Statistical Methods II

Derek L. Sonderegger

2017-08-31

Contents

Chapter 1

Matrix Theory

Almost all of the calculations done in classical statistics require formulas with large number of subscripts and many different sums. In this chapter we will develop the mathematical machinery to write these formulas in a simple compact formula using *matrices*.

1.1 Types of Matrices

We will first introduce the idea behind a matrix and give several special types of matrices that we will encounter.

1.1.1 Scalars

To begin, we first define a *scalar*. A scalar is just a single number, either real or complex. The key is that a scalar is just a single number. For example, 6 is a scalar, as is -3. By convention, variable names for scalars will be lower case and not in bold typeface.

Examples could be $a = 5$, $b = \sqrt{3}$, or $\sigma = 2$.

1.1.2 Vectors

A vector is collection of scalars, arranged as a row or column. Our convention will be that a vector will be a lower cased letter but written in a bold type. In other branches of mathematics is common to put a bar over the variable name to denote that it is a vector, but in statistics, we have already used a bar to denote a mean.

Examples of column vectors could be

$$
\boldsymbol{a} \;\; = \;\; \begin{bmatrix} 2 \\ -3 \\ 4 \end{bmatrix} \quad
\boldsymbol{b} = \begin{bmatrix} 2 \\ 8 \\ 3 \\ 4 \\ 1 \end{bmatrix}
$$

and examples of row vectors are

$$\boldsymbol{c} = \begin{bmatrix} 8 & 10 & 43 & -22 \end{bmatrix}$$

$$\boldsymbol{d} = \begin{bmatrix} -1 & 5 & 2 \end{bmatrix}$$

To denote a specific entry in the vector, we will use a subscript. For example, the second element of $\boldsymbol{d}$ is $d_2 = 5$. Notice, that we do not bold this symbol because the second element of the vector is the scalar value 5.

1.1.3 Matrix

Just as a vector is a collection of scalars, a matrix can be viewed as a collection of vectors (all of the same length). We will denote matrices with bold capitalized letters. In general, I try to use letters at the end of the alphabet for matrices. Likewise, I try to use symmetric letters to denote symmetric matrices.

For example, the following is a matrix with two rows and three columns

$$\boldsymbol{W} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

and there is no requirement that the number of rows be equal, less than, or greater than the number of columns. In denoting the size of the matrix, we first refer to the number of rows and then the number of columns. Thus $\boldsymbol{W}$ is a 2×3 matrix and it sometimes is helpful to remind ourselves of this by writing $\boldsymbol{W}_{2\times3}$.

To pick out a particular element of a matrix, I will again use a subscripting notation, always with the row number first and then column. Notice the notational shift to lowercase, non-bold font.

$$w_{1,2} = 2 \qquad \text{and} \qquad w_{2,3} = 6$$

There are times I will wish to refer to a particular row or column of a matrix and we will use the following notation

$$\boldsymbol{w}_{1,\cdot} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$$

is the first row of the matrix $\boldsymbol{W}$. The second column of matrix $\boldsymbol{W}$ is

$$\boldsymbol{w}_{\cdot,2} = \begin{bmatrix} 2 \\ 5 \end{bmatrix}$$

1.1.4 Square Matrices

A square matrix is a matrix with the same number of rows as columns. The following are square

$$\boldsymbol{Z} = \begin{bmatrix} 3 & 6 \\ 8 & 10 \end{bmatrix} \qquad \boldsymbol{X} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 1 & 2 \\ 3 & 2 & 1 \end{bmatrix}$$

1.1.5 Symmetric Matrices

In statistics we are often interested in square matrices where the i, j element is the same as the j, i element. For example, $x_{1,2} = x_{2,1}$ in the above matrix $\boldsymbol{X}$.

Consider a matrix $\boldsymbol{D}$ that contains the distance from four towns to each of the other four towns. Let $d_{i,j}$ be the distance from town i to town j. It only makes sense that the distance doesn't matter which direction you are traveling, and we should therefore require that $d_{i,j} = d_{j,i}$.

In this example, it is the values $d_{i,i}$ represent the distance from a town to itself, which should be zero. It turns out that we are often interested in the terms $d_{i,i}$ and I will refer to those terms as the *main diagonal* of matrix $\boldsymbol{D}$.

Symmetric matrices play a large role in statistics because matrices that represent the covariances between random variables must be symmetric because $Cov\,(Y, Z) = Cov\,(Z, Y)$.

1.1.6 Diagonal Matrices

A square matrix that has zero entries in every location except the main diagonal is called a diagonal matrix. Here are two examples:

$$\boldsymbol{Q} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 6 \end{bmatrix} \qquad \boldsymbol{R} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 3 \end{bmatrix}$$

Sometimes to make matrix more clear, I will replace the 0 with a dot to emphasize the non-zero components.

$$\boldsymbol{R} = \begin{bmatrix} 1 & \cdot & \cdot & \cdot \\ \cdot & 2 & \cdot & \cdot \\ \cdot & \cdot & 2 & \cdot \\ \cdot & \cdot & \cdot & 3 \end{bmatrix}$$

1.1.7 Identity Matrices

A diagonal matrix with main diagonal values exactly 1 is called the identity matrix. The 3×3 identity matrix is denoted I_3.

$$\boldsymbol{I}_3 = \begin{bmatrix} 1 & \cdot & \cdot \\ \cdot & 1 & \cdot \\ \cdot & \cdot & 1 \end{bmatrix}$$

1.2 Operations on Matrices

1.2.1 Transpose

The simplest operation on a square matrix matrix is called *transpose*. It is defined as $\boldsymbol{M} = \boldsymbol{W}^T$ if and only if $m_{i,j} = w_{j,i}$.

