Simple Regression with R

Code

R. Nicholls / D.-L. Couturier / M. Fernandes

Last modified: 04 Mar 2019

Section 1: Correlation Coefficients

We'll start by generating some synthetic data to investigate correlation coefficients.

Generate 50 random numbers in the range [0,50]:

Hide

```
x = runif(50,0,50)
```

Now let's generate some y-values that are linearly correlated with the x-values with gradient=1, applying a random Normal offset (with sd=5):

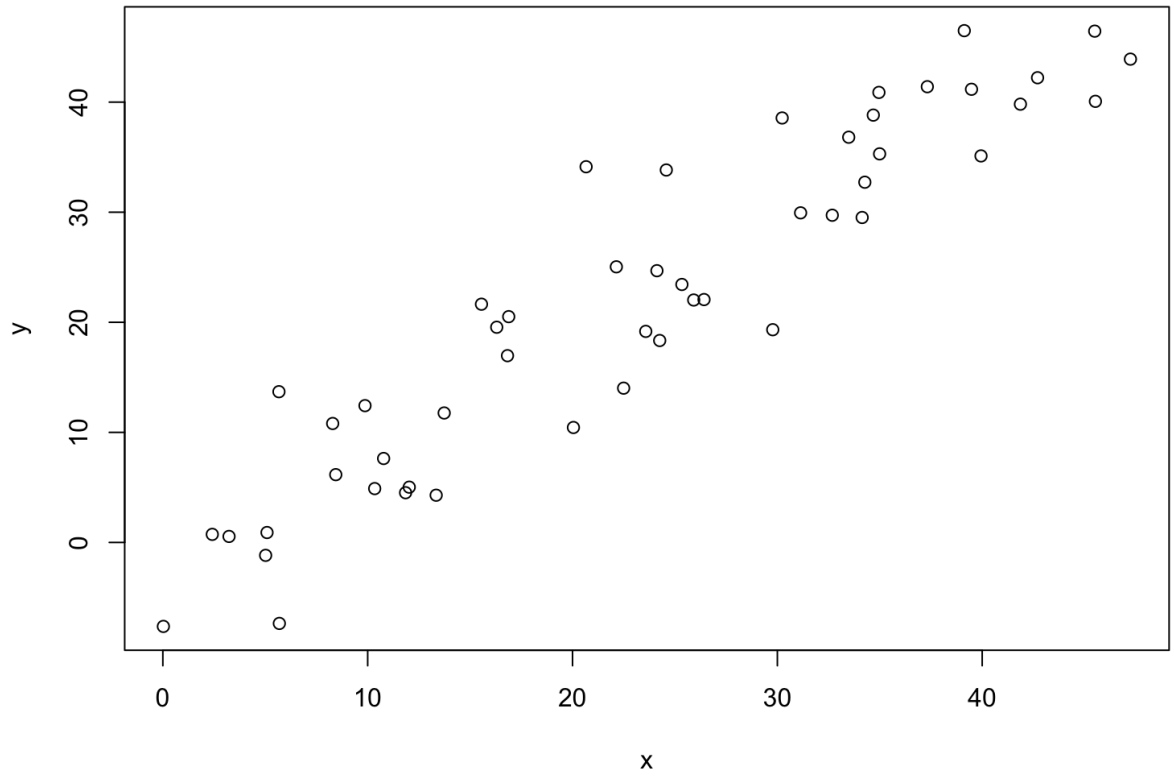
Hide

```
y = x + rnorm(50,0,5)
```

Plotting y against x, you'll observe a positive linear relationship:

Hide

```
plot(y~x)
```



This strong linear relationship is reflected in the correlation coefficient and in the coefficient of determination (R^2):

Hide

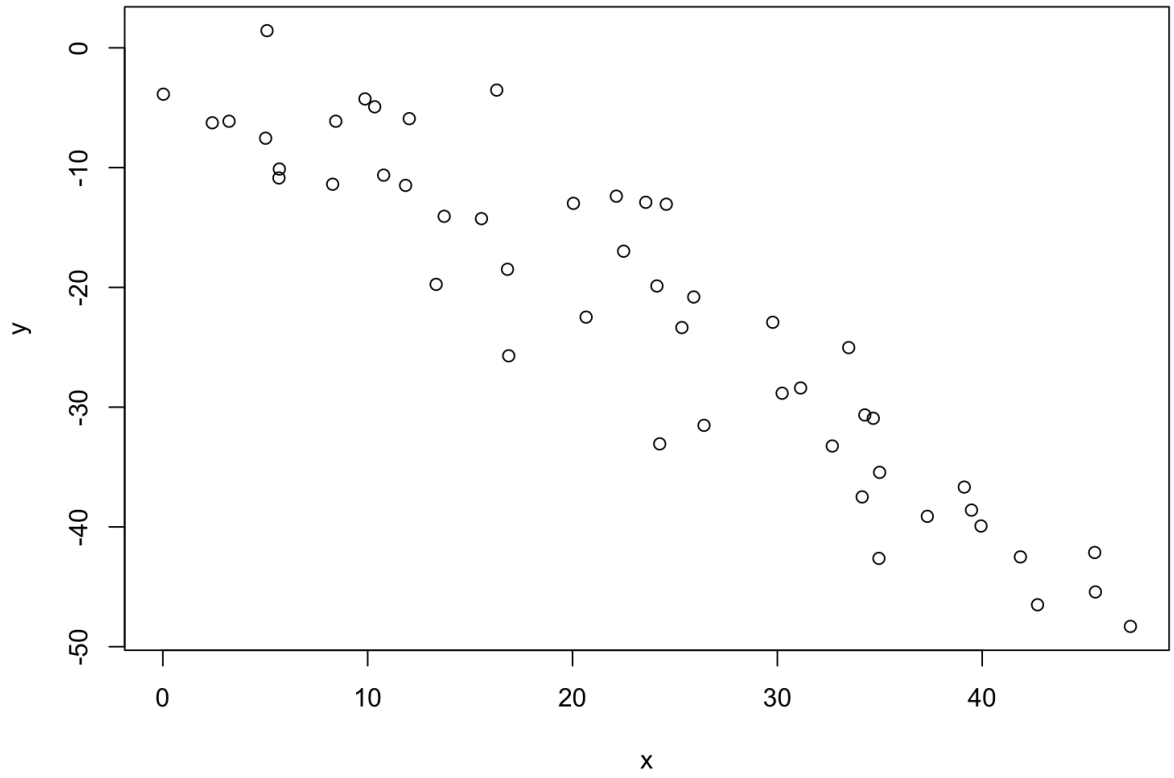
```
pearson_cor_coef = cor(x,y)
list("cor"=pearson_cor_coef,"R^2"=pearson_cor_coef^2)
```

```
## $cor
## [1] 0.9328884
##
## $`R^2`
## [1] 0.8702808
```

If the data exhibit a negative linear correlation then the correlation coefficient will become strong and negative, whilst the R^2 value will remain strong and positive:

Hide

```
y = -x + rnorm(50,0,5)
plot(y~x)
```



Hide

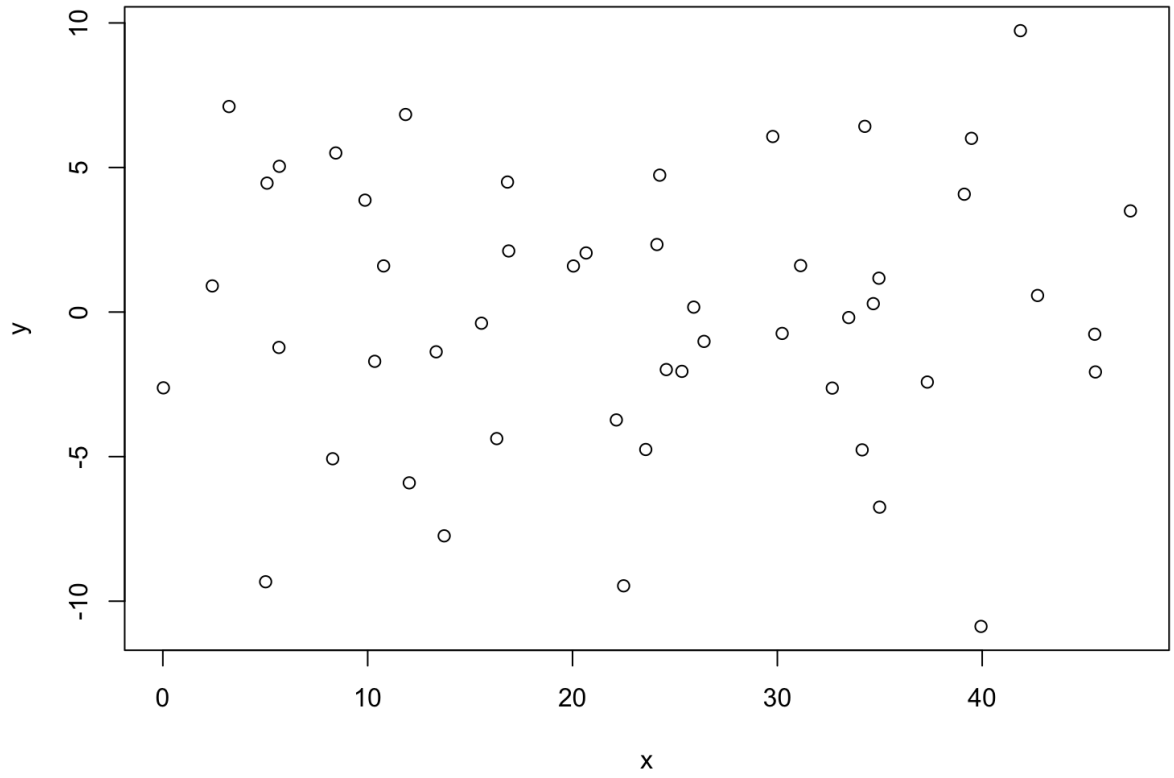
```
pearson_cor_coef = cor(x,y)
list("cor"=pearson_cor_coef,"R^2"=pearson_cor_coef^2)
```

```
## $cor
## [1] -0.9274861
##
## $`R^2`
## [1] 0.8602304
```

If data are uncorrelated then both the correlation coefficient and R^2 values will be close to zero:

Hide

```
y = rnorm(50,0,5)
plot(y~x)
```



Hide

```
pearson_cor_coef = cor(x,y)
list("cor"=pearson_cor_coef,"R^2"=pearson_cor_coef^2)
```

```
## $cor
## [1] 0.02191476
##
## $`R^2`
## [1] 0.0004802566
```

The significance of a correlation can be tested using `cor.test()`, which also provides a 95% confidence interval on the correlation:

Hide

```
cor.test(x,y)
```

```
##
##  Pearson's product-moment correlation
##
## data:  x and y
## t = 0.15187, df = 48, p-value = 0.8799
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2580068  0.2984420
## sample estimates:
##        cor
## 0.02191476
```

In this case, the value 0 is contained within the confidence interval, indivating that there is insufficient evidence to reject the null hypothesis that the true correlation is equal to zero.

Section 2: Simple Regression

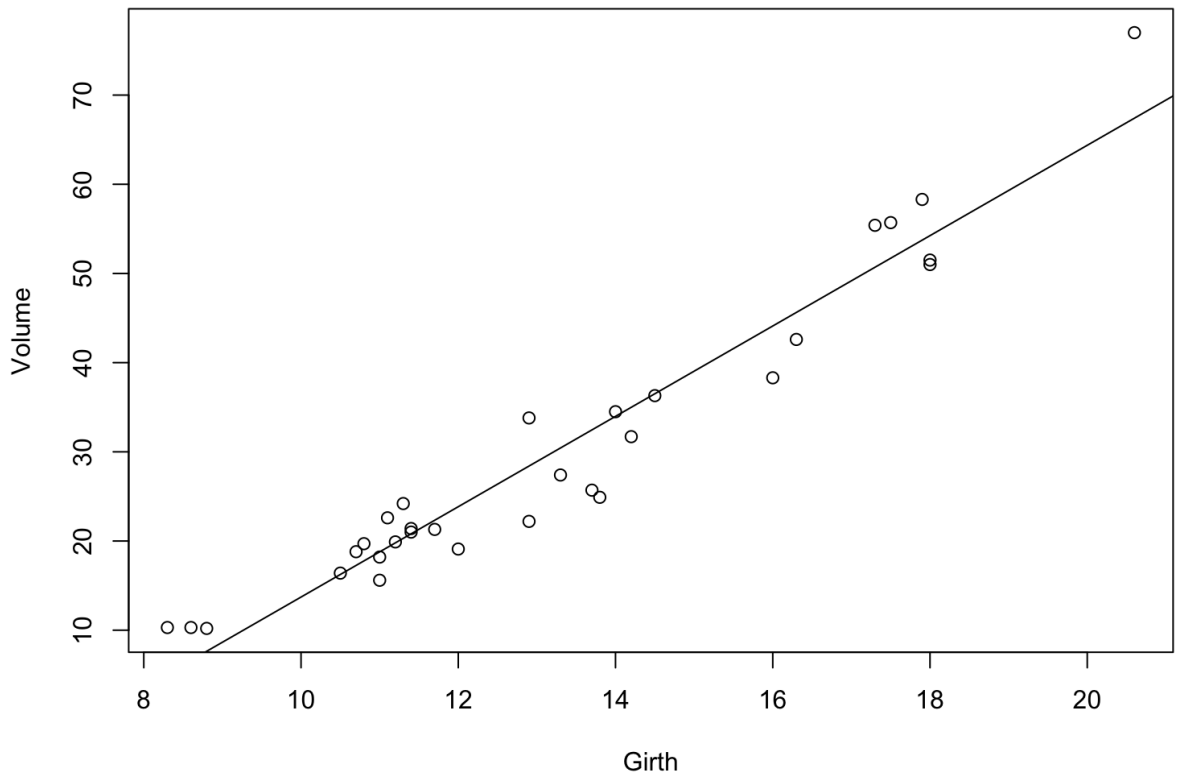
Now let's look at some real data.

The in-built dataset `trees` contains data pertaining to the `Volume`, `Girth` and `Height` of 31 felled black cherry trees.

We will now attempt to construct a simple linear model that uses `Girth` to predict `Volume`.

Hide

```
plot(Volume~Girth,data=trees)
m1 = lm(Volume~Girth,data=trees)
abline(m1)
```



Hide

```
cor.test(trees$Volume,trees$Girth)
```

```
##
##  Pearson's product-moment correlation
##
## data:  trees$Volume and trees$Girth
## t = 20.478, df = 29, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9322519 0.9841887
## sample estimates:
##       cor
## 0.9671194
```

It is evident that `Volume` and `Girth` are highly correlated.

