Applying Matrix Rules to Derive the OLS estimator

To get the OLS estimator $\widehat{\boldsymbol{\beta}}_{\boldsymbol{ols}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}$ we find the parameter vector $\boldsymbol{\beta}$ = [$\beta_1, \beta_2, \dots, \beta_k$] which minimizes the squared error $\sum_{i=1}^{N} u_i{}^2 = \boldsymbol{u}'\boldsymbol{u} = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})$. This is the objective function, $f(\boldsymbol{\beta})$, of our optimization problem. The first order conditions to solve this problem require setting the partial derivatives of f with respect to $\beta_1, \beta_2, \dots, \beta_k$ equal to zero. In vector notation this means

$$\frac{\boldsymbol{df}}{\boldsymbol{d\beta}} \equiv \left[\frac{\partial f}{\partial \beta_1}, \frac{\partial f}{\partial \beta_2}, \dots \frac{\partial f}{\partial \beta_k}\right]' = \mathbf{0} \text{ , where } \mathbf{0} \equiv [0,0, \dots ,0]' \text{ is the zero vector.}$$

Note that according to general convention $\frac{\boldsymbol{df}}{\boldsymbol{d\beta}}$ is a column vector with the partial derivatives arranged in a way that the ith component of $\boldsymbol{\beta}$, β_i, corresponds to the ith component of $\frac{\boldsymbol{df}}{\boldsymbol{d\beta}}$, $\frac{df}{d\beta_i}$. Also, the dimension of $\frac{\boldsymbol{df}}{\boldsymbol{d\beta}}$ is the same as that of $\boldsymbol{\beta}$. This is true, in general, when taking the derivative of a scalar valued function with respect to a vector. Alternatively, if we had a vector valued function that evaluated to an $N \times 1$ vector, differentiating with respect an $M \times 1$ vector would result in an $M \times N$ matrix called the Jacobian (or $N \times M$ according to another convention). Fortunately, this is not a concern for this problem because $f(\boldsymbol{\beta}) = \sum_{i=1}^{N} u_i{}^2$ is a scalar. When this is so, $\frac{\boldsymbol{df}}{\boldsymbol{d\beta}}$ is also called the gradient vector. An alternative notation for the gradient is $\nabla f = [f_1, f_2, \dots, f_k]'$

Before differentiation, it's useful to simplify our objective function. We do so, first, by applying the distributive property of matrices (i.e. $\mathbf{A}(\mathbf{B} + \mathbf{C}) = \mathbf{AB} + \mathbf{AC}$ which is not in general equal to $\boldsymbol{BA} + \boldsymbol{CA}$ because matrix multiplication does NOT obey a commutative property).

$$f(\boldsymbol{\beta}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) = \boldsymbol{y}'\boldsymbol{y} - \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{y} - \boldsymbol{y}'\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{X}\boldsymbol{\beta}$$

Two other matrix rules which were applied for the above manipulation are $(\boldsymbol{A} + \boldsymbol{B})' = \boldsymbol{A}' + \boldsymbol{B}'$ and $(\boldsymbol{AB})' = \boldsymbol{B}'\boldsymbol{A}'$. We can see the term $\boldsymbol{y}'\boldsymbol{y}$ has no $\boldsymbol{\beta}$ in it and so it is not relevant for the optimization problem. Another simplification can be made by observing $\boldsymbol{y}'\boldsymbol{X}\boldsymbol{\beta} = (\boldsymbol{y}'\boldsymbol{X}\boldsymbol{\beta})' = \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{y}$. This is, of course, not true in general of transposes, but follows in this case because this matrix product evaluates to a scalar (and the transpose of a scalar is the same scalar). Now our optimization problem simplifies to.

$$\min_{\boldsymbol{\beta}} -2\boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{y} + \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{X}\boldsymbol{\beta}$$

We can already see the $\boldsymbol{X}'\boldsymbol{y}$ and the $\boldsymbol{X}'\boldsymbol{X}$ from the $\widehat{\boldsymbol{\beta}}_{\boldsymbol{ols}}$ formula in the objective function above. But because the required vector differentiation may still be somewhat involved for the inexperienced, we will show differentiation for each of the above terms separately.

