Linear Models, List 3

Klaudia Weigel

1 Exercise 1

a) *Use R to find the critical value that you would use for a two-tailed t signicance test with* $\alpha = 0.05$ *and 10 degrees of freedom.*
For a two-sided test with a test statistic $T \sim_{H_0} t(df)$, we find the critical value from

$$P_{H_0}(|T| > t_c) = \alpha \implies t_c = t^*(1 - \alpha/2, df).$$

```
df = 10
alpha = 0.05
t_c = qt(1 - alpha/2, df)
t_c
```

```
## [1] 2.228139
```

b) *Use R to find the critical value that you would use for an F significance test with* $\alpha = 0.05$*, one degree of freedom in the numerator and 10 degrees of freedom in the denominator.*

We have a test statistic $F \sim F(df1, df2)$. We reject the null hypothesis when the value of our statistic We find F_c from the fact that

$$P(F > F_c) = \alpha \implies F_c = F^*(1 - \alpha, df1, df2).$$

```
F_c = qf(1-alpha, 1, 10)
F_c
```

```
## [1] 4.964603
```

c) *Verify that the square of tc is Fc.*

```
t_c^2
```

```
## [1] 4.964603
```

```
F_c
```

```
## [1] 4.964603
```

2 Exercise 2

We have a part of ANOVA table:

Source	df	SS
Model	1	100
Error	20	400

A full ANOVA table for a linear model $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$, $i = 1, ..., n$ looks as follows:

Source	df	SS	MS
Model	dfM = 1	SSM = $\sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2$	MSM = SSM/dfM
Error	dfE = n - 2	SSE = $\sum_{i=1}^{n}(Y_i - \hat{Y})^2$	MSE = SSE/dfE = s^2
Total	dfT = n - 1	SST = $\sum_{i=1}^{n}(Y_i - \bar{Y})^2$	MST = SST/dfT

In addition we have SST = SSM + SSE, dfT = dfM + dfE.

a) *How many observations do you have in your file?*
We have n = dfE + 2 = 22 observations.

```
dfM = 1; dfE = 20; SSM = 100; SSE = 400
n = dfE + 2
```

b) *Calculate the estimate of σ.*
$\sigma = \sqrt{SSE/dfE} = \sqrt{400/20} = 2\sqrt{5}$.

```
MSM = SSM/dfM; MSE = SSE/dfE; SST = SSM + SSE
sqrt(MSE)
```

```
## [1] 4.472136
```

c) *Test if β_1 is equal to zero. (Give the test statistic with the numbers of degrees of freedom and the conclusion).*

We will use an F test. The test statistic is $F = MSM/MSE$. The statistic has the F distribution with 1 and n-2 degrees of freedom. We reject the null hypothesis stating that $\beta_1 = 0$, when $F > F_c = F^*(1 - \alpha, 1, n - 2)$.

```
F_stat = MSM/MSE
F_c = qf(1-alpha, 1, n-2)
F_stat > F_c
```

```
## [1] TRUE
```

Since $F > F_c$ we reject the null hypothesis and conclude that $\beta_1 \neq 0$. We can also compute the p-value.

```
# p-value
(1 - pf(abs(F_stat), 1, n-2))
```

```
## [1] 0.03690484
```

The p-value confirms our previous conclusion as it is smaller that the chosen significance level 0.05.

d) *What proportion of the variation of the response variable is explained by your model?*

We calculate this proportion with $R^2 = SSM/SST$.

```
R_2 = SSM/SST
R_2
```

```
## [1] 0.2
```

e) *What is the sample correlation coefficient between your response and explanatory variables?*

When we have only one predictor variable the correlation coefficient between the response and the predictor variable is $\sqrt{R^2}$.

```
r = sqrt(R_2)
r
```

```
## [1] 0.4472136
```

3 Exercise 3

For this and the next problem we will use a data set containing the grade point average (GPA) [second column], score on a standard IQ test [third column], gender and a score on the Piers-Harris Childrens Self-Concept Scale (a psychological test, fifth column) for 78 seventh-grade students.

a) *Use a simple regression model to describe the dependence of gpa on the results of iq test. Report the fitted regression equation and R2. Test the hypothesis that gpa is not correlated with iq : give the test statistic, p-value and the conclusion in words.*

Our theoretical model is:

$$GPA_i = \beta_0 + \beta_1 IQ_i + \epsilon_i, \quad i = 1, ..., 78.$$

Where epsilons are iid and come from $N(0, \sigma^2)$ distribution.

The estimators for β_0 and β_1 are:

$$\hat{\beta_0} = \overline{GPA} - \beta_1 \overline{IQ}, \qquad \hat{\beta_1} = \frac{\sum_{i=1}^{78}(IQ_i - \overline{IQ})(GPA_i - \overline{GPA})}{\sum_{i=1}^{78}(IQ_i - \overline{IQ})^2}.$$

The estimator of the variance of the residuals is:

$$s^2 = \frac{1}{76}\sum_{i=1}^{78}(GPA_i - \widehat{GPA_i})^2.$$

```
dat = read.table("tabela1_6.txt", col.names = c("id", "gpa", "iq", "gender", "piers_harris"))
Y = dat$gpa; X = dat$iq; n = length(Y)
model = lm(gpa ~ iq, data = dat)
model$coefficients
```

```
## (Intercept)          iq
##  -3.5570558   0.1010217
```

The regression equation is

$$E(GPA) = -3.5570558 + 0.1010217 IQ.$$

We will get the value of R^2 from `summary`.

```
summary(model)$r.squared
```

```
## [1] 0.4016146
```

To test whether gpa and iq are correlated we will use a t-test in order to test the significance of β_1 ($H_0 : \beta_1 = 0$ against $H_1 : \beta_1$). The test statistic is $T = \frac{\hat{\beta_1}}{s(\hat{\beta_1})} \sim t(76)$, where $s^2(\hat{\beta_1}) = \frac{s^2}{\sum_{i=1}^{78}(IQ_i - \overline{IQ})^2}$.

```
summary(model)[["coefficients"]][, "t value"][2]
```

```
##      iq
## 7.14202
```

The p-value

```
summary(model)[["coefficients"]][, "Pr(>|t|)"][2]
```

```
##           iq
## 4.737341e-10
```

The p-value is less than 0.05, therefore for this significance level we reject the null hypothesis and conclude that gpa depends on iq.

b) *Predict gpa for a student whose iq is equal to 100. Report 90% prediction interval.*

The prediction is $\widehat{GPA} = \hat{\beta}_0 + \hat{\beta}_1 * 100 = -3.5570558 + 0.1010217 * 100 = 6.545114$. The 90% prediction interval is

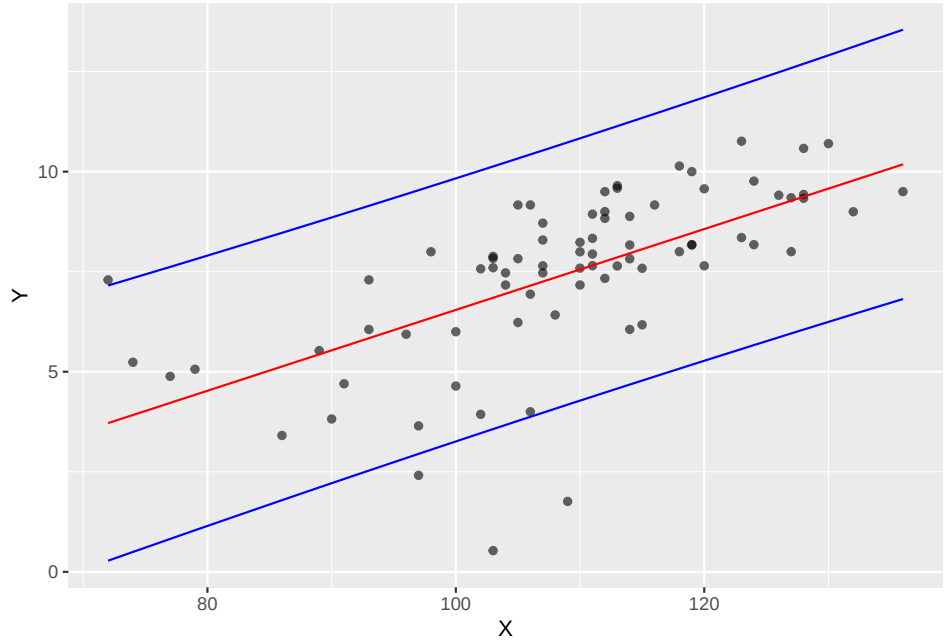
$$\widehat{GPA}_h \pm t^*(0.9, 76)s(\widehat{GPA}_h), \quad \text{where } s(\widehat{GPA}_h) = s^2\left(1 + \frac{1}{78} + \frac{(IQ_h - \overline{IQ})^2}{\sum_{i=1}^{78}(IQ_i - \overline{IQ})^2}\right).$$

```
predict(model, data.frame(iq = c(100)), interval='predict', level = 0.9)
```

```
##        fit     lwr      upr
## 1 6.545114 3.79753 9.292698
```

