
> par(mfrow=c(2,2))
> plot(fit.b)
```



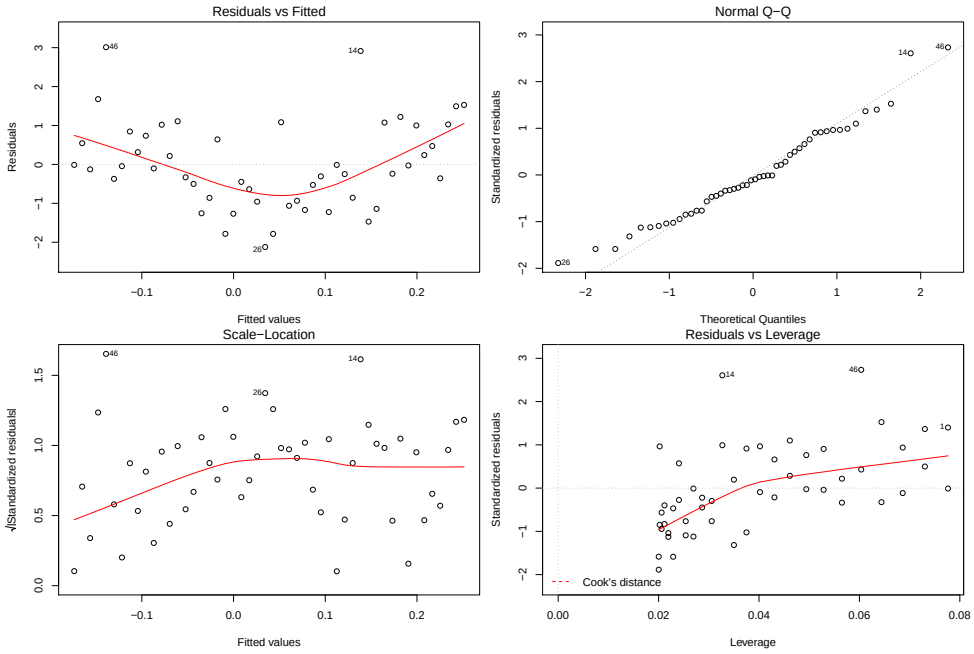
Again the residuals have non-constant variance. However, it is less accentuated as in the previous example. The residuals have smaller values than before.

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



Model `fit.d` is clearly non-linear. This can be seen in the residual plots. The Tukey-Anscombe plot exhibits a U-shaped pattern. From this we can conclude the existence of a non-linear relation between response and predictors.

```
> par(mfrow=c(2,2))
> plot(fit.d)
```



d) Repeat generating the random numbers a few times and study the variation in the resulting plots. You can also change the number of observations and track the changes in the plots.
These plots should be generated repeatedly. Manipulating the number of observations n is also helpful and instructive. However, the above described structures are of general nature and will largely remain.

Assessing normal plots is equally difficult[1]. Even drawing samples from a normal distribution does not result in observations lying directly on the straight line. Check out how a skewed, a long-tailed and a short-tailed distribution look like.

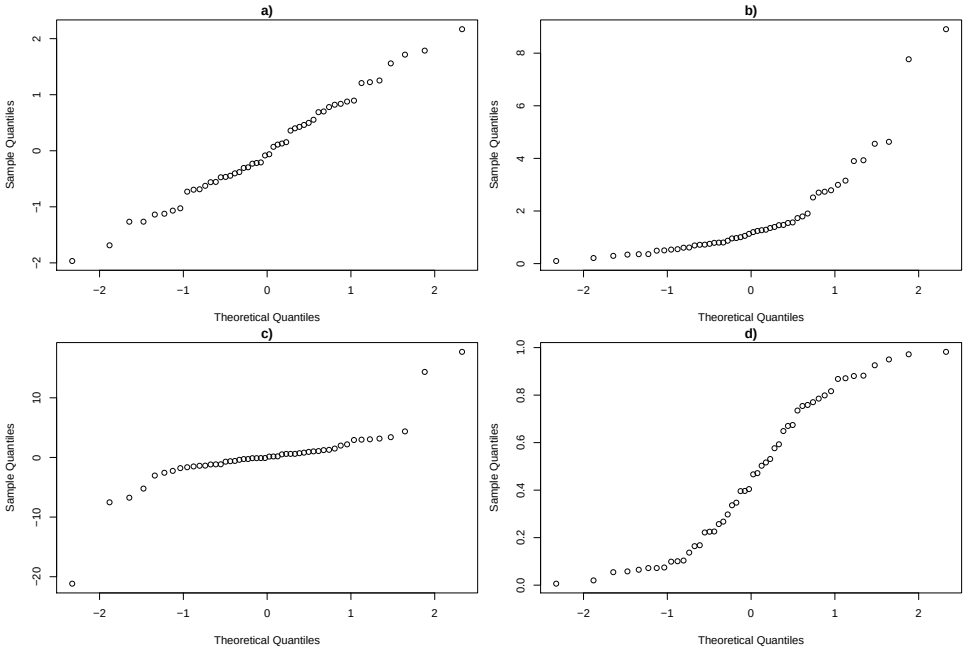
R hints:

```
> qqnorm(rnorm(n), main=c("a)"))
> qqnorm(exp(rnorm(n)), main=c("b)"))
> qqnorm(rcauchy(n), main=c("c)"))
> qqnorm(runif(n), main=c("d)"))
```

e) Decide which random numbers are normal, skewed, short-tailed and long-tailed.
The normal Q-Q plot for example **a)** illustrates that the observations were drawn from a normal distribution. The sample quantiles fit nicely to the theoretical quantiles of a normal distribution. Deviations from the diagonal line are to be expected due to randomness.
In the plot of **b)** the curve is bent upwards. This indicates a positively skewed distribution of the sample points, i.e., the distribution is not symmetric.
In **c)** the distribution is fairly symmetric. However, the generated realisations of our random number have longer "tails" than those of a normal distribution. The resulting curve has the shape of an inverted S. The smaller quantiles are always below the diagonal line, larger quantiles are always above.
In **d)** we have the opposite case of **c)**. Here, the smaller quantiles tend to be above the diagonal line and the larger quantiles are below it. The curve is S-shaped and we conclude that this sample has a short-tailed distribution.

```
> par(mfrow=c(2,2))
> set.seed(123)
> qqnorm(rnorm(n), main=c("a)"))
> qqnorm(exp(rnorm(n)), main=c("b)"))
> qqnorm(rcauchy(n), main=c("c)"))
> qqnorm(runif(n), main=c("d)"))
```

[1] In the StatsNotes of the Department of Mathematics and Statistics at Murdoch University it reads: *A sufficiently trained statistician can read the vagaries of a Q-Q plot like a shaman can read a chicken's entrails, with a similar recourse to scientific principles. Interpreting Q-Q plots is more a visceral than an intellectual exercise. The uninitiated are often mystified by the process. Experience is the key here.*



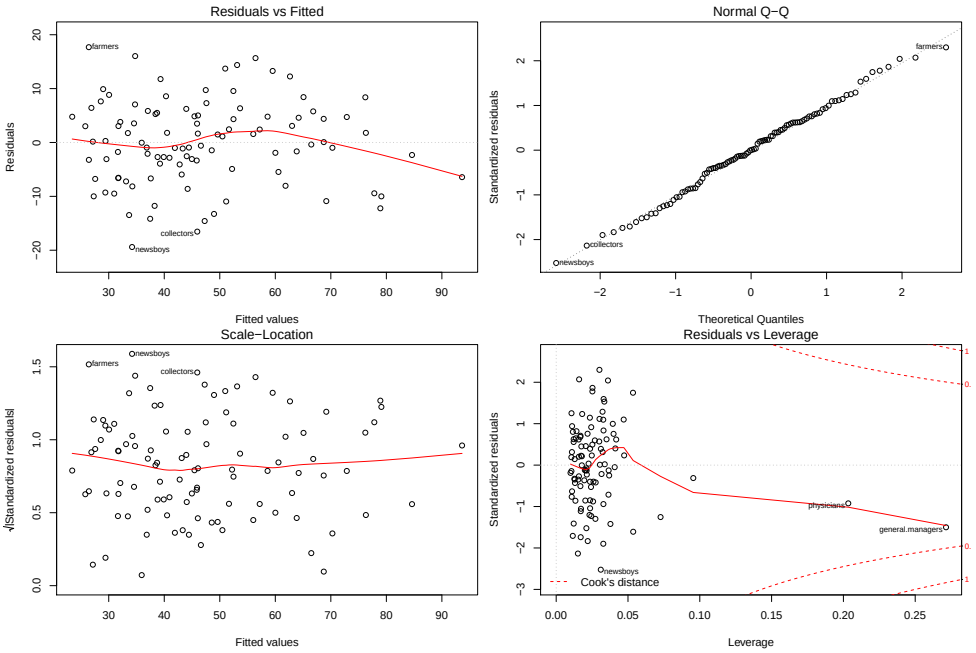
f) Repeat generating the random numbers a few times and study the variation in the resulting Q-Q plots. You can also change the number of observations and track the changes in the plots. Repeat drawing from the random numbers and vary the number of observations as well.

2. a) Partial residual plots Use the "prestige" data set from the package `library(car)`. Fit the following model

$$\texttt{prestige} \sim \texttt{income + education}.$$

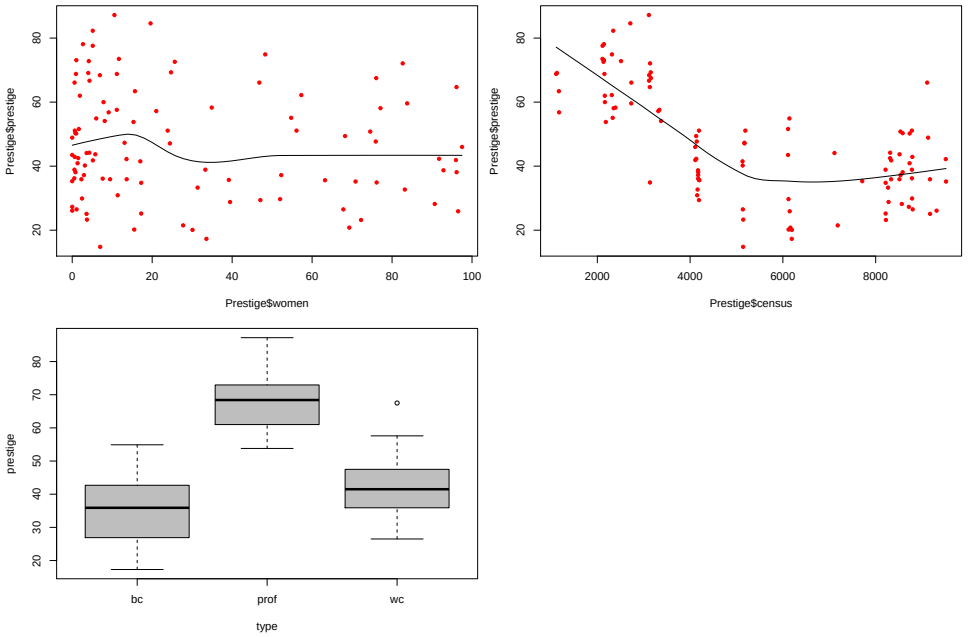
Generate the partial residual plots and perform a general residual analysis. Improve the model by transformation. Plot the resulting residuals versus the variables in the data set not used in the model so far. Considering these plots which variables do you expect to have a strong influence on the response? Add these variables in a stepwise manner as predictors to the model. Keep an eye on the summary output and the diagnostic plots to fit an optimal model.

