Simple Linear Models with R

Alastair Kerr, based on a CRUK workshop: (D.-L. Couturier / M. Dunning / R. Nicholls) and the minitab manual)

Last modified: 05 Mar 2018

Contents

Section 1: Correlation Coefficients

A correlation coefficient measures the extent to which two variables tend to change together. The two key tests in the class are *Pearson product moment correlation* and *Spearman rank-order correlation*.

Pearson product moment correlation

The Pearson correlation (**P** or **cor**) evaluates the linear relationship between *two continuous variables*. A relationship is *linear* when a change in one variable is associated with a *proportional* change in the other variable.

For example, you might use a Pearson correlation to evaluate if weight changes with calorie intake.

Spearman rank-order correlation

The Spearman correlation (**S** or **rho**) evaluates the *monotonic* relationship between two continuous or ordinal variables. In a monotonic relationship, the variables tend to change together, but *not necessarily at a constant rate*. The Spearman correlation coefficient is based on the ranked values for each variable rather than the raw data.

Spearman correlation is often used to evaluate relationships involving ordinal variables which are ordered catagoriess. e.g. giving the value 1, 2 & 3 for low, medium & high.

Which to use?

Each is trivial to compute in R, and the difference between S and P can be interesting. i.e. S >> P represents a correlation that is monotonic but not linear. Note that in both cases a positive value near 1 indicates a strong relationship and a negative value approaching -1 indicates a strong inverse relationship.

Examples

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

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

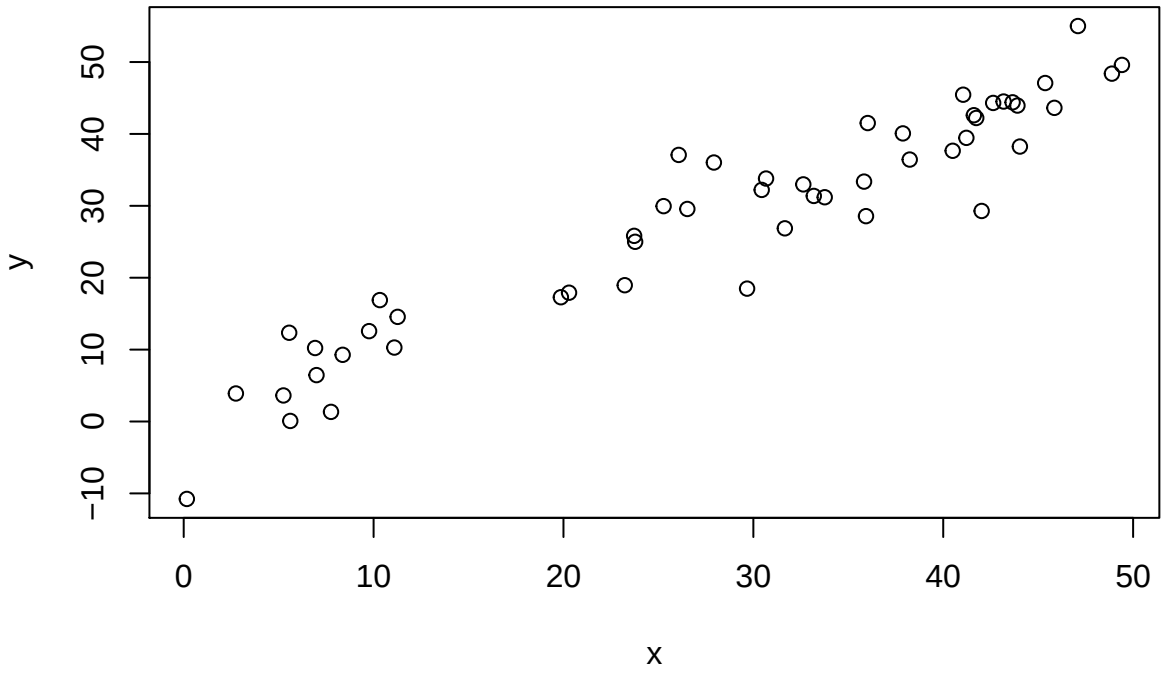
```
x = runif(50,0,50)
```

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):

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

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

```
plot(x,y)
```



This strong linear relationship is reflected in the correlation coefficient. The significance of a correlation can be tested using `cor.test()`, which also provides a 95% confidence interval on the correlation

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

```
##
##  Pearson's product-moment correlation
##
## data:  x and y
## t = 21.171, df = 48, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```

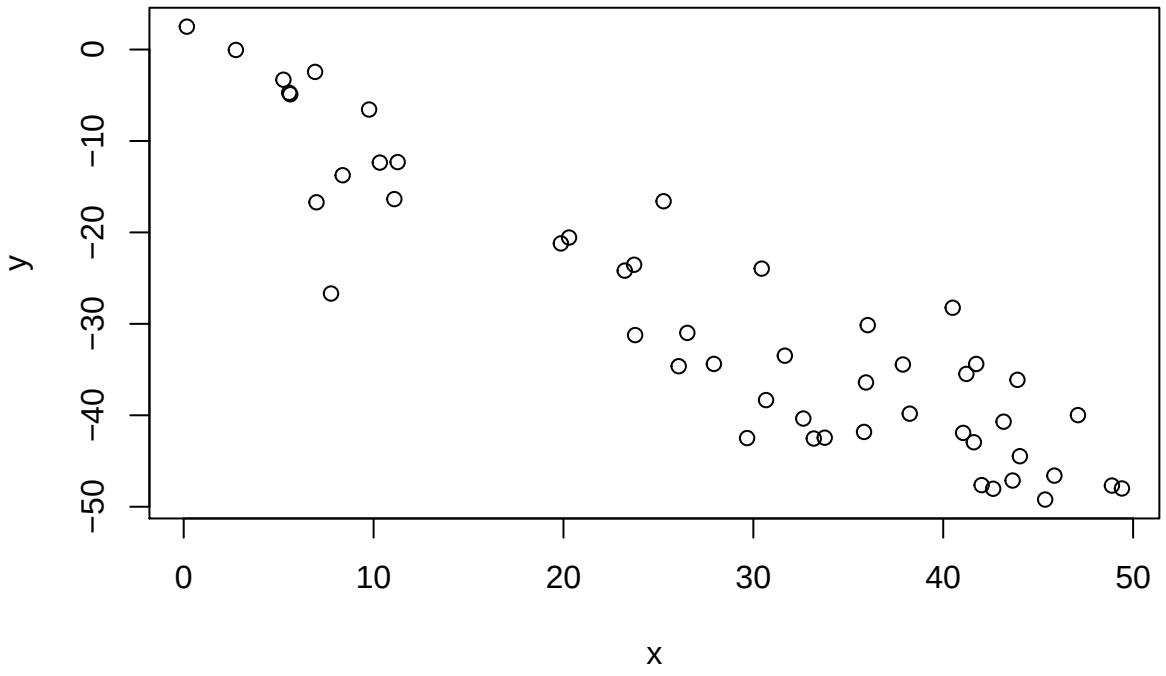
```
##  0.9137906 0.9716953
## sample estimates:
##       cor
## 0.9504021
```

```
cor.test(x,y, method="spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  x and y
## S = 1276, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9387275
```

If the data exhibit a negative linear correlation then the correlation coefficient will become strong and negative

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




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

```
##
##  Pearson's product-moment correlation
##
## data:  x and y
## t = -16.038, df = 48, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.9528696 -0.8592021
## sample estimates:
##       cor
```

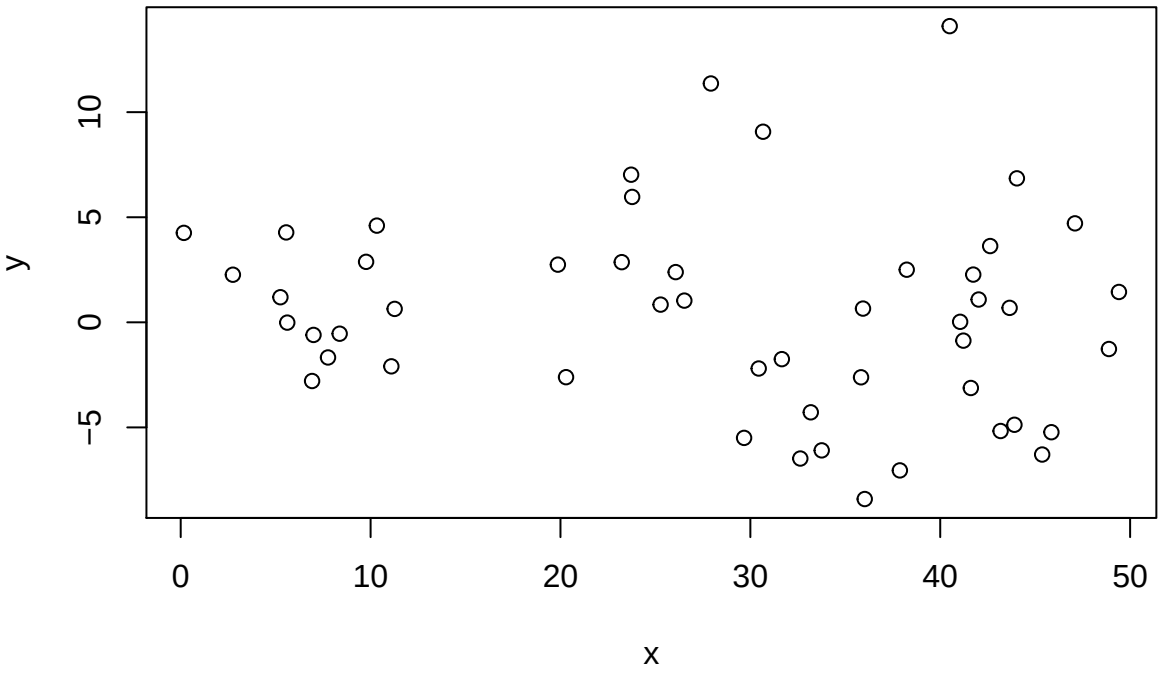
```
## -0.918003
```

```
cor.test(x,y, method="spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  x and y
## S = 39308, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## -0.887539
```

If data are uncorrelated then both the correlation coefficients will be close to zero:

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




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

```
##
##  Pearson's product-moment correlation
##
## data:  x and y
## t = -1.0476, df = 48, p-value = 0.3001
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.4107632  0.1344326
## sample estimates:
##        cor
## -0.1495097
```

```
cor.test(x,y, method="spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  x and y
## S = 24832, p-value = 0.1802
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## -0.192413
```

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

Practical 1 : 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).

