The Mathematical Derivation of Least Squares

Back when the powers that be forced you to learn matrix algebra and calculus, I bet you all asked yourself the age-old question: "When the hell will I use this stuff?" Well, at long last, that "when" is *now*! Given the centrality of the linear regression model to research in the social and behavioral sciences, your decision to become a psychologist more or less ensures that you will regularly use a tool that is critically dependent on matrix algebra and differential calculus in order to do some quantitative heavy lifting.

As you know, both bivariate and multiple OLS regression requires us to estimate values for a critical set of parameters: a *regression constant* and *one regression coefficient* for each independent variable in our model. The regression constant tells us the predicted value of the dependent variable (DV, hereafter) when all of the independent variables (IVs, hereafter) equal 0. The *unstandardized* regression coefficient for each IV tells us how much the predicted value of the DV would change with a one-unit increase in the IV, *when all other IVs are at 0*.

OLS estimates these parameters by finding the values for the constant and coefficients that minimize the sum of the squared errors of prediction, i.e., the differences between a case's actual score on the DV and the score we predict for them using actual scores on the IVs. For both the bivariate and multiple regression cases, this handout will show how this is done – hopefully shedding light on the conceptual underpinnings of regression itself.

The Bivariate Case

For the case in which there is only one IV, the classical OLS regression model can be expressed as follows:

$$y_i = b_0 + b_1 x_i + e_i \qquad (1)$$

where y_i is case i's score on the DV, x_i is case i's score on the IV, b_0 is the regression constant, b_1 is the regression coefficient for the effect of x, and e_i is the error we make in predicting y from x.

Now, in running the regression model, what are trying to do is to minimize the sum of the squared errors of prediction – i.e., of the e_i values – across all cases. Mathematically, this quantity can be expressed as:

$$SSE = \sum_{i=1}^{N} e_i^2 \qquad (2)$$

Specifically, what we want to do is find the values of b_0 and b_1 that minimize the quantity in Equation 2 above.

So, how do we do this? The key is to think back to differential calculus and remember how one goes about *finding the minimum value* of a mathematical function. This involves taking the *derivative* of that function. As you may recall, if y is some mathematical function of variable x, the derivative of y with respect to x is *the amount of change in y that occurs with a tiny change in x*.[1] Roughly, it's the instantaneous rate of change in y with respect to changes in x.

So, what does this have to do with the minimum of a mathematical function? Well, the derivative of function y with respect to x – the extent to which y changes with a tiny change in x – equals zero when y is at its minimum value. If we find the value of x for which the derivative of y equals zero, then we have found the value of x for which y is neither increasing nor decreasing with respect to x.[2]

Thus, if we want to find the values of b_0 and b_1 that minimize SSE, we need to express SSE in terms of b_0 and b_1, take the derivatives of SSE with respect to b_0 and b_1, set these derivatives to zero, and solve for b_0 and b_1.

---

[1] Formally, for the mathematically inclined, the derivative of y with respect to x – dy/dx – is defined as:

$$\frac{dy}{dx} = \lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x}$$

In plain English, it's the value that the change in y – Δy – relative to the change in x – Δx – converges on as the size of Δx approaches zero. It is an *instantaneous* rate of change in y.

[2] Note that the value of x for which the derivative of y equals zero can also indicate a maximum. However, we can be sure that we have found a minimum if the *second derivative* of y with respect to x – i.e., the derivative of the derivative of y with respect to x – has a positive value at the value of x for which the derivative of y equals zero. As we will see below, this is the case with regard to the derivatives of SSE with respect to the regression constant and coefficient.

However, since SSE is a function of two critical variables – b_0 and b_1 – we will need to take the *partial derivatives* of SSE with respect to b_0 and b_1. In practice, this means we will need to take the derivative of SSE with regard to each of these critical variables one at a time, while treating the other critical variable as a constant (keeping in mind that the derivative of a constant always equals zero). In effect, what this does is take the derivative of SSE with respect to one variable while holding the other constant.