The summary for the linear model provides information regarding the quality of the model:

Hide

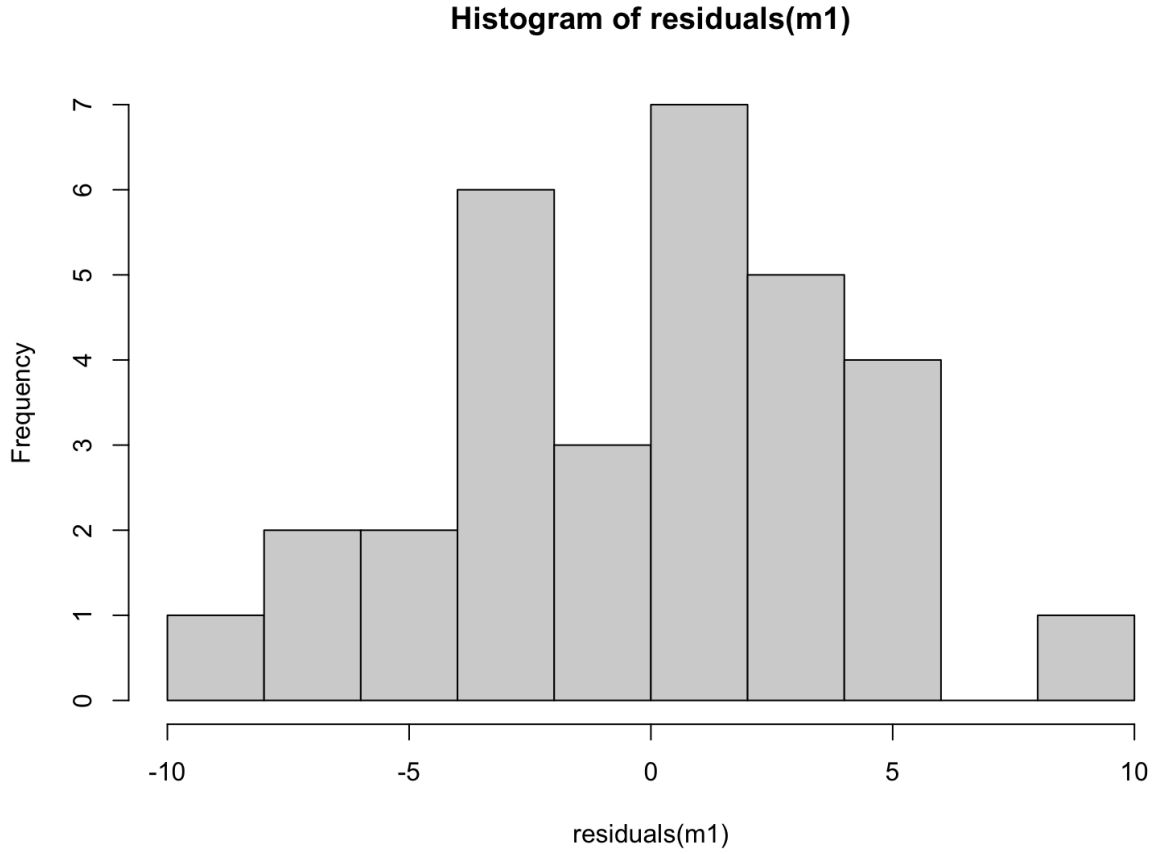
```
summary(m1)
```

```
##
## Call:
## lm(formula = Volume ~ Girth, data = trees)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.065 -3.107  0.152  3.495  9.587
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -36.9435     3.3651  -10.98 7.62e-12 ***
## Girth         5.0659     0.2474   20.48  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.252 on 29 degrees of freedom
## Multiple R-squared:  0.9353, Adjusted R-squared:  0.9331
## F-statistic: 419.4 on 1 and 29 DF,  p-value: < 2.2e-16
```

Model residuals can be readily accessed using the `residuals()` function:

Hide

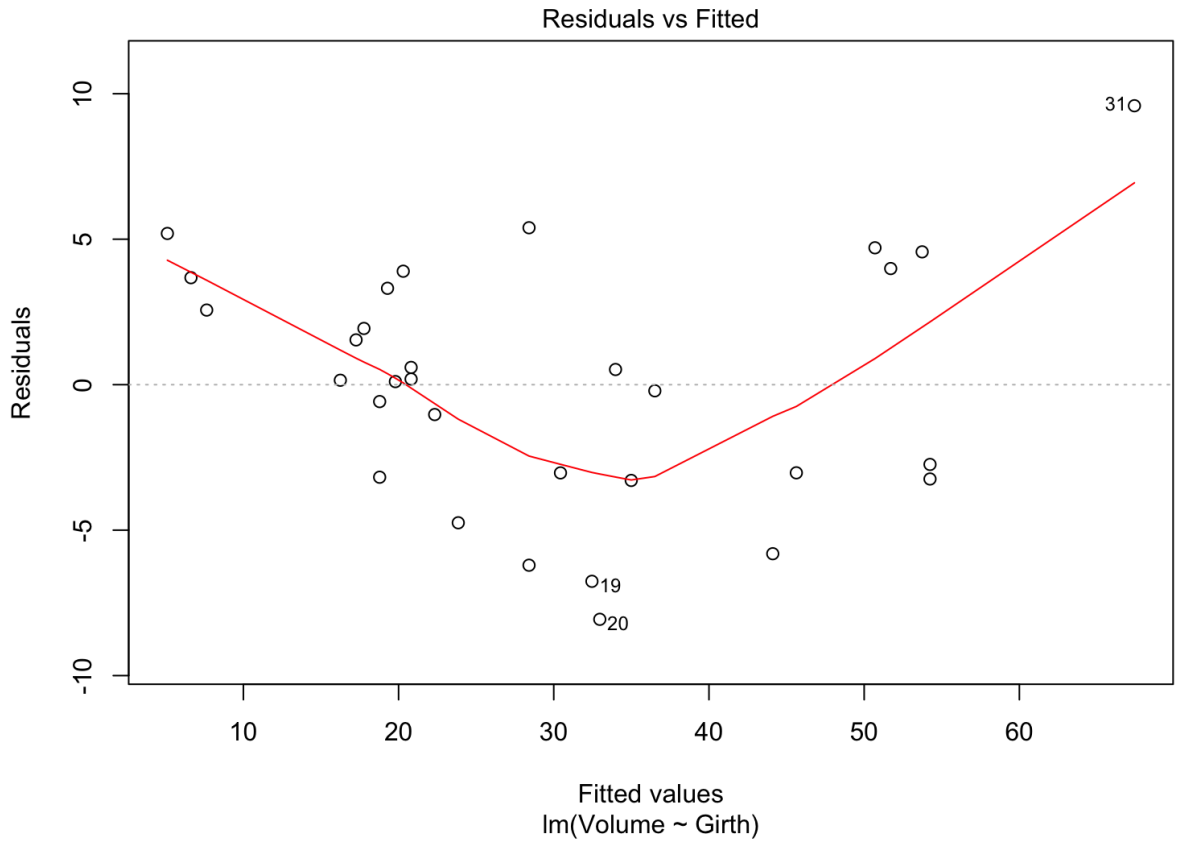
```
hist(residuals(m1),breaks=10,col="light grey")
```

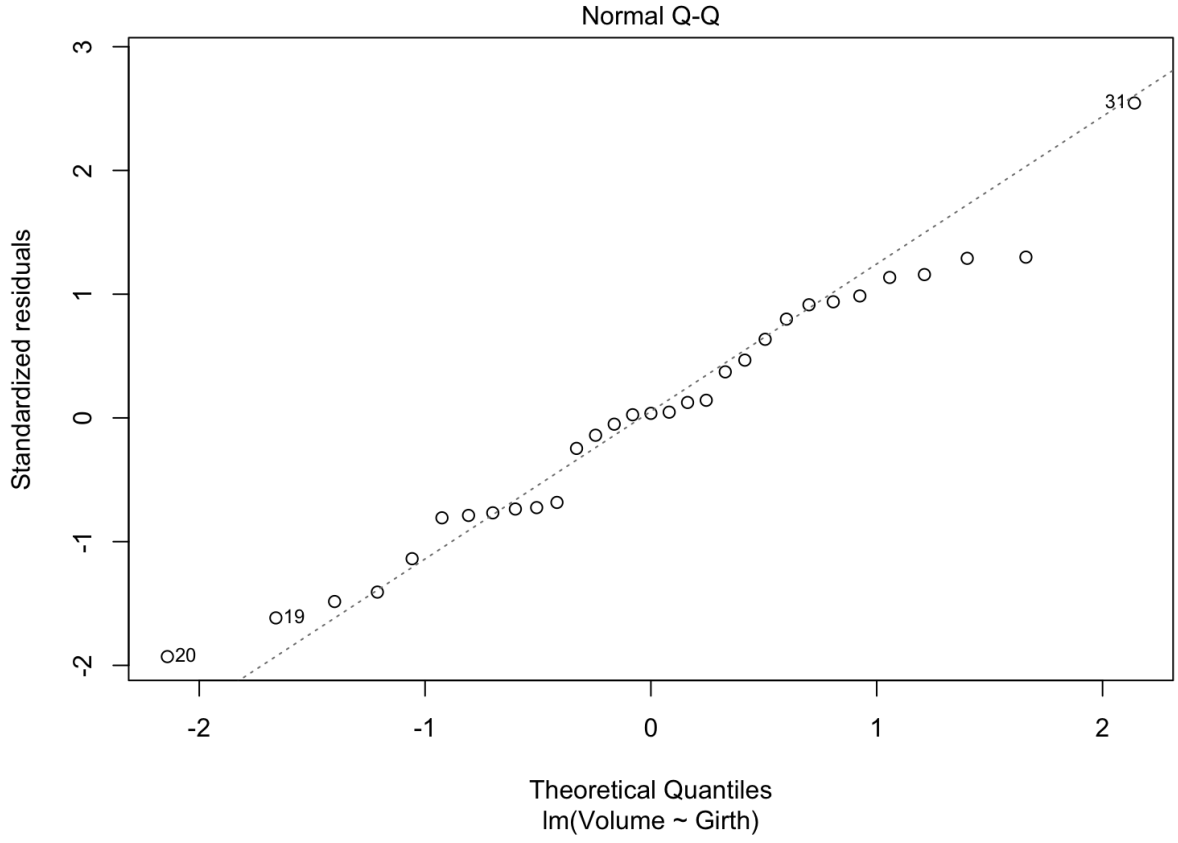


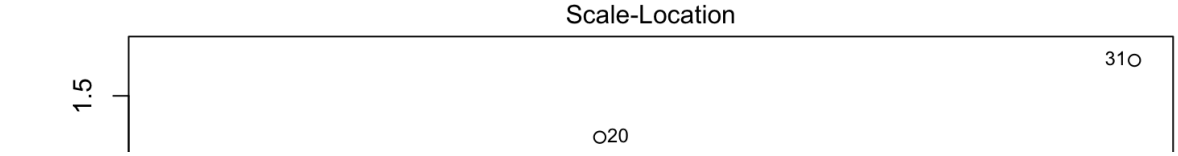
Diagnostic plots for the model can reveal whether or not modelling assumptions are reasonable. In this case, there is visual evidence to suggest that the assumptions are not satisfied - note in particular the trend observed in the plot of residuals vs fitted values:

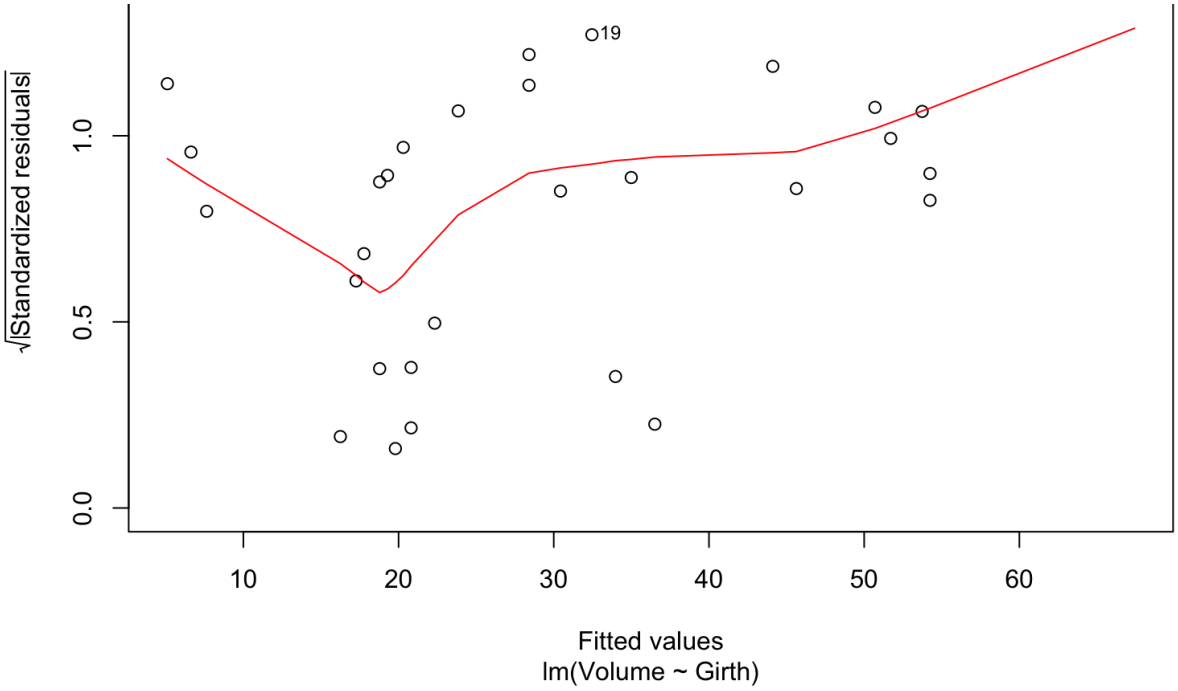
Hide

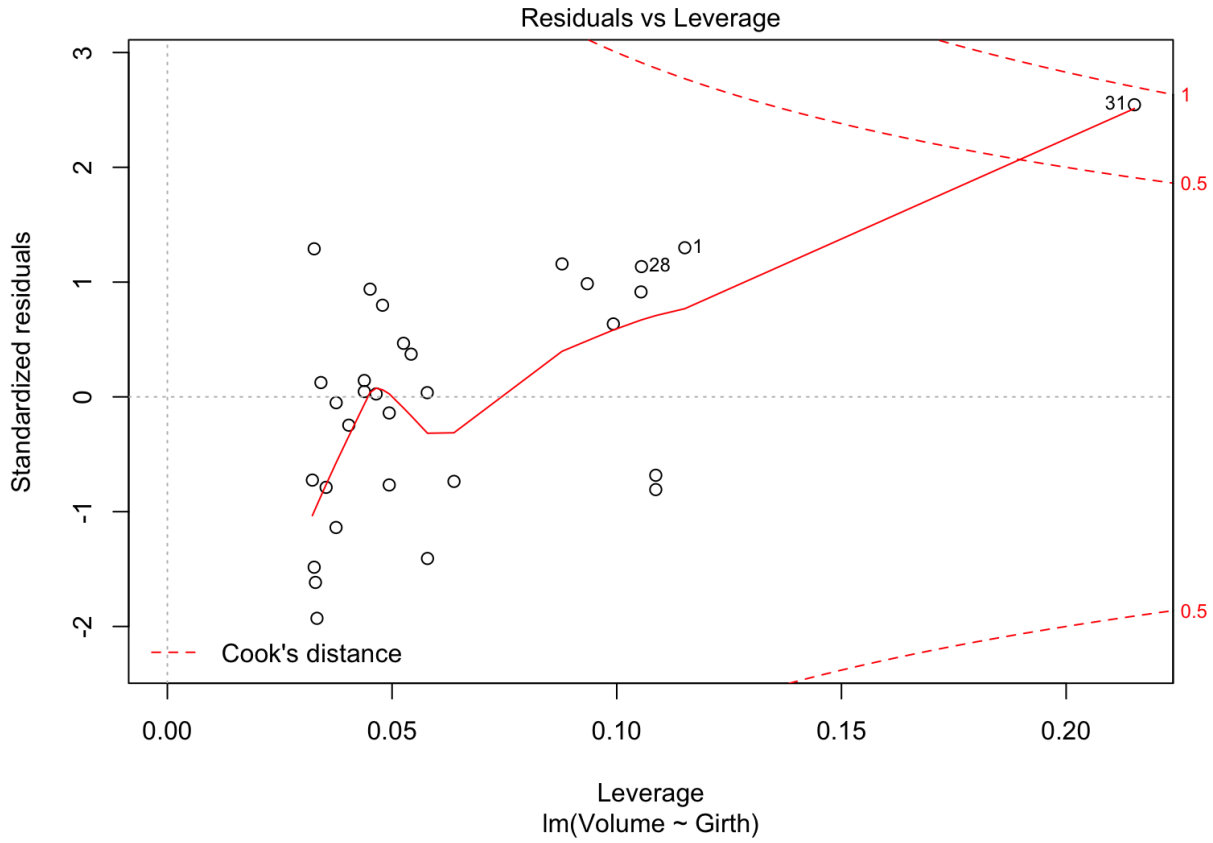
```
plot(m1)
```


Residuals vs Fitted
Residuals
Fitted values
lm(Volume ~ Girth)
31
19
20


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Volume ~ Girth)
31
19
20


Scale-Location
31
20



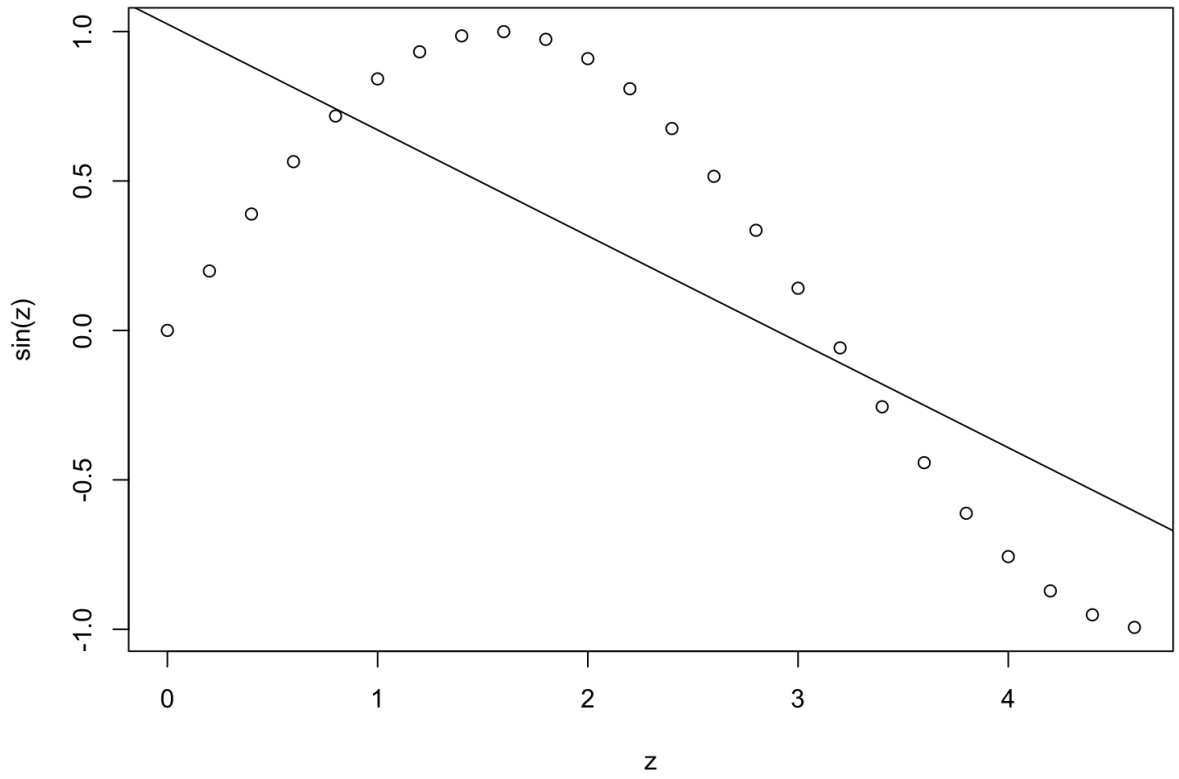


Section 3: Assessing the quality of linear models

Let's see what happens if we try to describe a non-linear relationship using a linear model. Consider the sine function in the range [0,1.5*pi):

Hide

```
z = seq(0,1.5*pi,0.2)
plot(sin(z)~z)
m0 = lm(sin(z)~z)
abline(m0)
```



In this case, it is clear that a linear model is not appropriate for describing the relationship. However, we are able to fit a linear model, and the linear model summary does not identify any major concerns:

Hide

```
summary(m0)
```

```
##
## Call:
## lm(formula = sin(z) ~ z)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.02542 -0.37054  0.01294  0.42276  0.59274
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.02542    0.18641   5.501 1.58e-05 ***
## z           -0.35443    0.06944  -5.104 4.09e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.471 on 22 degrees of freedom
## Multiple R-squared:  0.5422, Adjusted R-squared:  0.5214
## F-statistic: 26.05 on 1 and 22 DF,  p-value: 4.094e-05
```

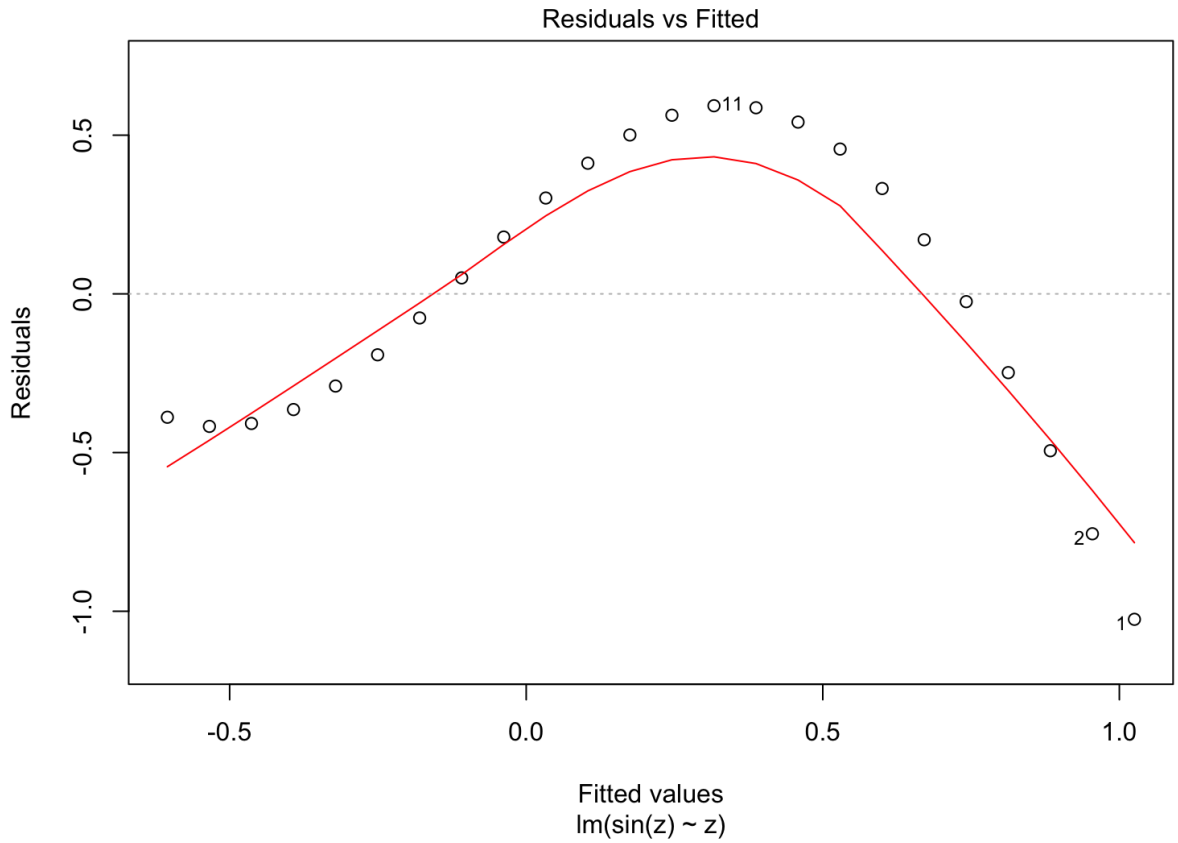
Here we see that the overall p-value is low enough to suggest that the model has significant utility, and both terms (the intercept and the coefficient of `z`) are significantly different from zero. The R^2 value of 0.5422 is high enough to indicate that there is a reasonably strong correlation between `sin(z)` and `z` in this range.

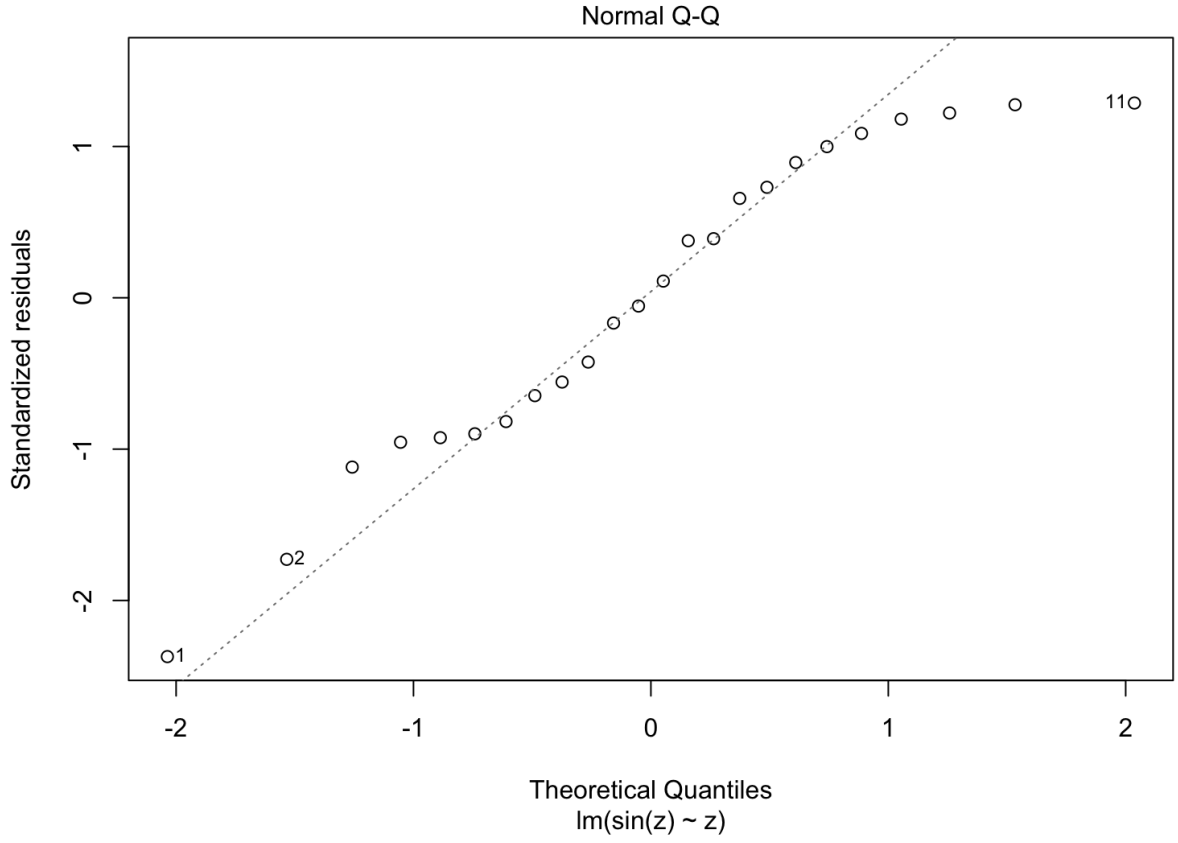
This information is misleading, as we know that a linear model is inappropriate in this case. Indeed, the linear model summary does not check whether the underlying model assumptions are satisfied.

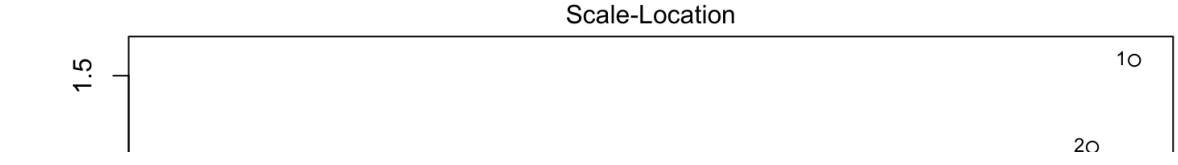
By observing strong patterns in the diagnostic plots, we can see that the modelling assumptions are not satisified in this case.