c) *Draw a band for 95% prediction intervals (i.e. join the limits of the prediction intervals with the smooth line). How many observations fall outside this band?*

```
dat_plot = data.frame(X, Y)
pred = predict(model, interval = "prediction")
dat_plot = cbind(dat_plot, pred)
library(ggplot2)
ggplot(data = dat_plot, aes(x=X)) +
  geom_point(aes(y = Y), alpha = 0.6) +
  geom_line(aes(y = fit), color = "red") +
  geom_line(aes(y = lwr), color = "blue") +
  geom_line(aes(y = upr), color = "blue")
```



There are 4 observations that fall outside the prediction bounds.

4 Exercise 4

a) Our theoretical model is:

$$GPA_i = \beta_0 + \beta_1 Piers - Harris_i + \epsilon_i, \quad i = 1, ..., 78.$$

The formulas for the β_0, β_1 and σ^2 estimators are similar to the ones given in the previous exercise, the only difference being that IQ is swapped for $Piers - Harris$.

```
model2 = lm(gpa~piers_harris, data = dat)
model2$coefficients
```

```
##  (Intercept) piers_harris
##    2.2258827    0.0916523
```

The regression equation is

$$E(GPA) = 2.2258827 + 0.0916523 * Piers - Harris.$$

The R^2 is

```
summary(model2)$r.squared
```

```
## [1] 0.2935829
```

b)

We will use the same procedure as in the previous exercise point a).

```
summary(model2)[["coefficients"]][, "t value"][2]
```

```
## piers_harris
##     5.620068
```

The p-value

```
summary(model2)[["coefficients"]][, "Pr(>|t|)"][2]
```

```
## piers_harris
## 3.006416e-07
```

The p-value is less than 0.05, therefore for this significance level we reject the null hypothesis and conclude that gpa depends on the results of Piers-Harris test.

c)

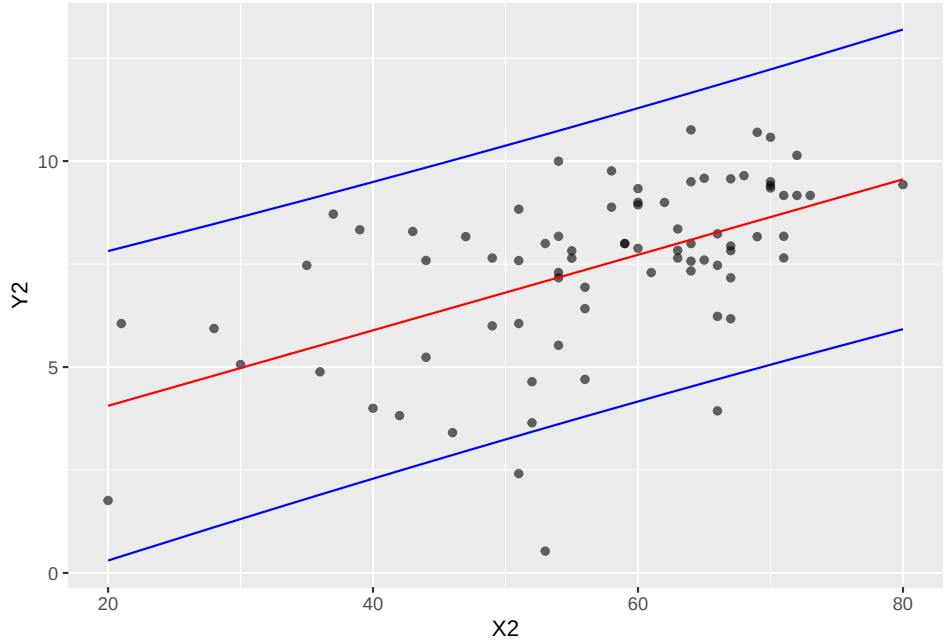
```
predict(model2, data.frame(piers_harris = c(60)), interval='predict', level = 0.9)
```

```
##       fit      lwr      upr
## 1 7.72502 4.747302 10.70274
```

d) *Draw a band for 95% prediction intervals. How many observations fall outside this band?*

```
Y2 = dat$gpa
X2 = dat$piers_harris
dat_plot = data.frame(X2, Y2)
pred = predict(model2, interval = "prediction")
dat_plot = cbind(dat_plot, pred)
#dat_plot = dat_plot[order(dat_plot$X2),]

ggplot(data = dat_plot, aes(x=X2)) +
  geom_point(aes(y = Y2), alpha = 0.6) +
  geom_line(aes(y = fit), color = "red") +
  geom_line(aes(y = lwr), color = "blue") +
  geom_line(aes(y = upr), color = "blue")
```



We see that 3 observations fall outside of the prediction bounds.

e) *Which of the two variables : result of iq test or result of Piers-Harris test, is a better predictor of gpa?*

Since both our models have the same number of predictor variables (1) we can use R^2 to see which model better fits the data.

```
c(summary(model)$r.squared, summary(model2)$r.squared)
```

```
## [1] 0.4016146 0.2935829
```

We see that the value of R^2 is bigger for the first model $GPA \sim IQ$, therefore we can say that the results of iq test better predict gpa.

5 Exercise 5

For the next two problems we will use the copier maintenance data. Second column contains the number of copiers and the first column contains the time (in hours) needed to maintain these copiers.
Let's first remind the assumptions of a linear model:

1. The relationship between the response variable and the explanatory variable is approximately linear,
2. The random errors are independent,
3. The random errors have equal variance,
4. The random errors come from a normal distribution $N(0, \sigma^2)$.

We investigate the properties of the random errors using the residuals.

a) *Verify that the sum of the residuals is zero.*

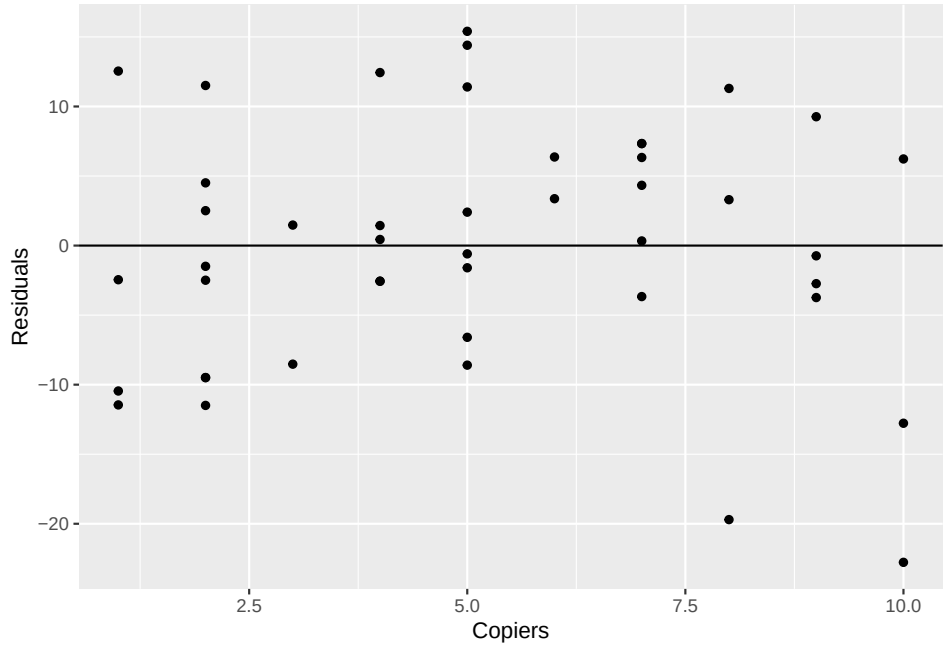
```
dat2 = read.table("ch01pr20.txt", col.names = c("hours", "copiers"))
model3 = lm(hours~copiers, data = dat2)

round(sum(model3$residuals), 10)
```

```
## [1] 0
```

b) *Plot the residuals versus the explanatory variable and briefly describe the plot noting any unusual patterns or points.*