We begin by rearranging the basic OLS equation for the bivariate case so that we can express e_i in terms of y_i, x_i, b_0, and b_1. This gives us:

$$e_i = y_i - b_0 - b_1 x_i \tag{3}$$

Substituting this expression back into Equation (2), we get

$$SSE = \sum_{i=1}^{N} (y_i - b_0 - b_1 x_i)^2 \tag{4}$$

where N = the sample size for the data. It is this expression that we actually need to differentiate with respect to b_0 and b_1. Let's start by taking the partial derivative of SSE with respect to the regression constant, b_0, i.e.,

$$\frac{\partial SSE}{\partial b_0} = \frac{\partial}{\partial b_0}\left[\sum_{i=1}^{N} (y_i - b_0 - b_1 x_i)^2\right]$$

In doing this, we can move the summation operator (Σ) out front, since the derivative of a sum is equal to the sum of the derivatives:

$$\frac{\partial SSE}{\partial b_0} = \sum_{i=1}^{N}\left[\frac{\partial}{\partial b_0}(y_i - b_0 - b_1 x_i)^2\right]$$

We then focus on differentiating the squared quantity in parentheses. Since this quantity is a composite – we do the math in parentheses and then square the result – we need to use the chain rule in order to obtain the partial derivative of SSE with respect to the regression constant.[3] In order to do this, we treat y_i, b_1, and x_i as constants. This gives us:

[3] Use of the chain rule in this context is a two-step procedure. In the first step, we take the partial derivative of the quantity in parentheses with respect to b_0. Here, we treat y_i, b_1, and x_i as

$$\frac{\partial SSE}{\partial b_0} = \sum_{i=1}^{N}\left[-2\left(y_i - b_0 - b_1 x_i\right)\right]$$

Further rearrangement gives us a final result of:

$$\frac{\partial SSE}{\partial b_0} = -2\sum_{i=1}^{N}\left(y_i - b_0 - b_1 x_i\right) \tag{5}$$

For the time being, let's put this result aside and take the partial derivative of SSE with respect to the regression coefficient, b_1, i.e.,

$$\frac{\partial SSE}{\partial b_1} = \frac{\partial}{\partial b_1}\left[\sum_{i=1}^{N}\left(y_i - b_0 - b_1 x_i\right)^2\right]$$

Again, we can move the summation operator (Σ) out front:

$$\frac{\partial SSE}{\partial b_1} = \sum_{i=1}^{N}\left[\frac{\partial}{\partial b_1}\left(y_i - b_0 - b_1 x_i\right)^2\right]$$

We then differentiate the squared quantity in parentheses, again using the chain rule. This time, however, we treat y_i, b_0, and x_i as constants. With some subsequent rearrangement, this gives us:

$$\frac{\partial SSE}{\partial b_1} = -2\sum_{i=1}^{N} x_i\left(y_i - b_0 - b_1 x_i\right) \tag{6}$$

---

constants, meaning that the derivatives of first and last terms in this quantity equal zero. In order to take the derivative of the middle term ($-b_0$), we subtract one from the value of the exponent on $-b_0$ (i.e., 1 – 1 = 0) and multiply this result by the exponent on $-b_0$ (i.e., 1) from the original expression. Since raising b_0 to the power of zero gives us 1, the derivative for the quantity in parentheses is -1. In the second step, we take the derivative of $(y_i - b_0 - b_1 x_i)^2$ with respect to $(y_i - b_0 - b_1 x_i)$. We do this by subtracting one from the value of the exponent on the quantity in parentheses (i.e., 2 – 1 = 1) and multiply this result by the exponent on the quantity in parentheses (i.e., 2) from the original expression. This gives us $2(y_i - b_0 - b_1 x_i)$. Multiplying this by the result from the first step, we get a final result of $-2(y_i - b_0 - b_1 x_i)$.

