# MAST90104 - Lecture 4 Part I

Weichang Yu

Room 108, Old Geology South Bldg
School of Mathematics and Statistics, University of Melbourne

# Linear Models in Matrix Representation

We remind ourselves what a linear model is:

- We have $n$ subjects, labelled 1 to $n$;
- Random response variable $(y)$ denoted $y_1, y_2, \ldots, y_n$;
- Fixed predictors $\mathbf{x}_1, \ldots, \mathbf{x}_k$ for subject $i$ denoted $x_{i1}, x_{i2}, \ldots, x_{ik}$.

# Linear Models in Matrix Representation

The linear model is

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

for all $i = 1, 2, \ldots, n$, where $n > k + 1$, or

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1k} \\ 1 & x_{21} & x_{22} & \ldots & x_{2k} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \ldots & x_{nk} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

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

What assumptions do we make?

# Gauss-Markov assumptions (for non-random $\mathbf{X}$)

Let's be clear on the assumptions:

**Assumption (I)**: The true relationship between $\mathbf{X}$ and $\mathbf{y}$ is $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, where $\mathbf{X}$ is a $n$ by $k+1$ matrix and $\boldsymbol{\beta}$ is a $(k+1)$-dimensional vector.

**Assumption (II)**: $\mathbf{X}$ is a full rank matrix, i.e. $r(\mathbf{X}) = k+1$.

**Assumption (III)**: The random errors are zero-centered, i.e., $\mathbb{E}(\boldsymbol{\epsilon}) = \mathbf{0}$ and hence $\mathbb{E}(\mathbf{y}) = \mathbf{X}\boldsymbol{\beta}$.

**Assumption (IV)**: The random errors are uncorrelated, and have homogeneous variance, i.e.,

$$Var(\boldsymbol{\epsilon}) = \sigma^2\mathbf{I} = \begin{pmatrix} \sigma^2 & 0 & \dots & 0 \\ 0 & \sigma^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \sigma^2 \end{pmatrix}.$$

# Gauss-Markov assumptions (for non-random $\mathbf{X}$)

Assumption (IV) does not imply that $\epsilon$ is MVN-distributed.

Assumption (IV) implies that $Cov(\epsilon_i, \epsilon_{i'}) = 0$ for any $i \neq i'$. But it does not imply independence!

# Least squares estimator

Least squares criterion:

$$C(\beta) = \|\mathbf{y} - \mathbf{X}\beta\|^2.$$

## Theorem 4.1

*Under assumption (II), $C(\beta)$ is uniquely minimised by*

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

.

**Remark.** Outline of proof

(1) We use matrix calculus to derive the form of $\widehat{\beta}$. Derivation is found in lecture 2, but we repeat it for completeness.

(2) We show that any $\mathbf{b} \in \mathbb{R}^{k+1}$ yields a $C(\mathbf{b})$ that is at least as large as $C(\widehat{\beta})$.

(3) We argue that $\widehat{\beta}$ is unique.

# Least squares estimator

Part (1) of proof:

Let $C(\boldsymbol{\beta}) = \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2$. Then,

$$\begin{aligned} C(\boldsymbol{\beta}) &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 \\ &= (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \\ &= \mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\beta}^T\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} \end{aligned}$$

Then,

$$\begin{aligned} \frac{\partial}{\partial\boldsymbol{\beta}} C(\boldsymbol{\beta}) &= \tfrac{\partial\mathbf{y}^T\mathbf{y}}{\partial\boldsymbol{\beta}} - 2\tfrac{\partial\mathbf{y}^T\mathbf{X}\boldsymbol{\beta}}{\partial\boldsymbol{\beta}} + \tfrac{\partial\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{X}\boldsymbol{\beta}}{\partial\boldsymbol{\beta}} \\ &= -2\mathbf{y}^T\mathbf{X} + 2\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{X} \end{aligned}$$

Then, setting LHS equals to $\mathbf{0}$, $\boldsymbol{\beta} = \widehat{\boldsymbol{\beta}}$, we have the *normal equations* $\widehat{\boldsymbol{\beta}}^T\mathbf{X}^T\mathbf{X} = \mathbf{y}^T\mathbf{X}$. Hence $\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$.

# Least squares estimator

Part (2) of proof:

Consider the residual vector $\mathbf{e} = \mathbf{y} - \mathbf{X}\widehat{\beta}$ and vector $\mathbf{Xb}$, where $\mathbf{b} \in \mathbb{R}^{k+1}$ (this means we consider a $k+1$ dimensional column vector).