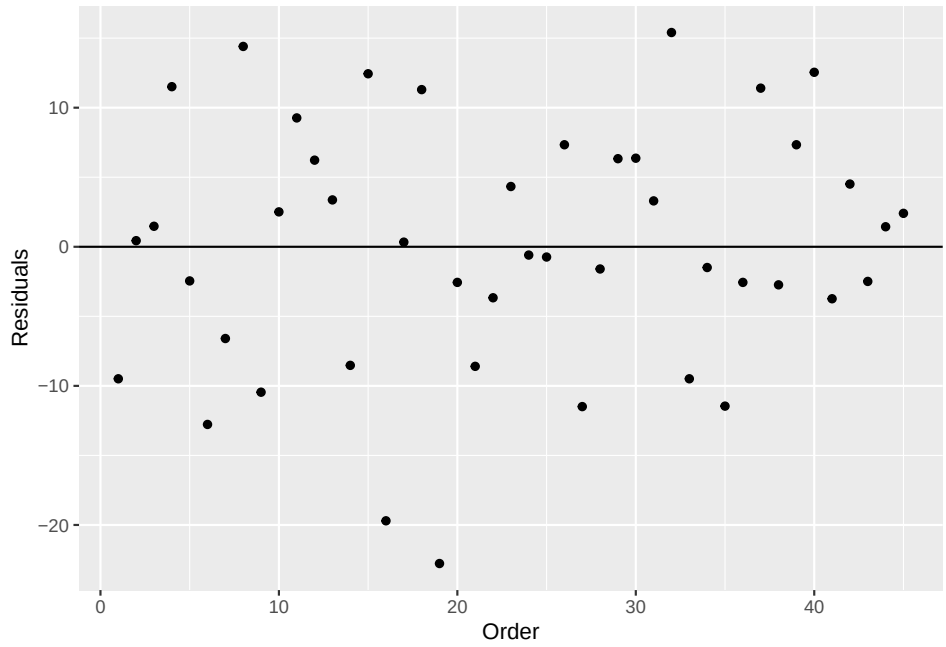
```
plot_dat_ex5 = data.frame(dat2$copiers, model3$residuals)
ggplot(data = plot_dat_ex5, aes(x = dat2.copiers, y= model3.residuals)) +
  geom_point() +
  geom_hline(yintercept = 0) +
  labs(x="Copiers", y="Residuals")
```



There seems to be no visible pattern in the plot, therefore we have no reason to think that the assumption about the constant variance of the errors is violated. The plot indicates the presence of potential outliers. As we can see, most of the residuals are in the range of -15 and 15, however, there are two residuals significantly smaller than -15.

c) *Plot the residuals versus the order in which the data appear in the data file and briefly describe the plot noting any unusual patterns or points.*

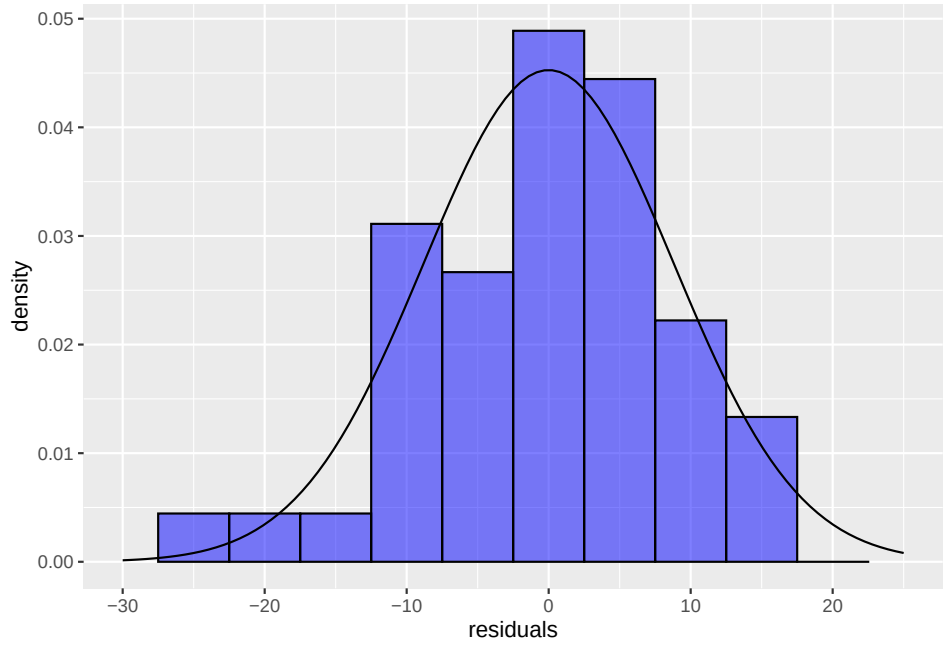
```
plot_dat_ex5_ord = data.frame(1:45, model3$residuals)
colnames(plot_dat_ex5_ord) = c("ord", "res")
ggplot(data = plot_dat_ex5_ord, aes(x = ord, y= res)) +
  geom_point() +
  geom_hline(yintercept = 0) +
  labs(x="Order", y="Residuals")
```



The residual versus order plot does not show any evidence that the error terms are correlated over order in which the measurements were taken.

d) *Examine the distribution of the residuals by getting a histogram and a normal probability plot. What do you conclude?*

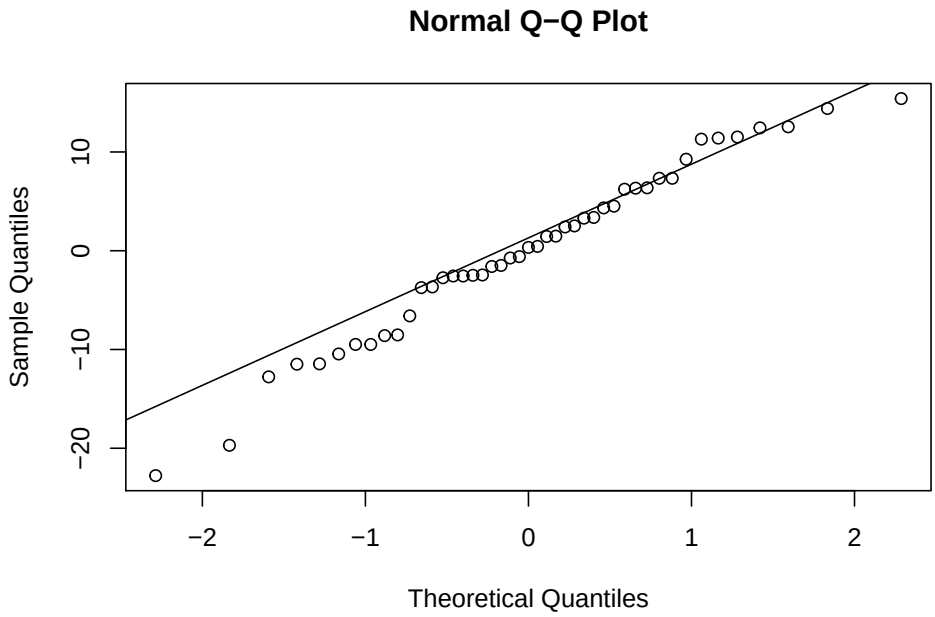
```
res_df = data.frame(model3$residuals)
colnames(res_df) = c("residuals")
ggplot(data = res_df, aes(x = residuals)) +
  geom_histogram(aes(y = ..density..), binwidth = 5, fill = "blue", color = "black", alpha=0.5) +
  stat_function(fun = dnorm, args = list(mean = mean(res_df$residuals), sd = sd(res_df$residuals))) +
  xlim(-30, 25)
```



The outliers we have detected earlier cause the distribution of the residuals to have a heavy left tail. Other

than that the shape of our histogram seems as if it comes from the normal distribution. We can also take a look at the qqplot.

```
qqnorm(model3$residuals)
qqline(model3$residuals)
```



Again the plot indicates a left tail in the distribution. The problem could probably be fixed with a larger sample size.

