STAT 210
Applied Statistics and Data Analysis
Problem list 8 - Solution
(Due on week 9)

Exercise 1

In this exercise we will use the data set `iris`.

(i) Extract the data corresponding to species `setosa` to a separate data frame. Plot the numerical variables for this set in a matrix of plots.

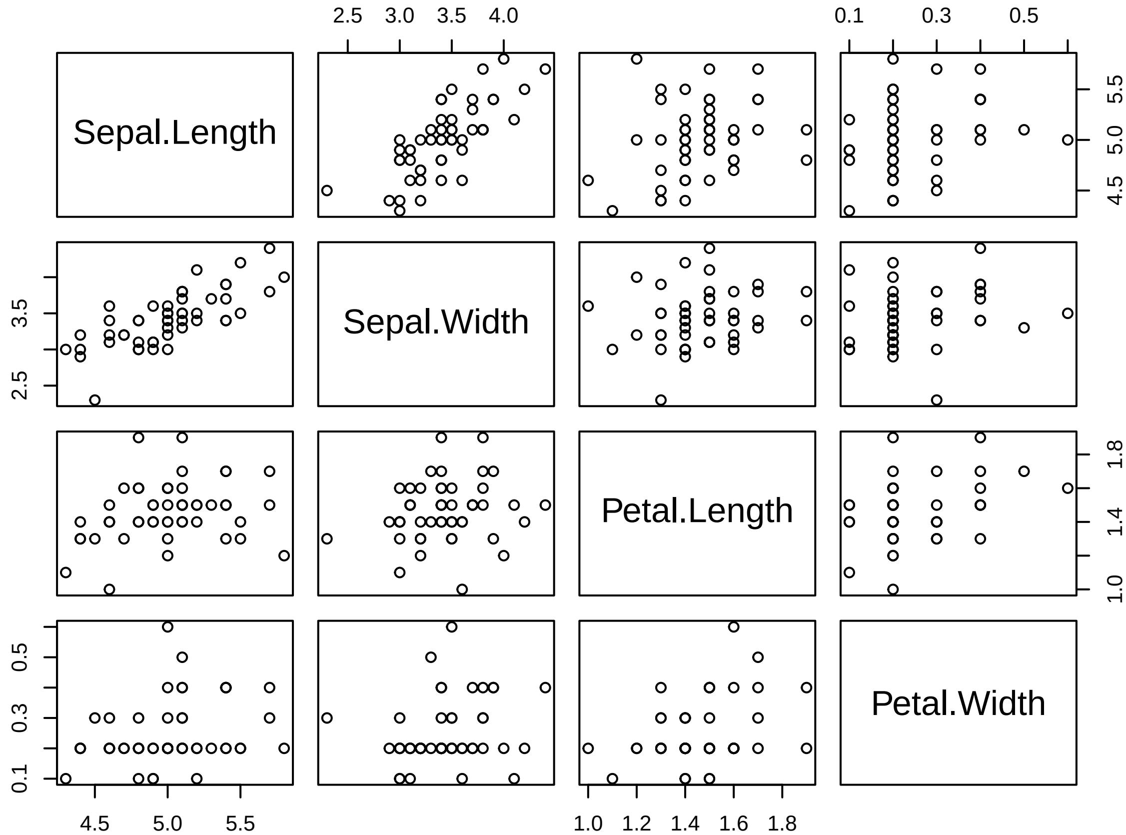
```
str(iris)
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```
iris.set <- subset(iris, Species == 'setosa')
str(iris.set)
```

```
## 'data.frame':    50 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```
plot(iris.set[,1:4])
```



Since we have extracted the data for the Setosa species, we have 50 data points in the new data set. However, some of the plots seem to have fewer points. For instance, the plot of `Petal.Length` against `Petal.Width` shows fewer data points. The reason for this is that the precision of the measurements limits the possible values and produces overlapping points. In particular, there are only six different valuesfor `Petal.Width` for this particular species.

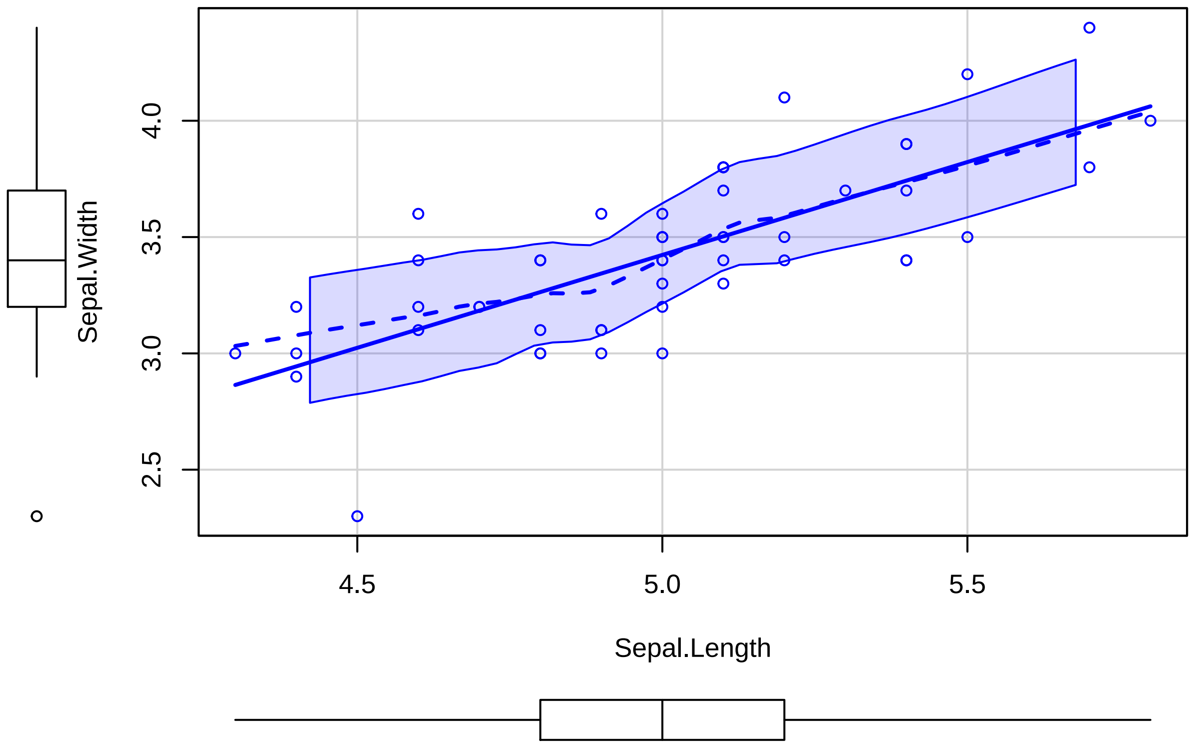
This fact also makes the plots involving this variable harder to interpret. For instance, the `Sepal.Length` against `Petal.Width` plot shows an increasing trend, but this is not as clear as what we see in the `Sepal.Length` against `Sepal.Width` plot.

(ii) Use the function `scatterplot` from the `car` package to plot `Sepal.Width` as a function of `Sepal.Length`. Comment on the graph.

```
library(car)
```

```
## Loading required package: carData
```

```
scatterplot(Sepal.Width ~ Sepal.Length, data = iris.set)
```



The graph shows the points together with the regression line (solid) and a local smoother (broken line). Th etwo are close and this indicates that the regression model is probably adequate. %>%

(iii) Fit a linear regression model for `Sepal.Width` as a function of `Sepal.Length`. Produce a table using `summary` and discuss the results.

```
modelA <- lm(Sepal.Width ~ Sepal.Length, data = iris.set)
summary(modelA)
```

```
##
## Call:
## lm(formula = Sepal.Width ~ Sepal.Length, data = iris.set)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.72394 -0.18273 -0.00306  0.15738  0.51709
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.5694     0.5217  -1.091    0.281
## Sepal.Length   0.7985     0.1040   7.681 6.71e-10 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2565 on 48 degrees of freedom
## Multiple R-squared:  0.5514, Adjusted R-squared:  0.542
## F-statistic: 58.99 on 1 and 48 DF,  p-value: 6.71e-10
```

The summary values for the residuals are reasonably symmetric. The t test for the slope is significant but for the intercept we cannot reject the null hypothesis that this coefficient is equal to zero. The estimated standard deviation for the errors (Residual standard error) is 0.2565.

(iv) Find the R^2 and verify that for simple linear regression, this coefficient is equal to the square of the correlation between the two variables.

