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

subscript

This is the ~~**index**~~ **form** of the sample version of the linear model.

- Using summation notation, we can equivalently write

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

subscript

This is the **summation variant** of the ~~index~~ form of the linear model.

- 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

Check LM3 does match LM1. E.g. check what y_1 is from the matrix multiplication in LM3.

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

This is the **matrix form** of the sample version of the linear model.

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 $j = 1, \dots, p-1$.
- $\mathbb{X}$ 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.

The expanded matrix form of the linear model

- We can write $\mathbb{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \ldots \ \mathbf{x}_p]$, where $\mathbf{x}_i = (x_{1i}, x_{2i}, \ldots, x_{ni})$ is the column vector of values of the ith predictor for each of the n units.
- The matrix form of the linear model, $\mathbf{y} = \mathbb{X}\,\mathbf{b} + \mathbf{e}$, can then be **expanded** to

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{n1} \end{bmatrix} b_1 + \begin{bmatrix} x_{12} \\ x_{22} \\ \vdots \\ x_{n2} \end{bmatrix} b_2 + \cdots + \begin{bmatrix} x_{1p} \\ x_{2p} \\ \vdots \\ x_{np} \end{bmatrix} b_p + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

- Often, the matrix of predictors includes a column of ones, commonly called the **intercept**. For example, when $\mathbf{x}_p = (1, 1, \ldots, 1)$ we get

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{n1} \end{bmatrix} b_1 + \cdots + \begin{bmatrix} x_{1\,p-1} \\ x_{2\,p-1} \\ \vdots \\ x_{np-1} \end{bmatrix} b_{p-1} + \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} b_p + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

Question 3.1. Suppose $\mathbb{X}$ is $n \times 2$ and the second column is an intercept, $\mathbf{x}_2 = (1, 1, \ldots, 1)$. This is called "**one predictor plus an intercept**".

(a) Write out this linear model in expanded matrix form.

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{n1} \end{bmatrix} b_1 + \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} b_2 + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix} \qquad (*)$$

(b) Write out the model in subscript form. Hence, explain why $\mathbf{x}_2$ is called the intercept. Look at the top row of $(*)$: $y_1 = x_{11} b_1 + b_2 + e_1$

Generalizing to the i^{th} row:

$$y_i = x_{i1} b_1 + b_2 + e_i$$

If you graph $y = x b_1 + b_2$, the line of fitted values then b_2 is the intercept, where the line hits the y axis.

(c) Would it be more proper to call b_2 the intercept?

People often use the same name for the predictor & the coefficient. R does this too. Column names of the design matrix are used to name coefficients.

Choosing the coefficient vector, b, by least squares

subscript form: $y_i = \sum_j x_{ij} b_j + e_i$

$\Rightarrow e_i = y_i - \underline{\sum_j x_{ij} b_j}$ ← fitted value

- We seek the **least squares** choice of $\mathbf{b}$ that minimizes the **residual sum of squares**, $\mathrm{RSS} = \sum_{i=1}^{n} e_i^2$.
- $\mathbb{X}\,\mathbf{b}$ is the vector of **fitted values**. $\equiv \sum_j x_{ij} b_j$
- The **residual** for unit i is $\underline{e_i = y_i - [\mathbb{X}\,\mathbf{b}]_i}$. 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.

squaring exaggerates large errors & gets rid of negatives.

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.
- Often, $b_1, \dots, b_p$ are called the **coefficients** of the linear model, and $\mathbf{b}$ is the **coefficient vector**.
- Sometimes, $b_1, \dots, b_p$ 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.

u

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

this is our $\underset{\sim}{b}$

$$\mathbb{X} = \begin{bmatrix} u_1 & 1 \\ u_2 & 1 \\ \vdots & \vdots \\ u_n & 1 \end{bmatrix}$$

```
## (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).

argument 1 argument 2

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

function

life expectancy.

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

implementation of $(\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T \underset{\sim}{y}$

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

named arguments to cbind get used to give names to the resulting matrix.

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.

```
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
```

```
length(U_detrended)

## [1] 68

dim(X)

## [1] 68  2
```

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

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

the fitted values are also predicted values for a new measurement at the same predictors.

- 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]
```