6 Exercise 6

Change the data set by changing the value of service time for the first observation from 20 to 2000.

a) *Run the regression with changed data and make a table comparing the results of this analysis with the results of the analysis of the original data. Include in the table the following: fitted equation, t-test for the slope with P-value, R^2, and the estimate of σ^2. Briefly summarize the differences.*

```
dat2_mod = dat2
dat2_mod$hours[1] = 2000

model3_mod = lm(hours~copiers, data = dat2_mod)
equations = sprintf("E(Time) = %.3f + %.3f*Copiers",
                    c(model3$coefficients[1], model3_mod$coefficients[1]),
                    c(model3$coefficients[2], model3_mod$coefficients[2]))

comp_df = rbind(equations,
                round(c(summary(model3)[["coefficients"]][, "t value"][2],
                        summary(model3_mod)[["coefficients"]][, "t value"][2]), digits = 3),
                round(c(summary(model3)[["coefficients"]][, "Pr(>|t|)"][2],
                        summary(model3_mod)[["coefficients"]][, "Pr(>|t|)"][2]), digits = 3),
                round(c(summary(model3)$r.squared, summary(model3_mod)$r.squared), digits = 3),
                round(c(sigma(model3)^2, sigma(model3_mod)^2), digits = 3))

comp_df = data.frame(comp_df)
```

```
colnames(comp_df) = c("Original model", "Modified model")
rownames(comp_df) = c("Regression equations", "t value", "Pr(>|t|)", "$R^2$", "$\\sigma^2$")
```

Table 3: Comparison of models

	Original model	Modified model
Regression equations	E(Time) = -0.580 + 15.035*Copiers	E(Time) = 135.900 + -3.059*Copiers
t value	31.123	-0.193
Pr(>|t|)	0	0.848
R^2	0.957	0.001
σ^2	79.451	85759.433

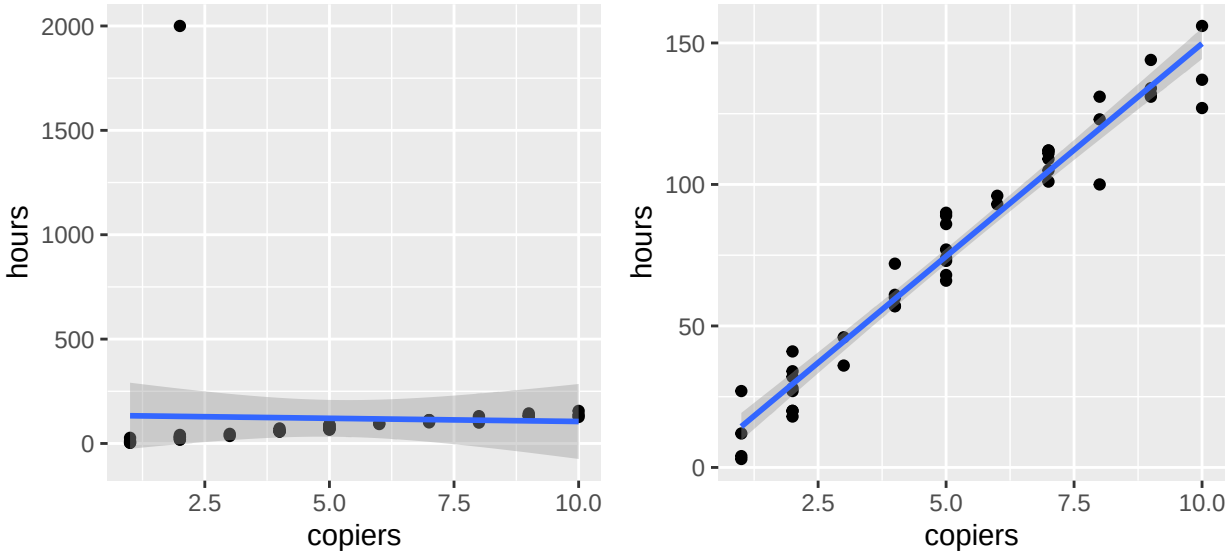
We see from the table that adding one outlier observation significantly change value of all the regression estimators $\hat{\beta}_0$, $\hat{\beta}_1$ and s^2. The slope and intercept in the modified model no longer capture the general structure of our data set as the regression line was pulled towards one outlier. The s^2 has gotten very large which is to be expected since when calculating it we sum squared residuals, and the residual value of the outlier observation is large. The R^2 for the modified model is very close to zero, which shows that the outlier has a strong impact on it.

Plots of regression lines for the original and the modified problem, together with a 95% confidence interval.

```
p1 = ggplot(dat2_mod, aes(x=copiers, y=hours))+
    geom_point()+
    geom_smooth(method=lm, se=TRUE)

p2 = ggplot(dat2, aes(x=copiers, y=hours))+
    geom_point()+
    geom_smooth(method=lm, se=TRUE)
```

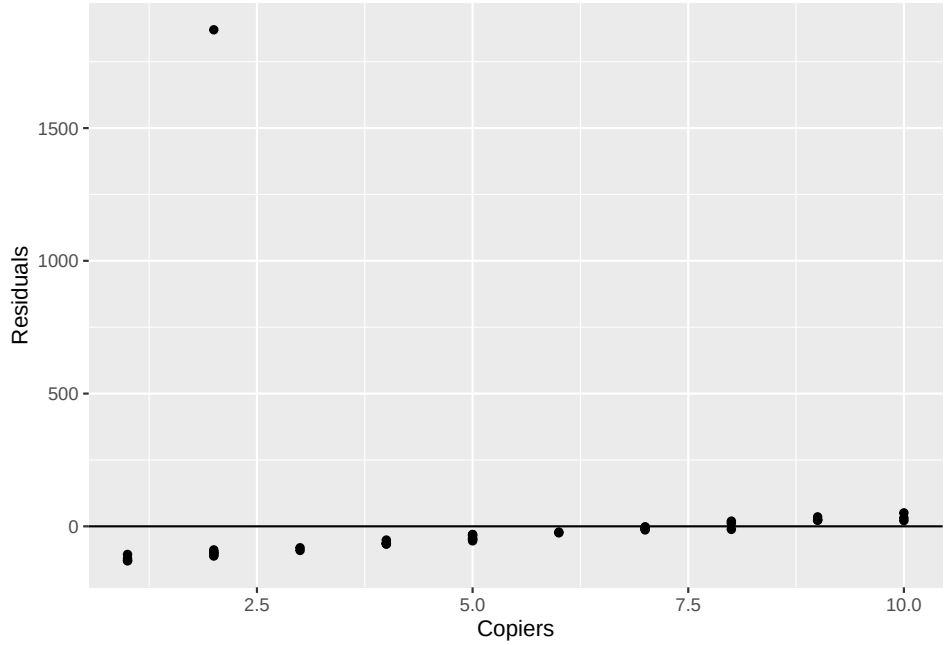
```
cowplot::plot_grid(p1, p2)
```



b) *Repeat points (b), (c) and (d) from problem 5 above on the modified data set and show the unusual observation on each of these plots.*