```
data("anscombe")
anscombe
```

```
##    x1 x2 x3 x4    y1   y2    y3    y4
## 1  10 10 10  8  8.04 9.14  7.46  6.58
## 2   8  8  8  8  6.95 8.14  6.77  5.76
## 3  13 13 13  8  7.58 8.74 12.74  7.71
## 4   9  9  9  8  8.81 8.77  7.11  8.84
## 5  11 11 11  8  8.33 9.26  7.81  8.47
## 6  14 14 14  8  9.96 8.10  8.84  7.04
## 7   6  6  6  8  7.24 6.13  6.08  5.25
## 8   4  4  4 19  4.26 3.10  5.39 12.50
## 9  12 12 12  8 10.84 9.13  8.15  5.56
## 10  7  7  7  8  4.82 7.26  6.42  7.91
## 11  5  5  5  8  5.68 4.74  5.73  6.89
```

- For each of the four datasets, calculate and test the correlation between the x and y variables using pearson. What do you conclude?
- Now use spearman methods for correlation. How is it different in each case?
- For each of the four datasets, create a plot of y against x. What do you conclude?

Hint: you can use "attach(anscombe)" to avoid having to type the name of the data frame each time

```
cor.test(anscombe$x1, anscombe$y1)
#or
attach(anscombe)
cor.test(x1, y1)
cor.test(x2,y2)
```

Section 2: Simple Regression

The aim of linear regression is to model a continuous variable Y as a mathematical function of one or more X variable(s), so that we can use this regression model to predict the Y when only the X is known. This mathematical equation can be generalized as follows:

Y = 1 + 2X +

where, 1 is the intercept and 2 is the slope. Collectively, they are called regression coefficients. is the error term, the part of Y the regression model is unable to explain.

In R the simplest way to use such a model is lm(), the linear model function. The 1st argument is the formula for the model, described using the tidle sign "~". Left of the sign is the response variable y. Right of the sign is one or more variables which you wish to model. So the simplest model is

```
model = lm(y~x)
```

However, more complicated models are possible such as adding extra coefficients and relationships between them

- − for an additional coefficient as in A+B
- : for interactions, as in A:B
- * for both main effects and interactions, so A*B = A + B + A:B
- "y ~ x -1" a line through the origin
- "y ~ 0 + x" a model with no intercept

More details on the formula help page (?formula).

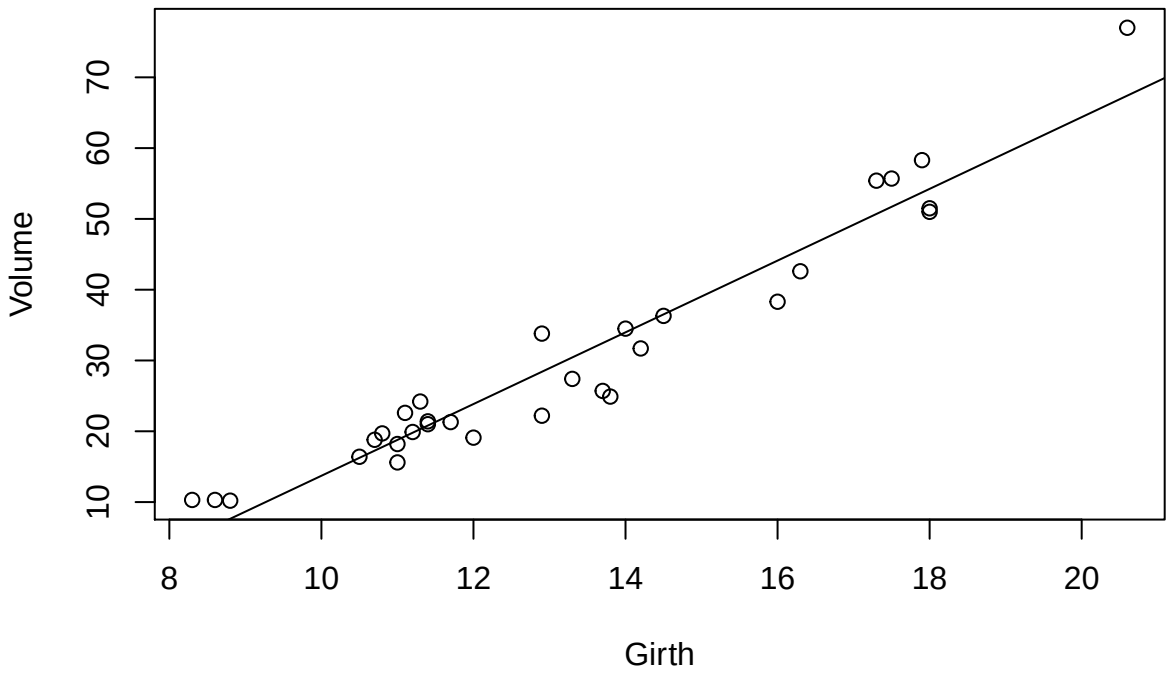
Examine using tree data

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.

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



```r
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:

```r
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
```

There are several values stored here we need to be aware of: F-statistics, R-squared, Residuals,

What is the F-statistic

In general, an F-test in regression compares the fits of different linear models. Unlike t-tests that can assess only one regression coefficient at a time, the F-test can assess multiple coefficients simultaneously.

The F-test of the overall significance is a specific form of the F-test. It compares a model with no predictors to the model that you specify. A regression model that contains no predictors is also known as an intercept-only model.

What Is R-squared?

The more variance that is accounted for by the regression model the closer the data points will fall to the fitted regression line. Theoretically, if a model could explain 100% of the variance, the fitted values would always equal the observed values and, therefore, all the data points would fall on the fitted regression line.

R-squared is a statistical measure of how close the data are to the fitted regression line. It is also known as the coefficient of determination, or the coefficient of multiple determination for multiple regression.

The definition of R-squared is fairly straight-forward; it is the fraction of the response variable variation that is explained by a linear model. Or:

- R-squared = Explained variation / Total variation
- R-squared is always between 0 and 1.

In the case above, R^2 is 0.9, indicating that 90% of the variance can be explained using this model,

In some fields, it is entirely expected that your R-squared values will be low. For example, any field that attempts to predict human behavior, such as psychology, typically has R-squared values lower than 50%. Humans are simply harder to predict than, say, physical processes.

Furthermore, if your R-squared value is low but you have statistically significant predictors, you can still draw important conclusions about how changes in the predictor values are associated with changes in the response value. Regardless of the R-squared, the significant coefficients still represent the mean change in the response for one unit of change in the predictor while holding other predictors in the model constant. Obviously, this type of information can be extremely valuable.

A low R-squared is most problematic when you want to produce predictions that are reasonably precise (have a small enough prediction interval). How high should the R-squared be for prediction? Well, that depends on your requirements for the width of a prediction interval and how much variability is present in your data. While a high R-squared is required for precise predictions, it's not sufficient by itself, as we shall see.

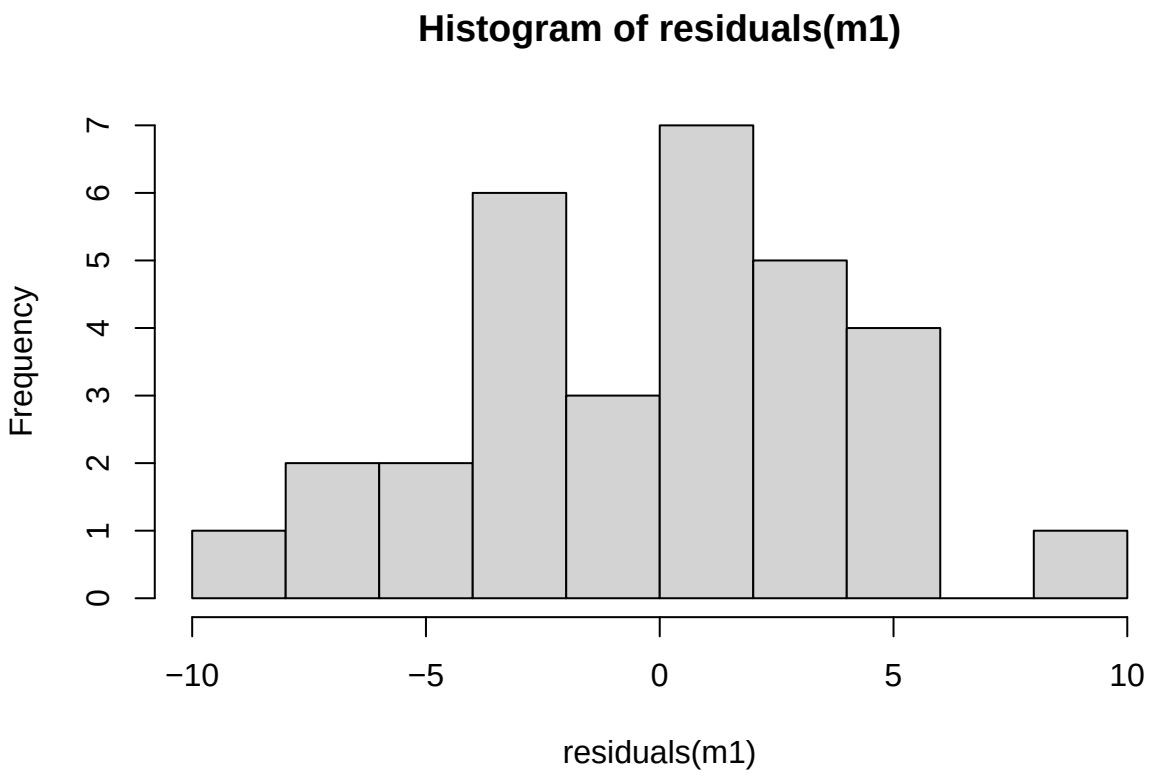
What are Residuals?

R-squared does not indicate whether a regression model is adequate. You can have a low R-squared value for a good model, or a high R-squared value for a model that does not fit the data!

Residuals are the distances along the y axis from each point to the fitted line.

