Chapter 3. Fitting a linear model to data by least squares

- Recall the sample version of the linear model. Data are $y_1, y_2, \ldots, y_n$ and on each unit i we have p explanatory variables $x_{i1}, x_{i2}, \ldots, x_{ip}$.

(LM1) $$y_i = b_1 x_{i1} + b_2 x_{i2} + \cdots + b_p x_{ip} + e_i \quad \text{for } i = 1, 2, \ldots, n$$

- Using summation notation, we can equivalently write

(LM2) $$y_i = \sum_{j=1}^{p} x_{ij} b_j + e_i \qquad \text{for } i = 1, 2, \ldots, n$$

- We can also use matrix notation. Define column vectors $\mathbf{y} = (y_1, y_2, \ldots, y_n)$, $\mathbf{e} = (e_1, e_2, \ldots, e_n)$ and $\mathbf{b} = (b_1, b_2, \ldots, b_p)$. Define the matrix of explanatory variables, $\mathbb{X} = [x_{ij}]_{n\times p}$. In matrix notation, writing (LM1) or (LM2) is exactly the same as

(LM3) $$\boxed{\mathbf{y} = \mathbb{X}\,\mathbf{b} + \mathbf{e}}$$

- Matrices give a compact way to write the linear model, and also a good way to carry out the necessary computations.

Choosing the coefficient vector, $\mathbf{b}$, by least squares

- We seek the **least squares** choice of $\mathbf{b}$ that minimizes the **residual sum of squares**, $\sum_{i=1}^{n} e_i^2$.
- $\mathbb{X}\,\mathbf{b}$ is the vector of **fitted values**.
- The **residual** for unit i is $e_i = y_i - [\mathbb{X}\,\mathbf{b}]_i$. This is small when the fitted value is close to the data.
- Intuitively, the fit with smallest RSS has fitted values closest to the data, so should be preferred.
- One could use some other criterion, e.g., minimizing the sum of absolute residuals, $\sum_{i=1}^{n} |e_i|$.
- We will find out that RSS is convenient for its mathematical and statistical properties.

The least squares formula

- The least squares choice of $\mathbf{b}$ turns out to be

$$\text{(LM4)} \qquad \mathbf{b} = \left(\mathbb{X}^{\mathrm{T}}\mathbb{X}\right)^{-1}\mathbb{X}^{\mathrm{T}}\mathbf{y}$$

- We will check that this is the formula R uses to fit a linear model.
- We will also gain understanding of (LM4) by studying the **simple linear regression** model $y_i = b_1 x_i + b_2 + e_i$ for which $p = 2$.
- In the simple linear regression model, b_1 and b_2 are called the slope and the intercept.
- In general, $b_1, \ldots, b_p$ are called the **coefficients** of the linear model, and $\mathbf{b}$ is the **coefficient vector**.
- Sometimes, $b_1, \ldots, b_p$ are called **parameters** of the linear model, and $\mathbf{b}$ is the **parameter vector**.
- In R, we obtain $\mathbf{b}$ using the `coef()` function as demonstrated below.

Checking the coefficient estimates from R

- Consider the example from Chapter 1, where `L_detrended` is life expectancy for each year, after subtracting a linear trend, and `U_detrended` is the corresponding detrended unemployment.

```
lm1 <- lm(L_detrended~U_detrended)
coef(lm1)

## (Intercept) U_detrended
##   0.2899928   0.1313673
```

- Now, we can construct the $\mathbb{X}$ matrix corresponding to this linear model and ask R to compute the coefficients using the formula (LM4).

```
X <- cbind(U_detrended,intercept=rep(1,length(U_detrended)))

solve( t(X) %*% X ) %*% t(X) %*% L_detrended

##                  [,1]
## U_detrended 0.1313673
## intercept   0.2899928
```

Checking the $\mathbb{X}$ matrix we constructed

- The matrix calculation on the previous slide matches the coefficients produced by `lm()`.
- We're fairly sure we got the computation right, because we exactly matched `lm()`, but it is a good idea to look at the $\mathbb{X}$ matrix we constructed.

```r
head(X)
```

```
##   U_detrended intercept
## 1  -1.0075234         1
## 2   1.1027941         1
## 3   0.4881116         1
## 4  -1.5349043         1
## 5  -1.8662535         1
## 6  -2.0059360         1
```

```r
length(U_detrended)
```

```
## [1] 68
```

```r
dim(X)
```

```
## [1] 68  2
```

Naming the $\mathbb{X}$ matrix in the linear model $\mathbf{y} = \mathbb{X}\mathbf{b} + \mathbf{e}$