Check that $\mathbf{e}$ and $\mathbf{Xb}$ are orthogonal for any $\mathbf{b} \in \mathbb{R}^{k+1}$:

$$\begin{aligned}(\mathbf{Xb})^T\mathbf{e} &= \mathbf{b}^T(\mathbf{X}^T\mathbf{y} - \mathbf{X}^T\mathbf{X}\widehat{\beta}) \\ &= \mathbf{b}^T(\mathbf{X}^T\mathbf{y} - \underbrace{\mathbf{X}^T\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}}_{=\mathbf{I}}\mathbf{X}^T\mathbf{y}) = \mathbf{0}.\end{aligned}$$

# Least squares estimator

Part (2) of proof:

Now consider any $\mathbf{b} \in \mathbb{R}^{k+1}$, then

$$\begin{aligned}
C(\mathbf{b}) &= (\mathbf{y} - \mathbf{X}\mathbf{b})^T(\mathbf{y} - \mathbf{X}\mathbf{b}) \\
&= (\mathbf{y} - \mathbf{X}\widehat{\beta} + \mathbf{X}\widehat{\beta} - \mathbf{X}\mathbf{b})^T(\mathbf{y} - \mathbf{X}\widehat{\beta} + \mathbf{X}\widehat{\beta} - \mathbf{X}\mathbf{b}) \\
&= (\mathbf{y} - \mathbf{X}\widehat{\beta})^T(\mathbf{y} - \mathbf{X}\widehat{\beta}) + (\widehat{\beta} - \mathbf{b})^T\mathbf{X}^T\mathbf{X}(\widehat{\beta} - \mathbf{b}) \\
&+ 2\underbrace{(\mathbf{X}(\widehat{\beta} - \mathbf{b}))^T(\mathbf{y} - \mathbf{X}\widehat{\beta})}_{=\mathbf{0}} \\
&= C(\widehat{\beta}) + \|\mathbf{X}(\widehat{\beta} - \mathbf{b})\|^2 \\
&\geq C(\widehat{\beta})
\end{aligned}$$

# Least squares estimator

Part (3) of proof:

Let $\mathbf{x}_j$ denote the j-th column of $\mathbf{X}$. Recall that for any $\mathbf{u} = (u_1, u_2, \ldots, u_{k+1})^T$, we may write

$$\mathbf{X}\mathbf{u} = u_1\mathbf{x}_1 + \ldots u_{k+1}\mathbf{x}_{k+1}$$

Since $\mathbf{X}$ is full rank, the only solution satisfying

$$u_1\mathbf{x}_1 + \ldots u_{k+1}\mathbf{x}_{k+1} = \mathbf{0}$$

is $\mathbf{u} = \mathbf{0}$.

# Least squares estimator

Part (3) of proof:

Now, for any **b** such that:

$$\begin{aligned} C(\mathbf{b}) &= C(\widehat{\boldsymbol{\beta}}) \\ \mathbf{X}(\widehat{\boldsymbol{\beta}} - \mathbf{b}) &= \mathbf{0} \\ \widehat{\boldsymbol{\beta}} - \mathbf{b} &= \mathbf{0} \\ \mathbf{b} &= \widehat{\boldsymbol{\beta}} \end{aligned}$$

# Example 1: Simple linear regression

**Example.**
Consider the simple linear regression model:

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

Use theorem 4.1 to verify that the least squares estimator of $\beta_0$ and $\beta_1$ are:

$$\widehat{\beta}_0 = \overline{y} - \widehat{\beta}_1 \overline{x}, \quad \widehat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{n}(x_i - \overline{x})^2}.$$

# Example 1: Simple linear regression

**Solution:** Note that simple linear regression can be cast in the framework of a linear model, where the response variable $y$ depends on only one variable $x$:

$$y_i = \beta_0 + \beta_1 x_i + \boldsymbol{\epsilon}_i.$$

For $n$ responses, this gives the linear equations

$$\begin{array}{rcl} y_1 & = & \beta_0 + \beta_1 x_1 + \epsilon_1 \\ y_2 & = & \beta_0 + \beta_1 x_2 + \epsilon_2 \\ & \vdots & \\ y_n & = & \beta_0 + \beta_1 x_n + \epsilon_n \end{array}$$

# Example 1: Simple linear regression

In the matrix formulation, we have

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \\ 1 & x_n \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}, \quad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}.$$

**X** has full rank if the $x_i$ are not all the same.

# Example 1: Simple linear regression

We have

$$(\mathbf{X}^T\mathbf{X})^{-1} = \frac{1}{n\sum_i x_i^2 - \left(\sum_i x_i\right)^2}\left[\begin{array}{cr} \sum_i x_i^2 & -\sum_i x_i \\ -\sum_i x_i & n \end{array}\right].$$

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

$$\begin{aligned}
\widehat{\boldsymbol{\beta}} &= (\mathbf{X}^T\mathbf{X})^{-1}X^T\mathbf{y} \\
&= \frac{1}{n\sum_i x_i^2 - \left(\sum_i x_i\right)^2}\left[\begin{array}{cr} \sum_i x_i^2 & -\sum_i x_i \\ -\sum_i x_i & n \end{array}\right]\left[\begin{array}{c} \sum_i y_i \\ \sum_i x_i y_i \end{array}\right] \\
&= \frac{1}{n\sum_i x_i^2 - \left(\sum_i x_i\right)^2}\left[\begin{array}{c} \sum_i x_i^2\sum_i y_i - \sum_i x_i\sum_i x_i y_i \\ n\sum_i x_i y_i - \sum_i x_i\sum_i y_i \end{array}\right].
\end{aligned}$$

# Example 1: Simple linear regression

The estimator for the slope of the regression line is

$$\widehat{\beta}_1 = \frac{n \sum_i x_i y_i - \sum_i x_i \sum_i y_i}{n \sum_i x_i^2 - \left(\sum_i x_i\right)^2}$$

which may be a familiar formula for us.

