# Linear statistical models
# The full rank model

Yao-ban Chan

Lec Monday March 15

Yao-ban commented on Assignment I

1. To be consistent, I will suggest you to watch 0 minutes ⟷ 13:39
2. Gradescope: submit earlier

# Linear models

We remind ourselves what a linear model is:

- We have $n$ subjects, labelled 1 to $n$;
- Responses ($y$ variable) denoted $y_1, y_2, \ldots, y_n$;
- Explanatory (design) variables $x_1, \ldots, x_k$, with measured values for subject $i$ denoted $x_{i1}, x_{i2}, \ldots, x_{ik}$.

# Linear models

The linear model is

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_k x_{ik} + \varepsilon_i$$

for all $i = 1, 2, \ldots, n$, or

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1k} \\ 1 & x_{21} & x_{22} & \ldots & x_{2k} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \ldots & x_{nk} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

$$\mathbf{y} = X \boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

vector Matrix

# Linear models

Random vs Constant

Under the terminology we have developed, $\mathbf{y}$ and $\boldsymbol{\varepsilon}$ are random vectors. A common assumption is that $\boldsymbol{\varepsilon}$ is multivariate normal with mean $\mathbf{0}$ and variance $\sigma^2 I$.

$\underline{\varepsilon} \sim MVN(\underline{0}, \sigma^2 I)$

$X$ and $\boldsymbol{\beta}$ are NOT random vectors. Although it is common for $X$ to be a measurement, we assume that there is no uncertainty/error in these measurements.

# Pulling rank

$n > k+1$

We say the *model* has full rank when the design matrix $X$ has full rank, i.e. $r(X) = k+1$. This small condition is of crucial importance in the analysis of the model.

For this section, we assume that $X$ is of full rank.

This means that $X^T X$ is invertible, i.e. $(X^T X)^{-1}$ exists.

# The full rank model

**Example.** We want to analyse the selling price of a house ($y$). We think that this depends on two variables, its age ($x_1$) and the house area ($x_2$). Our linear model is

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \varepsilon_i.$$

We sample 5 random houses and obtain the data:

| Price ($\times$ \$10k) | Age (years) | Area ($\times 100m^2$) |
|---|---|---|
| 50 | 1 | 1 |
| 40 | 5 | 1 |
| 52 | 5 | 2 |
| 47 | 10 | 2 |
| 65 | 20 | 3 |

$k = 2$

$n = 5$

# The full rank model

The model generates the 5 linear equations

$$
\begin{aligned}
50 &= \beta_0 + 1\beta_1 + 1\beta_2 + \varepsilon_1 \\
40 &= \beta_0 + 5\beta_1 + 1\beta_2 + \varepsilon_2 \\
52 &= \beta_0 + 5\beta_1 + 2\beta_2 + \varepsilon_3 \\
47 &= \beta_0 + 10\beta_1 + 2\beta_2 + \varepsilon_4 \\
65 &= \beta_0 + 20\beta_1 + 3\beta_2 + \varepsilon_5
\end{aligned}
$$

# The full rank model

The matrix form of the model is $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, where

$$\mathbf{y} = \begin{bmatrix} 50 \\ 40 \\ 52 \\ 47 \\ 65 \end{bmatrix}, \quad X = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 5 & 1 \\ 1 & 5 & 2 \\ 1 & 10 & 2 \\ 1 & 20 & 3 \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix}.$$

R: $(X^TX)^{-1}$ solve(t(X) %*% X)

Direct calculation will show that $X$ is of full rank. This is an example of *multiple regression*.

R: rank

# The full rank model

**Example.** Simple linear regression can be cast in the framework of a linear model, where the response variable $y$ depends on only one variable $x$:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i.$$

For $n$ responses, this gives the linear equations

$$\begin{aligned} y_1 &= \beta_0 + \beta_1 x_1 + \varepsilon_1 \\ y_2 &= \beta_0 + \beta_1 x_2 + \varepsilon_2 \\ &\vdots \\ y_n &= \beta_0 + \beta_1 x_n + \varepsilon_n \end{aligned}$$

# The full rank model

In the matrix formulation, we have

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \\ 1 & x_n \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}.$$

$X$ has full rank, provided the $x_i$ are not all the same.

We will show later how the linear model framework can be used to derive the well-known regression formulas for the parameters $\beta_0$ and $\beta_1$.

# Parameter estimation using least squares

The first thing we do with the linear model is to estimate the parameters $\beta_0, \beta_1, \ldots, \beta_k$. We do this using the *method of least squares*.

Firstly, we assume that the error vector $\boldsymbol{\varepsilon}$ has mean $\mathbf{0}$ and variance $\sigma^2 I$; in other words, that the model is unbiased and that the errors are independent of the responses and uncorrelated with each other.

We do NOT necessarily assume that the errors are *independent* of each other. Nor do we assume they are normal (at first).

# Parameter estimation using least squares

Modell

$$\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

The error term is the only random term in the model, so

$$E[\mathbf{y}] = X\boldsymbol{\beta}$$

and

$$\text{Var } \mathbf{y} = \sigma^2 I.$$

The expected value of each response is a linear function of the design variables (hence the term "linear model").

# Parameter estimation using least squares

Suppose that $b_0, b_1, \ldots, b_k$ are estimates of the parameters $\beta_0, \beta_1, \ldots . \beta_k$.

Then we can estimate the expected value of $y_i$ by

$$\widehat{E[y_i]} = b_0 + b_1 x_{i1} + \ldots + b_k x_{ik}.$$

The $i$th residual is defined to be the difference between the observed value and the estimated value:

$$e_i = y_i - \widehat{E[y_i]}$$

$e_i \to 0$

~~$e_i \neq 0$~~

In practice $e_i \neq 0$

# Parameter estimation using least squares

If our estimates are good, the residuals should be very close to the errors:

$$\begin{aligned} \varepsilon_i &= y_i - E[y_i] = y_i - \widehat{E[y_i]} + \widehat{E[y_i]} - E[y_i] \\ &= e_i + \widehat{E[y_i]} - E[y_i]. \end{aligned}$$

We choose our estimates to minimise the residuals; specifically, we minimise the sum of the squares of the residuals. This is *least squares estimation* of the parameters.

(We could also try to minimise e.g. the sum of the absolute values of the residuals, but this is much harder because the absolute value is not differentiable.)

# Parameter estimation using least squares

Define the vectors of estimated parameters and residuals:

$$\mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix}, \quad \mathbf{e} = \begin{bmatrix} e_0 \\ e_1 \\ \vdots \\ e_n \end{bmatrix}.$$

Then we have

$$\mathbf{y} = X\mathbf{b} + \mathbf{e}$$

so

$$\sum_{i=1}^{n} e_i^2 = \mathbf{e}^T\mathbf{e} = (\mathbf{y} - X\mathbf{b})^T(\mathbf{y} - X\mathbf{b}).$$

$$= \sum_{i=1}^{n} e_i^2$$

# Parameter estimation using least squares

1. Least squares
2. Vector/Matrix

We choose $\mathbf{b}$ to minimise

$$\begin{aligned}\mathbf{e}^T\mathbf{e} &= (\mathbf{y}-X\mathbf{b})^T(\mathbf{y}-X\mathbf{b})\\ &= \mathbf{y}^T\mathbf{y}-\mathbf{y}^TX\mathbf{b}-\mathbf{b}^TX^T\mathbf{y}+\mathbf{b}^TX^TX\mathbf{b}\\ &= \mathbf{y}^T\mathbf{y}-2\mathbf{y}^TX\mathbf{b}+\mathbf{b}^TX^TX\mathbf{b}\\ &= \mathbf{y}^T\mathbf{y}-2(X^T\mathbf{y})^T\mathbf{b}+\mathbf{b}^T(X^TX)\mathbf{b}.\end{aligned}$$

That is, we need

$$\frac{\partial \mathbf{e}^T\mathbf{e}}{\partial \mathbf{b}} = \mathbf{0}.$$

$y^TXb = b^TX^Ty$

1. both scalar

$a = X^Ty$

$$\vec{a} = (a_1, \cdots, a_n)^T$$

$$\vec{b} = (b_1, \cdots, b_n)^T$$

$$\vec{a}^T \vec{b} = \vec{b}^T \vec{a}$$

$$\sum_{i=1}^{n} a_i b_i$$

# Parameter estimation using least squares

Since our measurements $\mathbf{y}$ do not depend on our parameter estimates $\mathbf{b}$, we get

constant

$$\frac{\partial}{\partial \mathbf{b}} \mathbf{y}^T \mathbf{y} = \mathbf{0}$$

$$\frac{\partial}{\partial \mathbf{b}} (-2(X^T \mathbf{y})^T \mathbf{b}) = -2X^T \mathbf{y}$$

Line 1

$$\frac{\partial}{\partial \mathbf{b}} (\mathbf{b}^T (X^T X) \mathbf{b}) = (X^T X)\mathbf{b} + (X^T X)^T \mathbf{b}.$$

Line 3

$$= 2(X^T X)\mathbf{b},$$

Thus we need

$$-2X^T \mathbf{y} + 2(X^T X)\mathbf{b} = \mathbf{0}.$$

Cp 02 – slide 69

# Parameter estimation using least squares

Rearranging gives the *normal equations*:

$$X^T X \mathbf{b} = X^T \mathbf{y}.$$

Because $X$ is of full rank, $X^T X$ has an inverse. Therefore we can solve for $\mathbf{b}$ to find the least squares estimator

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}.$$

# Parameter estimation using least squares

Least square estimator $\underline{b}$

minimizes $(\underline{y}-X\underline{\beta})^T(\underline{y}-X\underline{\beta})$

**Theorem 4.1**

*Let $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ where $X$ is a $n \times (k+1)$ matrix of full rank, $\boldsymbol{\beta}$ is a $(k+1) \times 1$ vector of parameters, and $\boldsymbol{\varepsilon}$ is a $n \times 1$ random vector with mean $\mathbf{0}$. Then the least squares estimator for $\boldsymbol{\beta}$ is*

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}.$$

1. $\underline{\varepsilon} \sim MVN(0, \sigma^2 I)$ ← p-value
2. ~~$b = \arg$~~ $(\underline{y}-X\beta)^T(\underline{y}-X\beta)$

$b$

# Parameter estimation using least squares

**Example.** We return to the house price example. Our data are the house prices (response) and the house age and area (design):

$$\mathbf{y} = \begin{bmatrix} 50 \\ 40 \\ 52 \\ 47 \\ 65 \end{bmatrix}, \quad X = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 5 & 1 \\ 1 & 5 & 2 \\ 1 & 10 & 2 \\ 1 & 20 & 3 \end{bmatrix}$$

# Parameter estimation using least squares

```
> y <- c(50,40,52,47,65)
> (X <- matrix(c(rep(1,5),1,5,5,10,20,1,1,2,2,3),5,3))
     [,1] [,2] [,3]
[1,]    1    1    1
[2,]    1    5    1
[3,]    1    5    2
[4,]    1   10    2
[5,]    1   20    3
> (b <- solve(t(X)%*%X,t(X)%*%y))
           [,1]
[1,] 33.0626151
[2,] -0.1896869
[3,] 10.7182320
```

# Parameter estimation using least squares

Therefore our fitted model is

$$y_i = 33.06 - 0.19x_{i1} + 10.72x_{i2} + \varepsilon_i.$$

Note that we often drop the index $i$ when writing down the model:

$$\begin{aligned} y &= 33.06 - 0.19x_1 + 10.72x_2 + \varepsilon \\ \text{price} &= 33.06 - 0.19\,\text{age} + 10.72\,\text{area} + \varepsilon \end{aligned}$$

# Simple linear regression

**Example.** Recall that the simple linear regression model can be written as a linear model with two parameters

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i,$$

which gives

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}.$$

# Simple linear regression

Then

$$X^T X = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}$$

$$= \begin{bmatrix} n & \sum_i x_i \\ \sum_i x_i & \sum_i x_i^2 \end{bmatrix}$$

and

$$X^T \mathbf{y} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$= \begin{bmatrix} \sum_i y_i \\ \sum_i x_i y_i \end{bmatrix}.$$

# Simple linear regression

We have

$$(X^T X)^{-1} = \frac{1}{n\sum_i x_i^2 - \left(\sum_i x_i\right)^2} \begin{bmatrix} \sum_i x_i^2 & -\sum_i x_i \\ -\sum_i x_i & n \end{bmatrix}.$$

Therefore the least squares estimator for $\boldsymbol{\beta}$ is

$$\begin{aligned} \mathbf{b} &= (X^T X)^{-1} X^T \mathbf{y} \\ &= \frac{1}{n\sum_i x_i^2 - \left(\sum_i x_i\right)^2} \begin{bmatrix} \sum_i x_i^2 & -\sum_i x_i \\ -\sum_i x_i & n \end{bmatrix} \begin{bmatrix} \sum_i y_i \\ \sum_i x_i y_i \end{bmatrix} \\ &= \frac{1}{n\sum_i x_i^2 - \left(\sum_i x_i\right)^2} \begin{bmatrix} \sum_i x_i^2 \sum_i y_i - \sum_i x_i \sum_i x_i y_i \\ n\sum_i x_i y_i - \sum_i x_i \sum_i y_i \end{bmatrix}. \end{aligned}$$

# Simple linear regression

The estimator for the slope of the regression line is

$$\begin{aligned} b_1 &= \frac{n\sum_i x_i y_i - \sum_i x_i \sum_i y_i}{n\sum_i x_i^2 - \left(\sum_i x_i\right)^2} \\ &= \frac{\frac{1}{n}\sum_i x_i y_i - \frac{1}{n^2}\sum_i x_i \sum_i y_i}{\frac{1}{n}\sum_i x_i^2 - \frac{1}{n^2}\left(\sum_i x_i\right)^2} \\ &= \frac{\overline{xy} - \bar{x}\,\bar{y}}{\overline{x^2} - \bar{x}^2} \\ &= \frac{\mathsf{Cov}(x, y)}{\mathsf{Var}\ x}, \end{aligned}$$

which may be familiar from simple linear regression.

# Simple linear regression

The estimator for the intercept of the regression line is

$$b_0 = \frac{\sum_i x_i^2 \sum_i y_i - \sum_i x_i \sum_i x_i y_i}{n \sum_i x_i^2 - \left(\sum_i x_i\right)^2}$$

which may not look so familiar.

Usually the estimator is given as $\bar{y} - b_1 \bar{x}$, but it is quite simple to show that they are the same.

# How good is the least squares estimator?

What makes an estimator "good"?

Two desirable properties for an estimator are unbiased (on target) and of small variance.

## Theorem 4.2

*In the general linear model, the least squares estimator* $\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}$ *is an unbiased estimator for* $\boldsymbol{\beta}$*. In other words,*

$$E[\mathbf{b}] = \boldsymbol{\beta}.$$

*Furthermore,*

$$Var\ \mathbf{b} = (X^T X)^{-1} \sigma^2.$$

# How good is the least squares estimator?

**Proof.** Here is where some random vector theory comes in handy!

$$\begin{aligned} E[\mathbf{b}] &= E[(X^T X)^{-1} X^T \mathbf{y}] \\ &= (X^T X)^{-1} X^T E[\mathbf{y}] \\ &= (X^T X)^{-1} X^T (X\boldsymbol{\beta}) \\ &= \boldsymbol{\beta}. \end{aligned}$$

$$\begin{aligned} \text{Var } \mathbf{b} &= \text{Var } (X^T X)^{-1} X^T \mathbf{y} \\ &= (X^T X)^{-1} X^T \sigma^2 I \left((X^T X)^{-1} X^T\right)^T \\ &= (X^T X)^{-1} X^T X((X^T X)^T)^{-1} \sigma^2 \\ &= (X^T X)^{-1} \sigma^2. \end{aligned}$$

# But really, how good is the least squares estimator?

Let's look at *linear* estimators. These are estimators which take the form $L\mathbf{y}$, where $L$ is a matrix of constants. The least squares estimator is a linear estimator with $L = (X^T X)^{-1} X^T$.

Now suppose we have a model with some parameters $\boldsymbol{\beta}$ and linear estimators $\mathbf{b}$ for these parameters.

### Definition 4.3

If $E[\mathbf{b}] = \boldsymbol{\beta}$ and the variances of $b_0, b_1, \ldots, b_k$ are minimised over all linear estimators, then $\mathbf{b}$ is called a *best linear unbiased estimator* of $\boldsymbol{\beta}$ (or BLUE).

# I'm feeling BLUE

## Theorem 4.4 (Gauss-Markov Theorem)

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$*, the least squares estimator* $\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}$ *is the unique BLUE for* $\boldsymbol{\beta}$*.*

**Proof.**

Suppose we have another unbiased linear estimator for $\boldsymbol{\beta}$, called $\mathbf{b}^*$. We can write this as

$$\mathbf{b}^* = [(X^T X)^{-1} X^T + B]\mathbf{y}$$

where $B$ is a $(k+1) \times n$ matrix.

# I'm feeling BLUE

## Theorem 4.4 (Gauss-Markov Theorem)

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$*, the least squares estimator* $\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}$ *is the unique BLUE for* $\boldsymbol{\beta}$*.*

**Proof.**
We then take expectations of both sides:

$$\begin{aligned} E[\mathbf{b}^*] &= [(X^T X)^{-1} X^T + B] E[\mathbf{y}] \\ &= [(X^T X)^{-1} X^T + B] X\boldsymbol{\beta} \\ &= [I + BX]\boldsymbol{\beta}. \end{aligned}$$

Since $\mathbf{b}^*$ is an unbiased estimator for $\boldsymbol{\beta}$, we know that $E[\mathbf{b}^*] = \boldsymbol{\beta}$. Therefore $[I + BX]\boldsymbol{\beta} = \boldsymbol{\beta}$, which means that $BX = 0$.

# I'm feeling BLUE

## Theorem 4.4 (Gauss-Markov Theorem)

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$*, the least squares estimator* $\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}$ *is the unique BLUE for* $\boldsymbol{\beta}$*.*

**Proof.**
Now look at the variance of $\mathbf{b}^*$:

$$\begin{aligned}
\text{Var } \mathbf{b}^* &= \text{Var } [(X^T X)^{-1} X^T + B]\mathbf{y} \\
&= [(X^T X)^{-1} X^T + B]\sigma^2 I[(X^T X)^{-1} X^T + B]^T \\
&= \sigma^2[(X^T X)^{-1} X^T + B][X(X^T X)^{-1} + B^T] \\
&= \sigma^2[(X^T X)^{-1} X^T X(X^T X)^{-1} + (X^T X)^{-1} X^T B^T \\
&\qquad + BX(X^T X)^{-1} + BB^T].
\end{aligned}$$

# I'm feeling BLUE

## Theorem 4.4 (Gauss-Markov Theorem)

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$*, the least squares estimator* $\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}$ *is the unique BLUE for* $\boldsymbol{\beta}$*.*

**Proof.**

Now $BX = 0$ and $X^T B^T = (BX)^T = 0$, which gives

$$\begin{aligned} \text{Var } \mathbf{b}^* &= \sigma^2[(X^T X)^{-1} + BB^T] \\ &= (X^T X)^{-1}\sigma^2 + BB^T\sigma^2 \\ &= \text{Var } \mathbf{b} + BB^T\sigma^2. \end{aligned}$$

Let's look at the variances of $b_0^*, b_1^*, \ldots, b_k^*$ (ignoring the covariances for now.)

# I'm feeling BLUE

## Theorem 4.4 (Gauss-Markov Theorem)

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$*, the least squares estimator* $\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}$ *is the unique BLUE for* $\boldsymbol{\beta}$*.*

**Proof.**
The variances are given by

$$\text{Var } b_i^* = [\text{Var } \mathbf{b}^*]_{ii} = \text{Var } b_i + \sigma^2 \sum_{j=1}^{n} B_{ij}^2.$$

Each term in the sum is non-negative, so the variance of $b_i^*$ can never go below Var $b_i$.

Moreover, the minimum is obtained if and only if $B_{ij} = 0$ for all $i, j$, in which case $B = 0$ and $\mathbf{b}^* = \mathbf{b}$.

# Gauss-Markov Theorem

**Example.** Consider the house price example. The variance of the least squares estimators is given by

$$(X^T X)^{-1}\sigma^2 = \begin{bmatrix} 2.31 & 0.16 & -1.88 \\ 0.16 & 0.03 & -0.2 \\ -1.88 & -0.2 & 1.98 \end{bmatrix} \sigma^2.$$

This means that there is no (unbiased) linear estimator of $\beta_0$ which has a smaller variance than $2.31\sigma^2$, and no linear estimator of $\beta_1$ which has a smaller variance than $0.03\sigma^2$, etc.

This is true even though we don't know what $\sigma^2$ is!

# Estimation of linear functions

What if we want to estimate something other than the parameters?

We are often interested in estimating some linear function of the parameters, $\mathbf{t}^T\boldsymbol{\beta}$, where $\mathbf{t}$ is a $(k+1) \times 1$ vector of constants. How can we estimate these?

It turns out that obvious answer is correct: we simply take the identical linear function of the least squares estimator.

# Estimation of linear functions

**Theorem 4.5**

*Take the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ *and let* $\mathbf{t}$ *be a* $(k+1) \times 1$ *vector of constants. Then the best linear unbiased estimator for* $\mathbf{t}^T\boldsymbol{\beta}$ *is* $\mathbf{t}^T\mathbf{b}$*, where* $\mathbf{b}$ *is the least squares estimator for* $\boldsymbol{\beta}$*.*

The proof of this theorem is very similar to that of the Gauss-Markov theorem.

# Estimation of linear functions

The most common use of this theorem is to predict the value of the response variable given certain values of the predictor variables.

**Example.** Consider the house price example. The model is

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon,$$

where $y$ is the house price, $x_1$ is its age, and $x_2$ is its area.

Suppose we are given a specific house with age $x_1^*$ and area $x_2^*$, and we wish to estimate what price it will fetch.

# Estimation of linear functions

We want to estimate the linear function of the parameters

$$E[y] = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* = \mathbf{t}^T \boldsymbol{\beta}$$

where $\mathbf{t} = \begin{bmatrix} 1 & x_1^* & x_2^* \end{bmatrix}^T$.

Therefore an unbiased estimator for the house price is

$$\mathbf{t}^T \mathbf{b} = \begin{bmatrix} 1 & x_1^* & x_2^* \end{bmatrix} \mathbf{b} = b_0 + b_1 x_1^* + b_2 x_2^*$$

where $\mathbf{b}$ is the least squares estimator for $\boldsymbol{\beta}$.

# Estimation of linear functions

For example, suppose we have a house which is 15 years old and has an area of 250 $m^2$.

```
> b
```

```
             [,1]
[1,] 33.0626151
[2,] -0.1896869
[3,] 10.7182320
```

```
> c(1,15,2.5)%*%b
```

```
         [,1]
[1,] 57.01289
```

We expect the house to sell for $570,129.

# Variance estimation

Remember that we assume that the errors $\boldsymbol{\varepsilon}$ (and thus $\mathbf{y}$) have covariance matrix $\sigma^2 I$. We will also want to estimate the common variance $\sigma^2$.

One reason to do this is to create confidence intervals for the true values of the parameters.

# Variance estimation

How should we estimate $\sigma^2$?

$\sigma^2$ can be written as

$$\sigma^2 = E\left[\frac{(\mathbf{y} - X\boldsymbol{\beta})^T(\mathbf{y} - X\boldsymbol{\beta})}{n}\right]$$

and so a reasonable estimator for the variance might be

$$\hat{\sigma^2} = \frac{(\mathbf{y} - X\mathbf{b})^T(\mathbf{y} - X\mathbf{b})}{n}.$$

It turns out that this is slightly biased; we need to make a small adjustment.

# Variance estimation

## Theorem 4.6

*The sample variance*

$$s^2 = \frac{(\mathbf{y} - X\mathbf{b})^T(\mathbf{y} - X\mathbf{b})}{n - (k+1)}$$

*is an unbiased estimator for* $\sigma^2$.

