STAT 210
Applied Statistics and Data Analysis
Linear Regression V:
Model Assessment

Joaquin Ortega

Model Assessment

Model Assessment

The quality of a model depends, to a large extent, on the veracity of the assumptions we have made, which are the basis for the estimation of the parameters.

We also need to check the goodness-of-fit of the model and the possible presence of outliers or highly influential data points.

The techniques we will consider are mainly graphical. Graphs are a fundamental tool for statistical practice and in particular for model assessment.

We start by recalling Anscombe's quartet. The code that follows is from the `R` documentation of `anscombe`:

Model Assessment

```
summary(anscombe)
```

```
##        x1              x2              x3              x4
##  Min.   : 4.0   Min.   : 4.0   Min.   : 4.0   Min.   : 8
##  1st Qu.: 6.5   1st Qu.: 6.5   1st Qu.: 6.5   1st Qu.: 8
##  Median : 9.0   Median : 9.0   Median : 9.0   Median : 8
##  Mean   : 9.0   Mean   : 9.0   Mean   : 9.0   Mean   : 9
##  3rd Qu.:11.5   3rd Qu.:11.5   3rd Qu.:11.5   3rd Qu.: 8
##  Max.   :14.0   Max.   :14.0   Max.   :14.0   Max.   :19
##        y1              y2              y3              y4
##  Min.   : 4.260   Min.   :3.100   Min.   : 5.39   Min.   : 5.250
##  1st Qu.: 6.315   1st Qu.:6.695   1st Qu.: 6.25   1st Qu.: 6.170
##  Median : 7.580   Median :8.140   Median : 7.11   Median : 7.040
##  Mean   : 7.501   Mean   :7.501   Mean   : 7.50   Mean   : 7.501
##  3rd Qu.: 8.570   3rd Qu.:8.950   3rd Qu.: 7.98   3rd Qu.: 8.190
##  Max.   :10.840   Max.   :9.260   Max.   :12.74   Max.   :12.500
```

Model Assessment

```
## Analysis of Variance Table
##
## Response: y1
##           Df Sum Sq Mean Sq F value  Pr(>F)
## x1         1 27.510 27.5100   17.99 0.00217 **
## Residuals  9 13.763  1.5292
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Analysis of Variance Table
##
## Response: y2
##           Df Sum Sq Mean Sq F value   Pr(>F)
## x2         1 27.500 27.5000  17.966 0.002179 **
## Residuals  9 13.776  1.5307
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Analysis of Variance Table
##
## Response: y3
##           Df Sum Sq Mean Sq F value   Pr(>F)
## x3         1 27.470 27.4700  17.972 0.002176 **
## Residuals  9 13.756  1.5285
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Analysis of Variance Table
##
## Response: y4
##           Df Sum Sq Mean Sq F value   Pr(>F)
## x4         1 27.490 27.4900  18.003 0.002165 **
## Residuals  9 13.742  1.5269
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model Assessment

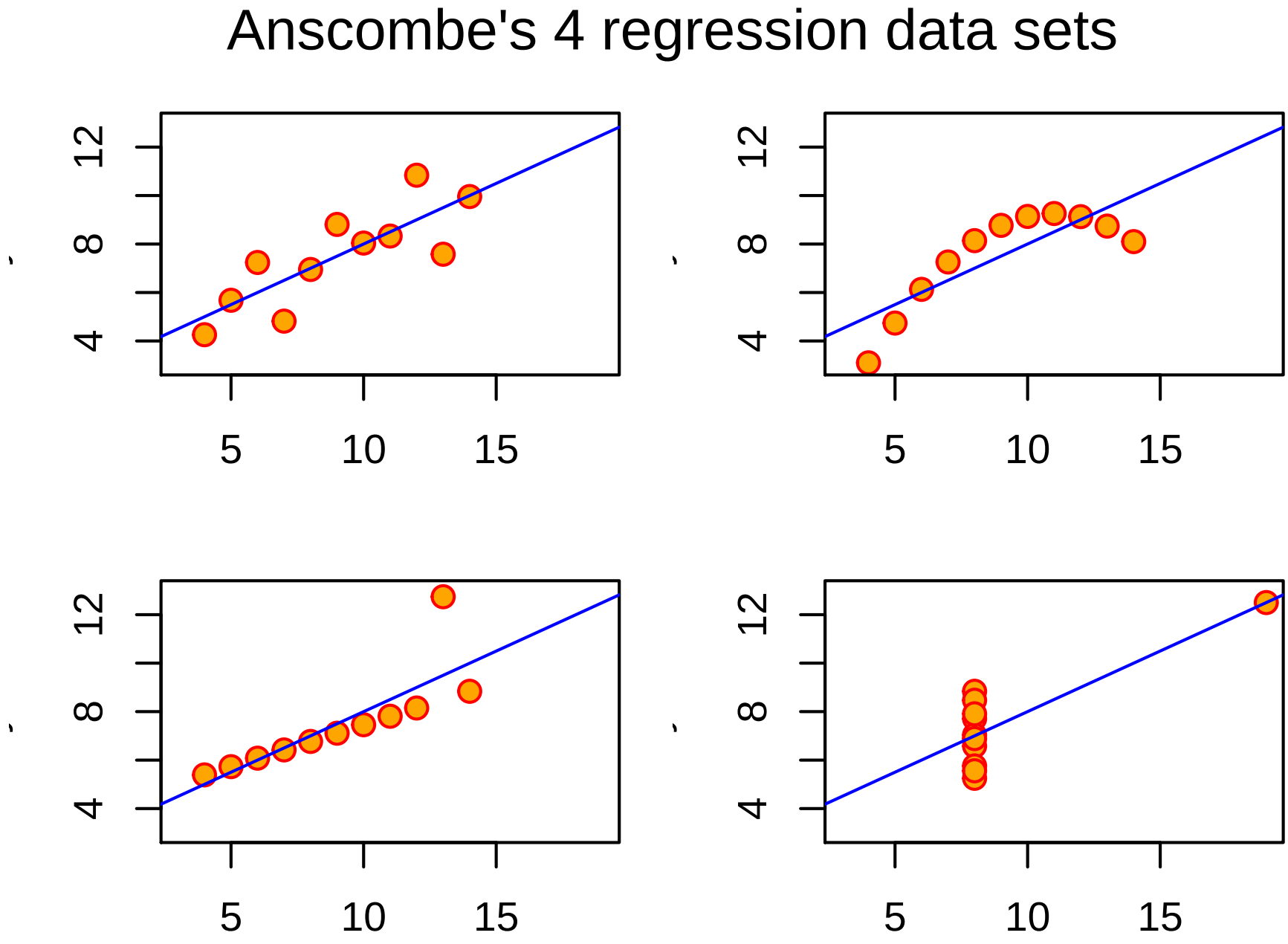
```
##                    lm1      lm2       lm3       lm4
## (Intercept) 3.0000909 3.000909 3.0024545 3.0017273
## x1          0.5000909 0.500000 0.4997273 0.4999091

## $lm1
##              Estimate Std. Error  t value    Pr(>|t|)
## (Intercept) 3.0000909  1.1247468 2.667348 0.025734051
## x1          0.5000909  0.1179055 4.241455 0.002169629
##
## $lm2
##             Estimate Std. Error  t value    Pr(>|t|)
## (Intercept) 3.000909  1.1253024 2.666758 0.025758941
## x2          0.500000  0.1179637 4.238590 0.002178816
##
## $lm3
##              Estimate Std. Error  t value    Pr(>|t|)
## (Intercept) 3.0024545  1.1244812 2.670080 0.025619109
## x3          0.4997273  0.1178777 4.239372 0.002176305
##
## $lm4
##              Estimate Std. Error  t value    Pr(>|t|)
## (Intercept) 3.0017273  1.1239211 2.670763 0.025590425
## x4          0.4999091  0.1178189 4.243028 0.002164602
```

Model Assessment

So far the sets look very similar. We have four data sets that have the same linear regression model. However, we have not looked at the data, and we may be in for a surprise!