we could also (i) look at documentation; (ii) look at source code.

```
## [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()`.

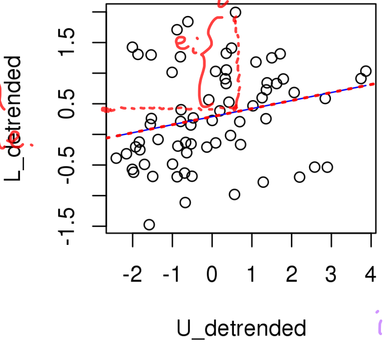
Plotting the data

- 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.

neither is right or wrong.

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

for this to work, the argument should be (intercept, slope) not (slope, intercept)



Question 3.2. Learn about the `abline()` and `lines()` functions. Explain to yourself why the solid blue line and the dotted red line coincide.

?abline and ?lines.
lines() is connecting the points $(x_i, \hat{y}_i)$ and abline() is drawing the line with slope & intercept matching the lm coefficients.

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

Question 3.3. Can we take a constant outside a sum sign? Is it true that

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

$$\begin{aligned}\sum_{i=1}^{n} cy_i &= cy_1 + cy_2 + \dots + cy_n \\ &= c(y_1 + y_2 + \dots + y_n) \\ &= c\sum_{i=1}^{n} y_i\end{aligned}$$

distributive rule

Question 3.4. What happens if we sum a constant? Explain why

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

This is adding c n times.

$$\sum_{i=1}^{n} c = \underbrace{c + c + \dots + c}_{n \text{ terms}} = nc.$$

$$\sum_{j=a}^{b} x_i = \underbrace{\underset{j=a}{x_i} + \underset{j=a+1}{x_i} + \dots \underset{j=b}{x_i}} = (b-a+1)x_i$$

Deriving the formula for the least squares coefficient vector

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

$\underset{\sim}{b} = (\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\underset{\sim}{y}$

here, $\underset{\sim}{b} = (m, c)$

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

- To minimize RSS, we differentiate. Differentiation will not be tested in quizzes and exams. We present it here to understand where the formula (LM4) for $\mathbf{b}$ comes from.
- 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

- Recall that $\mathrm{RSS} = \sum_{i=1}^{n} \left(y_i - mx_i - c\right)^2$. $f(x) = x^2$,

Worked example 3.1. Apply the chain rule to differentiate the ith term in the sum for RSS. Check that

$\frac{d}{dx} f(g(x)) = f'(g(x))\, g'(x)$.

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

This is a direct application of the chain rule.

Worked example 3.2. Since the derivative of a sum is the sum of the derivatives, check that

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

$$\frac{\partial}{\partial m}\mathrm{RSS} = \sum_{i=1}^{n} (-x_i)\cdot 2\left(y_i - mx_i - c\right)$$
$$= 2\sum_{i=1}^{n}\left[m x_i^2 + c x_i - x_i y_i\right]$$

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

note: x_i and y_i are not unknown. They are data!

- 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

$$\text{(LS2)} \qquad \mathbb{X}^{\mathrm{T}}\mathbb{X}\mathbf{b} = \mathbb{X}^{\mathrm{T}}\mathbf{y}$$

$, \underset{\sim}{b} = (m, c)$

$\mathbb{X} = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix}$

- Therefore, the solution to the normal equations is

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

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

Simple linear regression in matrix form

- Recall the subscript form for the simple linear regression model,

$$y_i = mx_i + c + e_i, \quad \text{for } i = 1, \dots, n$$

why is the intercept column a column of ones?

- The matrix form for this model is

ANS: This makes sure the intercept shows up in the equations.

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

where $\mathbf{y} = (y_1, \dots, y_n)$, $\mathbf{b} = (m, c)$, $\mathbf{e} = (e_1, \dots, e_n)$, and $\mathbb{X} = [\,\mathbf{x}\;\mathbf{1}\,]$ for column vectors $\mathbf{x} = (x_1, \dots, x_n)$ and $\mathbf{1} = (1, 1, \dots, 1)$.

- Written out in full, this matrix form is

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

This is a case where moving from matrix notation to subscript notation adds clarity.

Evaluating $\mathbb{X}^{\mathrm{T}}\mathbb{X}$ and $\mathbb{X}^{\mathrm{T}}\mathbf{y}$ for simple linear regression

Question 3.5. For this $\mathbb{X}$, check that $\mathbb{X}^{\mathrm{T}}\mathbb{X} = \begin{bmatrix} \boxed{\sum_{i=1}^n x_i^2} & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & n \end{bmatrix}$

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

Now do matrix multiplication, using $\sum_{i=1}^n 1 = 1$

Question 3.6. Also, check that $\mathbb{X}^{\mathrm{T}}\mathbf{y} = \begin{bmatrix} \boxed{\sum_{i=1}^n x_i y_i} \\ \sum_{i=1}^n y_i \end{bmatrix}$

$$\mathbb{X}^{\mathrm{T}}\underset{\sim}{y} = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} =$$

The normal equations in matrix form

Question 3.7. Check that $(\mathrm{LS1})$ in matrix form is

$$\overbrace{\begin{bmatrix} \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & n \end{bmatrix}}^{\mathbb{X}^\mathrm{T}\mathbb{X}} \overbrace{\begin{bmatrix} m \\ c \end{bmatrix}}^{\mathbf{b}} = \overbrace{\begin{bmatrix} \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{bmatrix}}^{\mathbb{X}^\mathrm{T}\mathbf{y}}$$

$$m \sum_{i=1}^n x_i^2 + c \sum_{i=1}^n x_i = \sum_{i=1}^n x_i y_i$$

$$m \sum_{i=1}^n x_i + nc = \sum_{i=1}^n y_i .$$

- Now we have found that $(\mathrm{LS2})$ and $(\mathrm{LS1})$ 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 $\boxed{\mathbf{b} = \left(\mathbb{X}^\mathrm{T}\mathbb{X}\right)^{-1}\mathbb{X}^\mathrm{T}\mathbf{y}}$ is the least squares coefficient vector for simple linear regression.