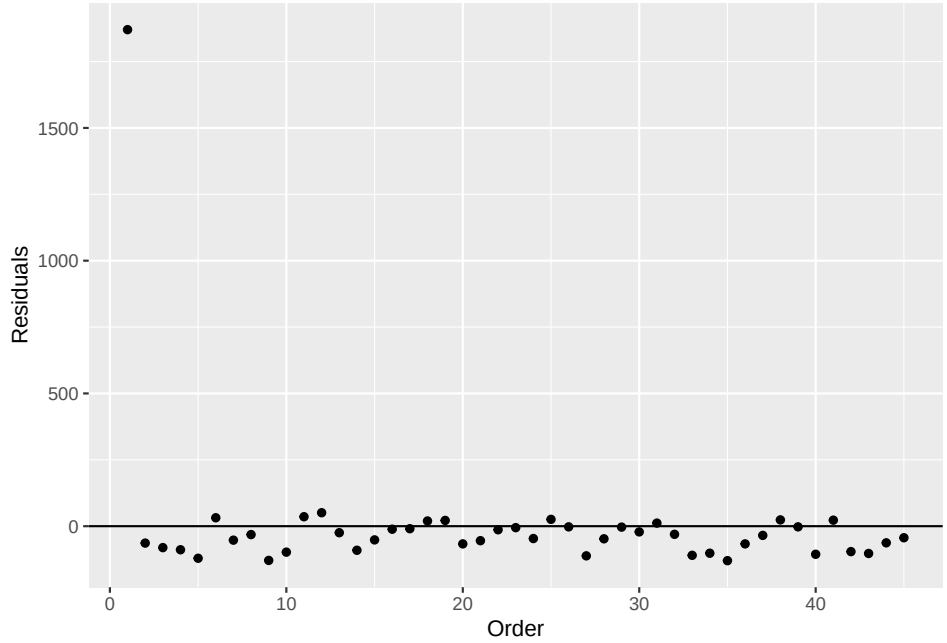
```
plot_dat_ex5 = data.frame(dat2_mod$copiers, model3_mod$residuals)
ggplot(data = plot_dat_ex5, aes(x = dat2_mod.copiers, y= model3_mod.residuals)) +
```

```
  geom_point() +
  geom_hline(yintercept = 0) +
  labs(x="Copiers", y="Residuals")
```



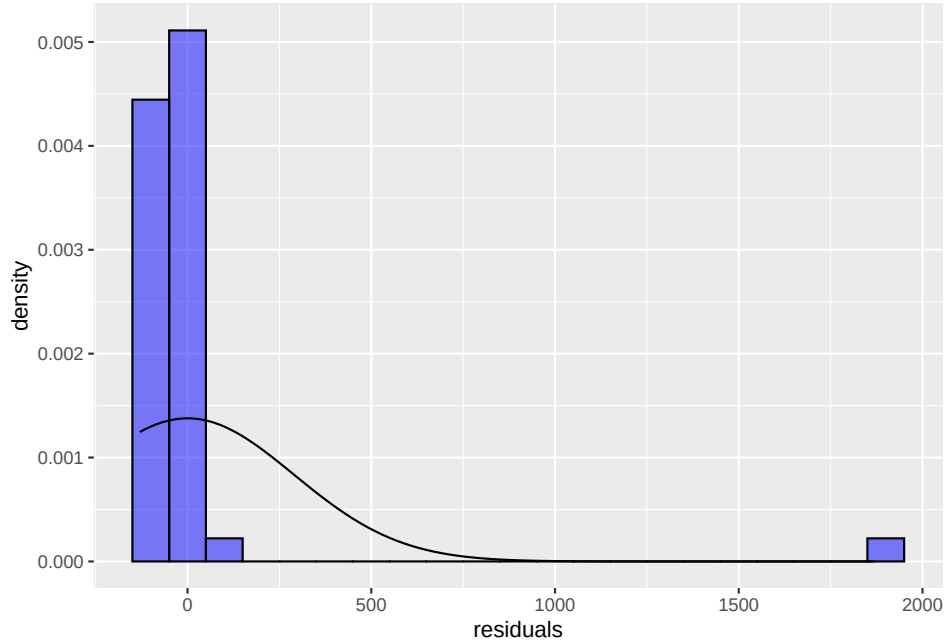
If the assumptions of a linear model were satisfied we shouldn't see any structure in the plot and the points should be randomly scattered around zero, which obviously is not happening here.

```
plot_dat_ex5_ord = data.frame(1:45, model3_mod$residuals)
colnames(plot_dat_ex5_ord) = c("ord", "res")
ggplot(data = plot_dat_ex5_ord, aes(x = ord, y= res)) +
  geom_point() +
  geom_hline(yintercept = 0) +
  labs(x="Order", y="Residuals")
```



There is a pattern in the plot suggesting that the errors are not independent.

```
res_df = data.frame(model3_mod$residuals)
colnames(res_df) = c("residuals")
ggplot(data = res_df, aes(x = residuals)) +
  geom_histogram(aes(y = ..density..), binwidth = 100, fill = "blue", color = "black", alpha=0.5) +
  stat_function(fun = dnorm, args = list(mean = mean(res_df$residuals), sd = sd(res_df$residuals)))
```



The histogram is a clear indication that the errors do not come from the normal distribution.

7 Exercise 7

For next six problems we will use the solution concentration data ch03pr15.txt. The first column gives the values of the solution concentration and the second column gives the time.

Run the linear regression with time as the explanatory variable and the solution concentration as the response variable. Summarize the regression results by giving the fitted regression equation, the value of R^2 and the results of the significance test for the null hypothesis that the solution concentration does not depend on time (formulate the statistical model, give null and alternative hypotheses in terms of the model parameters, test statistic with degrees of freedom, P-value, and brief conclusion in words).

```
dat3 = read.table("ch03pr15.txt", col.names = c("concentration", "time"))
X = dat3$time
Y = dat3$concentration
length(X)
```

```
## [1] 15
```

Let C be the concentration of the solution and T the time. Then the theoretical model is

$$C_i = \beta_0 + \beta_1 T_i + \epsilon_i, \quad i = 1, ..., 15, \quad \epsilon_i \sim N(0, \sigma^2).$$

```
model4 = lm(concentration~time, data = dat3)
model4$coefficients
```

```
## (Intercept)        time
##    2.575333   -0.324000
```

The regression equation is

$$E(C) = 2.575333 - 0.324 * T$$

Now we want to test whether the solution concentration depends on time, namely

$$H_0 : \beta_1 = 0 \quad \text{against} \quad H_1 : \beta_1 \neq 0.$$

The test statistic is

$$T = \frac{\hat{\beta}_1}{s(\hat{\beta}_1)} \sim t(13), \quad s^2(\hat{\beta}_1) = \frac{s^2}{\sum_{i=1}^{15}(T_i - \overline{T})}, \quad s^2 = \frac{1}{13}\sum_{i=1}^{15}(C_i - \hat{C}_i)^2.$$

```
summary(model4)[["coefficients"]][, "t value"][2]
```

```
##      time
## -7.482903
```

The p-value:

```
summary(model4)[["coefficients"]][, "Pr(>|t|)"][2]
```

```
##         time
## 4.611199e-06
```

Finally the R^2

```
summary(model4)$r.squared
```

```
## [1] 0.8115774
```

From the analysis above the obtained model seems reasonable, we reject the null hypothesis and conclude that the concentration depends on time. The value of R^2 is large, suggesting that the model fits reasonably well to the data.

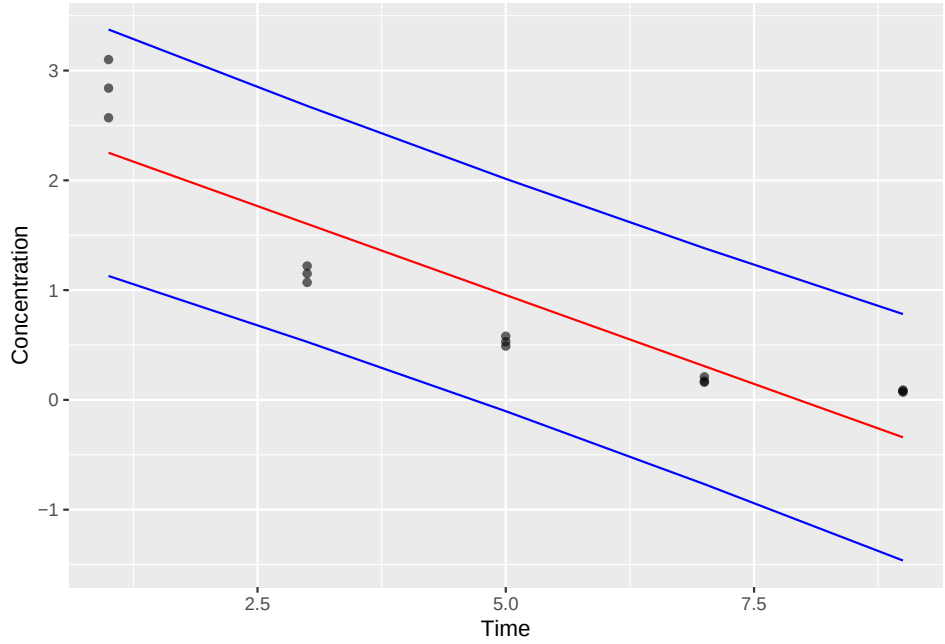
8 Exercise 8

Plot the solution concentration versus time. Add a fitted regression line and a band for 95% prediction intervals. What do you conclude? Calculate the correlation coefficient between the observed and predicted value of the solution concentration.