The R^2 from the regression output is 0.5514.

```
with(iris.set, cor(Sepal.Width,Sepal.Length)^2)
```

```
## [1] 0.5513756
```

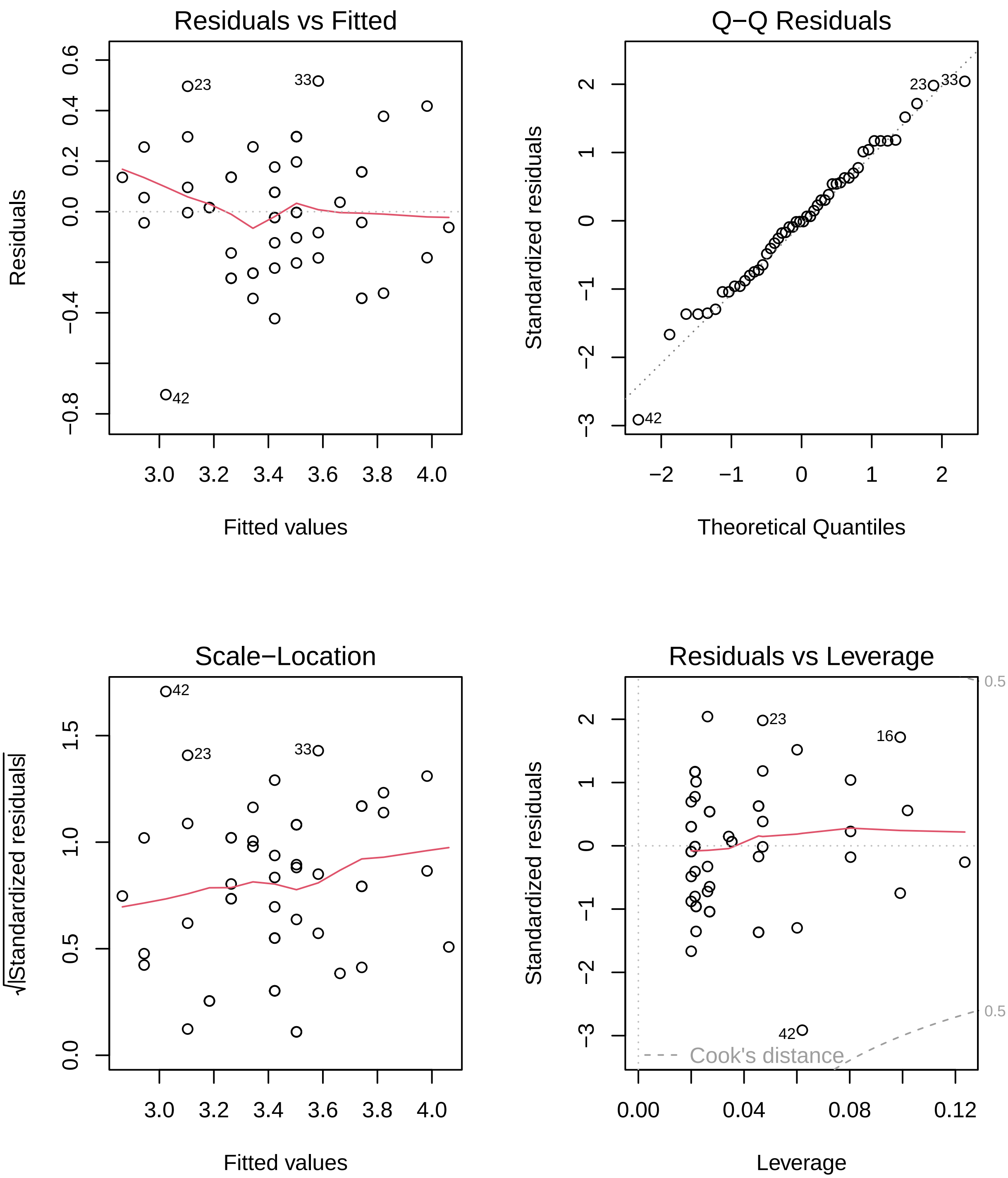
(v) Write down the equation for the regression line and interpret the parameters.

$$\texttt{Sepal.Width} = -0.5694 + 0.7985 * \texttt{Sepal.Length}$$

The slope represents the change in the

(vi) Do the diagnostic plots for this model and comment.

```
par(mfrow=c(2,2))
plot(modelA)
```




```r
par(mfrow=c(1,1))
```

(vii) In this case, the diagnostic plots give sufficient information about the normality assumption. However, if we wanted to test this assumption, we could use the Shapiro-Wilk test. Do this test and comment on the result.

```r
shapiro.test(rstandard(modelA))
```

```
##
##  Shapiro-Wilk normality test
```

```
##
## data:  rstandard(modelA)
## W = 0.98583, p-value = 0.8066
```

(viii) The assumption of uniform variance is not so clear from the plots, particularly from the Scale-Location graph. The test we used for analysis of variance does not work here, because we do not have grouped data. A test that can be used in this situation is the Score Test, proposed by Cook and Weisberg (1983) and described in Applied Linear Regression by S. Weisberg, Wiley. This test is available in the `car` package as `ncvTest`. Do this test and comment on the results.

```r
ncvTest(modelA)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.07602347, Df = 1, p = 0.78276
```

Exercise 2

For this question use the dataset PL825FQ2.

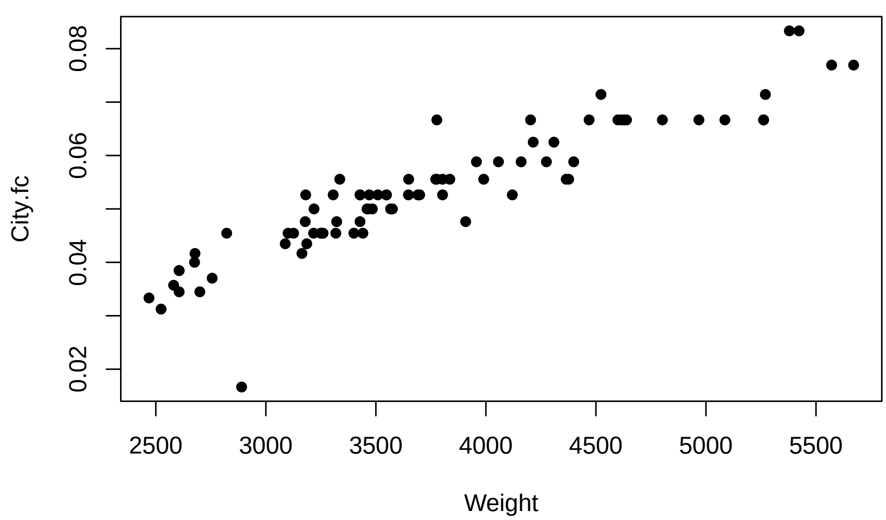
This dataset has information on fuel efficiency, measured in miles per gallon, and seven other variables for 80 different car models. There are two variables related to fuel efficiency, `City.Mpg` and `Highway.Mpg`. We will only consider `City.Mpg`, and we will work with the reciprocal of this variable, 1/`City.Mpg`, which we will call `City.fc` for fuel consumption. We want to explore the relation between `City.fc` and the car's weight (`Weight`).

(i) Read the data and define a new variable called `City.fc` in the data frame equal to the reciprocal of `City.Mpg`. Draw a scatterplot of `City.fc` as a function of `Weight`. Fit a simple linear regression for `City.fc` as a function of `Weight` and add the line to the plot. Comment. Obtain a summary of the regression and comment.

```
data1 <- read.table('PL825FQ2.txt', header = T)
str(data1)
```

