Chapter 7: Regression

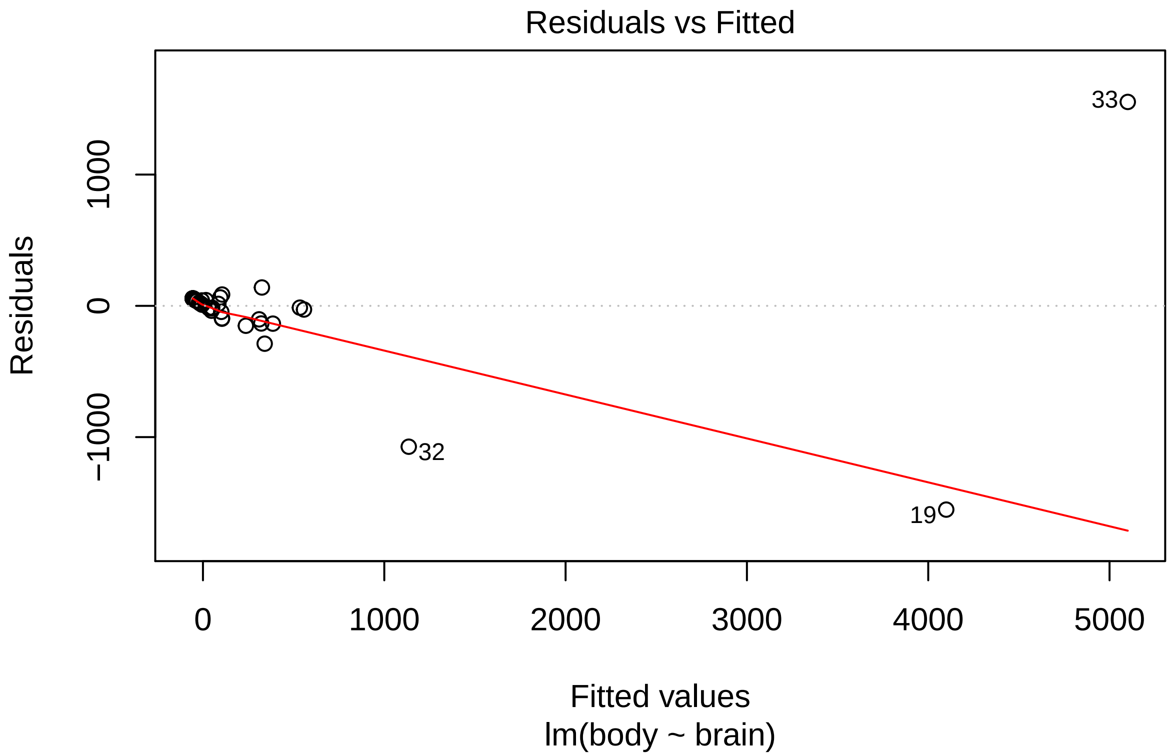
Alex Chi

Exercises

7.1 (mammals data).

The mammals data set in the MASS package records brain size and body size for 62 different mammals. Fit a regression model to describe the relation between brain size and body size. Display a residual plot using the plot method for the result of the lm function. Which observation (which mammal) has the largest residual in your fitted model?

```
library(MASS)
attach(mammals)
fit = lm(body ~ brain)
plot(fit, which=1)
```



```
mammals[33, ]
```

```
##                   body brain
## African elephant 6654  5712
```

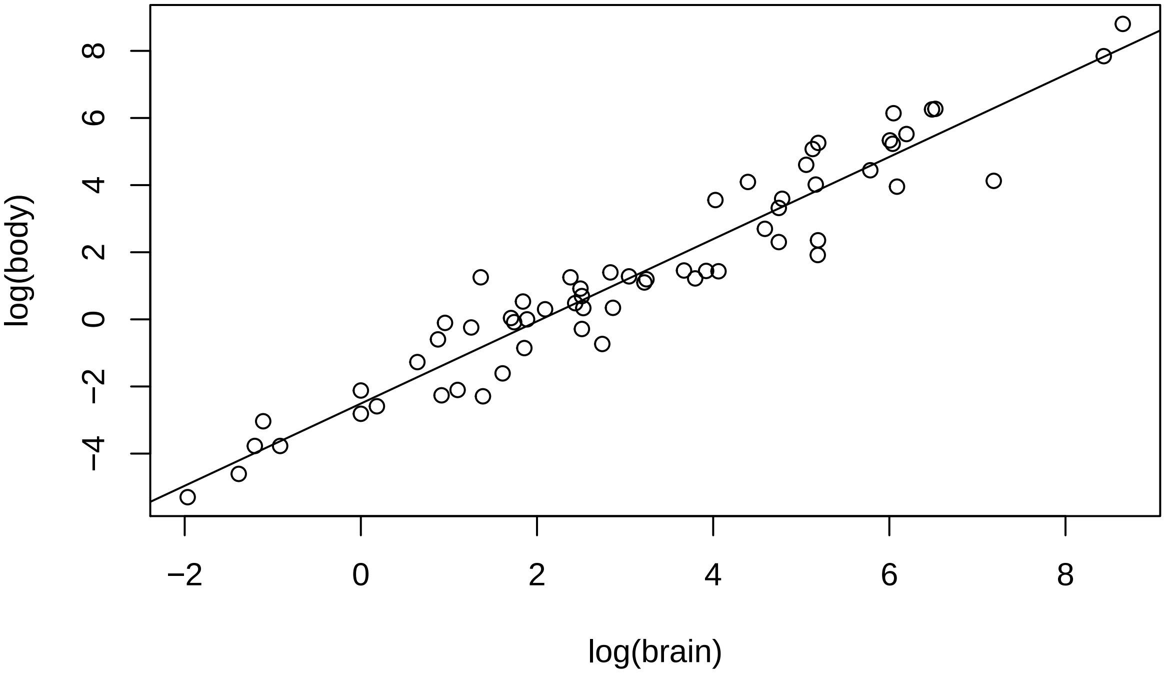
7.2 (mammals, continued).

