University of Leicester

DEPARTMENT OF ECONOMICS

On Kronecker Products, Tensor Products And Matrix Differential Calculus

D.S.G. Pollock, University of Leicester, UK

Working Paper No. 11/34
July 2011

ON KRONECKER PRODUCTS, TENSOR PRODUCTS AND MATRIX DIFFERENTIAL CALCULUS

By D.S.G. Pollock

University of Leicester

Email: stephen_pollock@sigmapi.u-net.com

The algebra of the Kronecker products of matrices is recapitulated using a notation that reveals the tensor structures of the matrices. It is claimed that many of the difficulties that are encountered in working with the algebra can be alleviated by paying close attention to the indices that are concealed beneath the conventional matrix notation.

The vectorisation operations and the commutation transformations that are common in multivariate statistical analysis alter the positional relationship of the matrix elements. These elements correspond to numbers that are liable to be stored in contiguous memory cells of a computer, which should remain undisturbed.

It is suggested that, in the absence of an adequate index notation that enables the manipulations to be performed without disturbing the data, even the most clear-headed of computer programmers is liable to perform wholly unnecessary and time-wasting operations that shift data between memory cells.

1. Introduction

One of the the bugbears of multivariate statistical analysis is the need to differentiate complicated likelihood functions in respect of their matrix arguments. To achieve this, one must resort to the theory of matrix differential calculus, which entails the use of Kronecker products, vectorisation operators and commutation matrices.

A brief account of the requisite results was provided by Pollock (1979), who described a theory that employs vectorised matrices. Relationships were established with the classical theory of Dwyer and McPhail (1948) and Dwyer (1967), which dealt with scalar functions of matrix augments and matrix functions of scalar arguments. A contemporaneous account was given by Henderson and Searle (1979), and similar treatments were provided, at roughly the same time, by Balestra (1976) and by Rogers (1980).

A more extensive account of matrix differential calculus, which relies exclusively on vectorised matrices, was provided by the text of Magnus and Neudecker (1988). This has become a standard reference. More recent accounts of matrix differential calculus have been provided by Turkington (2002) and by Harville (2008).

Notwithstanding this ample provision of sources, there continues to be confusion and difficulty in this area. A recent paper of Magnus (2010), which testifies to these problems, cites several instances in which inappropriate definitions of matrix derivatives have been adopted, and it gives examples that highlight the perverse effects of adopting such definitions.

One is bound to wonder why the difficulties persist. An answer to this conundrum, which is offered in the present paper, points to the absence of an accessible account that reveals the tensorial structures that underlie the essential results of matrix differential calculus and of the associated algebra.

Part of the problem lies in the fact that the conventional matrix notation, which has been employed, almost exclusively, to convey the theory of matrix differential calculus, conceals the very things that need to be manipulated, which are matrix elements bearing strings of indices that are ordered according to some permutation of a lexicographic ordering. To overcome the problem, we shall adopt a notation that reveals the indices.

This paper aims to avoid unnecessary abstractions by dealing only with concrete objects, which are matrices and their elements. A more abstract approach would employ the concepts and the notations of vector spaces and of their tensor products. Such an approach to the theory is exemplified by the texts in multilinear algebra of Bourbaki (1958), Greub (1967) and Marcus (1973). A primary source of multilinear algebra and differential geometry is Cartan (1952). One advantage of the approach that we shall take is that it is well adapted to the requirements of computer programming.

2. Bases for Vector Spaces

Consider an identity matrix of order N, which can be written as follows:

$$[\, e_1 \quad e_2 \quad \cdots \quad e_N \,] = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} = \begin{bmatrix} e^1 \\ e^2 \\ \vdots \\ e_N \end{bmatrix}. \tag{1}$$

On the LHS, the matrix is expressed as a collection of column vectors, denoted by $e_i; i = 1, 2, \ldots, N$, which form the basis of an ordinary N-dimensional Euclidean space, which is the primal space. On the RHS, the matrix is expressed as a collection of row vectors $e^j; j = 1, 2, \ldots, N$, which form the basis of the conjugate dual space. In general, vetors bearing a single superscript, including a prime, are to be regarded as row vectors.

The basis vectors can be used in specifying arbitrary vectors in both spaces. In the primal space, there is the column vector

$$a = \sum_i a_i e_i = (a_i e_i), \tag{2}$$

and in the dual space, there is the row vector

$$b' = \sum_j b_j e^j = (b_j e^j). \tag{3}$$

Here, on the RHS, there is a notation that replaces the summation signs by parentheses. When a basis vector is enclosed by parentheses, summations are to be taken in respect of the index or indices that it carries. Usually, such indices will be associated with scalar elements that will also be found within the parentheses. The advantage of this notation will become apparent at a later stage, when the summations are over several indices.