- "The $\mathbb{X}$ matrix" is not a great name since we would have the same model if we had called it $\mathbb{Z}$.
- Many names are used for $\mathbb{X}$ for the many different purposes of linear models.
- Sometimes $\mathbb{X}$ is called the **matrix of predictor variables** or **matrix of explanatory variables**.
- We call $\mathbb{X}$ the **design matrix** in situations where x_{ij} is the setting of adjustable variable j for the ith run of an experiment. For example, y_i could be the stregth of an alloy made up of a fraction x_{ij} of metal j for $j = 1, \ldots, p-1$. We would also want to include an intercept, $x_{ip} = 1$.
- $\mathbb{X}$ can also be called the **matrix of covariates**.
- Sometimes, $\mathbf{y}$ is called the **dependent variable** and $\mathbb{X}$ is the **matrix of independent variables**. Scientifically, an independent variable is one that can be set by the scientist. However, independence has a different technical meaning in statistics.

Fitted values

- The **fitted values** are the estimates of the data based on the explanatory variables. For our linear model, these fitted values are

$$\hat{y}_i = b_1 x_{i1} + b_2 x_{i2} + \cdots + b_p x_{ip}, \qquad \text{for } i = 1, 2, \ldots, n.$$

- The vector of least squares fitted values $\hat{\mathbf{y}} = (\hat{y}_1, \ldots, \hat{y}_n)$ is given by

$$\boxed{\text{(LM5)} \qquad \hat{\mathbf{y}} = \mathbb{X}\mathbf{b} = \mathbb{X}\big(\mathbb{X}^{\mathrm{T}}\mathbb{X}\big)^{-1}\mathbb{X}^{\mathrm{T}}\mathbf{y}.}$$

- It is worth checking we now understand how R produces the fitted values for predicting detrended life expectancy using unemployment:

```
my_fitted_values<-X %*% solve(t(X)%*%X) %*% t(X) %*% L_detrended
```

```
lm1$fitted.values[1:2]

## [1] 0.1576371 0.4348639
```

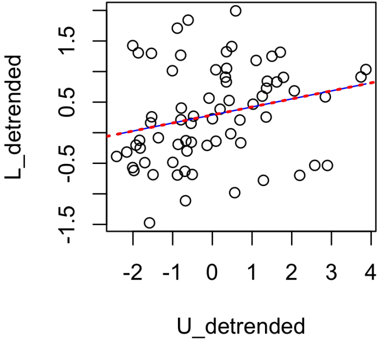
```
my_fitted_values[1:2]

## [1] 0.1576371 0.4348639
```

- We see that the matrix calculation (LM5) exactly matches the fitted values of the `lm1` model that we built earlier using `lm()`.

Plotting the data

- We have already seen plots of the life expectancy and unemployment data before. When you fit a linear model you should look at the data and the fitted values. We plot the fitted values two different ways.

```
plot(L_detrended~U_detrended)
lines(U_detrended,my_fitted_values,lty="solid",col="blue")
abline(coef(lm1),lty="dotted",col="red",lwd=2)
```



Question 3.1. Learn about the `abline()` and `lines()` functions. Explain to yourself why the solid blue line and the dotted red line coincide.

Review of summation signs

- To do statistics, we often want to sum things up over all data points so the **summation sign** $\sum_{i=1}^{n}$ comes up frequently.
- The basic trick to understand $\sum_{i=1}^{n}$ is that anything written using a summation sign can be written as a usual sum.
- As long as you can expand from $\sum_{i=1}^{n} z_i$ to $z_1 + z_2 + \cdots + z_n$, and then contract back again from $z_1 + z_2 + \cdots + z_n$ to $\sum_{i=1}^{n} z_i$, then you can use what you already know about $+$ to work with $\sum_{i=1}^{n}$.

Worked example 1. Can we take a constant outside a sum sign? Is it true that

$$\sum_{i=1}^{n} c y_i = c \sum_{i=1}^{n} y_i.$$

Solution 1. Expand the sum, apply the distributive rule, and contract again.

$$\sum_{i=1}^{n} c y_i \; = \; c y_1 + c y_2 + \cdots + c y_n \; = \; c(y_1 + \cdots + y_n) \; = \; c \sum_{i=1}^{n} y_i.$$

More worked examples using summation signs

Worked example 2. What happens if we sum a constant? Show that

$$\sum_{i=1}^{n} c = nc.$$

Solution 2. Expand the sum, and apply basic addition.

$$\sum_{i=1}^{n} c \;=\; c + c + \cdots + c \text{ } (n \text{ times}) \;=\; nc$$