```
> library(car)
> data(Prestige)
> fit00 <- lm(prestige ~ income + education, data=Prestige)
> summary(fit00)

Call:
lm(formula = prestige ~ income + education, data = Prestige)

Residuals:
     Min       1Q   Median       3Q      Max
-19.4040  -5.3308   0.0154   4.9803  17.6889

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -6.8477787  3.2189771  -2.127   0.0359 *
income       0.0013612  0.0002242   6.071 2.36e-08 ***
education    4.1374444  0.3489120  11.858  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.81 on 99 degrees of freedom
Multiple R-squared: 0.798,        Adjusted R-squared: 0.7939
F-statistic: 195.6 on 2 and 99 DF,  p-value: < 2.2e-16

> par(mfrow=c(2,2))
> plot(fit00)
```



Already, this model fits well. Global F-test and the two predictors are highly significant. Diagnostic plots look reasonable. We can see some deviation of the smoother from the x-axis in the Tukey-Anscombe plot. Physicians and General Managers seem to be leverage points. However, since both do not have large residuals nor Cook's distances we need not to be afraid.

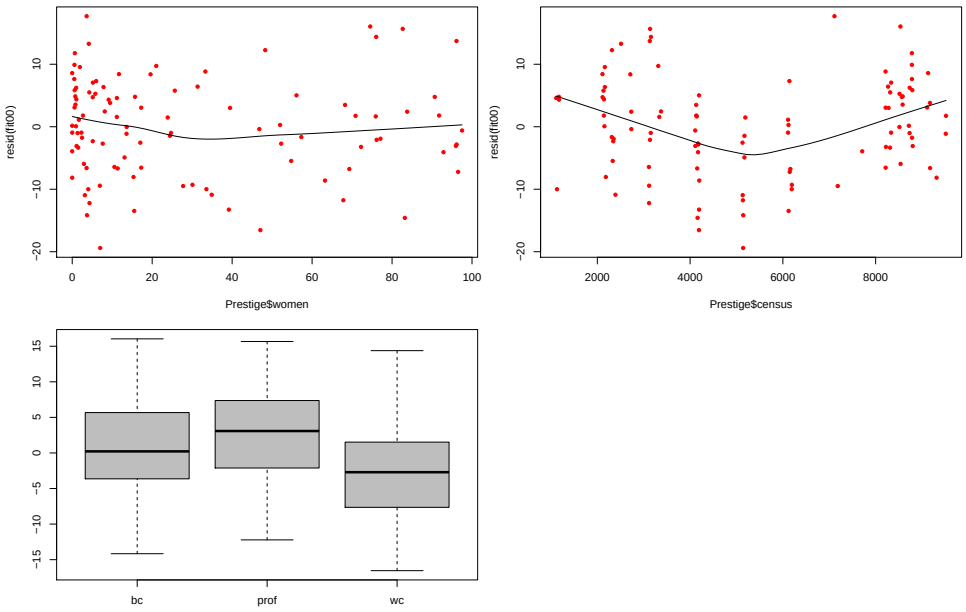
```
> par(mfrow = c(2,2))
> scatter.smooth(Prestige$women, Prestige$prestige, col="red", pch=20)
> scatter.smooth(Prestige$census, Prestige$prestige, col="red", pch=20)
> boxplot(prestige ~ type, data=Prestige, col="grey", ylab="prestige", xlab="type")
```



While the relation between women and prestige seems to be negligible, the relation between the variables census and prestige as well as type and prestige are quite strong.

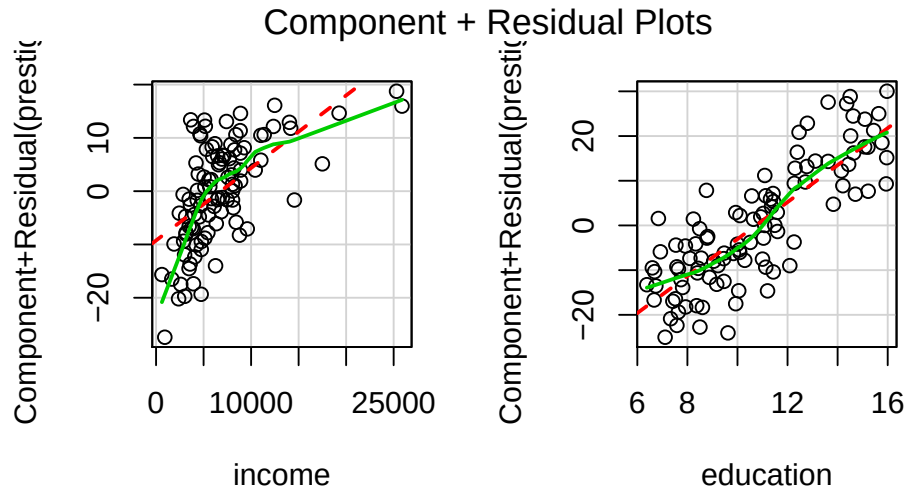
BUT: The differences we see in prestige due to different types can be explained by the higher salaries and longer education of the variables class prof. Rather, we would like to see the marginal influence of occupation on the response controlling for the two variables income and education. We can see this by plotting the residuals versus these predictors.

```
> par(mfrow=c(2,2))
> scatter.smooth(Prestige$women, resid(fit00), col="red", pch=20)
> scatter.smooth(Prestige$census, resid(fit00), col="red", pch=20)
> boxplot(resid(fit00) ~ Prestige$type, col="grey")
```



We do see now that the influence of type can be largely explained by income and education. The remaining marginal influence of type is not too large anymore (but it is there). Next, we check the partial residual plots of the two remaining predictors in the model:

```
> crPlots(fit00)
```

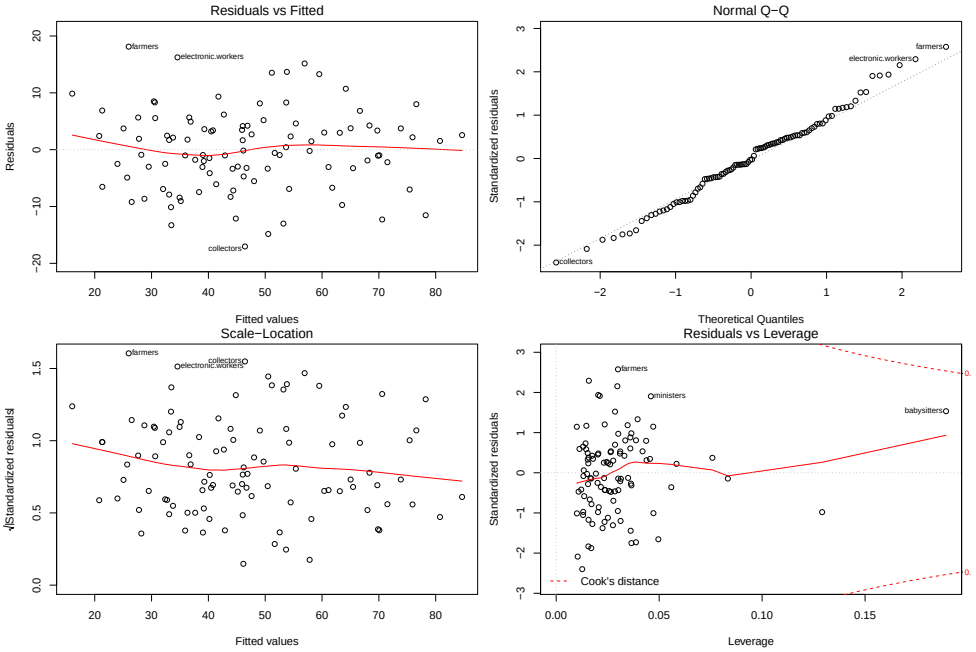


It seems as if there is no linear relation between the partial residuals and income. This would mean that we did not include income correctly into the model. Since the shape looks like a logarithm curve it would be reasonable to include log(income) instead of income. This is also sustainable with respect to the First-Aid-Transformation.
The contribution of education looks "more" linear. We can see some deviation, e.g., for occupations with shorter education time there seems to be a different relation than for those with longer duration. We will ignore this for the moment.

```
> fit01 <- lm(prestige ~ log(income) + education, data=Prestige)
> summary(fit01)

Call:
lm(formula = prestige ~ log(income) + education, data = Prestige)

Residuals:
     Min       1Q   Median       3Q      Max
-17.0346  -4.5657  -0.1857   4.0577  18.1270
```

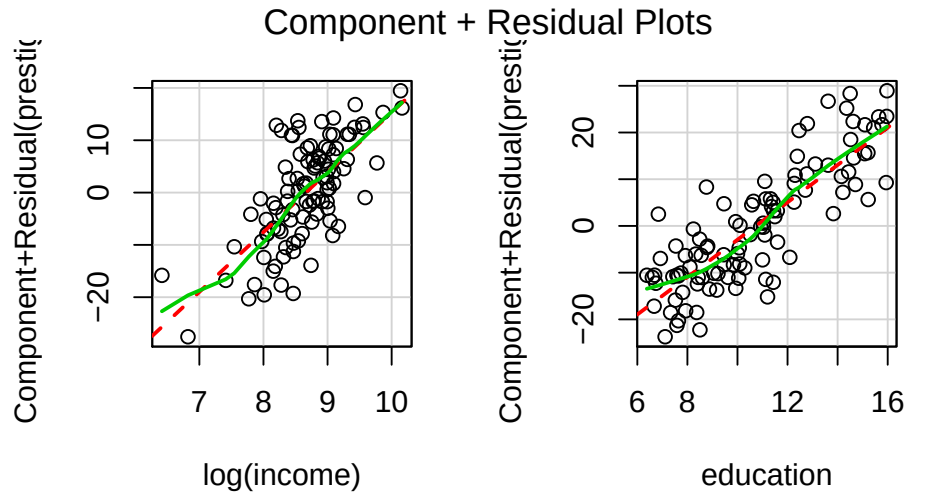
```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -95.1940    10.9979  -8.656 9.27e-14 ***
log(income)  11.4375     1.4371   7.959 2.94e-12 ***
education     4.0020     0.3115  12.846  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.145 on 99 degrees of freedom
Multiple R-squared: 0.831,        Adjusted R-squared: 0.8275
F-statistic: 243.3 on 2 and 99 DF,  p-value: < 2.2e-16
```

```
> par(mfrow=c(2,2))
> plot(fit01)
```



All of the parameters in the summary as well as the residual plots are better now. The transformation of income has clearly improved the fit. We check now the partial residual plots again:

```
> crPlots(fit01)
```



These also look fine suggesting that the transformation of income was successful. We now want to include the variable type into the model:

```
> fit02 <- lm(prestige ~ log(income) + education + type, data=Prestige)
> summary(fit02)
```

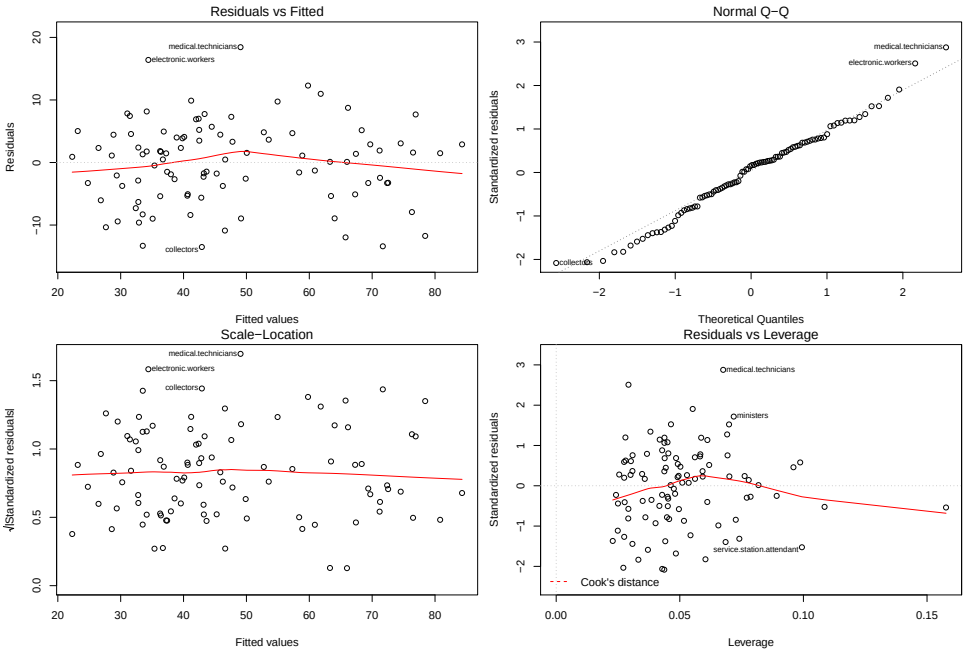
```
Call:
lm(formula = prestige ~ log(income) + education + type, data = Prestige)