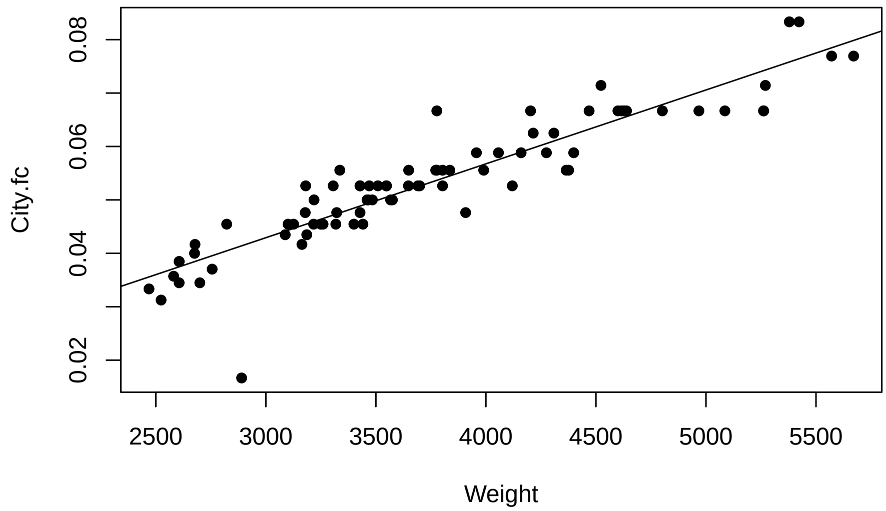
```
## 'data.frame':    81 obs. of  9 variables:
##  $ Model      : chr  "A4" "3_Series" "G35" "X-Type" ...
##  $ Eng.Size   : num  1.8 2.5 3.5 2.5 1.8 2.7 2.5 3 3 3.2 ...
##  $ Cylinders  : int  4 6 6 6 4 6 6 6 6 6 ...
##  $ MSRP       : int  25550 28100 28150 29330 29250 42650 39800 43730 38875 50575 ...
##  $ City.Mpg   : int  22 20 18 19 22 18 19 18 18 19 ...
##  $ Highway.Mpg: int  31 29 26 28 30 25 28 26 25 27 ...
##  $ Weight     : int  3252 3219 3336 3428 3250 3836 3428 3777 3649 3691 ...
##  $ Type       : chr  "Sedan" "Sedan" "Sedan" "Sedan" ...
##  $ Country    : chr  "Germany" "Germany" "Japan" "England" ...
```

```
data1$City.fc <- 1 / data1$City.Mpg
plot(City.fc ~ Weight, data = data1, pch = 16)
```



We fit a model with the function `lm` and add the regression line to the scatterplot:

```
model1 <- lm(City.fc ~ Weight, data = data1)
plot(City.fc ~ Weight, data = data1, pch = 16)
abline(model1)
```



The line seems to fit the data quite well. For the summary we write

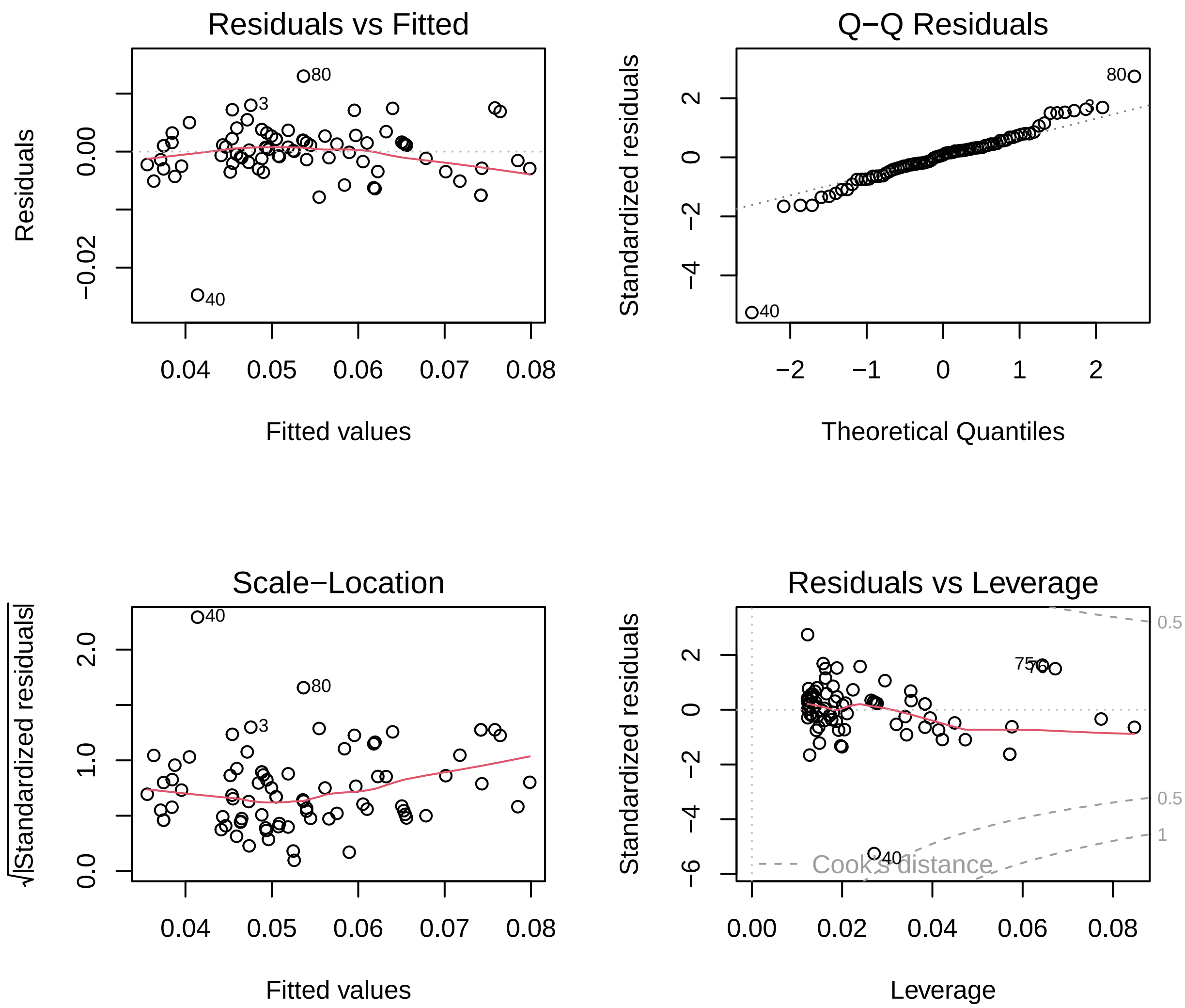
```
summary(model1)
```

```
##
## Call:
## lm(formula = City.fc ~ Weight, data = data1)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.024734 -0.002015  0.000386  0.002133  0.013001
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.439e-03  2.571e-03    0.56    0.577
## Weight      1.383e-05  6.700e-07   20.64   <2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.004771 on 79 degrees of freedom
## Multiple R-squared:  0.8435, Adjusted R-squared:  0.8416
## F-statistic: 425.9 on 1 and 79 DF,  p-value: < 2.2e-16
```

The estimated values for the intercept and slope are 1.44×10^{-3} and 1.38×10^{-5}, respectively. `Weight` has a very small p-value while the p-value for the intercept is big, which says that the intercept is not significantly different from zero..

(ii) Draw the diagnostic plots. Do you identify any point as an outlier? If you do, which point is this? Can you identify this point in the initial scatterplot? Can you find a reason why this point is different from the rest?

```
par(mfrow = c(2,2))
plot(model1)
```




```r
par(mfrow = c(1,1))
```

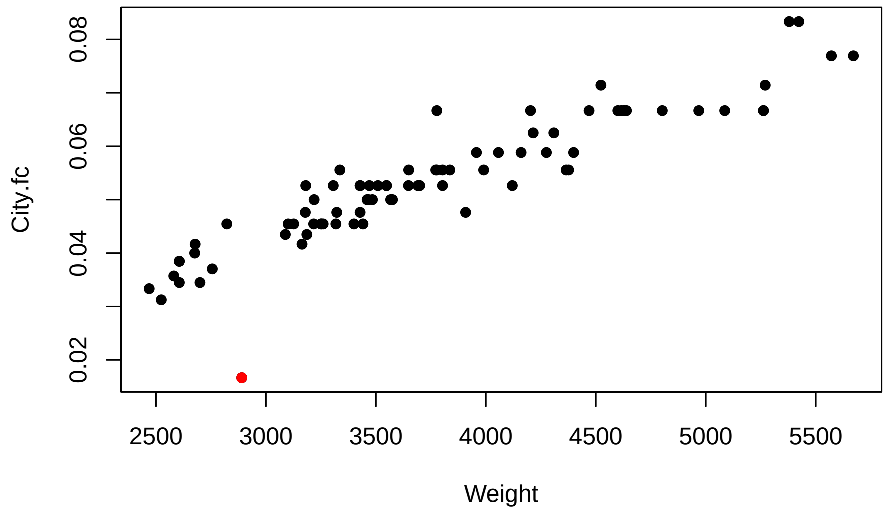
Point 40 is flagged in all the diagnostic plots. It is the point with biggest residual and in the quantile plot is very far from the rest of the points and the reference line. In the Scale-Location plot, the value corresponding to this point is bigger that 2. To identify the point in position 40 we write

```r
data1[40,]
```

```
##    Model Eng.Size Cylinders  MSRP City.Mpg Highway.Mpg
## 40 Prius      1.5         4 20295       60          51
##    Weight  Type Country    City.fc
## 40   2890 Sedan   Japan 0.01666667
```