$$\boldsymbol{Z} = \begin{bmatrix} 1 & 6 \\ 8 & 3 \end{bmatrix} \qquad \boldsymbol{Z}^T = \begin{bmatrix} 1 & 8 \\ 6 & 3 \end{bmatrix}$$

$$\boldsymbol{M} = \begin{bmatrix} 3 & 1 & 2 \\ 9 & 4 & 5 \\ 8 & 7 & 6 \end{bmatrix} \qquad \boldsymbol{M}^T = \begin{bmatrix} 3 & 9 & 8 \\ 1 & 4 & 7 \\ 2 & 5 & 6 \end{bmatrix}$$

We can think of this as swapping all elements about the main diagonal. Alternatively we could think about the transpose as making the first row become the first column, the second row become the second column, etc. In this fashion we could define the transpose of a non-square matrix.

$$\boldsymbol{W} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

$$\boldsymbol{W}^T = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$$

1.2.2   Addition and Subtraction

Addition and subtraction are performed *element-wise.* This means that two matrices or vectors can only be added or subtracted if their dimensions match.

$$\begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} + \begin{bmatrix} 5 \\ 6 \\ 7 \\ 8 \end{bmatrix} = \begin{bmatrix} 6 \\ 8 \\ 10 \\ 12 \end{bmatrix}$$

$$\begin{bmatrix} 5 & 8 \\ 2 & 4 \\ 11 & 15 \end{bmatrix} - \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & -6 \end{bmatrix} = \begin{bmatrix} 4 & 6 \\ -1 & 0 \\ 6 & 21 \end{bmatrix}$$

1.2.3   Multiplication

Multiplication is the operation that is vastly different for matrices and vectors than it is for scalars. There is a great deal of mathematical theory that suggests a useful way to define multiplication. What is presented below is referred to as the *dot-product* of vectors in calculus, and is referred to as the standard *inner-product* in linear algebra.

1.2.4   Vector Multiplication

We first define multiplication for a row and column vector. For this multiplication to be defined, both vectors must be the same length. The product is the sum of the element-wise multiplications.

$$\begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix} \begin{bmatrix} 5 \\ 6 \\ 7 \\ 8 \end{bmatrix} = (1 \cdot 5) + (2 \cdot 6) + (3 \cdot 7) + (4 \cdot 8) = 5 + 12 + 21 + 32 = 70$$

1.2.5   Matrix Multiplication

Matrix multiplication is just a sequence of vector multiplications. If $\boldsymbol{X}$ is a $m \times n$ matrix and $\boldsymbol{W}$ is $n \times p$ matrix then $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{W}$ is a $m \times p$ matrix where $z_{i,j} = \boldsymbol{x}_{i,\cdot}\boldsymbol{w}_{\cdot,j}$ where $\boldsymbol{x}_{i,\cdot}$ is the ith column of $\boldsymbol{X}$ and $\boldsymbol{w}_{\cdot,j}$ is the jth column of $\boldsymbol{W}$. For example, let

$$\boldsymbol{X} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{bmatrix} \qquad \boldsymbol{W} = \begin{bmatrix} 13 & 14 \\ 15 & 16 \\ 17 & 18 \\ 19 & 20 \end{bmatrix}$$

so $\boldsymbol{X}$ is 3×4 (which we remind ourselves by adding a 3×4 subscript to $\boldsymbol{X}$ as $\boldsymbol{X}_{3\times4}$) and $\boldsymbol{W}$ is $\boldsymbol{W}_{4\times2}$. Because the *inner* dimensions match for this multiplication, then $\boldsymbol{Z}_{3\times2} = \boldsymbol{X}_{3\times4}\boldsymbol{W}_{4\times2}$ is defined where

$$\begin{aligned} z_{1,1} &= \boldsymbol{x}_{1,\cdot}\boldsymbol{w}_{\cdot,1} \\ &= (1 \cdot 13) + (2 \cdot 15) + (3 \cdot 17) + (4 \cdot 19) = 170 \end{aligned}$$

and similarly

$$
\begin{aligned}
z_{2,1} &= \boldsymbol{x}_{2,\cdot}\boldsymbol{w}_{\cdot,1} \\
&= (5 \cdot 13) + (6 \cdot 15) + (7 \cdot 17) + (8 \cdot 19) = 426
\end{aligned}
$$

so that

$$
\boldsymbol{Z} = \left[\begin{array}{cc} 170 & 180 \\ 426 & 452 \\ 682 & 724 \end{array} \right]
$$

For another example, we note that

$$
\begin{aligned}
\left[\begin{array}{ccc} 1 & 2 & 3 \\ 2 & 3 & 4 \end{array} \right] \left[\begin{array}{cc} 1 & 2 \\ 2 & 2 \\ 1 & 2 \end{array} \right] &= \left[\begin{array}{cc} 1+4+3 & 2+4+6 \\ 2+6+4 & 4+6+8 \end{array} \right] \\
&= \left[\begin{array}{cc} 8 & 12 \\ 12 & 18 \end{array} \right]
\end{aligned}
$$

Notice that this definition of multiplication means that the order matters. Above, we calculated $\boldsymbol{X}_{3\times4}\boldsymbol{W}_{4\times2}$ but we cannot reverse the order because the inner dimensions do not match up.

1.2.6 Scalar times a Matrix

Strictly speaking, we are not allowed to multiply a matrix by a scalar because the dimensions do not match. However, it is often notationally convenient. So we define $a\boldsymbol{X}$ to be the *element-wise* multiplication of each element of $\boldsymbol{X}$ by the scalar a. Because this is just a notational convenience, the mathematical theory about inner-products does not apply to this operation.

$$
5 \left[\begin{array}{cc} 4 & 5 \\ 7 & 6 \\ 9 & 10 \end{array} \right] = \left[\begin{array}{cc} 20 & 25 \\ 35 & 30 \\ 45 & 50 \end{array} \right]
$$

Because of this definition, it is clear that $a\boldsymbol{X} = \boldsymbol{X}a$ and the order does not matter. Thus when mixing scalar multiplication with matrices, it is acceptable to reorder scalars, but not matrices.