Refer to the mammals data in package MASS. Display a scatterplot of log(brain) vs log(body). Fit a simple linear regres- sion model to the transformed data. What is the equation of the fitted model? Display a fitted line plot and comment on the fit. Compare your results with results of Exercise 7.1.

```r
plot(log(body) ~ log(brain))
fit = lm(log(body) ~ log(brain))
fit
```

```
##
## Call:
## lm(formula = log(body) ~ log(brain))
##
## Coefficients:
## (Intercept)   log(brain)
##      -2.509        1.225
```

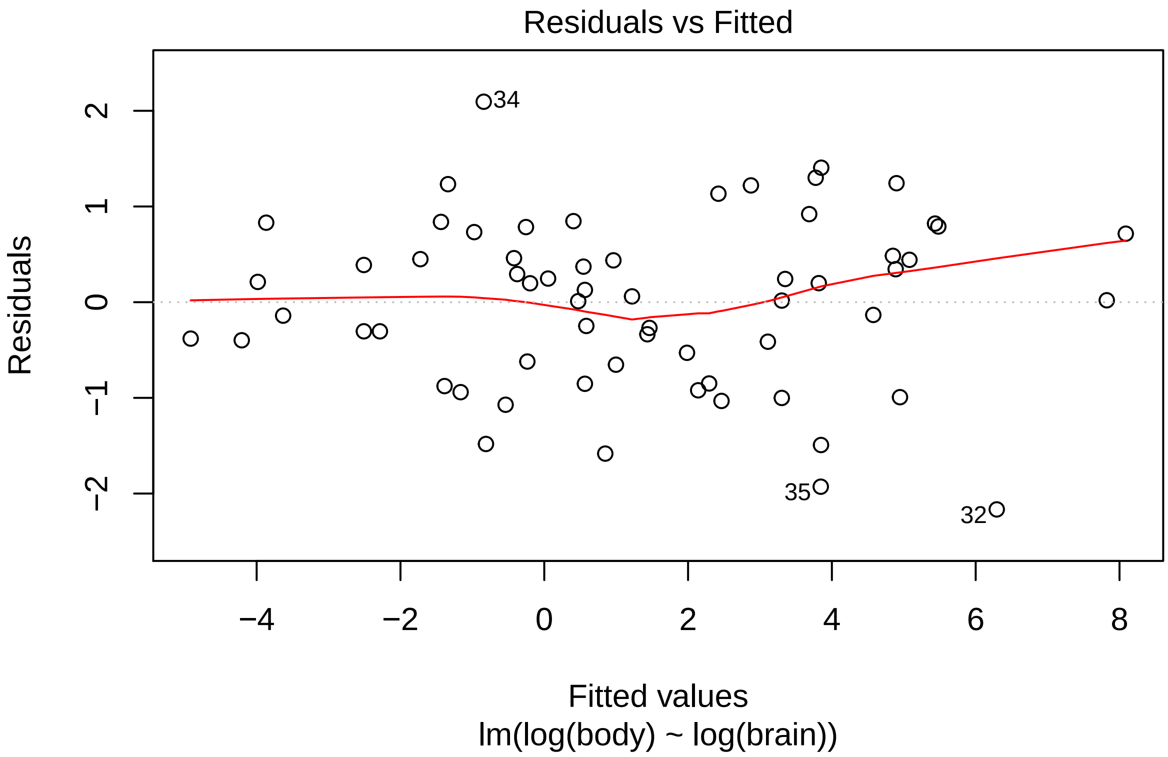
```r
abline(fit)
```



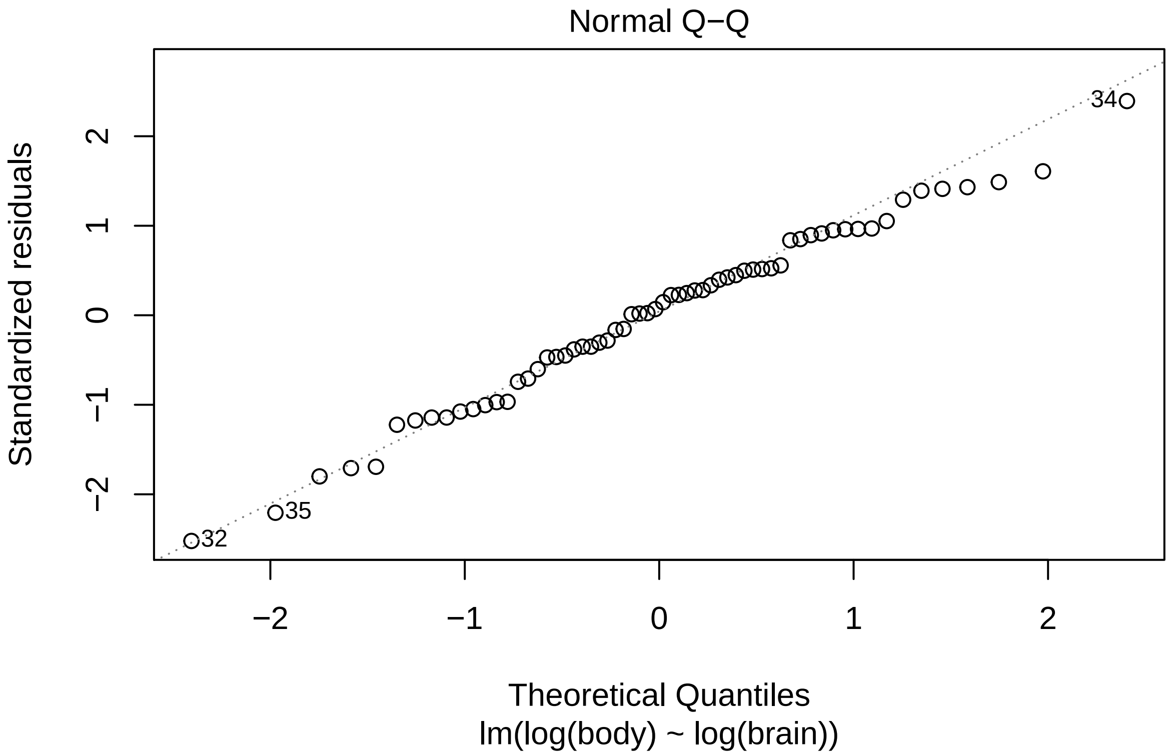
7.3 (mammals residuals).