Define the sum of squares of the residuals

$$SS_{Res} = (\mathbf{y} - X\mathbf{b})^T(\mathbf{y} - X\mathbf{b})$$

and denote the number of parameters by $p = k + 1$. Then we can write

$$s^2 = \frac{SS_{Res}}{n - p}.$$

# Variance estimation

**Proof.**

$$\begin{aligned} E[s^2] &= \frac{1}{n-(k+1)} E[(\mathbf{y}-X\mathbf{b})^T(\mathbf{y}-X\mathbf{b})] \\ &= \frac{1}{n-(k+1)} E[(\mathbf{y}-X(X^TX)^{-1}X^T\mathbf{y})^T(\mathbf{y}-X(X^TX)^{-1}X^T\mathbf{y})] \\ &= \frac{1}{n-(k+1)} E[\mathbf{y}^T(I-X(X^TX)^{-1}X^T)(I-X(X^TX)^{-1}X^T)\mathbf{y}]. \end{aligned}$$

It is a simple exercise to show that $I - X(X^TX)^{-1}X^T$ is idempotent, which gives

$$E[s^2] = \frac{1}{n-(k+1)} E[\mathbf{y}^T(I-X(X^TX)^{-1}X^T)\mathbf{y}].$$

# Variance estimation

The expectation of this quadratic form is given by Theorem 3.2:

$$E[\mathbf{y}^T A\mathbf{y}] = tr(AV) + \boldsymbol{\mu}^T A\boldsymbol{\mu}.$$

Here

$$\begin{aligned}
\boldsymbol{\mu}^T A\boldsymbol{\mu} &= (X\boldsymbol{\beta})^T(I - X(X^TX)^{-1}X^T)X\boldsymbol{\beta} \\
&= \boldsymbol{\beta}^T X^T X\boldsymbol{\beta} - \boldsymbol{\beta}^T X^T X(X^TX)^{-1}X^T X\boldsymbol{\beta} \\
&= 0
\end{aligned}$$

# Variance estimation

and

$$\begin{aligned} tr(AV) &= tr((I_n - X(X^TX)^{-1}X^T)\sigma^2 I_n) \\ &= \sigma^2(tr(I_n) - tr(X(X^TX)^{-1}X^T)) \\ &= \sigma^2(n - tr((X^TX)^{-1}X^TX)) \\ &= \sigma^2(n - tr(I_{k+1})) \\ &= \sigma^2(n - (k+1)) \end{aligned}$$

which gives the result.

# Variance estimation

**Example.** Back to the house price example.

```
> b
```

```
             [,1]
[1,] 33.0626151
[2,] -0.1896869
[3,] 10.7182320
```

```
> (e <- y - X%*%b)
```

```
           [,1]
[1,]  6.408840
[2,] -2.832413
[3,] -1.550645
[4,] -5.602210
[5,]  3.576427
```

# Variance estimation

```
> (SSRes <- sum(e^2))
[1] 95.67587
> (s2 <- SSRes/(5-3))
[1] 47.83794
```

The sample variance is $s^2 = 47.84$.

```
> diag(solve(t(X)%*%X))*s2
[1] 110.388463   1.233391  94.618683
```

This gives the estimated variances of the parameter estimators.

# Variance estimation

**Example.** A study is designed to predict the extent of the cracking of latex paint in field conditions, based on the extent of the cracking in 'accelerated' tests in the laboratory. We generate the data:

| Test cracking ($x$) | Actual cracking ($y$) |
|---|---|
| 2.0 | 1.9 |
| 3.0 | 2.7 |
| 4.0 | 4.2 |
| 5.0 | 4.8 |
| 6.0 | 4.8 |
| 7.0 | 5.1 |

# Variance estimation

```
> y <- c(1.9,2.7,4.2,4.8,4.8,5.1)
> (X <- matrix(c(rep(1,6),2:7),6,2))
     [,1] [,2]
[1,]    1    2
[2,]    1    3
[3,]    1    4
[4,]    1    5
[5,]    1    6
[6,]    1    7
> (b <- solve(t(X)%*%X,t(X)%*%y))
          [,1]
[1,] 0.9723810
[2,] 0.6542857
```

# Variance estimation

```
> (e <- y - X%*%b)
             [,1]
[1,] -0.38095238
[2,] -0.23523810
[3,]  0.61047619
[4,]  0.55619048
[5,] -0.09809524
[6,] -0.45238095
> (s2 <- sum(e^2)/(6-2))
[1] 0.2741905
```

Thus we estimate the common variance of the response variables to be $\approx 0.27$.

# Regression through the origin

So far we have always considered the linear model to include a parameter $\beta_0$, which is associated with a column of 1's in the design matrix $X$. This parameter is called the *intercept*.

Sometimes it is reasonable to assume (from prior knowledge of the data) that no intercept is needed, in which case we can remove it.

Surprisingly little changes. The model becomes

$$y = \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k + \varepsilon,$$

but to analyse it, the design matrix loses the first column, the parameter vector loses the first entry, and everything proceeds as before.

# Regression through the origin

The least squares estimator is still

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}$$

and the variance estimator is still

$$s^2 = \frac{SS_{Res}}{n - p}.$$

However, we now have $p = k$ instead of $k + 1$.

# Diagnostics: standardised residuals

To assess the fit of our linear models, and to observe possible departures from our model assumptions, we use various diagnostic tools.

Firstly we can look at our residuals. If there is an extremely large residual, or a pattern in the residuals, we might question our assumptions.

However we must be careful. The variance of the errors is (assumed to be) $\sigma^2 I$, but the variance of the *residuals* is not $\sigma^2 I$. In general, the variance of a particular residual depends on how 'far from the centre' the design variables are: the farther away, the lower the variance of the residual.

# Diagnostics: standardised residuals

To make this precise, we calculate the variance of the residuals

$$\mathbf{e} = \mathbf{y} - X\mathbf{b} = (I - X(X^T X)^{-1} X^T)\mathbf{y}.$$

We define $H = X(X^T X)^{-1} X^T$. This matrix is called the *hat* matrix, because it converts $\mathbf{y}$ (the observed responses) into $\hat{\mathbf{y}}$ (the estimated responses).

$$\begin{aligned} \text{Var } \mathbf{e} &= \text{Var } (I - H)\mathbf{y} \\ &= (I - H)\sigma^2 I (I - H)^T \\ &= \sigma^2 (I - H) \end{aligned}$$

since $I - H$ is symmetric and idempotent.

# Diagnostics: standardised residuals

To make residuals comparable, we would like to divide them by their standard deviations. However we don't know $\sigma^2$, and so use $s^2$ instead.

This creates the *standardised residuals*

$$z_i = \frac{e_i}{\sqrt{s^2(1 - H_{ii})}}.$$

The standardised residuals have (approximately) unit variance and thus can be compared reasonably.

# Diagnostics: leverage and Cook's distance

Consider what happens when we calculate the fitted values by $\hat{\mathbf{y}} = X\mathbf{b} = H\mathbf{y}$.

$H$ tends to have its largest values on the diagonal (the best estimate for the mean of $y_i$ is generally close to $y_i$). The size of $H_{ii}$ reflects how much $\hat{y}_i$ is based on $y_i$, as opposed to the other $y_j$. If $H_{ii}$ is particularly large, then $y_i$ has a large effect on the fit.

We thus define the *leverage* of point $i$ as $H_{ii}$.

# Diagnostics: leverage and Cook's distance

Points with large leverage have an unusually large effect on the estimated parameters. We must be extra careful with these points to avoid a bad fit.

By itself, a large leverage is not necessarily detrimental. However, if this is combined with a large residual, then the corresponding point may distort the fit.

To check this, we calculate the *Cook's distance* of each point. This measures the change in the estimated parameters $\mathbf{b}$ if we remove the point.

# Diagnostics: leverage and Cook's distance

The definition of Cook's distance is

$$D_i = \frac{(\mathbf{b}_{(-i)} - \mathbf{b})^T X^T X (\mathbf{b}_{(-i)} - \mathbf{b})}{(k+1)s^2} = \frac{1}{k+1} z_i^2 \left( \frac{H_{ii}}{1 - H_{ii}} \right)$$

where $\mathbf{b}_{(-i)}$ is the estimated parameters if point $i$ is removed.

We can see that this is large if both the standardised residual and the leverage is large — this is where we must be careful.

There is no particular 'must watch' value for Cook's distance, but it is generally considered large if it is greater than 1, and small if it is less than 0.5.

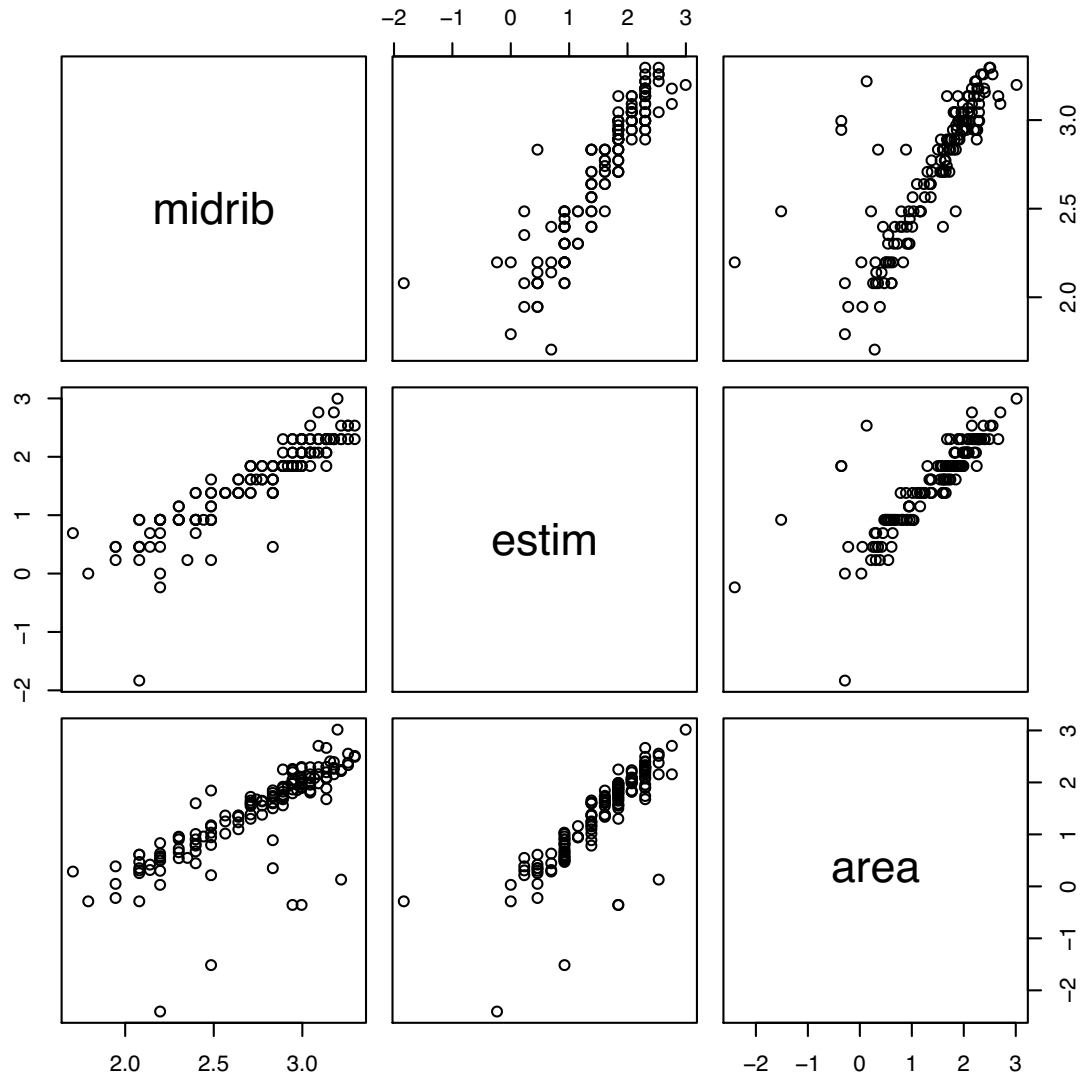
# R example: clover leaves

We estimate the area of a clover leaf (`area`) based on the midrib length (`midrib`) and estimated area by template (`estim`).

It turns out that (based on knowledge of the data) it is more appropriate to take the logarithms of the data.

```
> clover <- read.csv("../data/clover.csv")
> str(clover)

'data.frame':        145 obs. of  3 variables:
 $ midrib: num  5.5 6 7 7 7 8 8 8 8 8 ...
 $ estim : num  2 1 1.58 1.58 1.26 0.16 1.58 1.26 1.58 2.51 ...
 $ area  : num  1.33 0.75 0.8 1.05 1.47 0.75 1.29 1.36 1.42 1.6

> clover <- log(clover)
> pairs(clover)
```

# Clover leaves: data plot



# R example: clover leaves

Our model is

$$\text{area} = \beta_0 + \beta_1 \text{midrib} + \beta_2 \text{estim} + \varepsilon.$$

```
> y <- clover$area
> str(y)
```

```
 num [1:145] 0.2852 -0.2877 -0.2231 0.0488 0.3853 ...
```

```
> X <- cbind(1,clover$midrib,clover$estim)
> X[1:3,]
```

```
     [,1]     [,2]      [,3]
[1,]    1 1.704748 0.6931472
[2,]    1 1.791759 0.0000000
[3,]    1 1.945910 0.4574248
```

# R example: clover leaves

```
> library(Matrix)
> (n <- dim(X)[1])
[1] 145
> (p <- dim(X)[2])
[1] 3
> rankMatrix(X)[1]
[1] 3
```

so this is a full rank model.

# R example: clover leaves

```
> (b <- solve(t(X) %*% X, t(X) %*% y))
            [,1]
[1,] -1.1741275
[2,]  0.5239692
[3,]  0.7337812
> e <- y - X %*% b
> str(e)
 num [1:145, 1] 0.0575 -0.0524 -0.4043 -0.1323 0.3702 ...
```

# The R way

```
> model <- lm(area ~ midrib + estim,data=clover)
> summary(model)

Call:
lm(formula = area ~ midrib + estim, data = clover)

Residuals:
     Min       1Q   Median       3Q      Max
-2.31730 -0.07022  0.08005  0.18787  1.14160

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -1.1741     0.4604   -2.55   0.0118 *
midrib        0.5240     0.2248    2.33   0.0212 *
estim         0.7338     0.1157    6.34 2.87e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4659 on 142 degrees of freedom
Multiple R-squared:  0.7078,        Adjusted R-squared:  0.7036
F-statistic:   172 on 2 and 142 DF,  p-value: < 2.2e-16
```

# The R way

```
> model$coefficients

(Intercept)      midrib       estim
 -1.1741275   0.5239692   0.7337812

> str(model$residuals)

 Named num [1:145] 0.0575 -0.0524 -0.4043 -0.1323 0.3702 ...
 - attr(*, "names")= chr [1:145] "1" "2" "3" "4" ...

> str(model$fitted.values)

 Named num [1:145] 0.2277 -0.2353 0.1811 0.1811 0.0151 ...
 - attr(*, "names")= chr [1:145] "1" "2" "3" "4" ...

> model$rank

[1] 3

> model$df.residual

[1] 142
```

# Point estimation

Point estimate of the area of a leaf with midrib 10 and template area 10:

```
> tt <- c(1,log(10),log(10))
> tt %*% b

        [,1]
[1,] 1.72195
```

```
> newclover <- data.frame(midrib=log(10),estim=log(10))
> predict(model,newclover)

      1
1.72195
```

# Variance estimation

```
> (SSRes <- sum(e^2))
[1] 30.82559
> (s2 <- SSRes/(n-p))
[1] 0.2170816
> deviance(model)
[1] 30.82559
> deviance(model)/model$df.residual
[1] 0.2170816
```

# Diagnostic plots

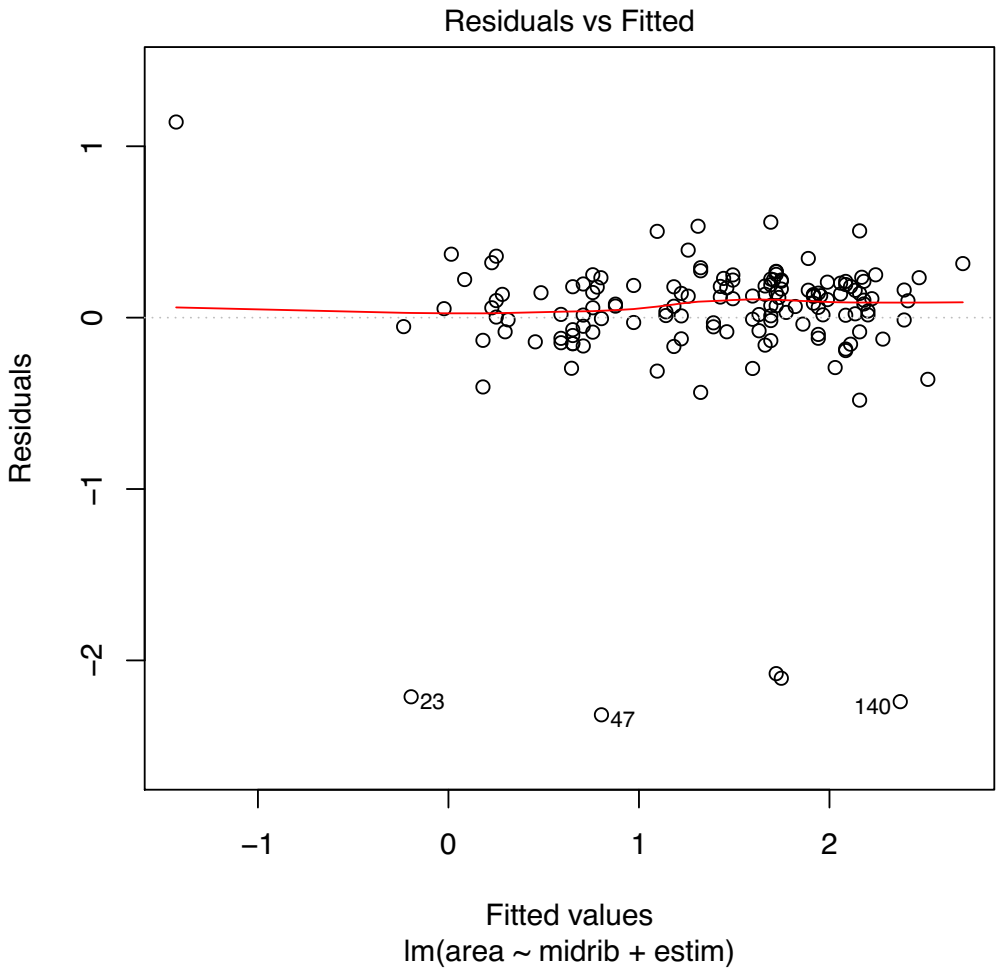
R (and in particular the `lm` command) produces many useful plots for checking the fit of the model and deviations from assumptions.

The first plot is residuals vs. fitted values. We look for:

- points with large residual (unequal variances);
- a trend in the residuals (bias);
- a pattern in the residuals (correlation).

# Diagnostic plots

```
> plot(model, which=1)
```



# Diagnostic plots

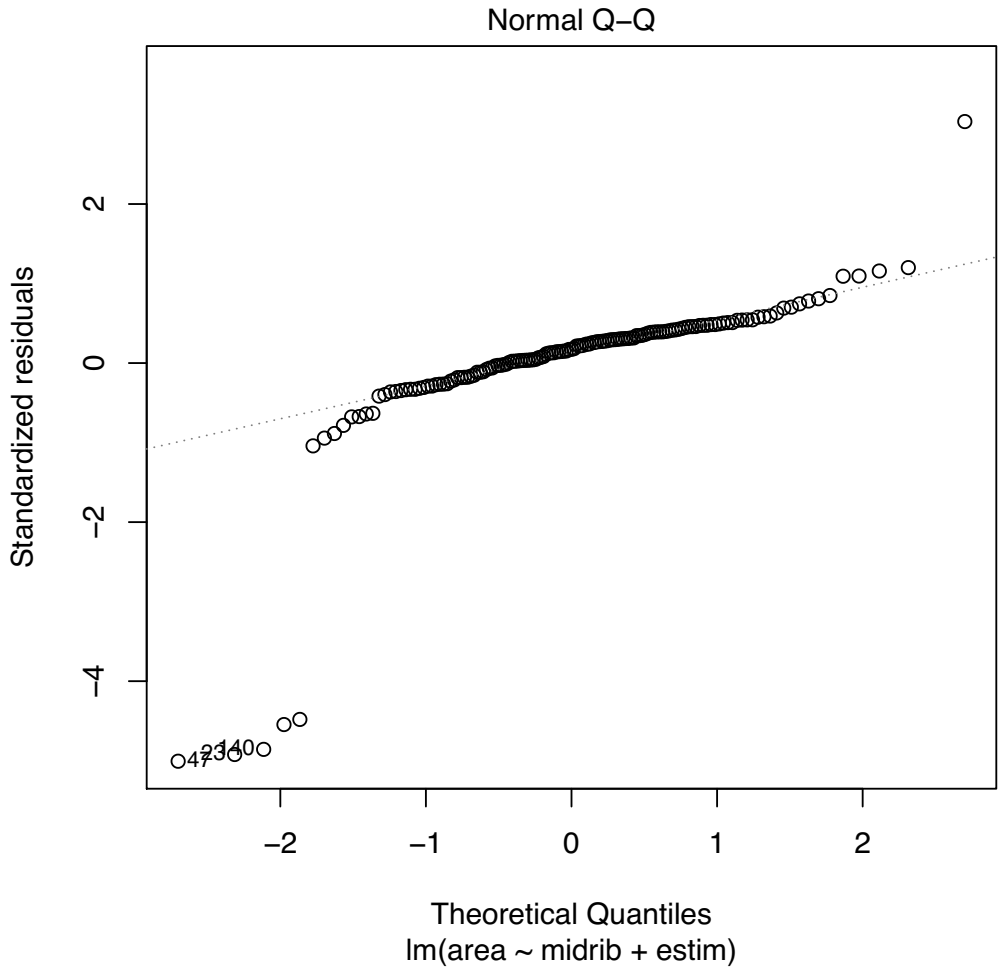
The second plot is a normal quantile-quantile plot of the standardised residuals.

We look for the points to follow the line (i.e. be normally distributed). If not, then we look for how they deviate — for example:

- a small number of outliers;
- over- or under-estimation in the tails;
- skewness.