Deriving the formula for the least squares coefficient vector

This material will not be tested in quizzes and exams. It is presented to explain where the formula (LM4) **for** $\mathbf{b}$ **comes from.**

- We derive (LM4) for the simple linear regression model (SLR1).
- For simple linear regression, the **residual sum of squares (RSS)** is

$$\mathrm{RSS} = \sum_{i=1}^{n} \left(y_i - m x_i - c\right)^2$$

- Calculus fact: To find m and c minimizing RSS, we can differentiate with respect to m and c and set the derivatives equal to zero.
- Calculus fact: Differentiating RSS with respect to m treating c as a constant is called a **partial derivative**, written as $\partial\mathrm{RSS}/\partial m$.
- Calculus fact: If we can find m and c with $\partial\mathrm{RSS}/\partial m = 0$ and $\partial\mathrm{RSS}/\partial c = 0$ we have found a **minimum or maximum** of RSS.
- RSS is non-negative and can get arbitrarily large for bad choices of m and c. It has a minimum but not a maximum.

Differentiating RSS with respect to m

Worked example. Apply the chain rule to differentiate one term in the sum. Check that

$$\frac{\partial}{\partial m}\big(y_i - mx_i - c\big)^2 = (-x_i)\cdot 2\big(y_i - mx_i - c\big)$$

Worked example. Since the derivative of a sum is the sum of the derivatives, check that

$$\frac{\partial}{\partial m}\mathrm{RSS} == 2m\sum_{i=1}^{n} x_i^2 - 2\sum_{i=1}^{n} x_i y_i + 2c\sum_{i=1}^{n} x_i.$$

Differentiating RSS with respect to c

A similar calculation, which you can check if you want the exercise, gives

$$\frac{\partial}{\partial c}\mathrm{RSS} = 2nc - 2\sum_{i=1}^{n} y_i + 2m\sum_{i=1}^{n} x_i.$$

The normal equations

- Now we set the derivatives to zero. This minimizes the residual sum of squares (RSS) giving the least squares values of m and c
- This gives a pair of simultaneous linear equations for m and c:

$$\text{(LS1)} \quad \begin{cases} m\sum_{i=1}^{n} x_i^2 + c\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} x_i y_i \\ m\sum_{i=1}^{n} x_i + cn = \sum_{i=1}^{n} y_i \end{cases}$$

- These are called the **normal equations**.
- We will show they can be written in matrix form as

$$\mathbb{X}^{\mathrm{T}}\mathbb{X}\mathbf{b} = \mathbb{X}^{\mathrm{T}}\mathbf{y}$$

- Therefore, the solution to the normal equations is

$$\mathbf{b} = \left[\mathbb{X}^{\mathrm{T}}\mathbb{X}\right]^{-1} \max X^{\mathrm{T}}\mathbf{y}$$

- This shows (LM4) solves (LS1) and so minimizes the RSS.

Simple linear regression in matrix form

- The matrix form of $y_i = mx_i + c + e_i$, for $i = 1, \ldots, n$, is $\mathbf{y} = \mathbb{X}\mathbf{b} + \mathbf{e}$ where $\mathbf{y} = (y_1, \ldots, y_n)$, $\mathbf{b} = (m, c)$, $\mathbf{e} = (e_1, \ldots, e_n)$, $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{1} = (1, 1, \ldots, 1)$ are column vectors, and $\mathbb{X} = [\,\mathbf{x}\ \mathbf{1}\,]$.
- Written out in full, this is

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix} \begin{bmatrix} m \\ c \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

Question 3.2. For this $\mathbb{X}$, check that $\mathbb{X}^{\mathrm{T}}\mathbb{X} = \begin{bmatrix} \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & n \end{bmatrix}$

Question 3.3. Also, check that $\mathbb{X}^{\mathrm{T}}\mathbf{y} = \begin{bmatrix} \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{bmatrix}$

The normal equations in matrix form

Question 3.4. Check that (LS1) in matrix form is

$$\begin{bmatrix} \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & n \end{bmatrix} \begin{bmatrix} m \\ c \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{bmatrix}$$

- Now we have found that (LS2) and (LS1) are the same equations. Therefore they must have the same solution, which is $\mathbf{b} = \left(\mathbb{X}^{\mathrm{T}}\mathbb{X}\right)^{-1}\mathbb{X}^{\mathrm{T}}\mathbf{y}$.
- We have shown that $\mathbf{b} = \left(\mathbb{X}^{\mathrm{T}}\mathbb{X}\right)^{-1}\mathbb{X}^{\mathrm{T}}\mathbf{y}$ is the least squares coefficient vector for simple linear regression.