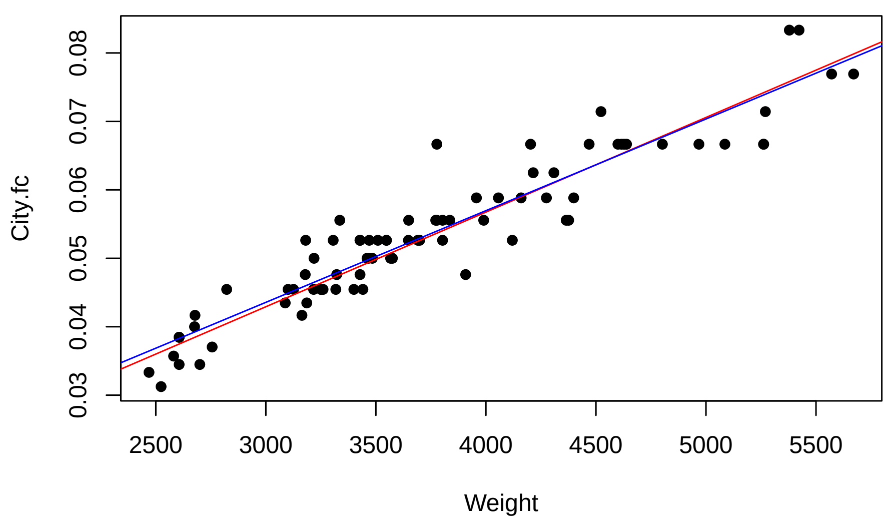
We see that the car model is Prius, a hybrid car, which uses a different system than the rest of the cars in this dataset. We repeat the scatterplot and plot this point in red

```r
plot(City.fc ~ Weight, data = data1, pch = 16)
points(City.fc ~ Weight, data = data1[40,], pch = 16, col = 'red')
```



(iii) Fit a new regression model excluding the outlier(s) you identified in the previous section. Draw a scatterplot with both regression lines. Compare the summary tables. Draw the diagnostic plots and comment.

```
data1N <- data1[-40,]
model2 <- lm(City.fc ~ Weight, data = data1N)
plot(City.fc ~ Weight, data = data1N, pch = 16)
abline(model1, col = 'red')
abline(model2, col = 'blue')
```



The two lines are very close, almost indistinguishable.

```
summary(model2)
```

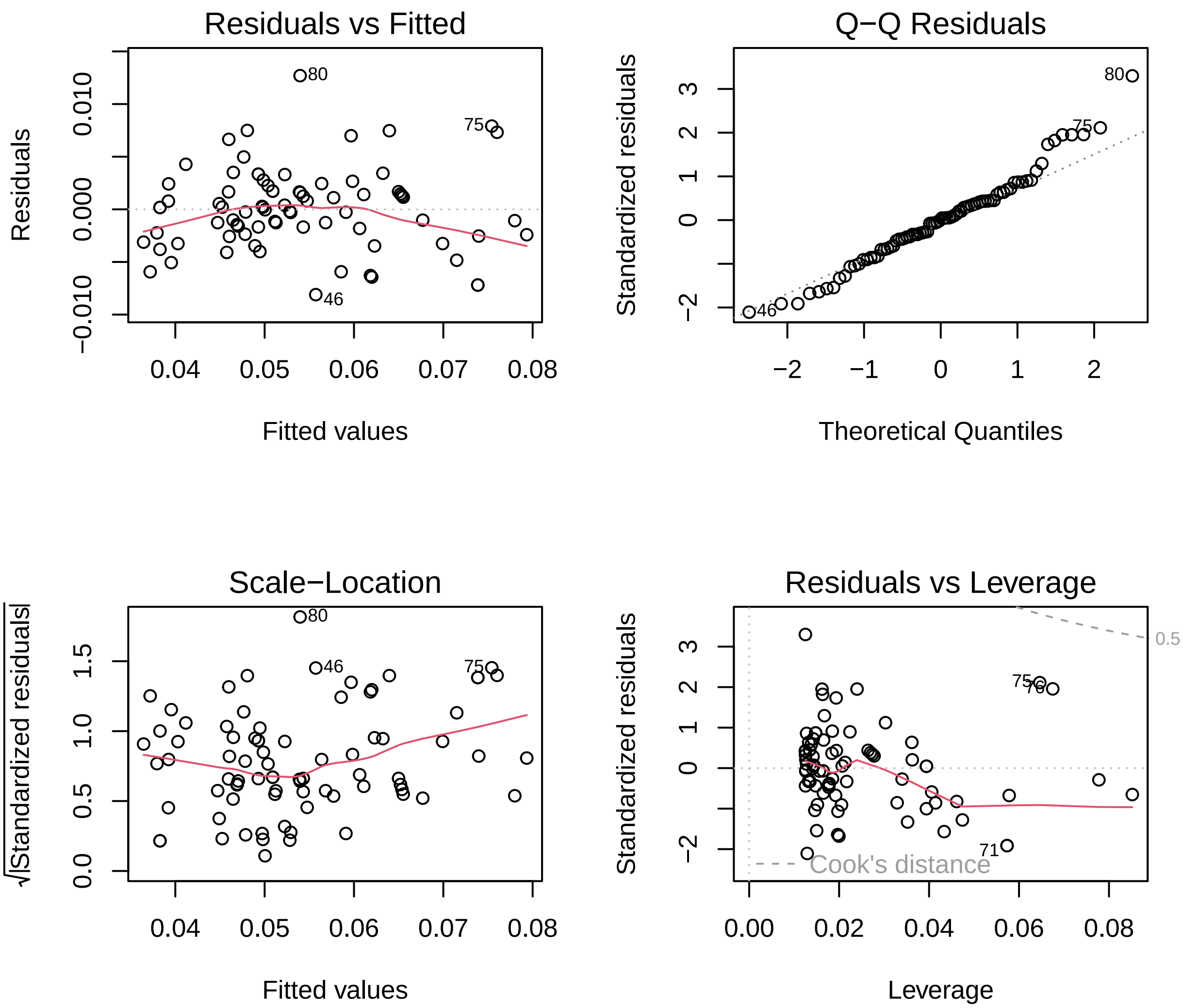
```
##
## Call:
## lm(formula = City.fc ~ Weight, data = data1N)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0081052 -0.0023791  0.0000656  0.0017292  0.0126971
##
## Coefficients:
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.380e-03  2.108e-03   1.604    0.113
## Weight      1.339e-05  5.479e-07  24.448   <2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.003872 on 78 degrees of freedom
## Multiple R-squared:  0.8846, Adjusted R-squared:  0.8831
## F-statistic: 597.7 on 1 and 78 DF,  p-value: < 2.2e-16
```

The summary tables show some differences but they are small. For instance, the intercept changes from 1.44×10^{-3} to 3.38×10^{-3} and the slope from 1.383×10^{-5} to 1.339×10^{-5}, which are small changes. The residual standard errors are 0.00477 and 0.00387, and the R^2 are 0.844 and 0.885.

We now plot the diagnostic graphs

```r
par(mfrow = c(2,2))
plot(model2)
```




```r
par(mfrow = c(1,1))
```

We see that the main difference with the previous model occurs in the quantile plot. The plot excluding point

40 looks better now.

(iv) Run the Shapiro-Wilk test on the residuals for both models and compare the results.

```
shapiro.test(residuals(model1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model1)
## W = 0.88433, p-value = 2.496e-06
```

```
shapiro.test(residuals(model2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model2)
## W = 0.97636, p-value = 0.1444
```

We see that when point 40 is included, the p-value is small and the null hypothesis of normality of the residuals is rejected. On the other hand, when this point is excluded, the p value is large and the null hypothesis is not rejected.

Summing up, point 40 is an outlier but not an influential point in the regression, since the regression equation is not substantially changed when the point is excluded. However, when the point is included, the assumption of normality for the residuals is not verified.

We also run a test for homogeneity of variance, the `ncvTest` in the `car` package:

```
library(car)
ncvTest(model1)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.5692541, Df = 1, p = 0.45056
```

```
ncvTest(model2)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 4.842164, Df = 1, p = 0.027772
```

Exercise 3

For this question use the data set PL825FQ3.