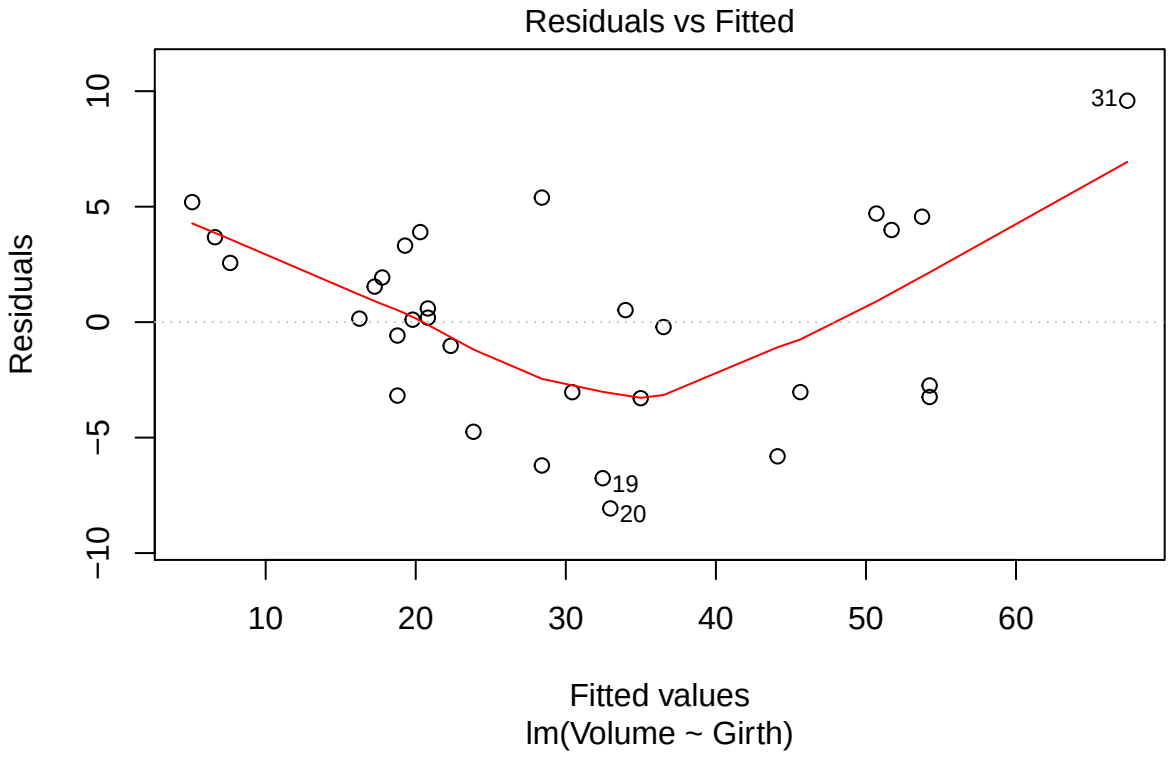
Model residuals can be readily accessed using the `residuals()` function:

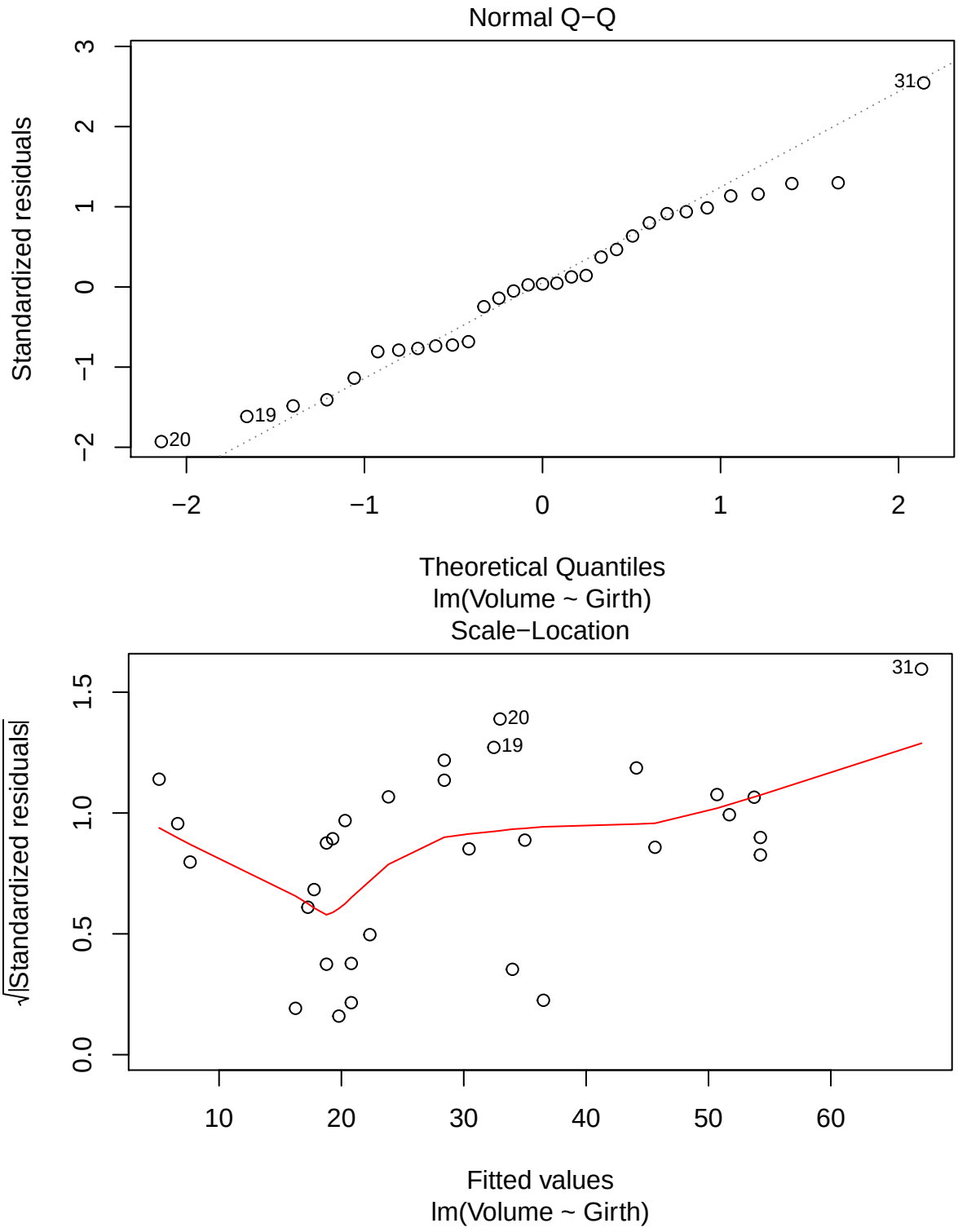
```
hist(residuals(m1),breaks=10,col="light grey")
```