With that, we have our two partial derivatives of SSE – in Equations (5) and (6).[4] The next step is to set each one of them to zero:

$$0 = -2\sum_{i=1}^{N}\left(y_i - b_0 - b_1 x_i\right) \tag{7}$$

$$0 = -2\sum_{i=1}^{N} x_i\left(y_i - b_0 - b_1 x_i\right) \tag{8}$$

Equations (7) and (8) form a system of equations with two unknowns – our OLS estimates, b_0 and b_1. The next step is to solve for these two unknowns. We start by solving Equation (7) for b_0. First, we get rid of the -2 by multiplying each side of the equation by -1/2:

$$0 = \sum_{i=1}^{N}\left(y_i - b_0 - b_1 x_i\right)$$

Next, we distribute the summation operator though all of the terms in the expression in parentheses:

$$0 = \sum_{i=1}^{N} y_i - \sum_{i=1}^{N} b_0 - \sum_{i=1}^{N} b_1 x_i$$

Then, we add the middle summation term on the right to both sides of the equation, giving us:

$$\sum_{i=1}^{N} b_0 = \sum_{i=1}^{N} y_i - \sum_{i=1}^{N} b_1 x_i$$

Since b_0 and b_1 the same for all cases in the original OLS equation, this further simplifies to:

$$Nb_0 = \sum_{i=1}^{N} y_i - b_1 \sum_{i=1}^{N} x_i$$

[4] The second partial derivatives of SSE with respect to b_0 and b_1 are $2N$ and $2Nx_i^2$, respectively. Since both of these values are necessarily positive (i.e., because 2, N, and the square of x_i will always be positive), we can be sure that the values of b_0 and b_1 that satisfy the equations generated by setting each partial derivative to zero refer to minimum rather than maximum values of SSE.

To isolate b_0 on the left side of the equation, we then divide both sides by N:

$$b_0 = \left(\frac{\sum_{i=1}^{N} y_i}{N} \right) - b_1 \left(\frac{\sum_{i=1}^{N} x_i}{N} \right) \tag{9}$$

Equation (9) will come in handy later on, so keep it in mind. Right now, though, it is important to note that the first term on the right of Equation (9) is simply the mean of y_i, while everything following b_1 in the second term on the right is the mean of x_i. This simplifies the equation for b_1 to the form from lecture:

$$b_0 = \bar{y} - b_1 \bar{x} \tag{10}$$

Now, we need to solve Equation (8) for b_1. Again, we get rid of the -2 by multiplying each side of the equation by -1/2:

$$0 = \sum_{i=1}^{N} x_i \left(y_i - b_0 - b_1 x_i \right)$$

Next, we distribute x_i through all of the terms in parentheses:

$$0 = \sum_{i=1}^{N} \left(y_i x_i - b_0 x_i - b_1 x_i^2 \right)$$

We then distribute the summation operator through all of the terms in the expression in parentheses:

$$0 = \sum_{i=1}^{N} y_i x_i - \sum_{i=1}^{N} b_0 x_i - \sum_{i=1}^{N} b_1 x_i^2$$

Next, we bring all of the constants in these terms (i.e., b_0 and b_1) out in front of the summation operators, as follows:

$$0 = \sum_{i=1}^{N} y_i x_i - b_0 \sum_{i=1}^{N} x_i - b_1 \sum_{i=1}^{N} x_i^2$$

We then add the last term on the right side of the equation to both sides:

$$b_1 \sum_{i=1}^{N} x_i^2 = \sum_{i=1}^{N} y_i x_i - b_0 \sum_{i=1}^{N} x_i$$

Next, we go back to the value for b_0 from Equation (9) and substitute it into the result we just obtained. This gives us:

$$b_1 \sum_{i=1}^{N} x_i^2 = \sum_{i=1}^{N} y_i x_i - \left[\left(\frac{\sum_{i=1}^{N} y_i}{N} \right) - b_1 \left(\frac{\sum_{i=1}^{N} x_i}{N} \right) \right] \sum_{i=1}^{N} x_i$$

