ST 437/537: Applied Multivariate and Longitudinal Data Analysis

Vectors and Matrices Using R

Arnab Maity
NCSU Department of Statistics
SAS Hall 5240   919-515-1937   amaity[at]ncsu.edu

Reference: **Multivariate Statistics with R (https://www.researchgate.net/publication/265406106_Multivariate_Statistics_with_R)** by Paul J. Hewson

Vectors

A vector is an **array of numbers**. Specifically, we will write

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix}$$

and call it a *column vector*. We often write $\boldsymbol{x} \in \mathrm{R}^p$. Similarly, a *row vector* is written as

$$\boldsymbol{x}^T = (x_1, x_2, \ldots, x_p).$$

Note that the notation $\boldsymbol{x}^T$ denotes ``transpose” of $\boldsymbol{x}$

> **Note:** In this course, we will always take a vector as a column vector by convention, and will always use the transpose to denote a row vector. Thus the statement “$\boldsymbol{a}$ is a vector” will imply that “$\boldsymbol{a}$ is a *column* vector.”

Example: Consider the following data table (available in R) providing quarterly earnings (first two quarters) in US dollars per Johnson & Johnson share during 1960 - 1964 (available in R).

```
##      Qtr1 Qtr2 Qtr3 Qtr4
## 1960 0.71 0.63 0.85 0.44
## 1961 0.61 0.69 0.92 0.55
## 1962 0.72 0.77 0.92 0.60
## 1963 0.83 0.80 1.00 0.77
## 1964 0.92 1.00 1.24 1.00
```

Now consider the earnings of quarter 1 over the years 1960 - 1964. This is an example of a 5×1 (column) vector

$$\boldsymbol{x} = \begin{pmatrix} 0.71 \\ 0.61 \\ 0.72 \\ 0.83 \\ 0.92 \end{pmatrix}$$

To create this vector in R, we can use the command:

```
x = c(0.71,  0.61, 0.72, 0.83, 0.92)
x
```

```
## [1] 0.71 0.61 0.72 0.83 0.92
```

`R` prints it using a single line but it still considers `x` as a column vector. To see this, try to view `x` in a matrix form using `as.matrix()`:

```
as.matrix(x)
```

```
##      [,1]
## [1,] 0.71
## [2,] 0.61
## [3,] 0.72
## [4,] 0.83
## [5,] 0.92
```

Try to take the transpose using `t()`:

```
t(x)
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,] 0.71 0.61 0.72 0.83 0.92
```

Addition and subtraction of two vectors

For two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathrm{R}^p$, the sum is defined as

$$\boldsymbol{a} + \boldsymbol{b} = \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \\ \vdots \\ a_p + b_p \end{pmatrix},$$

that is, a vector of same length as of $\boldsymbol{a}$ and $\boldsymbol{b}$, where each element is the sum of corresponding elements of $\boldsymbol{a}$ and $\boldsymbol{b}$. We can easily define $\boldsymbol{a} - \boldsymbol{b}$ in a similar fashion.

Example: Consider the earnings of quarters 1 and 2 over the years 1960 - 1964 in the Johnson & Johnson data set.

```
a = c(0.71,  0.61, 0.72, 0.83, 0.92)
b = c(0.63,  0.69, 0.77, 0.80, 1.00)
```

Total earnings over the first 2 quarters is

```
a + b
```

```
## [1] 1.34 1.30 1.49 1.63 1.92
```

Difference between earnings between first 2 quarters is

```
a - b
```

```
## [1]  0.08 -0.08 -0.05  0.03 -0.08
```

Vector multiplication

Here we define the *inner product* of two vectors. For two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathrm{R}^p$, the inner product is defined as:

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle = \boldsymbol{a}^T \boldsymbol{b} = \begin{pmatrix} a_1 & a_2 & \cdots & a_p \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_p \end{pmatrix} = a_1 b_1 + a_2 b_2 + \ldots + a_p b_p = \sum_{j=1}^{p} a_j b_j.$$

The result is a scalar.