A vector in the primary space can be converted to a vector in the conjugate dual space, and vice versa, by the operation of transposition. Thus, $a' = (a_i e^i)$ is formed from $a = (a_i e_i)$ via the conversion $e_i \to e^i$, whereas $b = (b_j e_j)$ is formed from $b' = (b_j e^j)$ via the conversion $e^j \to e_j$.

3. Elementary Tensor Products

A tensor product of two vectors is an outer product that entails the pairwise products of the elements of both vectors. Consider two primal vectors

$$\begin{aligned} a &= [a_t; t = 1, \ldots T] = [a_1, a_2, \ldots, b_T]' \quad \text{and} \\ b &= [b_j; j = 1, \ldots, M] = [b_1, b_2, \ldots, b_M]', \end{aligned} \tag{4}$$

which need not be of the same order. Then, two kinds of tensor products can be defined. First, there are covariant tensor products. The covariant product of a and b is a column vector in a primal space:

$$a \otimes b = \sum_t \sum_j a_t b_j (e_t \otimes e_j) = (a_t b_j e_{tj}). \tag{5}$$

Here, the elements are arrayed in a long column in an order that is determined by the lexicographic variation of the indices t and j. Thus, the index j undergoes a complete cycle from $j = 1$ to $j = M$ with each increment of the index t in the manner that is familiar from dictionary classifications. Thus

$$a \otimes b = \begin{bmatrix} a_1 b \\ a_2 b \\ \vdots \\ a_T b \end{bmatrix} = [a_1 b_1, \ldots, a_1 b_M, a_2 b_1, \ldots, a_2 b_M, \cdots, a_T b_1, \ldots, a_T b_M]'. \tag{6}$$

A covariant tensor product can also be formed from the row vectors a' and b' of the dual space. Thus, there is

$$a' \otimes b' = \sum_t \sum_j a_t b_j (e^t \otimes e^j) = (a_t b_j e^{tj}). \tag{7}$$

It will be observed that this is just the transpose of $a \otimes b$. That is to say,

$$(a \otimes b)' = a' \otimes b' \quad \text{or, equivalently,} \quad (a_t b_j e_{tj})' = (a_t b_j e^{tj}). \tag{8}$$

The order of the vectors in a covariant tensor product is crucial, since, as one can easily verify, it is the case that

$$a \otimes b \neq b \otimes a \qquad \text{and} \qquad a' \otimes b' \neq b' \otimes a'. \tag{9}$$

The second kind of tensor product of the two vectors is a so-called contravariant tensor product:

$$a \otimes b' = b' \otimes a = \sum_t \sum_j a_t b_j (e_t \otimes e^j) = (a_t b_j e_t^j). \tag{10}$$

This is just the familiar matrix product ab', which can be written variously as

$$\begin{bmatrix} a_1 b' \\ a_2 b' \\ \vdots \\ a_T b' \end{bmatrix} = [\, b_1 a \quad b_2 a \quad \cdots \quad b_M a \,] = \begin{bmatrix} a_1 b_1 & a_1 b_2 & \ldots & a_1 b_M \\ a_2 b_1 & a_2 b_2 & \ldots & a_2 b_M \\ \vdots & \vdots & & \vdots \\ a_T b_1 & a_T b_2 & \ldots & a_T b_M \end{bmatrix}. \tag{11}$$

The ordering of the vectors within such a binary contravariant tensor product is immaterial.

Observe that

$$(a \otimes b')' = a' \otimes b \quad \text{or, equivalently,} \quad (a_t b_j e_t^j)' = (a_t b_j e_j^t). \tag{12}$$

We now propose to dispense with the summation signs and to write the various vectors as follows:

$$a = (a_t e_t), \quad a' = (a_t e^t) \quad \text{and} \quad b = (a_j e_j), \quad b' = (b_j e^j). \tag{13}$$

As before, the convention here is that, when the products are surrounded by parentheses, summations are to be taken in respect of the indices that are associated with the basis vectors.

The convention can be applied to provide summary representations of the products under (5), (7) and (10):

$$a \otimes b' = (a_t e_t) \otimes (b_j e^j) = (a_t b_j e_t^j), \tag{14}$$

$$a' \otimes b' = (a_t e^t) \otimes (b_j e^j) = (a_t b_j e^{tj}), \tag{15}$$

$$a \otimes b = (a_t e_t) \otimes (b_j e_j) = (a_t b_j e_{tj}). \tag{16}$$

Such products are described as decomposable tensors.