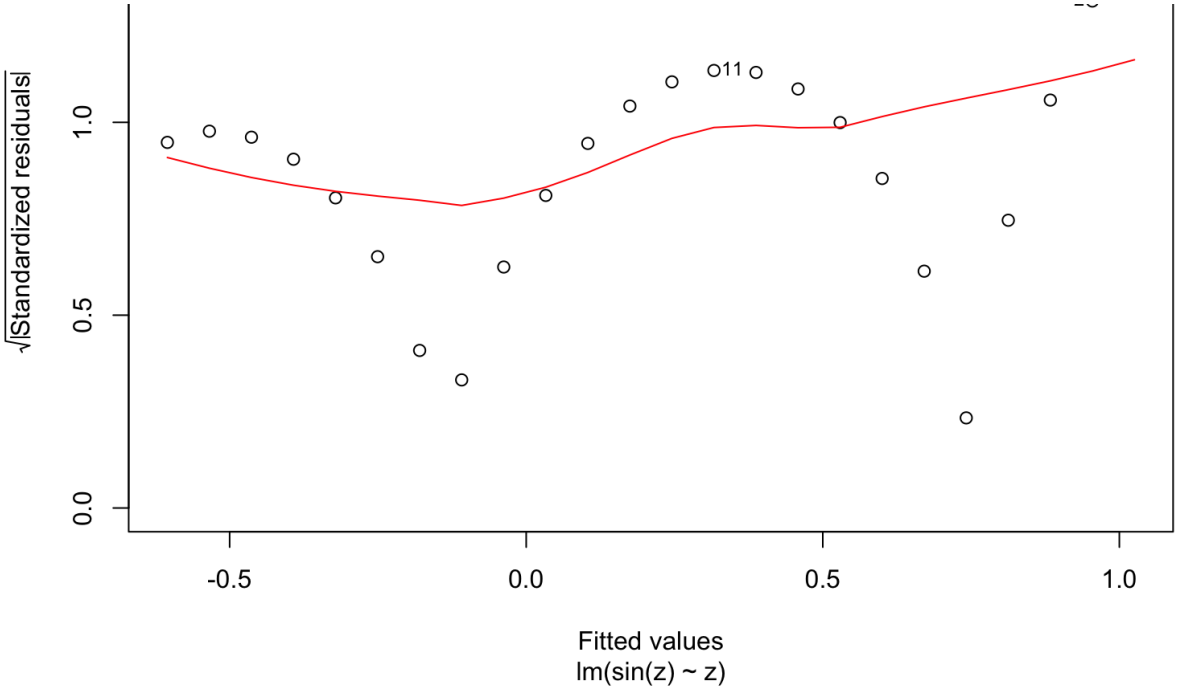
Hide

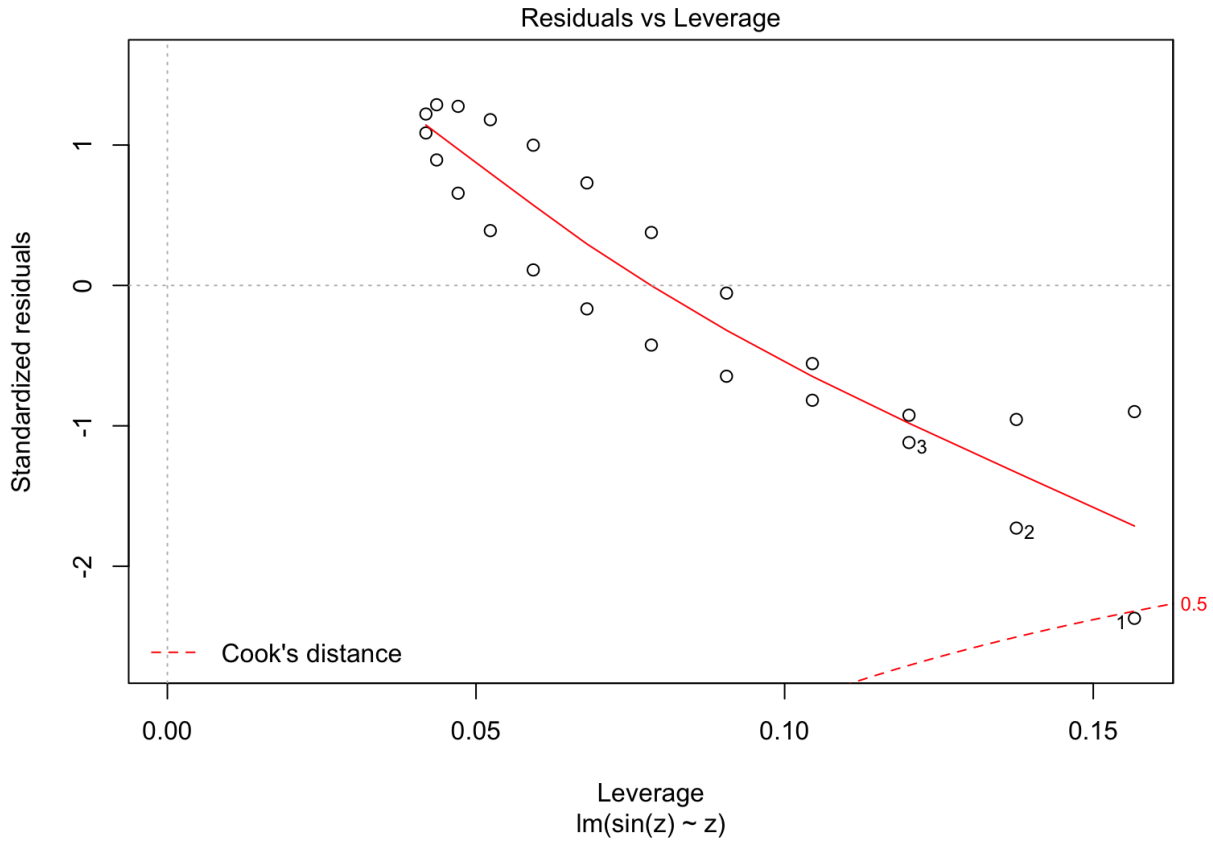
```
plot(m0)
```


Residuals vs Fitted
Residuals
Fitted values
lm(sin(z) ~ z)
11
2
1


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(sin(z) ~ z)
11
2
1


Scale-Location
1
2





Section 4: Modelling Non-Linear Relationships

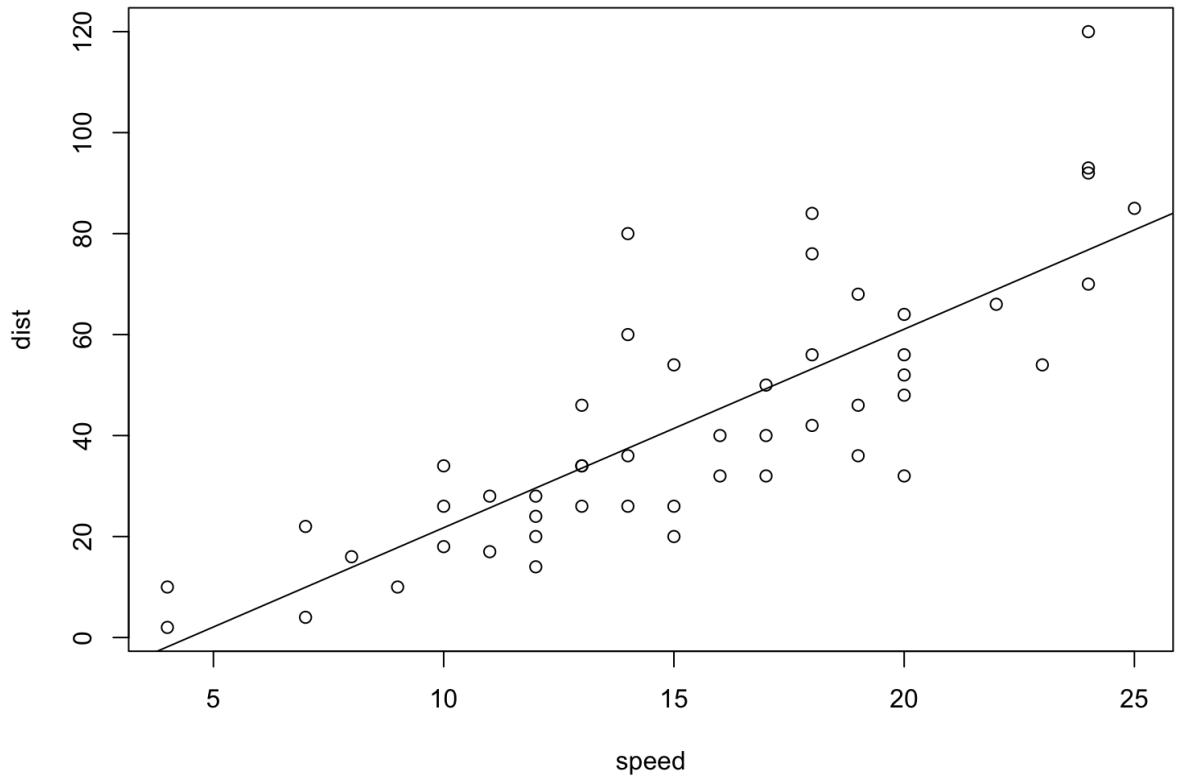
It is sometimes possible to use linear models to describe non-linear relationships (which is perhaps counterintuitive!). This can be achieved by applying transformations to the variable(s) in order to linearise the relationship, whilst ensuring that modelling assumptions are satisfied.

Another in-built dataset `cars` provides the speeds and associated stopping distances of cars in the 1920s.

Let's construct a linear model to predict stopping distance using speed:

Hide

```
plot(dist~speed,data=cars)
m2 = lm(dist~speed,data=cars)
abline(m2)
```