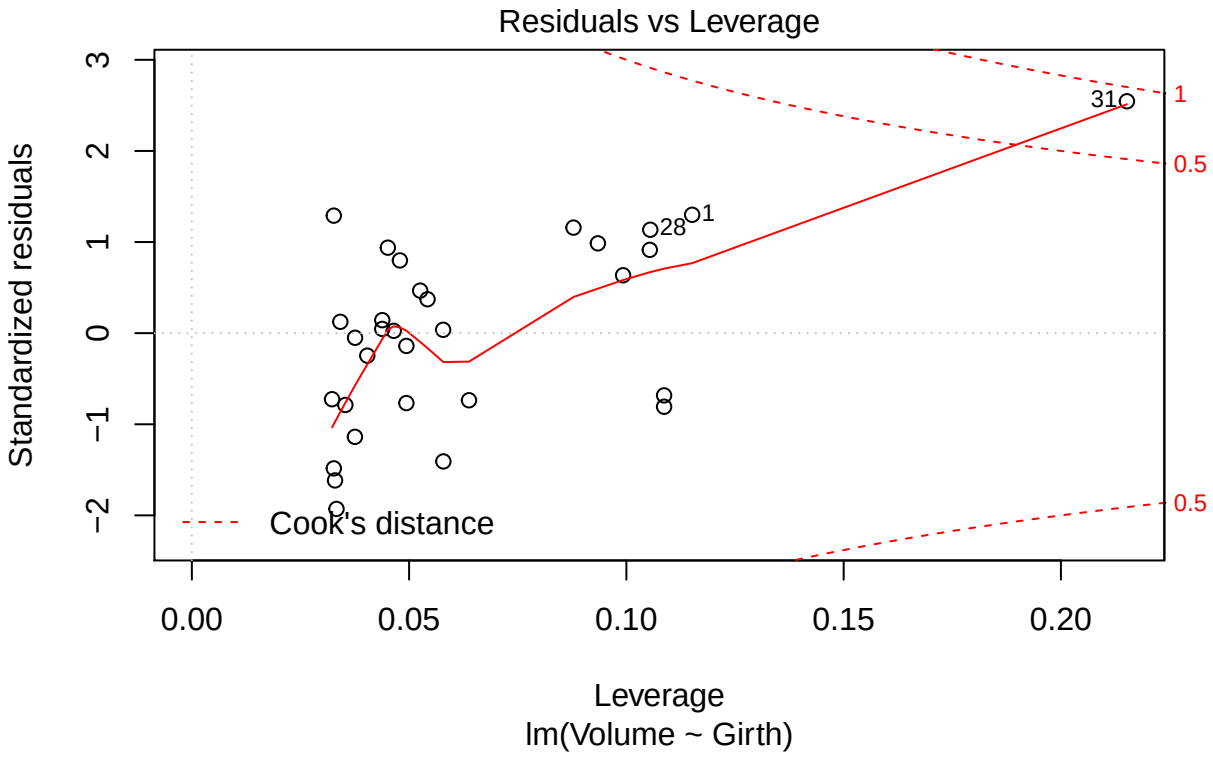


Diagnostic plots for the model can reveal whether or not modelling assumptions are reasonable. In a good model we expect there to be no systematic change when we plot residuals vs fitted. In this case, there is visual evidence to suggest that the assumptions are not satisfied - note in particular the trend observed in the plot of residuals vs fitted values:

```
plot(m1)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Volume ~ Girth)
Scale−Location
√|Standardized residuals|
Fitted values
lm(Volume ~ Girth)
31
20
19



Explanation of diagnostic plots

Information adapted from a University of Virginia blog post.

Residuals vs Fitted

This plot shows if residuals have non-linear patterns. There could be a non-linear relationship between predictor variables and an outcome variable and the pattern could show up in this plot if the model doesn't capture the non-linear relationship. If you find equally spread residuals around a horizontal line without distinct patterns, that is a good indication you don't have non-linear relationships.

Normal Q-Q

This plot shows if residuals are normally distributed. Do residuals follow a straight line well or do they deviate severely? It's good if residuals are lined well on the straight dashed line.

Scale-Location

It's also called Spread-Location plot. This plot shows if residuals are spread equally along the ranges of predictors. This is how you can check the assumption of equal variance (homoscedasticity). It's good if you see a horizontal line with equally (randomly) spread points.

Residuals vs Leverage

This plot helps us to find influential cases (i.e., subjects) if any. Not all outliers are influential in linear regression analysis (whatever outliers mean). Even though data have extreme values, they might not be

influential to determine a regression line. That means, the results wouldn't be much different if we either include or exclude them from analysis. They follow the trend in the majority of cases and they don't really matter; they are not influential. On the other hand, some cases could be very influential even if they look to be within a reasonable range of the values. They could be extreme cases against a regression line and can alter the results if we exclude them from analysis. Another way to put it is that they don't get along with the trend in the majority of the cases.

Unlike the other plots, this time patterns are not relevant. We watch out for outlying values at the upper right corner or at the lower right corner. Those spots are the places where cases can be influential against a regression line. Look for cases outside of a dashed line, Cook's distance. When cases are outside of the Cook's distance (meaning they have high Cook's distance scores), the cases are influential to the regression results. The regression results will be altered if we exclude those cases.

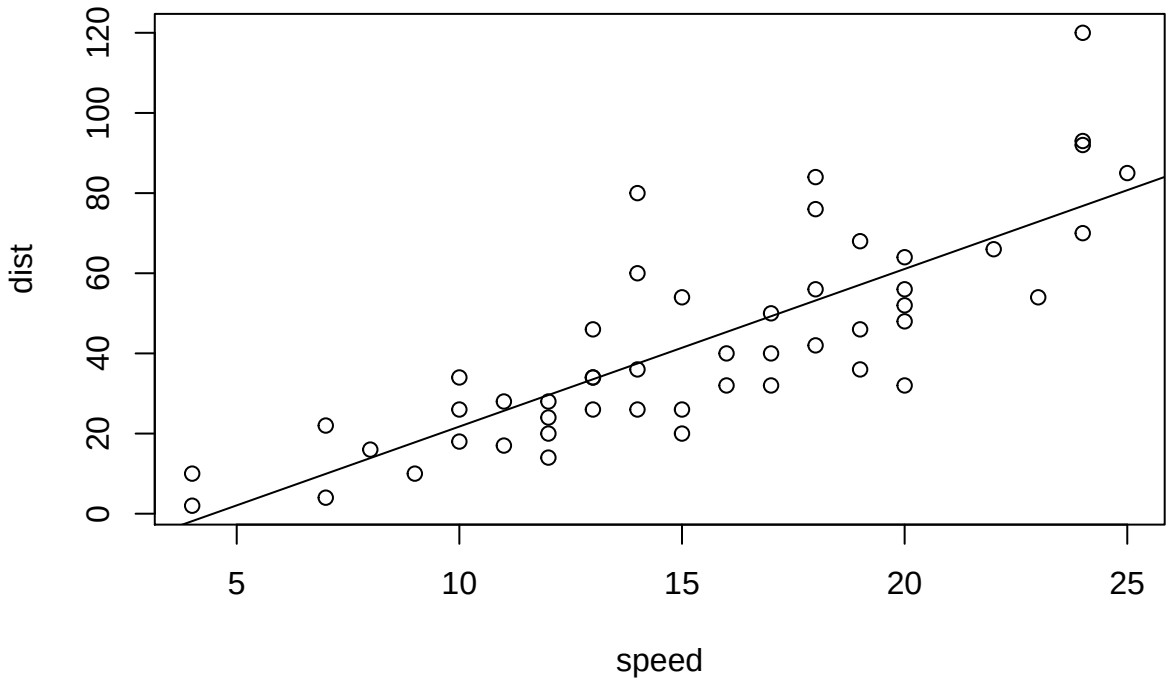
Section 3: Modelling Non-Linear Relationships

Perhaps counterintuitively, it is possible to use linear models to model non-linear relationships.

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:

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




```
summary(m2)
```

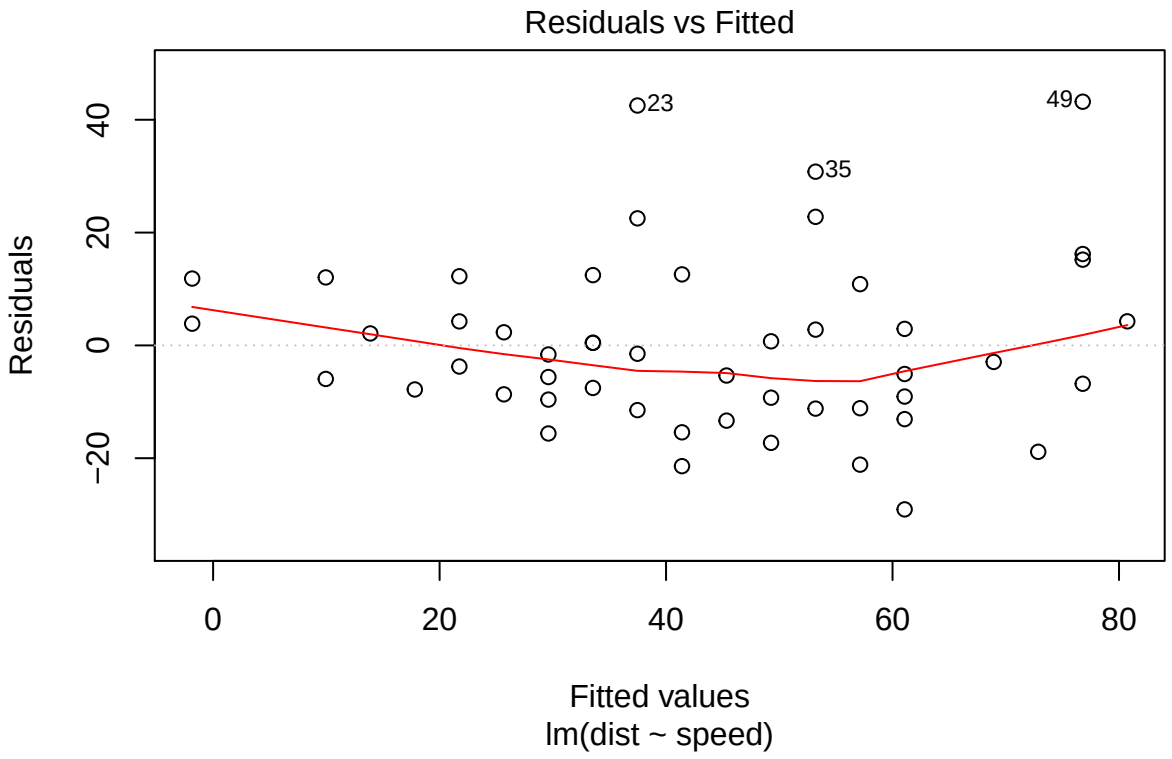
```
##
## Call:
## lm(formula = dist ~ speed, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

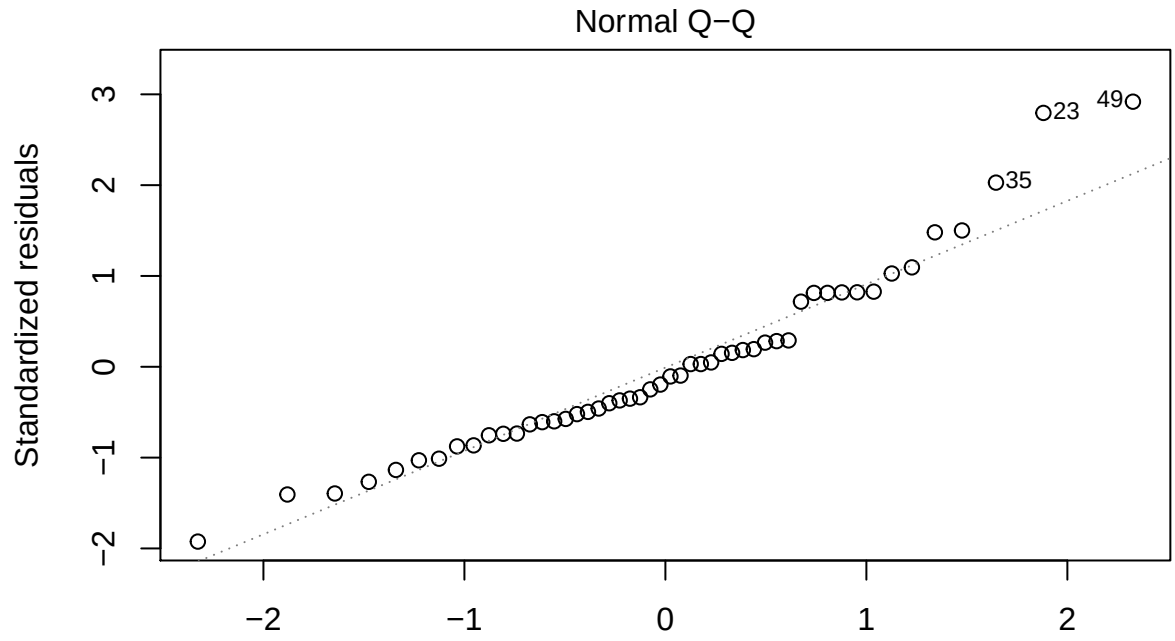
```
## -29.069  -9.525  -2.272   9.215  43.201
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.5791     6.7584  -2.601   0.0123 *
## speed         3.9324     0.4155   9.464 1.49e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## 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
```

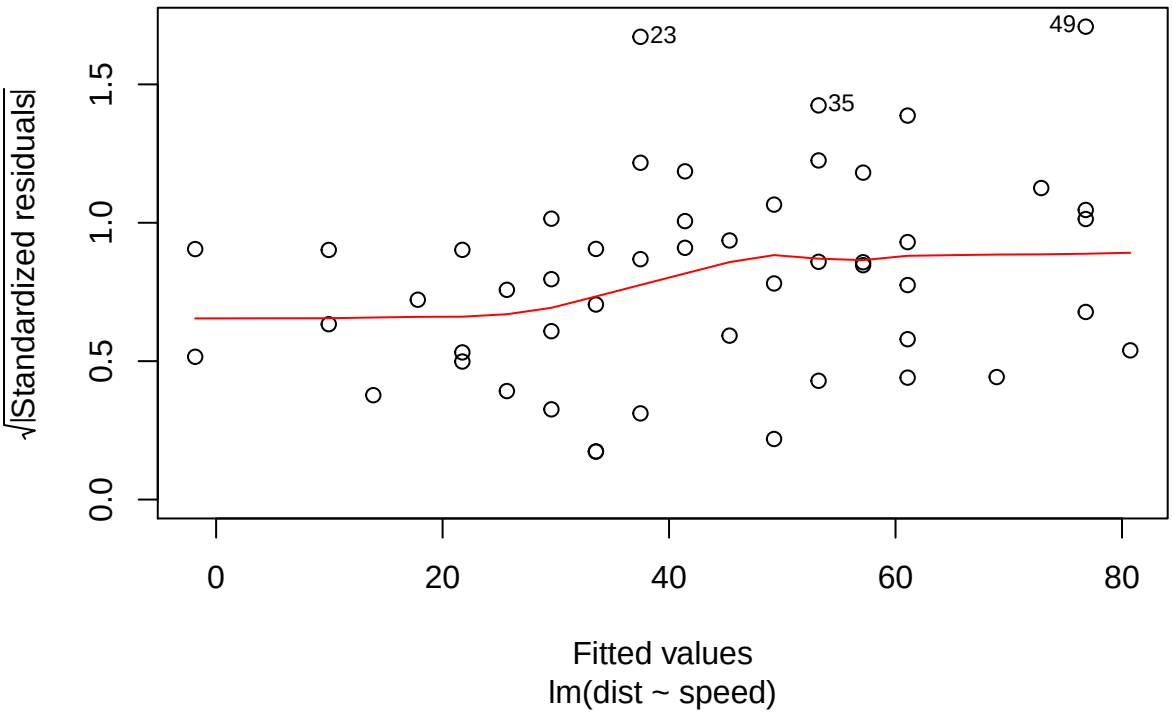
The model summary indicates that the intercept term does not have significant utility. So that term could/should be removed from the model.