Example: Suppose $\boldsymbol{a}^T = (1, 0, 2, 5)$ and $\boldsymbol{b} = \begin{pmatrix} 2 \\ 3 \\ 1 \\ 6 \end{pmatrix}$. Then we have

$$\boldsymbol{a}^T\boldsymbol{b} = \begin{pmatrix} 1 & 0 & 2 & 5 \end{pmatrix} \times \begin{pmatrix} 2 \\ 3 \\ 1 \\ 6 \end{pmatrix} = (1 \times 2) + (0 \times 3) + (2 \times 1) + (5 \times 6) = 34$$

In R, we can use the `%*%` operator to compute the inner product (or matrix multiplication in general). In this example

```
a <- c(1, 0, 2, 5)
b <- c(2, 3, 1, 6)

t(a) %*% b
```

```
##      [,1]
## [1,]   34
```

> **Note:** Be careful to use `%*%`. Be sure to put the `%` signs properly. Just using `*` without the `%` signs would give you elementwise product; in matrix algebra this is referred to as Hadamard product.
>
> ```
> a <- c(1, 0, 2, 5)
> b <- c(2, 3, 1, 6)
>
> t(a)*b
> ```
>
> ```
> ##      [,1] [,2] [,3] [,4]
> ## [1,]    2    0    2   30
> ```

Norm/Length of a Vector

The *length* of a vector is defined as its distance from the vector $\mathbf{0}$, the origin. It is defined as

$$||\boldsymbol{x}|| = \langle \boldsymbol{x}, \boldsymbol{x} \rangle^{1/2} = \left(x_1^2 + \ldots + x_p^2\right)^{1/2}.$$

In other words, the length of a vector $\boldsymbol{x}$ is the square root of the inner product of $\boldsymbol{x}$ with itself.

Try to compute length of `x` defined before:

```
sqrt(sum(x^2))
```

```
## [1] 1.711695
```

Orthogonal vectors

Two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ (that have the same number of elements) are said to be *orthogonal* if $\boldsymbol{a}^T\boldsymbol{b} = 0$. In other words, two vectors are orthogonal if their inner product is zero.

Two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ are *orthonormal* if their inner product is zero and each of them has length 1, that is, if $\boldsymbol{a}^T\boldsymbol{b} = 0$, and $\boldsymbol{a}^T\boldsymbol{a} = 1$ and $\boldsymbol{b}^T\boldsymbol{b} = 1$ then they are orthonormal.

Example: Recall the vectors `a` and `b` defined before. Are they orthogonal? Are they orthonormal?

Matrices

Matrices are array of numbers. For example, consider Johnson & Johnson data set as before:

```
##      Qtr1 Qtr2
## 1960 0.71 0.63
## 1961 0.61 0.69
## 1962 0.72 0.77
## 1963 0.83 0.80
## 1964 0.92 1.00
```

This is an example of a 5×2 matrix, where each column corresponds to a quarter and each row corresponds to a particular year. We can write this matrix as

$$\mathbf{M} = \begin{pmatrix} 0.71 & 0.63 \\ 0.61 & 0.69 \\ 0.72 & 0.77 \\ 0.83 & 0.80 \\ 0.92 & 1.00 \end{pmatrix}.$$

The size of the matrix M is 5×2 as it has 5 rows and 2 columns. In general, a matrix can have any number of rows and columns.

To create the matrix $\mathbf{M}$ in R, and then to print, we use the following commands:

```
M = cbind(c(0.71,  0.61, 0.72, 0.83, 0.92),
          c(0.63,  0.69, 0.77, 0.80, 1.00))
M
```

```
##      [,1] [,2]
## [1,] 0.71 0.63
## [2,] 0.61 0.69
## [3,] 0.72 0.77
## [4,] 0.83 0.80
## [5,] 0.92 1.00
```

This way of creating matrix is essentially taking each column and then joining them together.