Residuals:
    Min      1Q  Median      3Q     Max
-13.511  -3.746   1.011   4.356  18.438

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -81.2019    13.7431  -5.909 5.63e-08 ***
log(income)  10.4875     1.7167   6.109 2.31e-08 ***
education     3.2845     0.6081   5.401 5.06e-07 ***
typeprof      6.7509     3.6185   1.866   0.0652 .
typewc       -1.4394     2.3780  -0.605   0.5465
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.637 on 93 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared: 0.8555,        Adjusted R-squared: 0.8493
F-statistic: 137.6 on 4 and 93 DF,  p-value: < 2.2e-16
```

What happened? First of all, 4 observations were removed because the variable type contains missing values. Some occupations (athletes, babysitters, ...) cannot be assigned to a certain type. Furthermore, the question arises whether it is helpful to use type as predictor. Note that this cannot be read from an ordinary summary output. One has to perform a partial F-test. However, this can only be done if both models have been fit on the same data.

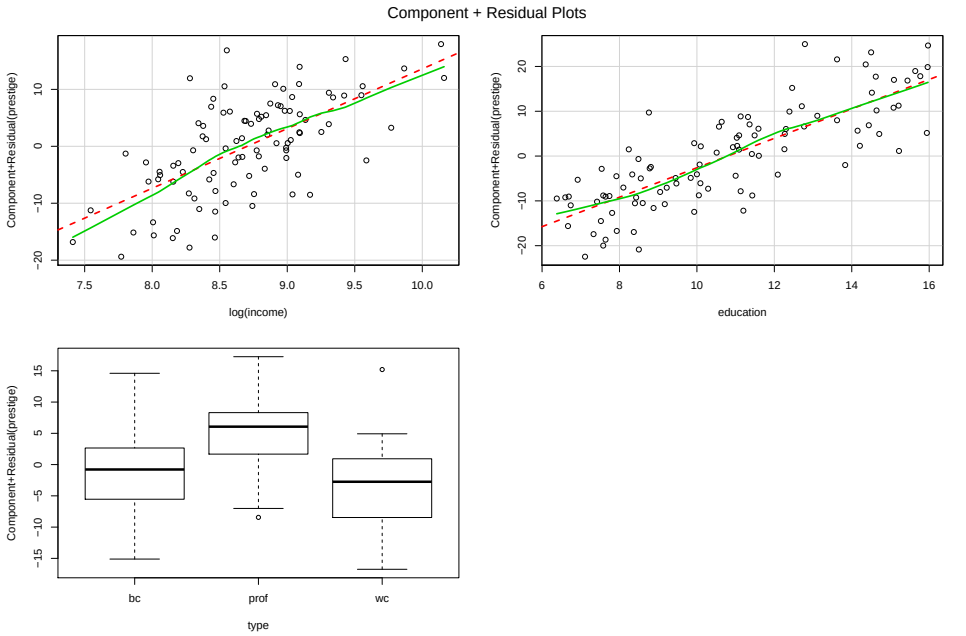
```
> fit01.rmNA <- lm(prestige ~ log(income) + education,
                   data=Prestige[-which(is.na(Prestige$type)),])
> anova(fit01.rmNA, fit02)
Analysis of Variance Table

Model 1: prestige ~ log(income) + education
Model 2: prestige ~ log(income) + education + type
  Res.Df    RSS Df Sum of Sq      F   Pr(>F)
1     95 4565.4
2     93 4096.3  2    469.07 5.3247 0.006465 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The test delivers a significant result, i.e., the null-hypothesis (the predictor type is superfluous) can be rejected. We leave type in the model and check the diagnostic plots as well as the partial residuals.

```
> par(mfrow=c(2,2))
> plot(fit02)
```




```
> crPlots(fit02)
```

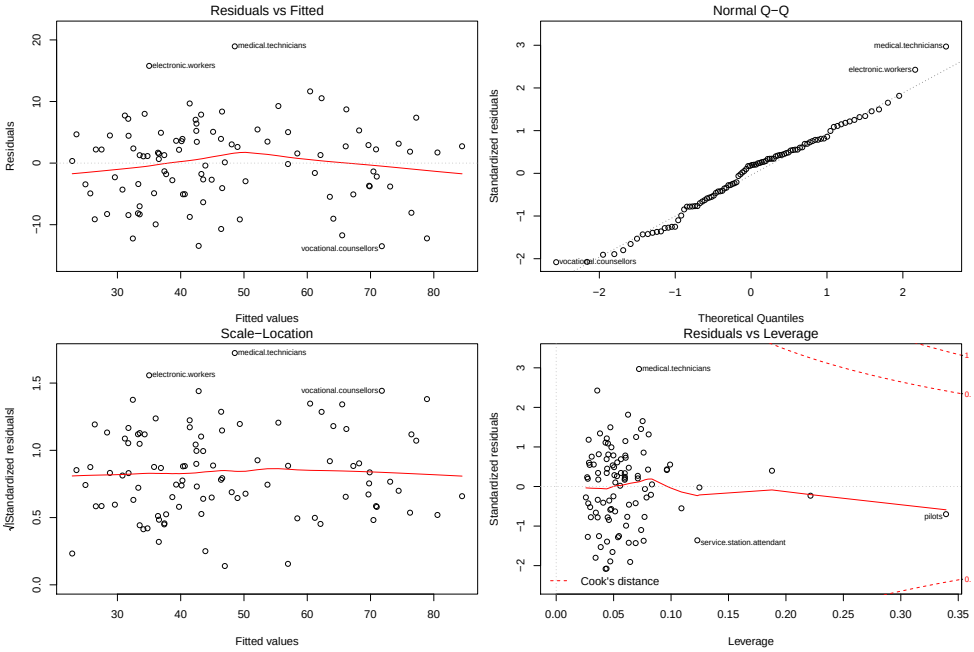


Partial residuals look good. The plot for type looks similar to the previous one using the smaller model where we plotted the residuals versus type. The residual diagnostics look also good - there is still a slight deviation from the x-axis in the Tukey-Anscombe plot. Next we check whether this could be solved with the still missing variable census.

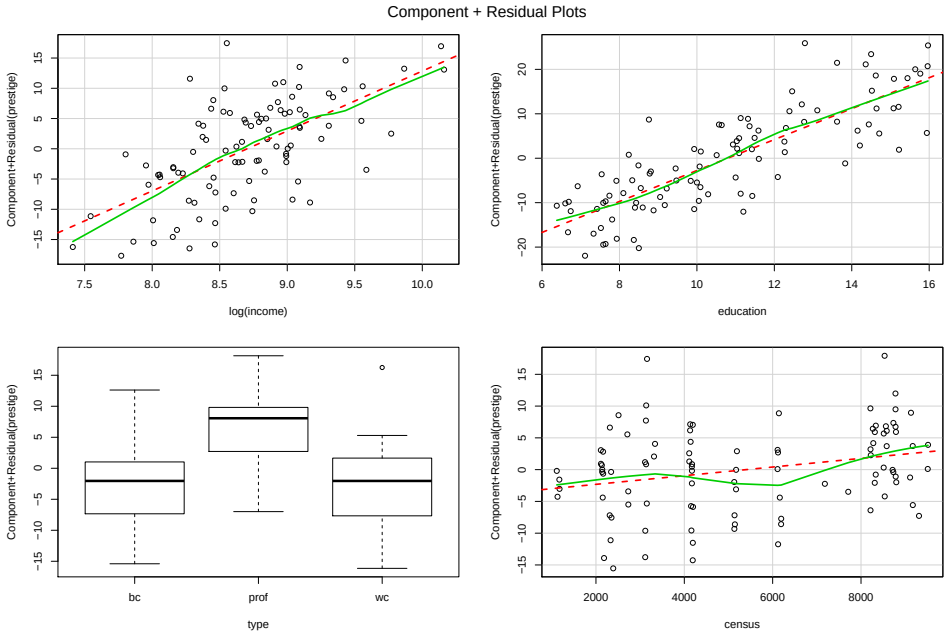
```
> fit03 <- lm(prestige ~ log(income) + education + type + census, data=Prestige)
> summary(fit03)

Call:
lm(formula = prestige ~ log(income) + education + type + census,
    data = Prestige)

Residuals:
    Min      1Q  Median      3Q     Max
-13.488  -4.239   1.214   3.932  18.954

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept) -8.331e+01  1.384e+01  -6.019 3.54e-08 ***
log(income)  9.907e+00  1.786e+00   5.546 2.79e-07 ***
education    3.481e+00  6.305e-01   5.520 3.11e-07 ***
typeprof     9.661e+00  4.409e+00   2.191    0.031 *
typewc       4.607e-01  2.891e+00   0.159    0.874
census       6.812e-04  5.919e-04   1.151    0.253
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.625 on 92 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared: 0.8575,        Adjusted R-squared: 0.8498
F-statistic: 110.8 on 5 and 92 DF,  p-value: < 2.2e-16
```

```
> par(mfrow=c(2,2))
> plot(fit03)
```

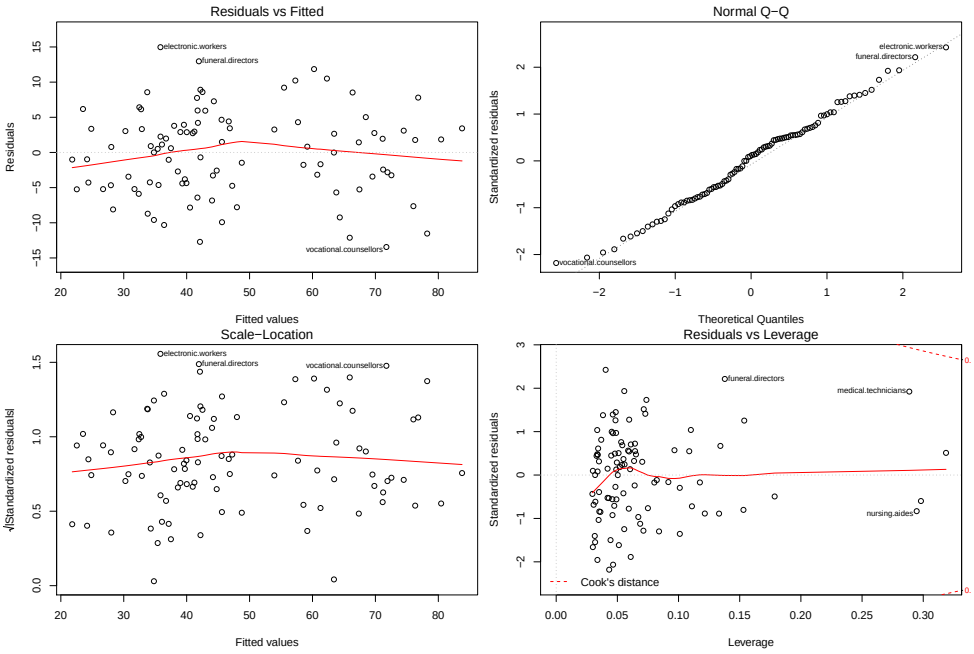



```
> par(mfrow=c(2,2))
> crPlots(fit03)
```