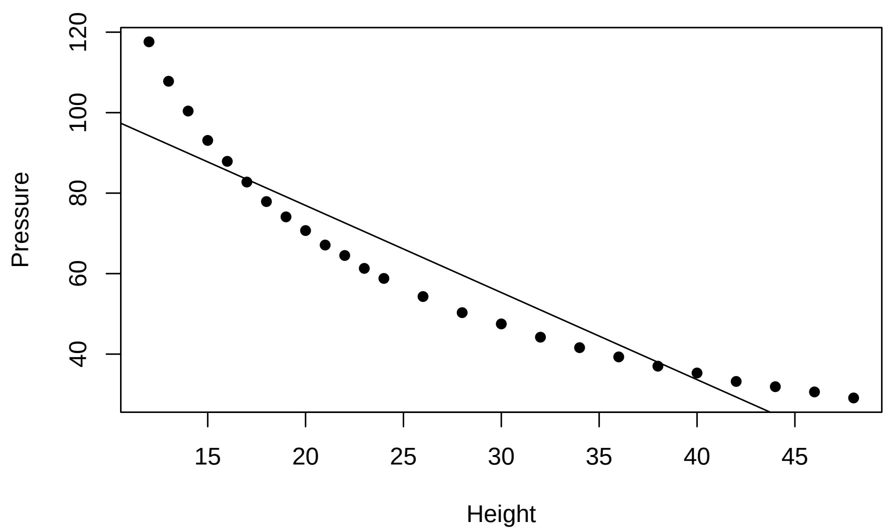
The data for this question come from an experiment to determine the relation between the volume of a gas and the pressure. The file has two variables, `Height` and `Pressure`. Height corresponds to the height of a cylindrical container with a fixed circular base and a movable top that allowed changing the volume of the container. `Height` is measured in inches. `Pressure` is measured in inches of mercury as in a barometer. We want to study the relation between these two variables.

(i) Read the data and plot `Pressure` as a function of `Height`. Fit a simple linear regression for `Pressure` as a function of `Height` and add the regression line to the plot. Comment. Obtain a summary for the regression and draw the diagnostic plots. Comment on the results

```
data2 <- read.table('PL825FQ3.txt', header = T)
str(data2)
```

```
## 'data.frame':    25 obs. of  2 variables:
##  $ Height  : int  48 46 44 42 40 38 36 34 32 30 ...
##  $ Pressure: num  29.1 30.6 31.9 33.2 35.3 37 39.3 41.6 44.2 47.5 ...
```

```
model3 <- lm(Pressure ~ Height, data = data2)
plot(Pressure ~ Height, data = data2, pch = 16)
abline(model3)
```



This is a clearly inadequate model, so we seek a transformation that leads to a better result. That is the purpose of the next section. The summary is

```
summary(model3)
```

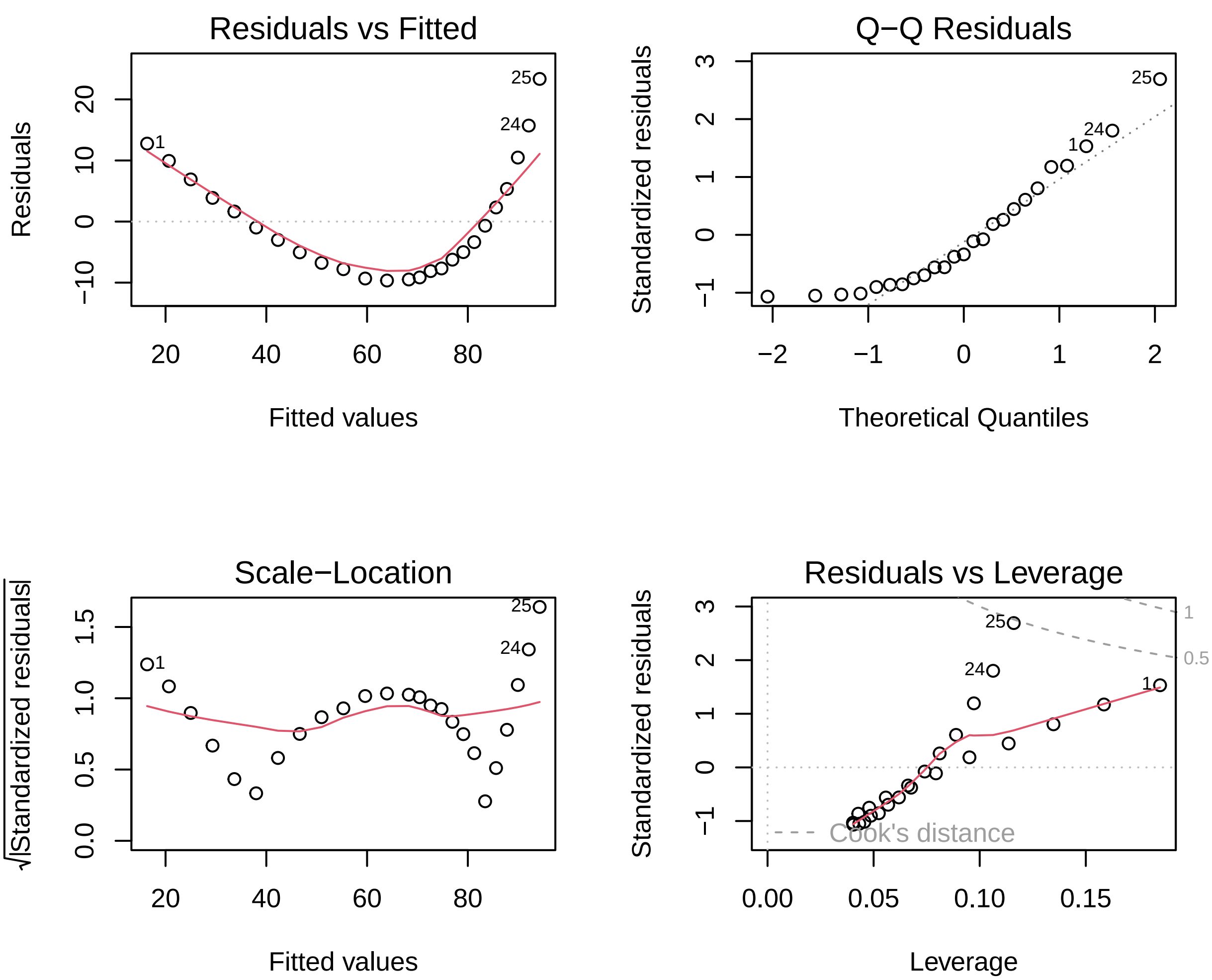
```
##
## Call:
## lm(formula = Pressure ~ Height, data = data2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.654 -7.675 -3.012  5.340 23.347
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 120.2233     4.9230   24.42  < 2e-16 ***
```

```
## Height       -2.1642     0.1683  -12.86 5.48e-12 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.228 on 23 degrees of freedom
## Multiple R-squared:  0.8779, Adjusted R-squared:  0.8726
## F-statistic: 165.4 on 1 and 23 DF,  p-value: 5.485e-12
```

Observe that the results in the summary do not reflect the fact that the model is not adequate. The p-values for the coefficients are both small, and the R^2 is almost 88%.

Let us look at the diagnostic plots

```r
par(mfrow = c(2,2))
plot(model3)
```

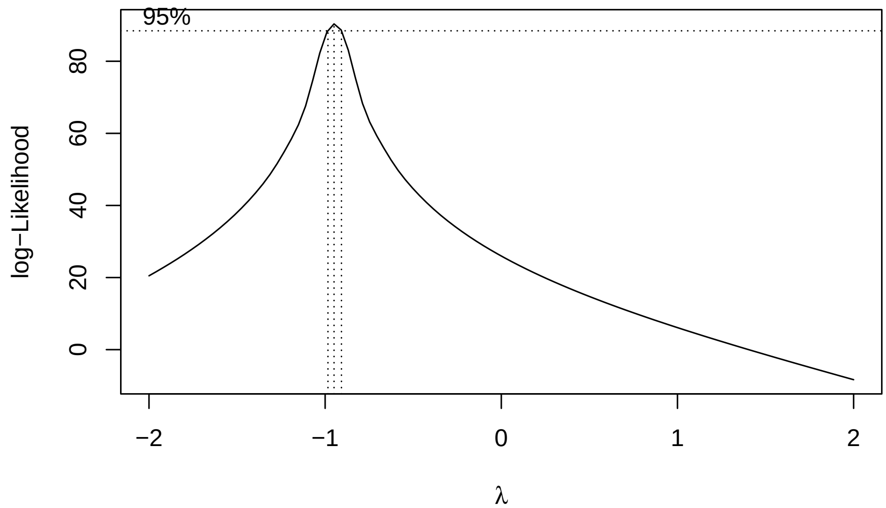



```r
par(mfrow = c(1,1))
```

These plots show clearly that the model is not adequate. All the assumptions made to fit the model are violated in this case.