```
op <- par(no.readonly = TRUE)
par(mfrow = c(2, 2), mar = 0.1+c(4,4,0.5,1),
          oma =  c(0, 0, 2, 0))
for(i in 1:4) {
  ff[2:3] <- lapply(paste0(c("y","x"), i), as.name)
  plot(ff, data = anscombe, col = "red", pch = 21,
       bg = "orange", cex = 1.2,
       xlim = c(3, 19), ylim = c(3, 13))
  abline(mods[[i]], col = "blue")
}
mtext("Anscombe's 4 regression data sets",
      outer = TRUE, cex = 1.2)
par(op)
```

Model Assessment



Facts About Residuals

Facts About Residuals

We have assumed that the errors $\epsilon_i, i = 1, \ldots, n$ have a centered Gaussian distribution with constant variance σ^2 and are independent.

The residuals are defined as

$$\hat{\epsilon}_i = y_i - \hat{y}_i,$$

the difference between the observed and the predicted values for $x = x_i$.

Unlike the errors, the residuals are not independent and do not have constant variance.

Facts About Residuals

They cannot be independent because we have shown that $\sum_i \hat{\epsilon}_i = 0$ and also that $\sum_i \hat{\epsilon}_i x_i = 0$.

To see that they do not have the same variance, let us calculate this parameter.

Recall that the regression parameters are given by

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$$

and once we have these parameters, the fitted values are given by

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}.$$

Facts About Residuals

The matrix $\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ is usually denoted by $\mathbf{H}$ and is known as the *hat* matrix, because it carries the observed vector $\mathbf{y}$ into the fitted values vector $\hat{\mathbf{y}}$.

$$\mathbf{H}\mathbf{y} = \hat{\mathbf{y}}$$

It is possible to show that it is a symmetric matrix.

It is the matrix of the orthogonal projection onto the column space of the design matrix:

$$\mathbf{H}\mathbf{Y} = \hat{\mathbf{Y}}.$$

Since it is a projection matrix, it is idempotent: $\mathbf{H}^2 = \mathbf{H}$.

Facts About Residuals

If h_{ij} are the elements of matrix $\mathbf{H}$ then

$$\hat{y}_i = \sum_j h_{ij} y_j.$$

Therefore, we can think of the h_{ij} as the 'weights' needed to go from the observed values to the regression values, and the bigger h_{ij} is, the more influential the observed value y_j will be in the determination of $\hat{y}_i$.

So the hat matrix gives a measure of the 'leverage' of the observations on the fitted model.

In general, the greatest impact of y_i occurs for $\hat{y}_i$ and hence we will focus on the diagonal elements of $\mathbf{H}$.

The **leverage** h_{ii} is the i-th entry in the diagonal of $\mathbf{H}$.

Facts About Residuals

Observe that

$$\begin{aligned} Cov(\hat{\boldsymbol{\epsilon}}) &= Cov(\mathbf{Y} - \hat{\mathbf{Y}}) = Cov(\mathbf{Y} - \mathbf{H}\mathbf{Y}) \\ &= Cov((\mathbf{I} - \mathbf{H})\mathbf{Y}) = (\mathbf{I} - \mathbf{H})\sigma^2\mathbf{I}(\mathbf{I} - \mathbf{H})' \\ &= \sigma^2(\mathbf{I} - \mathbf{H})(\mathbf{I} - \mathbf{H})'. \end{aligned}$$

Now, since $\mathbf{H}$ is symmetric, $\mathbf{I} - \mathbf{H}$ is also symmetric and it is easy to see that $(\mathbf{I} - \mathbf{H})^2 = \mathbf{I} - \mathbf{H}$ so $\mathbf{I} - \mathbf{H}$ is also idempotent.

Therefore, we get that $Var(\hat{\epsilon}_i) = \sigma^2(\hat{\epsilon}_i) = \sigma^2(1 - h_{ii})$.

Since the h_{ii} need not be equal, we see that the residuals do not have the same variance.

Also, since $\mathbf{H}$ need not be a diagonal matrix, the $\hat{\epsilon}_i$ are usually correlated and not independent.

Facts About Residuals

It is possible to prove that $0 \leq h_{ii} \leq 1$ and that

$$h_{ii} = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_i (x_i - \bar{x})^2}.$$

The **standardized** residuals are defined as

$$t_i = \frac{\hat{\epsilon}_i}{\hat{\sigma}(1 - h_{ii})^{1/2}}$$

where $\hat{\sigma}$ is the estimated error standard deviation.

Homoscedasticity and Linearity

Homoscedasticity and Linearity

The first graph that is usually drawn for model evaluation is a plot of residuals against fitted values

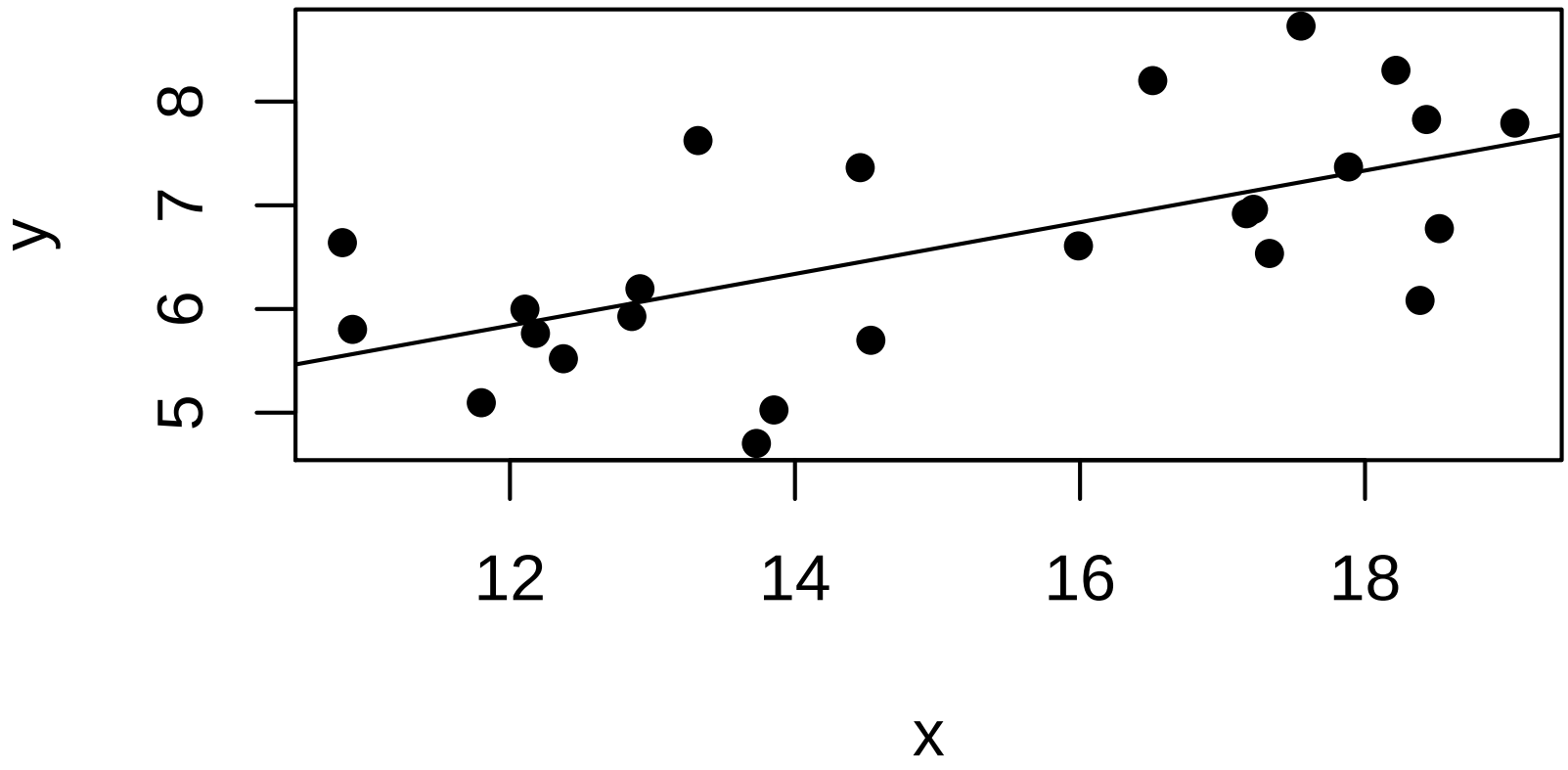
If the model assumptions are correct and we have captured as much variability as possible with the model, in this graph we would expect to see no patterns, but points scattered at random over the plotting region.

The presence of patterns may indicate that the assumption of equal variance (homoscedasticity) does not hold, or that there are still possible improvements in the model.

Let us see an example of this situation with simulated data. This will be `modelA`.

Homoscedasticity and Linearity

```
set.seed(456);xx <- runif(25,10,20);zz <- rnorm(25)
y1 <- 2 + 0.3*xx+zz
plot(xx,y1,pch=16, xlab='x', ylab='y')
modelA <- lm(y1~xx);abline(modelA)
```



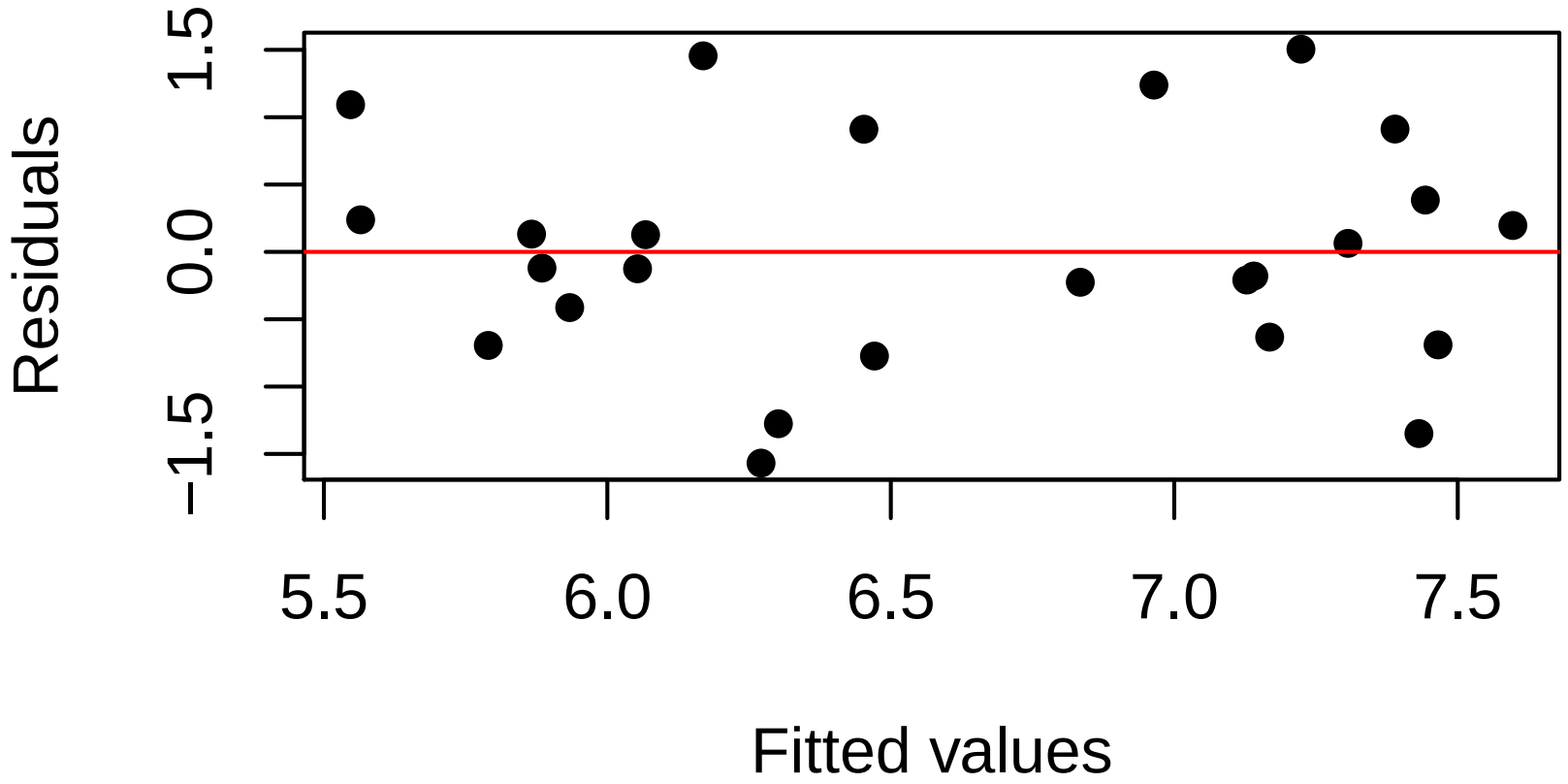
Homoscedasticity and Linearity

```
summary(modelA)
```

```
##
## Call:
## lm(formula = y1 ~ xx)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5680 -0.6330 -0.1202  0.3848  1.5047
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.84963    0.99632   2.860 0.008853 **
## xx           0.24921    0.06488   3.841 0.000834 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8659 on 23 degrees of freedom
## Multiple R-squared:  0.3908, Adjusted R-squared:  0.3643
## F-statistic: 14.76 on 1 and 23 DF,  p-value: 0.0008336
```

Homoscedasticity and Linearity