> The command `cbind()` concatinates vectors side by side as columns. Use `rbind()` to concatinate by row.
>
> One could also try the command `matrix()`. By default this command fills the matrix by columns. One could try to fill the matrix by rows by including the argument `byrow = TRUE` in the call to `matrix()`.

```
mydata = c(0.71,  0.61, 0.72, 0.83, 0.92, 0.63,  0.69, 0.77, 0.80, 1.00)
M = matrix(mydata, nrow=5, ncol=2, byrow=F)
M
```

```
##      [,1] [,2]
## [1,] 0.71 0.63
## [2,] 0.61 0.69
## [3,] 0.72 0.77
## [4,] 0.83 0.80
## [5,] 0.92 1.00
```

One could also read the matrix into R from an external file:

```
M = read.table(file="mydata.txt", header=FALSE)
```

where `mydata.txt` is an external file containing the values of the matrix with no column names (and hence `header=FALSE`). If column names are included in the file on top of each column, then use `header=TRUE` in the argument.

Transpose

Transposing matrices involves turning the first column into the first row, second column into second row and so on. We write $\mathbf{M}^T$ as the transpose of $\mathbf{M}$.

We can use `t()` to take a transpose in R:

```
Mt = t(M)
Mt
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,] 0.71 0.61 0.72 0.83 0.92
## [2,] 0.63 0.69 0.77 0.80 1.00
```

The dimensions of any matrix can be checked with `dim()`.

```
dim(M)
```

```
## [1] 5 2
```

```
dim(Mt)
```

```
## [1] 2 5
```

We can also access certain elements of the matrix. For example $\mathbf{M}_{12}$ denotes the element of $\mathbf{M}$ which is in the 1st row and 2nd column of the matrix:

```
M[1,2]
```

```
## [1] 0.63
```

Addition and subtraction

Addition and subtraction of matrices can be done if the matrices have the same size. The sum of two matrices `A` and `B` (of same size) is another matrix (of the same size) where each element is the sum of the corresponding elements of `A` and `B`.

```
A = cbind(c(0.71,  0.61, 0.72, 0.83, 0.92),
          c(0.63,  0.69, 0.77, 0.80, 1.00))
A
```

```
##      [,1] [,2]
## [1,] 0.71 0.63
## [2,] 0.61 0.69
## [3,] 0.72 0.77
## [4,] 0.83 0.80
## [5,] 0.92 1.00
```

```
B = matrix(c(1,2,3,4,5,6,7,8,9,10),5,2)
B
```

```
##      [,1] [,2]
## [1,]    1    6
## [2,]    2    7
## [3,]    3    8
## [4,]    4    9
## [5,]    5   10
```

```
# Summing two matrices
A + B
```

```
##      [,1]  [,2]
## [1,] 1.71  6.63
## [2,] 2.61  7.69
## [3,] 3.72  8.77
## [4,] 4.83  9.80
## [5,] 5.92 11.00
```

```
# Subtracting
A - B
```

```
##       [,1]  [,2]
## [1,] -0.29 -5.37
## [2,] -1.39 -6.31
## [3,] -2.28 -7.23
## [4,] -3.17 -8.20
## [5,] -4.08 -9.00
```

Two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are equal, that is, $\boldsymbol{A} = \boldsymbol{B}$ if any only if:

1. $\boldsymbol{A}$ and $\boldsymbol{B}$ have the same size, and
2. the (i, j)-th element of $\boldsymbol{A}$ is equal to the ijth element of $\boldsymbol{A}$ for all $1 \leq i \leq r$ and $1 \leq j \leq n$.

The following two zero matrices are not equal:

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \neq \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Matrix addition satisfies the usual commutative and associative laws.

$$\text{Commutative law: } \mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$$
$$\text{Associative law: } \mathbf{A} + (\mathbf{B} + \mathbf{C}) = (\mathbf{A} + \mathbf{B}) + \mathbf{C}$$

Multiplication

Multiplication of a matrix by a scalar: multiply every element in the matrix by the scalar.