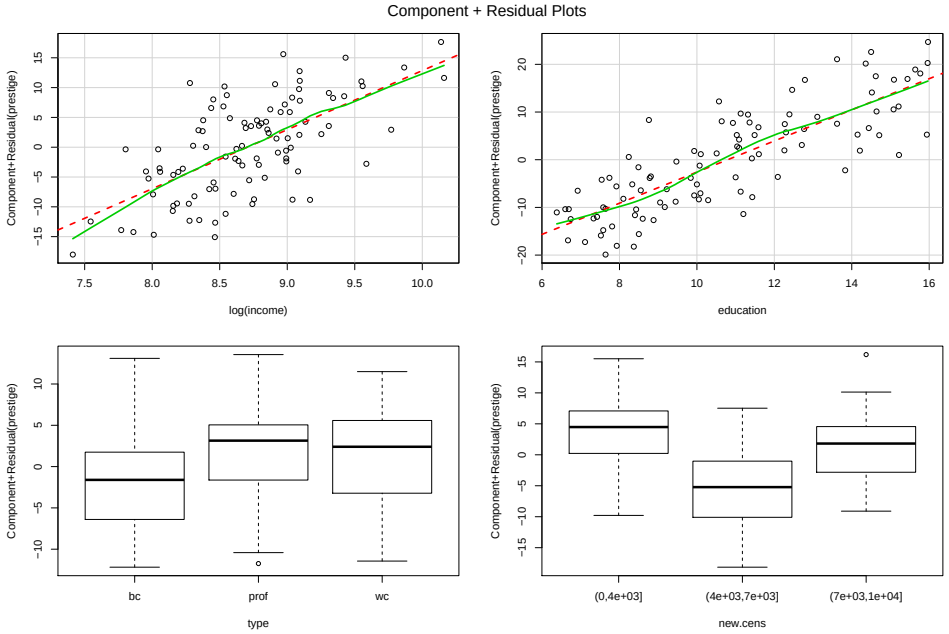


We can see that the variable census is not significant and does not improve the fit with respect to the residuals. However, the partial residual plot is conclusive: there seems to be a non-linear relation. We have noted this earlier by plotting the residuals from our starting model versus census. We will solve this issue by categorizing census:

```
> new.cens <- cut(Prestige$census, c(0, 4000, 7000, 10000))
> Prestige <- cbind(Prestige, new.cens)
> fit04 <- lm(prestige ~ log(income) + education + type + new.cens, data=Prestige)
> summary(fit04)

Call:
lm(formula = prestige ~ log(income) + education + type + new.cens,
    data = Prestige)

Residuals:
     Min       1Q   Median       3Q      Max
-13.4450  -4.3948   0.6931   3.7015  14.9686

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)           -72.3645    13.4415  -5.384 5.65e-07 ***
log(income)             9.9102     1.6819   5.892 6.35e-08 ***
education               3.2636     0.6189   5.273 8.98e-07 ***
typeprof                3.5725     4.1573   0.859   0.3924
typewc                  3.1384     2.6702   1.175   0.2429
new.cens(4e+03,7e+03]  -9.0947     3.5174  -2.586   0.0113 *
new.cens(7e+03,1e+04]  -2.4445     3.7138  -0.658   0.5120
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.304 on 91 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared: 0.8724,        Adjusted R-squared: 0.864
F-statistic: 103.7 on 6 and 91 DF,  p-value: < 2.2e-16

> par(mfrow=c(2,2))
> plot(fit04)
> anova(fit04, fit02)

Analysis of Variance Table

Model 1: prestige ~ log(income) + education + type + new.cens
Model 2: prestige ~ log(income) + education + type
  Res.Df    RSS Df Sum of Sq      F   Pr(>F)
1     91 3616.5
2     93 4096.3 -2   -479.81 6.0366 0.003453 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```




```
> par(mfrow=c(2,2))
> crPlots(fit04)
```



Using these categories we can illustrate the non-linear behaviour of census quite well. Furthermore, the factor variable is significant. We end our analysis here and decide to use `fit04` as our most suitable model.

b) Correlated errors

Use the "airquality" data set `library(faraway)`. Fit the model

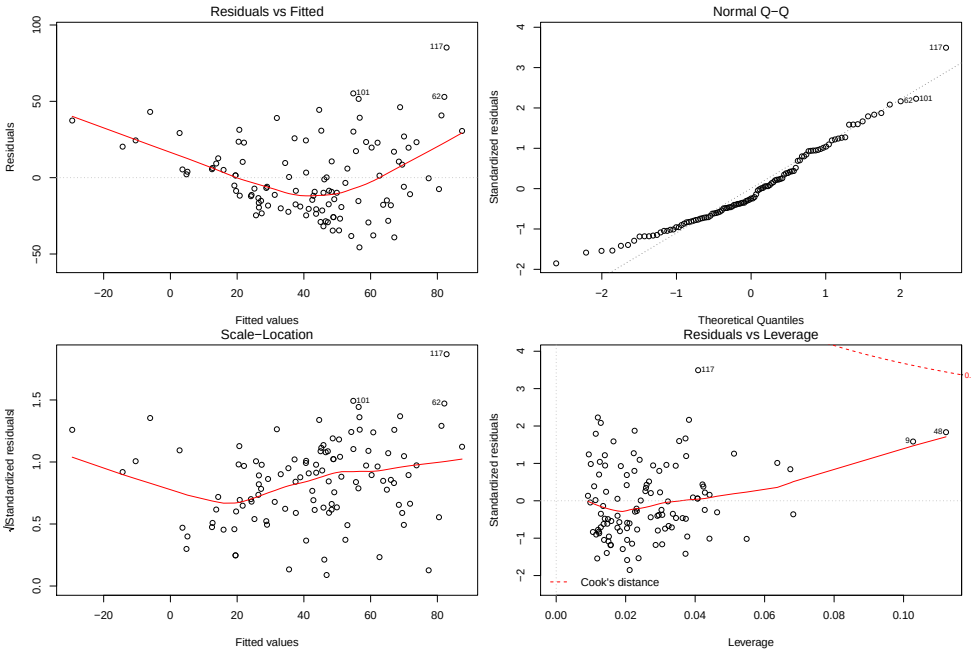
$$\texttt{Ozone} \sim \texttt{Solar.R + Wind}.$$

Perform model diagnostics and check for correlated residuals. Plot the residuals versus the variable `Temp`. Improve the model to get an optimal fit.

```
> library(faraway)
> data(airquality)
> fit00 <- lm(Ozone ~ Solar.R + Wind, data=airquality)
> summary(fit00)
```

```
Call:
lm(formula = Ozone ~ Solar.R + Wind, data = airquality)

Residuals:
    Min      1Q  Median      3Q     Max
-45.651 -18.164  -5.959  18.514  85.237

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 77.24604    9.06751   8.519 1.05e-13 ***
Solar.R      0.10035    0.02628   3.819 0.000224 ***
Wind        -5.40180    0.67324  -8.024 1.34e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 24.92 on 108 degrees of freedom
  (42 observations deleted due to missingness)
Multiple R-squared: 0.4495,        Adjusted R-squared: 0.4393
F-statistic: 44.09 on 2 and 108 DF,  p-value: 1.003e-14
```

```
> par(mfrow=c(2,2))
> plot(fit00)
```



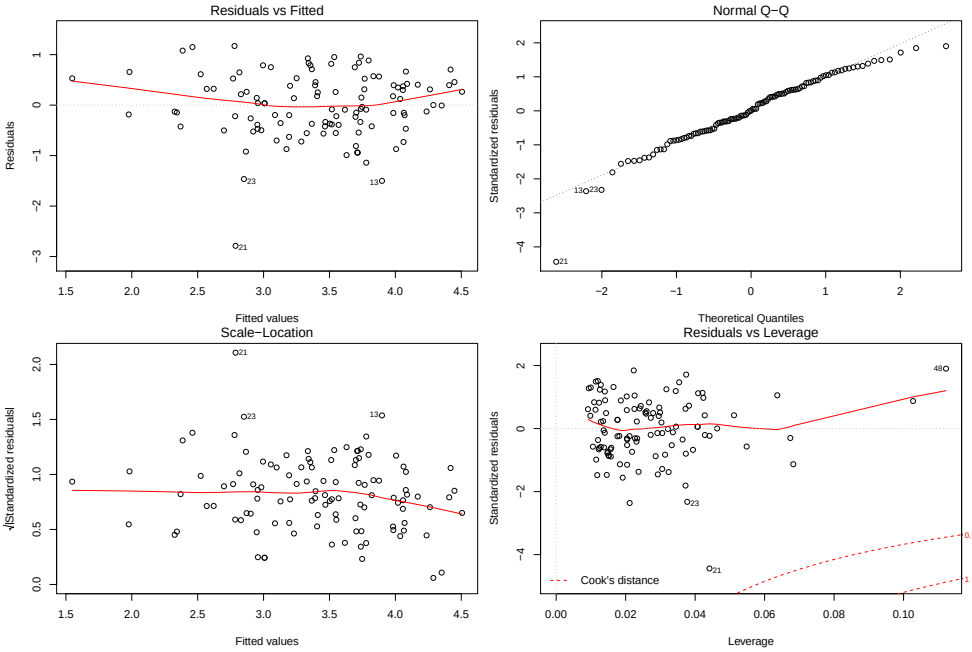
This model does not fit at all. We can see a massive systematic error in the Tukey-Anscombe plot which makes this initial model unacceptable. Additionally, several observations were removed due to missing values. First, we improve the model fit:

```
> fit01 <- lm(log(Ozone) ~ Solar.R + Wind, data=airquality)
> summary(fit01)
```

```
Call:
lm(formula = log(Ozone) ~ Solar.R + Wind, data = airquality)

Residuals:
     Min       1Q   Median       3Q      Max
-2.78747 -0.38971  0.00222  0.43882  1.17156

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.9519449  0.2337241  16.909  < 2e-16 ***
Solar.R      0.0037215  0.0006773   5.494 2.63e-07 ***
Wind        -0.1231183  0.0173535  -7.095 1.42e-10 ***
```

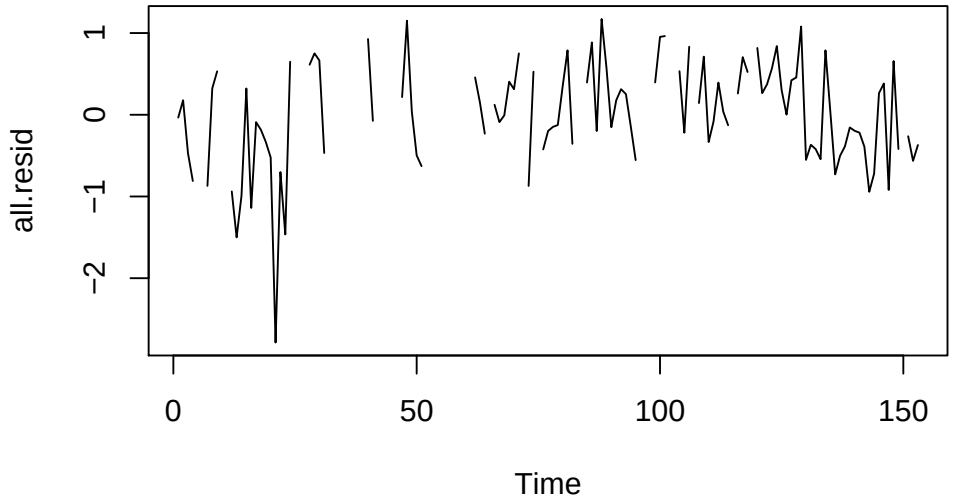
```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6423 on 108 degrees of freedom
  (42 observations deleted due to missingness)
Multiple R-squared: 0.4598,        Adjusted R-squared: 0.4498
F-statistic: 45.96 on 2 and 108 DF,  p-value: 3.612e-15
```