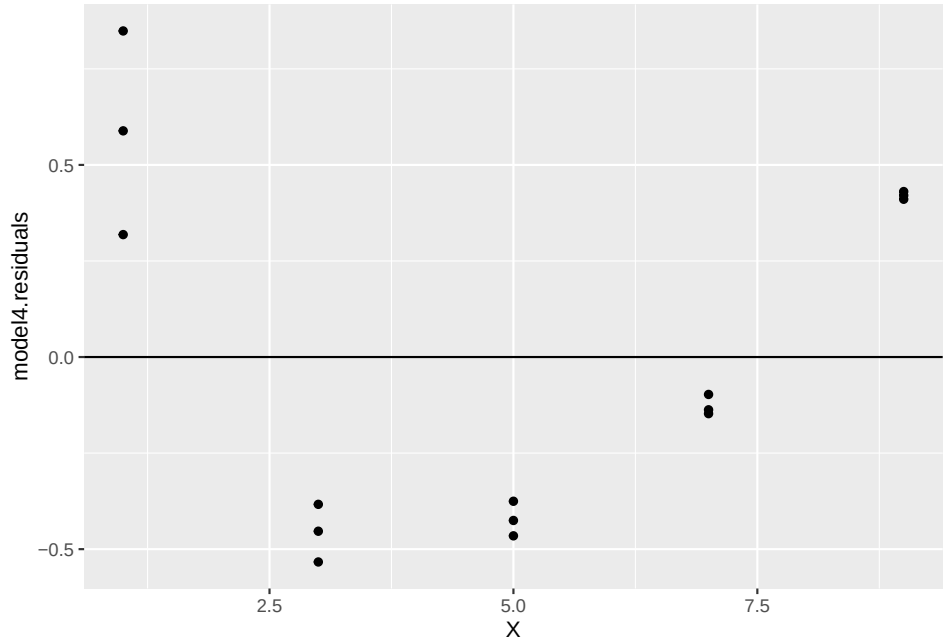
```
X = dat3$time
dat_plot = data.frame(X, Y)
pred = predict(model4, interval = "prediction")
dat_plot = cbind(dat_plot, pred)
#dat_plot = dat_plot[order(dat_plot$X2),]

p_ex8 = ggplot(data = dat_plot, aes(x=X)) +
  geom_point(aes(y = Y), alpha = 0.6) +
  geom_line(aes(y = fit), color = "red") +
  geom_line(aes(y = lwr), color = "blue") +
  geom_line(aes(y = upr), color = "blue") +
  labs(x="Time", y="Concentration")

p_ex8
```




```
ggplot(data.frame(X, model4$residuals), aes(x = X, y = model4.residuals)) +
  geom_point() +
  geom_hline(yintercept = 0)
```



From the plots we see that the conditions of linear regression seem to be violated. There is a clear non linear pattern in the data, which is further confirmed by the residuals versus explanatory variable plot.
Correlation between the predicted and observed values:

```
# correlation between the predicted and observed values = sqrt(R^2)
sqrt(summary(model4)$r.squared)
```

```
## [1] 0.9008759
```

We can also calculate it with `cor`:

```r
cor(matrix(c(dat3$concentration, predict(model4)), nrow = 15))[1,2]
```

```
## [1] 0.9008759
```

9 Exercise 9

Let Y be a response variable. For each real number λ the Box-Cox transformation is

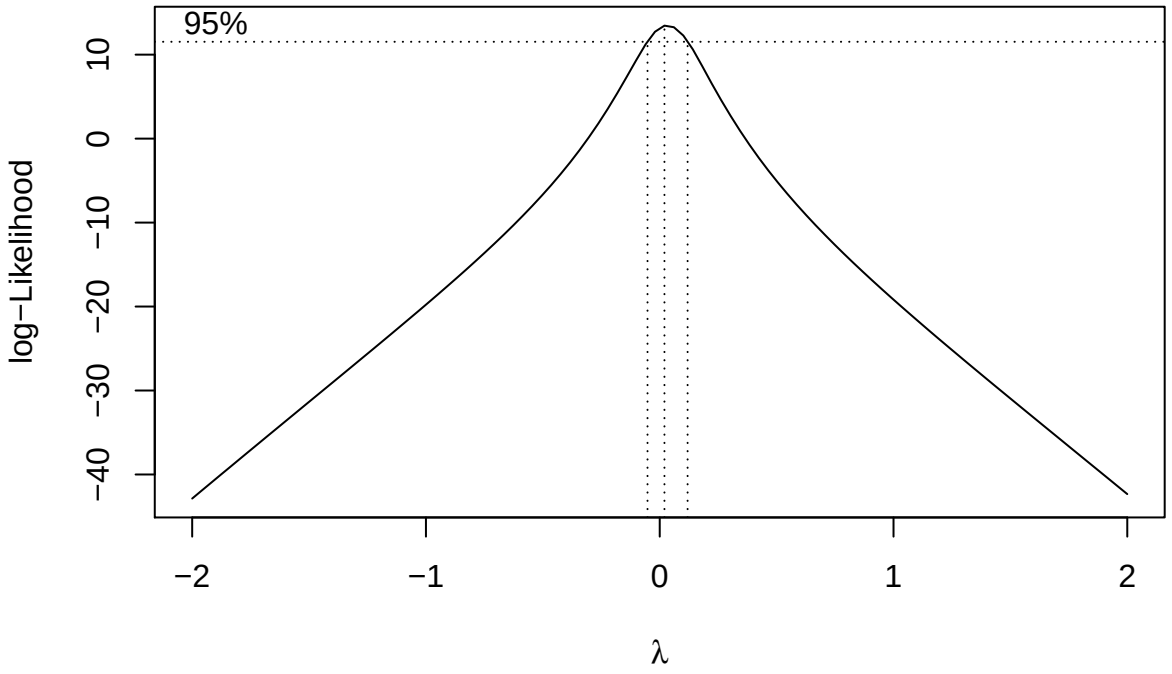
$$f_\lambda(Y) = \begin{cases} (Y^\lambda - 1)/\lambda & \text{if } \lambda \neq 0 \\ \log Y & \text{if } \lambda = 0. \end{cases}$$

The Box–Cox procedure selects a value λ so that after transformation, we obtain the following linear model:

$$f_\lambda(Y) = \beta_0 + \beta_1 X_i + \epsilon_i.$$

The Box-Cox procedure uses the method of maximum likelihood to estimate λ, as well as the other parameters β_0, β_1 and σ^2. We use this procedure if we want to correct skewness of the distributions of error terms, unequal error variances, and non linearity of the regression function.

```r
bc = MASS::boxcox(dat3$concentration~dat3$time)
```




```r
lambda = bc$x[which.max(bc$y)]
lambda
```

```
## [1] 0.02020202
```

Since λ is close to zero an appropriate transformation seems to be $\tilde{Y} = log(Y)$.

10 Exercise 10

Construct a new response variable by taking the log of the solution concentration (define logy = log(Y)). Repeat points 7 and 8 of this homework with logy as the response variable (and time as the explanatory variable). Summarize your results.