4. Non-decomposable Tensor Products

Non-decomposable tensors are the result of taking weighted sums of decomposable tensors. Consider an arbitrary matrix $X = [x_{tj}]$ of order $T \times M$. This can be expressed as the following weighted sum of the contravariant tensor products formed from the basis vectors:

$$X = (x_{tj} e_t^j) = \sum_t \sum_j x_{tj} (e_t \otimes e^j). \tag{17}$$

The indecomposability lies in the fact that the elements x_{tj} cannot be written as the products of an element indexed by t and an element indexed by j.

From $X = (x_{tj}e_t^j)$, the following associated tensors products may be derived:

$$X' = (x_{tj}e_j^t), \tag{18}$$

$$X^r = (x_{tj}e^{tj}), \tag{19}$$

$$X^c = (x_{tj}e_{jt}). \tag{20}$$

Here, X' is the transposed matrix, whereas X^c is a long column vector and X^r is a long row vector. Notice that, in forming X^c and X^r from X, the index that moves assumes a position at the head of the string of indices to which it is joined. It will be observed that whereas the indices of the elements of X^r follow a lexicographic ordering, with the leading index as the principal classifier, those of X^c follow a reverse lexicographic ordering.

The superscript letters c and r denote a pair of so-called vectorisation operators. It has become conventional to denote X^c by vec$X$. Turkington (2000 and 2002) has used the notation devec$X = (\text{vec}X')'$ to denote X^r and to indicate its relationship with vec$X$.

It is evident that

$$X^r = X'^{c\prime} \qquad \text{and} \qquad X^c = X'^{r\prime}. \tag{21}$$

Thus, it can be seen that X^c and X^r are not related to each other by simple transpositions.

The transformation that effects the reversal of the ordering of the two indices, and which thereby reverses the order of the vectors in a covariant tensor product, was described by Pollock (1979) as the tensor commutator. The transformation was denoted by a capital T inscribed in a circle. The copyright symbol is also an appropriate choice of notation, which leads one to write

$$X^{r\prime} = X'^c = \copyright X^c \qquad \text{and} \qquad X^c = \copyright X'^c. \tag{22}$$

It will be shown in Section 7 that the transformation © corresponds to an orthonormal matrix. This was described by Magnus and Neudecker (1979) as the commutation matrix, which they denoted by K.

Example. Consider the equation

$$y_{tj} = \mu + \gamma_t + \delta_j + \varepsilon_{tj} \tag{23}$$

wherein $t = 1, \ldots, T$ and $j = 1, \ldots, M$. This relates to a two-way analysis of variance. For a concrete interpretation, we may imagine that y_{tj} is an observation taken at time t in the jth region. Then, the parameter γ_t represents an effect that is common to all observations taken at time t, whereas the parameter δ_j represents a characteristic of the jth region that prevails through time.

In ordinary matrix notation, the set of TM equations becomes

$$Y = \mu\iota_T\iota_M' + \gamma\iota_M' + \iota_T\delta' + \mathcal{E}, \tag{24}$$

where $Y = [y_{tj}]$ and $\mathcal{E} = [\varepsilon_{tj}]$ are matrices of order $T \times M$, $\gamma = [\gamma_1, \ldots, \gamma_T]'$ and $\delta = [\delta_1, \ldots, \delta_M]'$ are vectors of orders T and M respectively, and ι_T and ι_M are vectors of units whose orders are indicated by their subscripts. In terms of the index notation, the TM equations are represented by

$$(y_{tj} e_t^j) = \mu(e_t^j) + (\gamma_t e_t^j) + (\delta_j e_t^j) + (\varepsilon_{tj} e_t^j). \tag{25}$$

An illustration is provided by the case where $T = M = 3$. Then equations (24) and (25) represent the following structure:

$$\begin{aligned} \begin{bmatrix} y_{11} & y_{12} & y_{13} \\ y_{21} & y_{22} & y_{23} \\ y_{31} & y_{32} & y_{33} \end{bmatrix} &= \mu \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} + \begin{bmatrix} \gamma_1 & \gamma_1 & \gamma_1 \\ \gamma_2 & \gamma_2 & \gamma_2 \\ \gamma_3 & \gamma_3 & \gamma_3 \end{bmatrix} \\ &\quad + \begin{bmatrix} \delta_1 & \delta_2 & \delta_3 \\ \delta_1 & \delta_2 & \delta_3 \\ \delta_1 & \delta_2 & \delta_3 \end{bmatrix} + \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & \varepsilon_{33} \end{bmatrix}. \end{aligned} \tag{26}$$

5. Multiple Tensor Products

The tensor product entails an associative operation that combines matrices or vectors of any order. Let $B = [b_{lj}]$ and $A = [a_{ki}]$ be arbitrary matrices of orders $t \times n$ and $s \times m$ respectively. Then, their tensor product $B \otimes A$, which is also know as a Kronecker product, is defined in terms of the index notation by writing

$$(b_{lj} e_l^j) \otimes (a_{ki} e_k^i) = (b_{lj} a_{ki} e_{lk}^{ji}). \tag{27}$$

Here, e_{lk}^{ji} stands for a matrix of order $st \times mn$ with a unit in the row indexed by lk—the $\{(l-1)s + k\}$th row—and in the column indexed by ji—the $\{(j-1)m + i\}$th column—and with zeros elsewhere.