```
plot(fitted(modelA), resid(modelA), pch=16,
     xlab='Fitted values', ylab='Residuals')
abline(h=0, col='red')
```

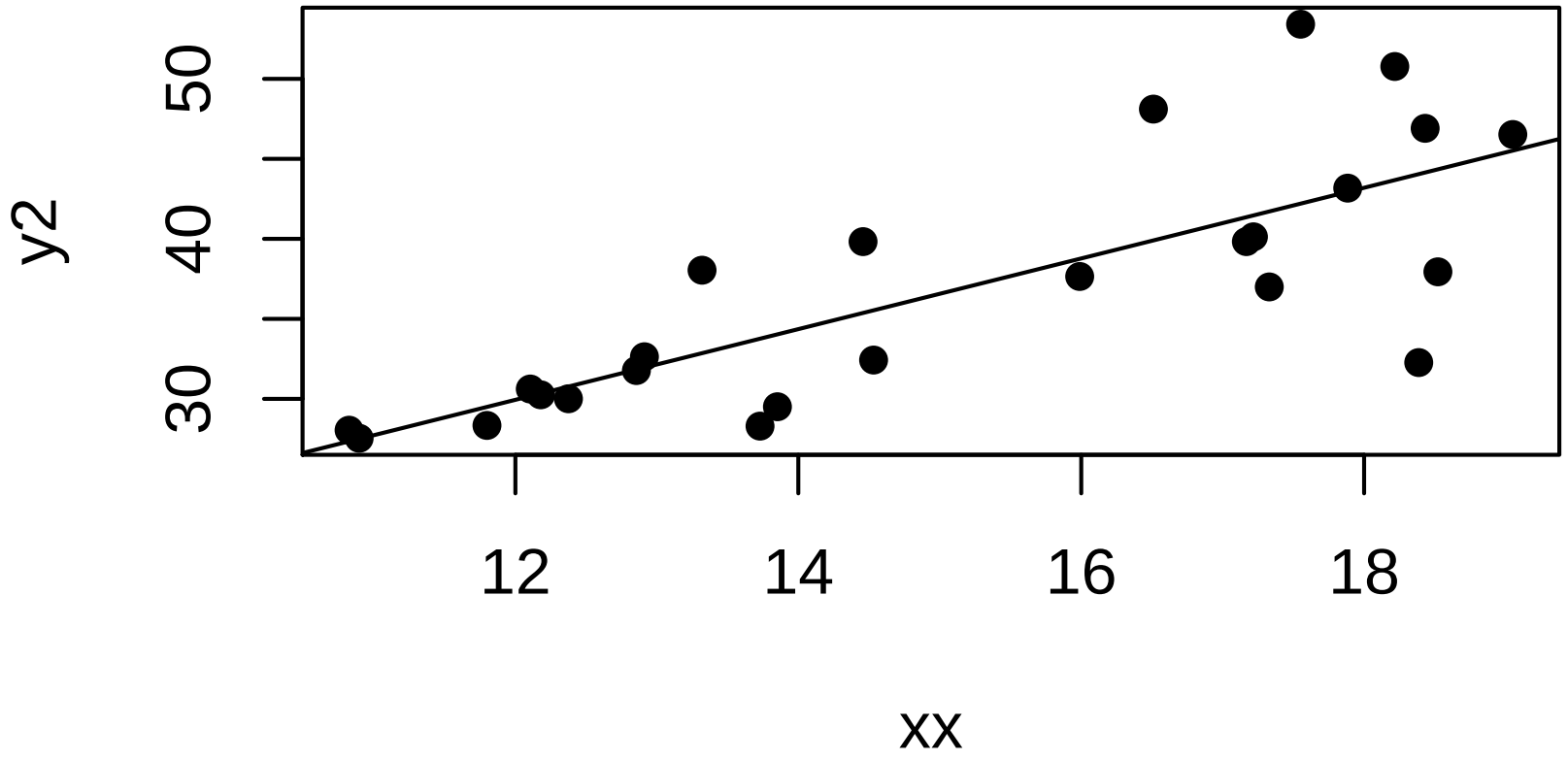


In this case the points are randomly distributed. We see no patterns in the dots.

Homoscedasticity and Linearity

One of the situations that is frequently encountered is that the variance increases or decreases with fitted values, as in the next two examples. The first one is `modelB`.

```
y2 <- 2 + 2.3*xx+((xx-10))*zz; plot(xx,y2,pch=16)
modelB <- lm(y2~xx); abline(modelB)
```



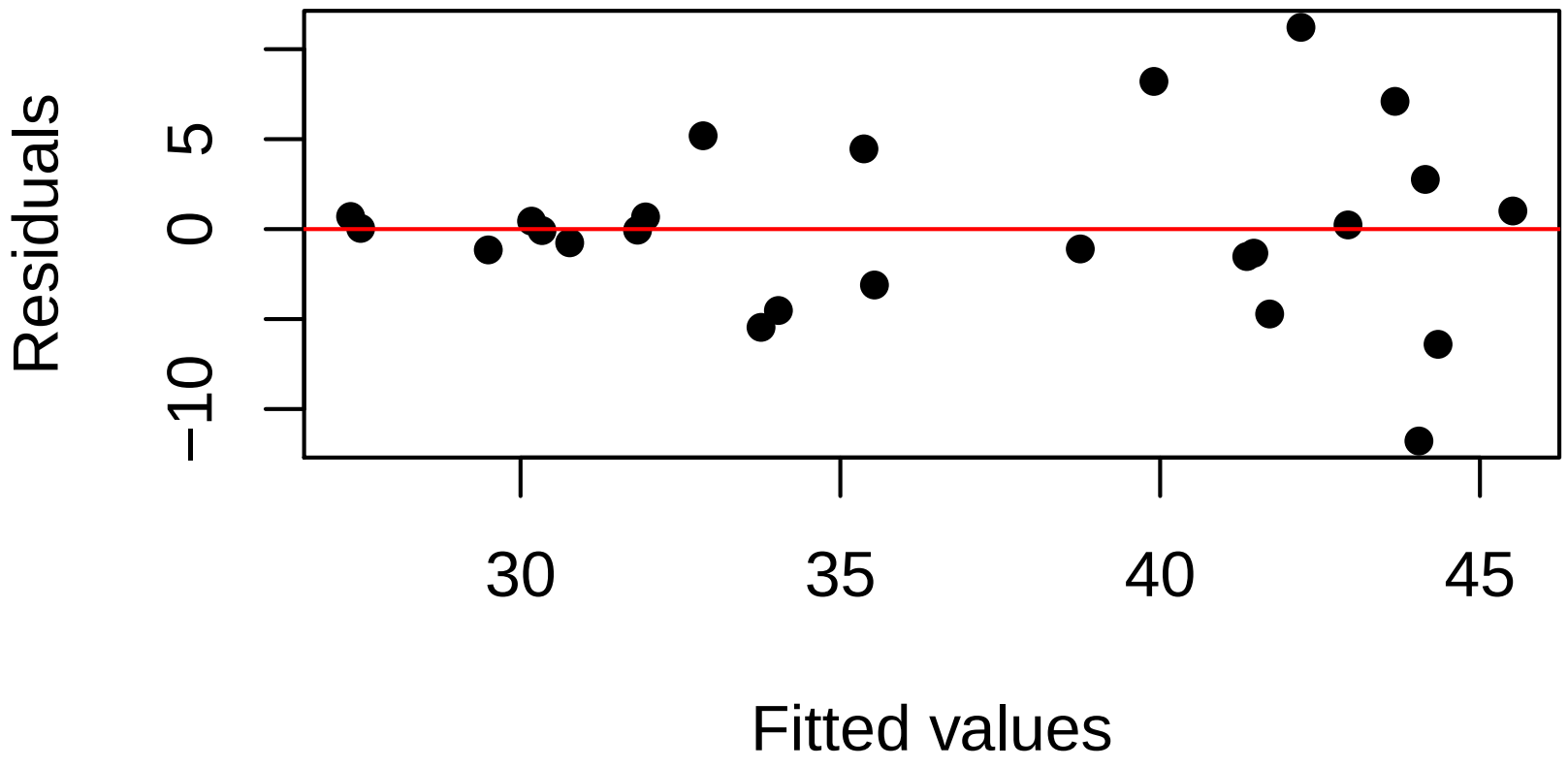
Homoscedasticity and Linearity

```
summary(modelB)
```

```
##
## Call:
## lm(formula = y2 ~ xx)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.7769  -1.5265  -0.0589   1.0082  11.2117
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.4322     5.7244   0.600    0.555
## xx            2.2090     0.3727   5.926 4.84e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.975 on 23 degrees of freedom
## Multiple R-squared:  0.6043, Adjusted R-squared:  0.5871
## F-statistic: 35.12 on 1 and 23 DF,  p-value: 4.843e-06
```

Homoscedasticity and Linearity

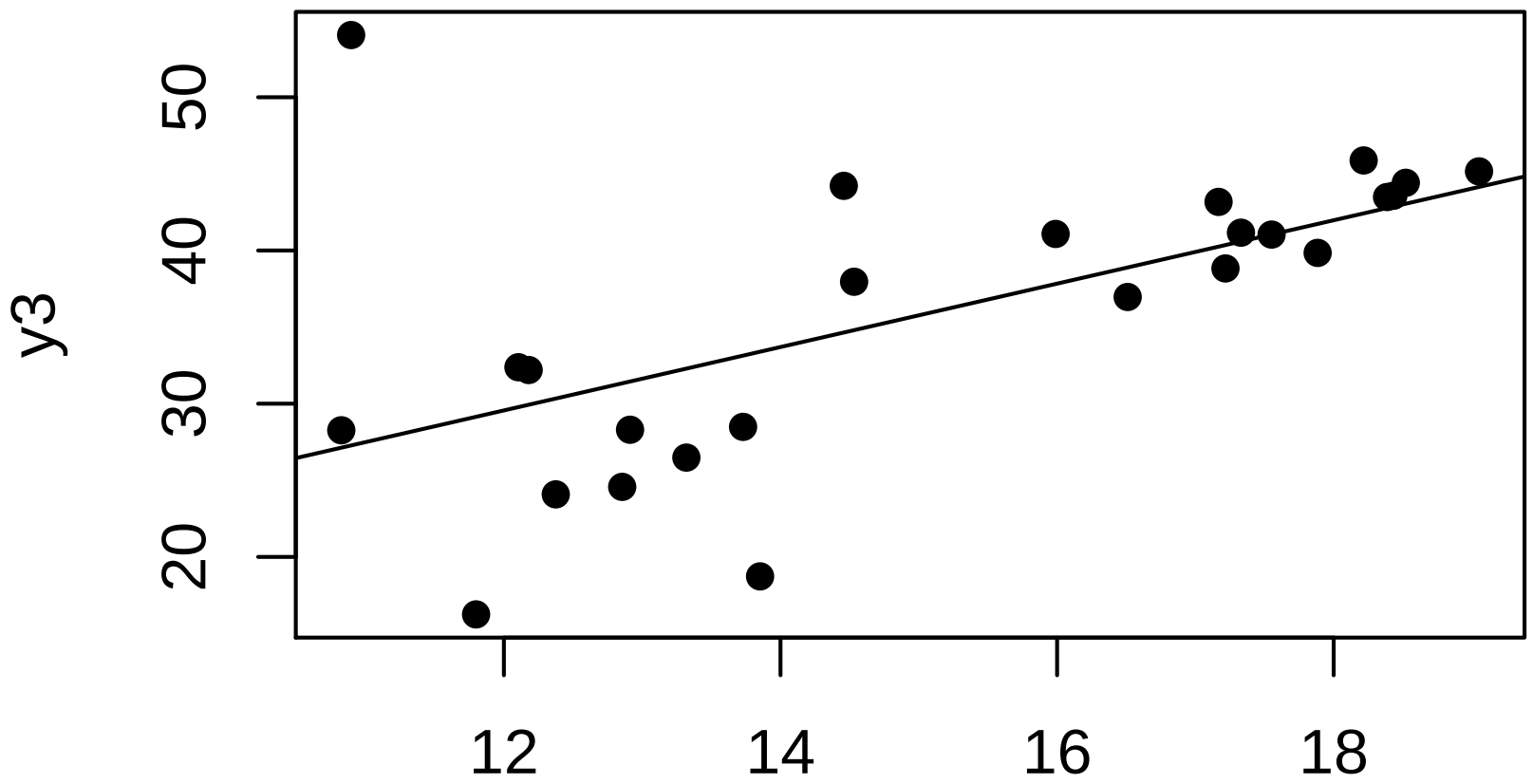
```
plot(fitted(modelB), resid(modelB), pch=16,
     xlab='Fitted values', ylab='Residuals')
abline(h=0, col='red')
```



Homoscedasticity and Linearity

In this example we see that the residuals 'open up' as the fitted values increase in value. In the next example, `modelC`, the reverse situation happens.