1.2.7 Determinant

The determinant is defined only for square matrices and can be thought of as the matrix equivalent of the absolute value or magnitude (i.e. $|-6| = 6$). The determinant gives a measure of the multi-dimensional size of a matrix (say the matrix $\boldsymbol{A}$) and as such is denoted $\det(\boldsymbol{A})$ or $|\boldsymbol{A}|$. Generally this is a very tedious thing to calculate by hand and for completeness sake, we will give a definition and small examples.
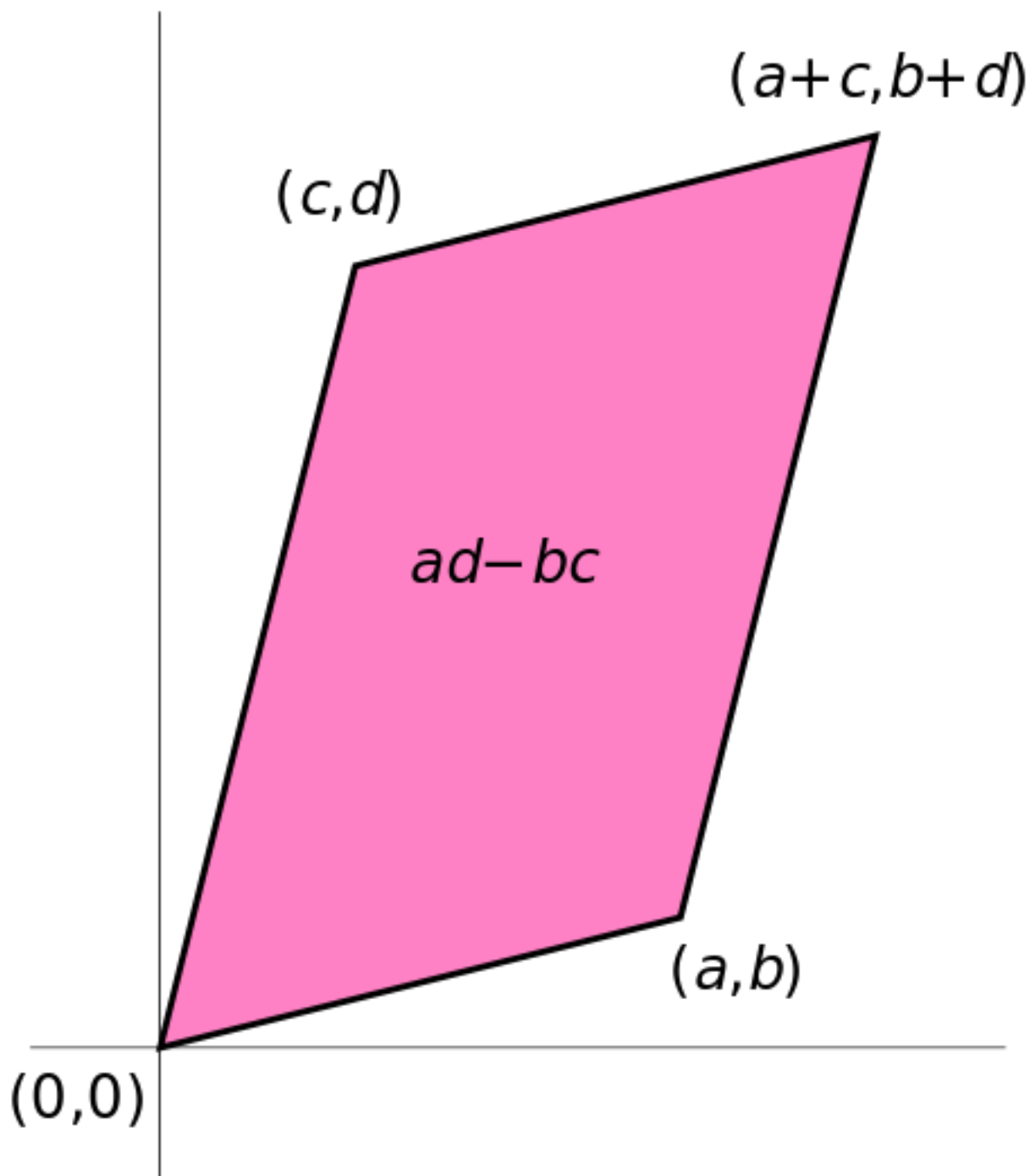
For a 2×2 matrix

$$
\left| \begin{array}{cc} a & c \\ b & d \end{array} \right| = ad - cb
$$

So a simple example of a determinant is

$$
\left| \begin{array}{cc} 5 & 2 \\ 3 & 10 \end{array} \right| = 50 - 6 = 44
$$

The determinant can be thought of as the area of the parallelogram created by the row or column vectors of the matrix.

1.2.8 Inverse

In regular algebra, we are often interested in solving equations such as

$$5x = 15$$

for x. To do so, we multiply each side of the equation by the inverse of 5, which is $1/5$.

$$
\begin{aligned}
5x &= 15 \\
\frac{1}{5} \cdot 5 \cdot x &= \frac{1}{5} \cdot 15 \\
1 \cdot x &= 3 \\
x &= 3
\end{aligned}
$$

For scalars, we know that the inverse of scalar a is the value that when multiplied by a is 1. That is we see to find a^{-1} such that $aa^{-1} = 1$.

In the matrix case, I am interested in finding $\boldsymbol{A}^{-1}$ such that $\boldsymbol{A}^{-1}\boldsymbol{A} = \boldsymbol{I}$ and $\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I}$. For both of these multiplications to be defined, $\boldsymbol{A}$ must be a square matrix and so the inverse is only defined for square matrices.