In the matrix array, the row indices lk follow a lexicographic order, as do the column indices ji. Also, the indices lk are not ordered relative to the indices ji. That is to say,

$$\begin{aligned} e_{lk}^{ji} &= e_l \otimes e_k \otimes e^j \otimes e^i \\ &= e^j \otimes e^i \otimes e_l \otimes e_k \\ &= e^j \otimes e_l \otimes e_k \otimes e^i \\ &= e_l \otimes e^j \otimes e^i \otimes e_k \\ &= e_l \otimes e^j \otimes e_k \otimes e^i \\ &= e^j \otimes e_l \otimes e^i \otimes e_k. \end{aligned} \tag{28}$$

The virtue of the index notation is that it makes no distinction amongst these various products on the RHS—unless a distinction can be found between such expressions as $e^{j\ i}_{\ l\ k}$ and $e_l^{\ j}{}_k^{\ i}$.

For an example, consider the Kronecker of two matrices as follows:

$$\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \otimes \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} b_{11}\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} & b_{12}\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \\ b_{21}\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} & b_{22}\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \end{bmatrix}$$
$$= \begin{bmatrix} b_{11}a_{11} & b_{11}a_{12} & b_{12}a_{11} & b_{12}a_{12} \\ b_{11}a_{21} & b_{11}a_{22} & b_{12}a_{21} & b_{12}a_{22} \\ b_{21}a_{11} & b_{21}a_{12} & b_{22}a_{11} & b_{22}a_{12} \\ b_{21}a_{21} & b_{21}a_{22} & b_{22}a_{21} & b_{22}a_{22} \end{bmatrix}. \tag{29}$$

Here, it can be see that the composite row indices lk, associated with the elements $b_{lj}a_{ki}$, follow the lexicographic sequence $\{11, 12, 21, 22\}$. The column indices follow the same sequence.

6. Compositions

In order to demonstrate the rules of matrix composition, let us consider the matrix equation

$$Y = AXB', \tag{30}$$

which can be construed as a mapping from X to Y. In the index notation, this is written as

$$\begin{aligned}(y_{kl}e_k^l) &= (a_{ki}e_k^i)(x_{ij}e_i^j)(b_{lj}e_j^l) \\ &= (\{a_{ki}x_{ij}b_{lj}\}e_k^l).\end{aligned} \tag{31}$$

Here, there is

$$\{a_{ki}x_{ij}b_{lj}\} = \sum_i \sum_j a_{ki}x_{ij}b_{lj}; \tag{32}$$

which is to say that the braces surrounding the expression on the LHS are to indicate that summations are taken with respect to the repeated indices i and j, which are associated with the basis vectors. The operation of composing two factors depends upon the cancellation of a superscript (column) index, or string of indices, in the leading factor with an equivalent subscript (row) index, or string of indices, in the following factor.

The matrix equation of (30) can be vectorised in a variety of ways. In order to represent the mapping from $X^c = (x_{ij}e_{ji})$ to $Y^c = (y_{kl}e_{lk})$, we may write

$$\begin{aligned}(y_{kl}e_{lk}) &= (\{a_{ki}x_{ij}b_{lj}\}e_{lk}) \\ &= (a_{ki}b_{lj}e_{lk}^{ji})(x_{ij}e_{ji}).\end{aligned} \tag{33}$$

Notice that the product $a_{ki}b_{lj}$ within $(a_{ki}b_{lj}e_{lk}^{ji})$ does not need to be surrounded by braces, since it contains no repeated indices. Nevertheless, there would be no harm in writing $\{a_{ki}b_{lj}\}$.

The matrix $(a_{ki}b_{lj}e_{lk}^{ji})$ is decomposable. That is to say

$$\begin{aligned}(a_{ki}b_{lj}e_{lk}^{ji}) &= (b_{lj}e_l^j) \otimes (a_{ki}e_k^i) \\ &= B \otimes A;\end{aligned} \tag{34}$$

and, therefore, the vectorised form of equation (30) is

$$\begin{aligned} Y^c &= (AXB')^c \\ &= (B \otimes A)X^c. \end{aligned} \tag{35}$$

Other allied results that readily derived from (35) are

$$Y'^c = Y^{r\prime} = (A \otimes B)X'^c, \tag{36}$$

$$Y^{c\prime} = Y'^r = X^{c\prime}(B' \otimes A'), \tag{37}$$

$$Y^r = X^r(A' \otimes B'). \tag{38}$$

Another result, which mat be noted in this context, concerns the trace of a matrix product. If $A = (a_{ij}e_i^j)$ and $B = (b_{ji}e_j^i)$, then

$$\text{Trace}(AB) = \{a_{ij}b_{ji}\} = (a_{ij}e^{ij})(b_{ji}e_{ij}) = A^r B^c. \tag{39}$$

Here, we are invoking the convention that $\{a_{ij}b_{ji}\}$ denotes the sum over the repeated indices i and j.