# Diagnostic plots

```
> plot(model, which=2)
```



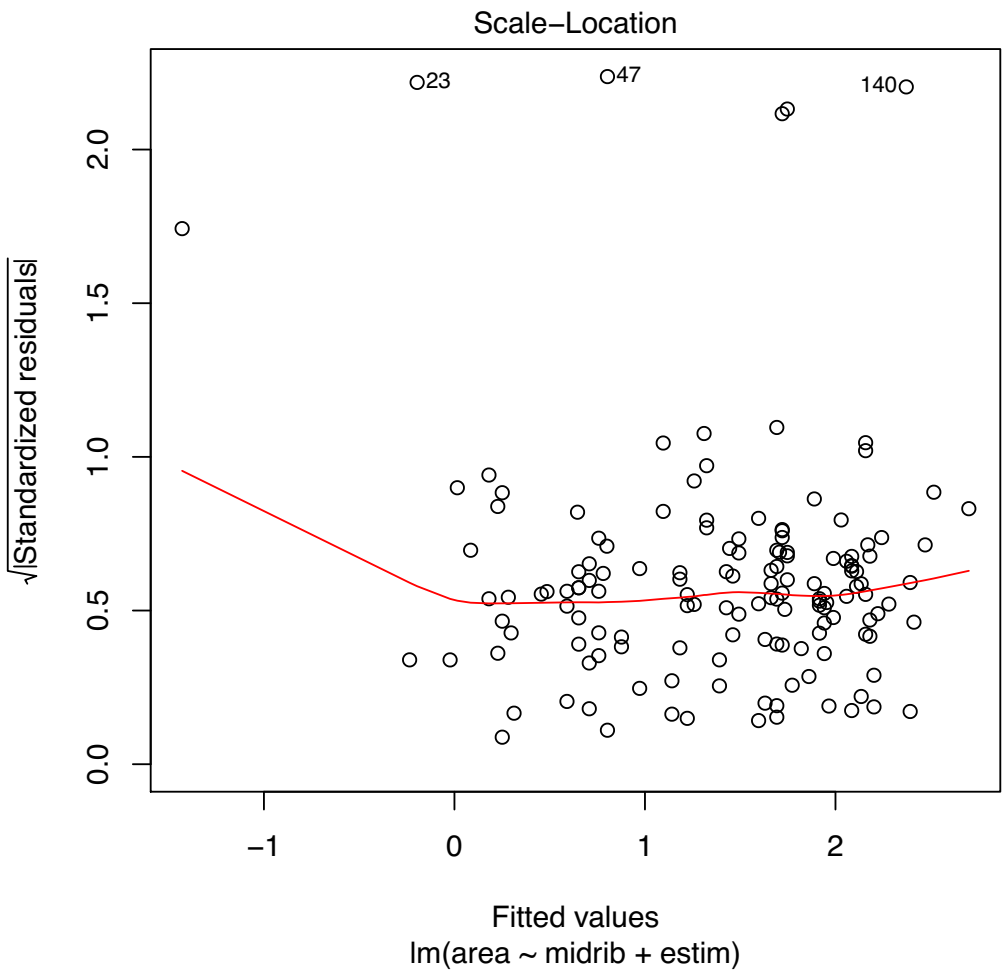
# Diagnostic plots

The third plot is square roots of absolute values of standardised residuals against fitted values. It is quite similar to the first plot. We look for:

- points with high residual (unequal variance);
- a trend in the size of the residuals (heteroskedasticity).

# Diagnostic plots

```
> plot(model, which=3)
```



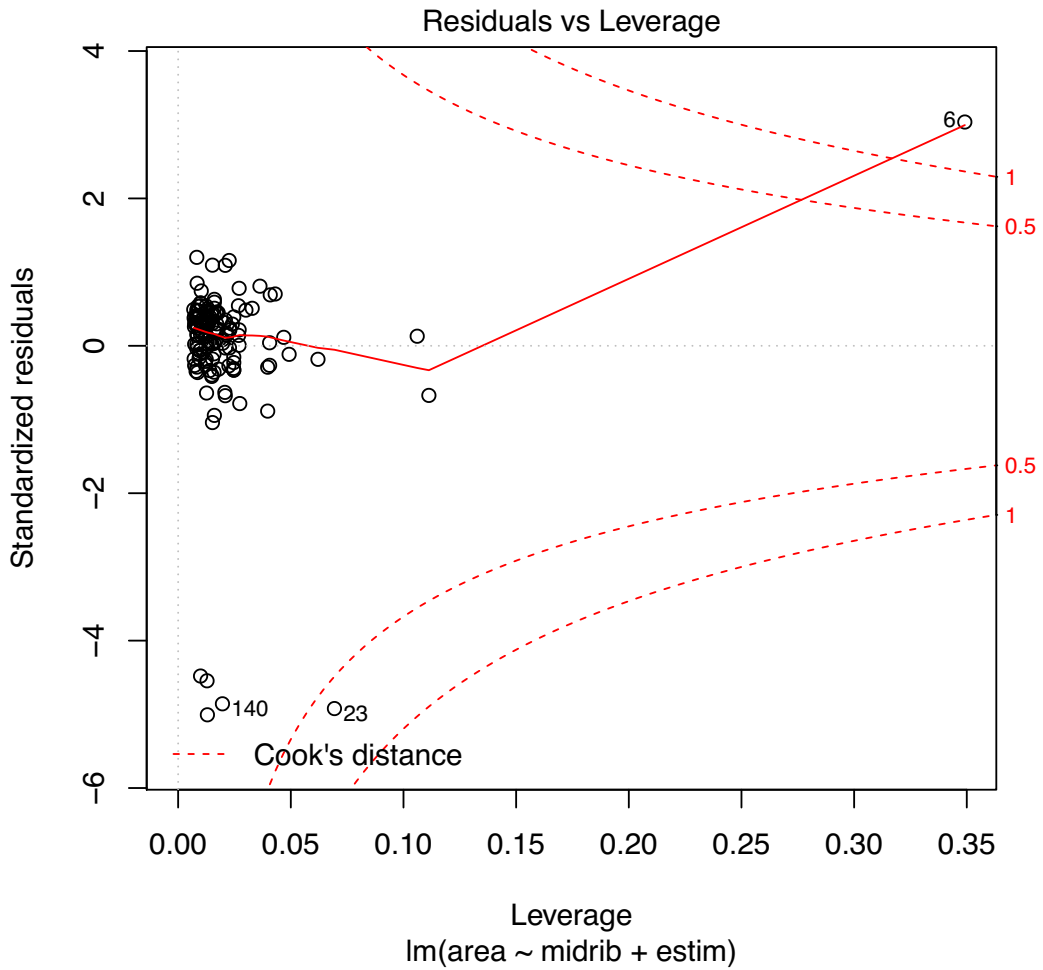
# Diagnostic plots

The fourth plot is leverage vs. standardised residuals. We look for:

- points with high residual (unequal variance);
- points with high leverage (potentially dangerous);
- points with high Cook's distance (poor fit);
- a pattern in the residuals (correlation).

# Diagnostic plots

```
> plot(model, which=5)
```

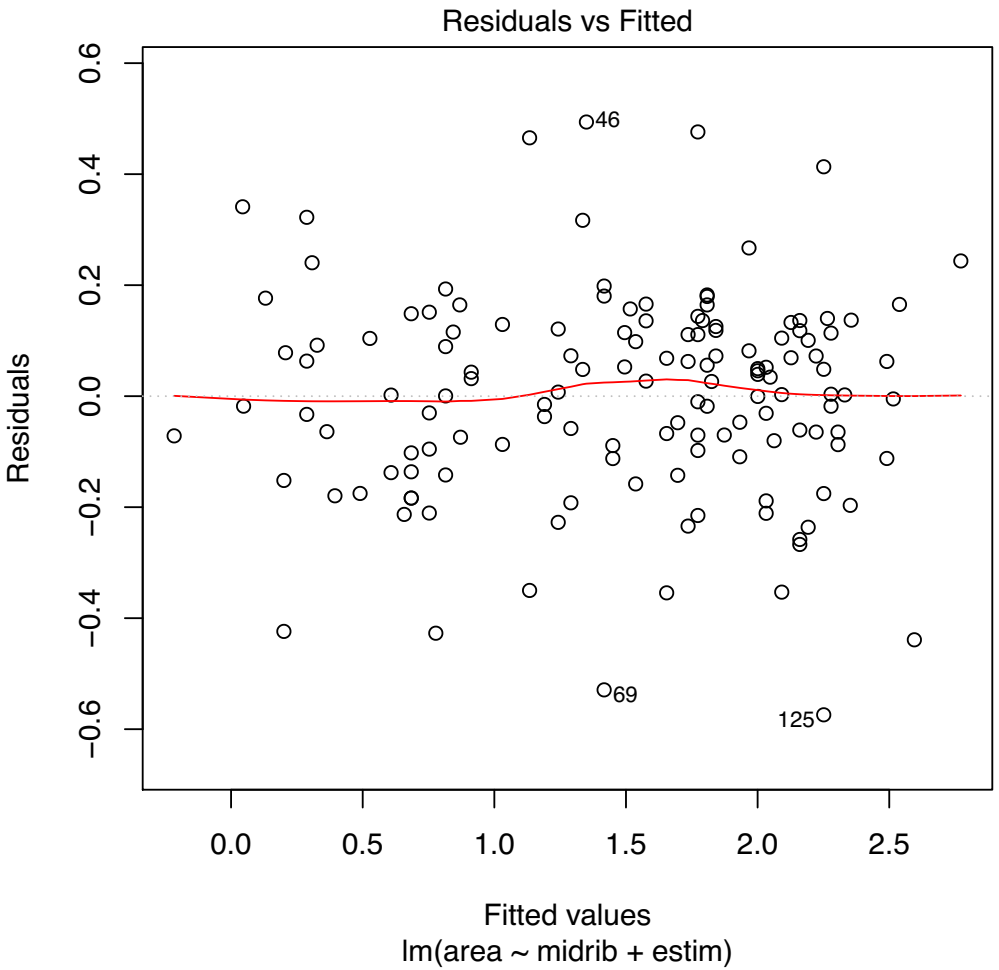


# What if we remove the offending points?

```
> goodclover <- clover[-c(6,23,47,97,111,140),]
> model2 <- lm(area ~ midrib + estim, data=goodclover)
> summary(model2)

Call:
lm(formula = area ~ midrib + estim, data = goodclover)

Residuals:
     Min       1Q   Median       3Q      Max
-0.57403 -0.10000  0.00737  0.11681  0.49398

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.38148    0.20516  -6.734 4.26e-10 ***
midrib       0.65037    0.10567   6.154 7.92e-09 ***
estim        0.69199    0.05958  11.615  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1863 on 136 degrees of freedom
Multiple R-squared:  0.9331,        Adjusted R-squared:  0.9321
F-statistic: 948.7 on 2 and 136 DF,  p-value: < 2.2e-16
```

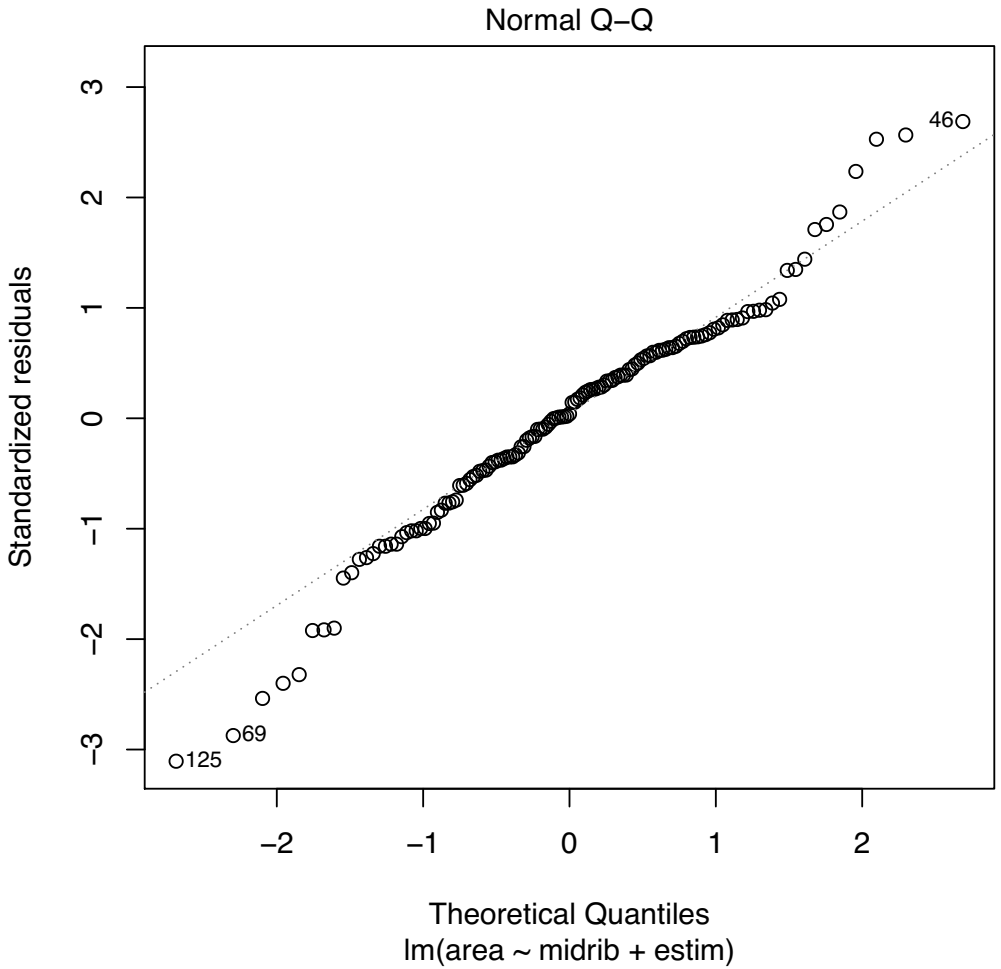
# What if we remove the offending points?

```
> plot(model2, which=1)
```



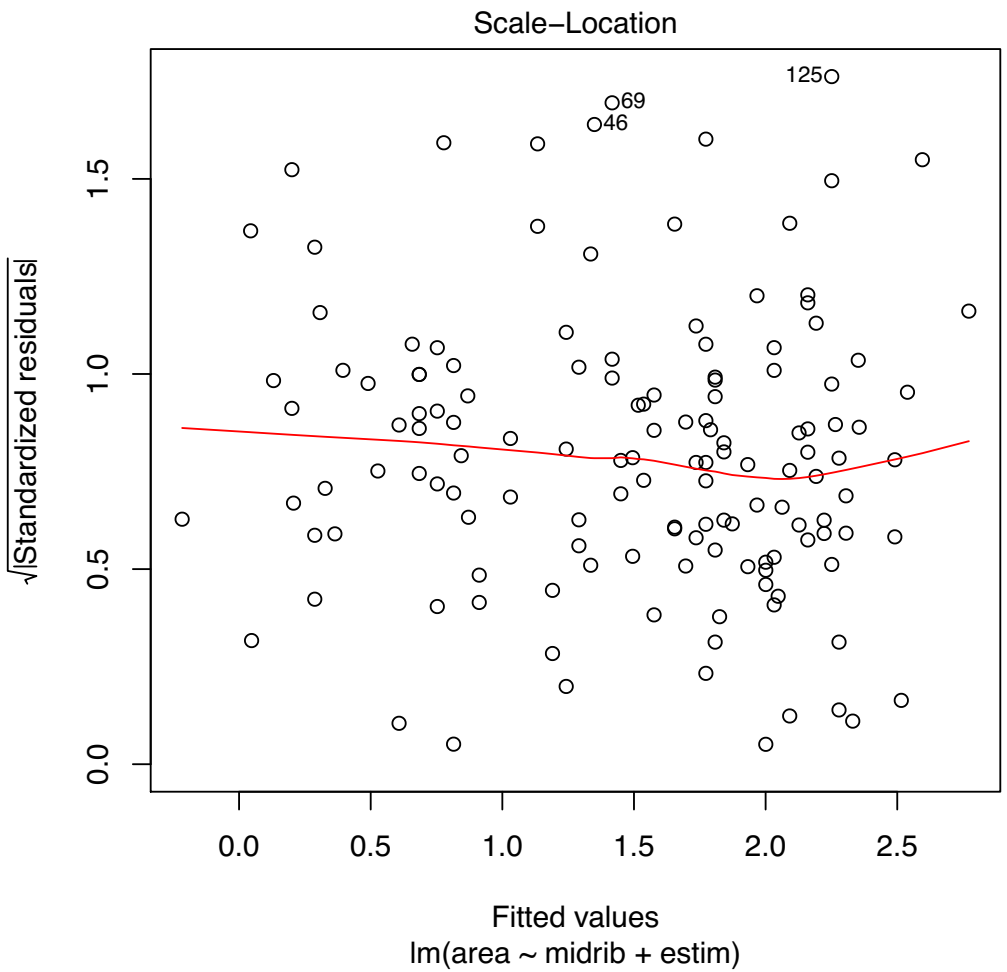
# What if we remove the offending points?

```
> plot(model2, which=2)
```



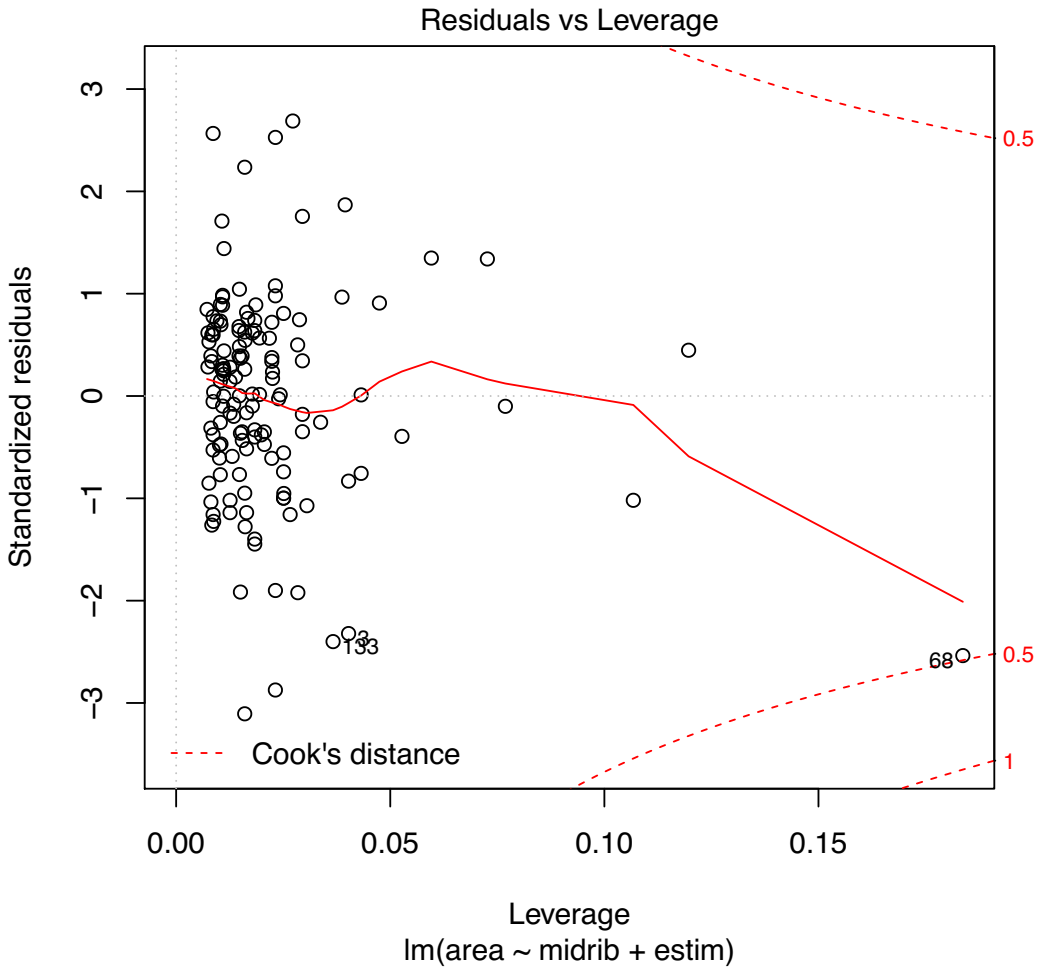
# What if we remove the offending points?

```
> plot(model2, which=3)
```



# What if we remove the offending points?

```
> plot(model2, which=5)
```

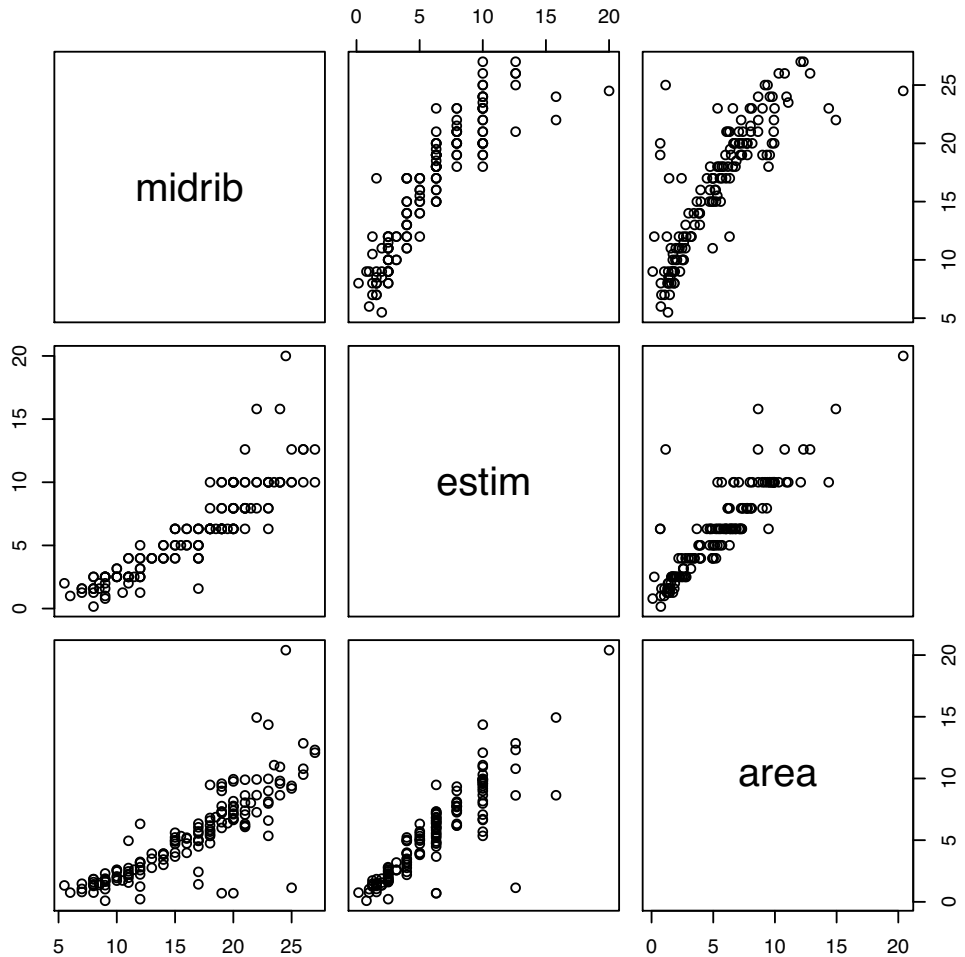


# What if we didn't take logarithms?

```
> expclover <- exp(clover)
> model3 <- lm(area ~ midrib + estim, data=expclover)
> summary(model3)

Call:
lm(formula = area ~ midrib + estim, data = expclover)

Residuals:
     Min       1Q   Median       3Q      Max
-10.0050  -0.3447   0.1299   0.6378   5.2594

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.06609    0.49919  -2.136   0.0344 *
midrib       0.15049    0.05265   2.858   0.0049 **
estim        0.67054    0.08158   8.219 1.16e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.599 on 142 degrees of freedom
Multiple R-squared:  0.7953,        Adjusted R-squared:  0.7924
F-statistic: 275.8 on 2 and 142 DF,  p-value: < 2.2e-16
```