For a 2×2 matrix

$$\boldsymbol{W} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

the inverse is given by:

$$\boldsymbol{W}^{-1} = \frac{1}{\det \boldsymbol{W}} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

For example, suppose

$$\boldsymbol{W} = \begin{bmatrix} 1 & 2 \\ 5 & 3 \end{bmatrix}$$

then $\det W = 3 - 10 = -7$ and

$$
\begin{aligned}
\boldsymbol{W}^{-1} &= \frac{1}{-7} \begin{bmatrix} 3 & -2 \\ -5 & 1 \end{bmatrix} \\
&= \begin{bmatrix} -\frac{3}{7} & \frac{2}{7} \\ \frac{5}{7} & -\frac{1}{7} \end{bmatrix}
\end{aligned}
$$

and thus

$$
\begin{aligned}
\boldsymbol{W}\boldsymbol{W}^{-1} &= \begin{bmatrix} 1 & 2 \\ 5 & 3 \end{bmatrix} \begin{bmatrix} -\frac{3}{7} & \frac{2}{7} \\ \frac{5}{7} & -\frac{1}{7} \end{bmatrix} \\
&= \begin{bmatrix} -\frac{3}{7} + \frac{10}{7} & \frac{2}{7} - \frac{2}{7} \\ -\frac{15}{7} + \frac{15}{7} & \frac{10}{7} - \frac{3}{7} \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \boldsymbol{I}_2
\end{aligned}
$$

Not every square matrix has an inverse. If the determinant of the matrix (which we think of as some measure of the magnitude or *size* of the matrix) is zero, then the formula would require us to divide by zero. Just as we cannot find the inverse of zero (i.e. solve $0x = 1$ for x), a matrix with zero determinate is said to have no inverse.

1.3   Exercises

Consider the following matrices:

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 6 & 5 & 4 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 6 & 4 & 3 \\ 8 & 7 & 6 \end{bmatrix} \qquad \mathbf{c} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \qquad \mathbf{d} = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} \qquad \mathbf{E} = \begin{bmatrix} 1 & 2 \\ 2 & 6 \end{bmatrix}$$

1. Find $\mathbf{Bc}$
2. Find $\mathbf{AB}^T$
3. Find $\mathbf{c}^T\mathbf{d}$
4. Find $\mathbf{cd}^T$
5. Confirm that $\mathbf{E}^{-1} = \begin{bmatrix} 3 & -1 \\ -1 & 1/2 \end{bmatrix}$ is the inverse of $\mathbf{E}$ by calculating $\mathbf{EE}^{-1} = \mathbf{I}$.

Chapter 2

Parameter Estimation

We have previously looked at ANOVA and regression models and, in many ways, they felt very similar. In this chapter we will introduce the theory that allows us to understand both models as a particular flavor of a larger class of models known as *linear models.*

First we clarify what a linear model is. A linear model is a model where the data (which we will denote using roman letters as $\boldsymbol{x}$ and $\boldsymbol{y}$) and parameters of interest (which we denote using greek letters such as $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$) interact only via addition and multiplication. The following are linear models:

Model	Formula
ANOVA	$y_{ij} = \mu + \tau_i + \epsilon_{ij}$
Simple Regression	$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$
Quadratic Term	$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \epsilon_i$
General Regression	$y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \cdots + \beta_p x_{i,p} + \epsilon_i$

Notice in the Quadratic model, the square is not a parameter and we can consider x_i^2 as just another column of data. This leads to the second example of multiple regression where we just add more slopes for other covariates where the pth covariate is denoted $\boldsymbol{x}_{\cdot,p}$ and might be some transformation (such as x^2 or $\log x$) of another column of data. The critical point is that the transformation to the data $\boldsymbol{x}$ does not depend on a parameter. Thus the following is *not* a linear model

$$y_i = \beta_0 + \beta_1 x_i^\alpha + \epsilon_i$$

2.1   Simple Regression

We would like to represent all linear models in a similar compact matrix representation. This will allow us to make the transition between simple and multiple regression (and ANCOVA) painlessly.

To begin, we think about how to write the simple regression model using matrices and vectors that correspond the the data and the parameters. Notice we have

$$y_1 = \beta_0 + \beta_1 x_1 + \epsilon_1$$
$$y_2 = \beta_0 + \beta_1 x_2 + \epsilon_2$$
$$y_3 = \beta_0 + \beta_1 x_3 + \epsilon_3$$
$$\vdots$$
$$y_{n-1} = \beta_0 + \beta_1 x_{n-1} + \epsilon_{n-1}$$
$$y_n = \beta_0 + \beta_1 x_n + \epsilon_n$$

where, as usual, $\epsilon_i \overset{iid}{\sim} N\left(0, \sigma^2\right)$. These equations can be written using matrices as

$$
\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix}}_{\boldsymbol{y}}
=
\underbrace{\begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \vdots & \vdots \\ 1 & x_{n-1} \\ 1 & x_n \end{bmatrix}}_{\boldsymbol{X}}
\underbrace{\begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}}_{\boldsymbol{\beta}}
+
\underbrace{\begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \vdots \\ \epsilon_{n-1} \\ \epsilon_n \end{bmatrix}}_{\boldsymbol{\epsilon}}
$$

and we compactly write the model as

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

where $\boldsymbol{X}$ is referred to as the *design matrix* and $\boldsymbol{\beta}$ is the vector of *location parameters* we are interested in estimating.