We can rewrite the numerator on the RHS as

$$\begin{aligned}
n \sum_i x_i y_i - \sum_i x_i \sum_i y_i &= n \sum_i (x_i - \overline{x} + \overline{x})(y_i - \overline{y} + \overline{y}) - \sum_i x_i \sum_i y_i \\
&= n \sum_i (x_i - \overline{x})(y_i - \overline{y}) + n\overline{y} \sum_i (x_i - \overline{x}) \\
&+ n\overline{x} \sum_i (y_i - \overline{y}) + n^2 \overline{x}\overline{y} - \sum_i x_i \sum_i y_i \\
&= n \sum_i (x_i - \overline{x})(y_i - \overline{y}) + n\overline{y} \sum_i (x_i - \overline{x})
\end{aligned}$$

# Example 1: Simple linear regression

Similarly, we can write the denominator as (detailed working as an exercise for you)

$$n\sum_i x_i^2 - \left(\sum_i x_i\right)^2 = n\sum_i (x_i - \overline{x})^2$$

Hence, the slope estimator can be re-expressed as

$$\widehat{\beta}_1 = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sum_i (x_i - \overline{x})^2}$$

# Example 1: Simple linear regression

The estimator for the intercept of the regression line is

$$\widehat{\beta}_0 = \frac{\sum_i x_i^2 \sum_i y_i - \sum_i x_i \sum_i x_i y_i}{n \sum_i x_i^2 - \left(\sum_i x_i\right)^2}$$

which may not look familiar.

$$\begin{aligned}
\frac{\sum_i x_i^2 \sum_i y_i - \sum_i x_i \sum_i x_i y_i}{n \sum_i x_i^2 - \left(\sum_i x_i\right)^2} &= \frac{\overline{y} \sum_i x_i^2 - \overline{x} \sum_i x_i y_i}{\sum_i (x_i - \overline{x})^2} \\
&= \frac{\overline{y} \sum_i (x_i - \overline{x} + \overline{x})^2}{\sum_i (x_i - \overline{x})^2} - \frac{\overline{x} \sum_i x_i y_i}{\sum_i (x_i - \overline{x})^2} \\
&= \overline{y} + \frac{n \overline{y} \overline{x}^2}{\sum_i (x_i - \overline{x})^2} - \frac{\overline{x} \sum_i x_i y_i}{\sum_i (x_i - \overline{x})^2} \\
&= \overline{y} - \overline{x} \frac{\sum_i x_i y_i - n \overline{x} \overline{y}}{\sum_i (x_i - \overline{x})^2} = \overline{y} - \widehat{\beta}_1 \overline{x}.
\end{aligned}$$

# Example 2: Housing prices

**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

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

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 |

# Example 2: Housing prices

The model generates the 5 linear equations

$$\begin{aligned}
50 &= \beta_0 + 1\beta_1 + 1\beta_2 + \epsilon_1 \\
40 &= \beta_0 + 5\beta_1 + 1\beta_2 + \epsilon_2 \\
52 &= \beta_0 + 5\beta_1 + 2\beta_2 + \epsilon_3 \\
47 &= \beta_0 + 10\beta_1 + 2\beta_2 + \epsilon_4 \\
65 &= \beta_0 + 20\beta_1 + 3\beta_2 + \epsilon_5
\end{aligned}$$

# Example 2: Housing prices

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

$$\mathbf{y} = \begin{bmatrix} 50 \\ 40 \\ 52 \\ 47 \\ 65 \end{bmatrix}, \quad X = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 5 & 1 \\ 1 & 5 & 2 \\ 1 & 10 & 2 \\ 1 & 20 & 3 \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}, \quad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \end{bmatrix}.$$

# Example 2: Housing prices

```
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)
```

# Example 2: Housing prices

Use R to compute $\widehat{\beta}$:

```
betahat <- solve(t(X)%*%X,t(X)%*%y)
```

Therefore our fitted model is

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

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

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

# Fitted values and residuals

The fitted values (also known as *predicted values*) are expressed as

$$\widehat{\mathbf{y}} = \mathbf{X}\widehat{\boldsymbol{\beta}}.$$

By assumptions (I)-(III),
$\mathbb{E}(\widehat{\mathbf{y}}) = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbb{E}(\mathbf{y}) = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} = \mathbf{X}\boldsymbol{\beta}$

The residuals are expressed as

$$\mathbf{e} = \mathbf{y} - \widehat{\mathbf{y}}$$

By assumptions (I)-(III), $\mathbb{E}(\mathbf{e}) = \mathbf{X}\boldsymbol{\beta} - \mathbb{E}(\widehat{\mathbf{y}}) = \mathbf{0}$. Note that $e_i \approx \epsilon_i$ when $n$ is large.