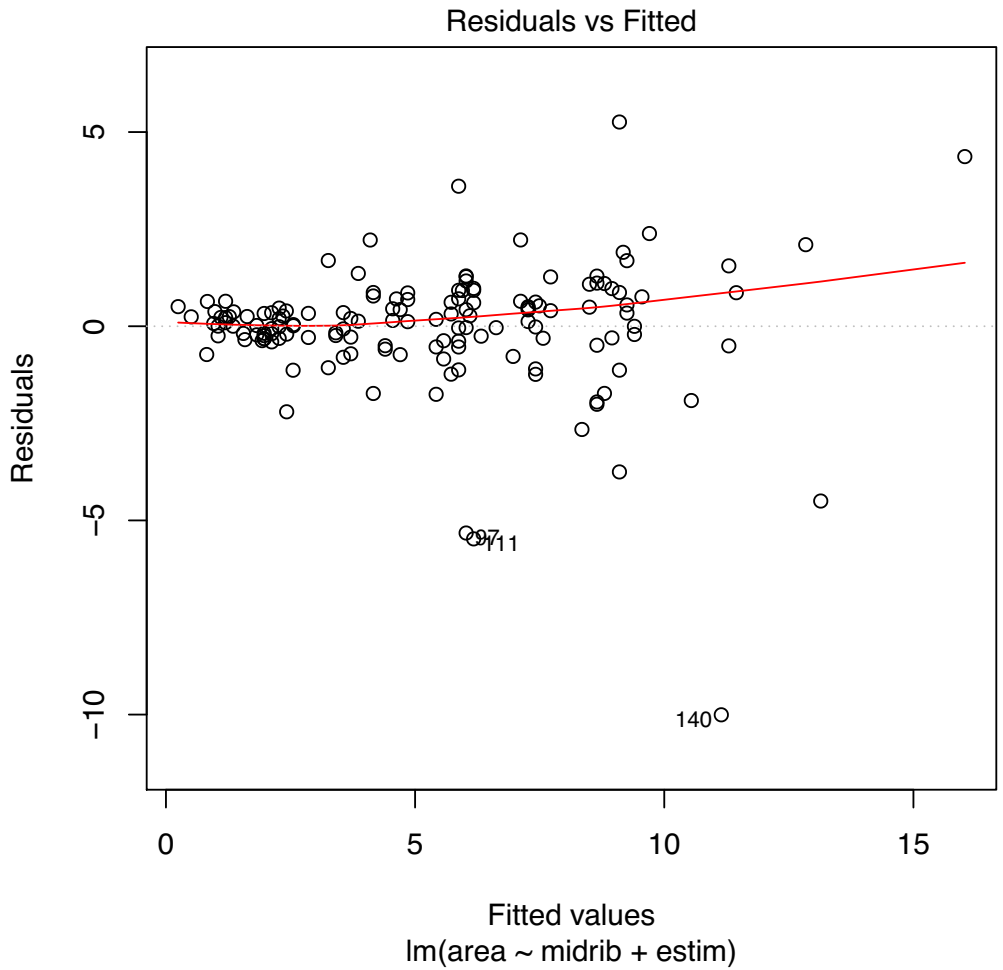
# What if we didn't take logarithms?

```
> pairs(expclover)
```



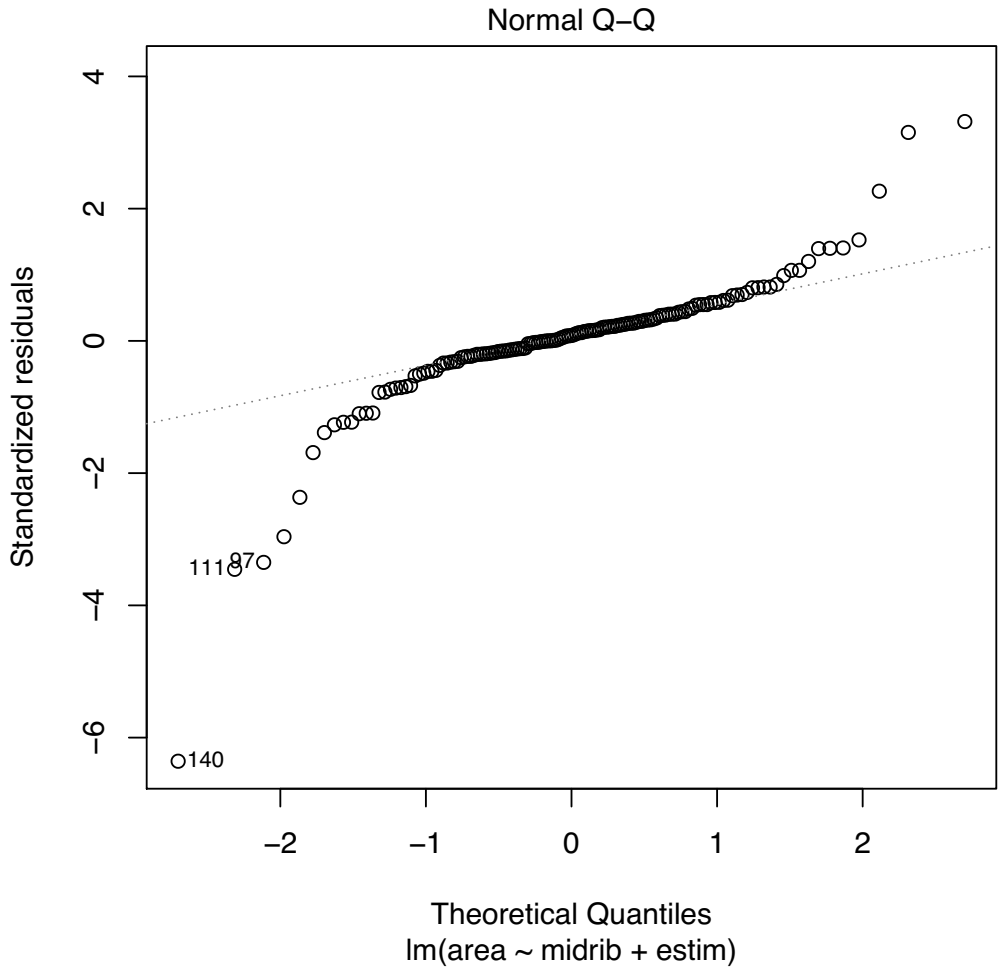
# What if we didn't take logarithms?

```
> plot(model3, which=1)
```



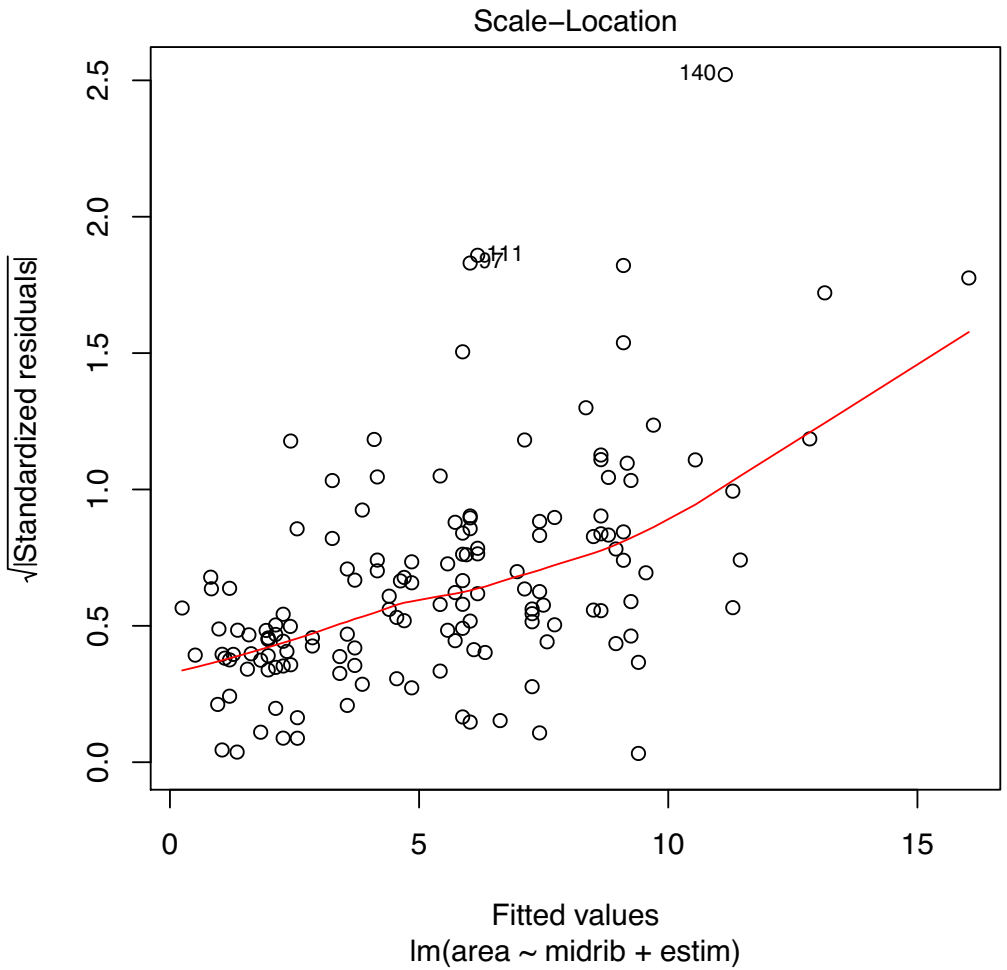
# What if we didn't take logarithms?

```
> plot(model3, which=2)
```



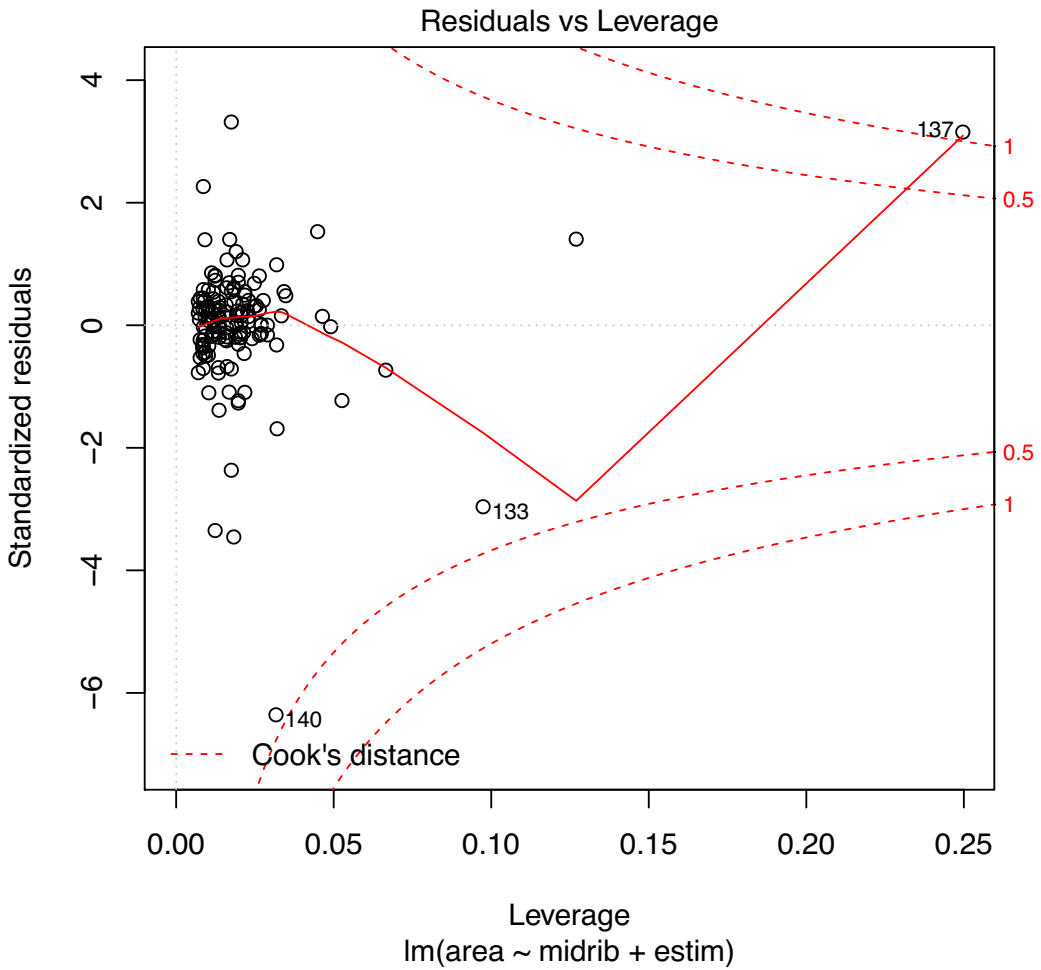
# What if we didn't take logarithms?

```
> plot(model3, which=3)
```



# What if we didn't take logarithms?

```
> plot(model3, which=5)
```



# What if we eliminate the intercept term?

```
> X3 <- cbind(goodclover$midrib,goodclover$estim)
> X3[1:3,]
         [,1]      [,2]
[1,] 1.704748 0.6931472
[2,] 1.791759 0.0000000
[3,] 1.945910 0.4574248

> y3 <- goodclover$area
> (b3 <- solve(t(X3) %*% X3, t(X3) %*% y3))
            [,1]
[1,] -0.04673437
[2,]  1.02242842
```

# What if we eliminate the intercept term?

```
> model4 <- lm(area ~ 0 + midrib + estim, data = goodclover)
> summary(model4)

Call:
lm(formula = area ~ 0 + midrib + estim, data = goodclover)

Residuals:
     Min       1Q   Median       3Q      Max
-0.59989 -0.14717  0.03691  0.12036  0.50081

Coefficients:
       Estimate Std. Error t value Pr(>|t|)
midrib -0.04673    0.02440  -1.915   0.0576 .
estim   1.02243    0.03887  26.302   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2144 on 137 degrees of freedom
Multiple R-squared:  0.9835,        Adjusted R-squared:  0.9832
F-statistic:  4080 on 2 and 137 DF,  p-value: < 2.2e-16
```

# Maximum likelihood estimation

In maximum likelihood estimation (MLE), we choose parameter values to maximise the 'probability' of having observed the given data. We can apply this idea to estimate the parameters of the linear model.

MLEs are popular because they have good *asymptotic* properties: as the sample size goes to $\infty$ they are unbiased, normally distributed, and have minimum variance under certain conditions.

To find MLEs, we need a distribution for the errors. We assume that the errors are $MVN(\mathbf{0}, \sigma^2 I)$. In particular, this means that the errors are independent (not just uncorrelated).

# Maximum likelihood estimation

Given observed values $\mathbf{y}$ of the response variable, the errors are $\mathbf{y} - X\boldsymbol{\beta}$. Since the errors are independent, their joint density is given by

$$\begin{aligned} L(\boldsymbol{\beta}, \sigma^2) &= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\varepsilon_i^2/2\sigma^2} \\ &= \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\sum_{i=1}^{n} \varepsilon_i^2/(2\sigma^2)} \\ &= \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-(\mathbf{y}-X\boldsymbol{\beta})^T(\mathbf{y}-X\boldsymbol{\beta})/(2\sigma^2)}. \end{aligned}$$

Considered as a function of the parameters $\boldsymbol{\beta}$ and $\sigma^2$, this is called the likelihood, and denoted $L(\boldsymbol{\beta}, \sigma^2)$.

# Maximum likelihood estimation

We maximise the likelihood with respect to $\boldsymbol{\beta}$ to generate maximum likelihood estimators for $\boldsymbol{\beta}$. To do this, we differentiate with respect to $\boldsymbol{\beta}$ and set the result to be $\mathbf{0}$.

In practice, it is usually easier to maximise the log-likelihood. Because $\log$ is a monotonic function, the maximum is at the same point.

$$\ln L(\boldsymbol{\beta}, \sigma^2) = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}(\mathbf{y} - X\boldsymbol{\beta})^T(\mathbf{y} - X\boldsymbol{\beta}).$$

# Maximum likelihood estimation

The first term is constant in $\boldsymbol{\beta}$, so using the derivative rules,

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\beta}} \ln L(\boldsymbol{\beta}, \sigma^2) &= -\frac{1}{2\sigma^2} \frac{\partial}{\partial \boldsymbol{\beta}} \left( \mathbf{y}^T \mathbf{y} - 2\mathbf{y}^T X \boldsymbol{\beta} + \boldsymbol{\beta}^T (X^T X) \boldsymbol{\beta} \right) \\
&= -\frac{1}{2\sigma^2} \left( -2(X^T \mathbf{y}) + 2(X^T X)\boldsymbol{\beta} \right) = 0 \\
(X^T X)\boldsymbol{\beta} &= X^T \mathbf{y}.
\end{aligned}$$

This is just the normal equations!

# Maximum likelihood estimation

**Theorem 4.7**

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$*, assume* $\boldsymbol{\varepsilon} \sim MVN(\mathbf{0}, \sigma^2 I)$*. Then the maximum likelihood estimator for* $\boldsymbol{\beta}$ *is also the least squares estimator:*

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}.$$

What about the variance?

# Maximum likelihood estimation

## Theorem 4.8

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$*, assume* $\boldsymbol{\varepsilon} \sim MVN(\mathbf{0}, \sigma^2 I)$*. Then the maximum likelihood estimator for* $\sigma^2$ *is given by*

$$\tilde{\sigma^2} = \frac{SS_{Res}}{n}.$$

This is a biased estimator: the MLE is only asymptotically unbiased. However, the sample variance

$$s^2 = \frac{SS_{Res}}{n-p} = \frac{n}{n-p}\tilde{\sigma^2}$$

has the same asymptotic properties as $\tilde{\sigma^2}$, but is unbiased for all $n$, making it the preferred estimator.

# Sufficiency

We've seen that the least squares estimator is the best linear unbiased estimator for $\boldsymbol{\beta}$, and that if the errors are normally distributed, it is also the maximum likelihood estimator.

We can in fact go a step further: given the assumption of normality, the least squares estimators are *sufficient*. That is, they use all 'relevant' information about the parameters that is contained in the observed response variables.

The Fisher-Neyman Factorization theorem gives a formal characterisation of sufficient statistics.

# Sufficiency

## Theorem 4.9 (Fisher-Neyman Factorization Theorem)

*Let $\mathbf{x}$ be a random variable with parameters $\boldsymbol{\theta}$, and let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be a random sample drawn from this distribution, with joint density $f(\mathbf{x}_1, \ldots, \mathbf{x}_n; \boldsymbol{\theta})$. Then the statistic $\mathbf{y} = u(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ is sufficient for $\boldsymbol{\theta}$ if and only if $f$ can be expressed as*

$$f(\mathbf{x}_1, \ldots, \mathbf{x}_n; \boldsymbol{\theta}) = g(\mathbf{y}; \boldsymbol{\theta}) h(\mathbf{x}_1, \ldots, \mathbf{x}_n).$$

We must be able to factorise the density into one part which depends only on $\mathbf{y}$ and $\boldsymbol{\theta}$, and another part which depends only on the $\mathbf{x}_i$'s.

# Sufficiency

**Theorem 4.10**

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, *assume* $\boldsymbol{\varepsilon} \sim MVN(\mathbf{0}, \sigma^2 I)$. *Then the estimators*

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y} \quad \text{and} \quad s^2 = \frac{SS_{Res}}{n-p}$$

*are jointly sufficient for* $\boldsymbol{\beta}$ *and* $\sigma^2$.

# Sufficiency

Maximum likelihood theory tells us that asymptotically $\mathbf{b}$ and $s^2$ have minimum variance. In fact, this is also true for finite samples:

**Theorem 4.11**

*In the full rank general linear model $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, assume $\boldsymbol{\varepsilon} \sim MVN(\mathbf{0}, \sigma^2 I)$. Then the estimators*

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y} \text{ and } s^2 = \frac{SS_{Res}}{n-p}$$

*have the lowest variance among all unbiased estimators of $\boldsymbol{\beta}$ and $\sigma^2$.*

This is a stronger condition than BLUE because it includes non-linear estimators. We call this UMVUE (uniformly minimum variance unbiased estimator).

# Interval estimation

The least squares estimator gives excellent *point* estimates for the parameters. But this only tells half the story.

To get an idea of how accurate these estimates are, we would like to find *interval* estimates.

We first need to know the distribution of our least squares estimators. This requires an assumption on the distribution of the errors.

Remember that we assume $\boldsymbol{\varepsilon} \sim MVN(\mathbf{0}, \sigma^2 I)$ for maximum likelihood estimation, but this is not required to derive the least squares estimators. We assume this from now on.

# Interval estimation

Now $\mathbf{y}$ and $\mathbf{b}$ are linear combinations of $\boldsymbol{\varepsilon}$, so they also have multivariate normal distributions.

**Theorem 4.12**

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, *assume* $\boldsymbol{\varepsilon} \sim MVN(\mathbf{0}, \sigma^2 I)$. *Then*

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}$$

*has a multivariate normal distribution with mean* $\boldsymbol{\beta}$ *and variance* $(X^T X)^{-1}\sigma^2$.

# Interval estimation

What about the sample variance?

**Theorem 4.13**

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$*, assume* $\boldsymbol{\varepsilon} \sim MVN(\mathbf{0}, \sigma^2 I)$*. Then*

$$\frac{(n-p)s^2}{\sigma^2} = \frac{SS_{Res}}{\sigma^2}$$

*has a* $\chi^2$ *distribution with* $n - p$ *degrees of freedom.*

# The hat matrix

The proof (and some others) are easier if we use the hat matrix, $H = X(X^T X)^{-1} X^T$. We have the properties:

- $H$ is symmetric and idempotent;
- $r(H) = p = k + 1$;
- $I - H$ is symmetric and idempotent;
- $r(I - H) = n - p$;
- $HX = X(X^T X)^{-1} X^T X = X$, $X^T H = X^T$.

# Interval estimation

**Proof.** We have shown earlier that the residual sum of squares can be expressed as the quadratic form

$$\begin{aligned} SS_{Res} &= (\mathbf{y} - X\mathbf{b})^T(\mathbf{y} - X\mathbf{b}) \\ &= \mathbf{y}^T[I - X(X^TX)^{-1}X^T]\mathbf{y} \\ &= \mathbf{y}^T[I - H]\mathbf{y}. \end{aligned}$$

By assumption, $\mathbf{y} \sim MVN(X\boldsymbol{\beta}, \sigma^2 I)$.

By Corollary 3.7, $\frac{1}{\sigma^2}\mathbf{y}^T[I - H]\mathbf{y}$ has a noncentral $\chi^2$ distribution, with $n - p$ d.f. and noncentrality parameter

$$\lambda = \frac{1}{2\sigma^2}\boldsymbol{\mu}^T[I - H]\boldsymbol{\mu}.$$

# Interval estimation

But $\boldsymbol{\mu} = X\boldsymbol{\beta}$, so

$$\begin{aligned}
\lambda &= \frac{1}{2\sigma^2}(X\boldsymbol{\beta})^T[I-H]X\boldsymbol{\beta} \\
&= \frac{1}{2\sigma^2}[\boldsymbol{\beta}^T X^T X\boldsymbol{\beta} - \boldsymbol{\beta}^T X^T HX\boldsymbol{\beta}] \\
&= \frac{1}{2\sigma^2}[\boldsymbol{\beta}^T X^T X\boldsymbol{\beta} - \boldsymbol{\beta}^T X^T X\boldsymbol{\beta}] \\
&= 0.
\end{aligned}$$

Thus $\frac{SS_{Res}}{\sigma^2}$ has a (central) $\chi^2$ distribution with $n-p$ degrees of freedom.

# Interval estimation

## Theorem 4.14

*In the full rank general linear model* $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$*, assume* $\boldsymbol{\varepsilon} \sim MVN(\mathbf{0}, \sigma^2 I)$*. Then* $\mathbf{b}$ *and* $s^2$ *are independent.*

**Proof.** We use Theorem 3.13. We have

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}, \quad \frac{SS_{Res}}{\sigma^2} = \mathbf{y}^T \frac{[I - H]}{\sigma^2} \mathbf{y}$$

and so

$$\begin{aligned} BVA &= (X^T X)^{-1} X^T \sigma^2 I \frac{[I - H]}{\sigma^2} \\ &= (X^T X)^{-1} X^T - (X^T X)^{-1} X^T H \\ &= 0. \end{aligned}$$

# The $t$ distribution

## Definition 4.15

Let $Z$ be a standard normal random variable and let $X_\gamma^2$ be an independent $\chi^2$ random variable with $\gamma$ degrees of freedom. Then

$$\frac{Z}{\sqrt{X_\gamma^2/\gamma}}$$

has a $t$ distribution with $\gamma$ degrees of freedom.