```
> par(mfrow=c(2,2))
> plot(fit01)
```



The Ozone variable suits a log-transformation. This improves the situation but the fit is still far from perfect. Next, we take care of the correlated residuals:

```
> all.resid <- rep(NA, 153)
> all.resid[as.numeric(names(resid(fit01)))] <- resid(fit01)
> ts.plot(all.resid)
```



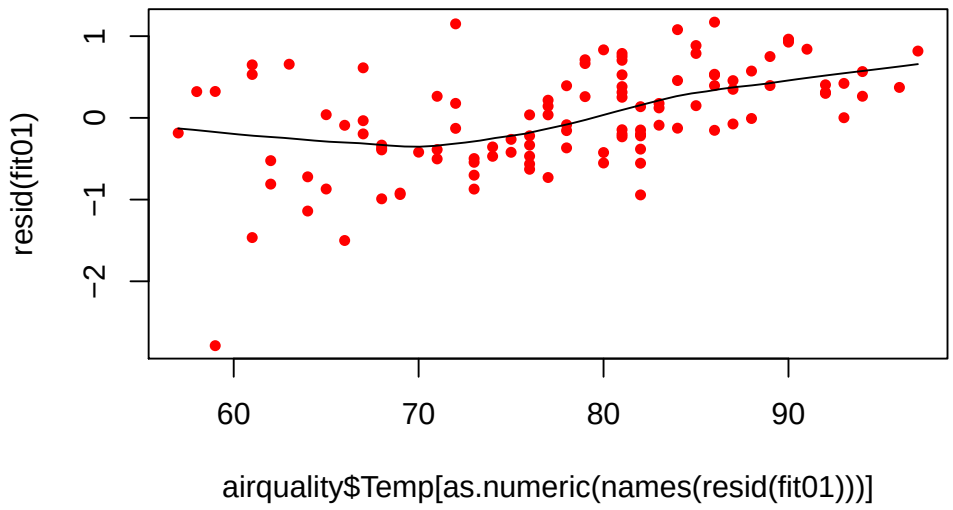
WARNING: We need to plot the residuals versus time. However, we also need to respect the time points where we have missing values. It seems difficult to tell whether the residuals are correlated from this time-series plot.

```
> library(lmtest)
> dwtest(log(Ozone) ~ Solar.R + Wind, data=airquality)
```

```
        Durbin-Watson test

data:  log(Ozone) ~ Solar.R + Wind
DW = 1.4551, p-value = 0.001734
alternative hypothesis: true autocorrelation is greater than 0
```

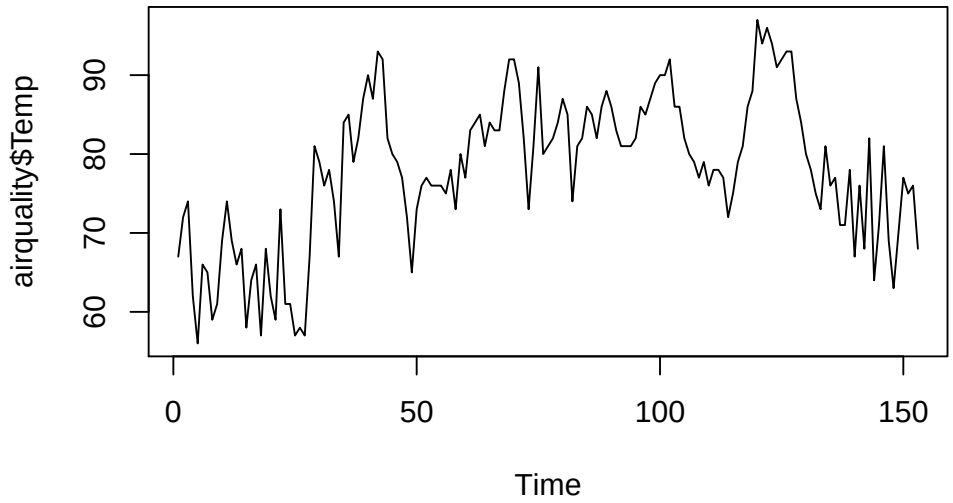
Opposed to the previous result the Durbin-Watson test is significant, rejecting the null-hypothesis of uncorrelated residuals. We now plot the residuals versus the temperature:

```
> scatter.smooth(airquality$Temp[as.numeric(names(resid(fit01)))],
                 resid(fit01), pch=20, col="red")
```



We can see a relation between the residuals and Temp. Residuals tend to be larger when the temperature is high, i.e., the ozone concentration is underestimated in this setting. Therefore, we have to include the variable Temp into the model. Additionally, Temp is autocorrelated (cross-correlated with itself).

```
> ts.plot(airquality$Temp)
```

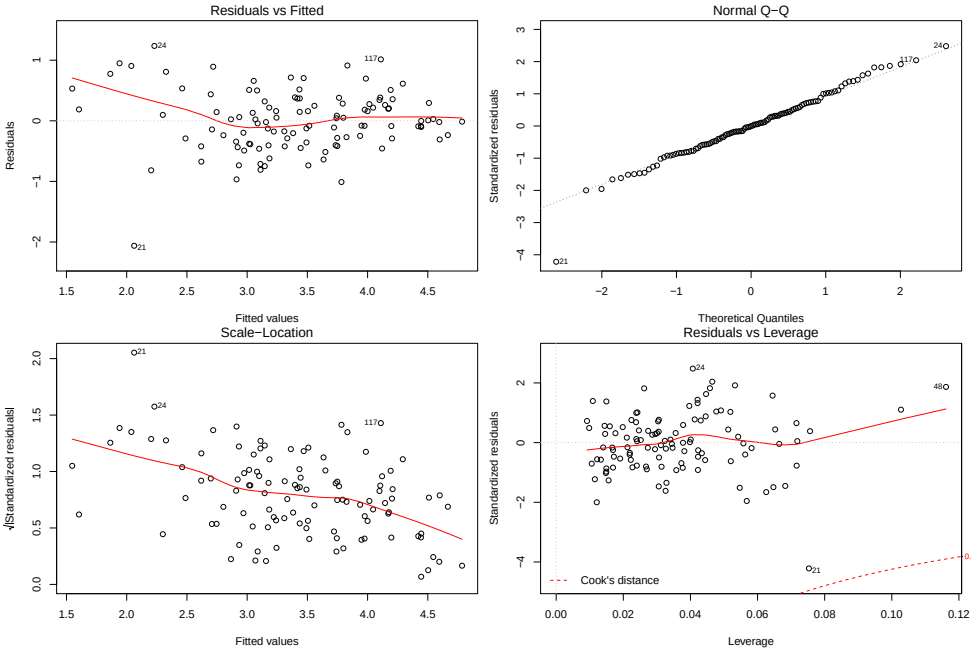


Days with high temperature are generally followed by days with high temperature, and the same holds for cold days. Since this missing predictors is temporally correlated the residuals are correlated as well.

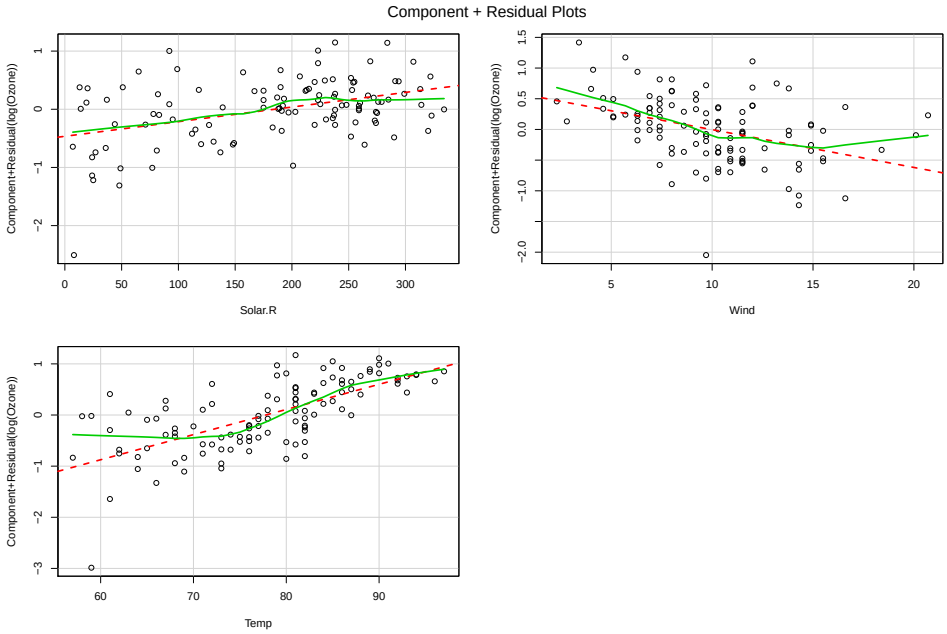
```
> fit02 <- lm(log(Ozone) ~ Solar.R + Wind + Temp, data=airquality)
> summary(fit02)

Call:
lm(formula = log(Ozone) ~ Solar.R + Wind + Temp, data = airquality)
```

```
Residuals:
     Min       1Q   Median       3Q      Max
-2.06193 -0.29970 -0.00231  0.30756  1.23578

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.2621323  0.5535669  -0.474 0.636798
Solar.R      0.0025152  0.0005567   4.518 1.62e-05 ***
Wind        -0.0615625  0.0157130  -3.918 0.000158 ***
Temp         0.0491711  0.0060875   8.077 1.07e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.5086 on 107 degrees of freedom
  (42 observations deleted due to missingness)
Multiple R-squared: 0.6644,        Adjusted R-squared: 0.655
F-statistic: 70.62 on 3 and 107 DF,  p-value: < 2.2e-16
```

```
> par(mfrow=c(2,2))
> plot(fit02)
```




```
> par(mfrow=c(2,2))
> crPlots(fit02)
> dwtest(log(Ozone) ~ Solar.R + Wind + Temp, data=airquality)