2.1.1 Estimation of Location Paramters

Our next goal is to find the best estimate of $\boldsymbol{\beta}$ given the data. To justify the formula, consider the case where there is no error terms (i.e. $\epsilon_i = 0$ for all i). Thus we have

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}$$

and our goal is to solve for $\boldsymbol{\beta}$. To do this, we must use a matrix inverse, but since inverses only exist for square matrices, we pre-multiple by $\boldsymbol{X}^T$ (notice that $\boldsymbol{X}^T\boldsymbol{X}$ is a symmetric 2×2 matrix).

$$\boldsymbol{X}^T\boldsymbol{y} = \boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta}$$

and then pre-multiply by $\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}$.

$$
\begin{aligned}
\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y} &= \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} \\
\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y} &= \boldsymbol{\beta}
\end{aligned}
$$

This exercise suggests that $\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$ is a good place to start when looking for the maximum-likelihood estimator for $\boldsymbol{\beta}$. It turns out that this quantity is in fact the maximum-likelihood estimator (and equivalently minimizes the sum-of-squared error). Therefore we will use it as our estimate of $\boldsymbol{\beta}$.

$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

2.1.2 Estimation of Variance Parameter

Recall our model is

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

where $\epsilon_i \overset{iid}{\sim} N\left(0, \sigma^2\right)$.

Using our estimates $\hat{\boldsymbol{\beta}}$ we can obtain predicted values for the regression line at any x-value. In particular we can find the predicted value for each x_i value in our dataset.

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$$

Using matrix notation, I would write $\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{\beta}}$.

As usual we will find estimates of the noise terms (which we will call residuals or errors) via

$$
\begin{aligned}
\hat{\epsilon}_i &= y_i - \hat{y}_i \\
&= y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 x_i\right)
\end{aligned}
$$

Writing $\hat{\boldsymbol{y}}$ in matrix terms we have

$$
\begin{aligned}
\hat{\boldsymbol{y}} &= \boldsymbol{X}\hat{\boldsymbol{\beta}} \\
&= \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y} \\
&= \boldsymbol{H}\boldsymbol{y}
\end{aligned}
$$

where $\boldsymbol{H} = \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T$ is often called the *hat-matrix* because it takes y to $\hat{y}$ and has many interesting theoretical properties.[1]

We can now estimate the error terms via

$$
\begin{aligned}
\hat{\boldsymbol{\epsilon}} &= \boldsymbol{y} - \hat{\boldsymbol{y}} \\
&= \boldsymbol{y} - \boldsymbol{H}\boldsymbol{y} \\
&= \left(\boldsymbol{I}_n - \boldsymbol{H}\right)\boldsymbol{y}
\end{aligned}
$$

As usual we estimate σ^2 using the mean-squared error

$$
\begin{aligned}
\hat{\sigma}^2 &= \frac{1}{n-2}\sum_{i=1}^{n}\hat{\epsilon}_i^2 \\
&= \frac{1}{n-2}\hat{\boldsymbol{\epsilon}}^T\hat{\boldsymbol{\epsilon}}
\end{aligned}
$$

In the general linear model case where $\boldsymbol{\beta}$ has p elements (and thus we have $n - p$ degrees of freedom), the formula is

$$
\hat{\sigma}^2 = \frac{1}{n-p}\hat{\boldsymbol{\epsilon}}^T\hat{\boldsymbol{\epsilon}}
$$

---

[1]Mathematically, $\boldsymbol{H}$ is the projection matrix that takes a vector in n-dimensional space and projects it onto a p-dimension subspace spanned by the vectors in $\boldsymbol{X}$. Projection matrices have many useful properties and much of the theory of linear models utilizes $\boldsymbol{H}$.

2.1.3   Expectation and variance of a random vector

Just as we needed to derive the expected value and variance of $\bar{x}$ in the previous semester, we must now do the same for $\hat{\boldsymbol{\beta}}$. But to do this, we need some properties of expectations and variances.

In the following, let $\boldsymbol{A}_{n \times p}$ and $\boldsymbol{b}_{n \times 1}$ be constants and $\boldsymbol{\epsilon}_{n \times 1}$ be a random vector.

Expectations are very similar to the scalar case where

$$E\left[\boldsymbol{\epsilon}\right] = \left[\begin{array}{c} E\left[\epsilon_1\right] \\ E\left[\epsilon_2\right] \\ \vdots \\ E\left[\epsilon_n\right] \end{array}\right]$$

and any constants are pulled through the expectation

$$E\left[\boldsymbol{A}^T \boldsymbol{\epsilon} + \boldsymbol{b}\right] = \boldsymbol{A}^T E\left[\boldsymbol{\epsilon}\right] + \boldsymbol{b}$$

Variances are a little different. The variance of the vector $\boldsymbol{\epsilon}$ is

$$Var\left(\boldsymbol{\epsilon}\right) = \left[\begin{array}{cccc} Var\left(\epsilon_1\right) & Cov\left(\epsilon_1, \epsilon_2\right) & \dots & Cov\left(\epsilon_1, \epsilon_n\right) \\ Cov\left(\epsilon_2, \epsilon_1\right) & Var\left(\epsilon_2\right) & \dots & Cov\left(\epsilon_2, \epsilon_n\right) \\ \vdots & \vdots & \ddots & \vdots \\ Cov\left(\epsilon_n, \epsilon_1\right) & Cov\left(\epsilon_n, \epsilon_2\right) & \dots & Var\left(\epsilon_1\right) \end{array}\right]$$

and additive constants are ignored, but multiplicative constants are pulled out as follows:

$$Var\left(\boldsymbol{A}^T \boldsymbol{\epsilon} + \boldsymbol{b}\right) = Var\left(\boldsymbol{A}^T \boldsymbol{\epsilon}\right) = \boldsymbol{A}^T Var\left(\boldsymbol{\epsilon}\right) \boldsymbol{A}$$