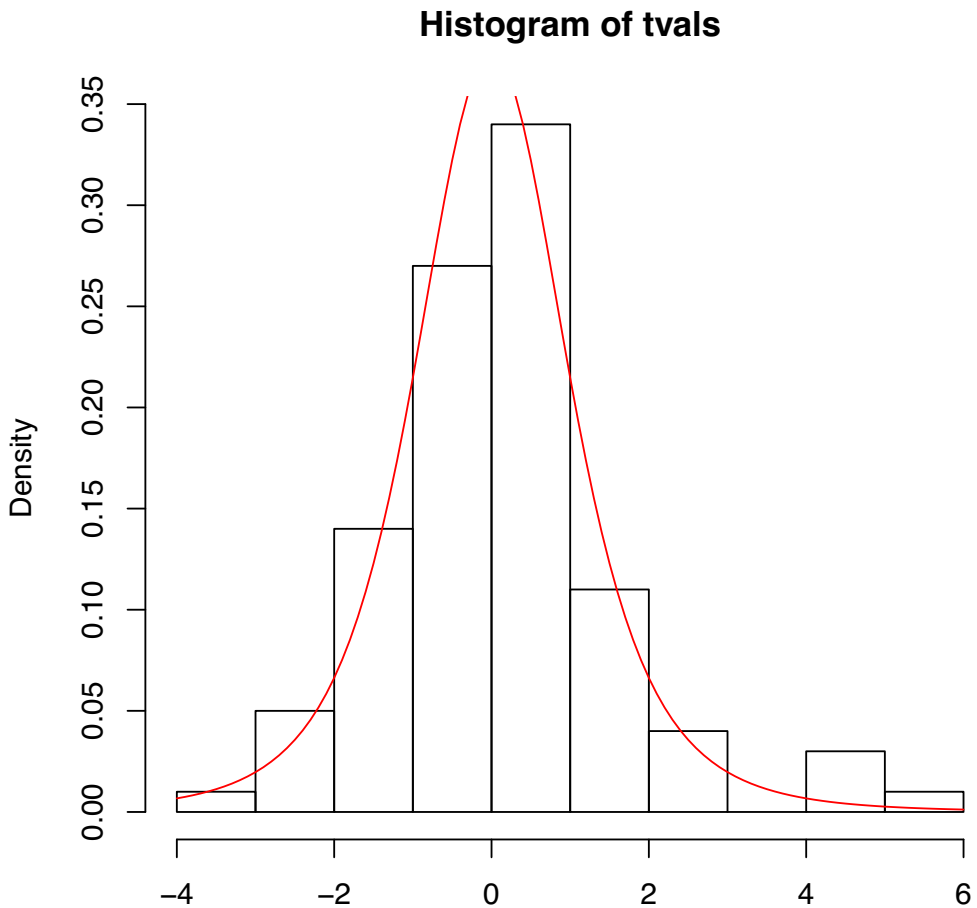
The density of the $t$ distribution is

$$f(x) = \frac{\Gamma((\gamma+1)/2)}{\sqrt{\gamma\pi}\Gamma(\gamma/2)}\left(1+\frac{x^2}{\gamma}\right)^{-(\gamma+1)/2}.$$

# $t$ time

```
> Z <- rnorm(100)
> X2 <- rchisq(100,4)
> tvals <- Z/sqrt(X2/4)
> hist(tvals,freq=FALSE)
> curve(dt(x,4),add=TRUE,col='red')
```

# $t$ time


Histogram of tvals
Density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
−4
−2
0
2
4
6

# Interval estimation

We can now create confidence intervals for the parameters. Firstly we will find a confidence interval for a single parameter, $\beta_i$.

Consider the covariance matrix of $\mathbf{b}$:

$$(X^T X)^{-1}\sigma^2 = \begin{bmatrix} c_{00} & c_{01} & \cdots & c_{0k} \\ c_{10} & c_{11} & \cdots & c_{1k} \\ \vdots & & \ddots & \vdots \\ c_{k0} & c_{k1} & \cdots & c_{kk} \end{bmatrix} \sigma^2.$$

# Interval estimation

The least squares estimator of $\beta_i$ is $b_i$. The variance of $b_i$ is the $i$th diagonal element of the covariance matrix, denoted $c_{ii}\sigma^2$.

Since $b_i$ is normal, this means that

$$\frac{b_i - \beta_i}{\sigma\sqrt{c_{ii}}}$$

has a standard normal distribution.

Of course, we do not know what $\sigma$ is...

# Interval estimation

...but from the above theory,

$$\left(\frac{b_i - \beta_i}{\sigma\sqrt{c_{ii}}}\right) / \left(\sqrt{\frac{SS_{Res}/\sigma^2}{n-p}}\right)$$

has a $t$ distribution with $n - p$ degrees of freedom.

Simplifying gives

$$\left(\frac{b_i - \beta_i}{\sigma\sqrt{c_{ii}}}\right) / \left(\sqrt{\frac{s^2}{\sigma^2}}\right) = \frac{b_i - \beta_i}{s\sqrt{c_{ii}}}.$$

# Interval estimation

It is now easy to derive a $100(1-\alpha)\%$ confidence interval:

$$\begin{aligned} P[-t_{\alpha/2} \leq (b_i - \beta_i)/(s\sqrt{c_{ii}}) \leq t_{\alpha/2}] &= 1-\alpha \\ P[-t_{\alpha/2} s\sqrt{c_{ii}} \leq b_i - \beta_i \leq t_{\alpha/2} s\sqrt{c_{ii}}] &= 1-\alpha \\ P[b_i - t_{\alpha/2} s\sqrt{c_{ii}} \leq \beta_i \leq b_i + t_{\alpha/2} s\sqrt{c_{ii}}] &= 1-\alpha. \end{aligned}$$

Therefore the confidence interval (using a $t$ distribution with $n-p$ d.f.) is

$$b_i \pm t_{\alpha/2} s\sqrt{c_{ii}},$$

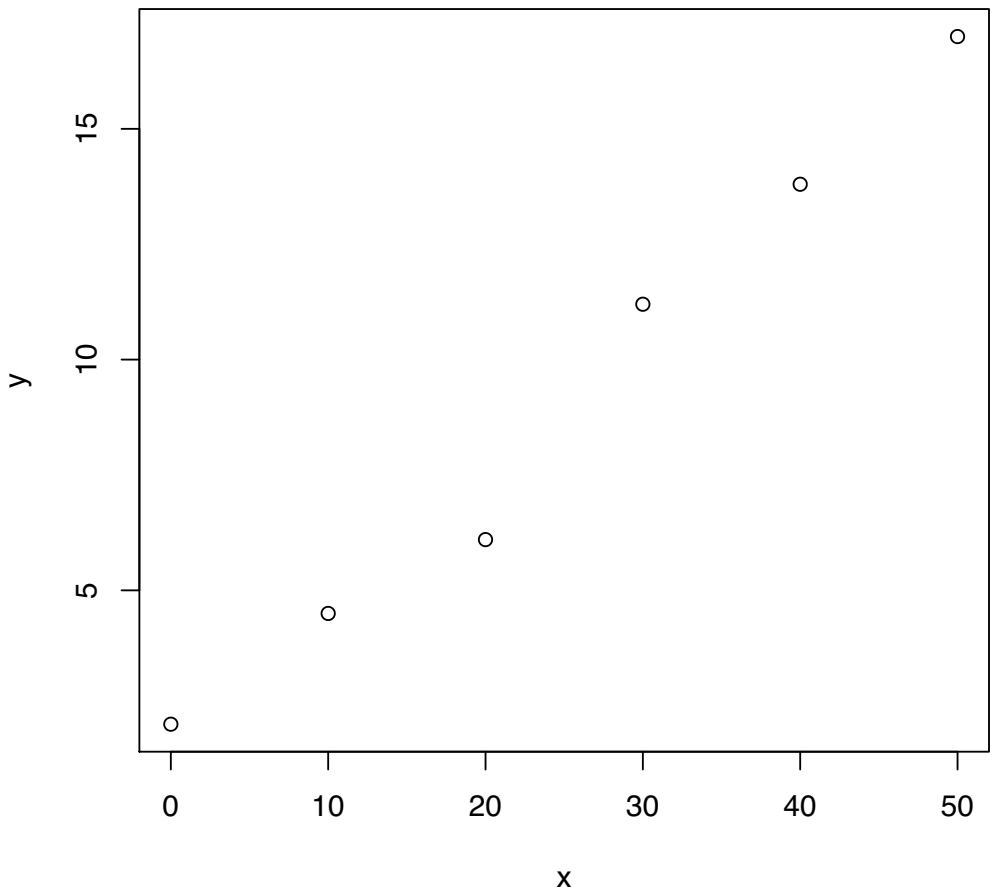
where $c_{ii}$ is the $i$th diagonal element of $(X^T X)^{-1}$.

# Interval estimation

**Example.** We model the amount of a chemical that dissolves in a fixed volume of water. This depends (in part) on the water temperature. An experiment is run $6$ times and the following data measured:

| Temperature ($x$) | Amount dissolved ($y$) |
|---:|---:|
| 0 | 2.1 |
| 10 | 4.5 |
| 20 | 6.1 |
| 30 | 11.2 |
| 40 | 13.8 |
| 50 | 17.0 |

# Interval estimation



# Interval estimation

```
> y <- c(2.1, 4.5, 6.1, 11.2, 13.8, 17.0)
> X <- matrix(c(rep(1,6),seq(0,50,10)),6,2)
> (b <- solve(t(X)%*%X, t(X)%*%y))

          [,1]
[1,] 1.4380952
[2,] 0.3071429

> (df <- 6-2)

[1] 4

> e <- y - X%*%b
> (s <- sqrt(sum(e^2)/df))

[1] 0.8629959
```

# Interval estimation

First we find a confidence interval on $\beta_0$, the intercept.

```
> c00 <- solve(t(X)%*%X)[1,1]
> alpha <- 0.05
> ta <- qt(1-alpha/2, df=df)
> b[1] + c(-1,1)*ta*s*sqrt(c00)
```

```
[1] -0.2960462  3.1722367
```

We are $95\%$ confident that the true amount of chemical dissolved at $0$ temperature lies between $-0.30$ and $3.17$.

Notably, we cannot say with $95\%$ confidence that it is untrue that no chemical dissolves at 0 temperature.

# Interval estimation

Next we find a confidence interval on $\beta_1$, the slope of the regression.

```
> c11 <- solve(t(X)%*%X)[2,2]
> b[2] + c(-1,1)*ta*s*sqrt(c11)
```

```
[1] 0.2498661 0.3644197
```

We are $95\%$ confident that for each rise in temperature of 1 degree, the amount of chemical dissolved goes up by an amount between 0.25 and 0.36.

In particular, we are (at least) $95\%$ sure that there is a positive relationship between temperature and chemical dissolved.

# Interval estimation

It is good that we can find confidence intervals for the parameters, but sometimes we want to estimate things other than just the parameters.

In particular, we often want to predict the value of the response variable for a given set of inputs.

This is an example of the more general case of linear functions of the parameters.

# Interval estimation

Remember that if we want to estimate the function $\mathbf{t}^T\boldsymbol{\beta}$, the best linear unbiased estimator is $\mathbf{t}^T\mathbf{b}$, where $\mathbf{b}$ is the least squares estimator of the parameters. What is its distribution?

Since $\mathbf{b}$ is multivariate normal, any linear combination of $b$'s is normally distributed. We have

$$E[\mathbf{t}^T\mathbf{b}] = \mathbf{t}^T\boldsymbol{\beta}$$

since $\mathbf{b}$ is an unbiased estimator for $\boldsymbol{\beta}$.

# Interval estimation

Variance results give us

$$\text{Var } \mathbf{t}^T\mathbf{b} = \mathbf{t}^T(X^TX)^{-1}\sigma^2\mathbf{t} = \mathbf{t}^T(X^TX)^{-1}\mathbf{t}\sigma^2.$$

Therefore

$$\frac{\mathbf{t}^T\mathbf{b} - \mathbf{t}^T\boldsymbol{\beta}}{\sqrt{\mathbf{t}^T(X^TX)^{-1}\mathbf{t}\sigma^2}}$$

has a standard normal distribution.

But again, we do not know what $\sigma$ is!

# Interval estimation

The solution should not be difficult to see: since $SS_{Res}/\sigma^2$ is independent of $\mathbf{b}$, it is independent of $\mathbf{t}^T\mathbf{b}$. Therefore

$$\frac{(\mathbf{t}^T\mathbf{b} - \mathbf{t}^T\boldsymbol{\beta})/(\sqrt{\mathbf{t}^T(X^TX)^{-1}\mathbf{t}\sigma^2})}{\sqrt{SS_{Res}/\sigma^2(n-p)}} = \frac{\mathbf{t}^T\mathbf{b} - \mathbf{t}^T\boldsymbol{\beta}}{s\sqrt{\mathbf{t}^T(X^TX)^{-1}\mathbf{t}}}$$

has a $t$ distribution with $n - p$ degrees of freedom.

Using similar steps to before, this gives the $100(1-\alpha)\%$ confidence interval

$$\mathbf{t}^T\mathbf{b} \pm t_{\alpha/2}s\sqrt{\mathbf{t}^T(X^TX)^{-1}\mathbf{t}}.$$

# Interval estimation

In particular, if we want to find a confidence interval for the expected response to a particular set of $x$ variables $x_1^*, x_2^*, \ldots, x_k^*$, we wish to predict

$$E[y] = \beta_0 + \beta_1 x_1^* + \ldots + \beta_k x_k^* = (\mathbf{x}^*)^T \boldsymbol{\beta}$$

where $\mathbf{x}^* = \begin{bmatrix} 1 & x_1^* & x_2^* & \ldots & x_k^* \end{bmatrix}^T$.

This is a linear function of $\boldsymbol{\beta}$, and therefore the $100(1-\alpha)\%$ confidence interval for it is

$$(\mathbf{x}^*)^T \mathbf{b} \pm t_{\alpha/2} s \sqrt{(\mathbf{x}^*)^T (X^T X)^{-1} \mathbf{x}^*}.$$

# Interval estimation

**Example.** In the house price example, we estimated the average selling price of a 15-year-old house with an area of 250 $m^2$ to be \$570,129. What is the 95% confidence interval for this number?

```
> (s <- sqrt(s2))

[1] 6.916497

> xst <- c(1,15,2.5)
> xst%*%b

         [,1]
[1,] 57.01289
```

# Interval estimation

**Example.** In the house price example, we estimated the average selling price of a 15-year-old house with an area of 250 $m^2$ to be \$570,129. What is the 95% confidence interval for this number?

```
> (ta <- qt(0.975,df=5-3))

[1] 4.302653

> xst%*%b - ta*s*sqrt(t(xst)%*%solve(t(X)%*%X)%*%xst)

          [,1]
[1,] 37.83522

> xst%*%b + ta*s*sqrt(t(xst)%*%solve(t(X)%*%X)%*%xst)

          [,1]
[1,] 76.19056
```

# Prediction intervals

Given a set of inputs, a $95\%$ confidence interval for the response gives an interval that contains the *expected* response $95\%$ of the time.

In contrast, given a set of inputs, a $95\%$ prediction interval produces an interval in which we are $95\%$ sure that *any given response with those inputs* lies in.

Because a single observation is more variable than the expected response, a prediction interval is wider than the corresponding confidence interval.

# Prediction intervals

Suppose we have inputs $\mathbf{x}^* = \begin{bmatrix} 1 & x_1^* & x_2^* & \dots & x_k^* \end{bmatrix}^T$, with corresponding response

$$y^* = (\mathbf{x}^*)^T \boldsymbol{\beta} + \varepsilon^*$$

where Var $\varepsilon^* = \sigma^2$ by assumption.

This will be (point) estimated by $(\mathbf{x}^*)^T \mathbf{b}$ with an error of

$$y^* - (\mathbf{x}^*)^T \mathbf{b} = (\mathbf{x}^*)^T \boldsymbol{\beta} + \varepsilon^* - (\mathbf{x}^*)^T \mathbf{b}.$$

# Prediction intervals

Only the last two terms are random. $\varepsilon^*$ is an error associated with the future observation $y^*$, and $\mathbf{b}$ depends only on the current observations $\mathbf{y}$. So we can say that they are independent.

This gives

$$\begin{aligned} \text{Var}\,(y^* - (\mathbf{x}^*)^T\mathbf{b}) &= \text{Var}\,\varepsilon^* + \text{Var}\,(\mathbf{x}^*)^T\mathbf{b} \\ &= \sigma^2 + (\mathbf{x}^*)^T(X^TX)^{-1}\sigma^2\mathbf{x}^* \\ &= [1 + (\mathbf{x}^*)^T(X^TX)^{-1}\mathbf{x}^*]\sigma^2 \end{aligned}$$

and since the estimator is unbiased, the expectation is $\mathbf{0}$.

# Prediction intervals

Following exactly the previous arguments, we derive that

$$\frac{y^* - (\mathbf{x}^*)^T\mathbf{b}}{s\sqrt{1 + (\mathbf{x}^*)^T(X^TX)^{-1}\mathbf{x}^*}}$$

has a $t$ distribution with $n - p$ degrees of freedom.

Thus a prediction interval for $y^*$ is

$$(\mathbf{x}^*)^T\mathbf{b} \pm t_{\alpha/2}s\sqrt{1 + (\mathbf{x}^*)^T(X^TX)^{-1}\mathbf{x}^*}.$$

The only difference with confidence intervals is the presence of the '1', which makes the interval wider (as expected).

# Prediction intervals

**Example.** In the previous example, we estimated the *average* selling price of a 15-year-old house with area 250 $m^2$ to be in the range [37.84,76.19].

What is the prediction interval for a *single* such house?

```
> xst%*%b - ta*s*sqrt(1+t(xst)%*%solve(t(X)%*%X)%*%xst)

          [,1]
[1,] 21.60953
```

```
> xst%*%b + ta*s*sqrt(1+t(xst)%*%solve(t(X)%*%X)%*%xst)

          [,1]
[1,] 92.41626
```

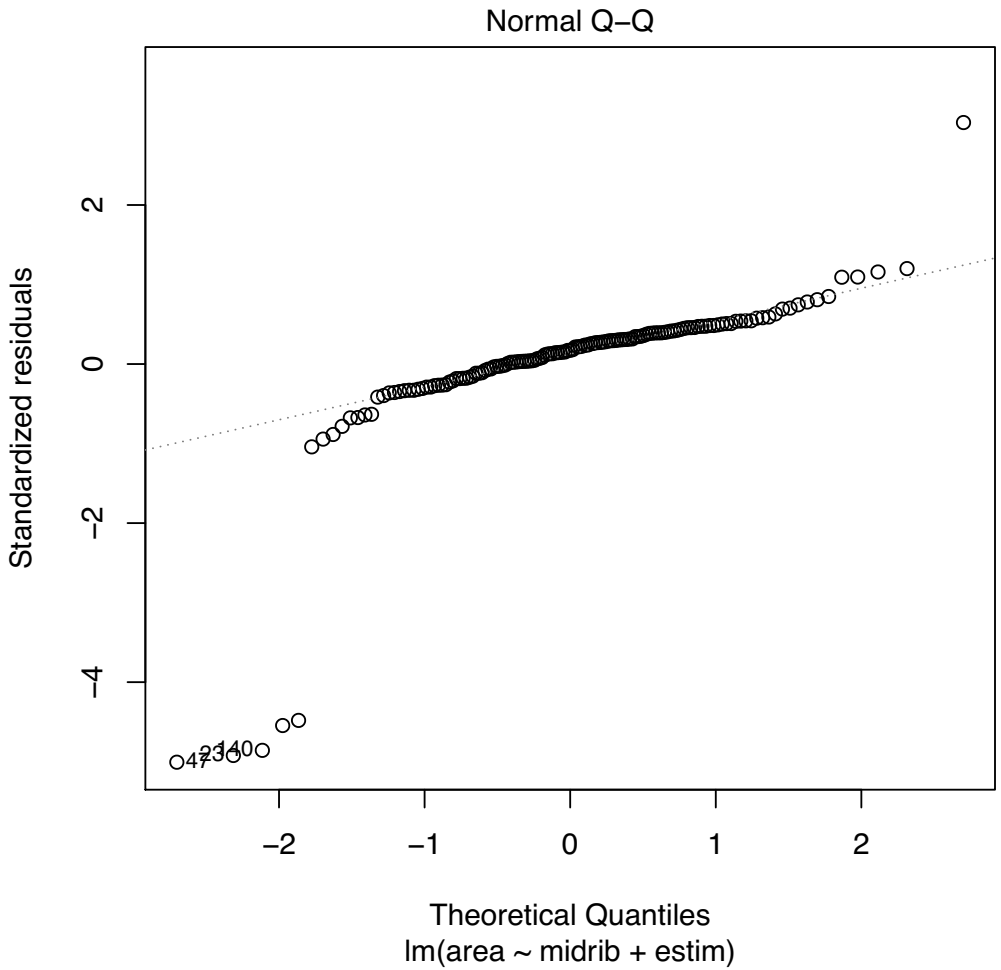
This is wider than the confidence interval for the mean.

# Clover example

We need i.i.d. normal errors for our confidence intervals to be accurate. We check this using a normal quantile-quantile plot.

```
> clover <- read.csv("../data/clover.csv")
> clover <- log(clover)
> model <- lm(area ~ midrib + estim,data=clover)
> plot(model, which=2)
```

# Clover example



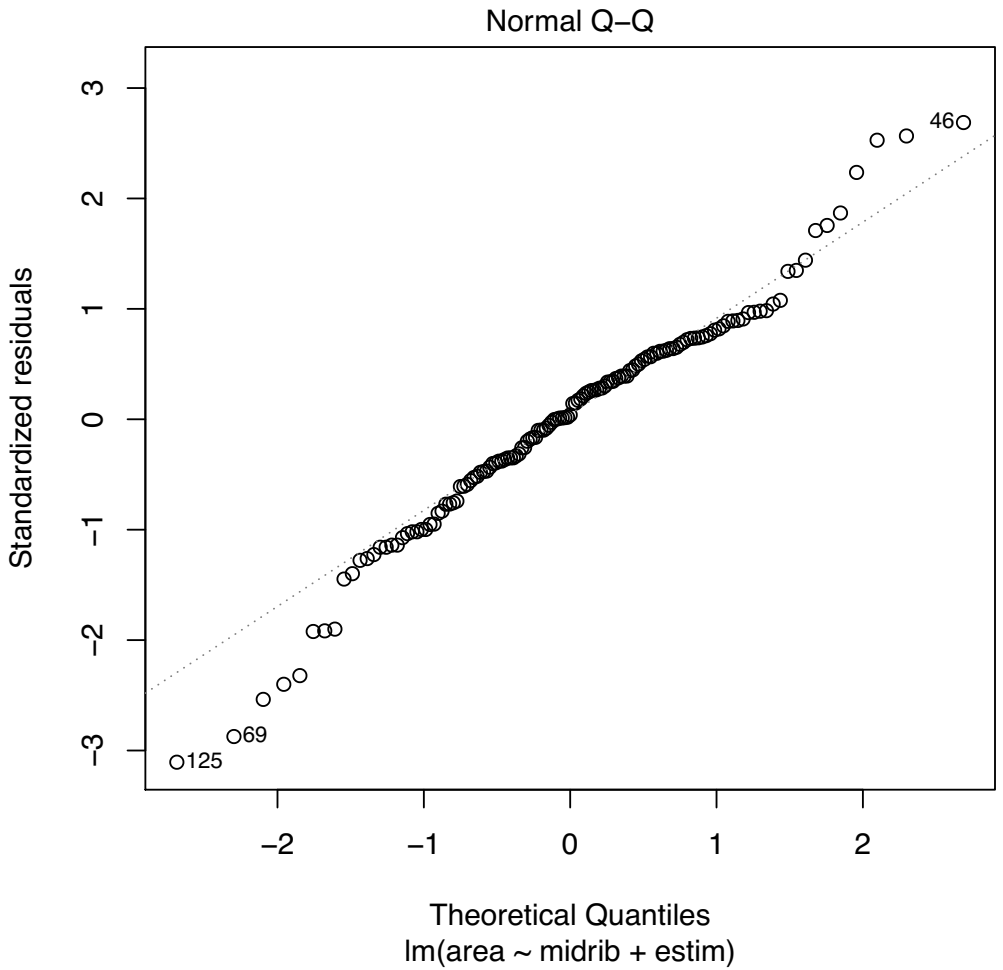
# Clover example

This is not good, so we remove the outliers and try again.

```
> goodclover <- clover[-c(6, 23, 47, 97, 111, 140), ]
> model2 <- lm(area ~ midrib + estim, data=goodclover)
> plot(model2, which=2)
```

The result is an improvement (though still not brilliant).