Refer to Exercise 7.2. Display a plot of residuals vs fitted values and a normal-QQ plot of residuals. Do the residuals appear to be approximately normally distributed with constant variance?

```r
plot(fit, which=1)
```



```r
plot(fit, which=2)
```



7.4 (mammals summary statistics).

Refer to Exercise 7.2. Use the sum- mary function on the result of lm to display the summary statistics for the model. What is the estimate of the error variance? Find the coefficient of determination (R2) and compare it to the square of the correlation between the response and predictor. Interpret the value of (R2) as a measure of fit.

```r
summary(fit)
```

```
##
## Call:
## lm(formula = log(body) ~ log(brain))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.16559 -0.59763  0.09433  0.65789  2.09470
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.50907    0.18408  -13.63   <2e-16 ***
## log(brain)   1.22496    0.04638   26.41   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8863 on 60 degrees of freedom
## Multiple R-squared:  0.9208, Adjusted R-squared:  0.9195
## F-statistic: 697.4 on 1 and 60 DF,  p-value: < 2.2e-16
```

```r
detach(mammals)
```

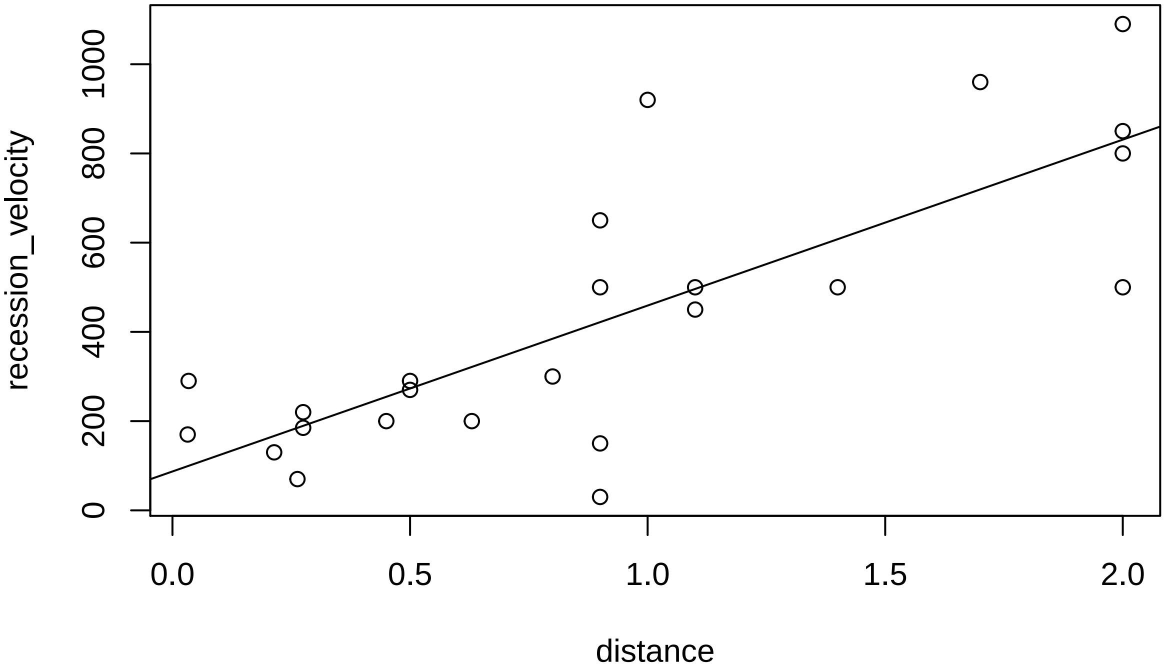
7.5 (Hubble's Law).