        Durbin-Watson test

data:  log(Ozone) ~ Solar.R + Wind + Temp
DW = 1.8068, p-value = 0.1334
alternative hypothesis: true autocorrelation is greater than 0
```



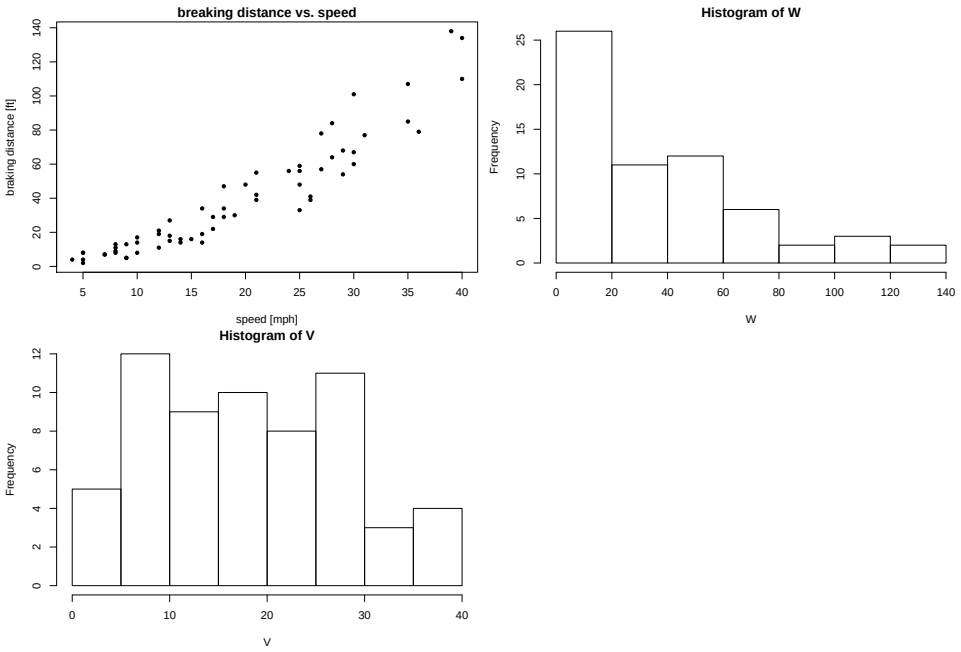
The problem with the correlated residuals is solved. The predictor Temp is significant and the summary looks tidy. Residual plots look better than before, although still including some deviations. We can see in the partial residual plots some deviation from linearity. We could use the same trick of categorizing as in exercise **1.** but in the present case that would be quite difficult. Using this data set we have reached the limits of the methods which are available to us. More flexible solutions like generalized additive models (GAM) would allow for a better modeling of the data.

3. Braking distance

The file bremsweg.rda contains measurements of braking distance (W, in feet) together with specific starting velocities (V, in mph). Perform a regression analysis.

a) Generate a scatter plot and solve any problems with the data if necessary.

```
> con <- url("http://stat.ethz.ch/education/semesters/as2011/asr/bremsweg.rda")
> load(con)
> par(mfrow=c(2,2))
> with(bremsweg, {
   plot(W ~ V, xlab= "speed [mph]", ylab="braking distance [ft]", pch=20)
   title("breaking distance vs. speed")
   hist(W)
   hist(V)
 })
```

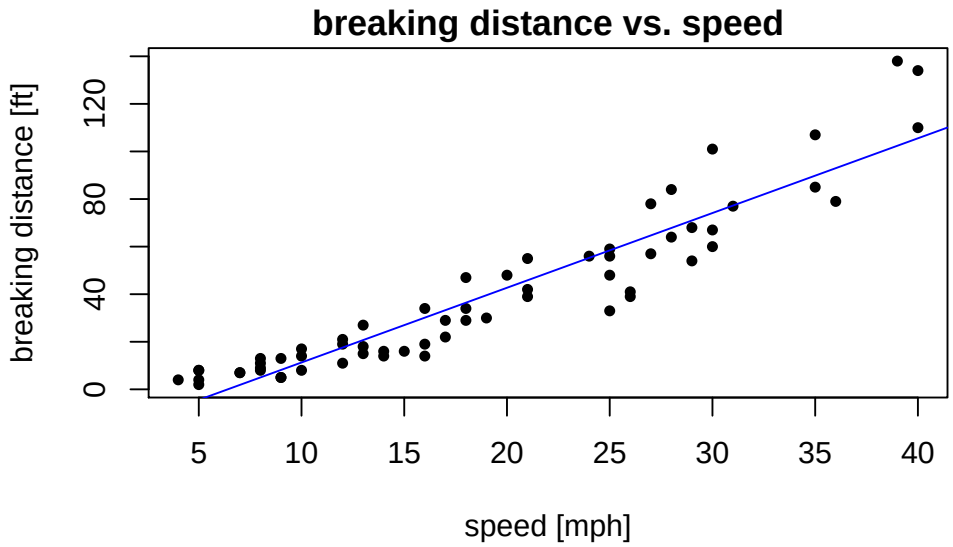


Braking distances are quite strongly right-skewed distributed. However, there are no further peculiarities. Although First-Aid-Transformations would be applicable we are not performing any because of the physical properties of braking distance suggesting that the relation between braking distance and velocity is probably described as a polynomial of second degree.

b) Fit a suitable polynomial regression model.

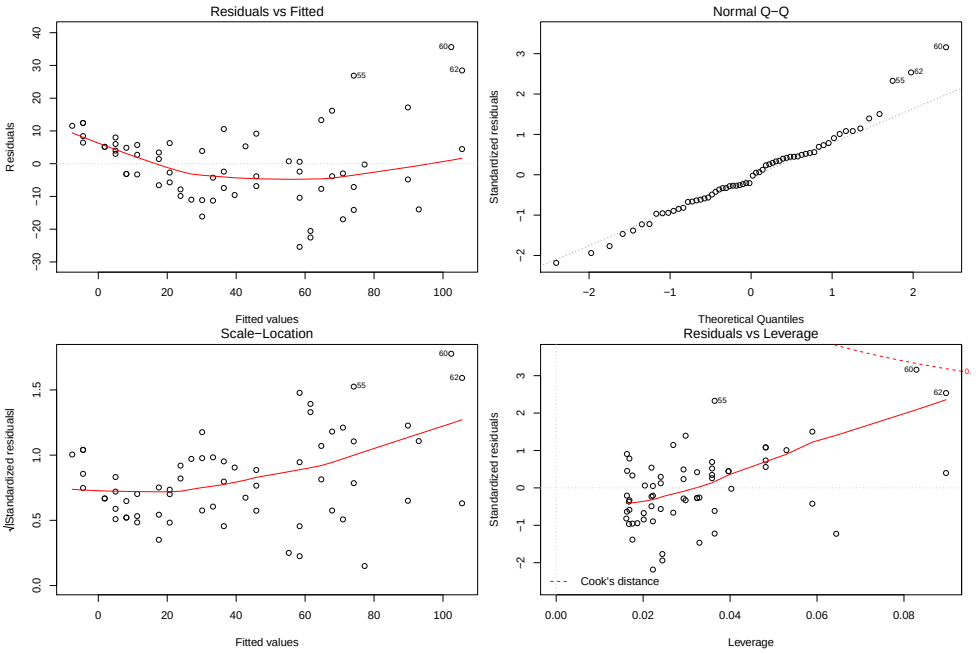
c) Do you think this model is physically reasonable?
First, we fit a linear regression.

```
> fit.ord1 <- lm(W ~ V, data=bremsweg)
> with(bremsweg, {
   plot(W ~ V, xlab="speed [mph]", ylab="breaking distance [ft]", pch=20)
   title("breaking distance vs. speed")
   abline(fit.ord1, col="blue")
   summary(fit.ord1)
 })

Call:
lm(formula = W ~ V, data = bremsweg)

Residuals:
    Min      1Q  Median      3Q     Max
-25.410  -7.343  -1.334   5.927  35.608

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -20.1309     3.2308  -6.231 5.04e-08 ***
V             3.1416     0.1514  20.751  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 11.77 on 60 degrees of freedom
Multiple R-squared: 0.8777,        Adjusted R-squared: 0.8757
F-statistic: 430.6 on 1 and 60 DF,  p-value: < 2.2e-16
```



The summary output looks promising. However, the scatter plots show that the regression line cannot model the curvilinear relation. Furthermore, the residual analysis exhibits clear patterns:

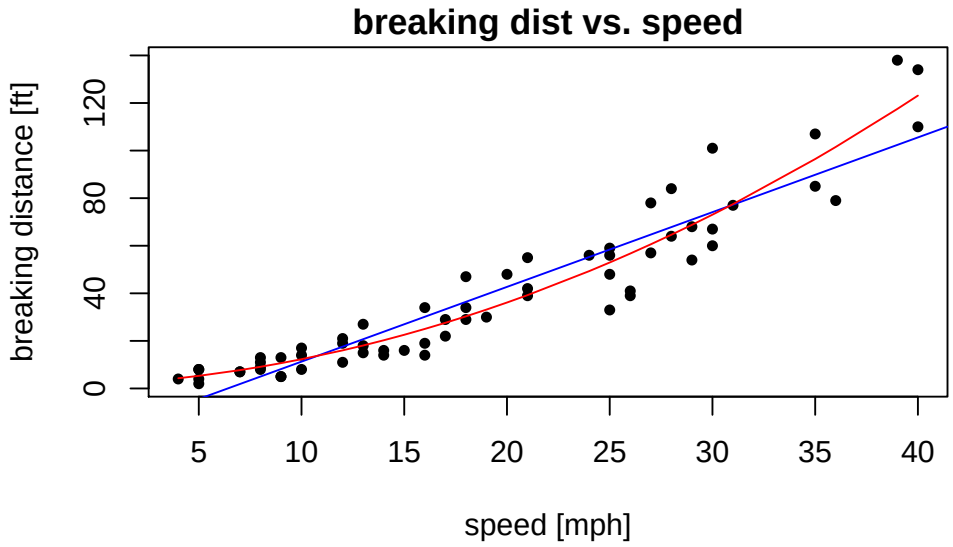
```
> par(mfrow=c(2,2))
> plot(fit.ord1)
```



The first thing meeting the eye are the structural deficiencies in the Tukey-Anscombe plot. Furthermore, the variance seems non-constant. This is why we try to fit a polynomial of second degree.

```
> fit.ord2 <- lm(W ~ V + I(V^2), data=bremsweg)
> with(bremsweg, {
   plot(W ~ V, xlab="speed [mph]", ylab="breaking distance [ft]", pch=20)
   title("breaking dist vs. speed")
   abline(fit.ord1, col="blue")
   lines(sort(V), fitted(fit.ord2)[order(V)], col="red")
   summary(fit.ord2)
 })

Call:
lm(formula = W ~ V + I(V^2), data = bremsweg)

Residuals:
     Min       1Q   Median       3Q      Max
```

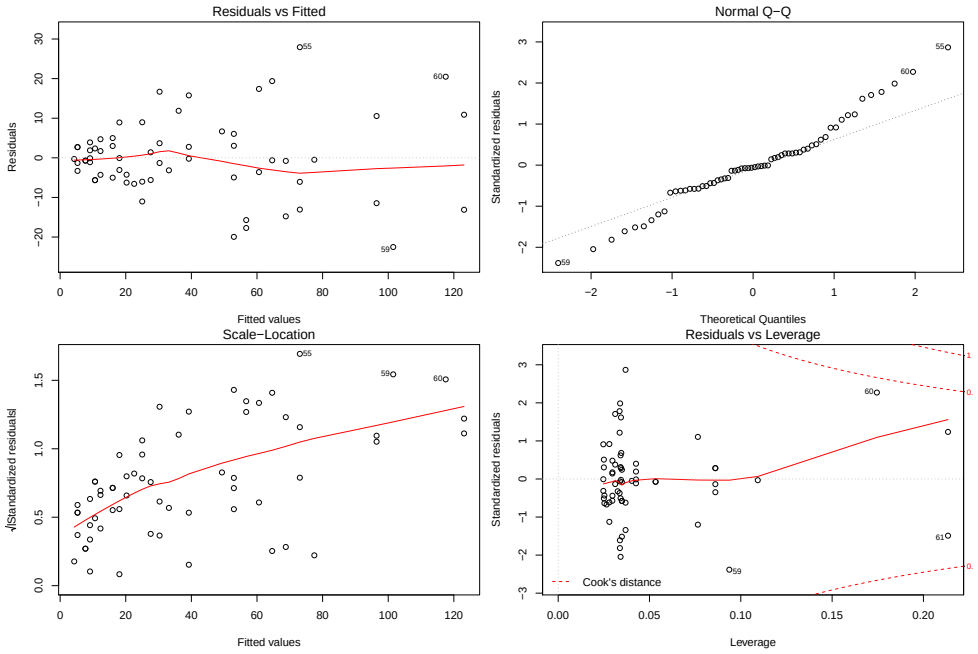
```
-22.5192  -5.4527  -0.5519   3.8442  27.9373

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.58036    5.10266   0.310    0.758
V            0.41607    0.55641   0.748    0.458
I(V^2)       0.06556    0.01303   5.033 4.83e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 9.927 on 59 degrees of freedom
Multiple R-squared: 0.9144,        Adjusted R-squared: 0.9115
F-statistic: 315.3 on 2 and 59 DF,  p-value: < 2.2e-16
```



The estimated error variance has reduced and adjusted R-squared has increase. The new model seems to fit better to the data. Note that the first order term is not significant but should not be removed. If higher order terms are involved, lower order terms should remain in the model even if they are not significant. Anyhow, the term can be physically explained by the reaction time. We can deduce the reaction time from our estimates. For each additional mile per hour of velocity (1.608km/h, resp. 0.446m/s) the braking distance is increased by 0.416068448746556 feet (i.e., 0.127m) due to the linear term. By dividing we get 0.127/0.447m/s = 0.28s - a reasonable value.

d) Perform a residual analysis. Which assumptions are violated?