# Clover example



# Clover example

```
> y <- goodclover$area
> X <- cbind(1,goodclover$midrib,goodclover$estim)
> n <- dim(X)[1]
> p <- dim(X)[2]
> b <- solve(t(X) %*% X, t(X) %*% y)
> e <- y - X %*% b
> SSRes <- sum(e^2)
> s2 <- SSRes/(n-p)
```

# Clover example

95% confidence interval for $\beta_0$, the intercept:

```
> C <- solve(t(X) %*% X)
> b[1] + c(-1,1)*qt(0.975,df=n-p)*sqrt(s2*C[1,1])
```

```
[1] -1.7871886 -0.9757665
```

95% confidence interval for $\beta_1$, the `midrib` coefficient:

```
> b[2] + c(-1,1)*qt(0.975,df=n-p)*sqrt(s2*C[2,2])
```

```
[1] 0.4413948 0.8593518
```

# Clover example

95% confidence interval for $\beta_2$, the `estim` coefficient:

```
> b[3] + c(-1,1)*qt(0.975,df=n-p)*sqrt(s2*C[3,3])
[1] 0.5741688 0.8098116
```

```
> confint(model2, level=0.95)
                 2.5 %     97.5 %
(Intercept) -1.7871886 -0.9757665
midrib       0.4413948  0.8593518
estim        0.5741688  0.8098116
```

# Clover example

95% confidence interval for the expected area of a leaf with midrib 10 and template area 10:

```
> tt <- c(1,log(10),log(10))
> halfwidth <- qt(0.975,df=n-p)*sqrt(s2 * t(tt) %*% C %*% tt)
> tt %*% b + c(-1,1)*halfwidth
```

```
[1] 1.538316 1.880541
```

```
> newclover <- data.frame(midrib=log(10),estim=log(10))
> predict(model2,newclover,interval="confidence",level=0.95)
```

```
       fit      lwr      upr
1 1.709429 1.538316 1.880541
```

# Clover example

95% *prediction* interval of the area of a leaf with midrib 10 and template area 10:

```
> halfwidth <- qt(0.975,df=n-p)*
+      sqrt(s2 * (1 + t(tt) %*% C %*% tt))
> tt %*% b + c(-1,1)*halfwidth
```

```
[1] 1.303147 2.115710
```

```
> predict(model2,newclover,interval="prediction",level=0.95)
```

```
       fit      lwr     upr
1 1.709429 1.303147 2.11571
```

# What about $\sigma^2$?

We can also find a confidence interval for the error variance $\sigma^2$. Once again, we work from the distribution of the estimator. We have $\frac{(n-p)s^2}{\sigma^2} \sim \chi^2_{n-p}$. Hence

$$\begin{aligned} P\left[\chi^2_{n-p}\left(\frac{\alpha}{2}\right) \leq \frac{(n-p)s^2}{\sigma^2} \leq \chi^2_{n-p}\left(1-\frac{\alpha}{2}\right)\right] &= 1-\alpha \\ P\left[\frac{(n-p)s^2}{\chi^2_{n-p}\left(1-\frac{\alpha}{2}\right)} \leq \sigma^2 \leq \frac{(n-p)s^2}{\chi^2_{n-p}\left(\frac{\alpha}{2}\right)}\right] &= 1-\alpha \end{aligned}$$

giving a $100(1-\alpha)\%$ confidence interval of

$$\left(\frac{(n-p)s^2}{\chi^2_{n-p}\left(1-\frac{\alpha}{2}\right)}, \frac{(n-p)s^2}{\chi^2_{n-p}\left(\frac{\alpha}{2}\right)}\right).$$

# Clover example

```
> (n-p)*s2/qchisq(c(0.975,0.025), n-p)
[1] 0.02774825 0.04471258
```

# Joint confidence intervals

Sometimes we want confidence intervals for more than one parameter, or linear combination of parameters, at once.

Finding confidence intervals individually for each parameter is misleading. If we find more than one $95\%$ confidence interval, we do *not* have $95\%$ confidence that all of them will be satisfied at once. The more confidence intervals we have, the more likely it is that at least one will be wrong!

We need to find a *joint* confidence *region* for a number of parameters at the same time.

# $F$ distribution

**Definition 4.16**

Let $X^2_{\gamma_1}$ and $X^2_{\gamma_2}$ be independent $\chi^2$ random variables with $\gamma_1$ and $\gamma_2$ degrees of freedom. Then

$$\frac{X^2_{\gamma_1}/\gamma_1}{X^2_{\gamma_2}/\gamma_2}$$

has an $F$ distribution with $\gamma_1$ and $\gamma_2$ degrees of freedom.

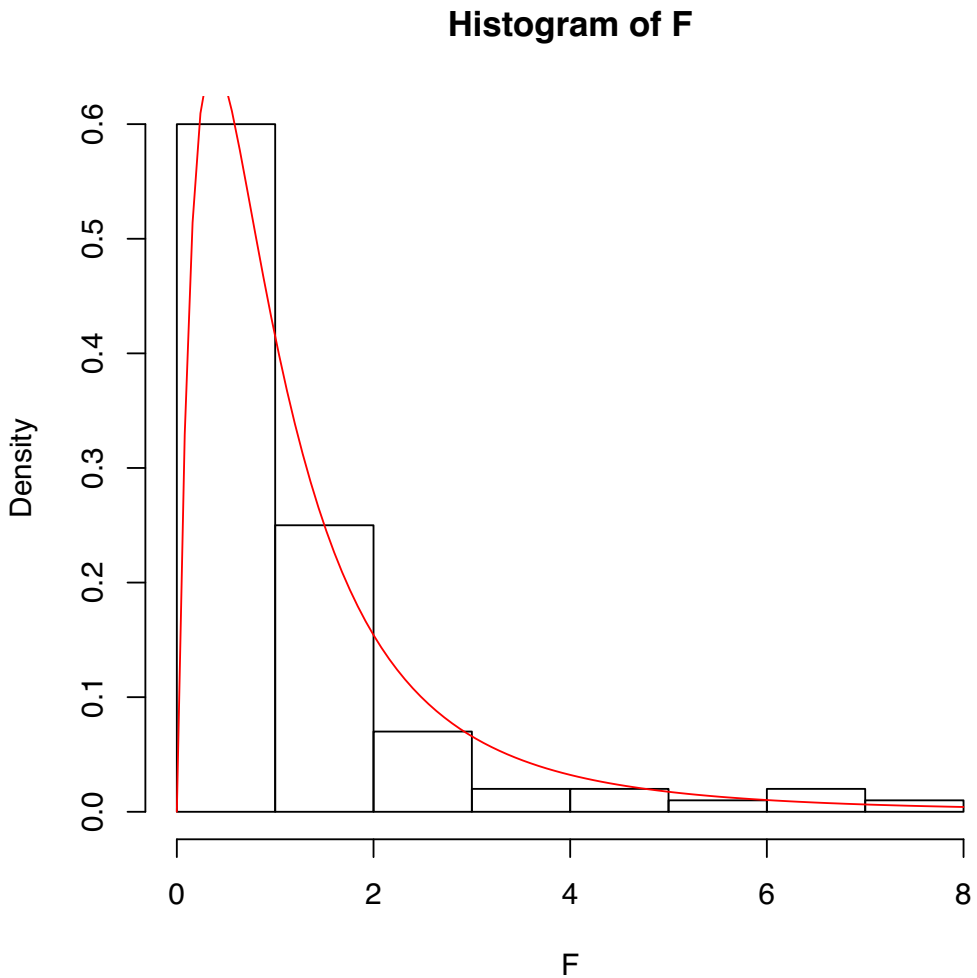
The $F$ distribution has the density

$$f(x;\gamma_1,\gamma_2) = \frac{1}{\beta(\gamma_1/2,\gamma_2/2)}\left(\frac{\gamma_1}{\gamma_2}\right)^{\gamma_1/2} x^{\gamma_1/2-1}\left(1+\frac{\gamma_1}{\gamma_2}x\right)^{-(\gamma_1+\gamma_2)/2}.$$

# $F$ distribution

```
> X1 <- rchisq(100,4)
> X2 <- rchisq(100,6)
> F <- (X1/4)/(X2/6)
> hist(F, freq=FALSE)
> curve(df(x,4,6), add=TRUE,col='red')
```

# $F$ distribution



# Joint confidence intervals

Let's derive a confidence region for $\boldsymbol{\beta}$. The least squares estimator is

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y} \sim MVN(\boldsymbol{\beta}, (X^T X)^{-1}\sigma^2).$$

From Corollary 3.10, the quadratic form

$$\frac{(\mathbf{b} - \boldsymbol{\beta})^T X^T X (\mathbf{b} - \boldsymbol{\beta})}{\sigma^2}$$

has a $\chi^2$ distribution with $p$ degrees of freedom (where $p$ is the number of parameters in the model).

We also know that

$$\frac{(n-p)s^2}{\sigma^2}$$

has a $\chi^2$ distribution with $n - p$ degrees of freedom.

# Joint confidence intervals

Since $\mathbf{b}$ and $s^2$ are independent, the two $\chi^2$ variables above are independent, which means that

$$\left(\frac{(\mathbf{b}-\boldsymbol{\beta})^T X^T X(\mathbf{b}-\boldsymbol{\beta})}{p\sigma^2}\right) / \left(\frac{(n-p)s^2}{(n-p)\sigma^2}\right)$$
$$= \frac{(\mathbf{b}-\boldsymbol{\beta})^T X^T X(\mathbf{b}-\boldsymbol{\beta})}{ps^2}$$

has an $F$ distribution with $p$ and $n-p$ degrees of freedom.

Because this statistic is based on $\mathbf{b}-\boldsymbol{\beta}$, which we want to be small, we use the right-hand tail of the $F$-distribution to create a confidence region.

# Joint confidence intervals

Let $f_\alpha$ be the critical value of the $F$ distribution with $p$ and $n-p$ d.f. and probability $\alpha$, then

$$P[(\mathbf{b}-\boldsymbol{\beta})^T X^T X(\mathbf{b}-\boldsymbol{\beta})/ps^2 \leq f_\alpha] = 1-\alpha$$

which gives the confidence region

$$(\mathbf{b}-\boldsymbol{\beta})^T X^T X(\mathbf{b}-\boldsymbol{\beta}) \leq ps^2 f_\alpha.$$

This region has the form of an ellipse or ellipsoid.

# Joint confidence intervals

**Example.** Modelling income against years of formal education. The data is

| Years of education | Income |
|---|---|
| 8 | 8 |
| 12 | 15 |
| 14 | 16 |
| 16 | 20 |
| 16 | 25 |
| 20 | 40 |

# Joint confidence intervals

```
> n <- 6
> p <- 2
> y <- c(8,15,16,20,25,40)
> X <- matrix(c(rep(1,n),8,12,14,16,16,20),n,p)
> t(X)%*%X
     [,1] [,2]
[1,]    6   86
[2,]   86 1316
> (b <- solve(t(X)%*%X,t(X)%*%y))
         [,1]
[1,] -15.568
[2,]   2.528
> (s2 <- sum((y-X%*%b)^2)/(n-p))
[1] 18.692
```

# Joint confidence intervals

So a joint 95% confidence interval is given by

$$\begin{bmatrix} -15.57-\beta_0 & 2.53-\beta_1 \end{bmatrix} \begin{bmatrix} 6 & 86 \\ 86 & 1316 \end{bmatrix} \begin{bmatrix} -15.57-\beta_0 \\ 2.53-\beta_1 \end{bmatrix} \leq 2 \times 18.69 \times 6.94$$

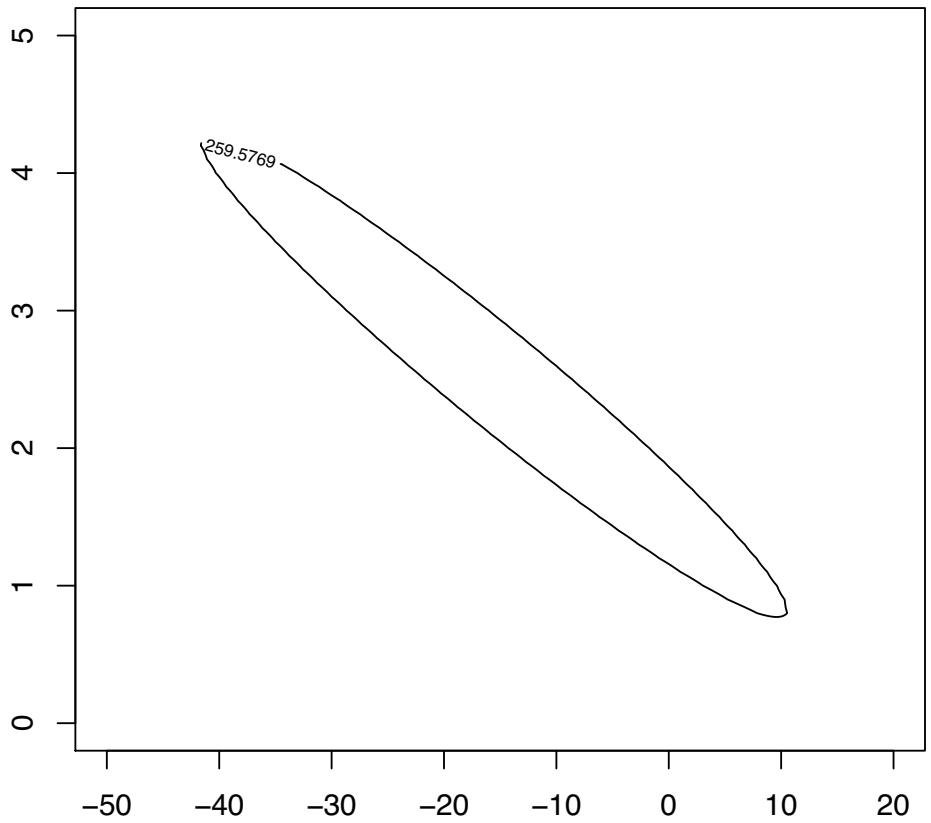
which simplifies to

$$6\beta_0^2 + 1316\beta_1^2 + 172\beta_0\beta_1 - 248.3\beta_0 - 3981\beta_1 + 3103 \leq 259.6.$$

# Joint confidence intervals

```
> b1 <- seq(-50, 20, .2)
> b2 <- seq(0, 5, .1)
> f <- function(beta1, beta2) {
+     f.out <- rep(0, length(beta1))
+     for (i in 1:length(beta1)) {
+         beta <- matrix(c(beta1[i], beta2[i]), 2, 1)
+         f.out[i] <- t(b - beta) %*% t(X) %*% X %*% (b - beta)
+     }
+     return(f.out)
+ }
> z <- outer(b1, b2, f)
> contour(b1, b2, z, levels=2*18.69*qf(0.95, 2, 4))
```

# Joint confidence intervals



# Generalised least squares

So far, we have made the assumption that the errors $\boldsymbol{\varepsilon}$ have mean $\mathbf{0}$ and variance $\sigma^2 I$, and sometimes that they are normally distributed. These assumptions do not always hold.

If the errors do not have $\mathbf{0}$ mean, then we should find another model!

It is not always satisfying to have normal errors, but they occur quite often in practice and the accompanying theory is very appealing.

What if the variance of $\boldsymbol{\varepsilon}$ is not $\sigma^2 I$?

# Generalised least squares

Suppose that $\varepsilon$ is multivariate normal but with a positive definite variance $V$. The maximum likelihood estimator now minimises

$$\mathbf{e}^T V^{-1}\mathbf{e} = (\mathbf{y} - X\mathbf{b})^T V^{-1}(\mathbf{y} - X\mathbf{b})$$

and thus satisfies the (equivalent of) the normal equations

$$X^T V^{-1} X\mathbf{b} = X^T V^{-1}\mathbf{y}.$$

This gives the *generalised least squares estimators*

$$\mathbf{b} = (X^T V^{-1} X)^{-1} X^T V^{-1}\mathbf{y}.$$

If $V = \sigma^2 I$, this reduces to ordinary least squares.

# Generalised least squares

We have

$$\begin{aligned} E[\mathbf{b}] &= \beta, \\ \text{Var } \mathbf{b} &= (X^T V^{-1} X)^{-1}. \end{aligned}$$

Moreover, it can be shown that the Gauss-Markov theorem still holds, i.e. the generalised least squares estimator is BLUE.

The proof is left as an exercise.

# Weighted least squares

In this situation, the errors are uncorrelated but do not have a common variance:

$$\text{Var } \boldsymbol{\varepsilon} = diag(\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2).$$

To estimate the parameters with ML, we minimise

$$(\mathbf{y} - X\mathbf{b})^T V^{-1} (\mathbf{y} - X\mathbf{b}) = \sum_{i=1}^{n} \left( \frac{e_i}{\sigma_i} \right)^2.$$

That is, we *weight* each residual by the inverse of the corresponding standard deviation. So a point with high variance influences $\mathbf{b}$ less than a point with low variance.

# Nonlinearities

All the models that we study are *linear* models, in the sense that they are linear w.r.t. the design variables. However, this does not mean that they can only model linear relationships. There is still some scope to model nonlinear relationships.

This is particularly true when you know, or have a good idea of what the type of relationship might be.

One way we can handle this is to include extra predictors which are nonlinear functions of the original predictors.

# Nonlinearities

For example, suppose lung capacity ($y$) was predicted by asking participants to blow a single breath into a balloon and measuring the diameter of the balloon ($x$).

Perhaps we could use a linear model for this, of the form

$$y = \beta_0 + \beta_1 x + \varepsilon.$$

# Nonlinearities

However, the diameter of the balloon is not a direct measure of lung capacity, and importantly it is not linearly related to lung capacity.

In fact, lung capacity is more likely to be related linearly to the *volume* of the balloon. The volume is much harder to measure, but is proportional to the cube of the diameter.

Therefore we might instead try a model like

$$y = \beta_0 + \beta_1 x^3 + \varepsilon.$$

# Nonlinearities

The analysis actually does not change at all: we have simply changed one design variable for another.

Another alternative might be to model the response on a polynomial that goes up to a cubic:

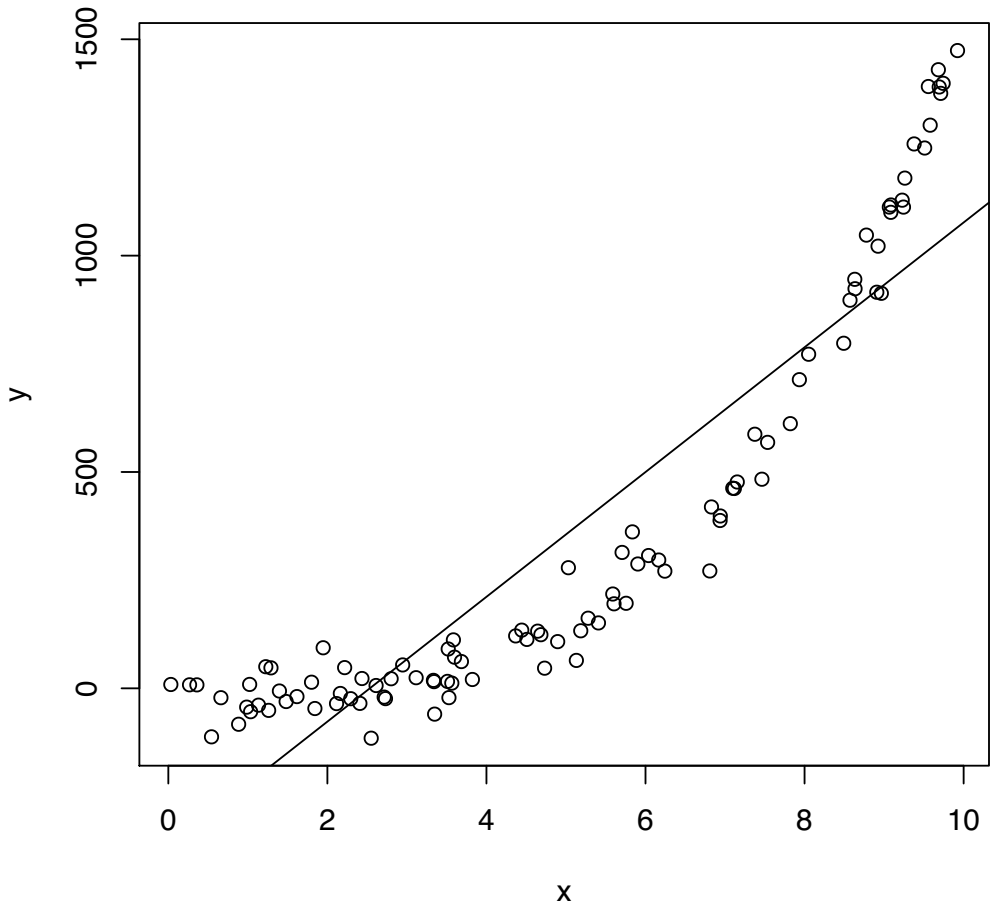
$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \varepsilon.$$

This introduces two extra design variables, but again the analysis is much the same.