2.1.4   Variance of Location Parameters

We next derive the sampling variance of our estimator $\hat{\boldsymbol{\beta}}$ by first noting that $\boldsymbol{X}$ and $\boldsymbol{\beta}$ are constants and therefore

$$\begin{aligned} Var\left(\boldsymbol{y}\right) &= Var\left(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}\right) \\ &= Var\left(\boldsymbol{\epsilon}\right) \\ &= \sigma^2 \boldsymbol{I}_n \end{aligned}$$

because the error terms are independent and therefore $Cov\left(\epsilon_i, \epsilon_j\right) = 0$ when $i \neq j$ and $Var\left(\epsilon_i\right) = \sigma^2$. Recalling that constants come out of the variance operator as the constant *squared,*

$$\begin{aligned} Var\left(\hat{\boldsymbol{\beta}}\right) &= Var\left(\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}\right) \\ &= \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T Var\left(\boldsymbol{y}\right) \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1} \\ &= \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T \sigma^2 \boldsymbol{I}_n \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1} \\ &= \sigma^2\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1} \\ &= \sigma^2\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1} \end{aligned}$$

Using this, the standard error (i.e. the estimated standard deviation) of $\hat{\beta}_j$ (for any j in $1, \ldots, p$) is

$$StdErr\left(\hat{\beta}_j\right) = \sqrt{\hat{\sigma}^2 \left[\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\right]_{jj}}$$

2.1.5 Confidence intervals and hypothesis tests

We can now state the general method of creating confidence intervals and perform hypothesis tests for any element of $\boldsymbol{\beta}$.

The confidence interval formula is (as usual)

$$\hat{\beta}_j \pm t_{n-p}^{1-\alpha/2}\, StdErr\left(\hat{\beta}_j\right)$$

and a test statistic for testing $H_0 : \beta_j = 0$ versus $H_a : \beta_j \neq 0$ is

$$t_{n-p} = \frac{\hat{\beta}_j - 0}{StdErr\left(\hat{\beta}_j\right)}$$

2.1.6 Summary of pertinent results

- $\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$ is the unbiased maximum-likelihood estimator of $\boldsymbol{\beta}$.

- The Central Limit Theorem applies to each element of $\boldsymbol{\beta}$. That is, as $n \to \infty$, the distribution of $\hat{\beta}_j \to N\left(\beta_j, \left[\sigma^2\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\right]_{jj}\right)$.

- The error terms can be calculated via

$$\begin{aligned} \hat{\boldsymbol{y}} &= \boldsymbol{X}\hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{\epsilon}} &= \boldsymbol{y} - \hat{\boldsymbol{y}} \end{aligned}$$

- The estimate of σ^2 is

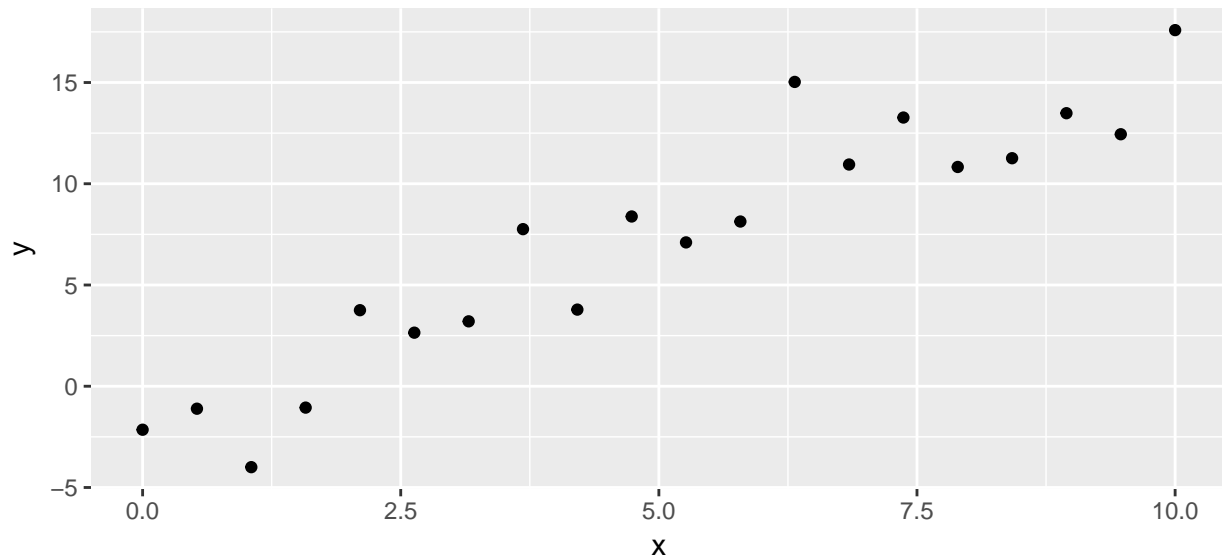$$\hat{\sigma}^2 = \frac{1}{n-p}\, \hat{\boldsymbol{\epsilon}}^T\hat{\boldsymbol{\epsilon}}$$

- The standard error (i.e. the estimated standard deviation) of $\hat{\beta}_j$ (for any j in $1, \ldots, p$) is

$$StdErr\left(\hat{\beta}_j\right) = \sqrt{\hat{\sigma}^2 \left[\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\right]_{jj}}$$

2.1.7 An example in R

Here we will work an example in R and see the calculations. Consider the following data:

```r
library(ggplot2)
n <- 20
x <- seq(0,10, length=n)
y <- -3 + 2*x + rnorm(n, sd=2)
my.data <- data.frame(x=x, y=y)
ggplot(my.data) + geom_point(aes(x=x,y=y))
```

First we must create the design matrix $\boldsymbol{X}$. Recall

$$\boldsymbol{X} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \vdots & \vdots \\ 1 & x_{n-1} \\ 1 & x_n \end{bmatrix}$$

and can be created in R via the following:

```
X <- cbind( rep(1,n), x)
```

Given $\boldsymbol{X}$ and $\boldsymbol{y}$ we can calculate

$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T \boldsymbol{X} \right)^{-1} \boldsymbol{X}^T \boldsymbol{y}$$

in R using the following code:

```
XtXinv <- solve( t(X) %*% X )
beta.hat <- XtXinv %*% t(X) %*% y
beta.hat
```

```
##          [,1]
##    -2.244625
## x  1.862125
```