```
z3 <- rnorm(25); y3 <- 2 + 2.3*xx + ((xx-20))*z3
plot(xx,y3,pch=16); modelC <- lm(y3~xx)
abline(modelC)
```



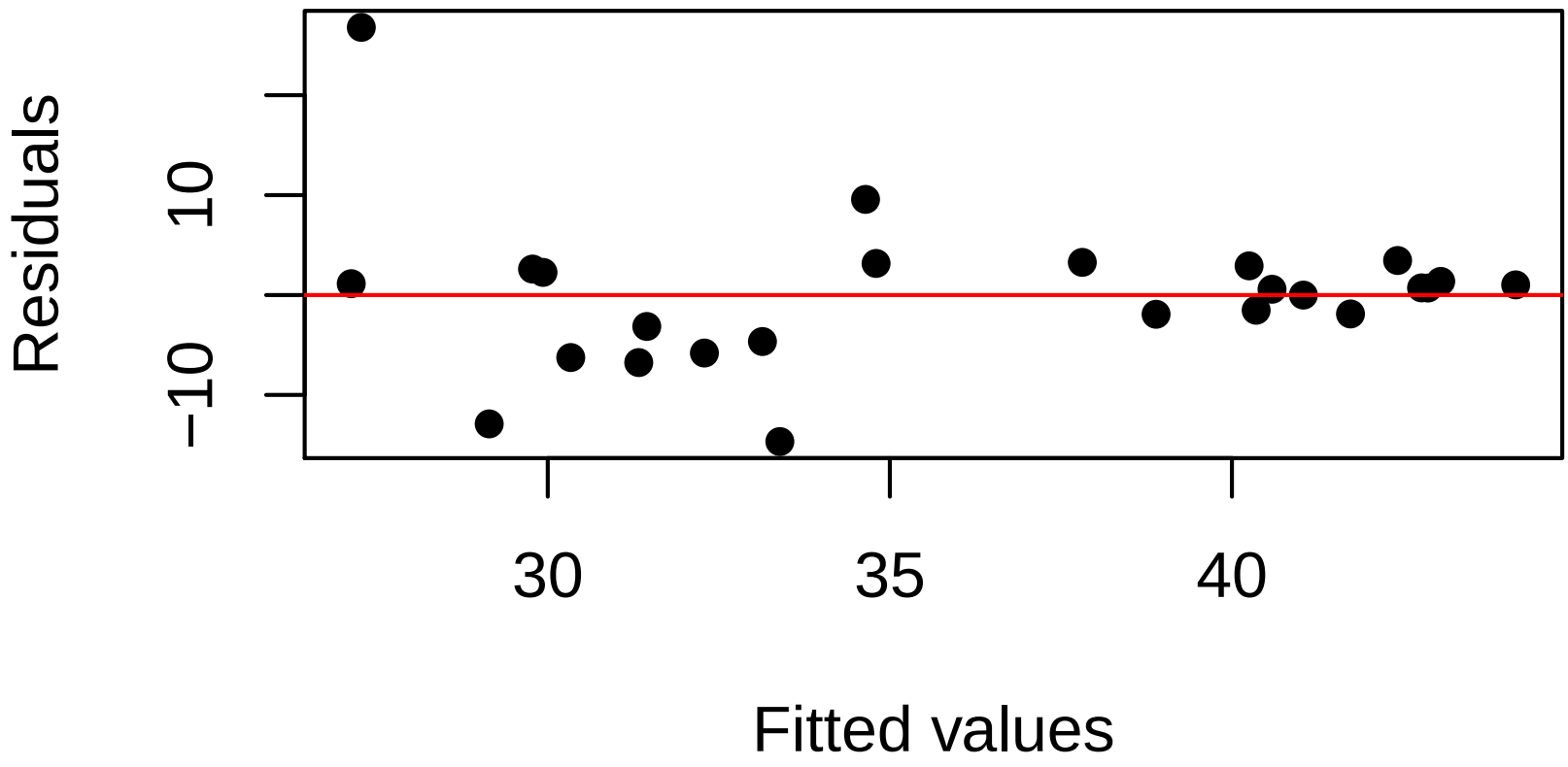
Homoscedasticity and Linearity

```
summary(model3)
```

```
##
## Call:
## lm(formula = y3 ~ xx)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.6647  -3.1477   0.7031   2.6000  26.7857
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.7352     9.0057   0.526   0.6041
## xx            2.0687     0.5864   3.528   0.0018 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.826 on 23 degrees of freedom
## Multiple R-squared:  0.3511, Adjusted R-squared:  0.3229
## F-statistic: 12.45 on 1 and 23 DF,  p-value: 0.001802
```

Homoscedasticity and Linearity

```
plot(fitted(modelC), resid(modelC), pch=16,
     xlab='Fitted values', ylab='Residuals')
abline(h=0, col='red')
```



Homoscedasticity and Linearity

In both cases we have a 'funnel' shape, although with different orientations. This is an indication that the variance is not constant.

A possible way to deal with this problem is to transform the data. Useful transformations in this case are the Box-Cox transformations. We won't go into any detail about this but for positive data the transformations are given by

$$T_\lambda(y) = \begin{cases} \frac{y^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \log(y) & \text{if } \lambda = 0 \end{cases}$$

The command `boxcox` in `R` calculates the optimal transformation for a given data set.

Homoscedasticity and Linearity

The graphs of residuals against fitted values are also useful to detect cases in which the model does not explain all the structure present in the data.

Example Q

As an example let us consider a quadratic relation between two variables that we try to model as a linear relation.

```
set.seed(4567)
xx <- runif(25,10,20)
zz <- rnorm(25,sd=4)
y4 <-  2 + 1.3*xx+ 3*(xx-10)^2+zz
modelD <- lm(y4~xx)
```

Homoscedasticity and Linearity