1. Differentiating $\boldsymbol{2\beta' X' y}$ with respect to $\boldsymbol{\beta}$

By letting $g = \boldsymbol{\beta' X' y}$ we can easily see that

$$\frac{\boldsymbol{d2\beta' X' y}}{\boldsymbol{d\beta}} = \begin{bmatrix} \frac{\partial(2g)}{\partial\beta_1} \\ \vdots \\ \frac{\partial(2g)}{\partial\beta_k} \end{bmatrix} = \begin{bmatrix} \frac{2\partial g}{\partial\beta_1} \\ \vdots \\ \frac{2\partial g}{\partial\beta_k} \end{bmatrix} = 2\begin{bmatrix} \frac{\partial g}{\partial\beta_1} \\ \vdots \\ \frac{\partial g}{\partial\beta_k} \end{bmatrix} = 2\frac{\boldsymbol{d\beta' X' y}}{\boldsymbol{d\beta}}$$

You should be able to see that this is a general property of vector differentiation so that $\frac{\partial c.\boldsymbol{A}}{\partial B} = c\frac{\partial.\boldsymbol{A}}{\partial B}$ for any scalar c, and vectors B and $A = A(B)$.

Next it useful to simplify notation by letting the $k \times 1$ vector $\boldsymbol{X'y = s} = [\boldsymbol{s_1, s_2, \dots, s_k}]'$**.** This gives us

$$\boldsymbol{\beta' X' y} = \boldsymbol{\beta' s} = [\beta_1, \dots \beta_k] \begin{bmatrix} s_1 \\ \vdots \\ s_k \end{bmatrix} = \beta_1 s_1 + \beta_2 s_2 + \cdots + \beta_k s_k$$

Now we can easily differentiate by computing the partial derivative components of the gradient

$$\frac{d\boldsymbol{\beta' s}}{d\boldsymbol{\beta}} \equiv \left[\frac{\partial\boldsymbol{\beta' s}}{\partial\beta_1}, \dots, \frac{\partial\boldsymbol{\beta' s}}{\partial\beta_k}\right]' = [\boldsymbol{s_1, s_2, \dots, s_k}]' = \boldsymbol{X' y}$$

Similarly, we can show for any two column vectors of the same dimension that $\frac{\boldsymbol{dA'B}}{\boldsymbol{dB}} = \frac{\boldsymbol{dB'A}}{\boldsymbol{dB}} = \boldsymbol{A}$

Although, not necessary to derive $\boldsymbol{\widehat{\beta}_{ols}}$ we can see that the components of $\boldsymbol{s}$ work out to the sums

$$\boldsymbol{X' y} = \begin{bmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{Nk} \end{bmatrix}' \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} x_{11} & \cdots & x_{N1} \\ \vdots & \ddots & \vdots \\ x_{1k} & \cdots & x_{Nk} \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} x_{11}y_1 + \cdots + x_{N1}y_N \\ \vdots \\ x_{1k}y_1 + \cdots + x_{Nk}y_N \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{N} x_{i1}y_i \\ \vdots \\ \sum_{i=1}^{N} x_{ik}y_i \end{bmatrix} = \begin{bmatrix} s_1 \\ \vdots \\ s_k \end{bmatrix}$$

2. Differentiating $\boldsymbol{\beta' X' X\beta}$ with respect to $\boldsymbol{\beta}$

Again to simplify notation we let the $k \times k$ matrix $\boldsymbol{X'X = M}$ with components defined as

$$\boldsymbol{X'X} = \begin{bmatrix} x_{11} & \cdots & x_{N1} \\ \vdots & \ddots & \vdots \\ x_{1k} & \cdots & x_{Nk} \end{bmatrix} \begin{bmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{Nk} \end{bmatrix}' = \begin{bmatrix} \sum_{i=1}^{N} x_{i1}x_{i1} & \cdots & \sum_{i=1}^{N} x_{i1}x_{ik} \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^{N} x_{ik}x_{i1} & \cdots & \sum_{i=1}^{N} x_{ik}x_{ik} \end{bmatrix} = \begin{bmatrix} m_{11} & \cdots & m_{1k} \\ \vdots & \ddots & \vdots \\ m_{k1} & \cdots & m_{kk} \end{bmatrix}$$