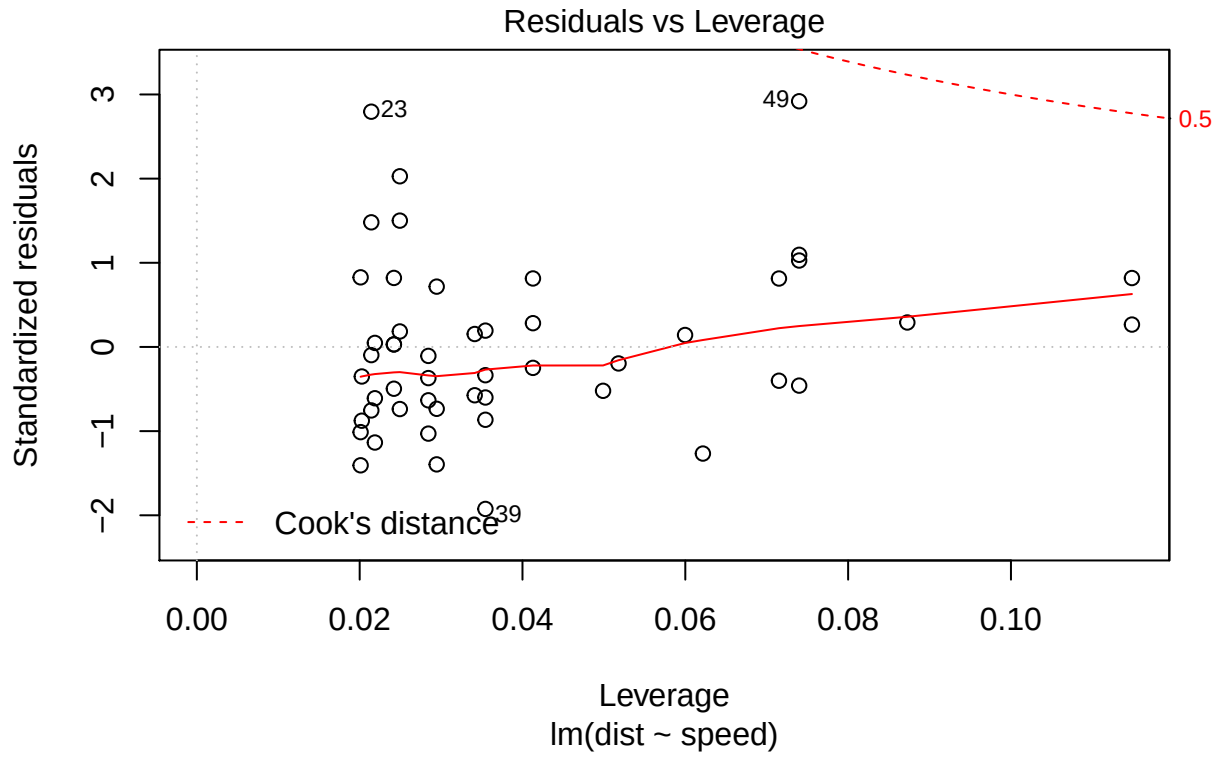
In addition, the plot of residuals versus fitted values indicates potential issues with variance stability:

```
plot(m2)
```



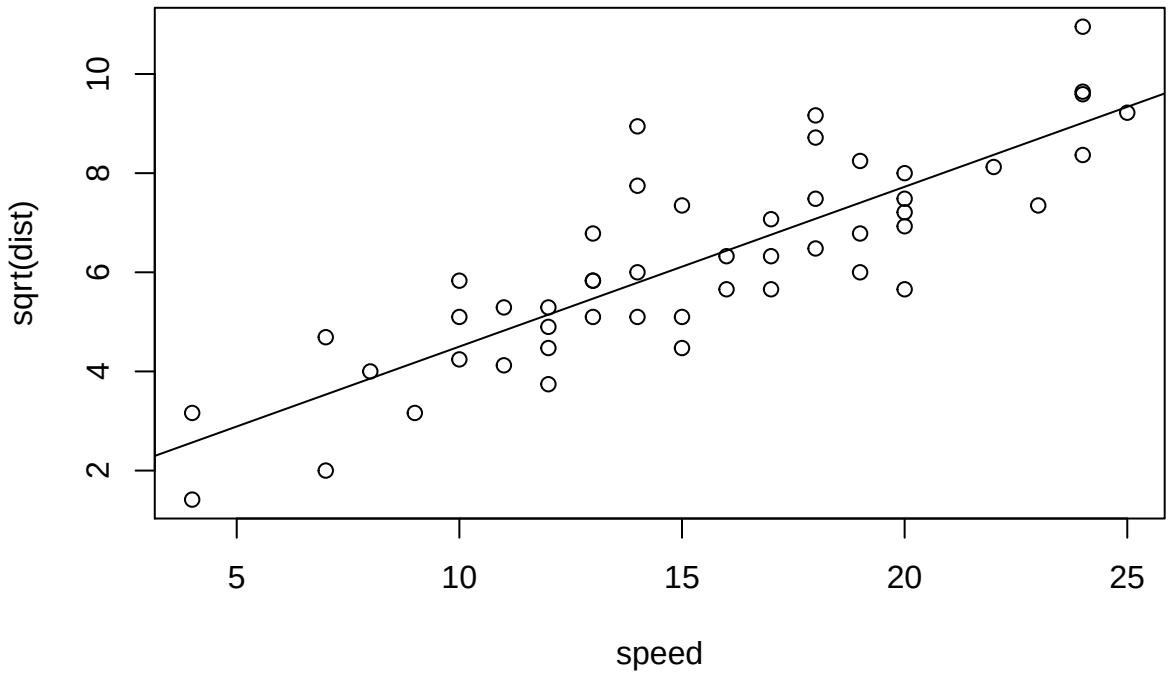

Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(dist ~ speed)
23
49
35
−2
−1
0
1
2
3


Scale−Location
√|Standardized residuals|
Fitted values
lm(dist ~ speed)
23
49
35
0.0
0.5
1.0
1.5
0
20
40
60
80