The theoretical model is

$$log(C_i) = \widetilde{C} = \beta_0 + \beta_1 T_i + \epsilon_i, \quad i = 1, ..., 15, \quad \epsilon_i \sim N(0, \sigma^2).$$

```
model5 = lm(log(concentration)~time, data = dat3)
model5$coefficients
```

```
## (Intercept)        time
##   1.5079164  -0.4499258
```

The regression equation is

$$E(\widetilde{C}) = 1.5079164 - 0.4499258 * T$$

Now we want to test whether the solution concentration depends on time, namely

$$H_0 : \beta_1 = 0 \quad \text{against} \quad H_1 : \beta_1 \neq 0.$$

The test statistic is

$$T = \frac{\hat{\beta}_1}{s(\hat{\beta}_1)} \sim t(13), \quad s^2(\hat{\beta}_1) = \frac{s^2}{\sum_{i=1}^{15}(T_i - \overline{T})}, \quad s^2 = \frac{1}{13}\sum_{i=1}^{15}(\tilde{C}_i - \widehat{\tilde{C}_i})^2.$$

```
summary(model5)[["coefficients"]][, "t value"][2]
```

```
##      time
## -42.87453
```

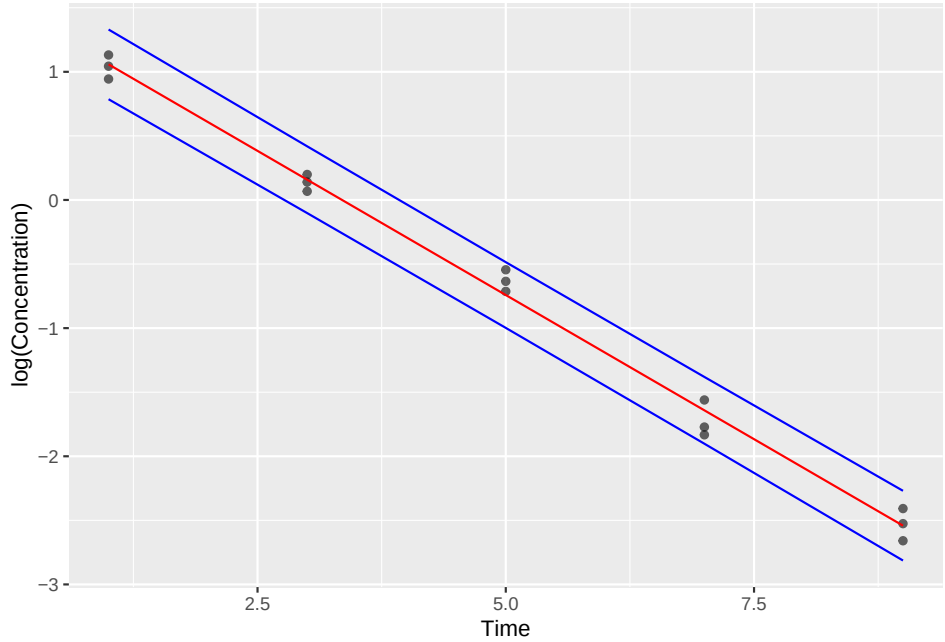
```
summary(model5)[["coefficients"]][,  "Pr(>|t|)"][2]
```

```
##         time
## 2.188252e-15
```

We reject the null hypothesis and conclude that the concentration depends on time.

```
X = dat3$time
Y2 = log(dat3$concentration)
dat_plot = data.frame(X, Y2)
pred = predict(model5, interval = "prediction")
dat_plot = cbind(dat_plot, pred)
#dat_plot = dat_plot[order(dat_plot$X2),]

ggplot(data = dat_plot, aes(x=X)) +
  geom_point(aes(y = Y2), alpha = 0.6) +
  geom_line(aes(y = fit), color = "red") +
  geom_line(aes(y = lwr), color = "blue") +
  geom_line(aes(y = upr), color = "blue") +
  labs(x="Time", y="log(Concentration)")
```



We see that the non linearity problem has been fixed.

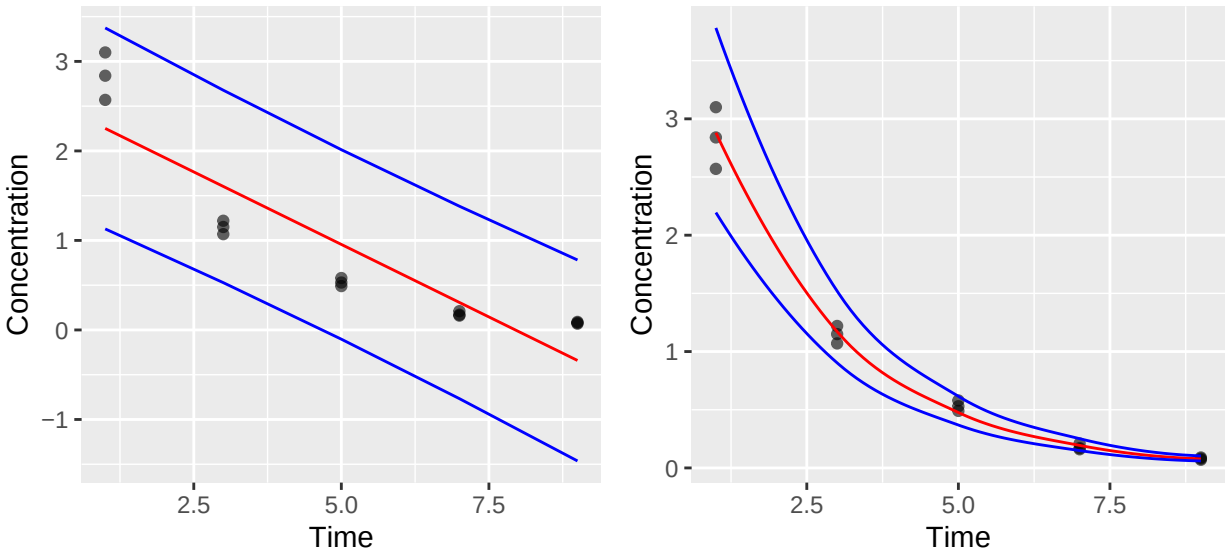
11 Exercise 11

Plot the solution concentration versus time. Add a regression curve and a band for 95% prediction intervals based on the results obtained in point 10. Compare to the graph obtained in point 8. Calculate the correlation coefficient between the observed solution concentration and the predictions based on the model from point 10.

```
dat_plot = data.frame(X, Y2)
pred = predict(model5, interval = "prediction")
dat_plot = cbind(dat_plot, pred)
#dat_plot = dat_plot[order(dat_plot$X2),]

p_ex11 = ggplot(data = dat_plot, aes(x=X)) +
  geom_point(aes(y = exp(Y2)), alpha = 0.6) +
  geom_smooth(aes(y = exp(fit)), color = "red", size = 0.5) +
  geom_smooth(aes(y = exp(lwr)), color = "blue", size = 0.5) +
  geom_smooth(aes(y = exp(upr)), color = "blue", size = 0.5)+
  labs(x="Time", y="Concentration")
```

```
cowplot::plot_grid(p_ex8, p_ex11)
```



The model with modified response variable better fits the data.

Correlation between the predicted and observed values:

```
sqrt(summary(model5)$r.squared)
```

```
## [1] 0.9964826
```

We can also calculate it with `cor`:

```
cor(matrix(c(log(dat3$concentration), predict(model5)), nrow = 15))[1,2]
```

```
## [1] 0.9964826
```

12 Exercise 12

Construct a new explanatory variable $t1 = time^{-1/2}$. Repeat points 10 and 11 of this exercise using the regression model with the solution concentration as the response variable and t1 as the explanatory variable. Summarize your results. Which model seems to be the best?