Example. The equation under (25), which relates to a two-way analysis of variance, can be vectorised to give

$$(y_{tj}e_{jt}) = \mu(e_{jt}) + (e_{jt}^{t})(\gamma_t e_t) + (e_{jt}^{j})(\delta_j e_j) + (\varepsilon_{tj}e_{jt}). \tag{40}$$

Using the notation of the Kronecker product, this can also be rendered as

$$Y^c = \mu(\iota_M \otimes \iota_T) + (\iota_M \otimes I_T)\gamma + (I_M \otimes \iota_T)\delta + \mathcal{E}^c. \tag{41}$$

The latter can also be obtained by applying the rule of (35) to equation (24). The various elements of (24) have been vectorised as follows:

$$\begin{aligned} (\mu\iota_T\iota_M')^c &= (\iota_T\mu\iota_M')^c = (\iota_M \otimes \iota_T)\mu, \\ (\gamma\iota_M')^c &= (I_t\gamma\iota_M')^c = (\iota_M \otimes I_T)\gamma, \\ (\iota_T\delta')^c &= (\iota_T\delta' I_M)^c = (I_M \otimes \iota_T)\delta'^c, \quad \delta'^c = \delta. \end{aligned} \tag{42}$$

Also, there is $(\iota_M \otimes \iota_T)\mu = \mu(\iota_M \otimes \iota_T)$, since μ is a scalar element that can be transposed or freely associated with any factor of the expression.

In comparing (40) and (41), we see, for example, that $(e_{jt}^{t}) = (e_j) \otimes (e_t^t) = \iota_M \otimes I_T$. We recognise that (e_t^t) is the sum over the index t of the matrices of order T which have a unit in the tth diagonal position and zeros elsewhere; and this sum amounts, of course, to the identity matrix of order T.

The vectorised form of equation (26) is

$$\begin{bmatrix} y_{11} \\ y_{21} \\ y_{31} \\ y_{12} \\ y_{22} \\ y_{32} \\ y_{13} \\ y_{23} \\ y_{33} \end{bmatrix} = \left[\begin{array}{c|ccc|ccc} 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ \hline 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ \hline 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right] \left[\begin{array}{c} \mu \\ \hline \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ \hline \delta_1 \\ \delta_2 \\ \delta_3 \end{array}\right] + \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{21} \\ \varepsilon_{31} \\ \varepsilon_{12} \\ \varepsilon_{22} \\ \varepsilon_{32} \\ \varepsilon_{13} \\ \varepsilon_{23} \\ \varepsilon_{33} \end{bmatrix}. \tag{43}$$

7. The Commutation Matrix

The transformation $X'^c = ©X^c$ from $X^c = (x_{tj}e_{jt})$ to $X'^c = (x_{tj}e_{tj})$ is effected by the commutation matrix $© = (e_{tj}^{jt})$. Thus

$$(x_{tj}e_{tj}) = (e_{tj}^{jt})(x_{tj}e_{jt}). \tag{44}$$

It is easy to see that © is an orthonormal matrix with $©' = ©^{-1}$.

Any pair of covariant indices that are found within a multiple tensor product may be commuted. Thus, for example, there is

$$©(A \otimes B)© = (B \otimes A), \tag{45}$$

which is expressed in the index notation by writing

$$(e_{lk}^{kl})(\{a_{ki}b_{lj}\}e_{kl}^{ij})(e_{ij}^{ji}) = (\{b_{lj}a_{ki}\}e_{lk}^{ji}). \tag{46}$$

Here, the four indices may have differing ranges and, therefore, the two commutation matrices of equation (45) may have differing structures and orders. Nevertheless, we have not bothered to make any notational distinction between them. If such a distinction were required, then it might be achieved by attaching to the symbols the indices that are subject to commutations.

Observe also that the braces, which have been applied to the scalar product $\{a_{ki}b_{lj}\}$ in equation (46), are redundant, since the string contains no repeated indices. Their only effect it to enhance the legibility.

Matrices that perform multiple commutations have been defined by Abadir and Magnus (2005) and by Turkington (2002). Such matrices can be composed from elementary commutation matrices, such as those defined above. The effects of such transformations is to permutate the sequence of the indices associated with the bases; and, therefore, they can be described in a straightforward manner using the index notation.