Hide

```
summary(m2)
```

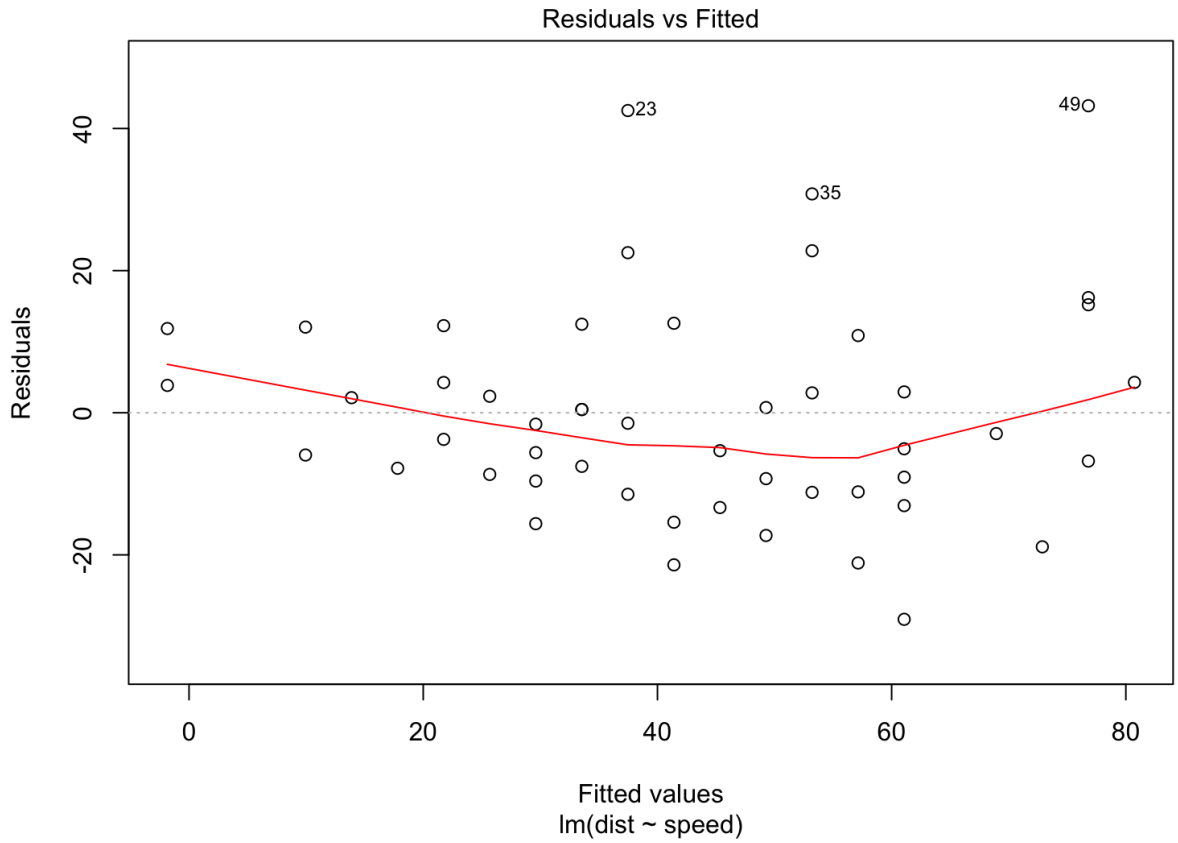
```
##
## Call:
## lm(formula = dist ~ speed, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.069  -9.525  -2.272   9.215  43.201
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.5791     6.7584  -2.601   0.0123 *
## speed         3.9324     0.4155   9.464 1.49e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.38 on 48 degrees of freedom
## Multiple R-squared:  0.6511, Adjusted R-squared:  0.6438
## F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

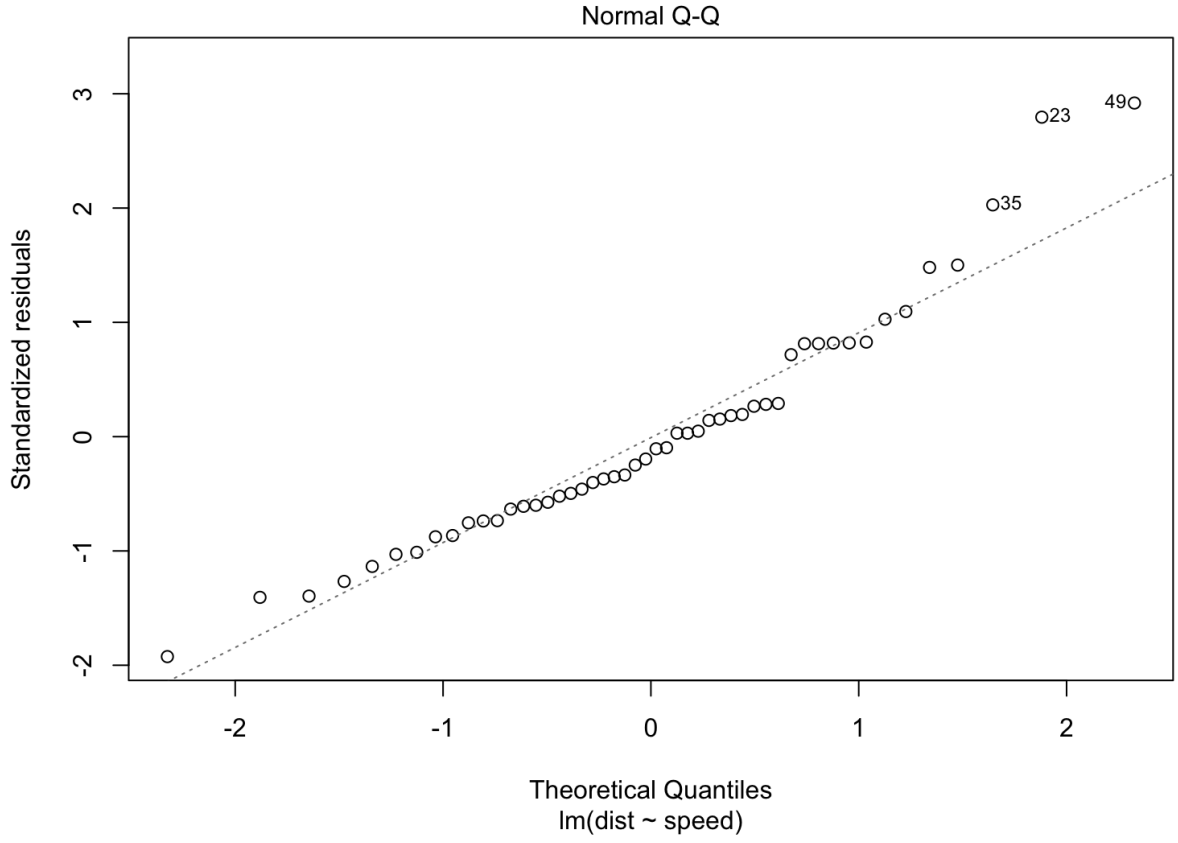
The model summary indicates that the intercept term does not have significant utility. So that term could/should be removed from the model.

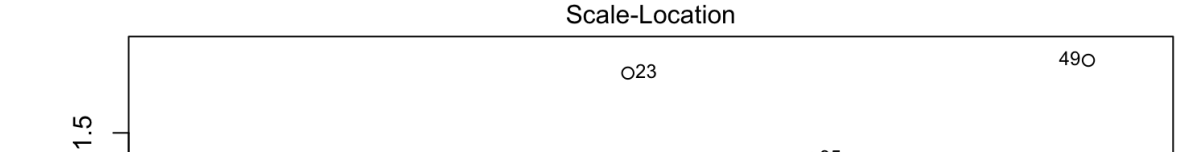
In addition, the plot of residuals versus fitted values indicates potential issues with variance stability:

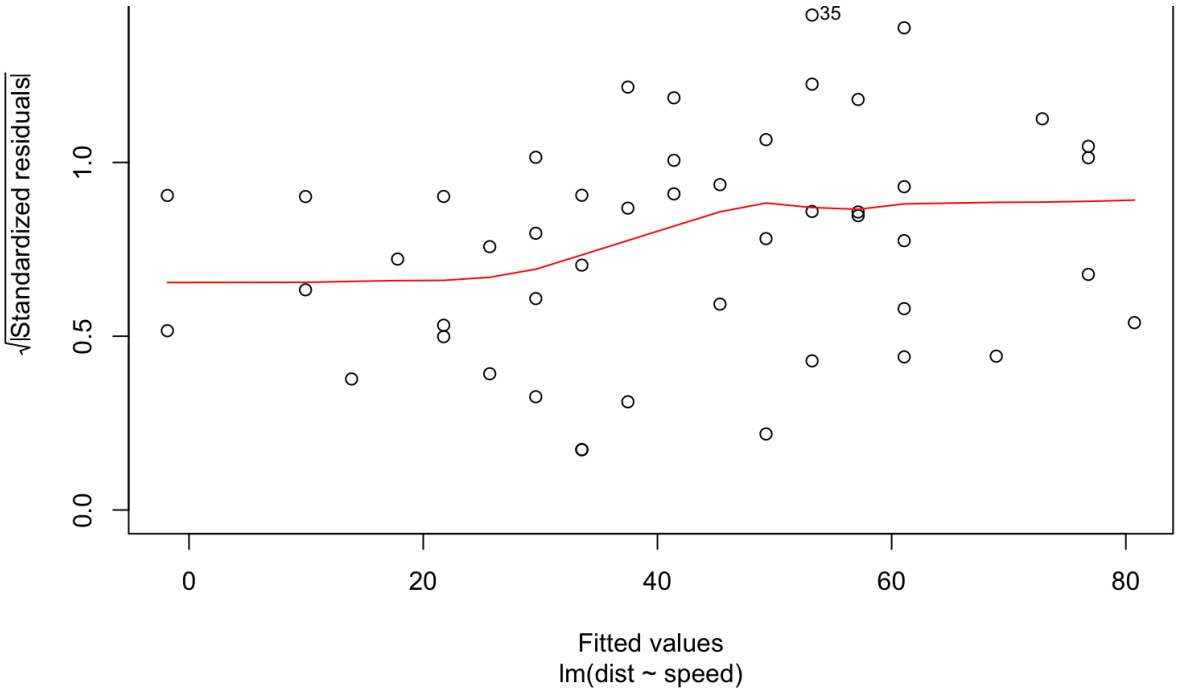
Hide

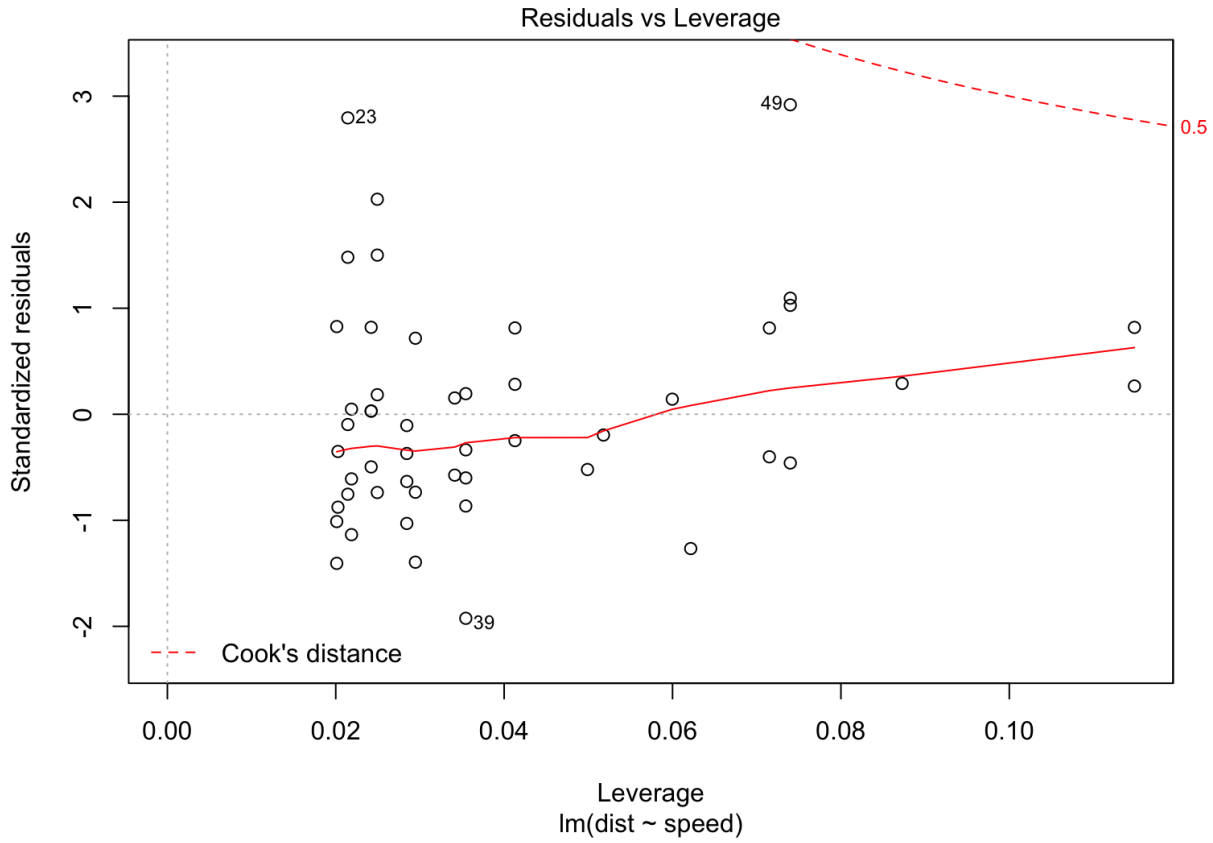
```
plot(m2)
```


Residuals vs Fitted
Residuals
Fitted values
lm(dist ~ speed)
23
35
49


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(dist ~ speed)
23
35
49


Scale-Location
23
49

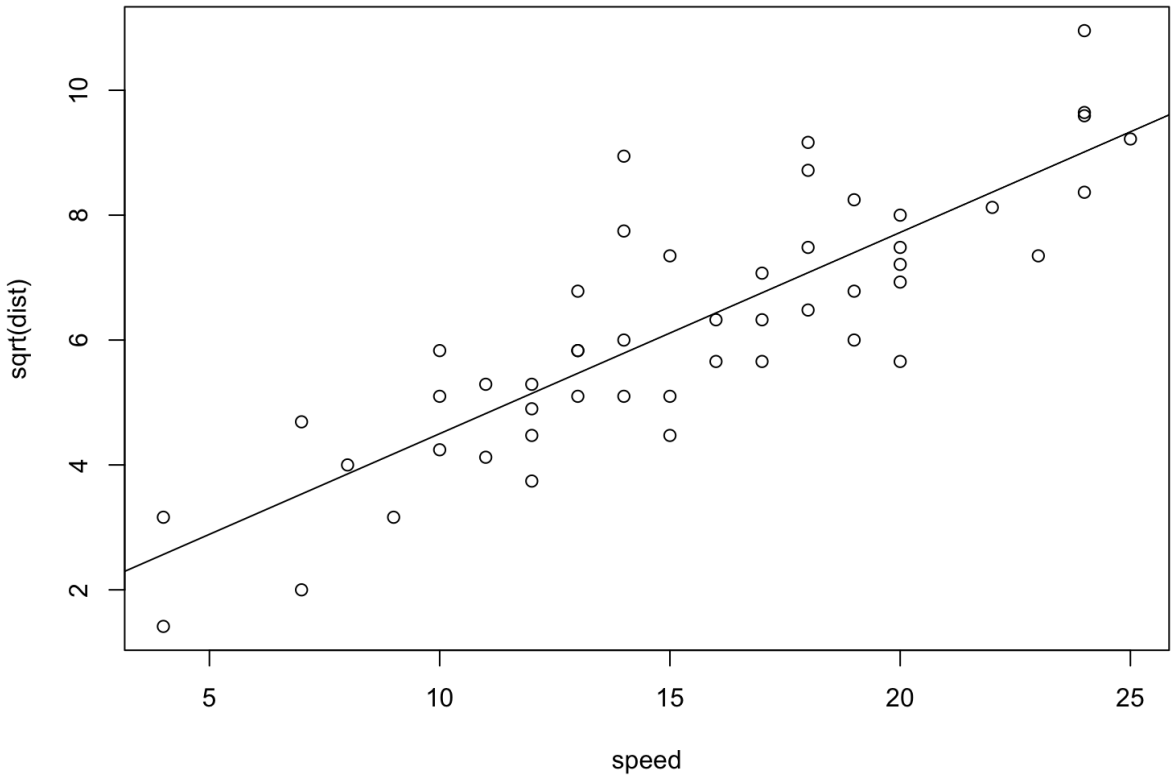




In this case, variance stability can be aided by a square-root transformation of the response variable:

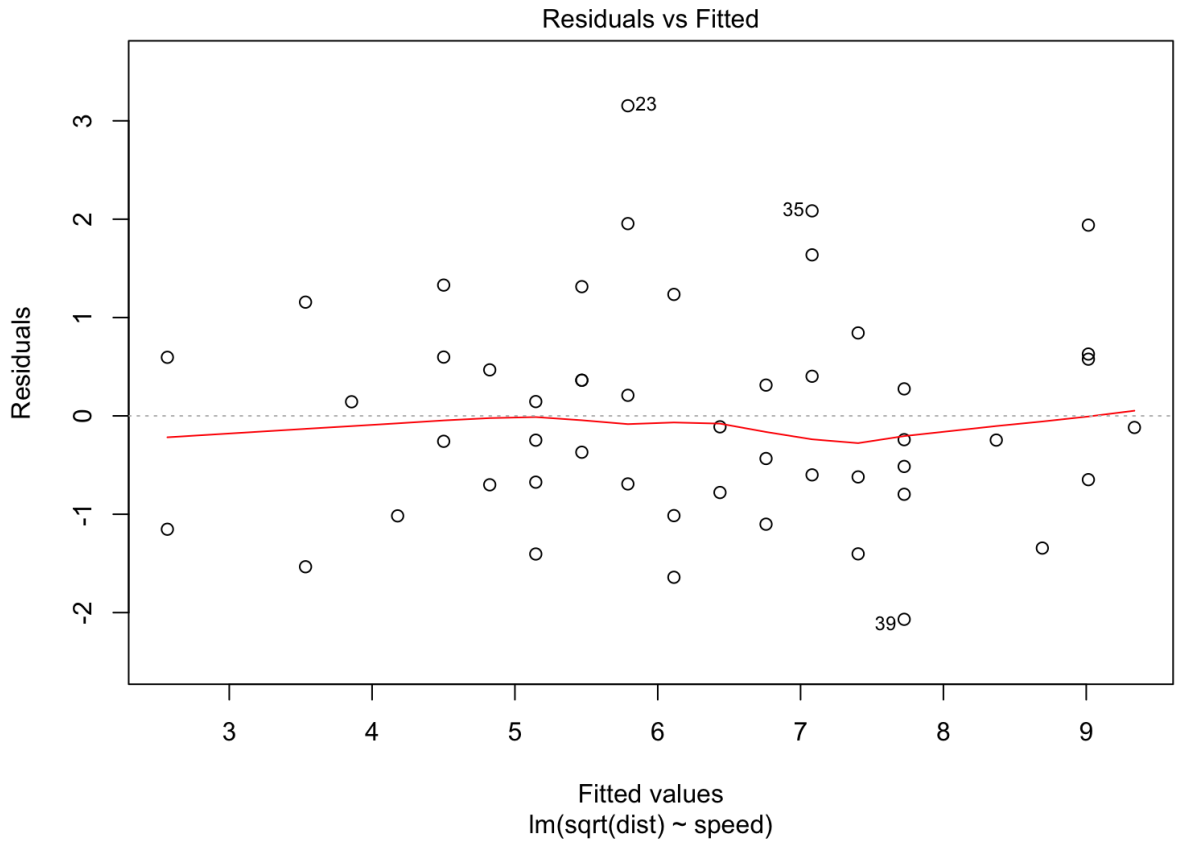
Hide

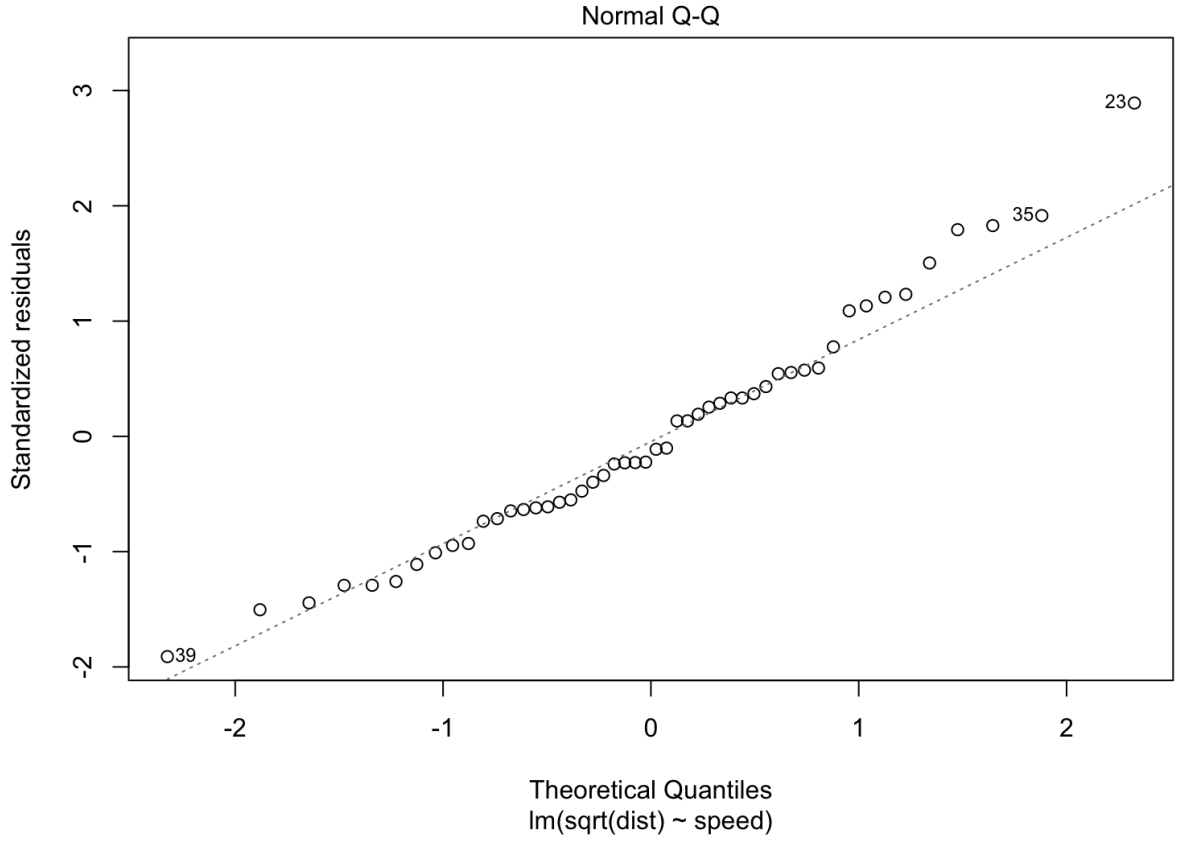
```
plot(sqrt(dist)~speed,data=cars)
m3 = lm(sqrt(dist)~speed,data=cars)
abline(m3)
```

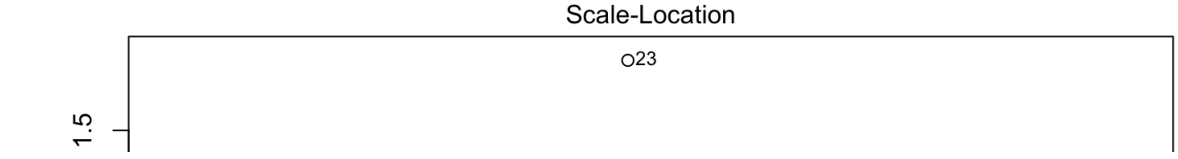


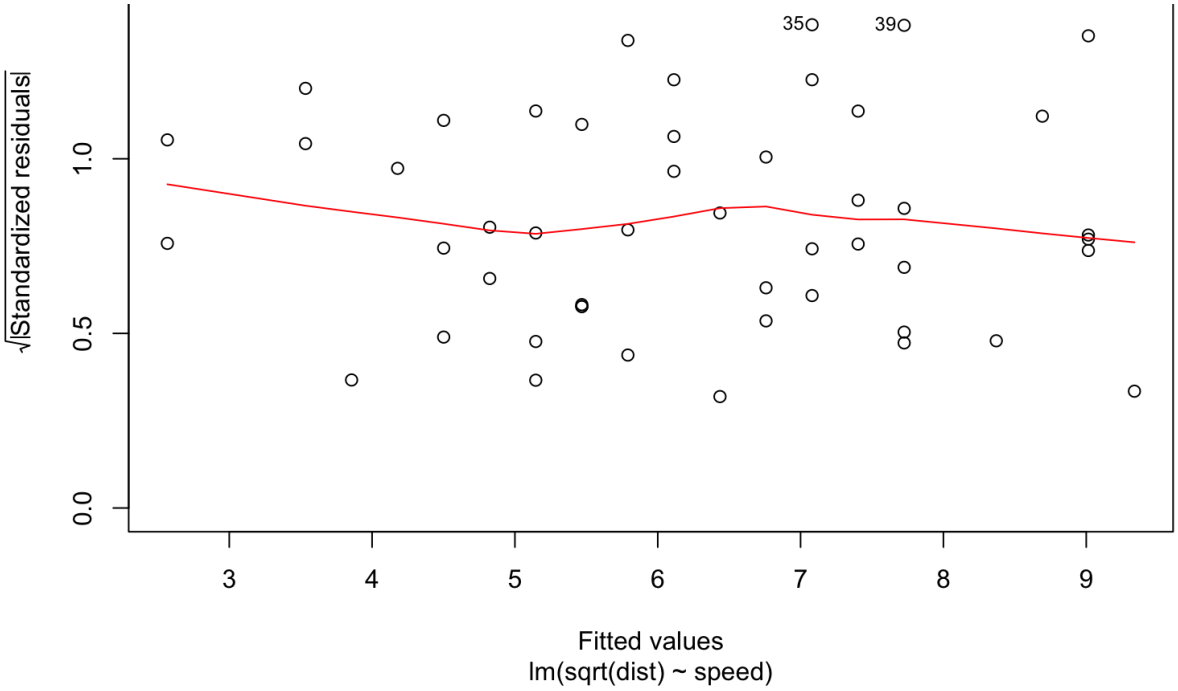
Hide

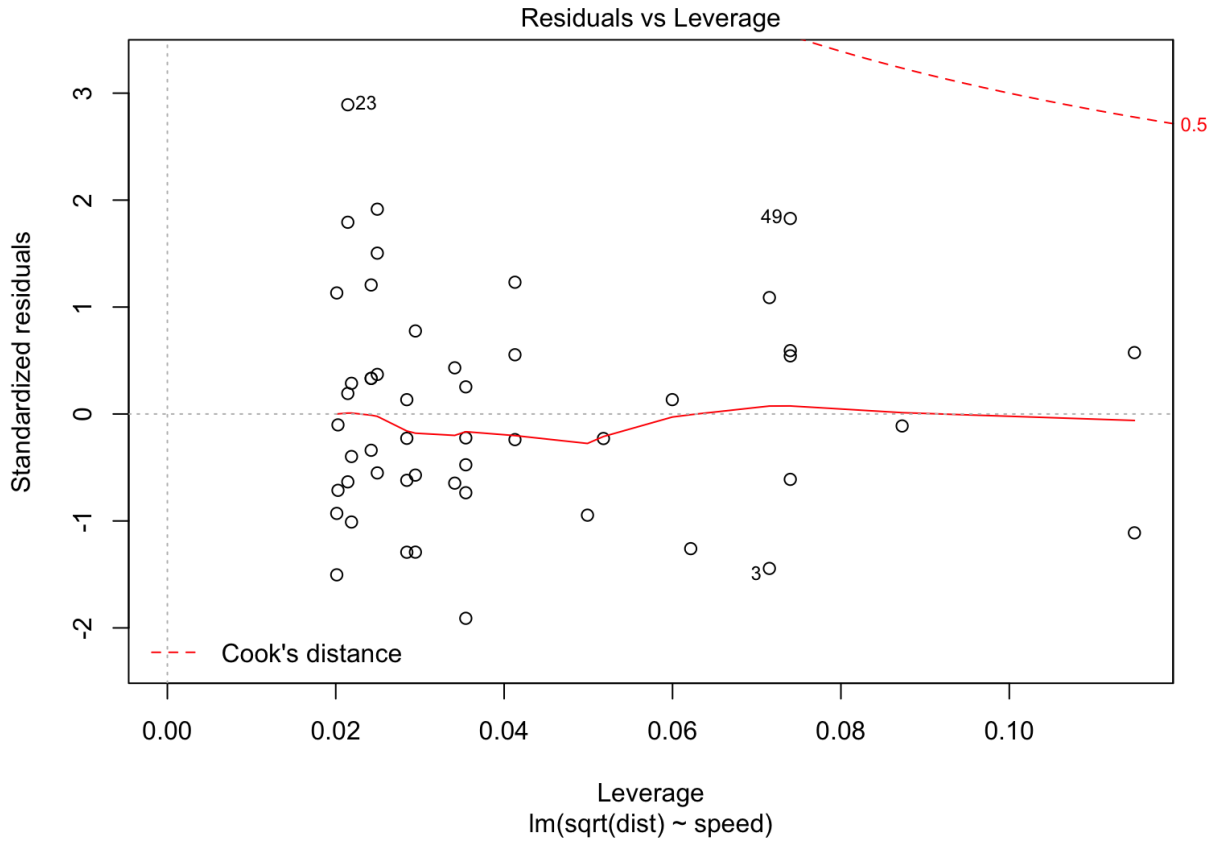
```
plot(m3)
```


Residuals vs Fitted
Residuals
Fitted values
lm(sqrt(dist) ~ speed)
23
35
39


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(sqrt(dist) ~ speed)
23
35
39


Scale-Location
23





Hide