Multiplying out the last term on the right, we get:

$$b_1 \sum_{i=1}^{N} x_i^2 = \sum_{i=1}^{N} y_i x_i - \frac{\sum_{i=1}^{N} y_i \sum_{i=1}^{N} x_i}{N} - b_1 \left(\frac{\left(\sum_{i=1}^{N} x_i \right)^2}{N} \right)$$

If we then add the last term on the right to both sides of the equation, we get:

$$b_1 \sum_{i=1}^{N} x_i^2 + b_1 \left(\frac{\left(\sum_{i=1}^{N} x_i \right)^2}{N} \right) = \sum_{i=1}^{N} y_i x_i - \frac{\sum_{i=1}^{N} y_i \sum_{i=1}^{N} x_i}{N}$$

On the left side of the equation, we can then factor out b_1:

$$b_1 \left[\sum_{i=1}^{N} x_i^2 + \frac{\left(\sum_{i=1}^{N} x_i \right)^2}{N} \right] = \sum_{i=1}^{N} y_i x_i - \frac{\sum_{i=1}^{N} y_i \sum_{i=1}^{N} x_i}{N}$$

Finally, if we divide both sides of the equation by the quantity in the large brackets on the left side, we can isolate b_1 and obtain the least-square estimator for the regression coefficient in the bivariate case. This is the form from lecture:

$$b_1 = \frac{\sum_{i=1}^{N} y_i x_i - \frac{\sum_{i=1}^{N} y_i \sum_{i=1}^{N} x_i}{N}}{\sum_{i=1}^{N} x_i^2 - \frac{\left(\sum_{i=1}^{N} x_i\right)^2}{N}} \tag{11}$$

The expression on the right, as you will recall, is the ratio of the sum of the cross-products of x_i and y_i over the sum of squares for x_i.

The Multiple Regression Case: Deriving OLS with Matrices

The foregoing math is all well and good if you have only one independent variable in your analysis. However, in the social sciences, this will rarely be the case: rather, we will usually be trying to predict a dependent variable using scores from several independent variables. Deriving a more general form of the least-squares estimator for situations like this requires the use of matrix operations. As you will recall from lecture, the basic OLS regression equation can be represented in the following matrix form:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{e} \tag{12}$$

where **Y** is an $N \times 1$ column matrix of cases' scores on the DV, **X** is an $N \times (k+1)$ matrix of cases' scores on the IVs (where the first column is a placeholder column of ones for the constant and the remaining columns correspond to each IV), **B** is a $(k+1) \times 1$ column matrix containing the regression constant and coefficients, and **e** is an $N \times 1$ column matrix of cases' errors of prediction.

As before, what we want to do is find the values for the elements of **B** that minimize the sum of the squared errors. The quantity that we are trying to minimize can be expressed as follows:

$$SSE = \mathbf{e}'\mathbf{e} \tag{13}$$

If you work out the matrix operations for the expression on the right, you'll notice that the result is a scalar – a single number consisting of the sum of the squared errors of prediction (i.e., multiplying a $1 \times N$ matrix by a $N \times 1$ matrix produces a 1×1 matrix, i.e., a scalar). In order to take the derivative of the quantity with regard to the **B** matrix, we first of all need to express **e** in terms of **Y**, **X**, and **B**:

$$\mathbf{e} = \mathbf{Y} - \mathbf{XB}$$

Substituting the expression on the right side into Equation (13), we get:

$$SSE = (\mathbf{Y} - \mathbf{XB})'(\mathbf{Y} - \mathbf{XB})$$

Next, the transposition operator on the first quantity in this product – **(Y - XB)′** – can distributed:[5]

$$SSE = (\mathbf{Y}' - \mathbf{B}'\mathbf{X}')(\mathbf{Y} - \mathbf{XB})$$

When this product is computed, we get the following:

$$SSE = \mathbf{Y}'\mathbf{Y} - \mathbf{Y}'\mathbf{XB} - \mathbf{B}'\mathbf{X}'\mathbf{Y} + \mathbf{B}'\mathbf{X}'\mathbf{XB}$$