Example. With $t \in [1, T]$ and $j \in [1, M]$, one can write

$$© = \big(e^j \otimes I_T \otimes e_j\big) = [\, I_T \otimes e_1, I_T \otimes e_2, \ldots, I_T \otimes e_M \,] \tag{47}$$

and

$$ \copyright = \big(e_t \otimes I_M \otimes e^t\big) = \begin{bmatrix} I_M \otimes e^1 \\ I_M \otimes e^2 \\ \vdots \\ I_M \otimes e^T \end{bmatrix}. \tag{48} $$

The expressions on the RHS of (47) and (48) partly conceal the indices; and they belie the simple nature of the commutation transformation.

8. Rules for Decomposable Tensor Products

The following rules govern the decomposable tensors product of matrices, which are commonly described as Kronecker products:

$$ \begin{aligned} &\text{(i)} && (A \otimes B)(C \otimes D) = AC \otimes BD, \\ &\text{(ii)} && (A \otimes B)' = A' \otimes B', \\ &\text{(iii)} && A \otimes (B + C) = (A \otimes B) + (A \otimes C), \\ &\text{(iv)} && \lambda(A \otimes B) = \lambda A \otimes B = A \otimes \lambda B, \\ &\text{(v)} && (A \otimes B)^{-1} = (A^{-1} \otimes B^{-1}). \end{aligned} \tag{49} $$

The Kronecker product is non-commutative, which is to say that $A \otimes B \neq B \otimes A$. However, observe that

$$ A \otimes B = (A \otimes I)(I \otimes B) = (I \otimes B)(A \otimes I). \tag{50} $$

This result follows immediately from (49, i).

Such lists of the rules of the Kronecker algebra have been provided in numerous textbooks of econometrics. Examples are the books of Theil (1971) and of Greene (2000) where, in both cases, the results are stated without proofs and without references. One is bound to wonder where the primary sources are to be found. Although the origins of the Kronecker product has been partially uncovered by Henderson *et al.* (1983), its archaeology awaits a full exploration.

To establish the rules of (49) by examples is laborious. However, they can be established readily using the index notation. Consider (49, i), for example, and let

$$ A = (a_{ki} e_k^i), \quad B = (b_{jl} e_j^l), \quad C = (c_{ir} e_i^r), \quad D = (d_{ls} e_l^s). \tag{51} $$

Then,

$$ \begin{aligned} (A \otimes B)(C \otimes D) &= (a_{ki} b_{jl} e_{kj}^{il})(c_{ir} d_{ls} e_{il}^{rs}) \\ &= (\{a_{ki} c_{ir}\} e_k^r) \otimes (\{b_{jl} d_{ls}\} e_j^s) \\ &= AC \otimes BD. \end{aligned} \tag{52} $$

9. Generalised Vectorisation Operations

Turkington (2000 and 2002) has described two matrix operators, called the vec_n and the devec_m operators, which are generalisations, respectively, of the vec operator, which converts a matrix to a long column vector, and of the devec operator, which converts a matrix to a long row vector.

The effect of the vec_n operator is to convert an $m \times np$ partitioned matrix $A = [A_1, A_2, \ldots, A_p]$ to a $pm \times n$ matrix in which the submatrix A_i stands on the shoulders of the submatrix A_{i+1}. The devec_m operator is the inverse of the vec_n operator; and its effect would be to convert the vertical stack of matrices into the horizontal array.

To reveal the nature of the vec_n and devec_m operators, let us consider a matrix $A = [a_{ijk}]$, where $i \in [1, m]$, $j \in [1, n]$ and $k \in [1, p]$. If i is a row index and k, j are (lexicographically ordered) column indices, then the matrix, which is of order $m \times np$, has a tensor structure that is revealed by writing

$$A = (a_{ijk} e_i^{kj}). \tag{53}$$

The effect of the vec_n operator is to convert the index k from a column index to a row index. Thus

$$\text{vec}_n A = (a_{ijk} e_i^{kj})^{\tau} = (a_{ijk} e_{ki}^{j}). \tag{54}$$

Here, the superscripted τ is an alternative notation for the operator.

The inverse of the vec_n operator is the devec_m operator. Its effect is revealed by writing

$$\text{devec}_m\{(\text{vec}_n(A)\} = (a_{ijk} e_{ki}^{j})^{\bar{\tau}} = (a_{ijk} e_i^{kj}) = A. \tag{55}$$

Here, the superscripted $\bar{\tau}$ is the alternative notation for the operator.