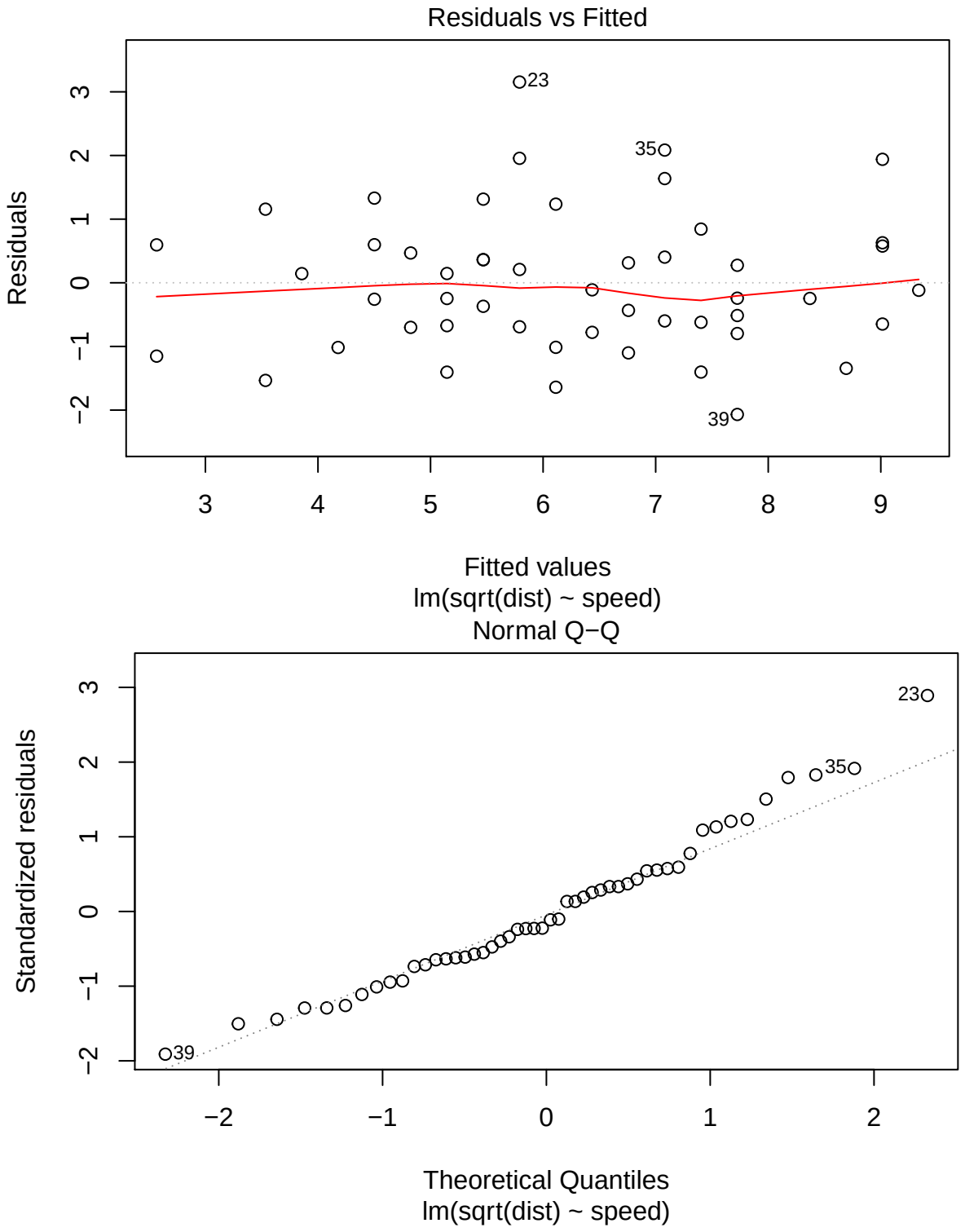


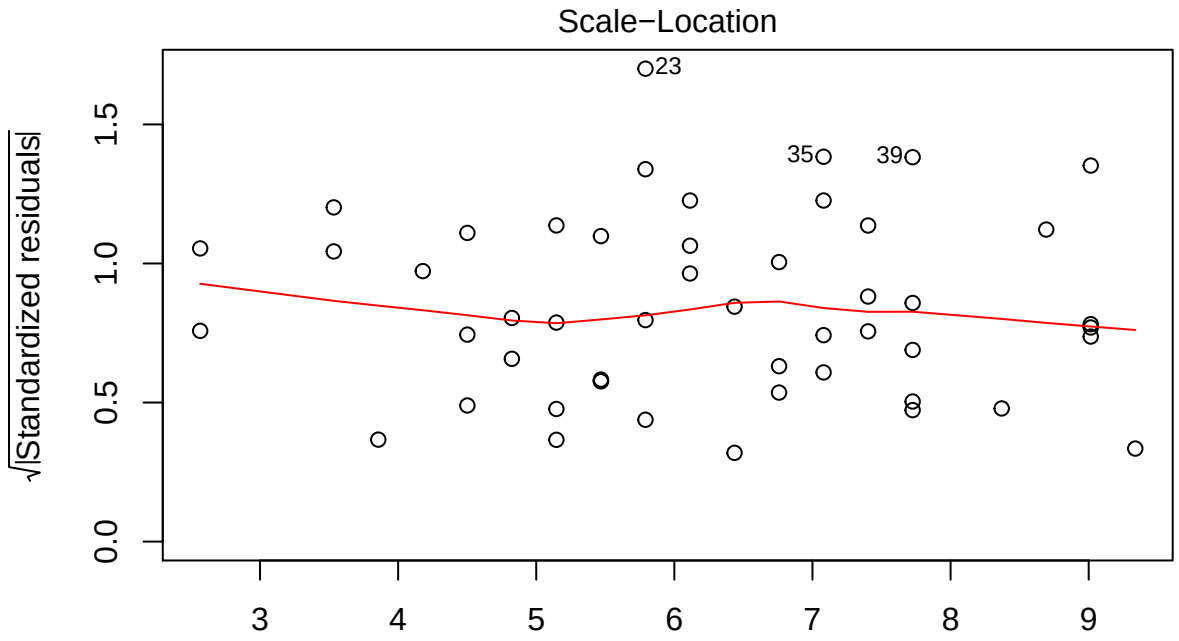
In this case, variance stability can be aided by a square-root transformation of the response variable:

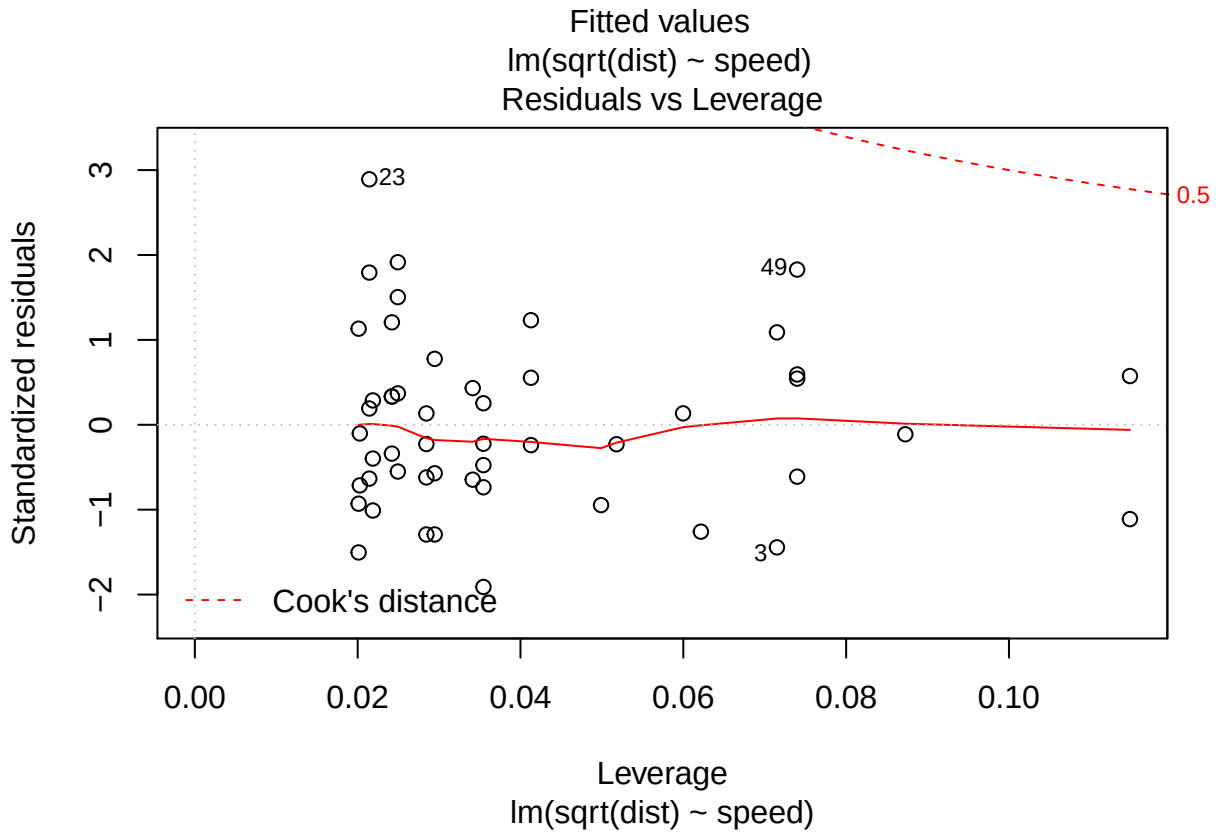
```
plot(sqrt(dist)~speed,data=cars)
m3 = lm(sqrt(dist)~speed,data=cars)
abline(m3)
```



```
plot(m3)
```


Residuals vs Fitted
Residuals
Fitted values
lm(sqrt(dist) ~ speed)
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(sqrt(dist) ~ speed)
23
35
39






```
summary(m3)
```

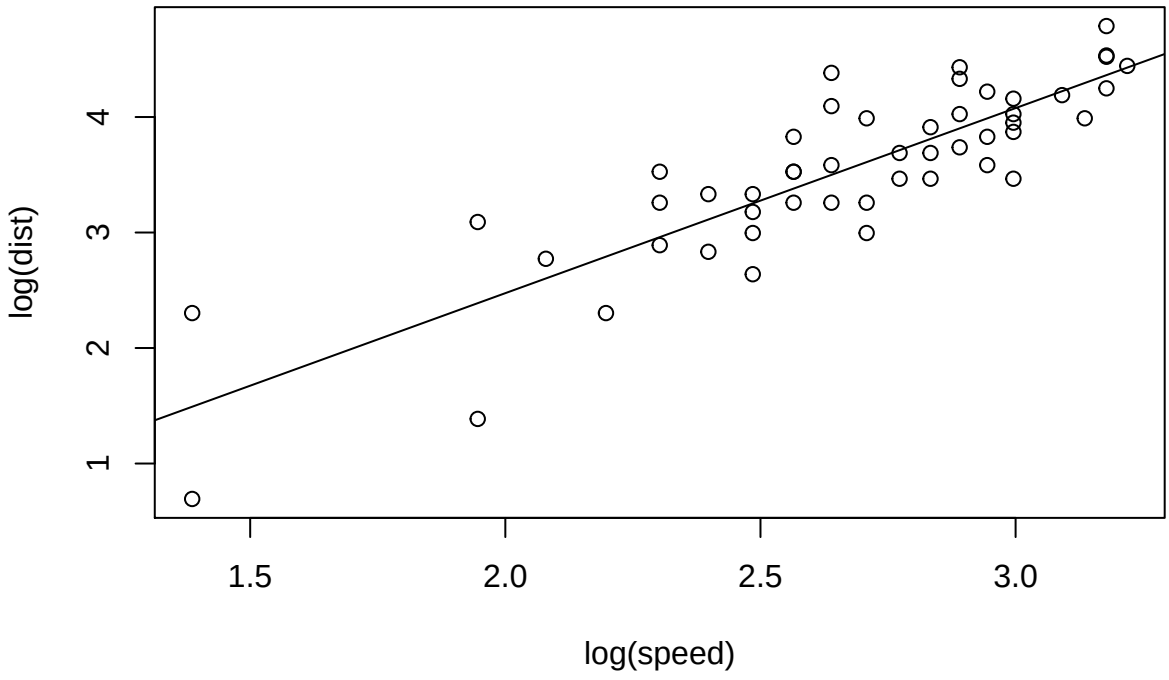
```
##
## Call:
## lm(formula = sqrt(dist) ~ speed, data = cars)
##
```

```
## 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
```

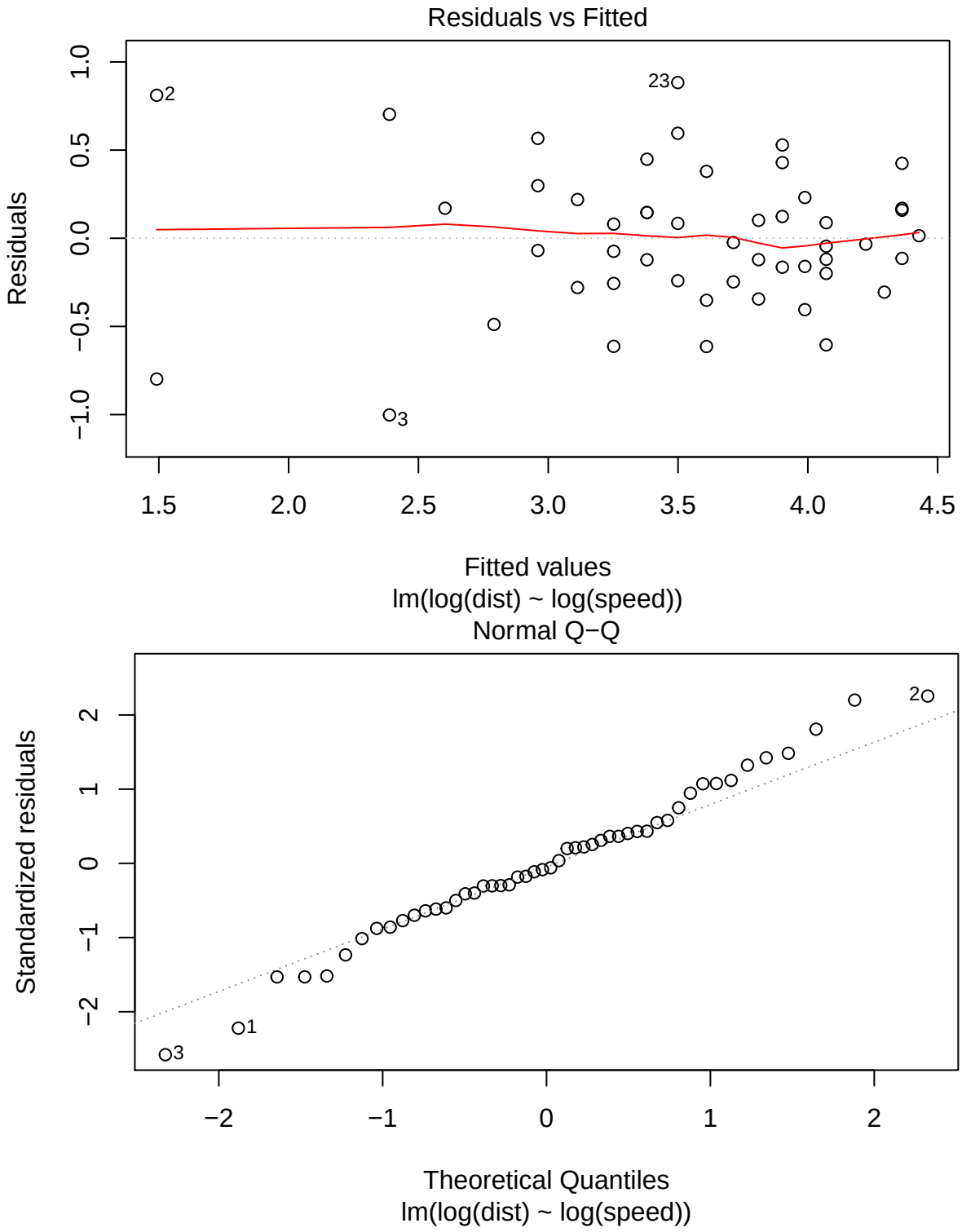
The R^2 value is improved over the previous model. Note that again that the intercept term is not significant.