```
summary(modelD)
```

```
##
## Call:
## lm(formula = y4 ~ xx)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -28.14 -20.18 -13.02  23.87  48.93
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -350.746     26.814  -13.08 3.88e-12 ***
## xx            31.569      1.718   18.37 3.06e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.13 on 23 degrees of freedom
## Multiple R-squared:  0.9362, Adjusted R-squared:  0.9334
## F-statistic: 337.5 on 1 and 23 DF,  p-value: 3.057e-15
```

Homoscedasticity and Linearity

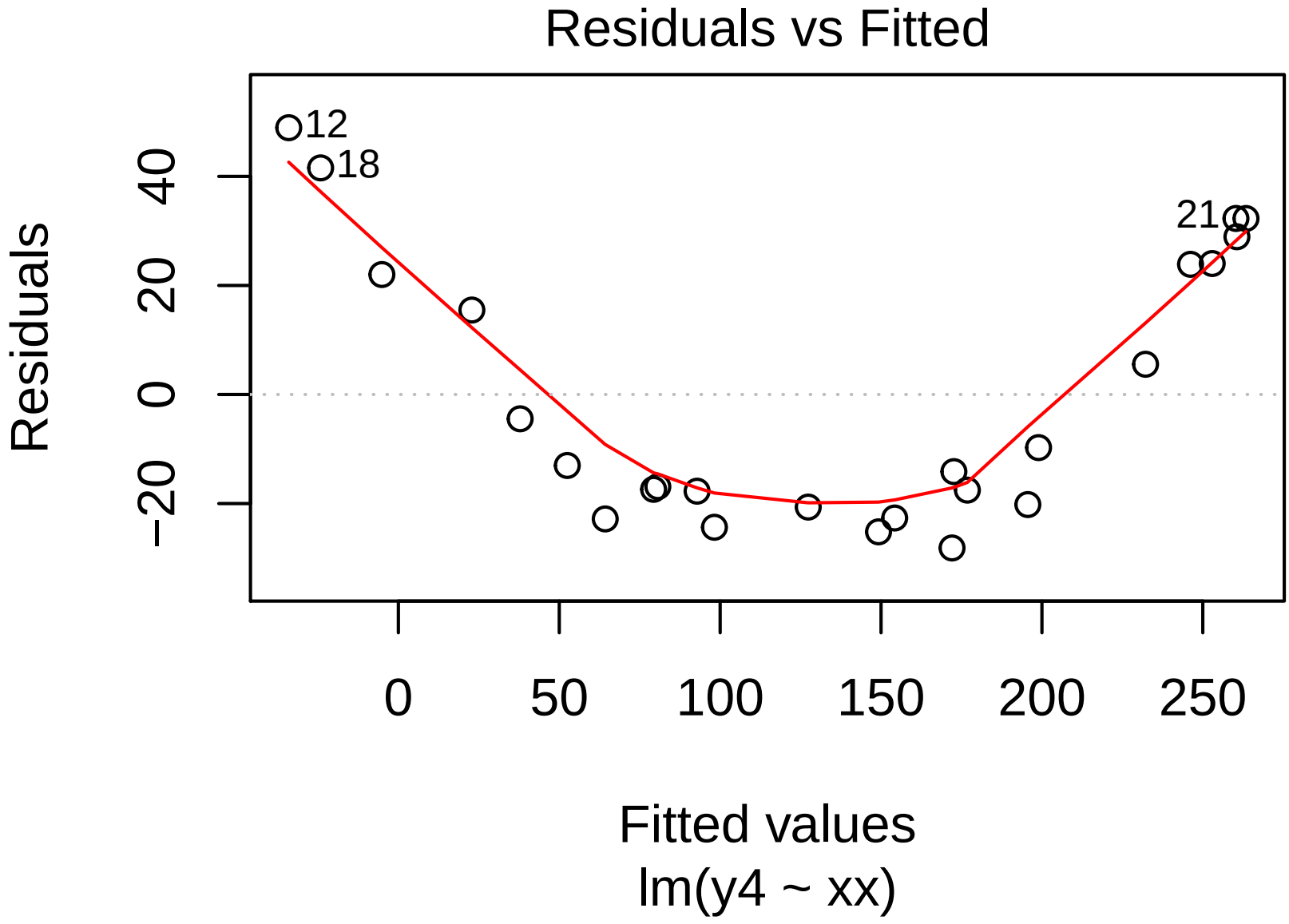
In the summary for the regression we see that slope and intersect are significant with a very low p-value and that the coefficient of determination R^2 has a (high) value of 0.936.

However, if we look at the summary data for the residuals, we see that the values do not correspond to a symmetric distribution, as one would expect if they followed a (centered) normal distribution.

The evaluation graphs obtained with the instructions below, show that there is a quadratic structure in the data that has not been taken into account.

Homoscedasticity and Linearity

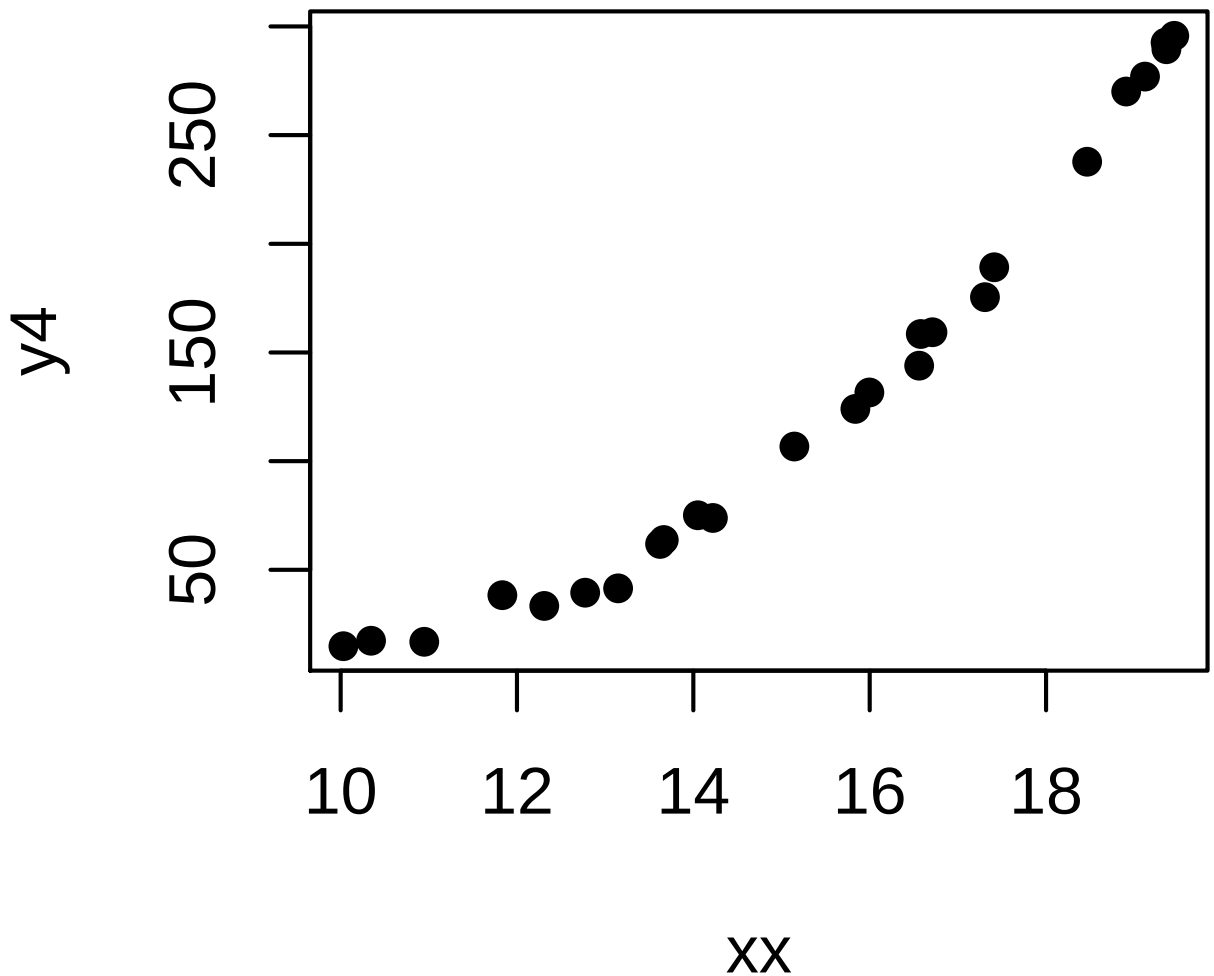
```
plot(modelD, which = 1)
```



Homoscedasticity and Linearity

Indeed, if we had looked at the data in the first place -something one should always do- we would have seen that a linear relation is not adequate for this data.

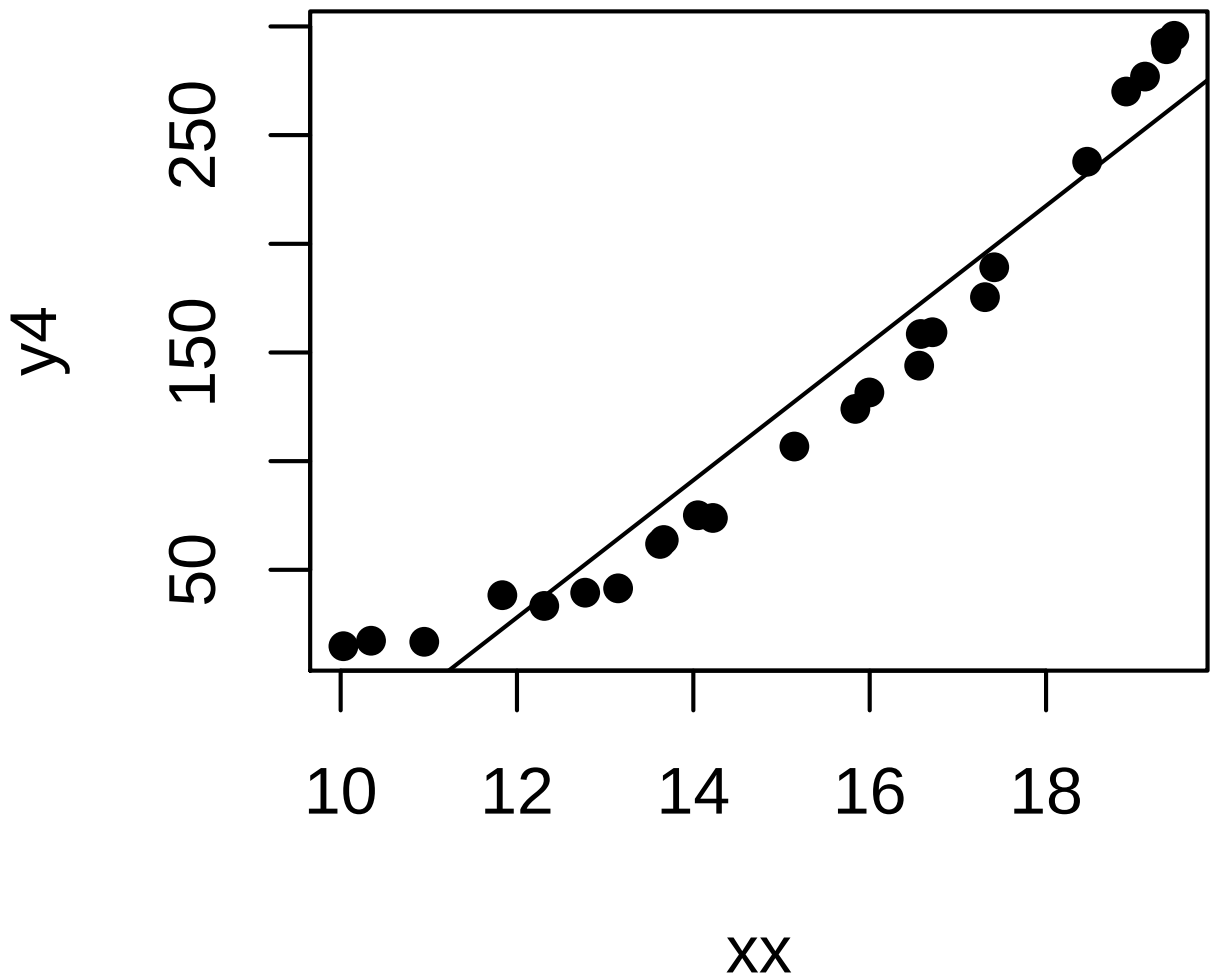
```
plot(xx,y4,pch=16)
```



Homoscedasticity and Linearity

Indeed, if we had looked at the data in the first place -something one should always do- we would have seen that a linear relation is not adequate for this data.

```
plot(xx,y4,pch=16); abline(modelD)
```



Homoscedasticity and Linearity

We can add a quadratic term to the regression to include this structure into account. We will look at multiple regression in detail later on, but for completeness, let's fit a quadratic model.