# Example

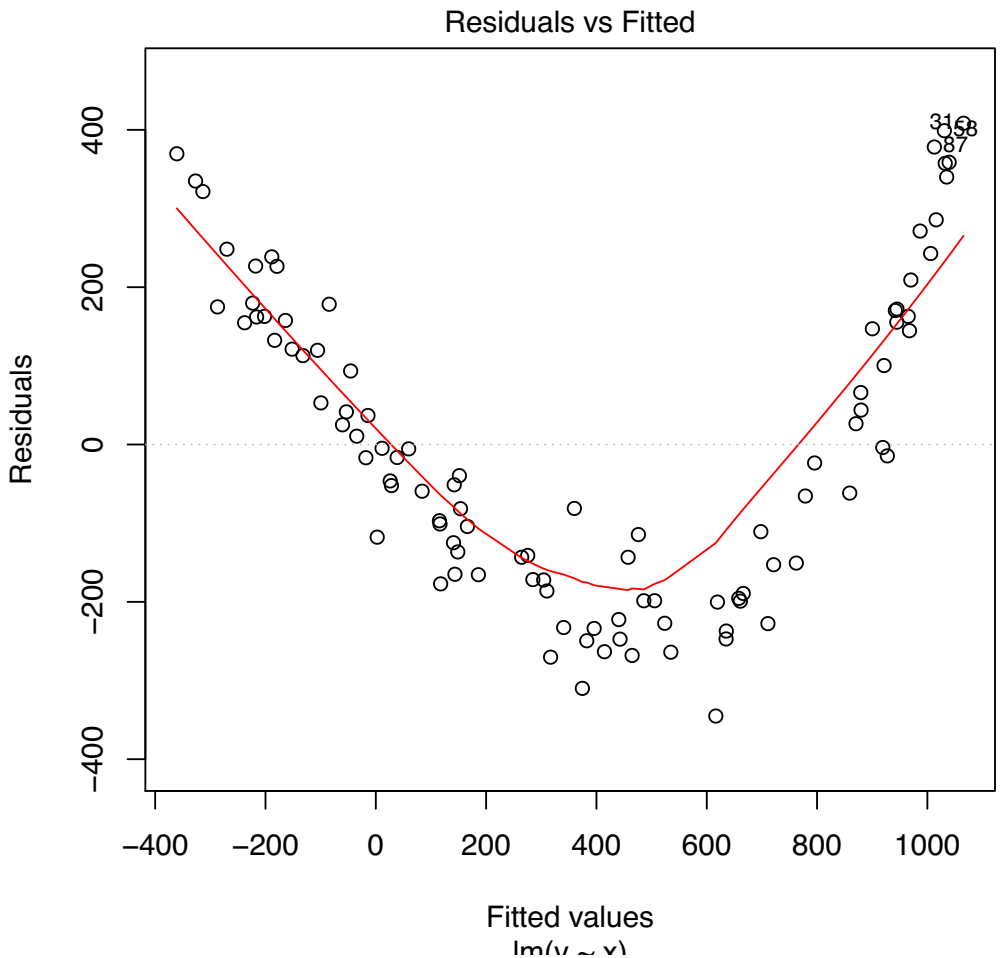
```
> x <- runif(100,0,10)
> y <- 2*x^3 - 5*x^2 + 1 + rnorm(100,0,50)
> model <- lm(y~x)
> summary(model)

Call:
lm(formula = y ~ x)

Residuals:
   Min     1Q Median     3Q    Max
-345.1 -167.0  -20.1  158.8  408.4

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -365.269     38.926  -9.384 2.64e-15 ***
x            144.187      6.584  21.898  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 196.2 on 98 degrees of freedom
```

# Example



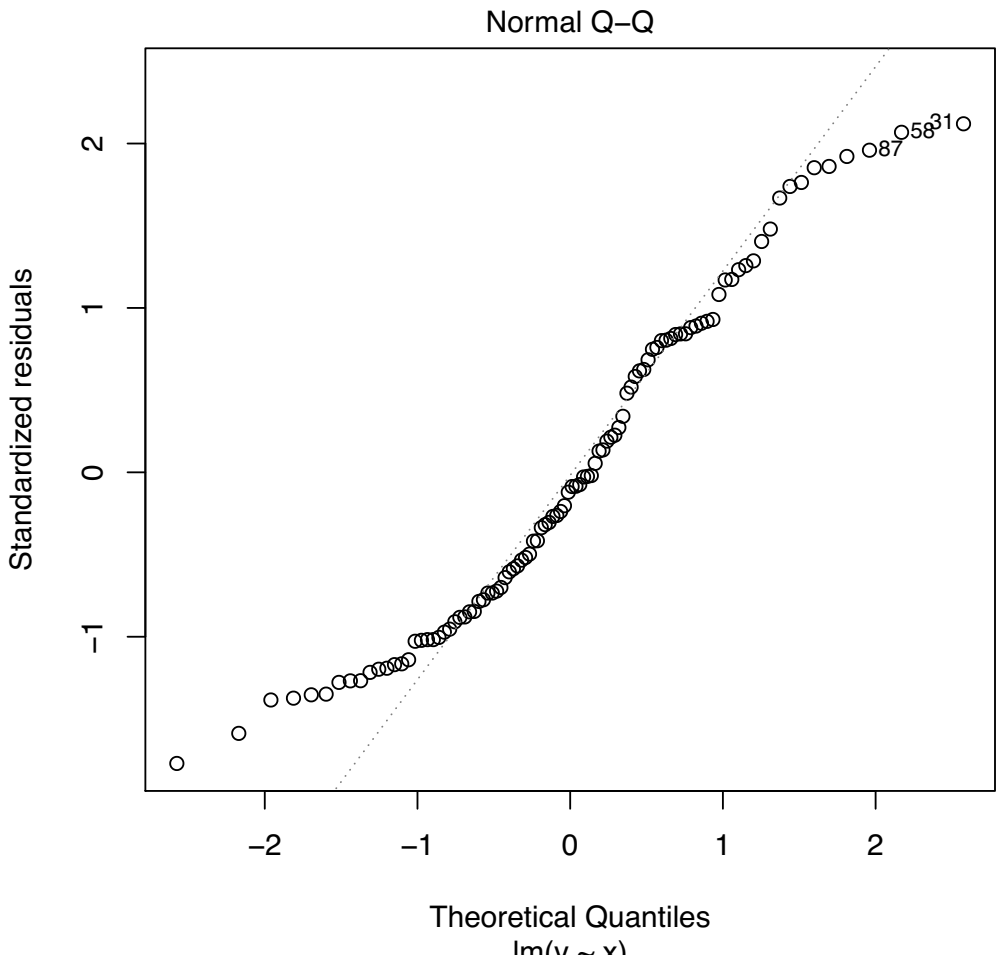
# Example

```
> plot(model,which=1)
```



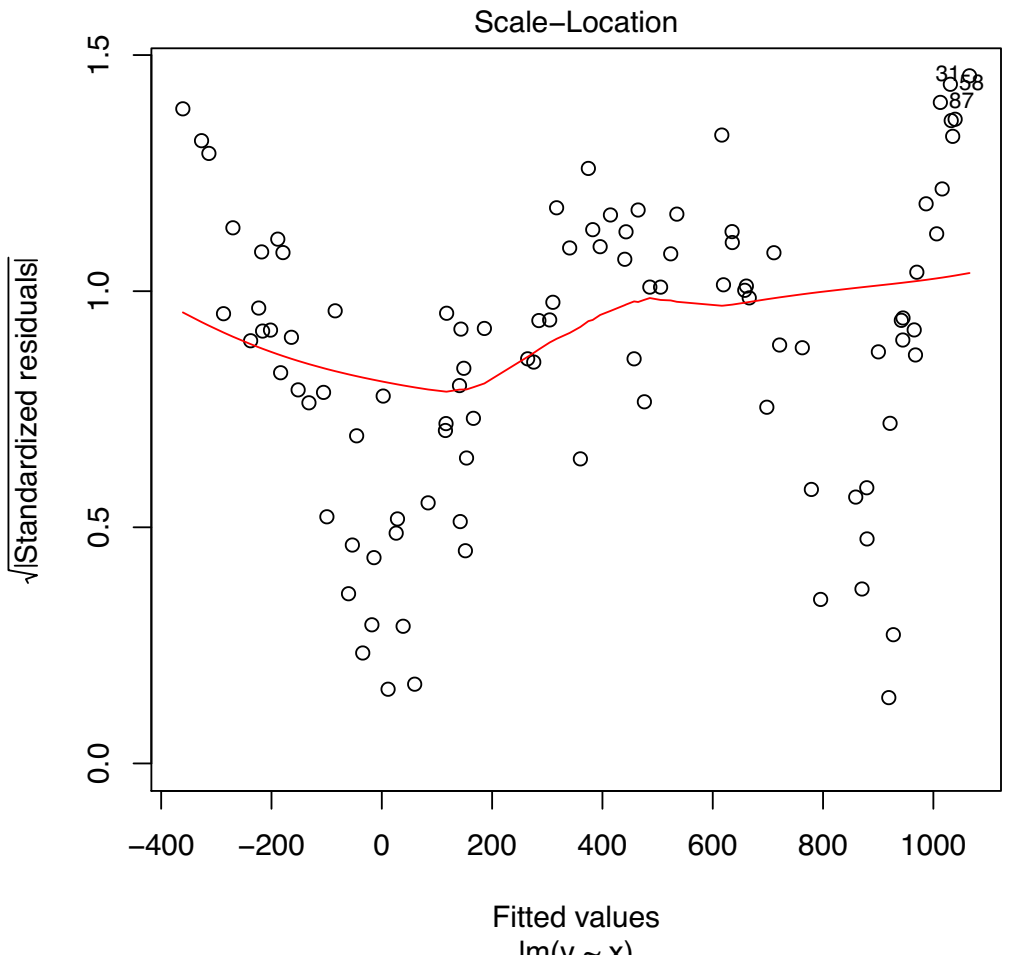
# Example

```
> plot(model,which=2)
```



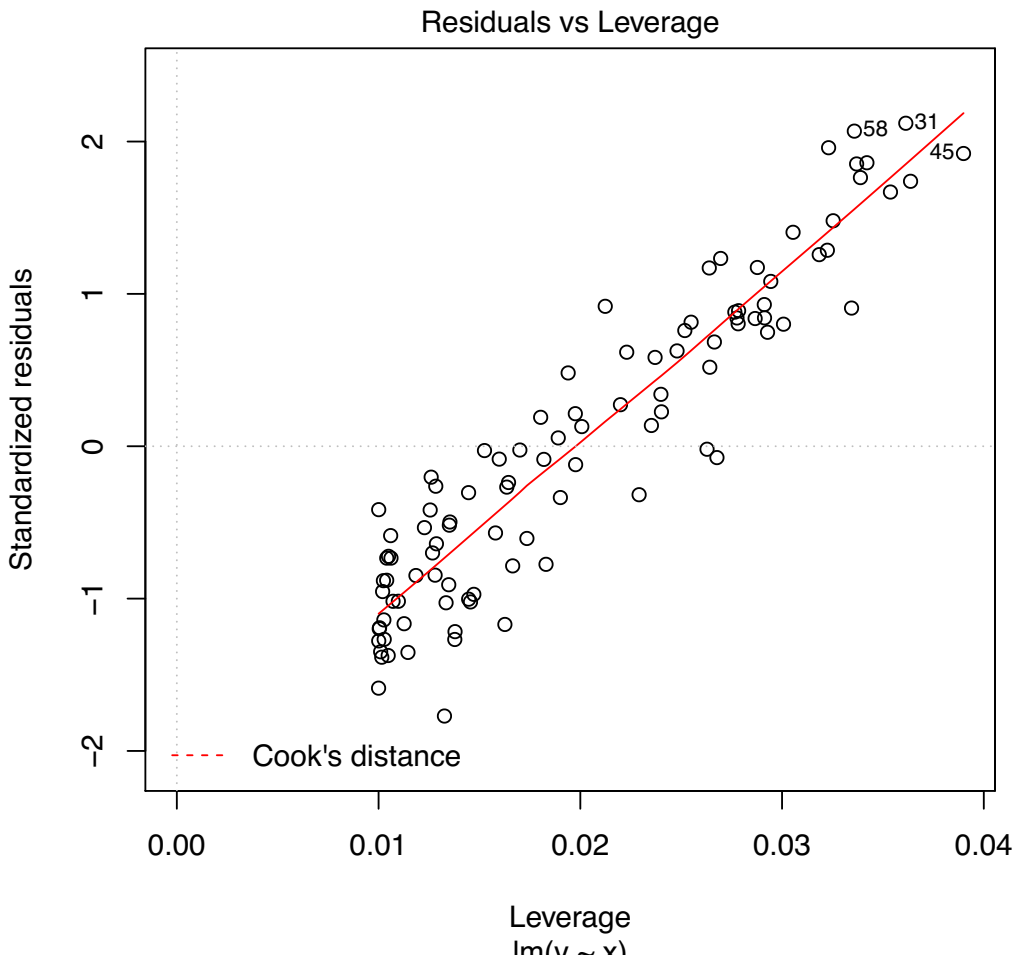
# Example

```
> plot(model,which=3)
```



# Example

```
> plot(model,which=5)
```

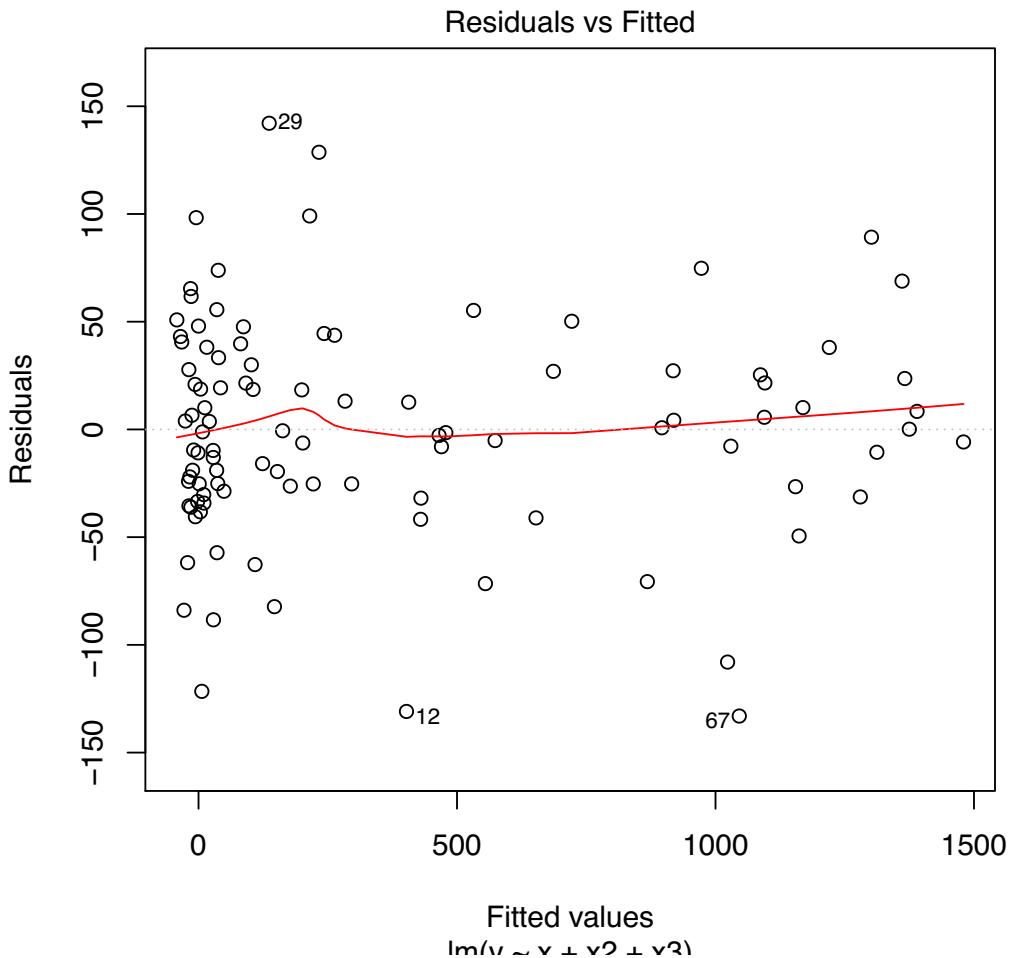


# Example

```
> data <- data.frame(y=y,x=x,x2=x^2,x3=x^3)
> model2 <- lm(y~x+x2+x3, data=data)
> summary(model2)

Call:
lm(formula = y ~ x + x2 + x3, data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-133.023  -27.099   -0.875   28.325  142.132

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -42.9772    23.2613  -1.848   0.0677 .
x            32.8909    19.4409   1.692   0.0939 .
x2          -11.3513     4.4307  -2.562   0.0120 *
x3            2.3682     0.2867   8.260 7.97e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

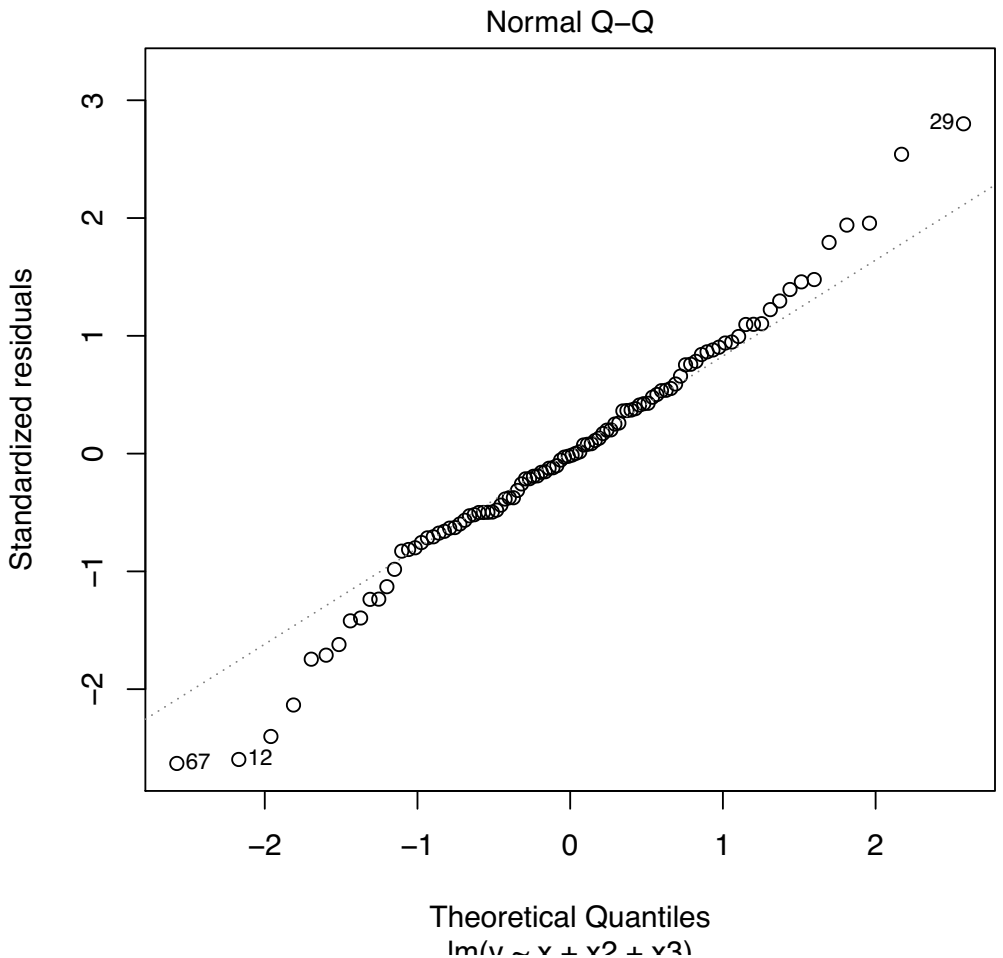
# Example

```
> plot(model2,which=1)
```



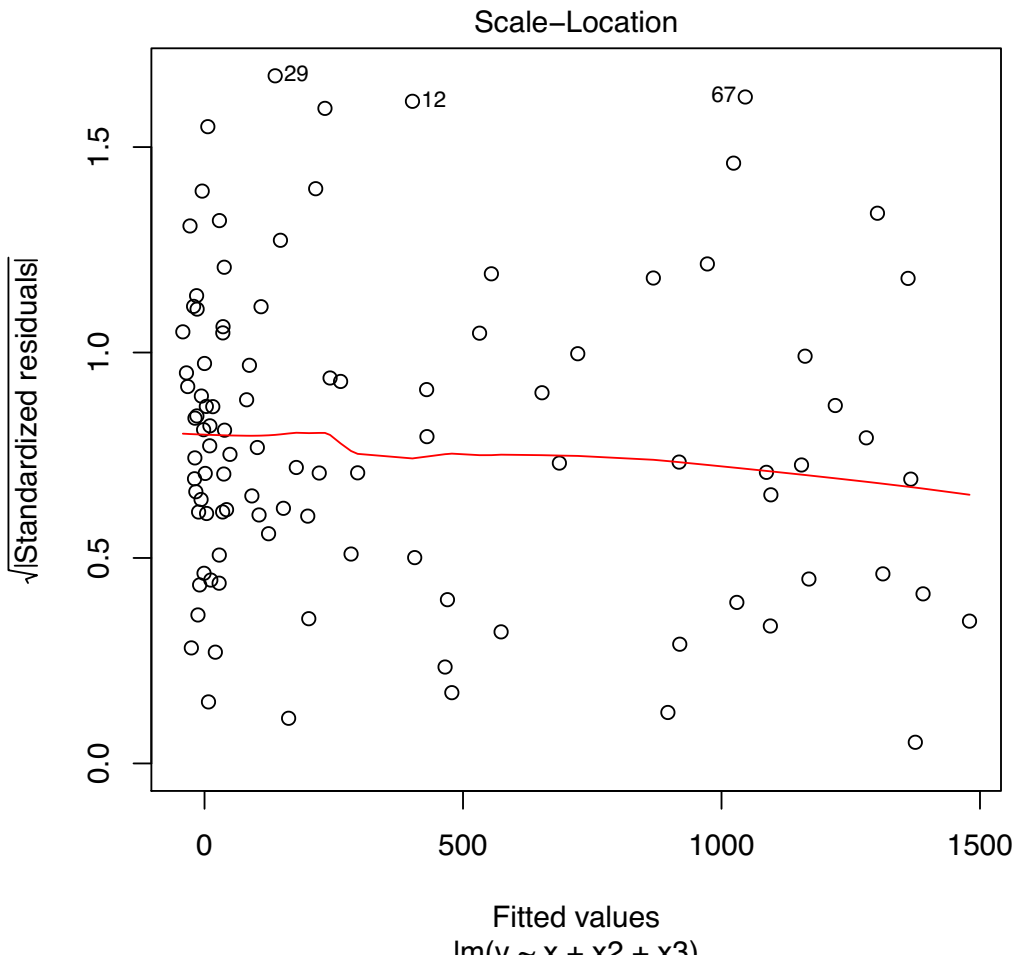
# Example

```
> plot(model2,which=2)
```



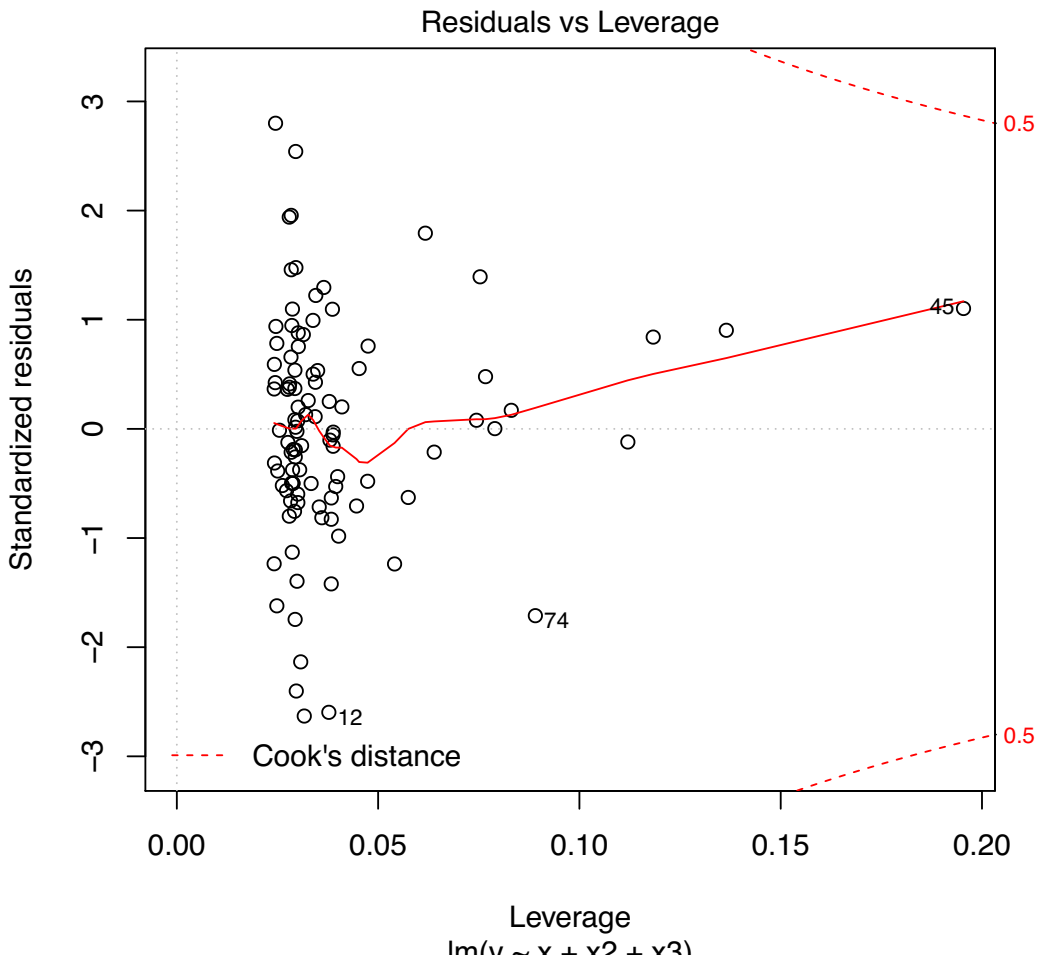
# Example