In 1929 Edwin Hubble investigated the relationship between distance and velocity of celestial objects. Knowledge of this rela- tionship might give clues as to how the universe was formed and what may happen in the future. Hubble's Law is is Recession Velocity = H0×Distance, where H0is Hubble's constant. This model is a straight line through the origin with slope H0. Data that Hubble used to estimate the constant H0are given on the DASL web at http://lib.stat.cmu.edu/DASL/Datafiles/Hubble. html. Use the data to estimate Hubble's constant by simple linear regression

```r
data = read.table("Rx-Data/Hubble.txt", header=TRUE)
attach(data)
recession_velocity=abs(recession_velocity)
fit = lm(recession_velocity ~ distance)
fit$coefficients
```

```
## (Intercept)    distance
##    87.09463   371.90733
```

```
plot(recession_velocity ~ distance)
abline(fit)
```



```
detach(data)
```

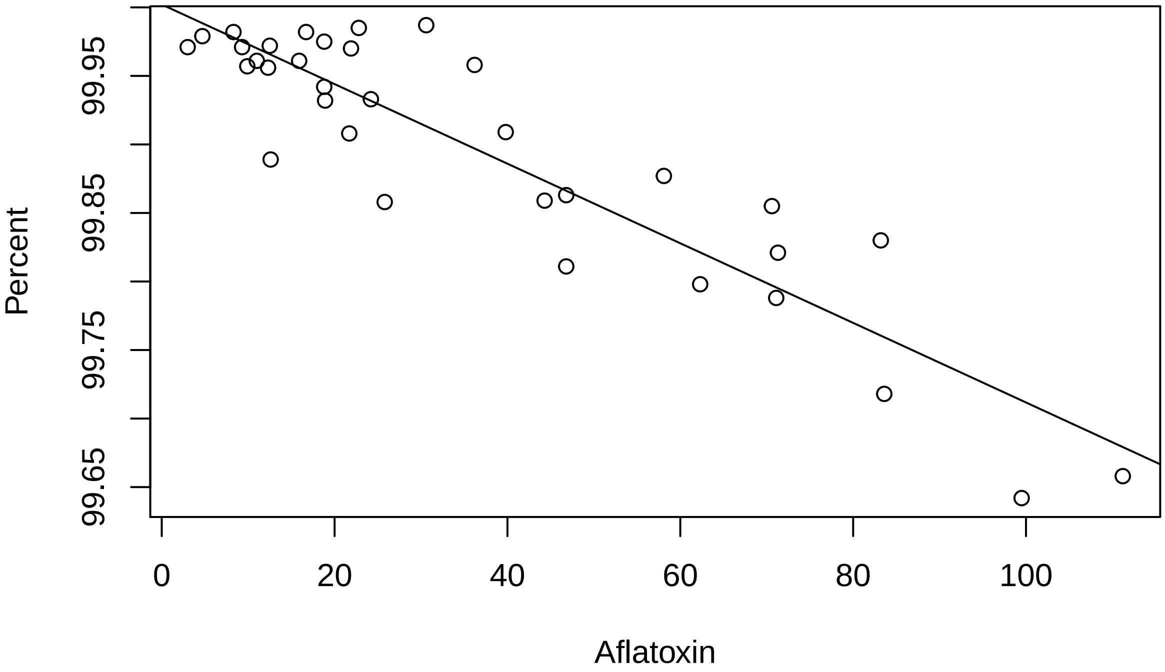
7.6 (peanuts data).

The data file “peanuts.txt” (Hand et al. [21]) records levels of a toxin in batches of peanuts. The data are the average level of aflatoxin X in parts per billion, in 120 pounds of peanuts, and percentage of non-contaminated peanuts Y in the batch. Use a simple linear regression model to predict Y from X. Display a fitted line plot. Plot residuals, and comment on the adequacy of the model. Obtain a prediction of percentage of non-contaminated peanuts at levels 20, 40, 60, and 80 of aflatoxin.