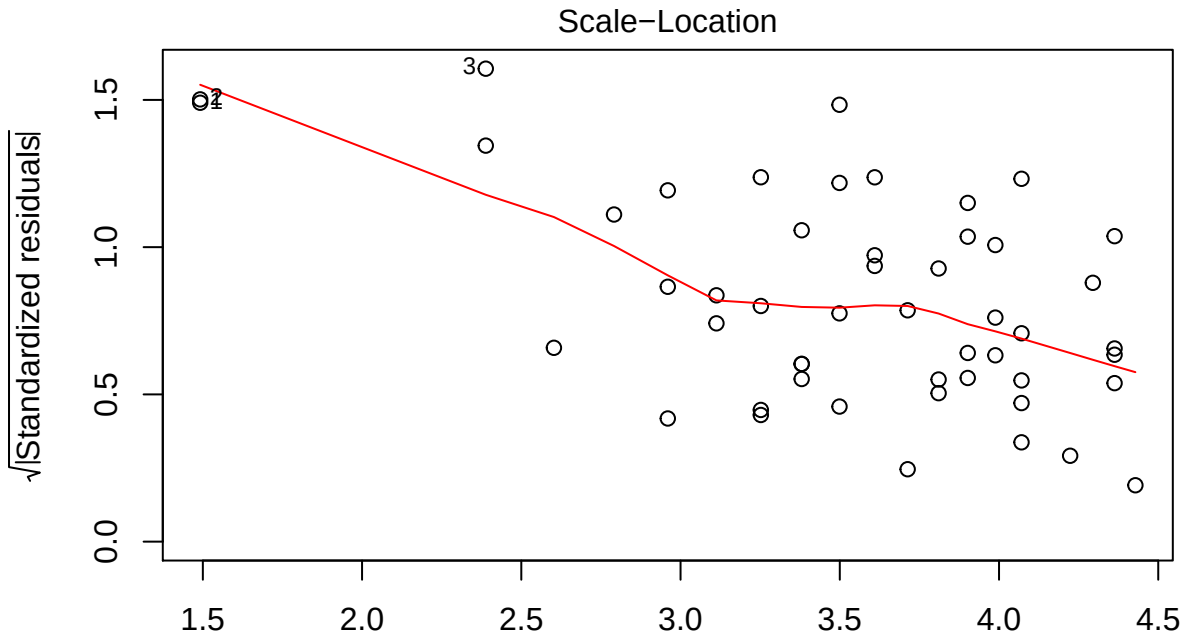
We'll now try a log-log transformation, that is applying a log transformation to be predictor and response variables. This represents a power relationship between the two variables.

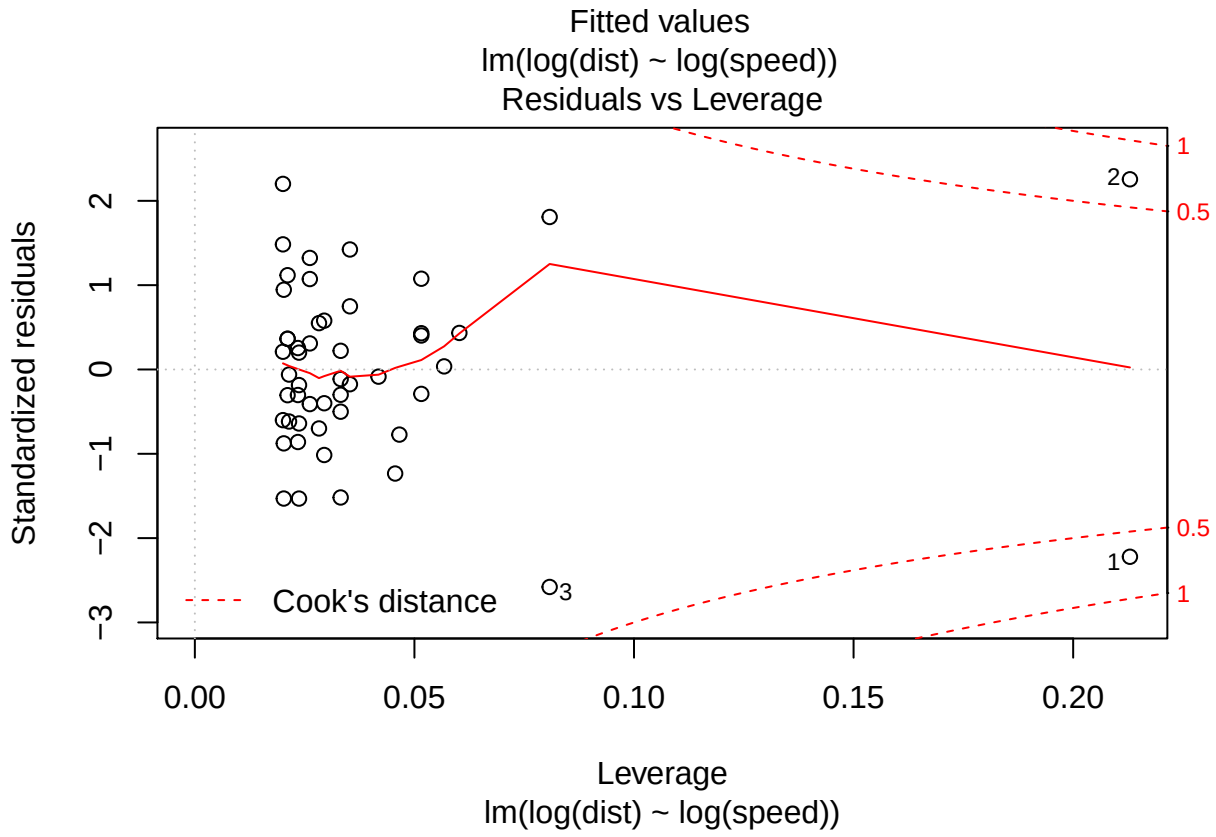
```
plot(log(dist)~log(speed),data=cars)
m4 = lm(log(dist)~log(speed),data=cars)
abline(m4)
```




```
plot(m4)
```


Residuals vs Fitted
Residuals
Fitted values
lm(log(dist) ~ log(speed))
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(log(dist) ~ log(speed))