Our next step is to calculate the predicted values $\hat{\boldsymbol{y}}$ and the residuals $\hat{\boldsymbol{\epsilon}}$

$$\begin{aligned} \hat{\boldsymbol{y}} &= \boldsymbol{X}\hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{\epsilon}} &= \boldsymbol{y} - \hat{\boldsymbol{y}} \end{aligned}$$

```
y.hat <- X %*% beta.hat
residuals <- y - y.hat
```

Now that we have the residuals, we can calculate $\hat{\sigma}^2$ and the standard errors of $\hat{\beta}_j$

$$\hat{\sigma}^2 = \frac{1}{n-p} \hat{\boldsymbol{\epsilon}}^T \hat{\boldsymbol{\epsilon}}$$

$$StdErr\left(\hat{\beta}_j\right) = \sqrt{\hat{\sigma}^2 \left[\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\right]_{jj}}$$

```
sigma2.hat <- ( t(residuals) %*% residuals) / (n-2)
sigma.hat <- sqrt( sigma2.hat )
std.errs <- sqrt( sigma2.hat * diag(XtXinv) )
```

We now print out the important values and compare them to the summary output given by the `lm()` function in R.

```
beta.hat
```

```
##        [,1]
##   -2.244625
## x  1.862125
```

```
sigma.hat
```

```
##          [,1]
## [1,] 2.241106
```

```
std.errs
```

```
## [1] 0.9657951 0.1651221
```

```
model <- lm(y~x)
summary(model)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7118 -1.6578 -0.2063  1.3562  5.5124
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.2446     0.9658  -2.324    0.032 *
## x             1.8621     0.1651  11.277 1.36e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.241 on 18 degrees of freedom
## Multiple R-squared:  0.876,  Adjusted R-squared:  0.8691
## F-statistic: 127.2 on 1 and 18 DF,  p-value: 1.363e-09
```

We calculate 95% confidence intervals via:

```
lwr <- beta.hat - qt(.975, n-2) * std.errs
upr <- beta.hat + qt(.975, n-2) * std.errs
CI <- cbind(lwr,upr)
colnames(CI) <- c('lower','upper')
rownames(CI) <- c('Intercept', 'x')
CI
```

```
##               lower      upper
## Intercept -4.273685 -0.2155646
## x          1.515216  2.2090331
```

These intervals are the same as what we get when we use the `confint()` function.

```
confint(model)
```

```
##                    2.5 %      97.5 %
## (Intercept) -4.273685 -0.2155646
## x            1.515216  2.2090331
```

2.2 ANOVA model

The anova model is also a linear model and all we must do is create a appropriate design matrix. Given the design matrix X, all the calculations are identical as in the simple regression case.

2.2.1 Cell means representation

Recall the cell means representation is

$$y_{i,j} = \mu_i + \epsilon_{i,j}$$

where $y_{i,j}$ is the jth observation within the ith group. To clearly show the creation of the X matrix, let the number of groups be $p = 3$ and the number of observations per group be $n_i = 4$. We now expand the formula to show all the data.

$$
\begin{aligned}
y_{1,1} &= \mu_1 + \epsilon_{1,1} \\
y_{1,2} &= \mu_1 + \epsilon_{1,2} \\
y_{1,3} &= \mu_1 + \epsilon_{1,3} \\
y_{1,4} &= \mu_1 + \epsilon_{1,4} \\
y_{2,1} &= \mu_2 + \epsilon_{2,1} \\
y_{2,2} &= \mu_2 + \epsilon_{2,2} \\
y_{2,3} &= \mu_2 + \epsilon_{2,3} \\
y_{2,4} &= \mu_2 + \epsilon_{2,4} \\
y_{3,1} &= \mu_3 + \epsilon_{3,1} \\
y_{3,2} &= \mu_3 + \epsilon_{3,2} \\
y_{3,3} &= \mu_3 + \epsilon_{3,3} \\
y_{3,4} &= \mu_3 + \epsilon_{3,4}
\end{aligned}
$$

In an effort to write the model as $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ we will write the above as

$$
\begin{aligned}
y_{1,1} &= 1\mu_1 + 0\mu_2 + 0\mu_3 + \epsilon_{1,1} \\
y_{1,2} &= 1\mu_1 + 0\mu_2 + 0\mu_3 + \epsilon_{1,2} \\
y_{1,3} &= 1\mu_1 + 0\mu_2 + 0\mu_3 + \epsilon_{1,3} \\
y_{1,4} &= 1\mu_1 + 0\mu_2 + 0\mu_3 + \epsilon_{1,4} \\
y_{2,1} &= 0\mu + 1\mu_2 + 0\mu_3 + \epsilon_{2,1} \\
y_{2,2} &= 0\mu + 1\mu_2 + 0\mu_3 + \epsilon_{2,2} \\
y_{2,3} &= 0\mu + 1\mu_2 + 0\mu_3 + \epsilon_{2,3} \\
y_{2,4} &= 0\mu + 1\mu_2 + 0\mu_3 + \epsilon_{2,4} \\
y_{3,1} &= 0\mu + 0\mu_2 + 1\mu_3 + \epsilon_{3,1} \\
y_{3,2} &= 0\mu + 0\mu_2 + 1\mu_3 + \epsilon_{3,2} \\
y_{3,3} &= 0\mu + 0\mu_2 + 1\mu_3 + \epsilon_{3,3} \\
y_{3,4} &= 0\mu + 0\mu_2 + 1\mu_3 + \epsilon_{3,4}
\end{aligned}
$$