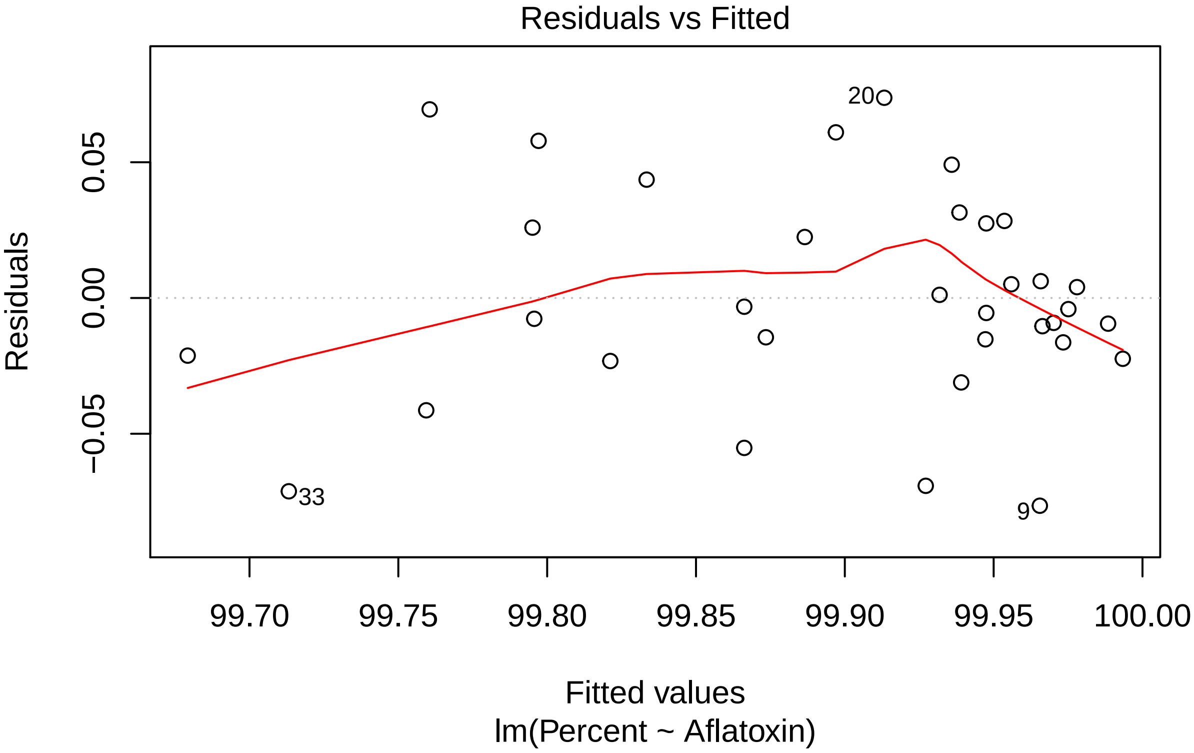
```
peanuts = read.table("Rx-Data/peanuts.txt", header=TRUE)
attach(peanuts)
peanuts
```

```
##     Percent Aflatoxin
## 1    99.971       3.0
## 2    99.979       4.7
## 3    99.982       8.3
## 4    99.971       9.3
## 5    99.957       9.9
## 6    99.961      11.0
## 7    99.956      12.3
## 8    99.972      12.5
## 9    99.889      12.6
## 10  99.961      15.9
## 11  99.982      16.7
## 12  99.975      18.8
## 13  99.942      18.8
## 14  99.932      18.9
## 15  99.908      21.7
## 16  99.970      21.9
## 17  99.985      22.8
## 18  99.933      24.2
## 19  99.858      25.8
## 20  99.987      30.6
## 21  99.958      36.2
## 22  99.909      39.8
## 23  99.859      44.3
## 24  99.863      46.8
## 25  99.811      46.8
## 26  99.877      58.1
## 27  99.798      62.3
## 28  99.855      70.6
## 29  99.788      71.1
## 30  99.821      71.3
## 31  99.830      83.2
## 32  99.718      83.6
## 33  99.642      99.5
## 34  99.658     111.2
```

```r
fit = lm(Percent ~ Aflatoxin)
plot(Percent ~ Aflatoxin)
abline(fit)
```



```
plot(fit, which=1)
```



```
aflatoxin = c(20, 40, 60, 80)
new = data.frame(Aflatoxin = aflatoxin)
predict(fit, newdata = new, interval = "pred")
```

```
##          fit       lwr       upr
## 1 99.94403 99.86237 100.02569
## 2 99.88596 99.80467  99.96725
## 3 99.82789 99.74585  99.90993
## 4 99.76982 99.68596  99.85368
```

```
detach(peanuts)
```

7.7 (cars data).

For the cars data in Example 7.1, compare the coefficient of determination R2for the two models (with and without intercept term in the model). Hint: Save the fitted model as L and use summary(L) to display R2. Interpret the value of R2as a measure of the fit.

```
attach(cars)
fit1 = lm(dist ~ speed)
fit2 = lm(dist ~ speed + 0)
summary(fit1)$r.squared
```

```
## [1] 0.6510794
```

```
summary(fit2)$r.squared
```

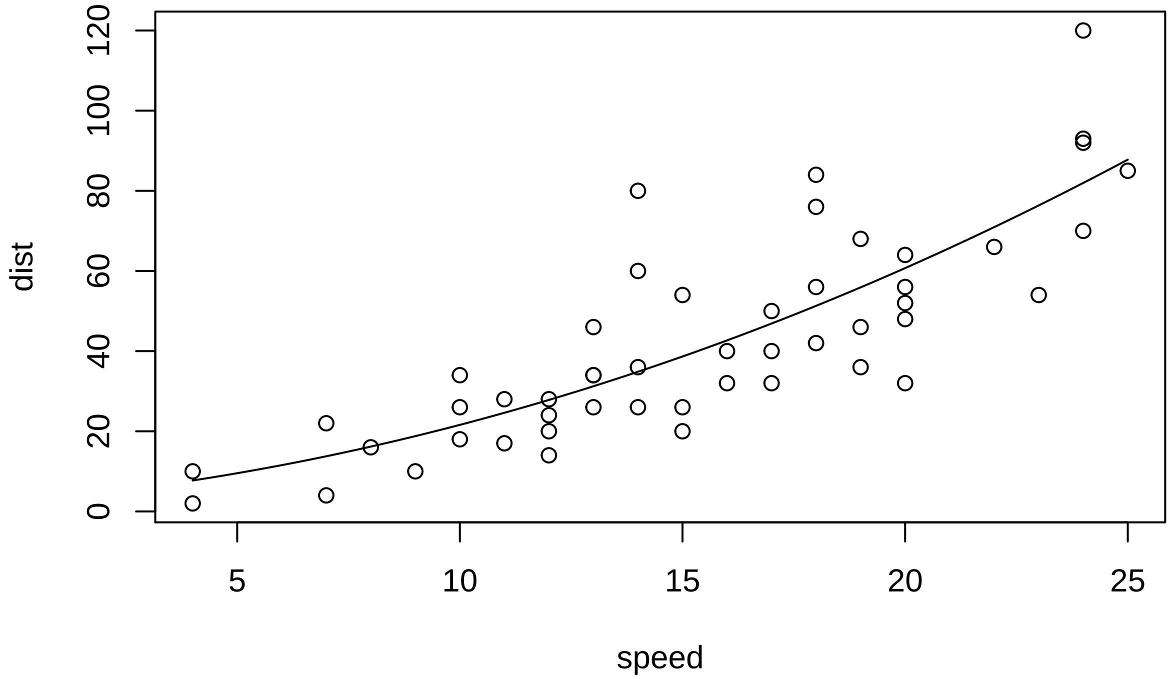
```
## [1] 0.8962893
```

```
detach(cars)
```

7.8 (cars data, continued).