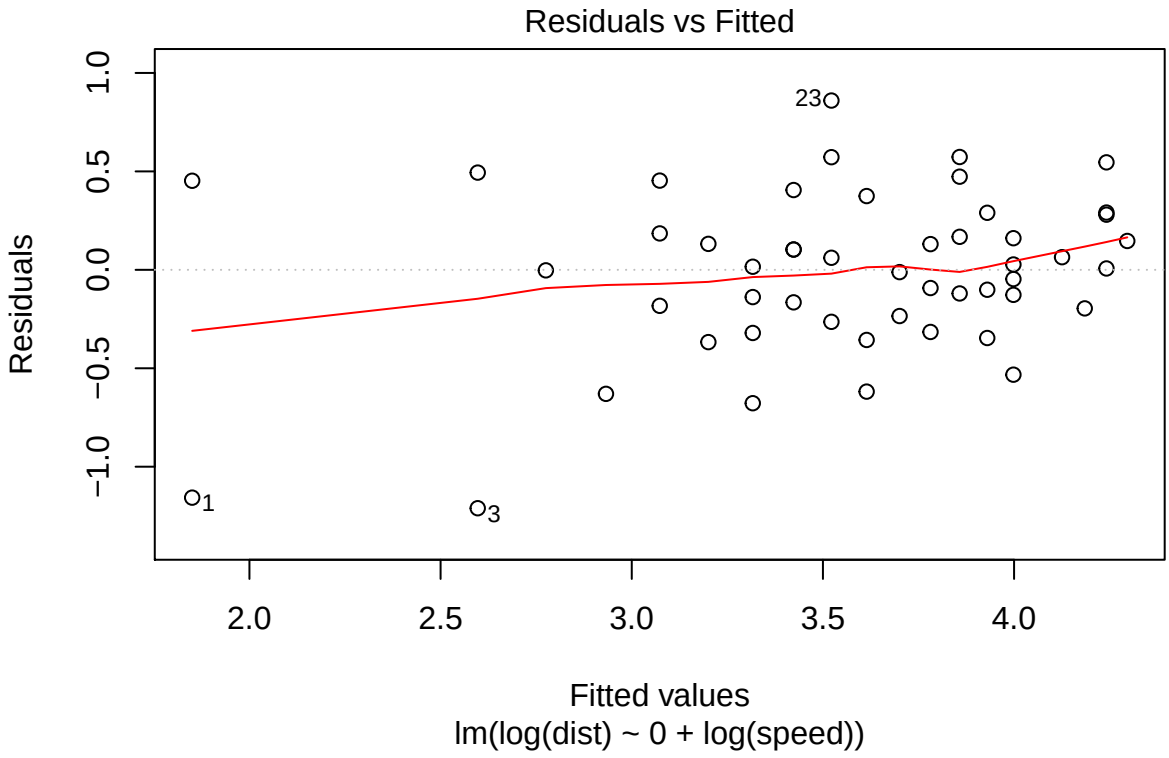

```
summary(m4)
```

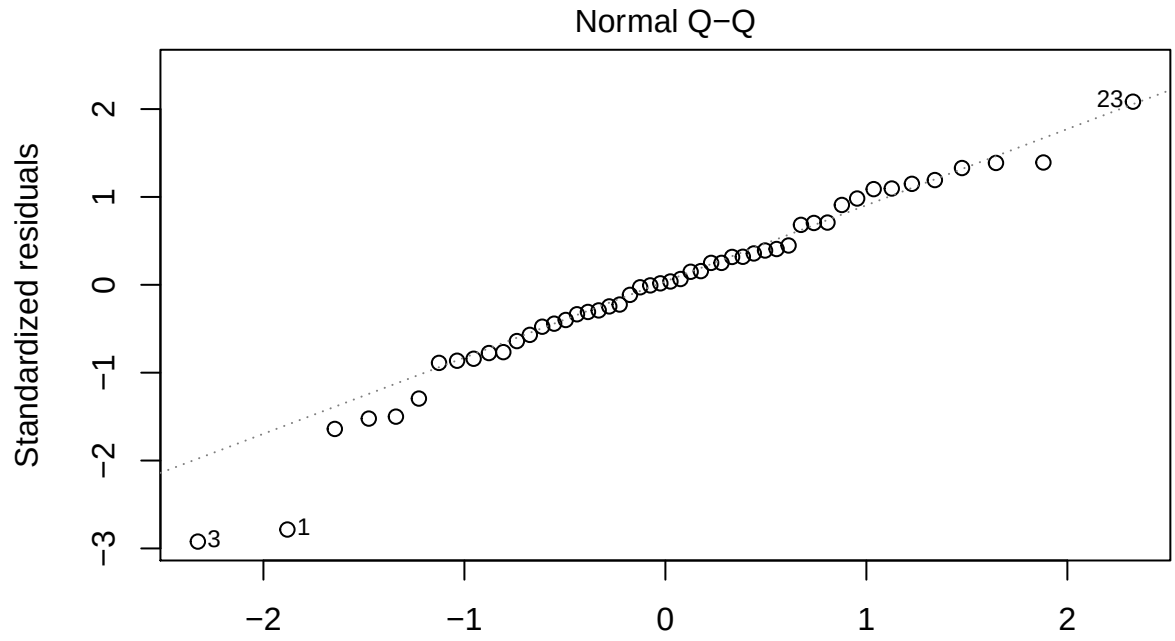
```
##
## Call:
## lm(formula = log(dist) ~ log(speed), data = cars)
##
```

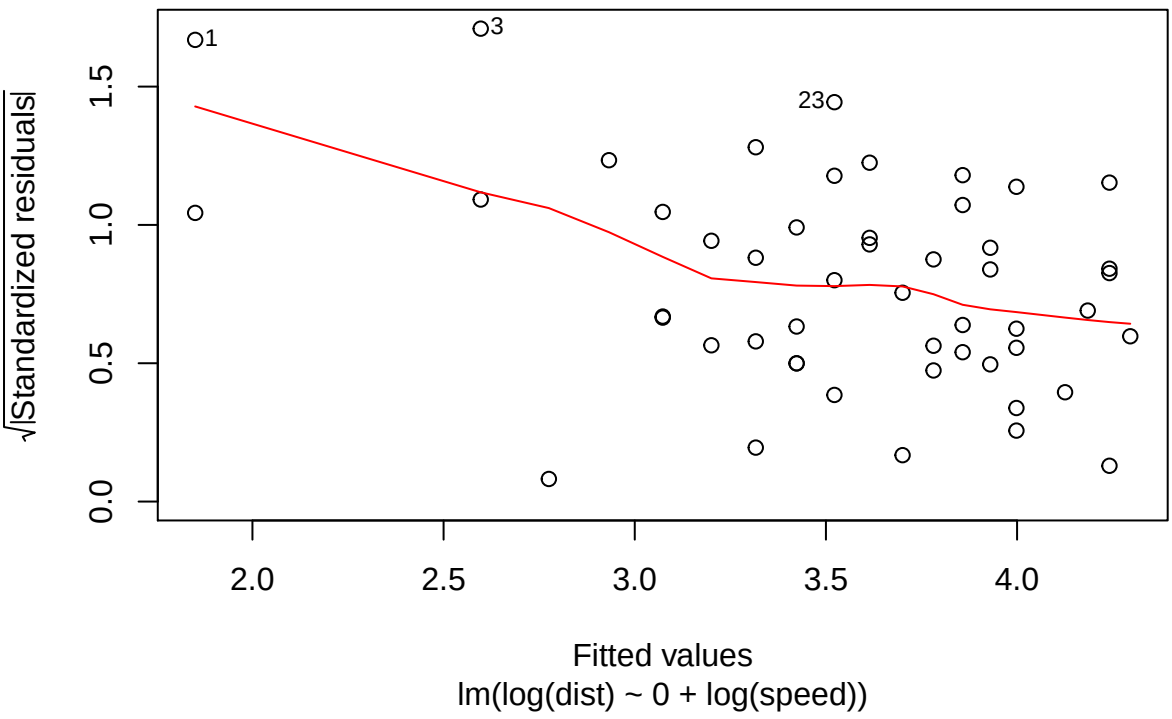
```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.00215 -0.24578 -0.02898  0.20717  0.88289
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.7297     0.3758  -1.941   0.0581 .
## log(speed)    1.6024     0.1395  11.484 2.26e-15 ***
## ---
## 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
```

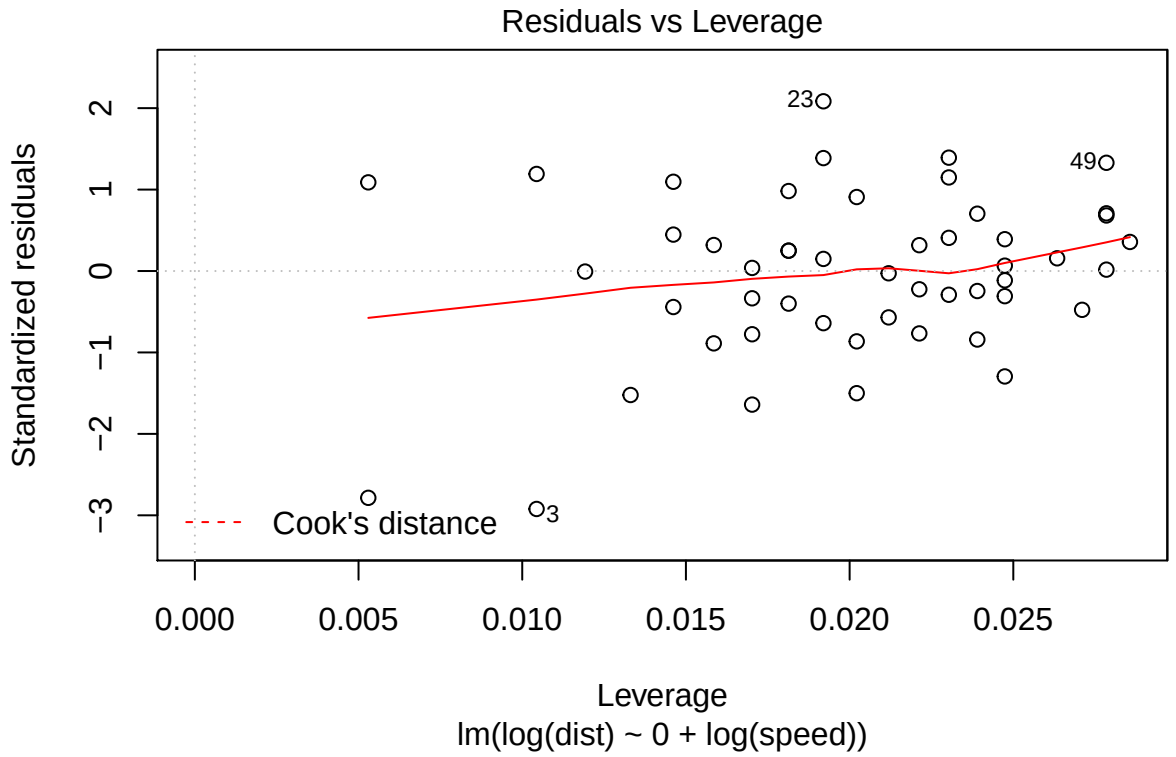
The R^2 value is improved, and the diagnostic plots don't look too unreasonable. However, again the intercept term does not have significant utility. So we'll now remove it from the model:

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




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


Scale−Location
√|Standardized residuals|
Fitted values
lm(log(dist) ~ 0 + log(speed))
1
3
23
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0



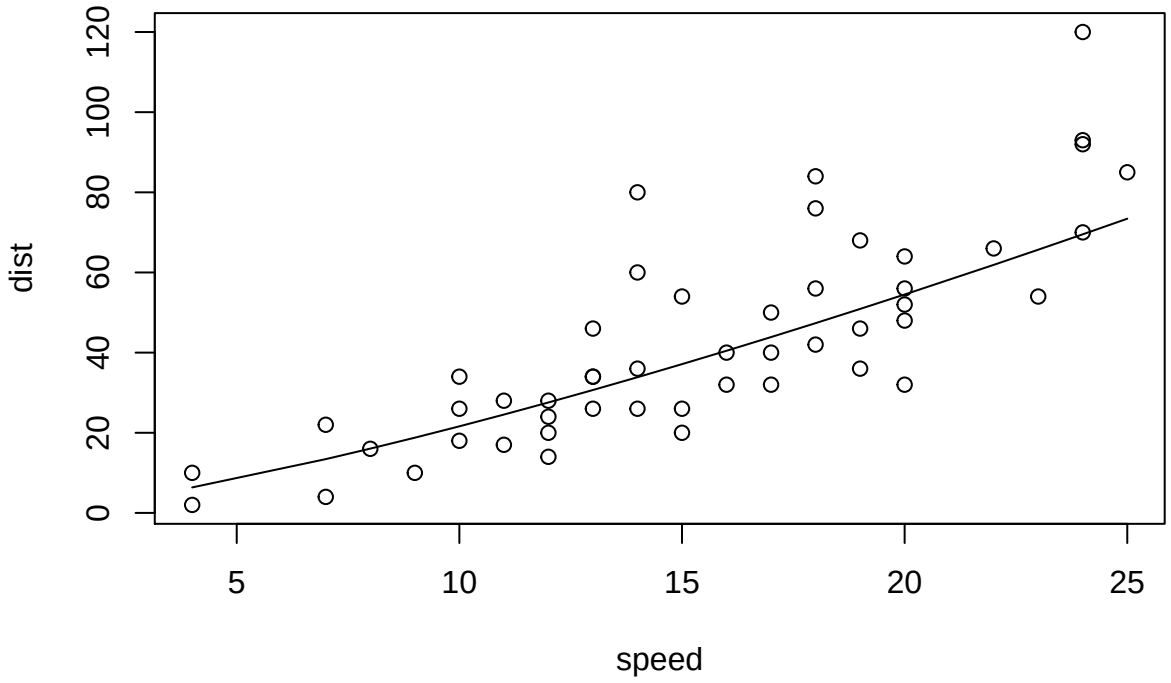

```
summary(m5)
```

```
##
## 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. Note however 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:

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



Section 4: Extra Practicals

Practical 2: 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 using waiting time 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.

Practical 3: Pharmacokinetics of Indomethacin

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

- Plot time versus 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.