Deriving OLS: a step-by-step Guide

Tom Glinnan

January 2022

This is a guide on how to get to the matrix expression for the OLS estimator:

$$\hat{\beta}_{ols} = (X'X)^{-1}X'y$$

The proof itself is actually quite simple once we're comfortable with matrices, so we'll use this as an excuse to look at them too

The Best Fit Line

OLS has many interpretations, but the most easy is that it finds the 'best fit' line between our data points. Suppose that the x-axis in the figure below is called x_2. From school-level math, we know

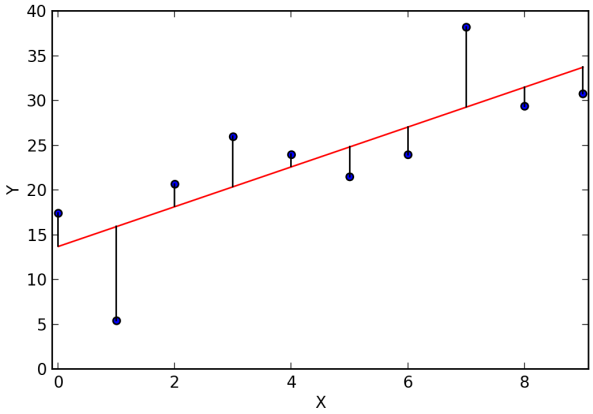

Figure 1: Line of Best Fit, from Wikipedia

that we can express any line in 2-dimensions, such as the red one in the figure, as $y = a_1 + a_2x_2$ for constants a_1 and a_2. Therefore, if we take a data point (x_{2i}, y_i) and project it down to the red line (ie, moving it in the direction of the black line), this point (where the black line meets the red) will be $(x_{2i}, a_1 + a_2x_{2i})$. Just like the different ways of writing the DGP, there are different ways of writing the equations of all of these points - the *scalar way* $y_i = a_1 + a_2x_{2i}$; the *vector way* $y_i = b'x_i$ with

$$x_i = \begin{bmatrix} 1 \\ x_{2i} \end{bmatrix} \qquad \text{and} \qquad b = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$$

And finally stacking together, in the *matrix way*: $y = Xb$. We can define the *residuals* as the black lines shown in the figure: as $\hat{\varepsilon}_i = y_i - x_i'b$ for any b, and so stacking the $\hat{\varepsilon}_i$s into one column gives us $\hat{\varepsilon} = y - Xb$. This makes $\hat{\varepsilon}$ a function of b - for different lines, we will have different residuals, as some of them will do better jobs than others in approximating the data points. For our definition of the 'best' line, we use the sum of the squares of the residuals $\hat{\varepsilon}_i$[1], so we are solving

$$\hat{\beta}_{ols} = \arg\min_b \sum_{i=1}^{n} \hat{\varepsilon}_i^2$$

To emphasise: there is a big difference between *residuals* and *errors*. Errors are defined as $\varepsilon = y - X\beta$, for the 'true'[2] best fit line β. Residuals instead are defined for any line, and we usually talk about the residuals from a regression (such as $y - X\hat{\beta}_{ols}$)

Re-writing the Problem in Matrix Form

As we want to work with matrices, let's re-write our problem. The first thing to notice is that we can write

$$\sum_{i=1}^{n} \hat{\varepsilon}_i^2 = \hat{\varepsilon}'\hat{\varepsilon}$$

Plugging in the definition of $\hat{\varepsilon} = y - Xb$ and expanding gives:

$$\hat{\varepsilon}'\hat{\varepsilon} = (y - Xb)'(y - Xb) = y'y - (Xb)'y - y'Xb + (Xb)'(Xb)$$

Now two small things. Firstly, from the fact that $(AB)' = B'A'$, we can write $(Xb)'(Xb) = b'(X'X)b$. Secondly, look at the size of the matrices. Xb and y are $n \times 1$ column vectors, so both $(Xb)'y$ and $y'Xb$ are real numbers. However, if you transpose $y'Xb$, you get $(Xb)'y$ - so in fact they must be equal to each other, as the transpose of a 1×1 matrix is itself[3]. Using all of this, we can re-write our problem as

$$\hat{\beta}_{ols} = \arg\min_b \quad y'y - 2y'Xb + b'(X'X)b$$

A Reminder: how to maximise

The only thing we've done so far is re-write $\sum_{i=1}^{n} \hat{\varepsilon}_i^2$ using matrices. If we define

$$S(b) = y'y - 2y'Xb + b'(X'X)b$$

then $S : \mathbb{R}^k \to \mathbb{R}$ is a function that takes in some value of b (the parameters of the line) and gives you a real number (the sum of the squared distances from that line to our data, as shown in the picture). Don't get overwhelmed by the matrices: this is just a regular multivariable calculus problem. From MA100, the conditions for a minimum are that

$$\nabla S(\hat{\beta}_{ols}) = 0 \qquad \text{and} \qquad \nabla^2 S(\hat{\beta}_{ols}) \text{ is positive semi-definite}$$