```
summary(m3)
```

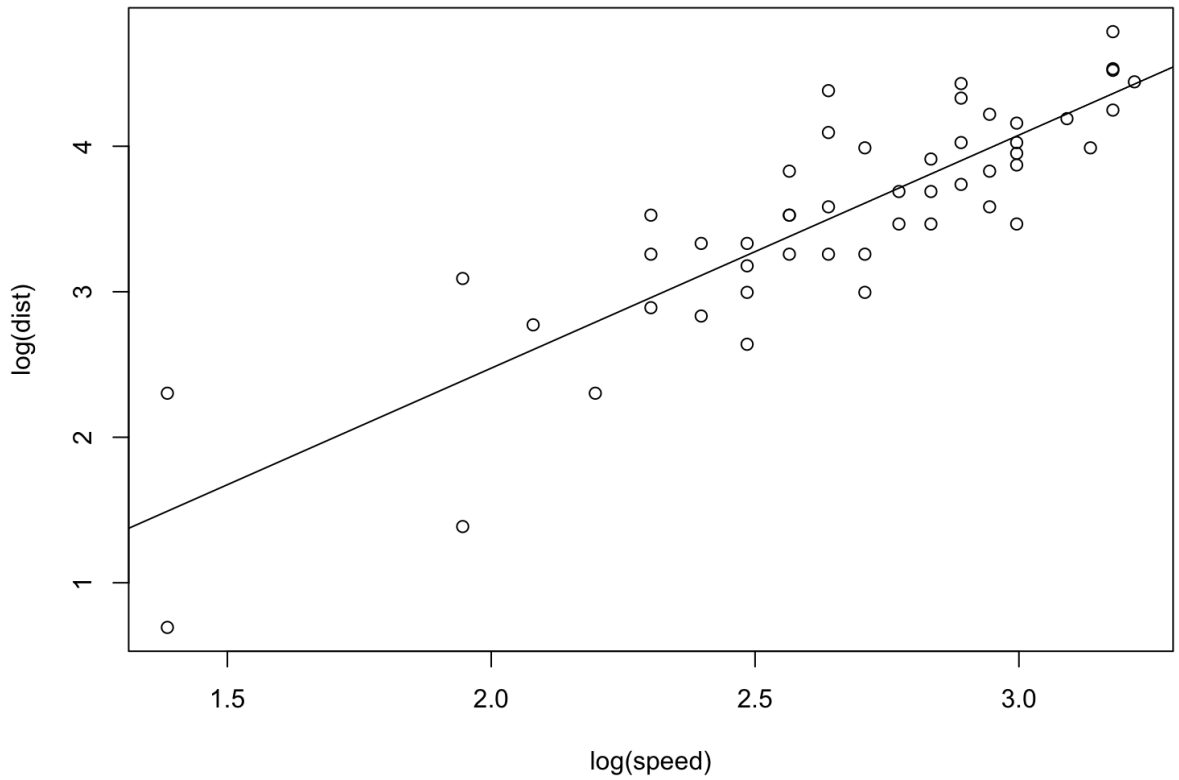
```
##
## Call:
## lm(formula = sqrt(dist) ~ speed, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0684 -0.6983 -0.1799  0.5909  3.1534
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.27705    0.48444   2.636   0.0113 *
## speed        0.32241    0.02978  10.825 1.77e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.102 on 48 degrees of freedom
## Multiple R-squared:  0.7094, Adjusted R-squared:  0.7034
## F-statistic: 117.2 on 1 and 48 DF,  p-value: 1.773e-14
```

The R^2 value is improved over the previous model. Note that again that the intercept term is not significant.

We'll now try a log-log transformation, that is applying a log transformation to the predictor and response variables. This represents a power relationship between the two variables.

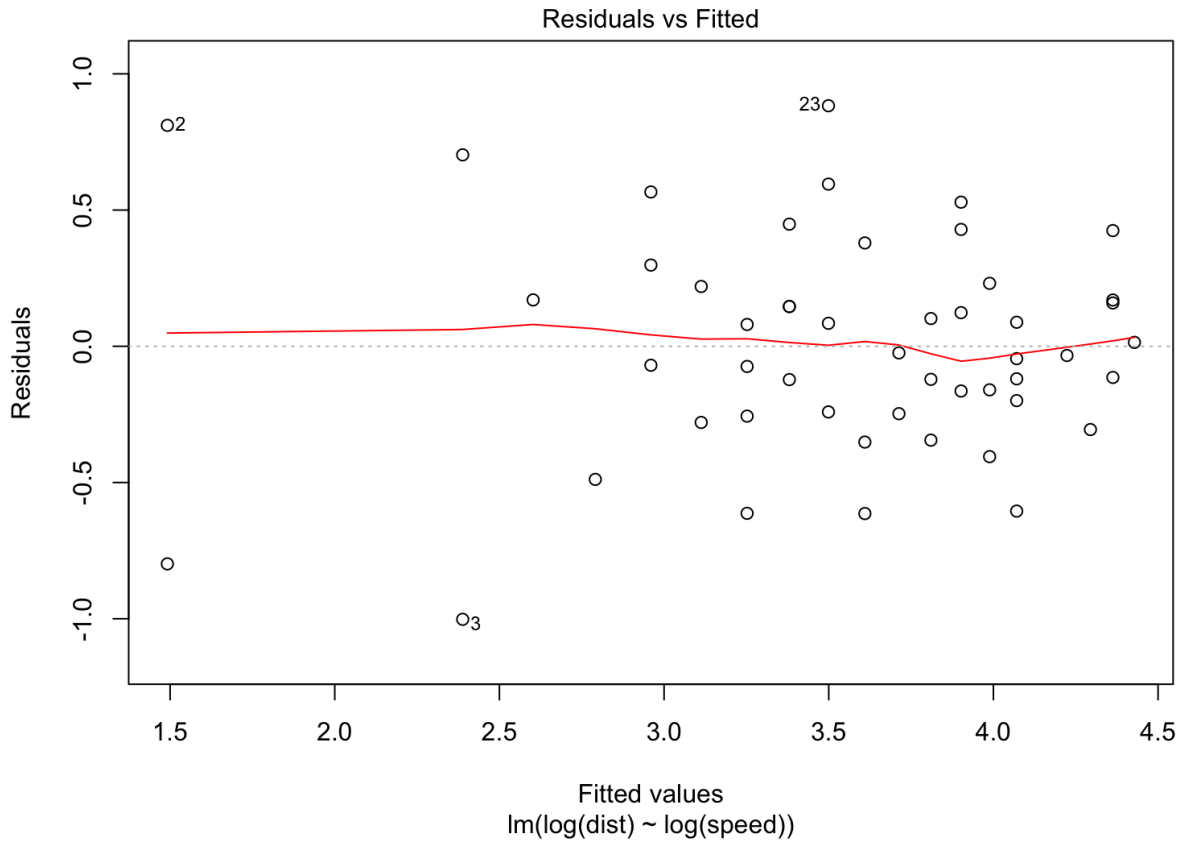
Hide

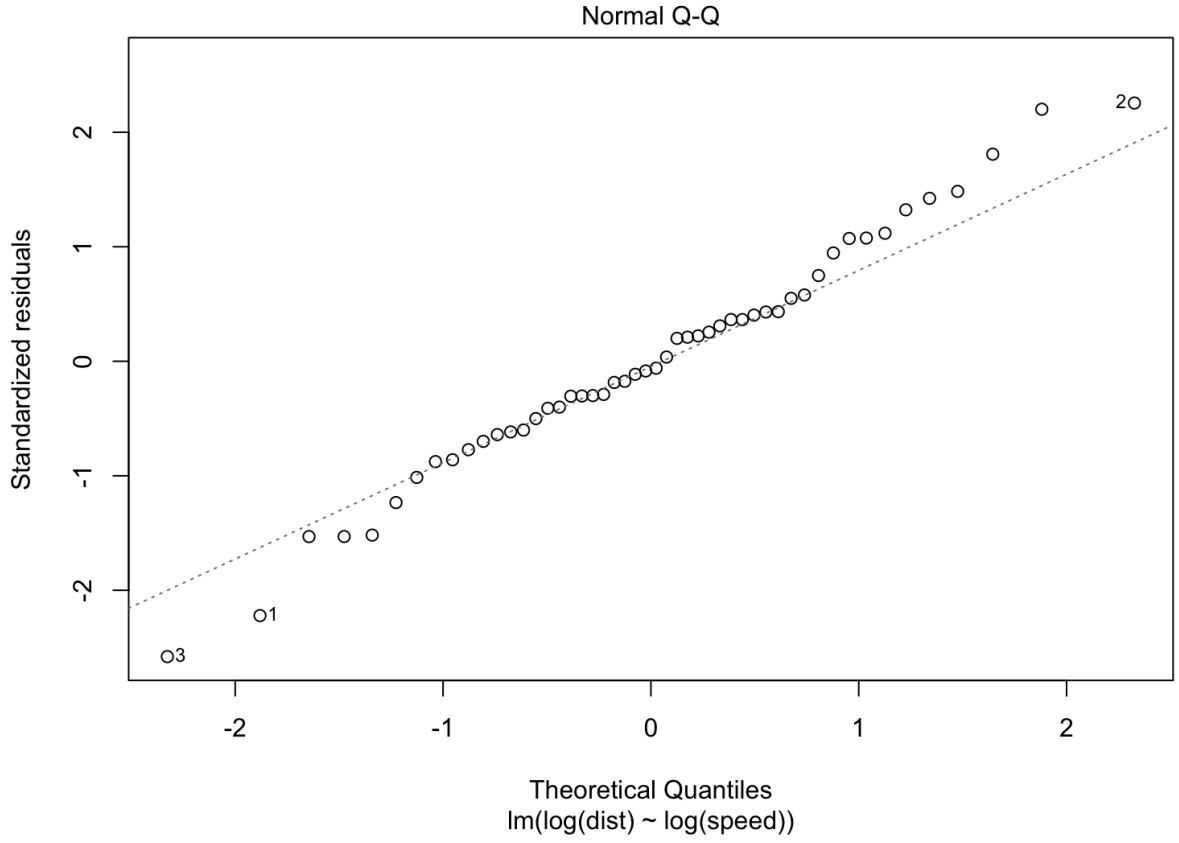
```
plot(log(dist)~log(speed),data=cars)
m4 = lm(log(dist)~log(speed),data=cars)
abline(m4)
```

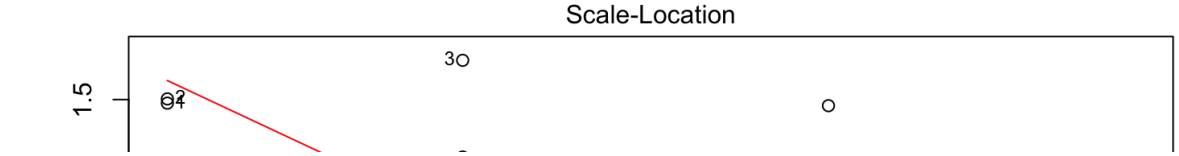


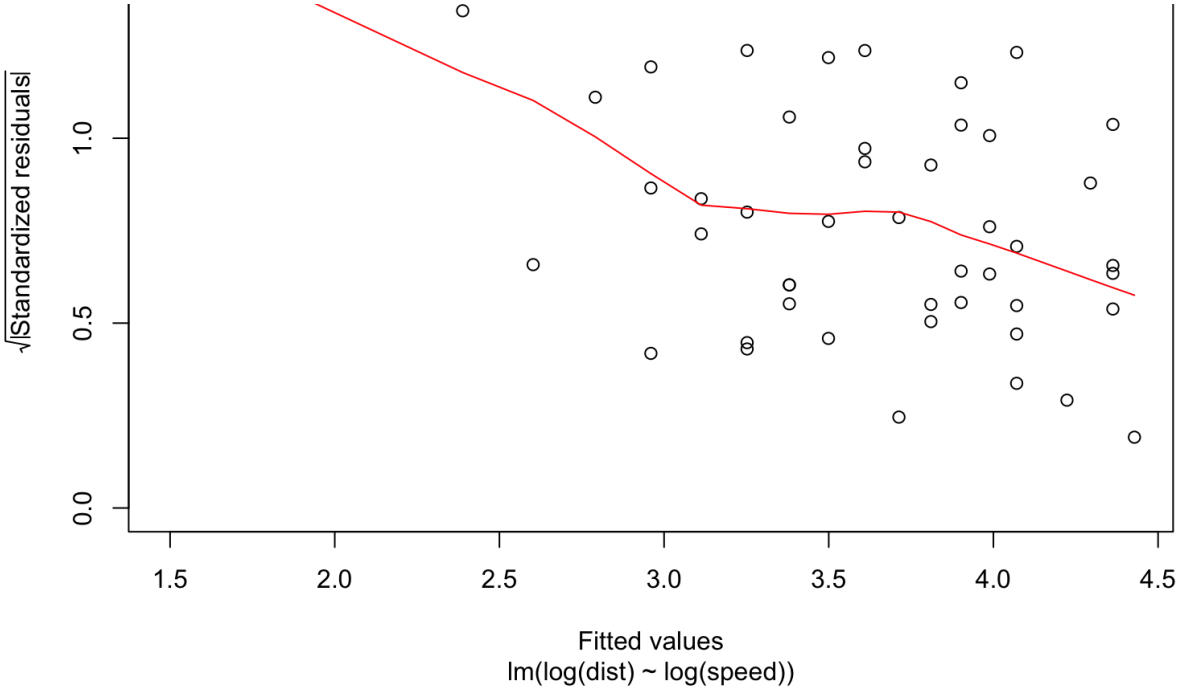
Hide

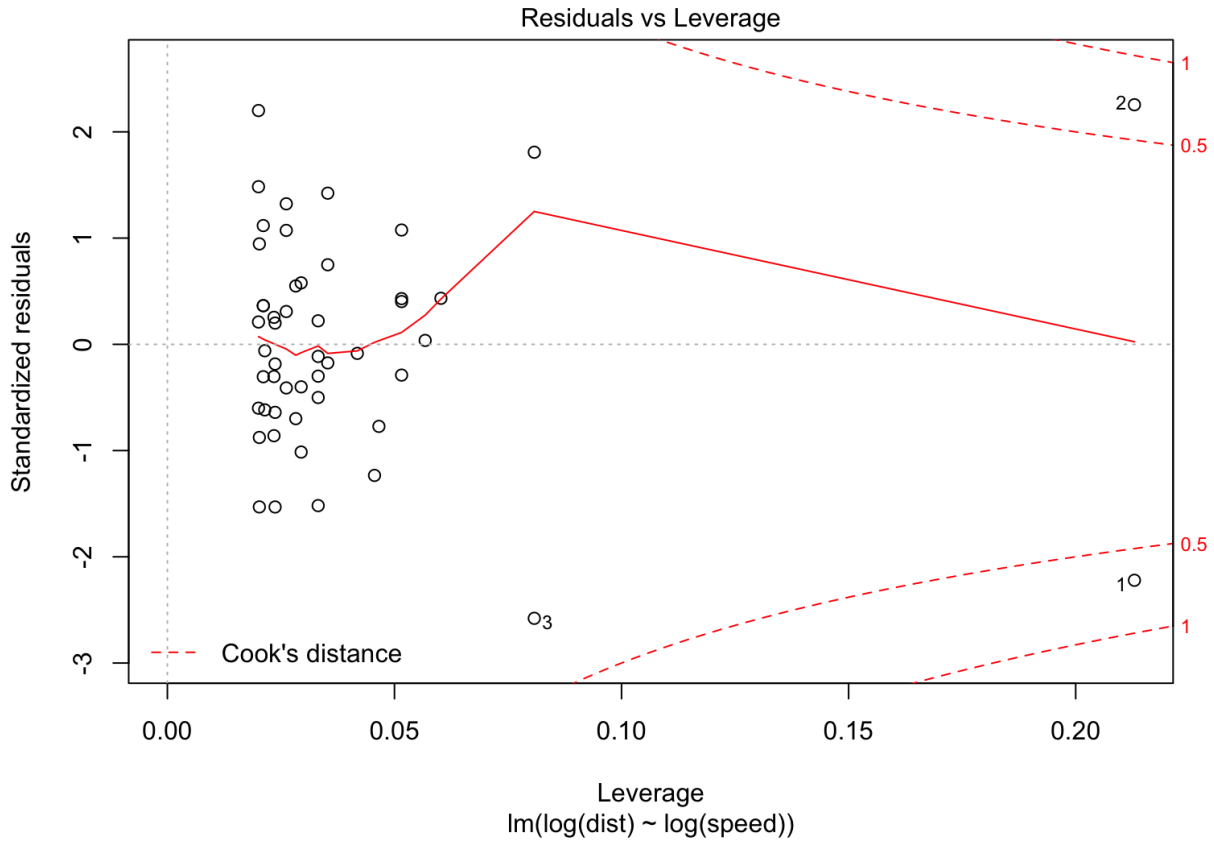
```
plot(m4)
```


Residuals vs Fitted
Residuals
Fitted values
lm(log(dist) ~ log(speed))
23
2
3


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(log(dist) ~ log(speed))
2
1
3


Scale-Location
3





Hide

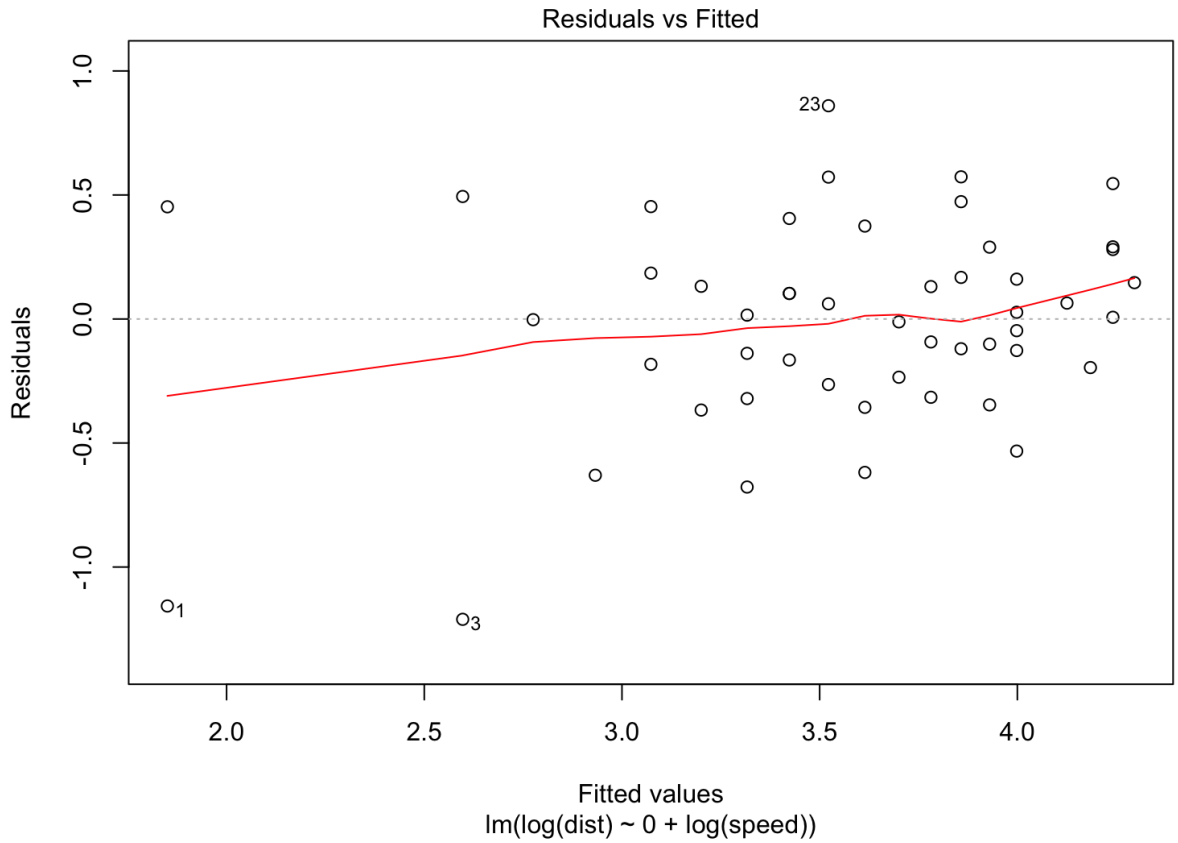
```
summary(m4)
```

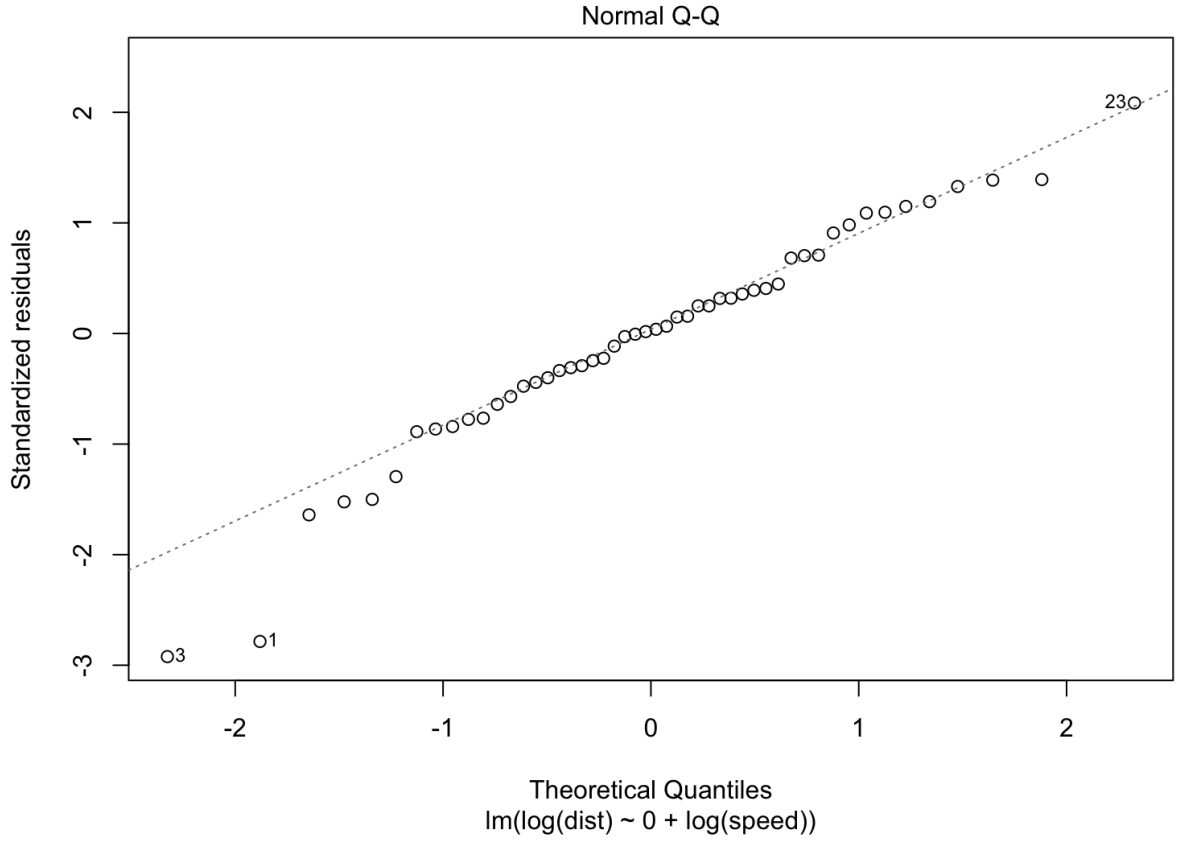
```
##
## Call:
## lm(formula = log(dist) ~ log(speed), data = cars)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.00215 -0.24578 -0.02898  0.20717  0.88289
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.7297     0.3758  -1.941   0.0581 .
## log(speed)    1.6024     0.1395  11.484 2.26e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4053 on 48 degrees of freedom
## Multiple R-squared:  0.7331, Adjusted R-squared:  0.7276
## F-statistic: 131.9 on 1 and 48 DF,  p-value: 2.259e-15
```

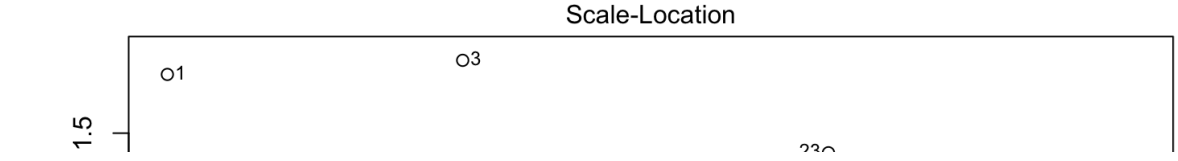
The R^2 value is improved, and the diagnostic plots don't look too unreasonable. However, again the intercept term does not have significant utility. So we'll now remove it from the model:

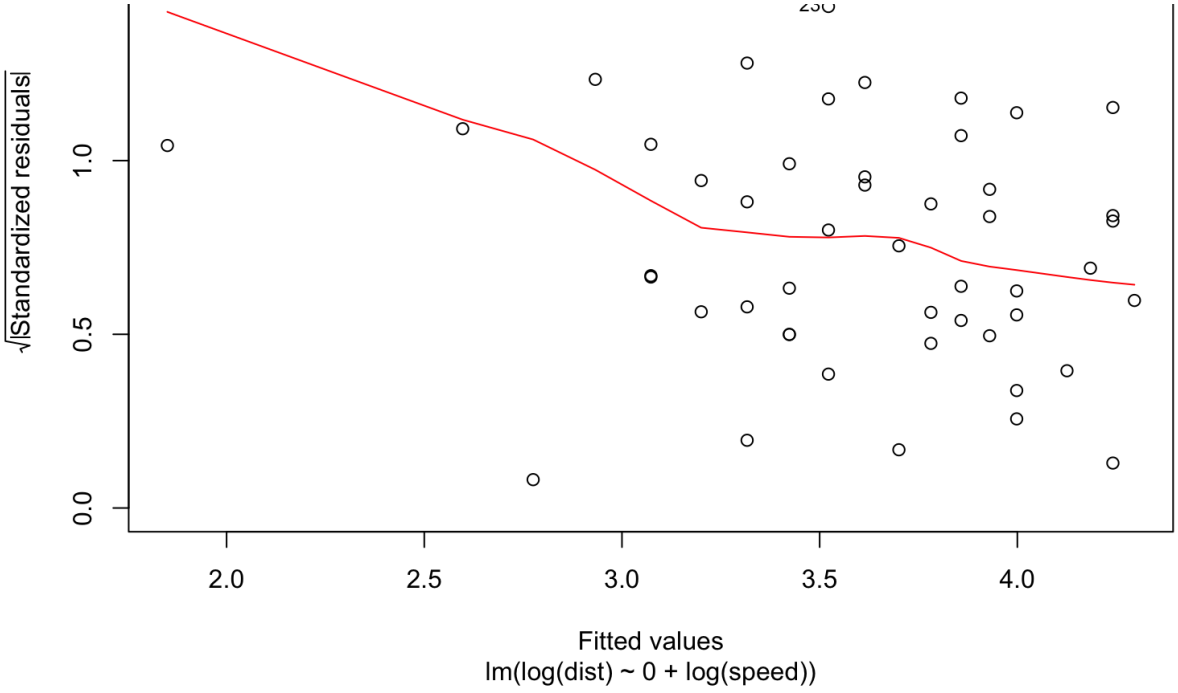
Hide

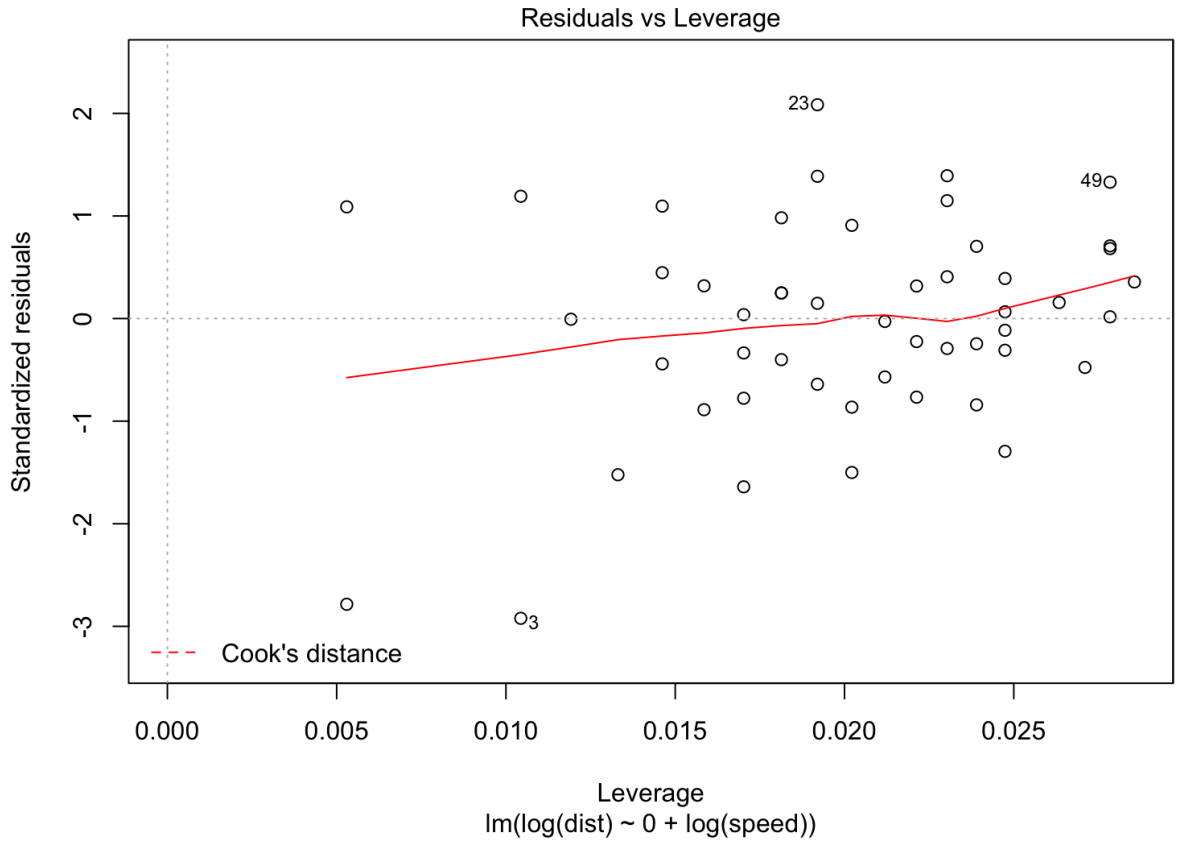
```
m5 = lm(log(dist)~0+log(speed),data=cars)
plot(m5)
```


Residuals vs Fitted
Residuals
Fitted values
lm(log(dist) ~ 0 + log(speed))
23
1
3
2.0
2.5
3.0
3.5
4.0
1.0
0.5
0.0
-0.5
-1.0


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(log(dist) ~ 0 + log(speed))
23
1
3
-2
-1
0
1
2


Scale-Location
1
3
1.5





Hide