Example. Turkington has provided numerous examples to illustrate the generalised vectorisation operators. Amongst them is the result that

$$\text{vec}_s[A(E \otimes D)] = (I_q \otimes A)(\text{vec}E \otimes D). \tag{56}$$

The matrices here are

$$\begin{aligned} A &= (a_{ijk} e_i^{kj}) \quad i \in [1, m], \quad j \in [1, n], \quad k \in [1, p], \\ D &= (d_{jf} e_j^{f}) \quad\ \ j \in [1, n], \quad f \in [1, s], \\ E &= (\varepsilon_{kg} e_k^{g}) \quad\ \ k \in [1, p], \quad g \in [1, q]. \end{aligned} \tag{57}$$

On the LHS of (56), there is

$$\begin{aligned} \text{vec}_s[A(E \otimes D)] &= \Big[(a_{ijk} e_i^{kj})(\varepsilon_{kg} d_{jf} e_{kj}^{gf})\Big]^{\tau} \\ &= (\{a_{ijk} \varepsilon_{kg} d_{jf}\} e_i^{gf})^{\tau}. \end{aligned} \tag{58}$$

On the RHS of (56), there is

$$\begin{aligned}(I_q \otimes A)(\text{vec}E \otimes D) &= (a_{ijk} e_{gi}^{gkj})(\varepsilon_{kg} d_{jf} e_{gkj}^{f}) \\ &= (\{a_{ijk}\varepsilon_{kg} d_{jf}\} e_{gi}^{f}).\end{aligned} \tag{59}$$

Here, the matrix $I_q = (e_g^g)$, with $g \in [1, q]$, is embedded within the first parentheses on the RHS without the accompaniment of any scalar terms. (Observe that one might also write $I_q = (\delta_{g\gamma} e_g^{\gamma})$, where $\delta_{g\gamma}$ is Kronecker's delta.) The equality of (56) follows immediately.

10. The Concept of a Matrix Derivative

Despite the insistence of the majority of the authors who have been cited in the introduction, who have favoured the so-called vectorial definition of the matrix function $Y = Y(X)$ with respect to its matrix argument X, much use continues to be made of alternative definitions. This circumstance has been noted recently by Magnus (2010), who has demonstrated various deleterious consequences of adopting the alternative non-vectorial definitions.

Amongst the alternative definitions that may be encountered is one that can be specified by writing

$$\left[\frac{\partial y_{kl}}{\partial X}\right] = \left(\frac{\partial y_{kl}}{\partial x_{ij}} e_{ki}^{lj}\right). \tag{60}$$

This matrix has the same structure as the product $Y \otimes X = (y_{kl} x_{ij} e_{ki}^{lj})$, which provides a reasonable recommendation for its use.

A principal reason for using the algebra of Kronecker products and the associated vectorisation operator in multivariate statistical analysis is to allow relationships that are naturally expressed in terms of matrices to be cast into the formats of vector equations. This is to enable the statistical techniques that are appropriate to vector equations to be applied to the matrix equations. To this end, the definition of a matrix derivative should conform to the same algebra as the derivative of a vector function of a vector.

It is commonly agreed that the derivative of the vector function $y = Ax$ is respect of the vector x should be the matrix $\partial y/\partial x = A$, and that of the scalar function $q = x'Ax$ should be $\partial q/\partial x = 2x'A$. If analogous definitions are to be applied to matrix equations, then there must be a consistent rule of vectorisation.

Given the pre-existing definition of the Kronecker product of two matrices, which depends upon the lexicographic ordering of the indices, the scope for alternative methods of vectorisation is strictly limited. The equation $Y = AXB'$ can be vectorised usefully in only two ways, which are represented by equations (35) and (36):

$$Y^c = (AXB')^c = (B \otimes A)X^c \quad \text{and} \tag{61}$$

$$Y^{\gamma} = (AXB')^{\gamma} = (A \otimes B)X^{\gamma}, \quad \text{where} \quad Y^{\gamma} = Y'^c. \tag{62}$$

Then, there are $\partial Y^c/\partial X^c = B \otimes A$ and $\partial Y^\gamma/\partial X^\gamma = A \otimes B$. Both definitions are viable, and both can lead to a fully-fledged theory of matrix differential calculus.

The advantage of (62) over (61) is that the elements of Y are arrayed within the long column vector Y^γ according to the lexicographic ordering of their two indices, which is how the elements of Y are liable to be stored within contiguous cells of a computer's memory.

The elements within the long column vector Y^c follow a reversed lexicographic ordering, which is mildly inconvenient. However, there are also some minor advantages that can be attributed to (61), which represents the conventional method of vectorisation. Thus, the canonical derivative of the matrix function $Y = Y(X)$ with respect to matrix argument X is the vectorial derivative

$$\frac{\partial Y^c}{\partial X^c} = \left(\frac{\partial y_{kl}}{\partial x_{ij}} e_{lk}^{ji} \right). \tag{63}$$

11. Conclusions

Once a multivariate statistical model has been cast in a vectorised format, there remains the task of estimating its parameters. This is commonly approached via the method of maximum-likelihood estimation or via some related method that requires the optimisation of a criterion function. At this stage, it becomes crucial to have a workable theory of matrix differential calculus.