[1] Which is where the 'least squares' in 'ordinary least squares' comes from. Also, in case you haven't seen it before, arg min means 'the thing that minimises'

[2] The β in the DGP of the data

[3] This is just like how $a'x = x'a$, as they both equal $\sum_i a_i x_i$

where ∇S is the gradient (a column vector, made up of stacked partial derivatives), and $\nabla^2 S$ is the Hessian (the matrix made up of second derivatives)

Sidenote: we can either use the notation above, or we can use the alternative notation

$$\nabla S = \frac{\partial S}{\partial b} \qquad \text{and} \qquad \nabla^2 S = \frac{\partial^2 S}{\partial b \, \partial b'}$$

where we sometimes emphasise that b is a vector by making it bold. When you see this, just remember that they are the Gradient and Hessian you've seen before - nothing new! In a similar vein, remember that the rules for doing matrix calculus are nothing new: just a matrix notation for things we know with sums. The three we need are that:

$$\nabla x'a = a \qquad \nabla x'Ax = (A + A')x \qquad \nabla x'A = A$$

Where the derivatives are with respect to x. See if you can check that these rules make sense to you, by taking the simple case that x, a and A are real numbers[4].

First and Second Order Conditions

Looking at the FOC:

$$\nabla S(b) = -2(y'X)' + (X'X + (X'X)')b = -2X'y + 2X'Xb$$

since $X'X$ is symmetric. Don't forget the SOC! To get the Hessian we have to do one small thing we don't have to do when we do calculus in one dimension - while in one dimension the second derivative is just the derivative of the first derivative, the Hessian is the gradient *of the transpose* of the gradient[5]. Of course, $(\nabla S(b))' = -2y'X + 2b'(X'X)$, and so:

$$\nabla^2 S(b) = X'X$$

Interpretation, and getting $\hat{\beta}_{ols}$

Let's start with the SOC. The Hessian is $X'X$, and a standard Linear Algebra theorem is that this is positive-semi definite (psd)[6]. Since this is true *for all* b, any stationary point must be not only a local minimum, but actually a global minimum. The expression we're minimising is convex - it's like a multivariate version of a quadratic

Now back to the FOC. Since $\hat{\beta}_{ols}$ is by definition the minimum, it has gradient 0, and so $\nabla S(\hat{\beta}_{ols}) = -2X'y + 2X'X\hat{\beta}_{ols} = 0$. An easy re-arranging gives

$$(X'X)\,\hat{\beta}_{ols} = X'y$$

Up to this point, we haven't made any assumptions. Now let's finally make one: that $X'X$ has an inverse. Inverting $X'X$ gives us the usual expression $\hat{\beta}_{ols} = (X'X)^{-1}X'y$

[4]This is a special case of what we are trying to do, as a number can be thought of as a 1-dimensional vector, or as a 1×1 matrix

[5]This is the reason why the notation $\frac{\partial^2 S}{\partial b \, \partial b'}$ has a transpose in the bottom

[6]The proof is easy. psd means that for all $a \neq 0$, $a'(X'X)a \geq 0$. This must be true as $a'(X'X)a = (Xa)'(Xa)$. Since Xa is a vector, this is the sum of the squares of that vector's components, which must be non-negative, as all of the numbers are real

Multicollinearity

What does this assumption mean? It helps to remember what X is - it's an $n \times k$ matrix (n rows representing people, k columns representing variables), so $X'X$ is a $k \times k$ matrix. This means that $X'X$ is invertible iff it has rank k. We interpret this by using the following idea

Proposition 1. *$X'X$ has the same rank as X*

Proof. We start by showing that the *null space* of $X'X$ is the same as the one for X. For a matrix M, this is the set of vectors a such that $Ma = 0$. To show one direction: if $Xa = 0$, then multiplying on the left by X' gives $X'Xa = 0$. In the other direction, if $X'Xa = 0$ then multiplying on the left by a' gives $a'X'Xa = 0$, which is $(Xa)'(Xa) = 0$. Since this is the sum of the squared components of Xa, all of the components must be 0, and so $Xa = 0$. Therefore the null spaces are the same

As the null spaces are the same, the dimension of the null spaces (the nullity) is the same. As the rank + the nullity add up to k in both cases[7], the rank of X is the same as the rank of $X'X$ □

So now the question is: what does it mean for X to have rank k? It means that the matrix has full column rank - no variable can be written as a linear combination of the other variables. This is exactly the **no multicollinearity** condition - violated by, for examples, dummy variables for all of the months of the year

What happens if we have multicollinearity? It just means that we can't invert $X'X$, and so the most we can say is that $(X'X)\hat{\beta}_{ols} = X'y$. The reason we can't go any further is because this equation has *multiple solutions* for $\hat{\beta}_{ols}$. This is exactly the intuition we have for multicolinearity!

[7] Formally, this is called the rank-nullity theorem