Refer to the cars data in Example 7.1. Create a new variable speed2 equal to the square of speed. Then use lm to fit a quadratic model dist = β0+β1speed+β2(speed)2+ε.

```
attach(cars)
speed2 = speed ^ 2
fit = lm(dist ~ speed + speed2)
plot(dist ~ speed)
curve(fit$coef[1] + fit$coef[2] * x + fit$coef[3] * (x ^ 2), add=TRUE)
```



```r
detach(cars)
```

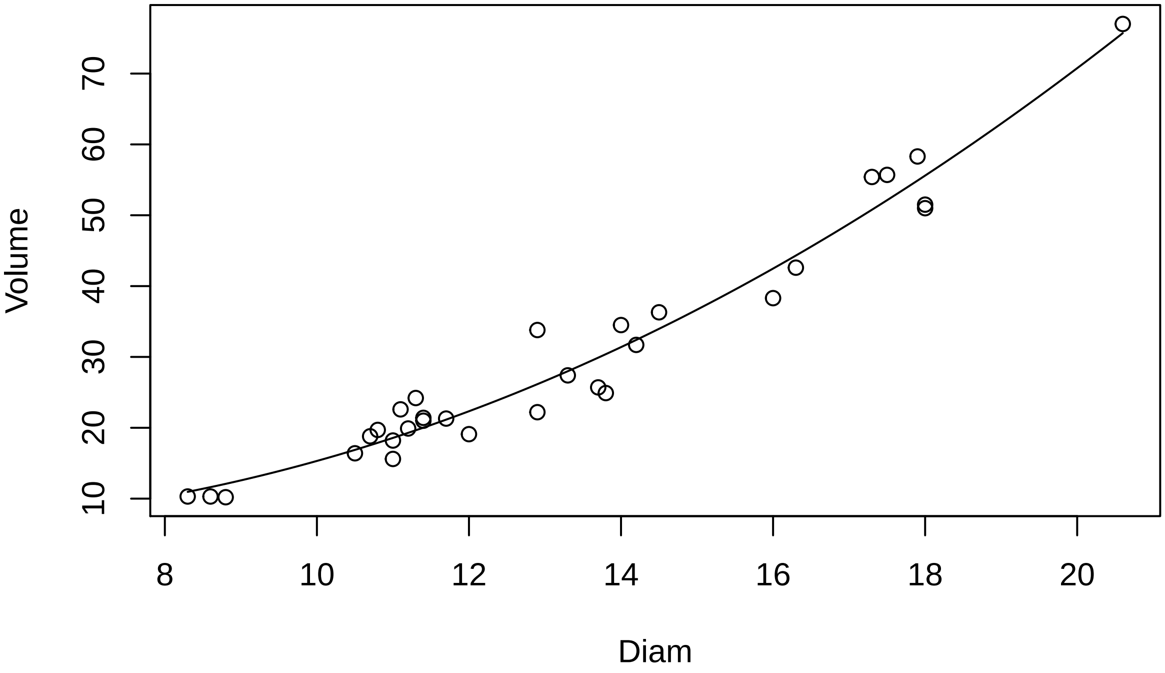
7.9 (Cherry Tree data, quadratic regression model).

Refer to the Cherry Tree data in Example 7.3. Fit and analyze a quadratic regression model y = b0+ b1x + b2x2for predicting volume y given diameter x. Check the residual plots and summarize the results.