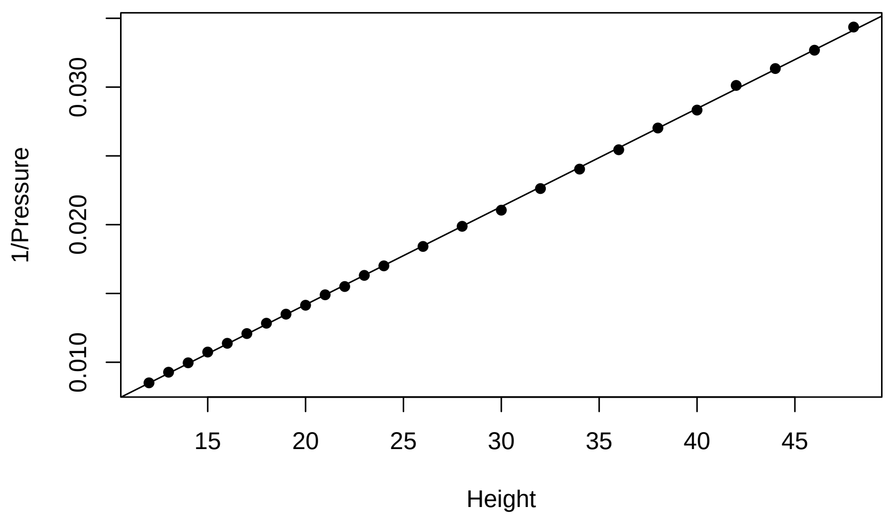
(ii) Use the function `boxcox` on the package `MASS` with the argument set to the model you fitted in (i). If the maximum value in the graph is close to an integer value, use a power transformation with exponent equal to the integer value for `Pressure` and fit a new model. Obtain a summary of the new regression and compare with the previous one. Draw the diagnostic plots and compare with the previous results.

```
library(MASS)
boxcox(model3)
```



The maximum is close to -1, so we make the power transformation $\texttt{Pressure}^{-1} = 1/\texttt{Pressure}$, and fit a new model

```
model4 <- lm(1/Pressure ~ Height, data = data2)
plot(1/Pressure ~ Height, data = data2, pch = 16)
abline(model4)
```



We see that the fit is excellent. Next, look at the summary table:

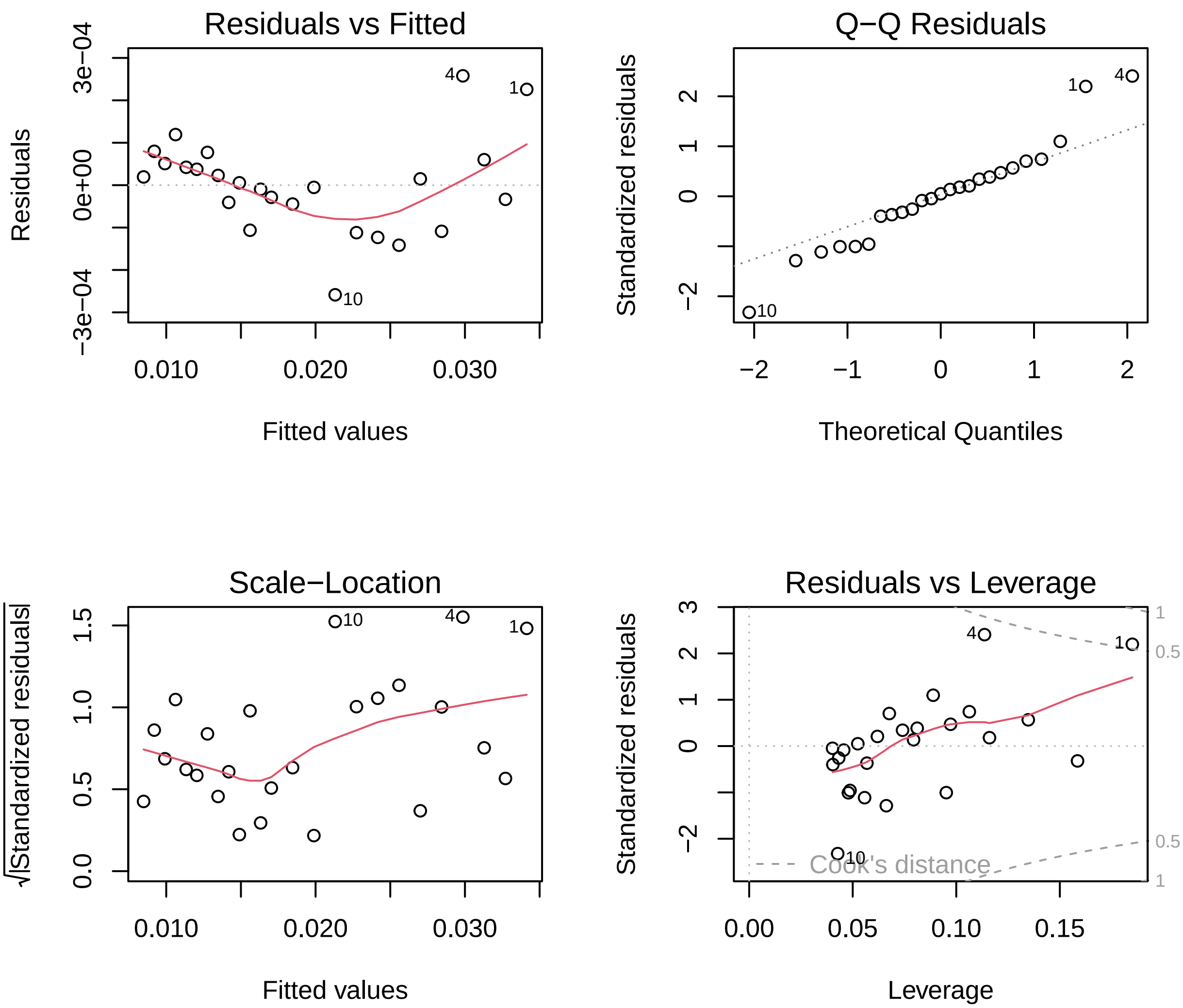
```
summary(model4)
```

```
##
## Call:
## lm(formula = 1/Pressure ~ Height, data = data2)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -2.586e-04 -4.452e-05  5.514e-06  5.090e-05  2.578e-04
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.746e-05  6.073e-05  -1.111    0.278
## Height       7.126e-04  2.076e-06 343.248   <2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0001138 on 23 degrees of freedom
## Multiple R-squared:  0.9998, Adjusted R-squared:  0.9998
## F-statistic: 1.178e+05 on 1 and 23 DF,  p-value: < 2.2e-16
```

The slope has a p-value equal to zero while the intercept has a large p-value, and we cannot reject the null hypothesis that it is equal to zero.

```r
par(mfrow = c(2,2))
plot(model4)
```




```r
par(mfrow = c(1,1))
```

(iii) If the p-value for the intercept is large, fit a model without intercept by adding `+ 0` at the end of the regression equation in the call to the `lm` function. Use this model to write down an equation for the relation between pressure and height for a gas. What would be the predicted `Pressure` for a point with `Height` $= 32$? Draw a scatterplot of `Pressure` against `Height` and add the regression line for the first model and the curve you obtained with the second regression.

```
model5 <- lm(1/Pressure ~ Height + 0, data = data2)
summary(model5)
```

```
##
## Call:
## lm(formula = 1/Pressure ~ Height + 0, data = data2)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -2.619e-04 -6.542e-05 -1.286e-05  2.861e-05  2.801e-04
##
## Coefficients:
##         Estimate Std. Error t value Pr(>|t|)
## Height 7.105e-04  7.821e-07   908.5   <2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0001144 on 24 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 8.253e+05 on 1 and 24 DF,  p-value: < 2.2e-16
```

The equation is

$$\frac{1}{P} = 7.105 \times 10^{-4} \times H$$

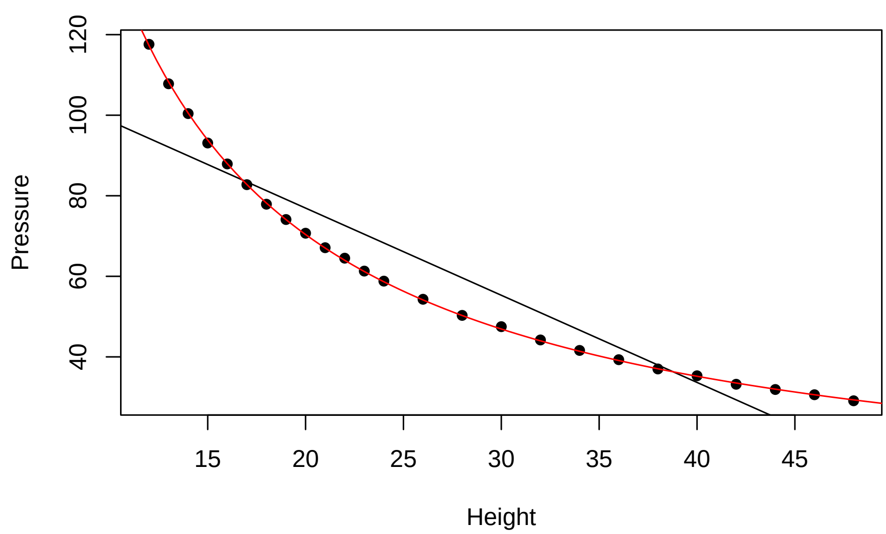
and inverting we get

$$P = \frac{1}{7.105} 10^4 \times \frac{1}{H} = 1407.46 \times \frac{1}{H}$$

If we set $H = 32$ with this model, the pressure is

$$P = 1407.46/32 = 43.983$$

```
plot(Pressure ~ Height, data = data2, pch = 16)
abline(model3)
curve(1407.46/x, 10, 50, add =T, col = 'red')
```



Note

Observe that in this experiment, volume is proportional to height because the gas is enclosed in a cylinder with variable height but fixed radius. The relation between pressure and volume for a gas was investigated by Robert Boyle, who proved that pressure is inversely proportional to volume:

$$P \propto \frac{1}{V}.$$

This is known as Boyle's law. The data that we used for this problem is Boyle's data and the equation we obtained is Boyle's law.