```
> par(mfrow=c(2,2))
> plot(fit.ord2)
```



The structural deficiencies are gone, i.e., the expectation of the error is zero. However, the assumption of constant variance is still violated. This is why the normal Q-Q plot shows a long-tailed pattern.

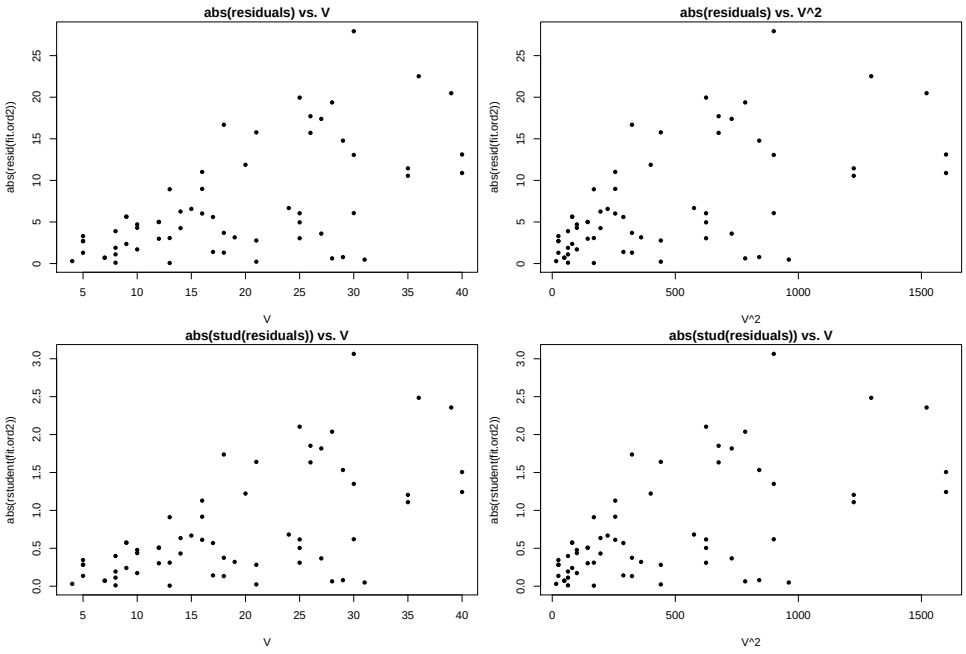
Since the fit opposed to the previous statement is good and respects the physical properties of the subject, a possible improvement can be obtained by using weighted regression which is able to handle such situations.

e) Weighted regression

Previously you have seen that the variance is not constant. Therefore, we fit a suitable weighted regression. Compare the results from the weighted and the not-weighted regression (e.g. summary, fitted values, plot fitted curves, residual analysis) and comment on the results.

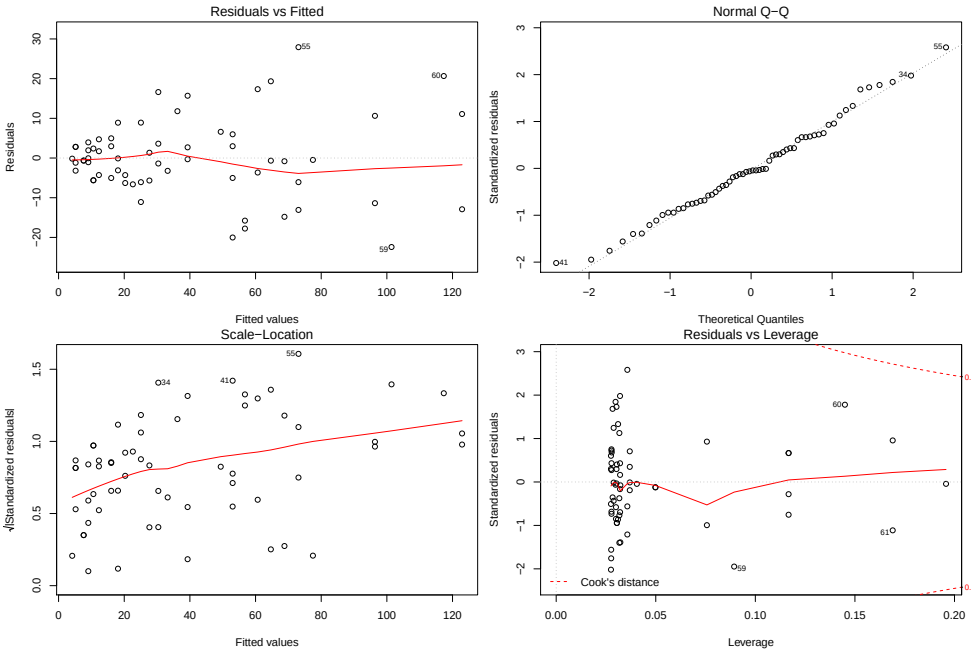
First, we need to choose suitable weights. To this end, we plot the (studentized) residuals of the polynomial model versus the predictors.

```
> par(mfrow=c(2,2))
> with(bremsweg, {
   plot(V, abs(resid(fit.ord2)), pch=20, main="abs(residuals) vs. V")
   plot(V^2, abs(resid(fit.ord2)), pch=20, main="abs(residuals) vs. V^2")
   plot(V, abs(rstudent(fit.ord2)), pch=20, main="abs(stud(residuals)) vs. V")
   plot(V^2, abs(rstudent(fit.ord2)), pch=20, main="abs(stud(residuals)) vs. V")
 })
```



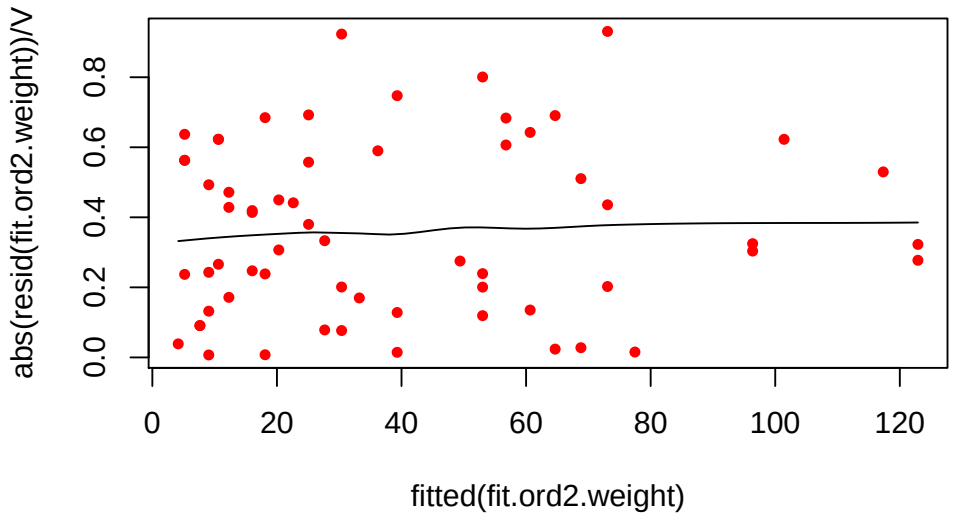
We can see that the variance is proportional to `V` and not V^2. Therefore, we choose the weights as `1/V`.

```
> fit.ord2.weight <- lm(W ~ V + I(V^2), weights=1/V, data=bremsweg)
> summary(fit.ord2.weight)

Call:
lm(formula = W ~ V + I(V^2), data = bremsweg, weights = 1/V)

Residuals:
    Min      1Q  Median      3Q     Max
-4.0037 -1.4120 -0.1054  1.2586  5.0984

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.32590    3.09898   0.428    0.670
V            0.44801    0.42065   1.065    0.291
I(V^2)       0.06479    0.01122   5.777 3.03e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.011 on 59 degrees of freedom
Multiple R-squared: 0.923,        Adjusted R-squared: 0.9204
F-statistic: 353.8 on 2 and 59 DF,  p-value: < 2.2e-16

> par(mfrow=c(2,2))
> plot(fit.ord2.weight)
```



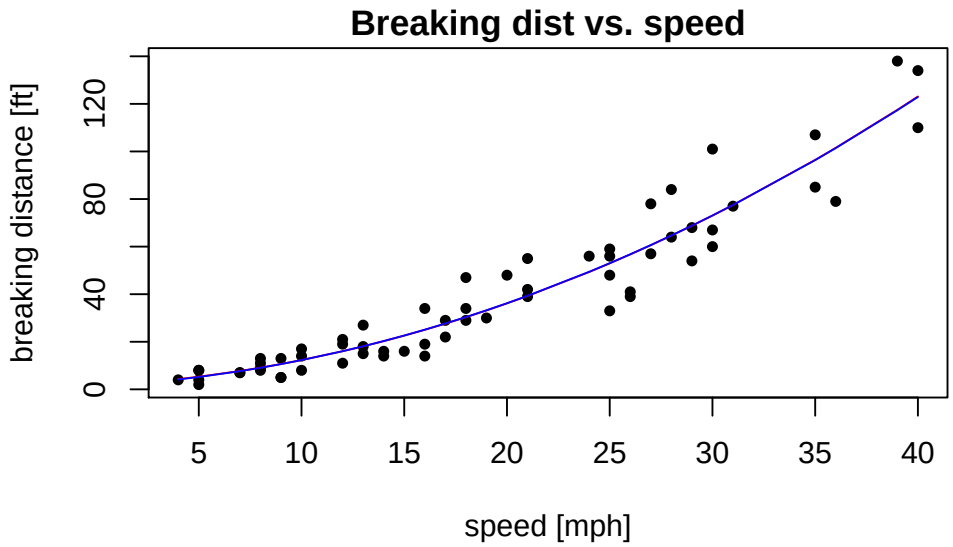
To check the variance assumption we need to plot the "weighted" residuals versus the fitted values, i.e., generating the scale-location plot by hand.

```
> with(bremsweg, {
    scatter.smooth(fitted(fit.ord2.weight), abs(resid(fit.ord2.weight))/V,
                   col="red", pch=20)
  })
```



This looks much better. The weighting was successful. We conclude with plotting the fit.