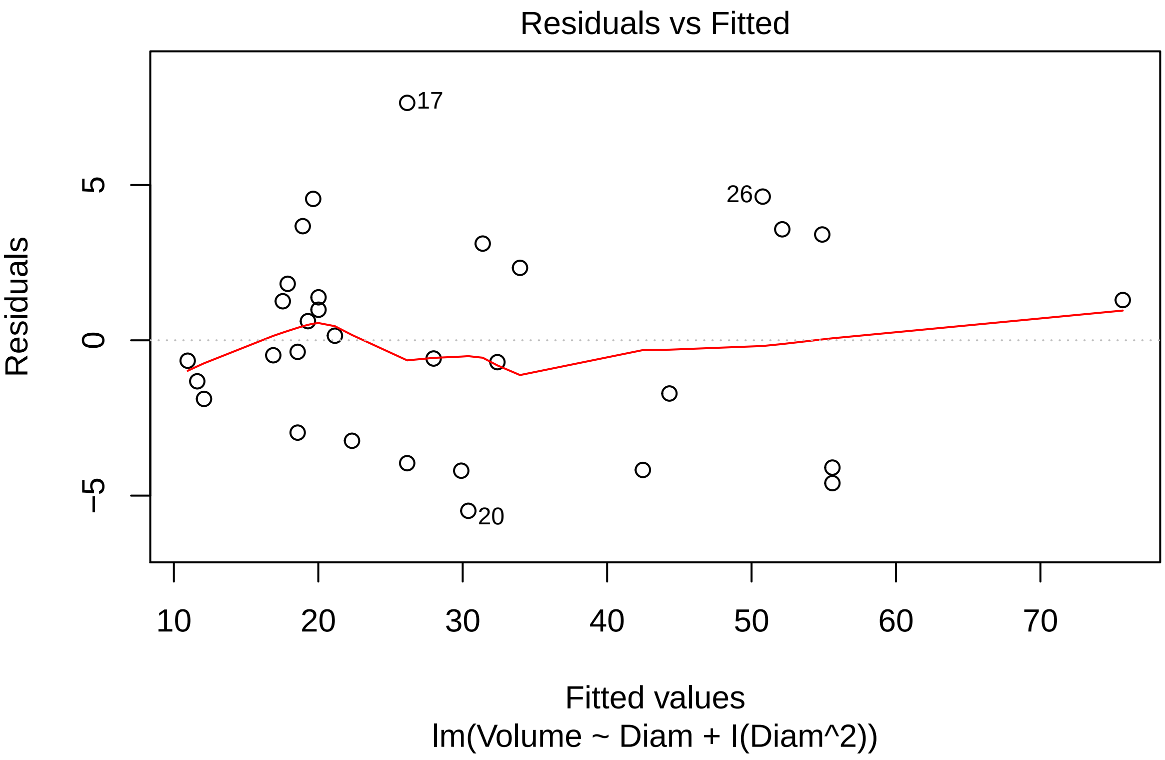
```r
cherry = read.table("Rx-Data/cherry.txt", header=TRUE)
attach(cherry)
cherry
```

```
##    Diam Height Volume
## 1   8.3     70   10.3
## 2   8.6     65   10.3
## 3   8.8     63   10.2
## 4  10.5     72   16.4
## 5  10.7     81   18.8
## 6  10.8     83   19.7
## 7  11.0     66   15.6
## 8  11.0     75   18.2
## 9  11.1     80   22.6
## 10 11.2     75   19.9
## 11 11.3     79   24.2
## 12 11.4     76   21.0
## 13 11.4     76   21.4
## 14 11.7     69   21.3
## 15 12.0     75   19.1
## 16 12.9     74   22.2
## 17 12.9     85   33.8
## 18 13.3     86   27.4
## 19 13.7     71   25.7
## 20 13.8     64   24.9
## 21 14.0     78   34.5
## 22 14.2     80   31.7
## 23 14.5     74   36.3
## 24 16.0     72   38.3
## 25 16.3     77   42.6
## 26 17.3     81   55.4
## 27 17.5     82   55.7
## 28 17.9     80   58.3
## 29 18.0     80   51.5
## 30 18.0     80   51.0
## 31 20.6     87   77.0
```

```
fit = lm(Volume ~ Diam + I(Diam^2))
plot(Volume ~ Diam)
curve(fit$coef[1] + fit$coef[2] * x + fit$coef[3] * x ^ 2, add=TRUE)
```



```
plot(fit, which=1)
```



```r
summary(fit)
```

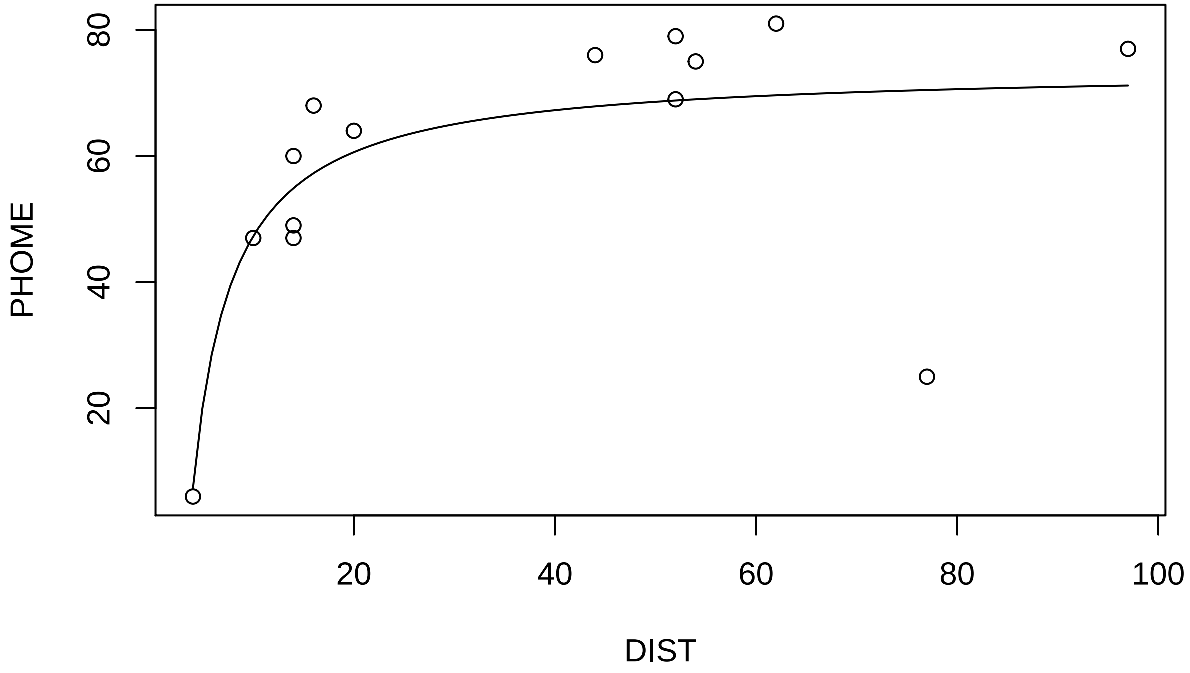
```
##
## Call:
## lm(formula = Volume ~ Diam + I(Diam^2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.4889 -2.4293 -0.3718  2.0764  7.6447
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.78627   11.22282   0.961 0.344728
## Diam        -2.09214    1.64734  -1.270 0.214534
## I(Diam^2)    0.25454    0.05817   4.376 0.000152 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.335 on 28 degrees of freedom
## Multiple R-squared:  0.9616, Adjusted R-squared:  0.9588
## F-statistic: 350.5 on 2 and 28 DF,  p-value: < 2.2e-16
```

```r
detach(cherry)
```

7.10 (lunatics data).