```
modelE <- lm(y4~xx+I(xx^2))
summary(modelE)
```

```
##
## Call:
## lm(formula = y4 ~ xx + I(xx^2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.2420 -1.8221  0.0683  2.8383  9.8580
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 309.9679    26.9643   11.50 9.03e-11 ***
## xx          -59.5669     3.6675  -16.24 9.81e-14 ***
## I(xx^2)       3.0235     0.1212   24.95  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.747 on 22 degrees of freedom
## Multiple R-squared:  0.9978, Adjusted R-squared:  0.9976
## F-statistic:  5039 on 2 and 22 DF,  p-value: < 2.2e-16
```

Homoscedasticity and Linearity

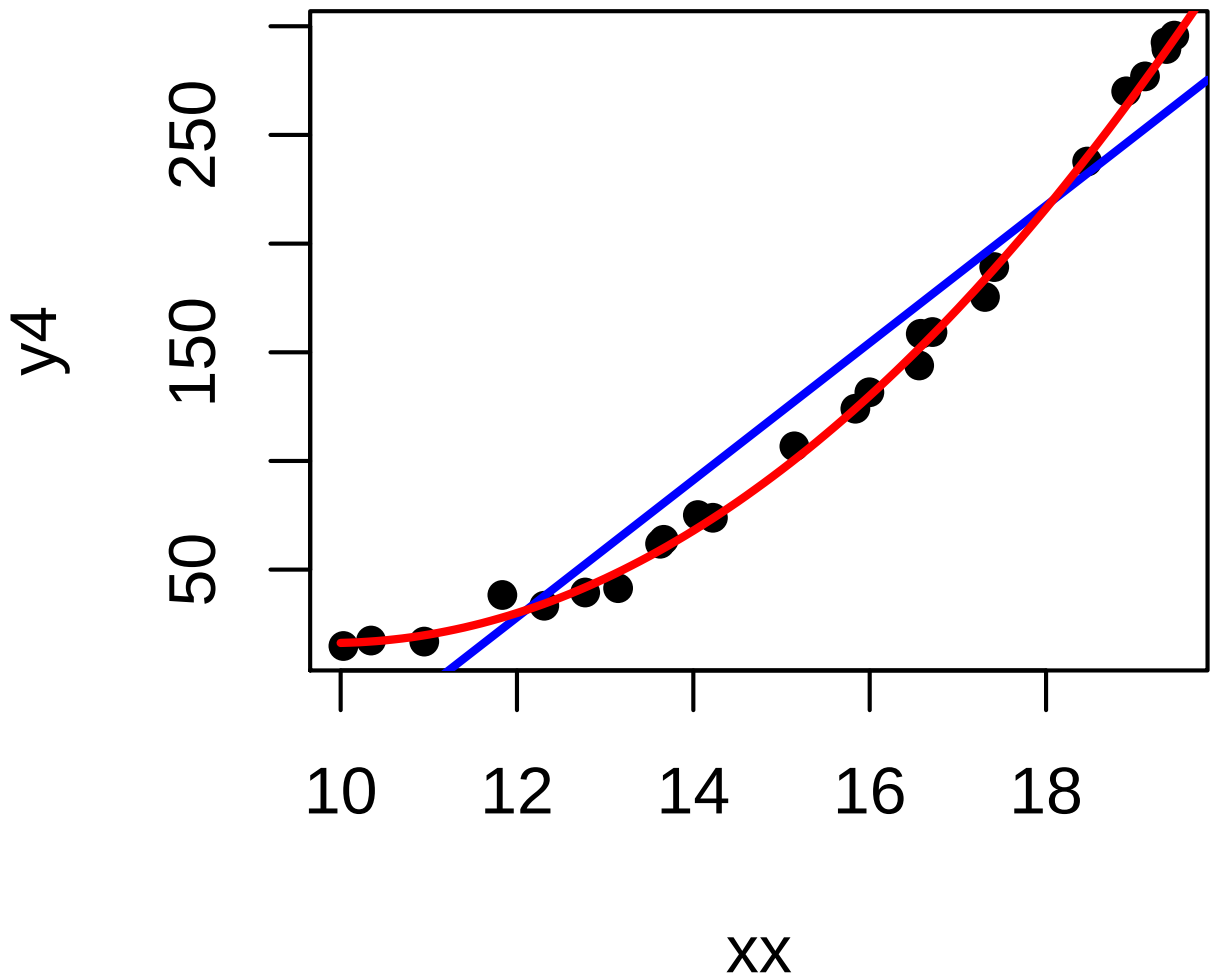
From the summary data for the regression we see that linear and quadratic terms are significant and that the summary data for the residuals is consistent with a symmetric distribution.

Also, the R^2 has increased to 0.998.

Next we plot the data, the regresion line (from the first regression model) and the quadratic curve we have just fitted.

Homoscedasticity and Linearity

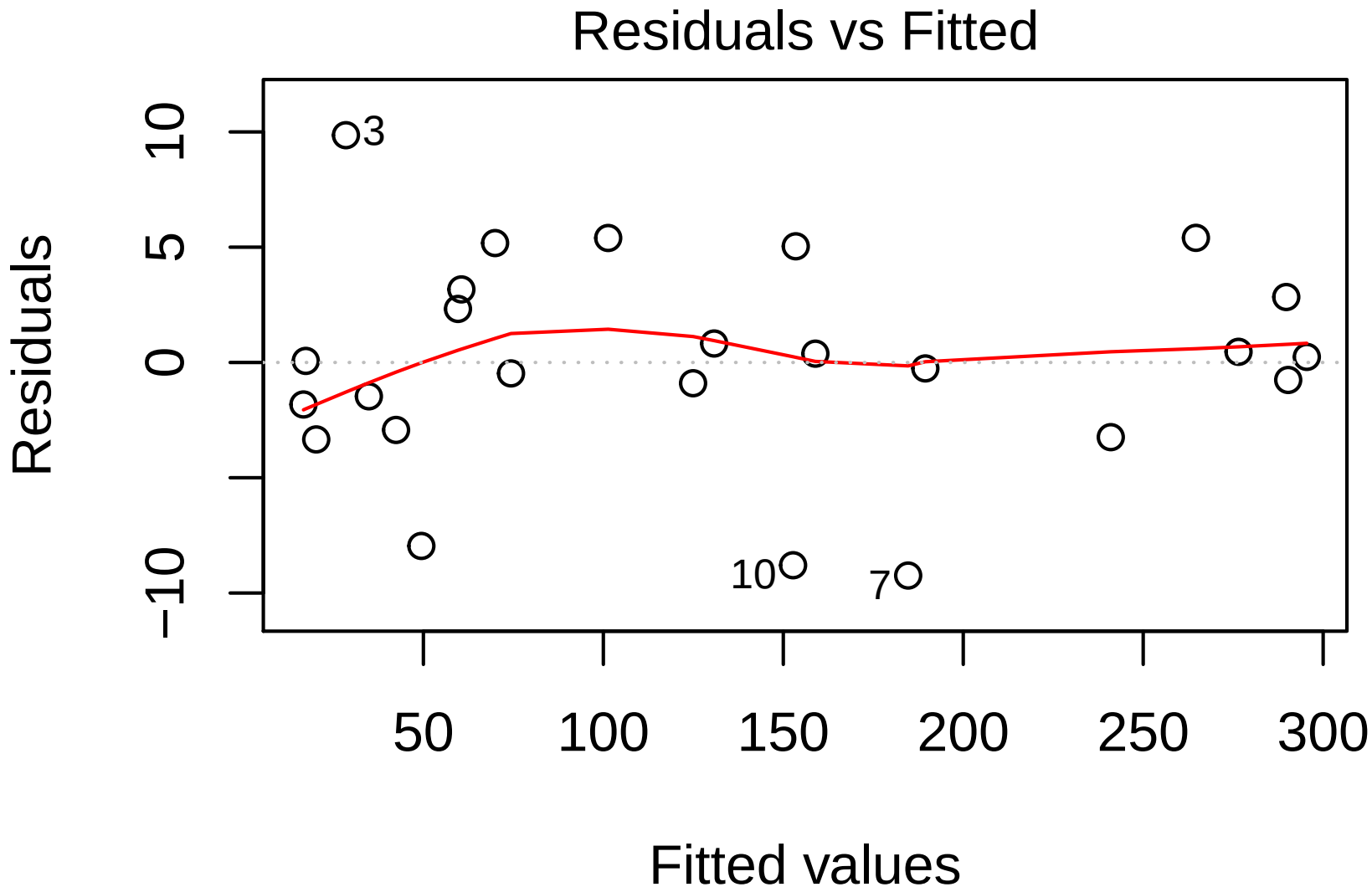
```
plot(xx,y4,pch=16)
abline(modelD, col='blue', lwd=2)
curve(309.99-59.57*x+3.02*x^2,10,20, add=T,
      col='red', lwd=2)
```



Homoscedasticity and Linearity

Finally, the graphs to evaluate the new model look much better than the those for the previous model.

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



Gaussianity

Gaussianity

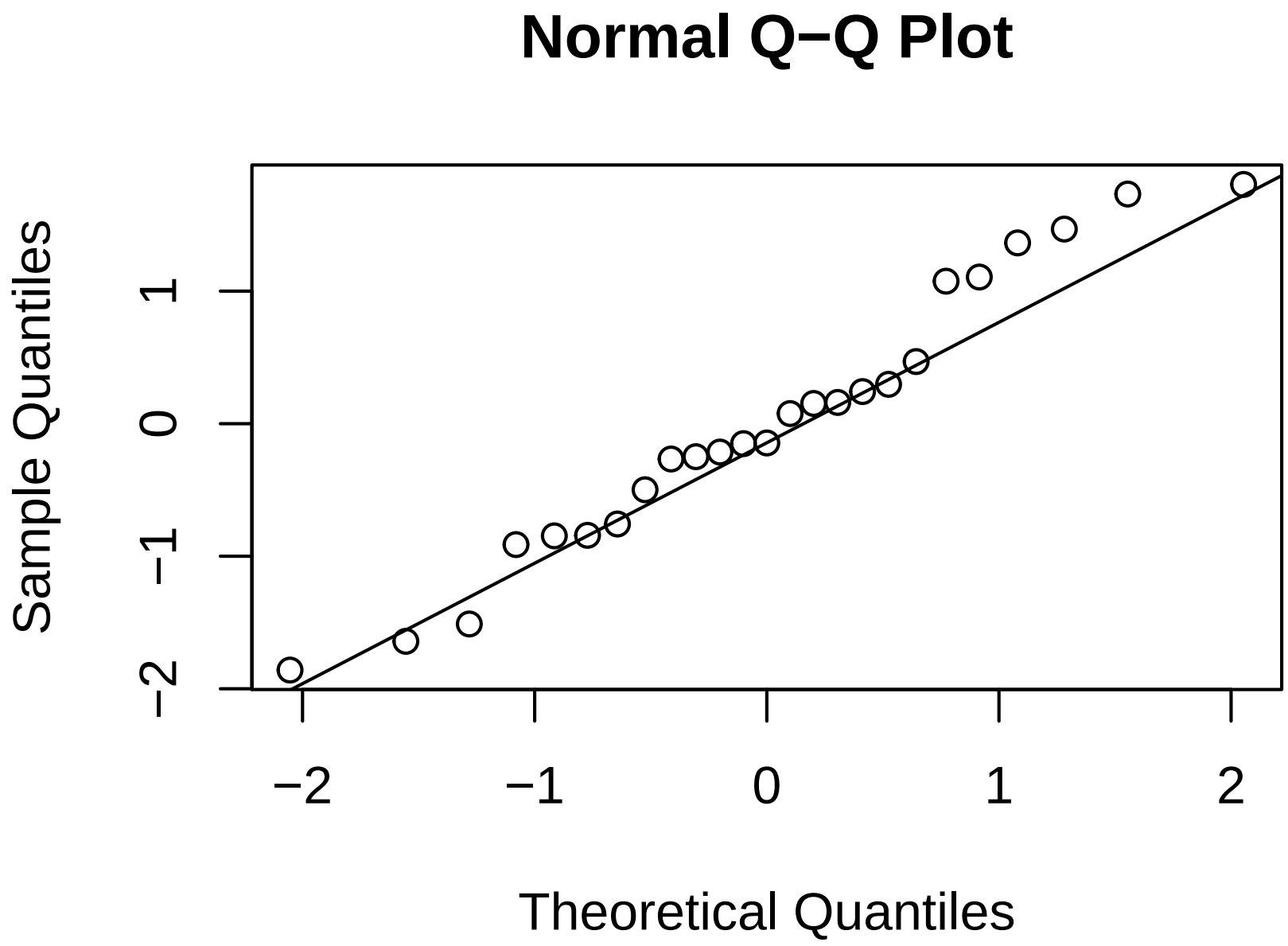
Another important assumption we have made is that the errors have a normal distribution.

To check this assumption it is usual to draw a quantile plot for the residuals. However, since we have seen that the residuals do not have constant variance, it is usual to plot the standardized residuals.

In `R`, standardized residuals are obtained with the `rstandard` command acting on an `lm` object

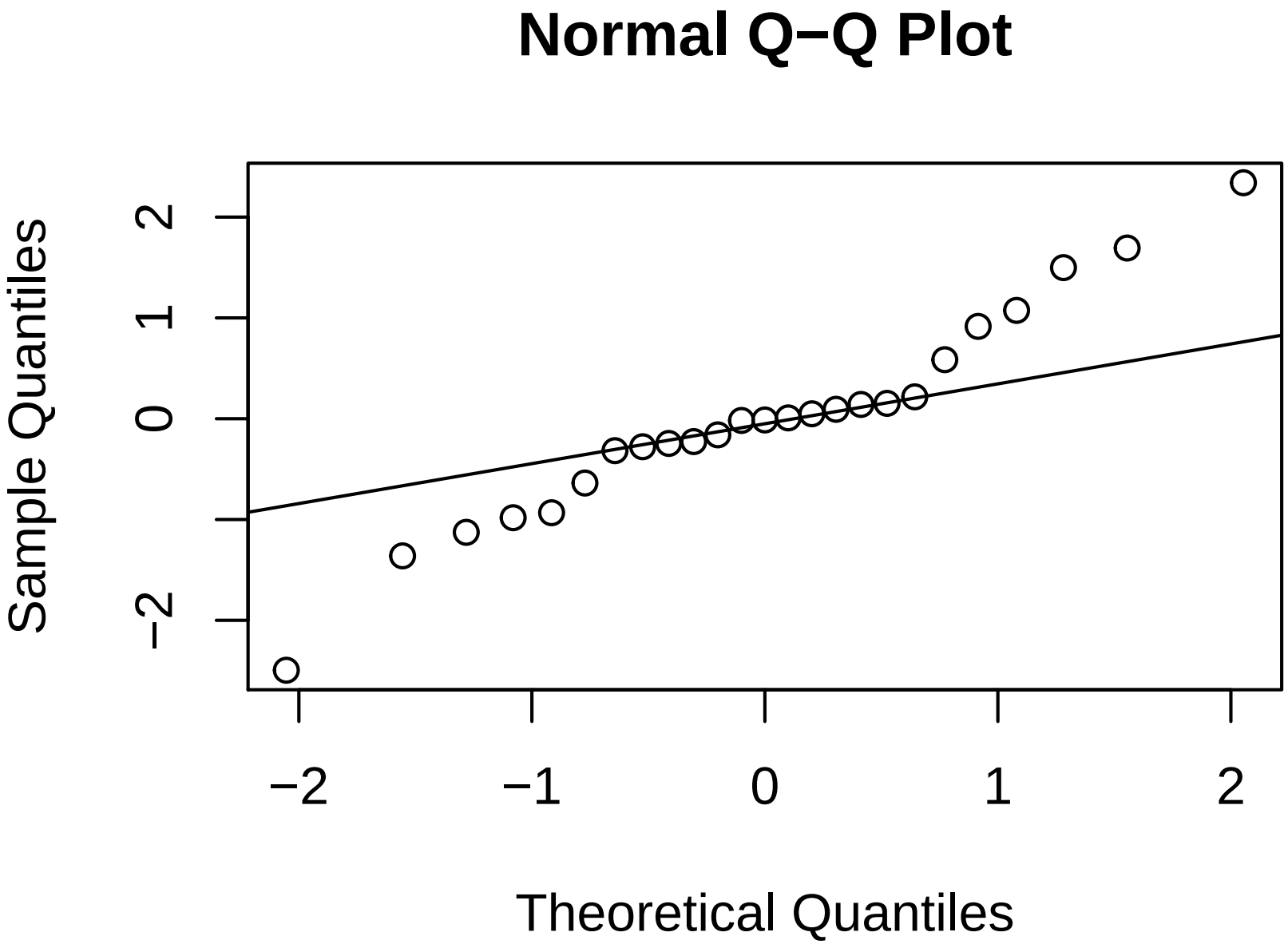
Gaussianity