```
summary(m5)
```

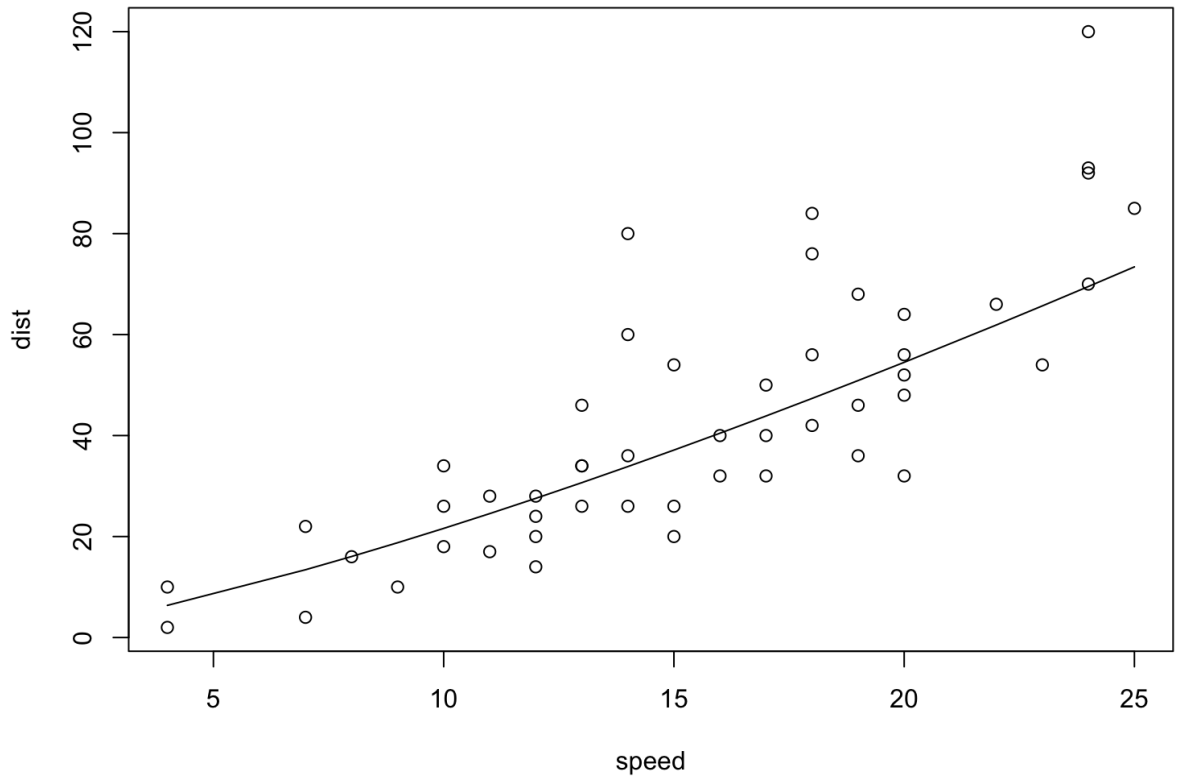
```
##
## Call:
## lm(formula = log(dist) ~ 0 + log(speed), data = cars)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.21083 -0.22501  0.01129  0.25636  0.85978
##
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## log(speed)  1.33466    0.02187   61.02   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4166 on 49 degrees of freedom
## Multiple R-squared:  0.987,  Adjusted R-squared:  0.9867
## F-statistic:  3724 on 1 and 49 DF,  p-value: < 2.2e-16
```

This model seems reasonable. However, remember that R^2 values corresponding to models without an intercept aren't meaningful (or at least can't be compared against models with an intercept term).

We can now transform the model back, and display the regression curve on the plot:

Hide