Refer to the “lunatics” data in Example 7.8. Repeat the analysis, after deleting the two counties that are offshore islands, NAN- TUCKET and DUKES counties. Compare the estimates of slope and intercept with those obtained in Example 7.8. Construct the plots and analyze the residuals as in Example 7.8.

```
lunatics = read.table(”Rx-Data/lunatics.txt”, header=TRUE)
attach(lunatics)
lunatics = lunatics[COUNTY ≠ ”NANTUCKET” & COUNTY ≠ ”DUKES”, ]
M = lm(PHOME ~ I(1 / DIST))
plot(PHOME ~ DIST)
curve(M$coef[1] + M$coef[2] / x, add=TRUE)
```



```
summary(M)
```

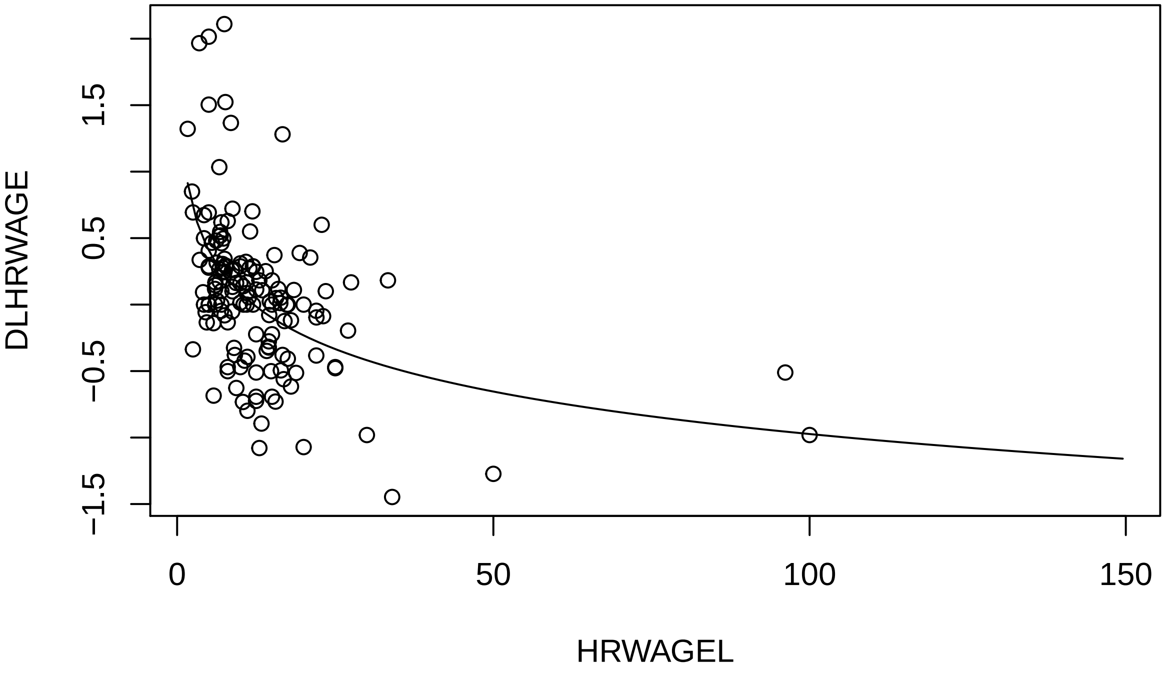
```
##
## Call:
## lm(formula = PHOME ~ I(1/DIST))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.468  -1.083   4.243   7.596  11.369
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   73.927      5.493  13.459 1.33e-08 ***
## I(1/DIST)   -266.324     66.211  -4.022  0.00169 **
## —
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.97 on 12 degrees of freedom
## Multiple R-squared:  0.5742, Adjusted R-squared:  0.5387
## F-statistic: 16.18 on 1 and 12 DF,  p-value: 0.001692
```

```
detach(lunatics)
```

7.11 (twins data).

Import the data file “twins.txt” using read.table. (The commands to read this data file are shown in the twins example in Section 3.3, page 85.) The variable DLHRWAGE is the difference (twin 1 minus twin 2) in the logarithm of hourly wage, given in dollars. The variable HRWAGEL is the hourly wage of twin 1. Fit and analyze a simple linear regression model to predict the difference DLHRWAGE given the logarithm of the hourly wage of twin 1.

```
twins = read.table(”Rx-data/twins.txt”, header=TRUE, sep=”,”, na.strings=”.”)
attach(twins)
M = lm(DLHRWAGE ~ I(log(HRWAGEL)))
plot(DLHRWAGE ~ HRWAGEL)
curve(M$coef[1] + M$coef[2] * log(x), add=TRUE)
```



```
summary(M)
```

```
##
## Call:
## lm(formula = DLHRWAGE ~ I(log(HRWAGEL)))
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.06338 -0.30402  0.01665  0.22892  1.88689 
## 
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)    
## (Intercept)      1.14922    0.15887   7.234 2.40e-11 ***
## I(log(HRWAGEL)) -0.46090    0.06545  -7.042 6.75e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.5029 on 147 degrees of freedom
##   (34 observations deleted due to missingness)
## Multiple R-squared:  0.2523, Adjusted R-squared:  0.2472 
## F-statistic: 49.59 on 1 and 147 DF,  p-value: 6.751e-11
```

```
detach(twins)
```