# Residuals orthogonal to the column space of $\mathbf{X}$

The general element of the column space is $\mathbf{Xa}$ where $\mathbf{a}$ is a $(k+1) \times 1$ vector. The elements of the column space of $\mathbf{X}$ and the residuals $\mathbf{e} = \mathbf{y} - \mathbf{X}\widehat{\beta}$ are *orthogonal* to each other.

This is because the residuals can be written as $(\mathbf{I} - \mathbf{H})\mathbf{y}$ where $\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ and

$$(\mathbf{Xa})^T(\mathbf{I} - \mathbf{H})\mathbf{y} = \mathbf{a}^T(\mathbf{X}^T(\mathbf{I} - \mathbf{H}))\mathbf{y}$$

and

$$\mathbf{X}^T(\mathbf{I} - \mathbf{H}) = \mathbf{X}^T - \mathbf{X}^T(\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T) = \mathbf{0}.$$

# Residuals orthogonal to the column space of $\mathbf{X}$

Since $\widehat{\mathbf{y}} = \mathbf{X}\widehat{\beta}$, it is in the column space of $\mathbf{X}$. Thus, $\widehat{\mathbf{y}}$ and $\mathbf{e}$ are orthogonal and hence *uncorrelated* with each other, i.e.,

$$\mathbb{E}\left[\{\widehat{\mathbf{y}} - \mathbb{E}(\widehat{\mathbf{y}})\}\{\mathbf{e} - \mathbb{E}(\mathbf{e})\}^T\right] = \mathbf{0}.$$

*Proof of the above statement is left as an exercise. You will need assumptions (I) - (IV).*

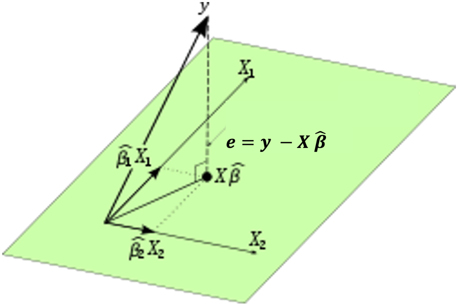


Figure: Geometric interpretation of OLS

# How good is the least squares estimator?

What makes an estimator "good"?

Two desirable properties for an estimator are that it is unbiased (on target) and of minimal variance.

## Theorem 4.2

*Under assumptions (I)-(IV), the least squares estimator* $\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ *is an unbiased estimator for* $\boldsymbol{\beta}$*. In other words,*

$$E[\widehat{\boldsymbol{\beta}}] = \boldsymbol{\beta}.$$

*Furthermore,*

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

# How good is the least squares estimator?

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

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

$$\begin{aligned}
\mathrm{V}ar(\widehat{\beta}) &= \mathrm{V}ar(\mathbf{X}^T\mathbf{X})^{-1}X^T\mathbf{y} \\
&= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\sigma^2\mathbf{I}\left((\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\right)^T \\
&= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}((\mathbf{X}^T\mathbf{X})^T)^{-1}\sigma^2 \\
&= \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}.
\end{aligned}$$

# Gauss-Markov Theorem

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

Now suppose we have a model with some parameters $\boldsymbol{\beta}$ and a linear estimator $\widehat{\boldsymbol{\beta}}$ for these parameters. If $\mathbb{E}(\widehat{\boldsymbol{\beta}}) = \boldsymbol{\beta}$, we say that $\widehat{\boldsymbol{\beta}}$ is an linear unbiased estimator (LUE) of $\boldsymbol{\beta}$.

## Definition 4.3

If $\widehat{\boldsymbol{\beta}} = \widetilde{\mathbf{L}}\mathbf{y}$ for some constant matrix $\widetilde{\mathbf{L}}$, $\mathbb{E}[\widehat{\boldsymbol{\beta}}] = \boldsymbol{\beta}$, and

$$\mathbf{P_L} = Var(\mathbf{Ly}) - Var(\widehat{\boldsymbol{\beta}})$$

is positive semi-definite for any $\mathbf{L}$ such that $\mathbb{E}(\mathbf{Ly}) = \boldsymbol{\beta}$, then $\widehat{\boldsymbol{\beta}}$ is called a *best linear unbiased estimator* of $\boldsymbol{\beta}$ (or BLUE).

# Gauss-Markov Theorem

## Theorem 4.4

*Under assumptions (I) - (IV), the least squares estimator $\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ is the unique BLUE for $\boldsymbol{\beta}$.*

**Proof.** In Theorem 3.2, we have shown that $\widehat{\boldsymbol{\beta}}$ is unbiased and $\mathrm{V}ar(\widehat{\boldsymbol{\beta}}) = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}$.

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

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

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

By Definition 3.3, we need to show that $\mathrm{V}ar(\mathbf{b}) - \mathrm{V}ar(\widehat{\boldsymbol{\beta}})$ is positive semi-definite.