```
> with(bremsweg, {
    plot(W ~ V, xlab="speed [mph]", ylab="breaking distance [ft]", pch=20)
    title("Breaking dist vs. speed")
    lines(sort(V), fitted(fit.ord2)[order(V)], col="red")
    lines(sort(V), fitted(fit.ord2.weight)[order(V)], col="blue")
  })
```



No differences can be seen from the scatter plot, although the standard error is estimated in different ways.

f) Robust regression

We use the data set `data(gala)` from the package `library(faraway)`. Fit a model with the following formula:

$$\texttt{Species} \sim \texttt{Area + Elevation + Scruz + Nearest + Adjacent}$$

Note that in this case the variables should be transformed. Take a look at the residual plots and fit a robust model. Compare the "blind fit" from the above formula with your best robust model fit using the transformed variables. Comment on your results. You can find additional information regarding the data set in the corresponding help file by using the command `?gala`.

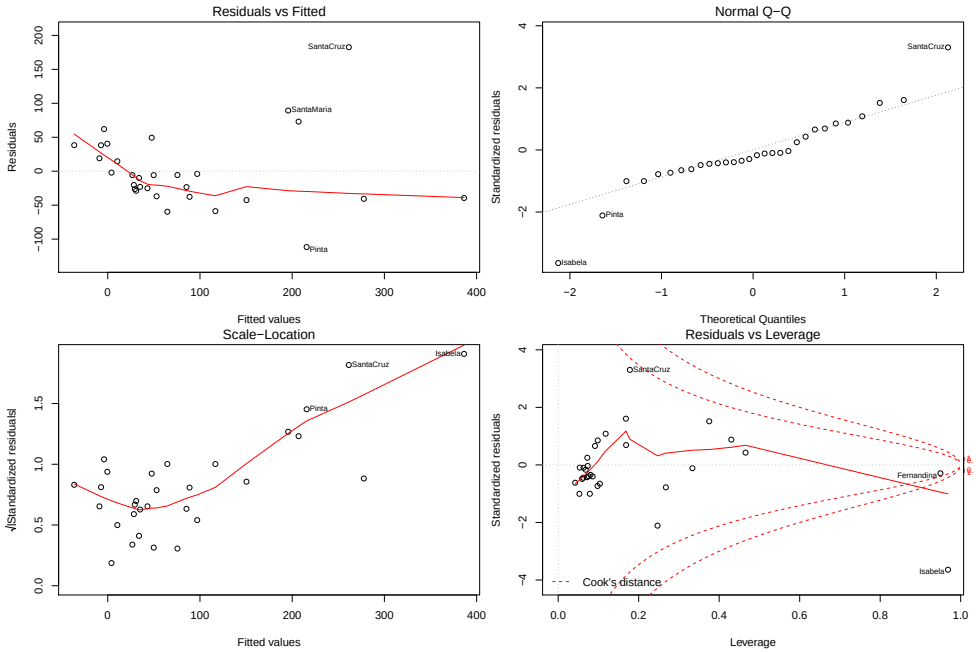
```
> data(gala, package="faraway")
> fit0 <- lm(Species ~ Area + Elevation + Scruz + Nearest + Adjacent, data=gala)
> summary(fit0)

Call:
lm(formula = Species ~ Area + Elevation + Scruz + Nearest + Adjacent,
    data = gala)

Residuals:
     Min       1Q   Median       3Q      Max
-111.679  -34.898   -7.862   33.460  182.584

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.068221  19.154198   0.369 0.715351
Area        -0.023938   0.022422  -1.068 0.296318
Elevation    0.319465   0.053663   5.953 3.82e-06 ***
Scruz       -0.240524   0.215402  -1.117 0.275208
Nearest      0.009144   1.054136   0.009 0.993151
Adjacent    -0.074805   0.017700  -4.226 0.000297 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 60.98 on 24 degrees of freedom
Multiple R-squared: 0.7658,        Adjusted R-squared: 0.7171
F-statistic:  15.7 on 5 and 24 DF,  p-value: 6.838e-07
```

We can see from the summary output and the residual plots that the present model is not suitable for describing the given data:

Only `Elevation` and `Adjacent` seem to have a significant influence on `Species`. Furthermore, the residual sum of square is quite large.

The diagnostic plots show a strong violation of constant variance and normality. Additionally, Cook's distances shwo two observations being quite influential and one observation (`Isabela`) being a leverage point.

```
> par(mfrow=c(2,2))
> plot(fit0)
```



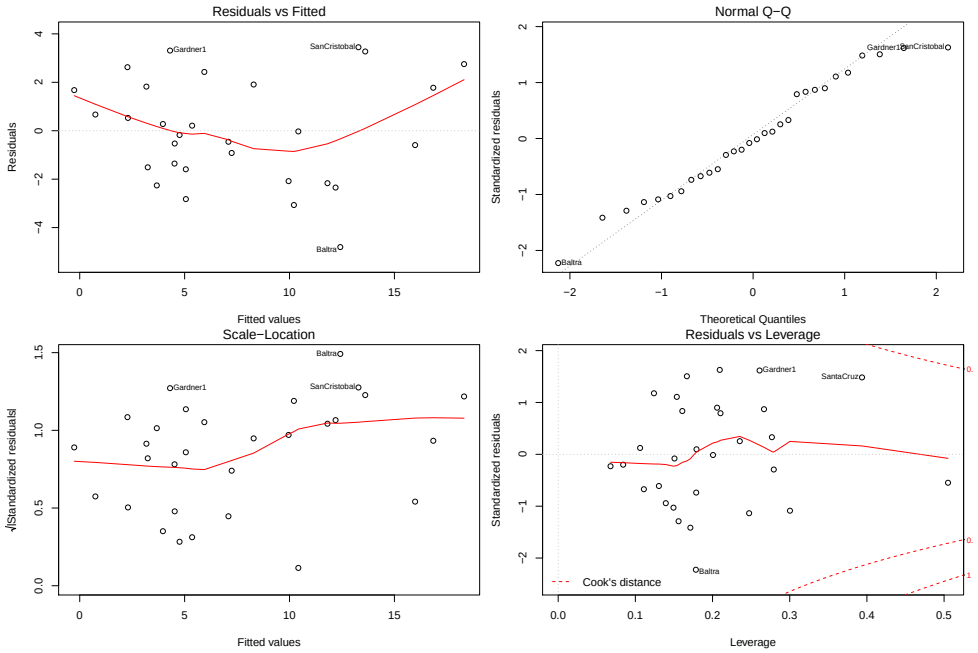
These structural deficiencies suggest the necessity of transformations. Since the response `Species` is a count, a square root transformation would be reasonable. We also log transform the predictors due to their right-skewed distribution. Note that `Scruz` contains some zero values which means we have to add the smallest positive value to all entries of `Scruz`.

```
> gala <- within(gala, {
    sqrt.Species <- sqrt(Species)
    log.Area <- log(Area)
    log.Elevation <- log(Elevation)
    log.Scruz <- log(Scruz + min(Scruz[Scruz > 0]))
    log.Nearest <- log(Nearest)
    log.Adjacent <- log(Adjacent)
  })
> fit1 <- lm(sqrt.Species ~ log.Area + log.Elevation + log.Scruz +
             log.Nearest + log.Adjacent, data=gala)
> summary(fit1)

Call:
lm(formula = sqrt.Species ~ log.Area + log.Elevation + log.Scruz +
    log.Nearest + log.Adjacent, data = gala)

Residuals:
    Min      1Q  Median      3Q     Max
-4.8046 -1.5716 -0.1018  1.8105  3.4476

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)     9.8839     5.0523   1.956   0.0622 .
log.Area        1.5609     0.3025   5.160 2.77e-05 ***
log.Elevation  -0.4440     0.9777  -0.454   0.6538
log.Scruz      -0.3989     0.3267  -1.221   0.2339
log.Nearest    -0.5273     0.3495  -1.509   0.1444
log.Adjacent   -0.2737     0.1387  -1.973   0.0602 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 2.382 on 24 degrees of freedom
Multiple R-squared: 0.8398,        Adjusted R-squared: 0.8064
F-statistic: 25.15 on 5 and 24 DF,  p-value: 8.176e-09
```

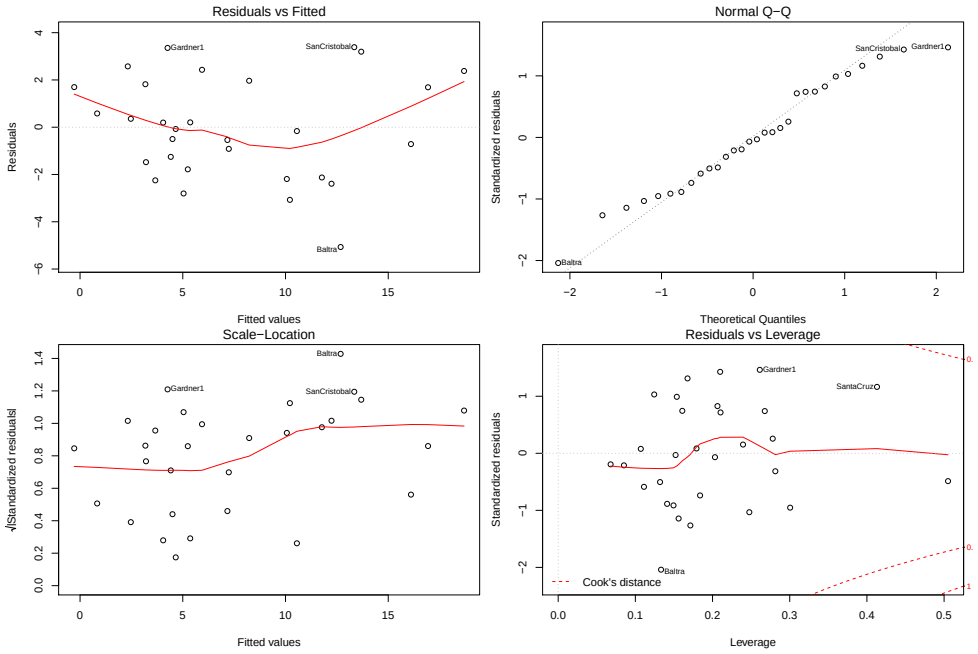
After transforming the variables the summary output looks remarkably better. Only `log.Area` remains significant but the residual sum of squares is much smaller and the adjusted R-squared higher than before.

```
> par(mfrow=c(2,2))
> plot(fit1)
```



The residual plots look somewhat better now, i.e., error variance looks more constant, normality seems valid and there are no more leverage points. Anyhow, the linear model seems still not suitable since the Tukey-Anscombe plot still exhibits a U-shaped pattern.
We will estimate the parameters in a robust fashion. Maybe we get a better fit that way.

```
> library(MASS)
> fit2 <- rlm(sqrt.Species ~ log.Area + log.Elevation + log.Scruz +
              log.Nearest + log.Adjacent, data=gala)
> summary(fit2)

Call: rlm(formula = sqrt.Species ~ log.Area + log.Elevation + log.Scruz +
    log.Nearest + log.Adjacent, data = gala)
Residuals:
    Min      1Q  Median      3Q     Max
-5.0685 -1.7052 -0.1185  1.7891  3.3866

Coefficients:
             Value   Std. Error t value
(Intercept)   10.2538  5.0181     2.0433
log.Area       1.5808  0.3004     5.2614
log.Elevation -0.4725  0.9711    -0.4866
log.Scruz     -0.4568  0.3245    -1.4076
log.Nearest   -0.5294  0.3471    -1.5250
log.Adjacent  -0.2811  0.1378    -2.0395

Residual standard error: 2.669 on 24 degrees of freedom

> par(mfrow=c(2,2))
> plot(fit2)
```



Using a robust estimation of the parameters did not improve the model fit. Probably because the residuals are sufficiently normal distributed after the transformation. Except for the test statistic of the intercept, the parameter estimates differ only slightly from the previous ones. The Tukey-Anscombe plot illustrates that the structural problems cannot be solved with robust estimation of the parameters.