Then $\boldsymbol{\beta' X' X\beta}$ works out to

$$\boldsymbol{\beta' X' X \beta} = [\beta_1, \dots, \beta_k] \begin{bmatrix} m_{11} & \cdots & m_{1k} \\ \vdots & \ddots & \vdots \\ m_{k1} & \cdots & m_{kk} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} = [\beta_1, \dots, \beta_k] \begin{bmatrix} m_{11}\beta_1 + \cdots + m_{1k}\beta_k \\ \vdots \\ m_{k1}\beta_k + \cdots + m_{kk}\beta_k \end{bmatrix}$$

$$= m_{11}{\beta_1}^2 + \beta_1 \sum_{i=1, i\neq 1}^{k} m_{1i}\beta_i$$

$$+ m_{22}{\beta_2}^2 + \beta_2 \sum_{i=1, i\neq 2}^{k} m_{2i}\beta_i$$

$$+ \cdots$$

$$+ \ m_{kk}{\beta_k}^2 + \beta_k \sum_{i=1, i\neq k}^{k} m_{ki}\beta_i$$

Where the notation $\sum_{i=1, i\neq j}^{k} a_i$ means to take the sum of all $a_1, \dots, a_n$ excluding a_j. For clarity, when k = 3

$$\boldsymbol{\beta' X' X \beta} = m_{11}{\beta_1}^2 + \beta_1(m_{12}\beta_2 + m_{13}\beta_3) + m_{22}{\beta_2}^2 + \beta_2(m_{21}\beta_1 + m_{23}\beta_3) + m_{33}{\beta_3}^2 + \beta_3(m_{31}\beta_1 + m_{32}\beta_2)$$

$$= (m_{11}{\beta_1}^2 + m_{12}\beta_1\beta_2 + m_{13}\beta_1\beta_3) + (m_{21}\beta_2\beta_1 + m_{22}{\beta_2}^2 + m_{23}\beta_2\beta_3) + (m_{31}\beta_3\beta_1 + m_{32}\beta_3\beta_2\, m_{33}{\beta_3}^2)$$

The brackets in the last line are present to help you see the pattern in m_{ij} coefficients and β_i, β_j pairs.

Then $\frac{\partial \boldsymbol{\beta}' \boldsymbol{X}' \boldsymbol{X} \boldsymbol{\beta}}{\partial \beta_1} = 2m_{11}\beta_1 + m_{12}\beta_2 + m_{13}\beta_3 + m_{21}\beta_2 + m_{31}\beta_3 = 2(m_{11}\beta_1 + m_{12}\beta_2 + m_{13}\beta_3)$

The last equality follows from the symmetry of $\boldsymbol{X'X} = \boldsymbol{M}$ meaning $m_{12} = m_{21}$ and $m_{13} = m_{31}$.

We can show the same for the remaining partial derivatives and generalize for any k, giving us

$$\frac{\boldsymbol{d\beta' X' X \beta}}{\boldsymbol{d\beta}} = \begin{bmatrix} 2(m_{11}\beta_1 + m_{12}\beta_2 + \cdots + m_{1k}\beta_k) \\ 2(m_{21}\beta_1 + m_{22}\beta_2 + \cdots + m_{2k}\beta_k) \\ \vdots \\ 2(m_{k1}\beta_1 + m_{k2}\beta_2 + \cdots + m_{kk}\beta_k) \end{bmatrix} = \boldsymbol{2M\beta = 2X'X\beta}$$

This rule of vector differentiation can be also generalized for any $N \times 1$ column vector $\boldsymbol{B}$, and $N \times N$ symmetric matrix $\boldsymbol{A}$, so that $\frac{\boldsymbol{dB'AB}}{\boldsymbol{dB}} = \boldsymbol{2AB}$. Matrix products of the form $\boldsymbol{B'AB}$ are called quadratic forms.

3) Putting it all together

From sections 1) and 2) we see that are first order conditions becomes $\frac{\boldsymbol{df}}{\boldsymbol{d\beta}} = \boldsymbol{-2X'y + 2X'X\beta} = \boldsymbol{0}$

Dividing both sides by 2 and rearranging we have $\boldsymbol{X'X\beta} = \boldsymbol{X'y}$. Multiply both sides by $\boldsymbol{(X'X)^{-1}}$ to get

$$\boldsymbol{\beta} = \boldsymbol{(X'X)^{-1}X'y}$$