# Gauss-Markov Theorem

Now,

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

Since **b** is an unbiased for all $\boldsymbol{\beta} \in \mathbb{R}^{k+1}$, we have

$$\mathbf{BX}\beta = \mathbf{0}$$

for all $\boldsymbol{\beta} \in \mathbb{R}^{k+1}$. Hence, $\mathbf{BX} = \mathbf{0}$.

# Gauss-Markov Theorem

Now,

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

# Gauss-Markov Theorem

Now, for all $\mathbf{u} \in \mathbb{R}^{k+1}$, we have

$$\begin{aligned}\mathbf{u}^T\{Var(\mathbf{b}) - Var(\widehat{\beta})\}\mathbf{u} &= \sigma^2\mathbf{u}^T\mathbf{B}\mathbf{B}^T\mathbf{u} \\ &= \sigma^2\|\mathbf{B}^T\mathbf{u}\|^2 \geq 0\end{aligned}$$

and hence we have shown that $Var(\mathbf{b}) - Var(\widehat{\beta})$ is positive semi-definite.

The uniqueness of $\widehat{\beta}$ as the BLUE follows by noting that $Var(\mathbf{b}) = Var(\widehat{\beta})$ if and only if $\mathbf{B} = \mathbf{0}$.

# How good is the least squares estimator?

### Corollary 4.5

*Under assumptions (I)-(IV), $\widehat{\beta}_j$ has the lowest variance among all linear estimators of $\beta_j$.*

**Proof:** From theorem 4.4, for any unbiased estimator $\mathbf{b} = \mathbf{L}\mathbf{y}$, the difference

$$\mathbf{P_L} = Var(\mathbf{b}) - Var(\widehat{\boldsymbol{\beta}})$$

is positive semi-definite. Therefore, for all $\mathbf{t} \in \mathbb{R}^{k+1}$, we have

$$\mathbf{t}^T Var(\mathbf{b})\mathbf{t} \geq \mathbf{t}^T Var(\widehat{\boldsymbol{\beta}})\mathbf{t}.$$

By choosing $\mathbf{t}^T = (1, \mathbf{0}_k)$, $\mathbf{t}^T = (0, 1, \mathbf{0}_{k-1})$, $\mathbf{t}^T = (\mathbf{0}_k, 1)$, we have

$$Var(b_0) \geq Var(\widehat{\beta}_0),\ Var(b_1) \geq Var(\widehat{\beta}_1),\ \ldots,\ Var(b_k) \geq Var(\widehat{\beta}_k)$$

# 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?

# Estimation of linear functions

## Theorem 4.6

*Assume (I) - (IV) holds. Take the full rank general linear model* $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \epsilon$ *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* $\boldsymbol{\beta}$*.*

Proof is omitted. In fact, it is very similar to that of Corollary 3.5.

# Estimation of linear functions

The most common use of this theorem is to estimate the (mean) value of the response variable given certain values of the predictor variables.

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

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

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 $\mathbf{t} = \left[ \begin{array}{ccc} 1 & x_1^* & x_2^* \end{array} \right]^T$.

Therefore an unbiased estimator for the house price is

$$\mathbf{t}^T \widehat{\boldsymbol{\beta}} = \left[ \begin{array}{ccc} 1 & x_1^* & x_2^* \end{array} \right] \widehat{\boldsymbol{\beta}} = \widehat{\beta}_0 + \widehat{\beta}_1 x_1^* + \widehat{\beta}_2 x_2^*$$

where $\widehat{\boldsymbol{\beta}}$ 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 $m^2$.

```
betahat

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

c(1,15,2.5)%*%betahat

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

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

# 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

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

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

But....the least squares estimator is still

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

where the design matrix **X** does not have leading column of 1's

$$\mathbf{X} = \begin{pmatrix} x_{11} & \dots & x_{1k} \\ \vdots & \vdots & \vdots \\ x_{n1} & \dots & x_{nk} \end{pmatrix}$$

# Variance estimation

Remember that we assume that the errors $\boldsymbol{\epsilon}$ (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

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

It turns out that this is slightly biased (proof in MM); we need to make a small adjustment.

# Variance estimation

## Theorem 4.7

*The mean-squared error (MSE):*

$$s^2 = \frac{(\mathbf{y} - \mathbf{X}\widehat{\beta})^T(\mathbf{y} - \mathbf{X}\widehat{\beta})}{n-p}$$

*is an unbiased estimator for $\sigma^2$, where $p$ is the number of columns in $\mathbf{X}$.*

Define the sum of squares of the residuals

$$SS_{Res} = (\mathbf{y} - \mathbf{X}\widehat{\beta})^T(\mathbf{y} - \mathbf{X}\widehat{\beta}).$$

Then we can write

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

# Variance estimation

**Proof.**

$$\begin{aligned} E[s^2] &= \frac{1}{n-p} E[(\mathbf{y} - \mathbf{X}\widehat{\beta})^T(\mathbf{y} - \mathbf{X}\widehat{\beta})] \\ &= \frac{1}{n-p} E[(\mathbf{y} - \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y})^T(\mathbf{y} - \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y})] \\ &= \frac{1}{n-p} E[\mathbf{y}^T(\mathbf{I} - \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T)(\mathbf{I} - \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T)\mathbf{y}]. \end{aligned}$$

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

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

# Variance estimation

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

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

where $Var(\mathbf{y}) = \mathbf{V} = \sigma^2\mathbf{I}$.

Here

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

# Variance estimation

and

$$\begin{aligned} tr(\mathbf{AV}) &= tr((\mathbf{I}_n - \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T)\sigma^2\mathbf{I}_n) \\ &= \sigma^2(tr(\mathbf{I}_n) - tr(\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T)) \\ &= \sigma^2(n - tr((\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X})) \\ &= \sigma^2(n - tr(\mathbf{I}_p)) \\ &= \sigma^2(n-p) \end{aligned}$$

which gives the result.

# Variance estimation

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