Typically, it is required to differentiate quadratic functions, traces and determinants in respect of matrix arguments. The essential methods of matrix differentiation also comprise product rules and chain rules. The necessary results have been codified in the sources that have been cited in the introduction. Most of these are readily accessible; and it is unnecessary to repeat the results here.

The optimisation of the criterion function leads directly to the estimating equations, of which the solutions are the estimates of the parameters of the model. Typically, a recursive method, implemented in computer code, is required in order to obtain the solutions.

It is in devising an efficient computer code for solving the estimating equations that many of the difficulties of multivariate models can arise. The estimating equations usually entail the Kronecker products of matrices together with commutation transformations and vectorisation operators. In interpreting these, programmers often perform wholly unnecessary and time-wasting manipulations that shift the data between memory cells.

A basic nostrum computer programming is that once the data elements have been assigned to the memory cells, these should never disturbed. The same applies to any products of the data that have been calculated and stored in memory. It follows that the operations of transposition commutation and vectorisation that abound in matrix differential calculus should be performed wholly within the realm of a pointer arithmetic that accompanies the programming and that serves to provide access to the data objects.

It is in connnection with such a pointer arithmetic that the advantages of the index notation that has been presented in this paper come to the fore. Indeed, it must be true that any programmer who has succeeded in implementing the multivariate estimation methods efficiently in a programnning language such as C or C++ will have availed themselves of something resembling the algebra of the index notation. The principal contention of this paper is that the index notation can also be employed profitably in deriving the essential results of the algebra of Kronecker products.

References

Abadir, K.M., and J.R. Magnus, (2005), *Matrix Algebra,* Cambridge University Press, Cambridge.

Balestra, P., (1976), *La Dérivation Matricielle,* Éditions Sirey, Paris.

Bourbaki, N, (1958), Algèbre multilinéaire, in *Eléments de Mathematique,* Book II (Algèbre), Herman, Paris, Chapter 3.

Cartan, E., (1952), *Géometrie des Espaces de Riemann,* Gauthier–Villars, Paris.

Dwyer, P.S., (1967), Some Applications of Matrix Derivatives in Multivariate Analysis, *Journal of the American Statistical Association,* 62, 607–625.

Dwyer, P.S., and M.S. MacPhail, (1948), Symbolic Matrix Derivatives, *Annals of Mathematical Statistics,* 19, 517–534.

Graham, A., (1981), *Kronecker Products and Matrix Calculus with Applications,* Ellis Horwood, Chichester.

Greene, W.H., (2000), *Econometric Analysis: Fourth Edition,* Prentice-Hall, New Jersey.

Greub, W.H., (1967), *Multilinear Algebra,* Springer-Verlag, Berlin.

Harville, D.A., (2008), *Matrix Algebra from a Statistician's Perspective,* Springer-Verlag, New York.

Henderson, H.V., and S.R. Searle, (1979), Vec and Vech Operators for Matrices, with Some Uses in Jacobians and Multivariate Statistics, *The Canadian Journal of Statistics,* 7, 65–81.

Henderson, H.V., F. Pukelsheim and S.R. Searle, (1983), On the history of the Kronecker Product, *Linear and Multilinear Algebra,* 14, 13–120.

Magnus, J.R., (2010), On the Concept of Matrix Derivative, *Journal of Multivariate Analysis,* 101, 2200–2206.

Magnus, J.R., and H. Neudecker, (1979), The Commutation Matrix: Some Properties and Applications, *Annals of Statistics,* 7, 381–394.

Magnus, J.R., and H. Neudecker, (1988), *Matrix Differential Calculus with Applications in Statistics and Econometrics,* John Wiley and Sons, Chichester.

Marcus, M., (1973), *Finite Dimensional Multilinear Algebra: Part I,* Marcel Dekker, New York.

Pollock, D.S.G., (1979), *The Algebra of Econometrics,* John Wiley and Sons, Chichester.

Rogers, G.S., (1980), *Matrix Derivatives,* Marcel Dekker, New York.

Theil, H., (1971), *Principles of Econometrics,* John Wiley and Sons, New York.

Turkington, D.A., (2000), Generalised vec Operators and the Seemingly Unrelated Regression Equations Model with Vector Correlated Disturbances, *Journal of Econometrics,* 99, 225–253.

Turkington, D.A., (2002), *Matrix Calculus and Zero-One Matrices,* Cambridge University Press, Cambridge.