Let $T1 = T^{-1/2}$, then the theoretical model is

$$log(C_i) = \widetilde{C} = \beta_0 + \beta_1 T1_i + \epsilon_i, \quad i = 1, ..., 15, \quad \epsilon_i \sim N(0, \sigma^2).$$

```
T1 = dat3$time^(-1/2)
model6 = lm(log(dat3$concentration)~T1)
model6$coefficients
```

```
## (Intercept)          T1
##   -3.324813    4.720816
```

The regression equation is

$$E(\widetilde{C}) = -3.324813 - 4.720816 * T1$$

Now we want to test whether the solution concentration depends on time, namely

$$H_0 : \beta_1 = 0 \quad \text{against} \quad H_1 : \beta_1 \neq 0.$$

The test statistic is

$$T = \frac{\hat{\beta}_1}{s(\hat{\beta}_1)} \sim t(13), \quad s^2(\hat{\beta}_1) = \frac{s^2}{\sum_{i=1}^{15}(T1_i - \overline{T1})}, \quad s^2 = \frac{1}{13}\sum_{i=1}^{15}(\tilde{C}_i - \widehat{\tilde{C}_i})^2.$$

```
summary(model6)[["coefficients"]][, "t value"][2]
```

```
##       T1
## 7.065941
```

```
summary(model6)[["coefficients"]][,  "Pr(>|t|)"][2]
```

```
##           T1
## 8.467748e-06
```

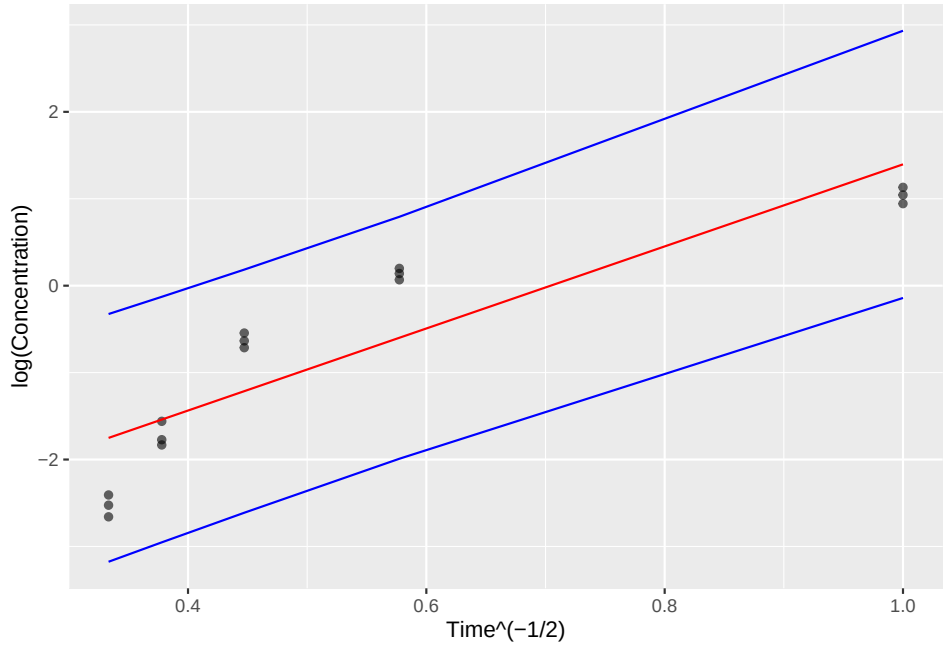
We reject the null hypothesis.

```
summary(model6)$r.squared
```

```
## [1] 0.7934131
```

```
X = T1
Y = log(dat3$concentration)
dat_plot = data.frame(X, Y)
pred = predict(model6, interval = "prediction")
dat_plot = cbind(dat_plot, pred)
#dat_plot = dat_plot[order(dat_plot$X2),]

ggplot(data = dat_plot, aes(x=X)) +
  geom_point(aes(y = Y), alpha = 0.6) +
  geom_line(aes(y = fit), color = "red") +
  geom_line(aes(y = lwr), color = "blue") +
  geom_line(aes(y = upr), color = "blue")+
  labs(x="Time^(-1/2)", y="log(Concentration)")
```



It seems that with substituting T with T1 with have reintroduced non linearity to our model.

Correlation between the predicted and observed values:

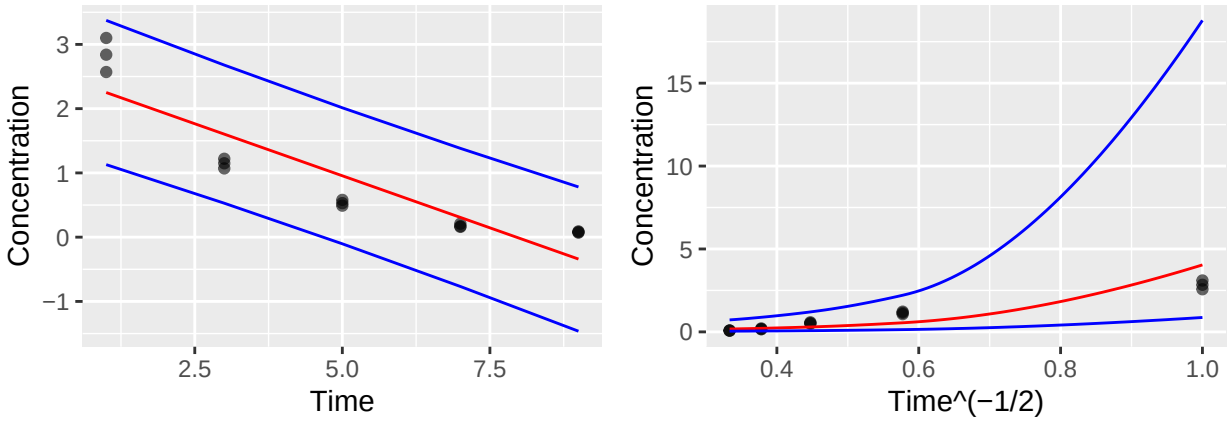
```
sqrt(summary(model6)$r.squared)
```

```
## [1] 0.8907374
```

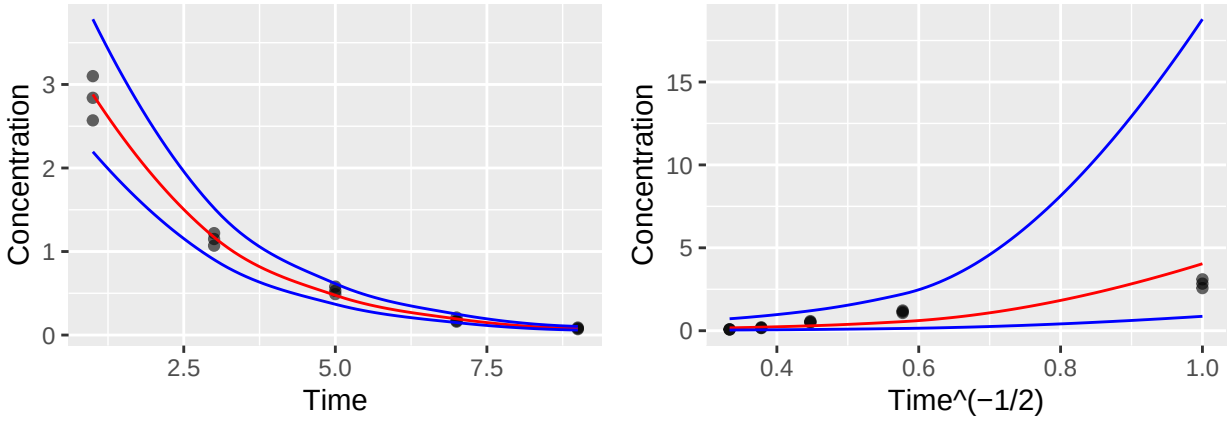
```
dat_plot = data.frame(X, Y)
pred = predict(model6, interval = "prediction")
dat_plot = cbind(dat_plot, pred)
#dat_plot = dat_plot[order(dat_plot$X2),]

p_ex12 = ggplot(data = dat_plot, aes(x=X)) +
  geom_point(aes(y = exp(Y)), alpha = 0.6) +
  geom_smooth(aes(y = exp(fit)), color = "red", size = 0.5) +
  geom_smooth(aes(y = exp(lwr)), color = "blue", size = 0.5) +
  geom_smooth(aes(y = exp(upr)), color = "blue", size = 0.5) +
  labs(x="Time^(-1/2)", y="Concentration")
```

```
cowplot::plot_grid(p_ex8, p_ex12)
```




```
cowplot::plot_grid(p_ex11, p_ex12)
```



We see that the curve doesn't fit the data as well the one in the previous exercise.

```
c(summary(model5)$r.squared, summary(model6)$r.squared)
```

```
## [1] 0.9929776 0.7934131
```

The best model seems to be the one constructed in exercise 11. It captures the non linearity of the data and compared to the model from exercise 12 has larger R^2.