```
betahat

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

(e <- y - X%*%betahat)

##             [,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 $s^2 = 47.84$.

# 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

(betahat <- solve(t(X)%*%X,t(X)%*%y))

##           [,1]
## [1,] 0.9723810
## [2,] 0.6542857
```

# Variance estimation

```
(e <- y - X%*%betahat)

##              [,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 as $\approx 0.27$.

# Diagnostics

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

Reference for this part: Foundations of Linear and Generalized Linear Models (Agresti, 2015), Linear Models with R (Faraway, 2005)

# Diagnostics: leverage

Consider what happens when we calculate the fitted values by $\hat{\mathbf{y}} = \mathbf{X}\widehat{\beta} = \mathbf{H}\mathbf{y}$, where $\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ is called the *hat matrix*. For the *i*-th data point,

$$\widehat{y}_i = \mathbf{h}_{i\cdot}\mathbf{y} = h_{i1}y_1 + \ldots + h_{i,i-1}y_{i-1} + h_{ii}y_i + h_{i,i+1}y_{i+1} + \ldots + h_{in}y_n$$

Leverage (self-sensitivity):

$$\frac{\partial \widehat{y}_i}{\partial y_i} = h_{ii} = \text{i-th element in main diagonal of } \mathbf{H}$$

It can be shown that $0 \leq h_{ii} \leq 1$. If $h_{ii}$ is large, then:

*Small change in* $y_i \Rightarrow$ *Large change in* $\widehat{y}_i$

By itself, a large leverage is not necessarily detrimental.

# Diagnostics: standardised residuals

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

However we must be careful. Under assumptions (I)-(IV), the variance of each random error equals $\sigma^2$, but the variance of the *residuals* depend on the $h_{ii}$, i.e., $Var(e_i) = \sigma^2(1 - h_{ii})$. (try proving this on your own).

For comparison between residuals, we may use the *standardised residuals*:

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

The standardised residuals have (approximately) equal variance.

# Diagnostics: sensitivity of least squares estimator

We are interested in identifying outliers that affect our regression fit. We need a way to quantify sensitivity of $\widehat{\beta}$ to each data point.

Standardised residuals or leverage, by themselves alone, do not quantify this sensitivity.

To quantify 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

$$D_i = \frac{(\widehat{\boldsymbol{\beta}}_{(-i)} - \widehat{\boldsymbol{\beta}})^T \mathbf{X}^T \mathbf{X} (\widehat{\boldsymbol{\beta}}_{(-i)} - \widehat{\boldsymbol{\beta}})}{ps^2} = \frac{z_i^2}{p} \left( \frac{h_{ii}}{1 - h_{ii}} \right)$$

where $\widehat{\boldsymbol{\beta}}_{(-i)}$ is the estimated parameters if point $i$ is removed and $p$ is the size of $\widehat{\boldsymbol{\beta}}$.

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`).

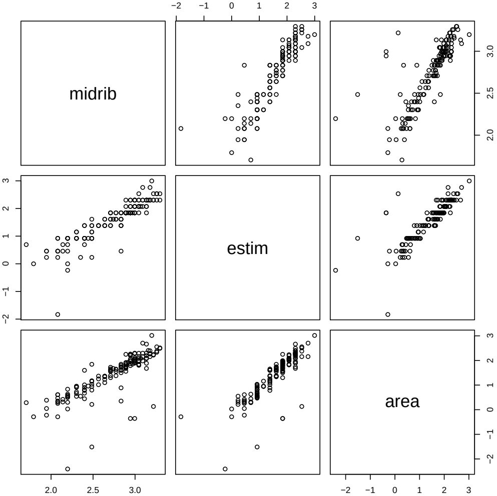
It turns out that (based on knowledge of the data) it is more appropriate to take the logarithms of the data.

```
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

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

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

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

X <- matrix(c(rep(1,145),clover$midrib,clover$estim)
,145,3)
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]
p <- dim(X)[2]

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
```