Exercise 4

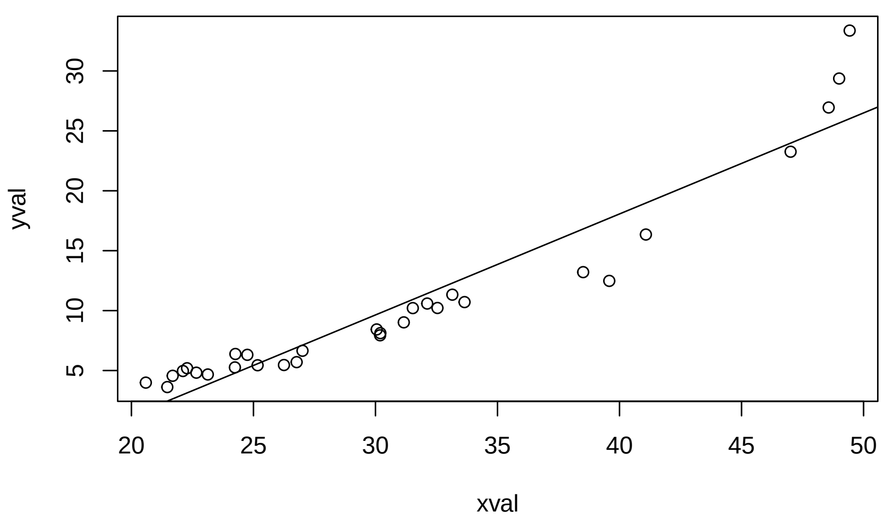
For this problem use the data set `PL825FQ4.txt`.

(i) Read the data and plot `yval` as a function of `xval`. Fit a simple linear regression for `yval` as a function of `xval` and add the regression line to the plot. Comment. Obtain a summary for the regression and draw the diagnostic plots. Comment on the results

```
data4 <- read.table('PL825FQ4.txt', header = T)
plot(yval ~ xval, data = data4)
model4 <- lm(yval ~ xval, data = data4)
summary(model4)
```

```
##
## Call:
## lm(formula = yval ~ xval, data = data4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.2335 -1.4823 -0.5922  1.5268  7.3540
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -15.64828    1.64074  -9.537 2.71e-10 ***
## xval          0.84289    0.05095  16.543 5.50e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.428 on 28 degrees of freedom
## Multiple R-squared:  0.9072, Adjusted R-squared:  0.9039
## F-statistic: 273.7 on 1 and 28 DF,  p-value: 5.503e-16
```

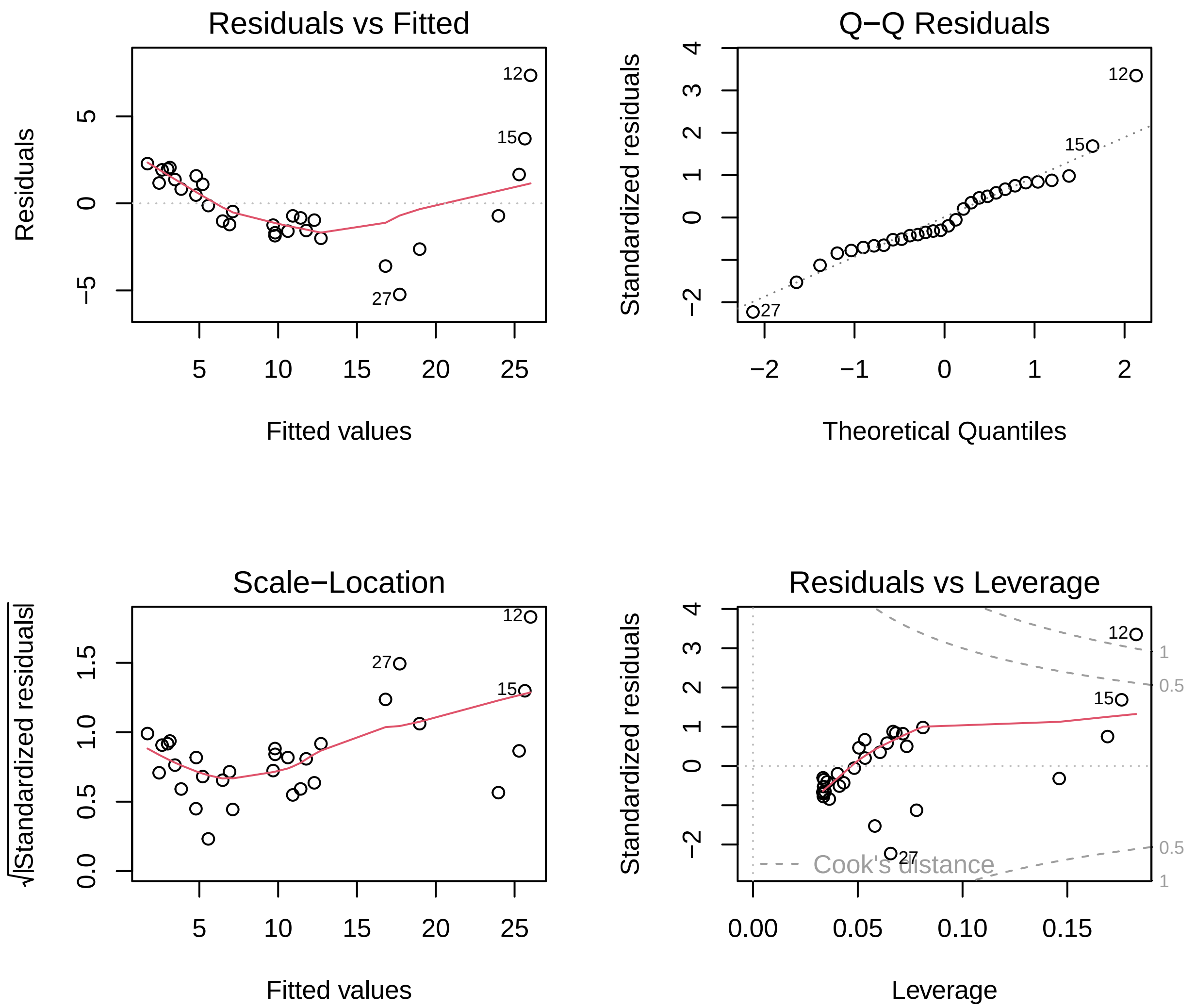
```
abline(model4)
```



From the summary table, things look reasonable. The summary for the residuals points to a symmetric distribution. The median is close to zero and the quartiles are approximately symmetric. The regression is significant and both parameters have small p-values. The R^2 is high, above 0.9.

From the graph, the model does not look good. We can see that points at the extremes are all above the regression line while points in the middle are below. There is a curvature in the data that is not reflected in the model.

```
par(mfrow=c(2,2))
plot(model4)
```

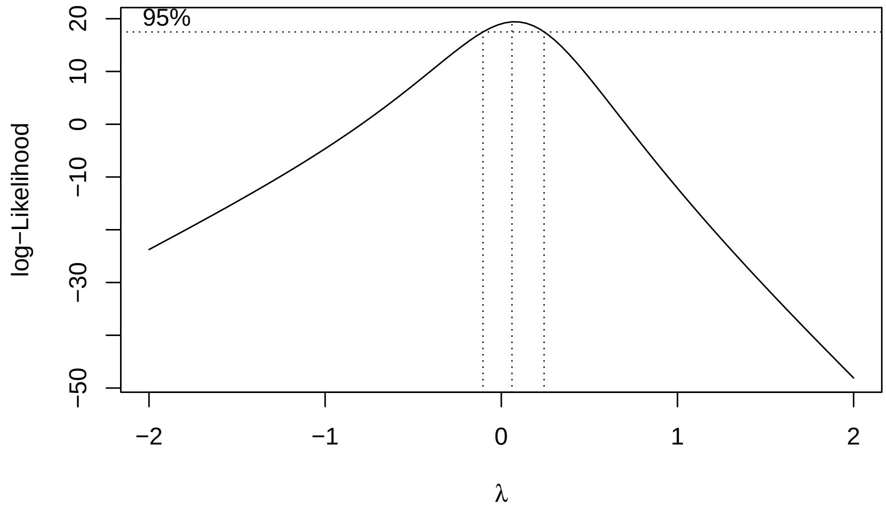



```
par(mfrow=c(1,1))
```

This inadequacy also shows in the diagnostic plots. The first plot shows a curved pattern in the residuals that should not be present if the model was appropriate. The normality plot also shows that the normality assumption for the residuals may be suspect.