and we will finally be able to write the matrix version

$$
\underbrace{\begin{bmatrix} y_{1,1} \\ y_{1,2} \\ y_{1,3} \\ y_{1,4} \\ y_{2,1} \\ y_{2,2} \\ y_{2,3} \\ y_{2,4} \\ y_{3,1} \\ y_{3,2} \\ y_{3,3} \\ y_{3,4} \end{bmatrix}}_{\boldsymbol{y}} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix}}_{\beta} + \underbrace{\begin{bmatrix} \epsilon_{1,1} \\ \epsilon_{1,2} \\ \epsilon_{1,3} \\ \epsilon_{1,4} \\ \epsilon_{2,1} \\ \epsilon_{2,2} \\ \epsilon_{2,3} \\ \epsilon_{2,4} \\ \epsilon_{3,1} \\ \epsilon_{3,2} \\ \epsilon_{3,3} \\ \epsilon_{3,4} \end{bmatrix}}_{\epsilon}
$$

Notice that each column of the $\boldsymbol{X}$ matrix is acting as an indicator if the observation is an element of the appropriate group. As such, these are often called *indicator variables*. Another term for these, which I find less helpful, is *dummy variables*.

2.2.2   Offset from reference group

In this model representation of ANOVA, we have an overall mean and then offsets from the control group (which will be group one). The model is thus

$$
y_{i,j} = \mu + \tau_i + \epsilon_{i,j}
$$

where $\tau_1 = 0$. We can write this in matrix form as

$$
\underbrace{\begin{bmatrix} y_{1,1} \\ y_{1,2} \\ y_{1,3} \\ y_{1,4} \\ y_{2,1} \\ y_{2,2} \\ y_{2,3} \\ y_{2,4} \\ y_{3,1} \\ y_{3,2} \\ y_{3,3} \\ y_{3,4} \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} \mu \\ \tau_2 \\ \tau_3 \end{bmatrix}}_{\beta} + \underbrace{\begin{bmatrix} \epsilon_{1,1} \\ \epsilon_{1,2} \\ \epsilon_{1,3} \\ \epsilon_{1,4} \\ \epsilon_{2,1} \\ \epsilon_{2,2} \\ \epsilon_{2,3} \\ \epsilon_{2,4} \\ \epsilon_{3,1} \\ \epsilon_{3,2} \\ \epsilon_{3,3} \\ \epsilon_{3,4} \end{bmatrix}}_{\epsilon}
$$

2.3   Exercises

1. We will do a simple ANOVA analysis on example 8.2 from Ott & Longnecker using the matrix representation of the model. A clinical psychologist wished to compare three methods for reducing hostility levels in university students, and used a certain test (HLT) to measure the degree of hostility. A high score on the test indicated great hostility. The psychologist used 24 students who obtained high and nearly equal scores in the experiment. eight were selected at random from among the 24 problem cases and were treated with method 1. Seven of the remaining 16 students were selected at random and treated with method 2. The remaining nine students were treated with method 3. All treatments were continued for a one-semester period. Each student was given the HLT test at the end of the semester, with the results show in the following table. (This analysis was done in section 8.3 of my STA 570 notes)

Method	Values
1	96, 79, 91, 85, 83, 91, 82, 87
2	77, 76, 74, 73, 78, 71, 80
3	66, 73, 69, 66, 77, 73, 71, 70, 74

We will be using the cell means model of ANOVA

$$y_{ij} = \beta_i + \epsilon_{ij}$$

where β_i is the mean of group i and $\epsilon_{ij} \overset{iid}{\sim} N\left(0, \sigma^2\right)$.

  a. Create one vector of all 24 hostility test scores $\mathbf{y}$. (Use the `c()` function.)

  b. Create a design matrix $\mathbf{X}$ with dummy variables for columns that code for what group an observation belongs to. Notice that $\mathbf{X}$ will be a 24 rows by 3 column matrix. An R function that might be handy is `cbind(a,b)` which will bind two vectors or matrices together along the columns. (There is also a corresponding `rbind()` function that binds vectors/matrices along rows.)

  c) Find $\hat{\beta}$ using the matrix formula given in class. The R function `t(A)` computes the matrix transpose $\mathbf{A}^T$, `solve(A)` computes $\mathbf{A}^{-1}$, and the operator `%*%` does matrix multiplication (used as `A %*% B`).

  d) Examine the matrix $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T$. What do you notice about it? In particular, think about the result when you right multiply by $\mathbf{y}$. How does this matrix calculate the appropriate group means and using the appropriate group sizes n_i?

2. We will calculate the y-intercept and slope estimates in a simple linear model using matrix notation. We will use a data set that gives the diameter at breast height (DBH) versus tree height for a randomly selected set of trees. In addition, for each tree, a ground measurement of crown closure (CC) was taken. Larger values of crown closure indicate more shading and is often associated with taller tree morphology (possibly). We will be interested in creating a regression model that predicts height based on DBH and CC. In the interest of reduced copying, we will only use 10 observations. *(Note: I made this data up and the DBH values might be unrealistic. Don't make fun of me.)*

DBH	30.5	31.5	31.7	32.3	33.3	35	35.4	35.6	36.3	37.8
CC	0.74	0.69	0.65	0.72	0.58	0.5	0.6	0.7	0.52	0.6
Height	58	64	65	70	68	63	78	80	74	76

We are interested in fitting the regression model

$$y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \epsilon_i$$

where β_0 is the y-intercept and β_1 is the slope parameter associated with DBH and β_2 is the slope parameter associated with Crown Closure.

a) Create a vector of all 10 heights **y**.
b) Create the design matrix **X**.
c) Find $\hat{\boldsymbol{\beta}}$ using the matrix formula given in class.
d) Compare your results to the estimated coefficients you get using the `lm()` function. To add the second predictor to the model, your call to `lm()` should look something like `lm(Height ~ DBH + CrownClosure)`.

Bibliography