```
> plot(model2,which=3)
```



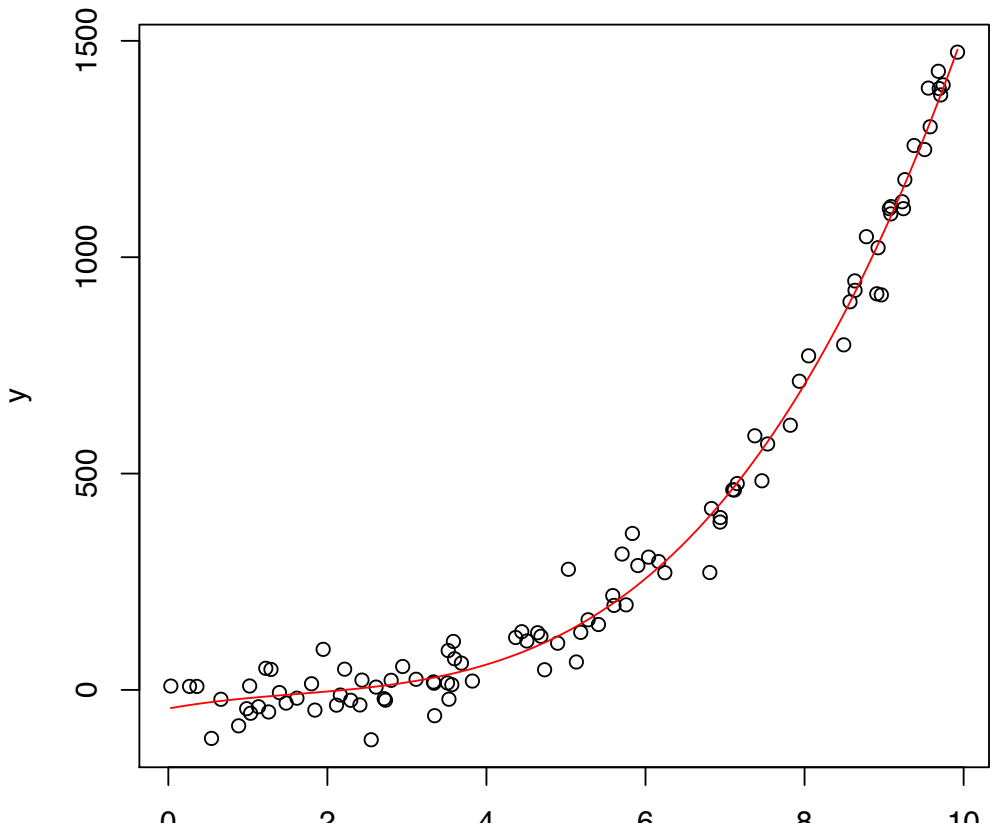
# Example

```
> plot(model2,which=5)
```



# Example

```
> plot(x,y)
> curve(model2$coef[1]+model2$coef[2]*x+model2$coef[3]*x^2
+           +model2$coef[4]*x^3,col='red',add=T)
```



# Nonlinearities

We can only do this because we understand the source of the data, and thus have a good idea about what kinds of potential relationships might occur.

If we observe an obviously non-linear relationship but have no idea about what the relationship might be, the situation is more difficult.

The best thing to do is to try and deduce the relationship from the data and then fit an appropriate model.

# Transformations

Certain kinds of relationships (in particular multiplicative relationships) also require the transformation of the *response* variable.

We have to be careful with this because a transformation of the response also transforms the error, and the form of the error.

Sometimes this can work in our favour, if the error needs to be transformed in order to fit with the assumptions of a linear model.

# Transformations

For example, if the true underlying model is

$$y = \alpha_1 e^{\alpha_2 x} \varepsilon,$$

then we would transform the response variable to $\ln y$:

$$\ln y = \ln \alpha_1 + \alpha_2 x + \ln \varepsilon.$$

We can then fit a linear model to $\ln y$ with design variable $x$ and recover the original coefficients with

$$\alpha_1 = e^{\beta_0}, \quad \alpha_2 = \beta_1.$$

# Transformations

On the other hand, if the true underlying model is

$$y = \beta_0 e^{\beta_1 x} + \varepsilon,$$

we can't do this.

We could estimate $\beta_1$ in some way (possibly by transforming and fitting as above), but ultimately we would fix it to a value.

Then we would fit a linear model to $y$ with the design variable $e^{\beta_1 x}$ and no intercept. This model will give us $\beta_0$.

# Transformations

Sometimes we have a good idea at the form of the true underlying model, because we understand the origin of the data.

However, most of the time we do not know the true underlying model and therefore cannot be sure what the correct transformation is.

In this case we usually try out a few reasonable-looking transformations and evaluate them in turn, using diagnostic plots.

# Transformations

There are certain signs which may indicate that a transformation is required:

- All the values are positive;
- The distribution of the data is skewed;
- There is an obvious non-linear relationship with another variable;
- The variances show a relationship with one of the variables.

# Transformations

Logarithmic transformations are very common because they convert multiplicative effects into additive ones. Useful transformations are:

| | |
|---|---|
| $\ln y,\ x$ | exponential |
| $\ln y,\ \ln x$ | power law |
| $\sqrt{y}$ | areas, or occurences inside areas |
| $\sqrt[3]{y}$ | volumes |
| $\frac{1}{y}$ | rates |
| $\ln \frac{y}{1-y}$ | proportions |

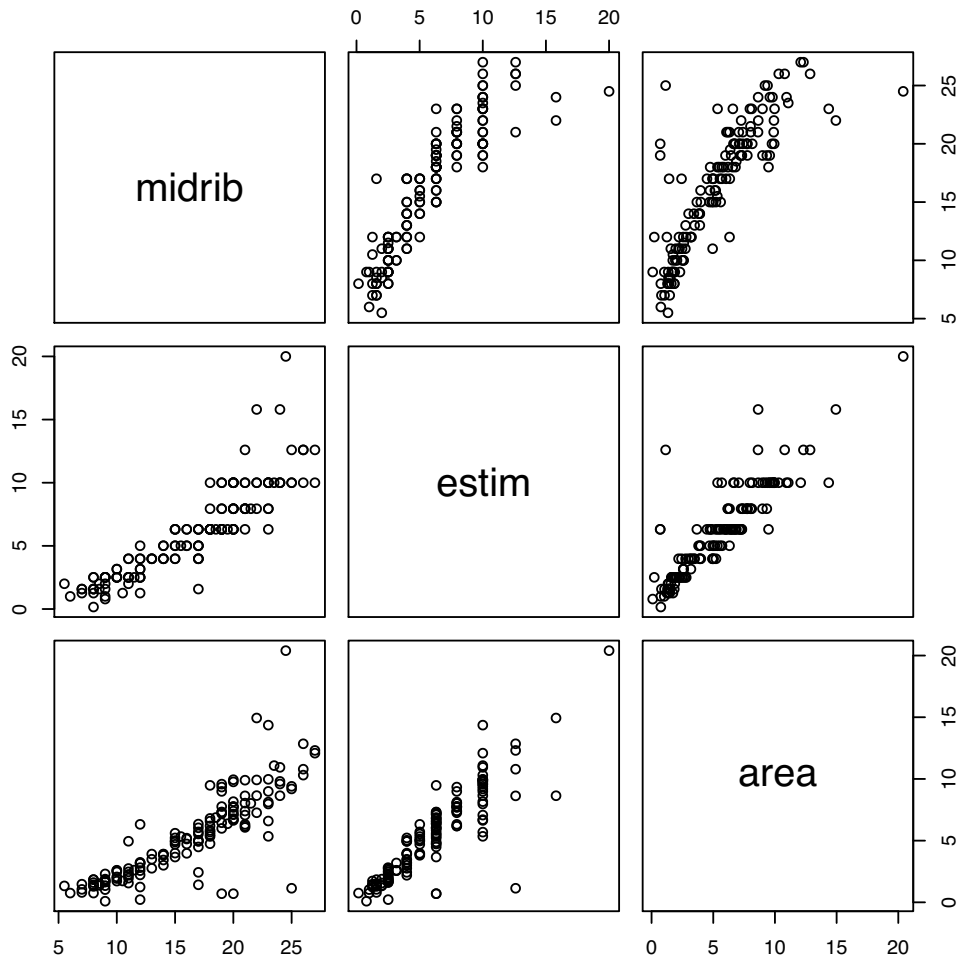
# Clover example

Recall that we first transformed the clover data by taking logarithms. Let us go through that decision process.

Firstly, we 'eyeball' the data.

```
> expclover <- read.csv("../data/clover.csv")
> pairs(expclover)
```

# Clover example



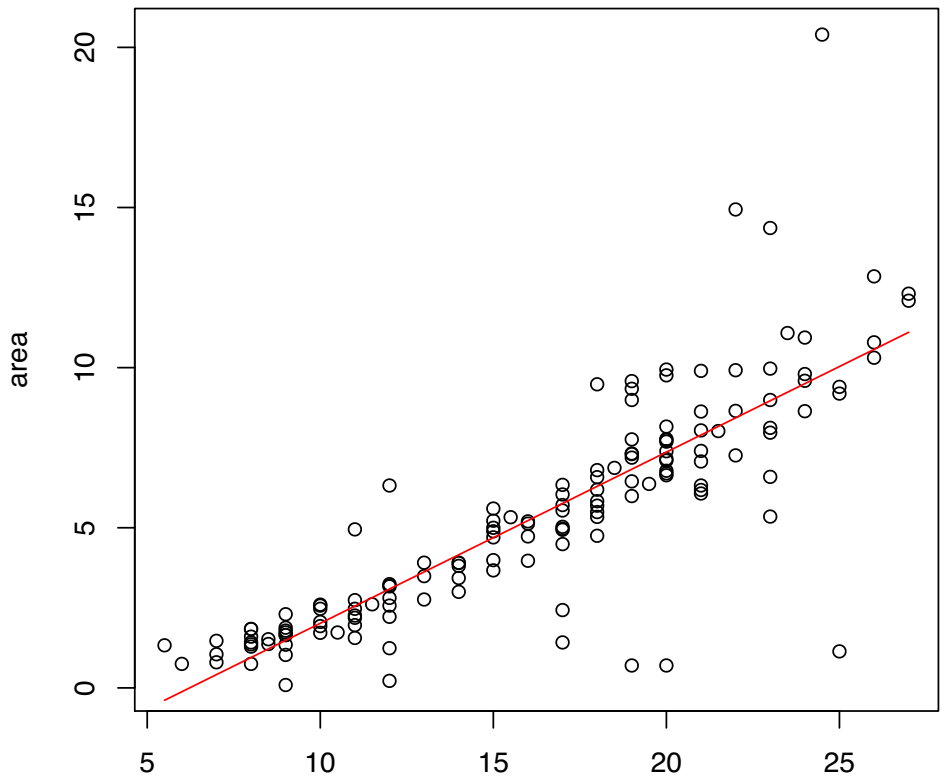
# Clover example

It is clear that there are some non-linearities which necessitate action before fitting a linear model.

Let us look closer at just the `area` to `midrib` relationship.

# Clover example

```
> plot(area ~ midrib, data=expclover)
> m <- lm(area ~ midrib, data=expclover)
> curve(m$coeff[1]+m$coeff[2]*x,add=T,col="red")
```



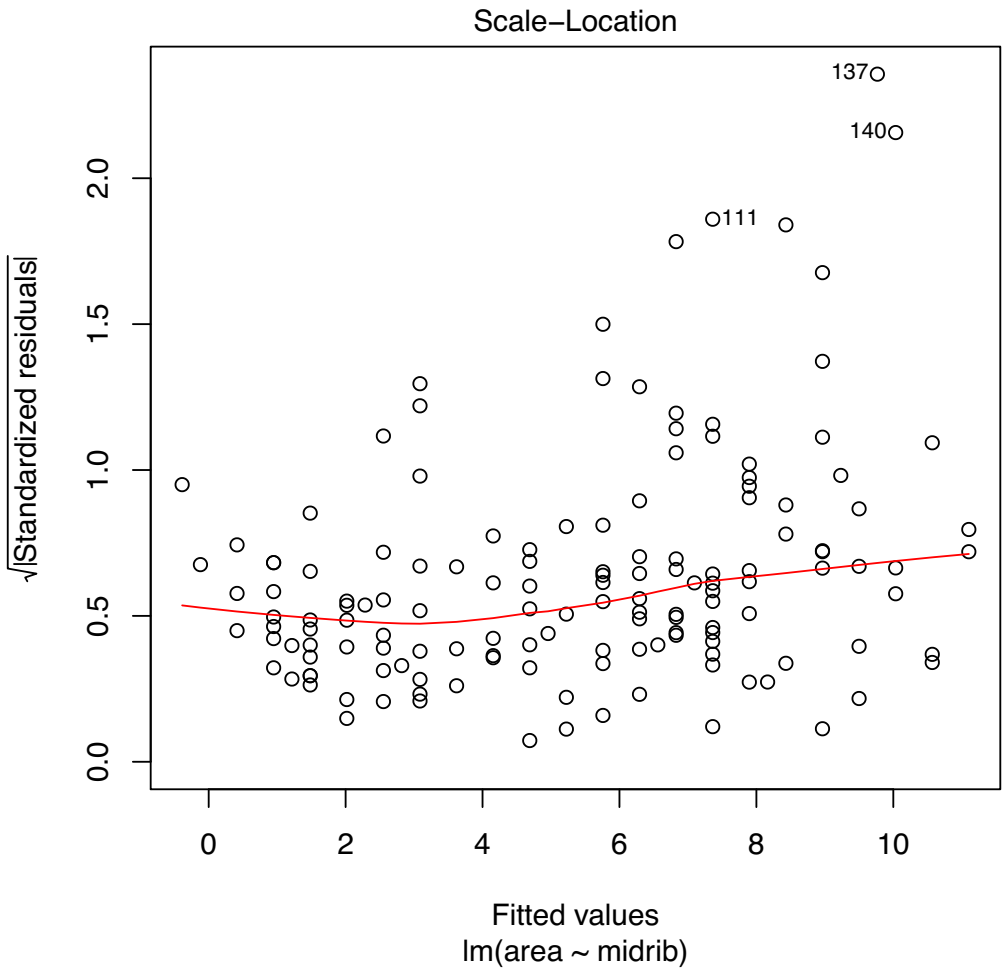
# Clover example

One thing which is very noticeable in the plot of `area` versus `midrib` is that the magnitude of the errors increase with both the variables.

This indicates a multiplicative error, which we can check with a diagnostic plot.

```
> plot(m, which=3)
```

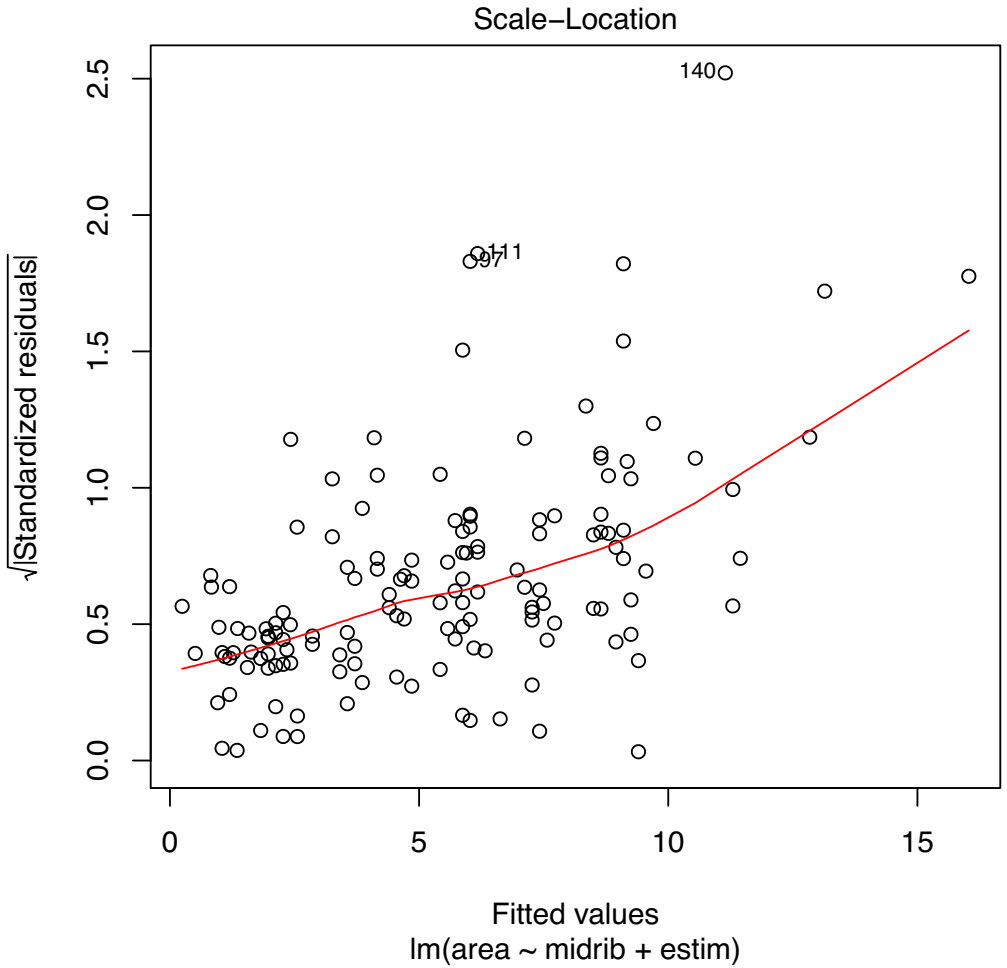
# Clover example




There is some evidence of increase, but not much in the trendline.

# Clover example

It becomes really obvious if we include both `midrib` and `estim`:



# Clover example

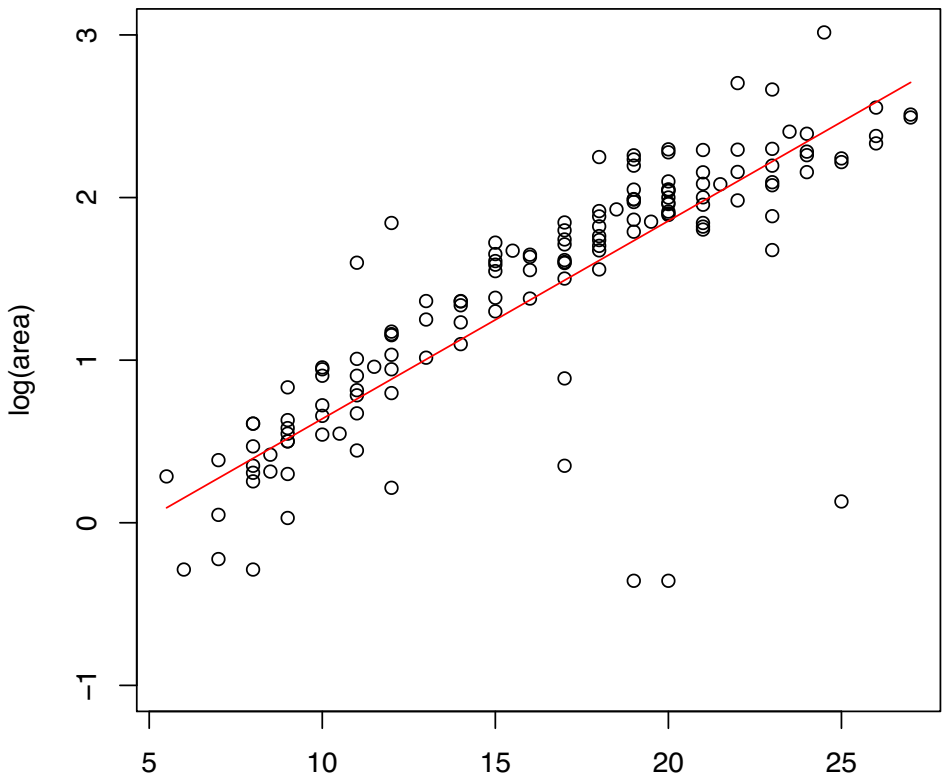
Now what sort of relationship could happen here?

Looking at the non-linear trend and multiplicative errors in the data, it would seem that the most likely kinds are power law or exponential relationships.

Let us try both types of transformations and see which one fits better.

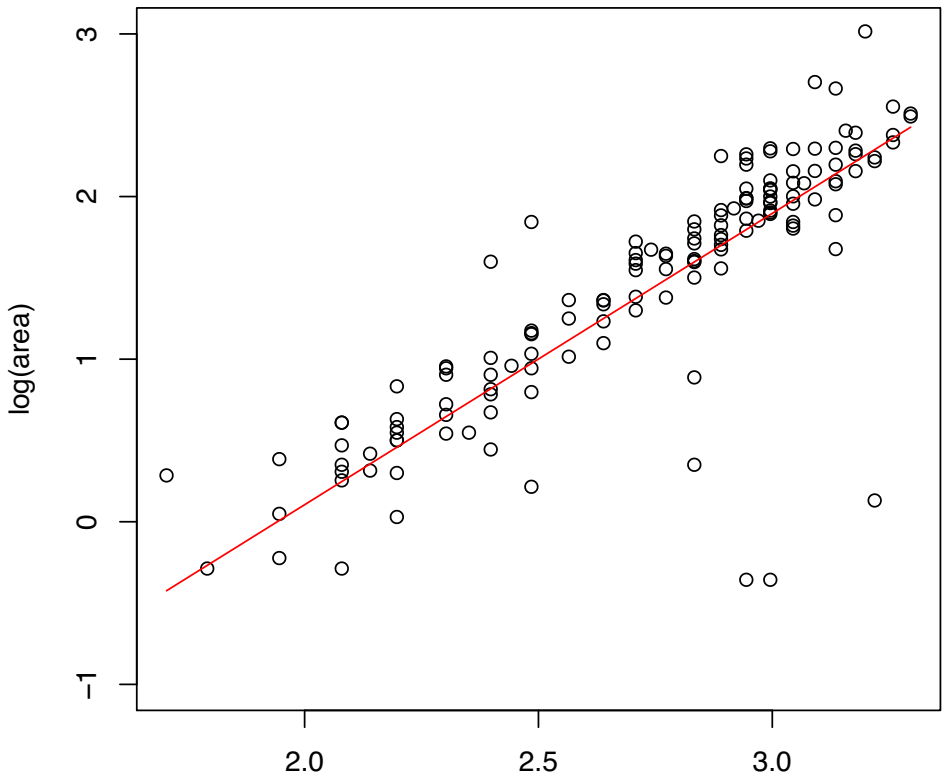
# Clover example

```
> plot(log(area) ~ midrib, data=expclover, ylim=c(-1,3))
> m <- lm(log(area) ~ midrib, data=expclover)
> curve(m$coeff[1]+m$coeff[2]*x,add=TRUE,col="red")
```



# Clover example

```
> plot(log(area) ~ log(midrib), data=expclover, ylim=c(-1,3))
> m <- lm(log(area) ~ log(midrib), data=expclover)
> curve(m$coeff[1]+m$coeff[2]*x,add=TRUE,col="red")
```



# Clover example

It is obvious that the model

$$\ln \texttt{area} = \beta_0 + \beta_1 \ln \texttt{midrib} + \varepsilon$$

works the best.

Similar reasoning can also be applied to the relationship between `area` and `estim` to deduce a power law.

It turns out that there are also botanical models which predict a power law, so there are good physical reasons to use it too.

# Clover example

Using our best fit from before:

```
> goodclover <- log(expclover[-c(6, 23, 47, 97, 111, 140), ])
> model2 <- lm(area ~ midrib + estim, data = goodclover)
> model2$coefficients
```

```
(Intercept)      midrib       estim
 -1.3814775   0.6503733   0.6919902
```

Our fitted model is

$$\ln \texttt{area} = -1.38 + 0.65 \ln \texttt{midrib} + 0.69 \ln \texttt{estim} + \varepsilon.$$

Converting back into the original measurements, we fit the model

$$\texttt{area} = e^{-1.38} \times \texttt{midrib}^{0.65} \times \texttt{estim}^{0.69} \times \varepsilon'.$$