(ii) Use the function `boxcox` on the package `MASS` with the argument set to the model you fitted in (i).

```
library(MASS)
boxcox(model4)
```



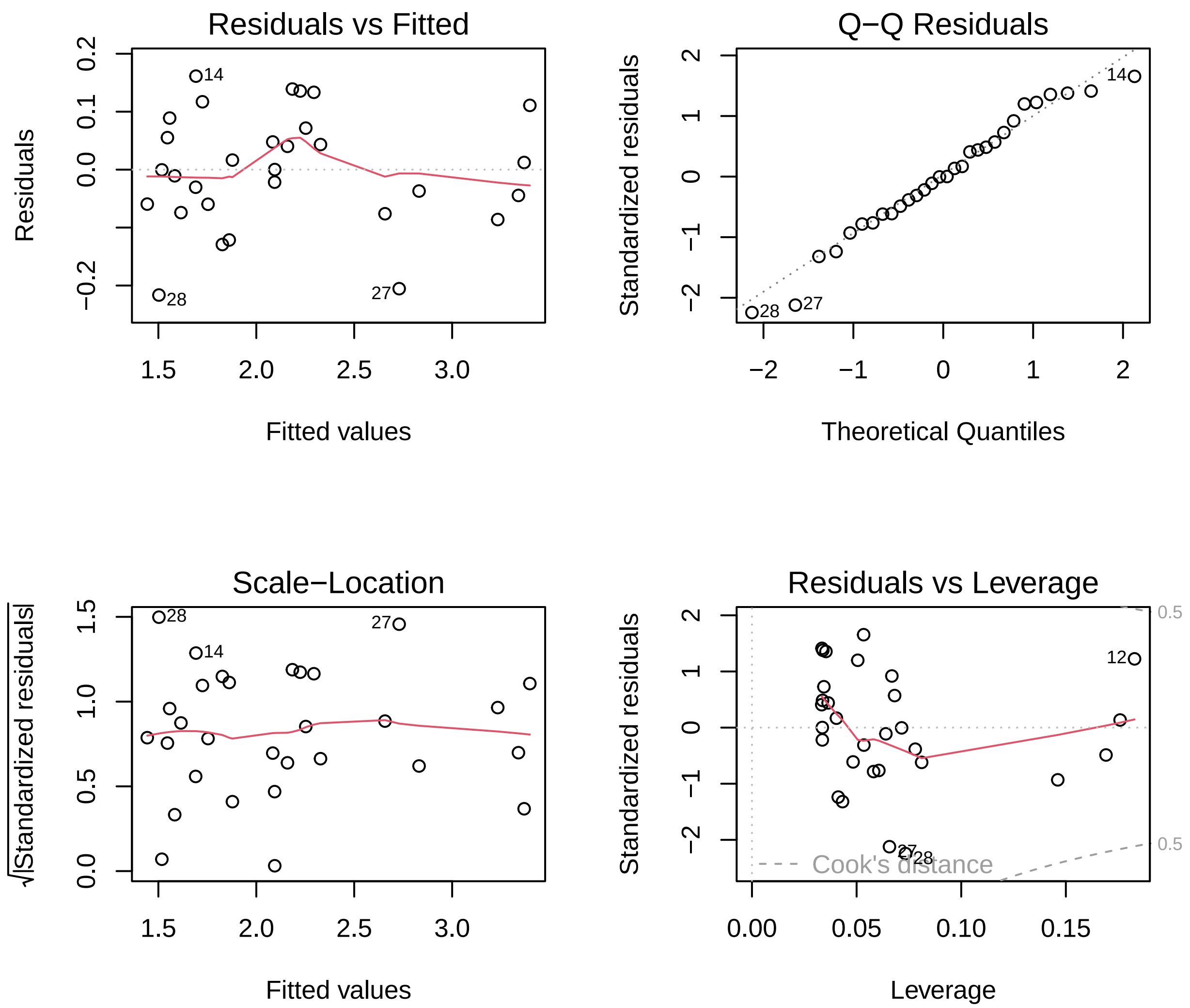
(iii) If the confidence interval in the graph includes zero, use a logarithmic transformation for `yval` and fit a new model. Obtain a summary of the new regression and compare with the previous one. Draw the diagnostic plots and compare with the previous results.

```
model5 <- lm(log(yval) ~ xval, data = data4)
summary(model5)
```

```
##
## Call:
## lm(formula = log(yval) ~ xval, data = data4)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.216419 -0.059677 -0.000187  0.067498  0.161290
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.048588   0.067694   0.718    0.479
## xval        0.067737   0.002102  32.223   <2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1002 on 28 degrees of freedom
## Multiple R-squared:  0.9737, Adjusted R-squared:  0.9728
## F-statistic:  1038 on 1 and 28 DF,  p-value: < 2.2e-16
```

The residual error has been halved and the R^2 (both versions) have increased from 0.9 to 0.97. The summary for the residuals still points to a symmetric distribution.

```
par(mfrow=c(2,2))
plot(model5)
```




```
par(mfrow=c(1,1))
```

The diagnostic plots look much better now. There is no pattern in the residuals, the quantile plot is good, pointing to normal residuals. There are few points with large leverage, none of them with large Cook's distance.

(iv) Write down the final model in terms of the original variables. Draw a scatterplot of `yval` against `xval` and add the regression line for the first model and the curve you obtained with the second regression.

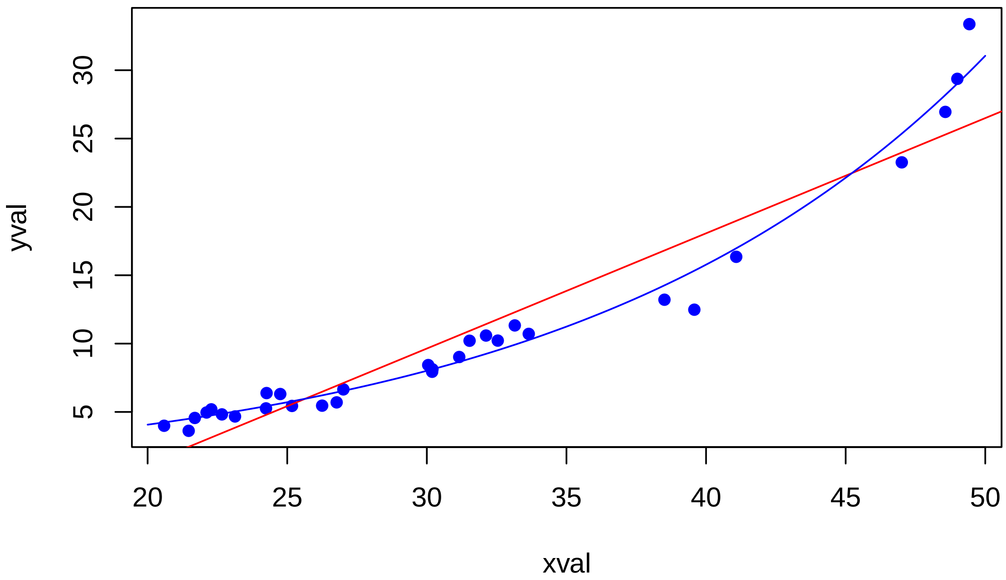
The model is

$$\ln(yval) = 0.04872 - 0.06761 \cdot xval$$

and in terms of the original variables

$$yval = \exp\{0.04872 + 0.06761 \cdot xval\} = 1.049926 \exp\{0.06761 \cdot xval\}$$

```
plot(yval ~ xval, data = data4, pch = 16, col = 'blue')
abline(model4, col = 'red')
curve(exp(0.04872+(0.06774*x)),20,50, add=TRUE, col = 'blue')
```


yval
5
10
15
20
25
30
20
25
30
35
40
45
50
xval