```
qqnorm(rstandard(modelA)); qqline(rstandard(modelA))
```



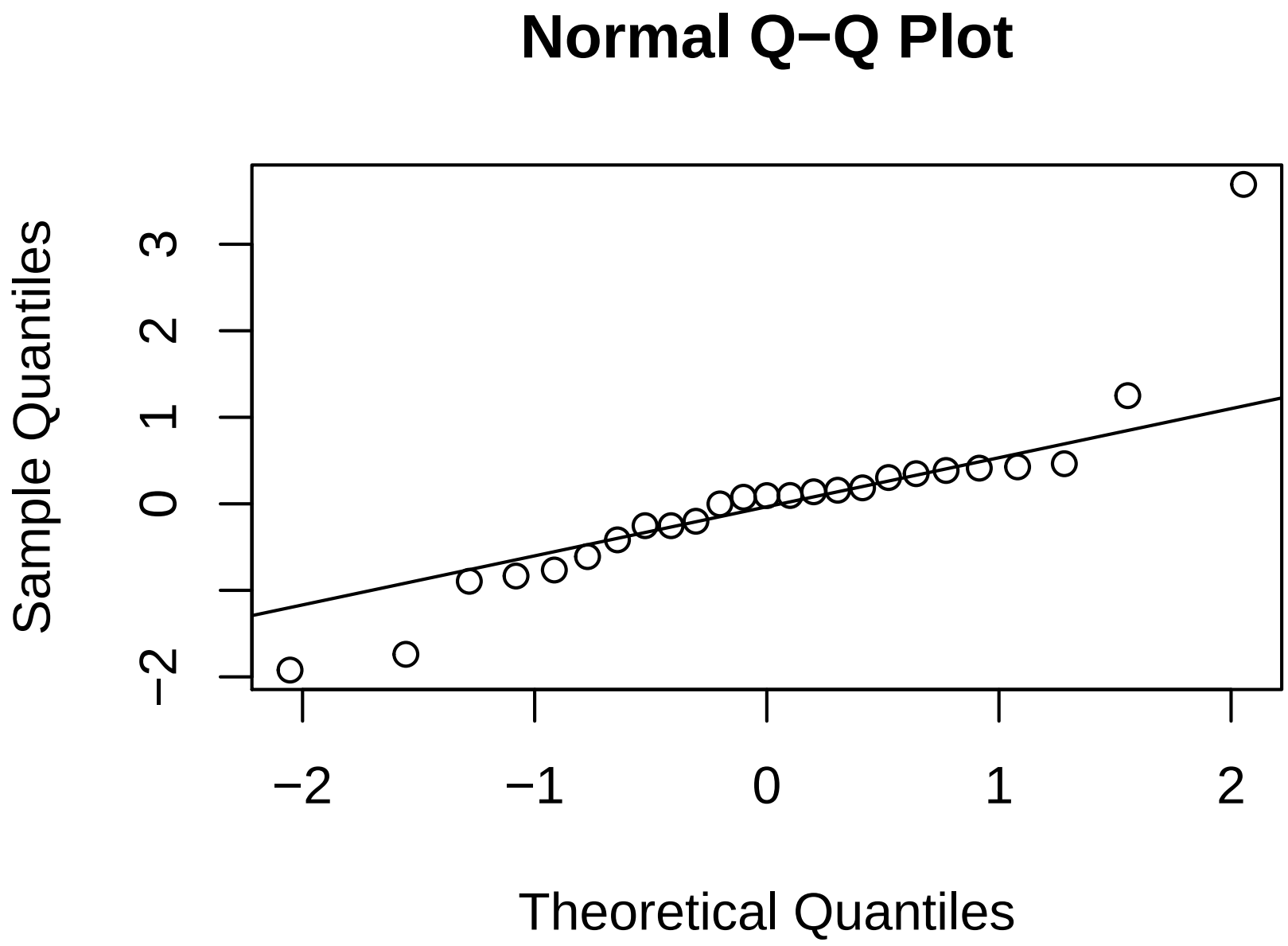
Gaussianity