```
plot(dist~speed,data=cars)
x = order(cars$speed)
lines(exp(fitted(m5))[x]~cars$speed[x])
```



Section 5: Relationship between the t-test, ANOVA and linear regression

In the ANOVA session we looked at the `diet` dataset, and performed the t-test and ANOVA. Here's a recap:

Hide

```
# import
diet = read.csv("data/diet.csv",row.names=1)
diet$weight.loss = diet$initial.weight - diet$final.weight
# comparison
t.test(weight.loss~diet.type,data=diet[diet$diet.type!="B",],var.equal = TRUE)
```

```
##
##  Two Sample t-test
##
## data:  weight.loss by diet.type
## t = -2.8348, df = 49, p-value = 0.006644
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.1582988 -0.5379975
## sample estimates:
## mean in group A mean in group C
##        3.300000        5.148148
```

Hide

```
summary(aov(weight.loss~diet.type,data=diet[diet$diet.type!="B",]))
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## diet.type    1   43.4    43.4   8.036 0.00664 **
## Residuals   49  264.6     5.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that the p-values for both the t-test and ANOVA are the same. This is because these tests are equivalent (in the 2-sample case). They both test the same hypothesis.

Also, the F-test statistic is equal to the square of the t-test statistic (-2.8348^2 = 8.036). Again, this is only true for the 2-sample case.

Now let's use a different strategy. Instead of directly testing whether there is a difference between the two groups, let's attempt to create a linear model describing the relationship between `weight.loss` and `diet.type`. Indeed, it is possible to construct a linear model where the independent variable(s) are categorical - they do not have to be continuous or even ordinal!

Hide

```
summary(lm(weight.loss~diet.type,data=diet[diet$diet.type!="B",]))
```

```
##
## Call:
## lm(formula = weight.loss ~ diet.type, data = diet[diet$diet.type !=
##     "B", ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6481 -1.5241  0.1519  1.6519  5.7000
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.3000     0.4744   6.957 7.74e-09 ***
## diet.typeC    1.8481     0.6520   2.835  0.00664 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.324 on 49 degrees of freedom
## Multiple R-squared:  0.1409, Adjusted R-squared:  0.1234
## F-statistic: 8.036 on 1 and 49 DF,  p-value: 0.006644
```

You can see that the p-value corresponding to the `diet.type` term is the same as the overall p-value of the linear model, which is also the same as the p-value from the t-test and ANOVA. Note also that the F-test statistic is the same as given by the ANOVA.

So, we are also able to use the linear model to test the hypothesis that there is a difference between the two diet groups, as well as provide a more detailed description of the relationship between `weight.loss` and `diet.type`.

Section 6: Practical Exercises

Old Faithful

The inbuilt R dataset `faithful` pertains to the waiting time between eruptions and the duration of the eruption for the Old Faithful geyser in Yellowstone National Park, Wyoming, USA.

- Create a simple linear regression model that models the eruption duration `faithful$eruptions` using waiting time `faithful$waiting` as the independent variable, storing the model in a variable. Look at the summary of the model.
    - What are the values of the estimates of the intercept and coefficient of 'waiting'?
    - What is the R^2 value?
    - Does the model have significant utility?
    - Are neither, one, or both of the parameters significantly different from zero?
    - Can you conclude that there is a linear relationship between the two variables?
- Plot the eruption duration against waiting time. Is there anything noticeable about the data?
- Draw the regression line corresponding to your model onto the plot. Based on this graphical representation, does the model seem reasonable?
- Generate the four diagnostic plots corresponding to your model. Contemplate the appropriateness of the model for describing the relationship between eruption duration and waiting time.

Anscombe datasets

Consider the inbuilt R dataset `anscombe`. This dataset contains four x-y datasets, contained in the columns: (x1,y1), (x2,y2), (x3,y3) and (x4,y4).

- For each of the four datasets, calculate and test the correlation between the x and y variables. What do you conclude?
- For each of the four datasets, create a linear model that regresses y on x. Look at the summaries corresponding to these models. What do you conclude?
- For each of the four datasets, create a plot of y against x. What do you conclude?

Pharmacokinetics of Indomethacin

Consider the inbuilt R dataset `Indometh`, which contains data on the pharmacokinetics of indometacin.

- Plot `Indometh$time` versus `Indometh$conc` (concentration). What is the nature of the relationship between `time` and `conc`?
- Apply monotonic transformations to the data so that a simple linear regression model can be used to model the relationship (ensure both linearity and stabilised variance, within reason). Create a plot of the transformed data, to confirm that the relationship seems linear.
- After creating the linear model, inspect the diagnostic plots to ensure that the assumptions are not violated (too much). Are there any outliers with large influence? What are the parameter estimates? Are both terms significant?
- Add a line to the plot showing the linear relationship between the transformed data.
- Now regenerate the original plot of `time` versus `conc` (i.e. the untransformed data). Using the `lines` function, add a curve to the plot corresponding to the fitted values of the model.