# 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 <- list(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

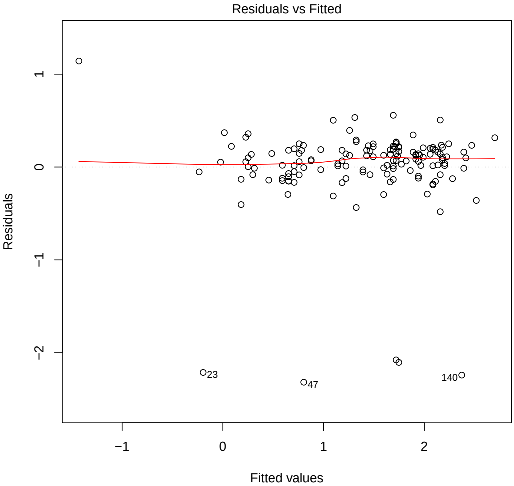
R (and in particular the `lm` command) produces many useful plots for checking the fit of the model and deviations from assumptions.

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

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

# Diagnostic plots

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



# Diagnostic plots

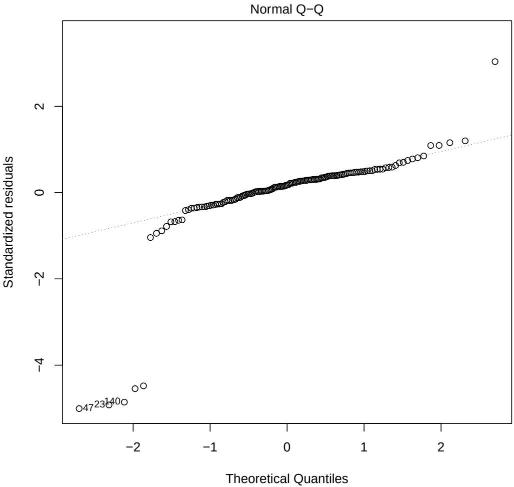
The second plot is a normal quantile-quantile plot of the standardised residuals.

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)
```



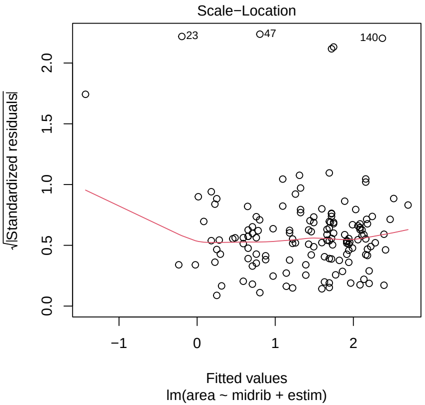
# Diagnostic plots

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 (potential outliers);
- a pattern in the size of the residuals (heteroskedasticity, model misspecification).

# Diagnostic plots

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



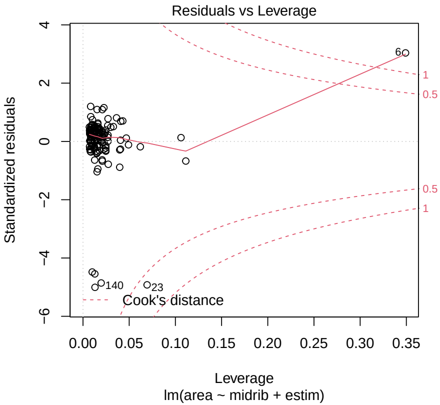
# Diagnostic plots

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

- points with high residual ;
- points with high leverage (influential points);
- points with high Cook's distance (may distort the fit);
- a pattern in the residuals (model misspecification, unequal variance).

# Diagnostic plots

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



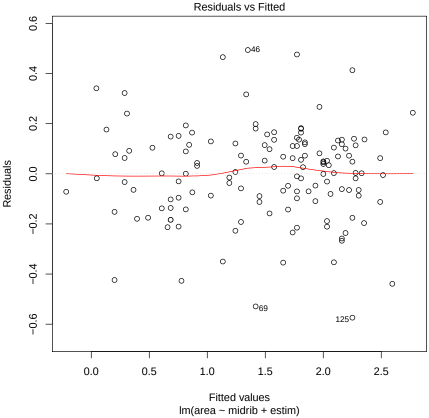
# What if we remove the offending points?