```
qqnorm(rstandard(modelB)); qqline(rstandard(modelB))
```



Gaussianity

```
qqnorm(rstandard(modelC)); qqline(rstandard(modelC))
```



Gaussianity

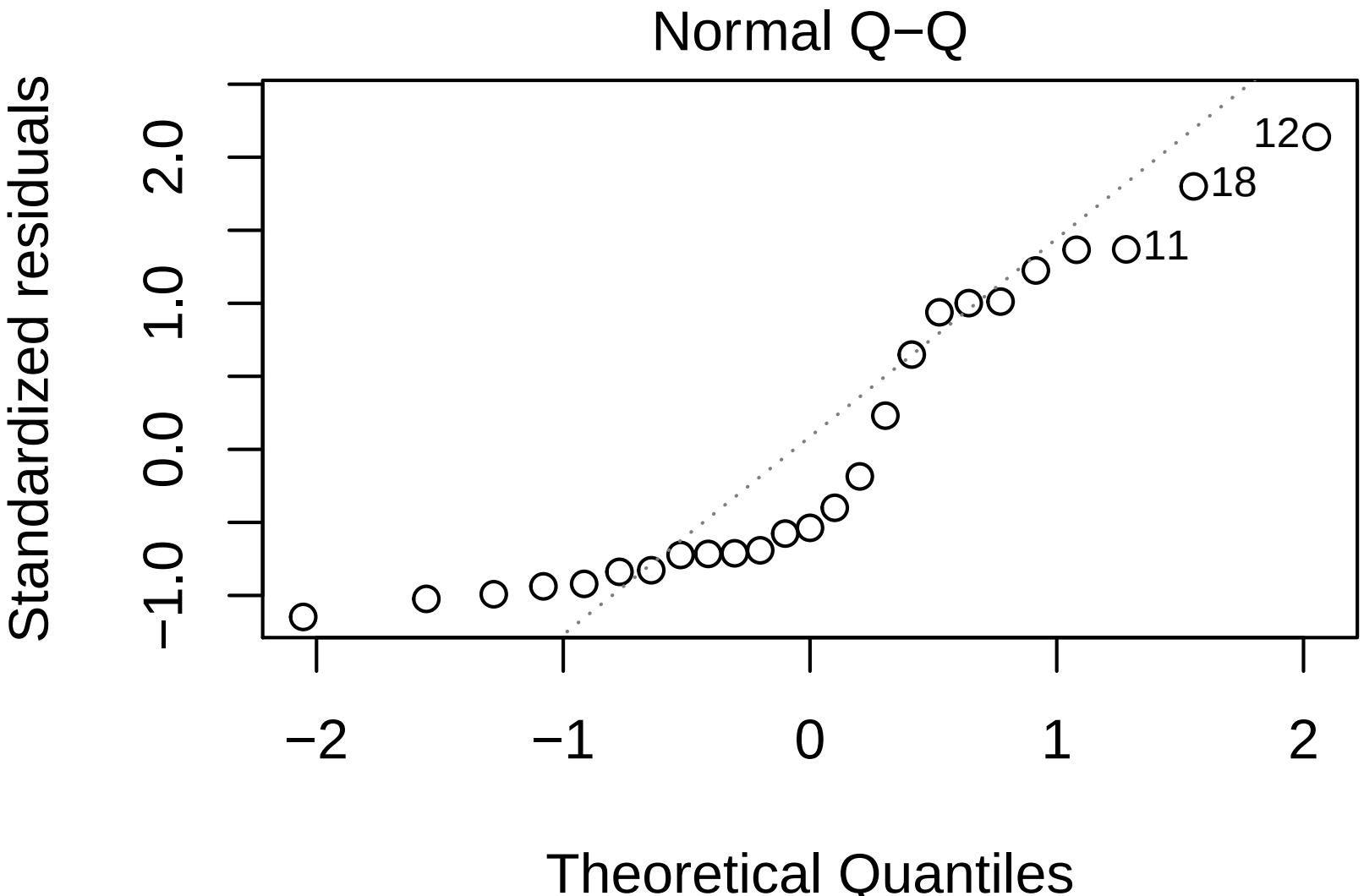
In these graphs we look at the fit of the points to the line. In the last two graphs we see that the central part of the data fits well but in the tails there are deviations from the reference line.

When compared with normal distribution, that has light tails, this means that the sample has heavier tales.

Gaussianity: Example Q

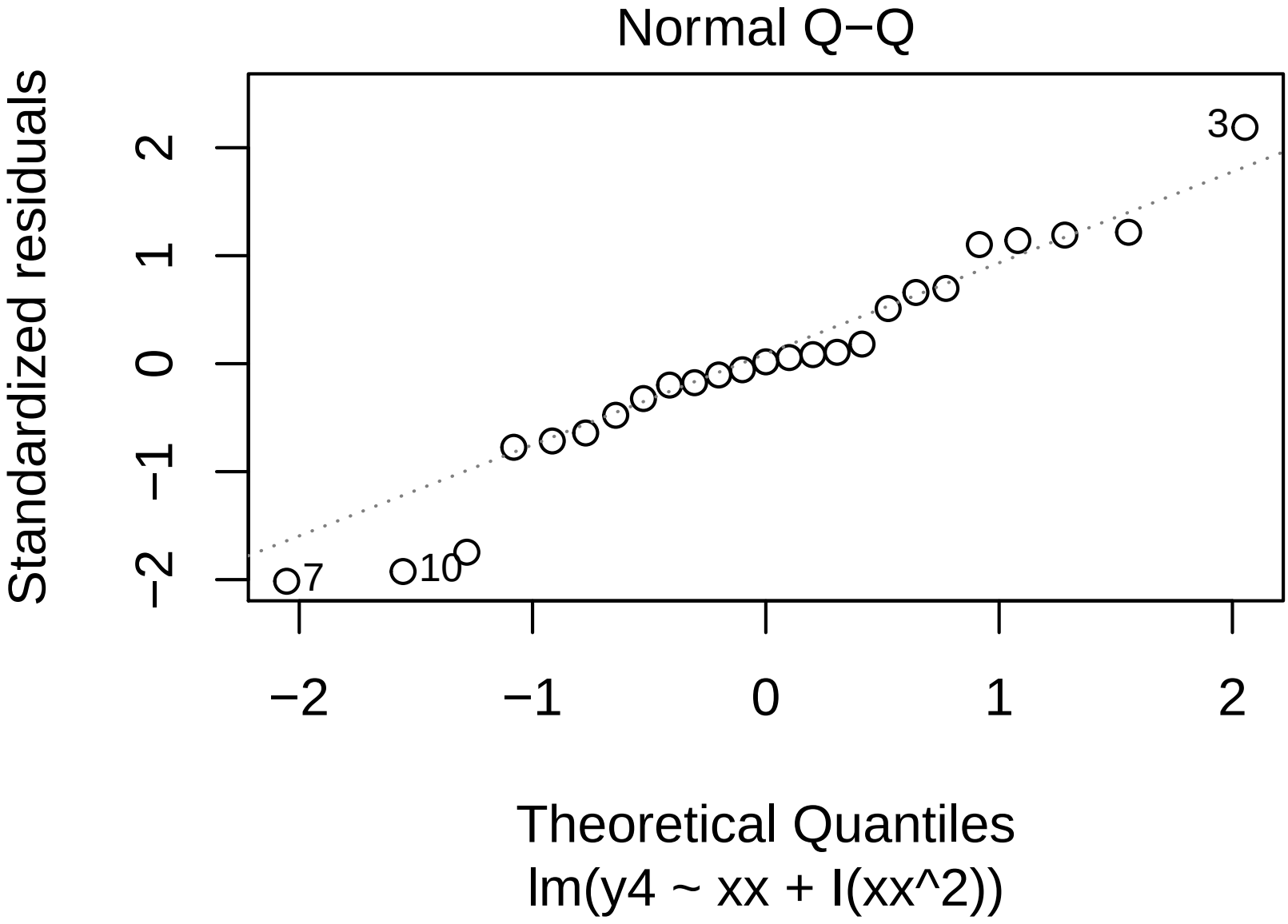
Let's go back to example Q and graph the quantiles plots before and after adding the quadratic term.

```
plot(modelD, which = 2)
```



Gaussianity: Example Q

```
plot(modelE, which = 2)
```



Diagnostic Plots

A third graph that is also useful for detecting departures from the assumptions, is similar to the first one on a different scale.

Instead of the residuals, the square root of the absolute value of the standardized residuals is plotted against fitted values, so all values in the y axis are positive.

Again, we expect to see no structure or patterns, but random points scattered on the graph.

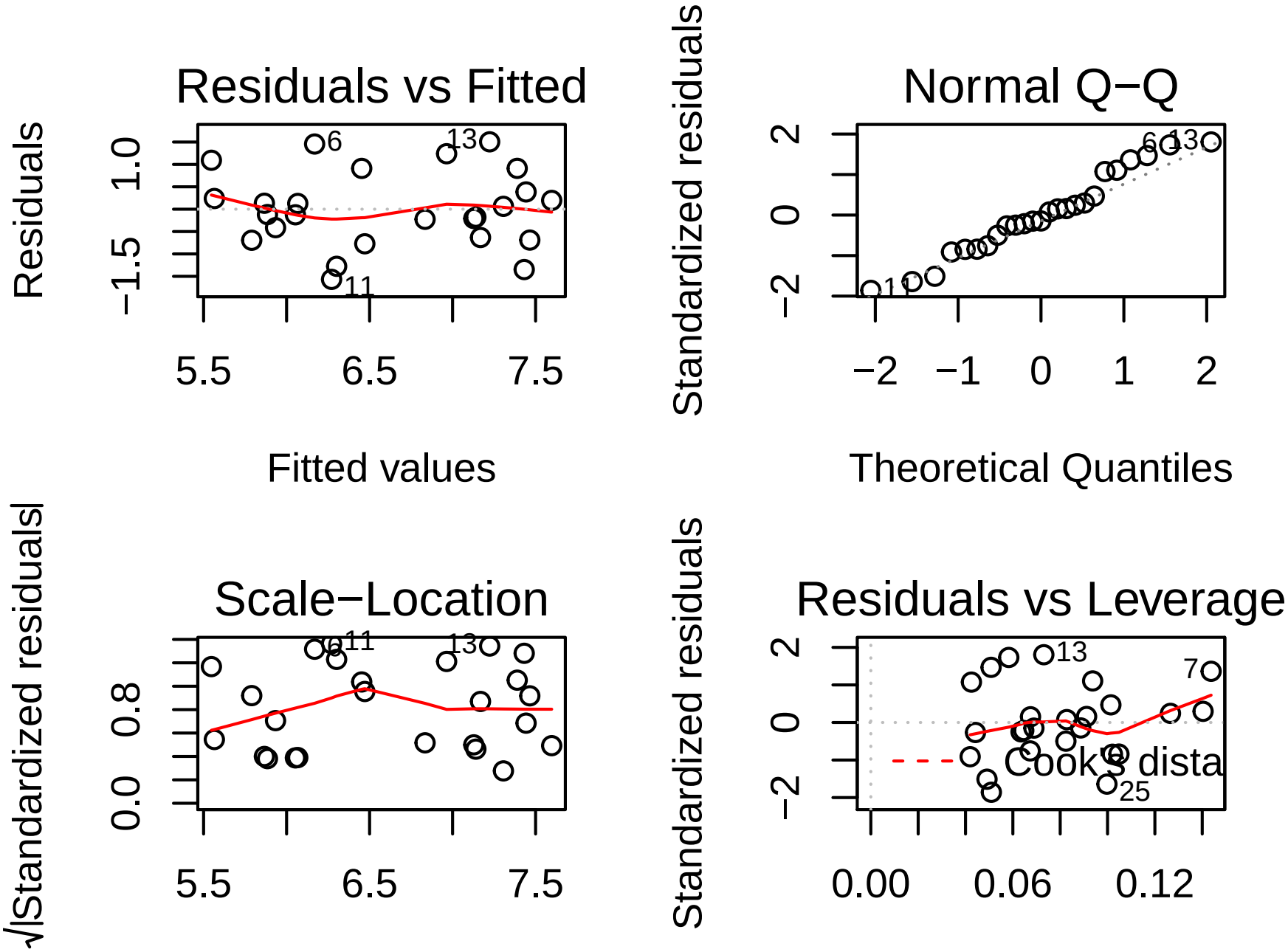
Additionally, since the residuals have been standardized, large values indicate possible atypical points.

Diagnostic Plots

Finally, a graph of standardized residuals against leverage is usually drawn. This plot highlights the values that have highest influence on the parameter estimates.

As we have seen before, these four graphs can be obtained from an `lm` object using the `plot` function if the screen has been previously partitioned into four, as the following instructions illustrate for the first two models we fitted previously.

Gaussianity



Gaussianity