Now, if multiplied out, the two middle terms – **Y′XB** and **B′X′Y** -- are identical: they produce the same scalar value. As such, the equation can be further simplified to:

$$SSE = \mathbf{Y}'\mathbf{Y} - 2\mathbf{Y}'\mathbf{XB} + \mathbf{B}'\mathbf{X}'\mathbf{XB} \qquad (14)$$

We now have an equation which expresses SSE in terms of **Y**, **X**, and **B**. The next step – as in the bivariate case – is to take the derivative of SSE with respect to the matrix **B**. Since we're really dealing with a set of variables in this differentiation problem – the constant and one regression coefficient for each IV – we again use the partial derivative operator:

$$\frac{\partial SSE}{\partial \mathbf{B}} = \frac{\partial}{\partial \mathbf{B}}(\mathbf{Y}'\mathbf{Y} - 2\mathbf{Y}'\mathbf{XB} + \mathbf{B}'\mathbf{X}'\mathbf{XB})$$

[5] Remember that for any two matrices **A** and **B** that can be multiplied together, **(AB)′ = B′A′**.

This looks like a complex problem, but it's actually quite similar to taking the derivative of a polynomial in the scalar context. First, since we are treating all matrices besides **B** as the equivalent of constants, the first term in parentheses – based completely on the **Y** matrix – has a derivative of zero.

Second, the middle term – known as a "linear form" in **B** – is the equivalent of a scalar term in which the variable we are differentiating with respect to is raised to the first power (i.e. a *linear* term), which means we obtain the derivative by dropping the B and taking the transpose of all the matrices in the expression which remain, giving us $-2\mathbf{X}'\mathbf{Y}$.

Finally, the third term – known as a "quadratic form" in B – is the equivalent of a scalar term in which the variable we are differentiating with respect to is raised to the second power (i.e., a *quadratic* term). This means we obtain the derivative by dropping the $\mathbf{B}'$ from the term and multiplying by two, giving us $\mathbf{2X}'\mathbf{XB}$. Thus, the full partial derivative is

$$\frac{\partial SSE}{\partial \mathbf{B}} = -2\mathbf{X}'\mathbf{Y} + 2\mathbf{X}'\mathbf{XB} \qquad (15)$$

The next step is to set this partial derivative to zero and solve for the matrix **B**. This will give us an expression for the matrix of estimates that minimize the sum of the squared errors of prediction. We start with the following:

$$0 = -2\mathbf{X}'\mathbf{Y} + 2\mathbf{X}'\mathbf{XB}$$

We then subtract $\mathbf{2X}'\mathbf{XB}$ from each side of the equation:

$$-2\mathbf{X}'\mathbf{XB} = -2\mathbf{X}'\mathbf{Y}$$

Next, we eliminate the -2 on each term by multiplying each side of the equation by -1/2:

$$\mathbf{X}'\mathbf{XB} = \mathbf{X}'\mathbf{Y}$$

Finally, we need to solve for B by pre-multiplying each side of the equation the inverse of $(\mathbf{X}'\mathbf{X})$, i.e., $(\mathbf{X}'\mathbf{X})^{-1}$. Remember that this is the matrix equivalent of dividing each side of the equation by $(\mathbf{X}'\mathbf{X})$:

$$\mathbf{B} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \qquad (16)$$

Equation (16) is, of course, the familiar OLS estimator we discussed in lecture. To tie this back to the bivariate case, note closely what the expression on the right does. While $\mathbf{X}'\mathbf{Y}$ gives the sum of the cross-products of **X** and **Y**, $\mathbf{X}'\mathbf{X}$ gives us the sum of squares for **X**. Since pre-multiplying $\mathbf{X}'\mathbf{Y}$ by $(\mathbf{X}'\mathbf{X})^{-1}$ is the matrix equivalent of *dividing* $\mathbf{X}'\mathbf{Y}$ by $\mathbf{X}'\mathbf{X}$, this expression is basically doing the same thing as the scalar expression for b_1 in Equation (11): dividing the sum of the cross products of the IV (or IVs) and the DV by the sum of squares for the IV (or IVs).