```
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
```

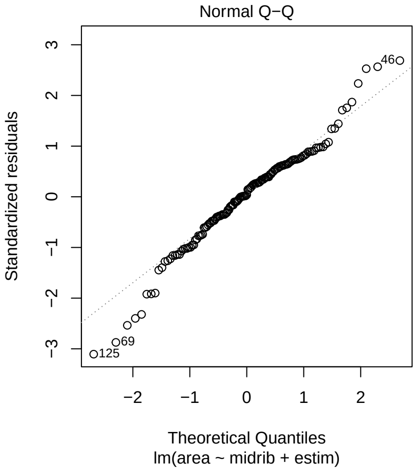
# What if we remove the offending points?

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



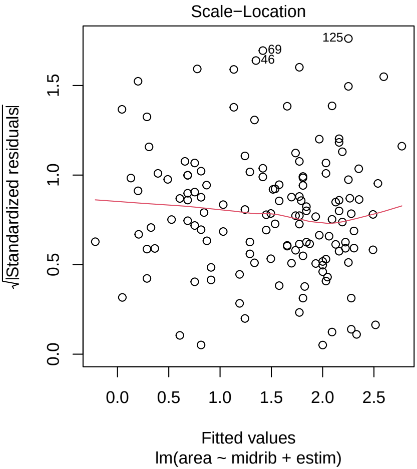
# What if we remove the offending points?

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



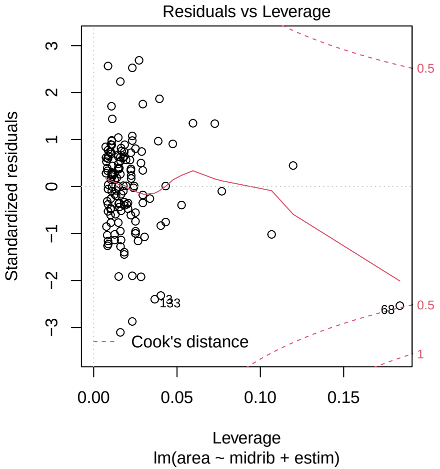
# What if we remove the offending points?

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



# What if we remove the offending points?

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



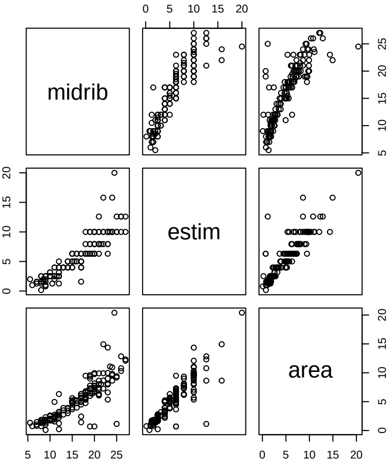
# What if we didn’t take logarithms?

```
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
```

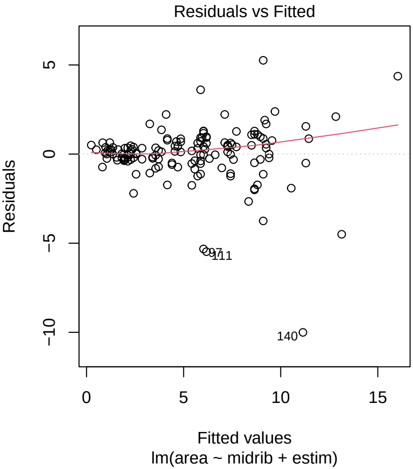
# What if we didn't take logarithms?

```r
pairs(expclover)
```



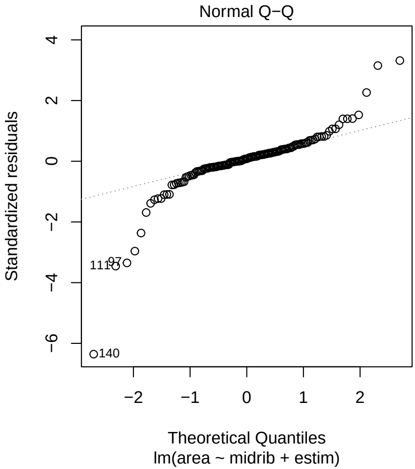
# What if we didn't take logarithms?

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



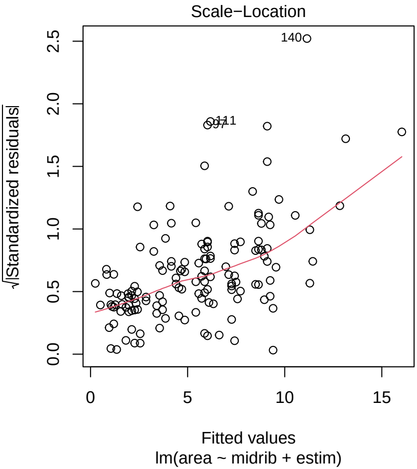
# What if we didn't take logarithms?

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



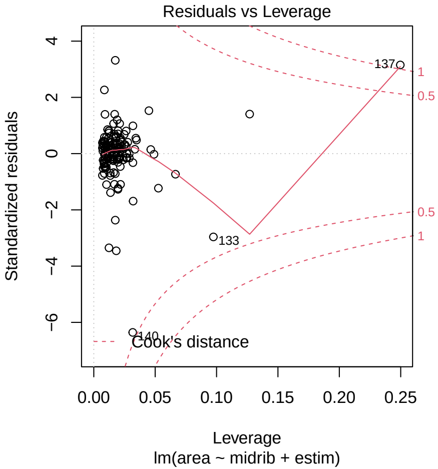
# What if we didn't take logarithms?

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



# What if we didn't take logarithms?

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



# What if we eliminate the intercept term?

```
X3 <- matrix(c(goodclover$midrib, goodclover$estim),
ncol=2)
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
```