So if $k = 0.4$, and

$$\mathbf{A} = \begin{pmatrix} 1 & 5 & 8 \\ 1 & 2 & 3 \end{pmatrix},$$

we can calculate $k\mathbf{A}$ as:

$$k\mathbf{A} = 0.4 \times \begin{pmatrix} 1 & 5 & 8 \\ 1 & 2 & 3 \end{pmatrix} = \begin{pmatrix} 0.4 & 2 & 3.2 \\ 0.4 & 0.8 & 1.6 \end{pmatrix}.$$

Matrix multiplication however follows vector multiplication, and therefore does not follow the same rules as basic multiplication.

To multiply two matrices `A` and `B`, one must first check that the number of columns in `A` is exactly the same as the number of rows in `B`. Otherwise, we can not multiply these two matrices. More generally,

$$\boldsymbol{A}_{\boldsymbol{m} \times \boldsymbol{n}} \times \boldsymbol{B}_{\boldsymbol{n} \times \boldsymbol{p}} = \boldsymbol{C}_{\boldsymbol{m} \times \boldsymbol{p}}.$$

Let be of size $m \times n$; represent using its row vectors $\boldsymbol{a_1^T}, \boldsymbol{a_2^T}, \ldots, \boldsymbol{a_m^T}$. Let be of size $n \times p$; represent using its columns vectors $\boldsymbol{b_1}, \boldsymbol{b_2}, \ldots, \boldsymbol{b_p}$. The multiplication operation for matrices is defined as:

$$\mathbf{AB} = \begin{pmatrix} a_1^T \\ a_2^T \\ \ldots \\ a_m^T \end{pmatrix} \begin{pmatrix} b_1 & b_2 & \ldots & b_p \end{pmatrix} = \begin{pmatrix} a_1^T b_1 & a_1^T b_2 & \ldots & a_1^T b_p \\ a_2^T b_1 & a_2^T b_2 & \ldots & a_2^T b_p \\ \vdots & \vdots & & \vdots \\ a_m^T b_1 & a_m^T b_2 & \ldots & a_m^T b_p \end{pmatrix}$$

Thus, (i, j)-th element of $\mathbf{AB}$ is the inner product of i-th row of $\boldsymbol{A}$ and j-th column of $\boldsymbol{B}$. Consider the following example.

```
A = cbind(c(0.71,  0.61, 0.72, 0.83, 0.92),
          c(0.63,  0.69, 0.77, 0.80, 1.00))
A
```

```
##      [,1] [,2]
## [1,] 0.71 0.63
## [2,] 0.61 0.69
## [3,] 0.72 0.77
## [4,] 0.83 0.80
## [5,] 0.92 1.00
```

```
B = matrix(c(1,2,3,4,5,6,7,8,9,10),2,5)
B
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    3    5    7    9
## [2,]    2    4    6    8   10
```

Here `A` has 2 columns and `B` has two rows, and hence we can multiply `A` with `B`. In R, we only need to use the `%*%` operator to ensure we are getting matrix multiplication:

```
C = A %*% B
C
```

```
##      [,1] [,2]  [,3]  [,4]  [,5]
## [1,] 1.97 4.65  7.33 10.01 12.69
## [2,] 1.99 4.59  7.19  9.79 12.39
## [3,] 2.26 5.24  8.22 11.20 14.18
## [4,] 2.43 5.69  8.95 12.21 15.47
## [5,] 2.92 6.76 10.60 14.44 18.28
```

Just to check, look at $\boldsymbol{C}_{23}$, the $(2,3)$-th element of $\boldsymbol{C}$.

$$\boldsymbol{C}_{23} = 7.19 = (0.61, 0.69)\begin{pmatrix} 5 \\ 6 \end{pmatrix} = (5 \times 0.61) + (6 \times 0.69) = 7.19.$$

You will get an error message if you multiply non-conformable matrices.

```
B %*% A
```

```
##       [,1]  [,2]
## [1,] 20.23 21.15
## [2,] 24.02 25.04
```

Unlike addition, matrix multiplication is not commutative:

$$\text{(non-commutative) } \mathbf{AB} \neq \mathbf{BA}$$
$$\text{Associative law } \mathbf{A}(\mathbf{BC}) = (\mathbf{AB})\mathbf{C}$$

The distributive laws of multiplication over addition still apply.

$$\mathbf{A}(\mathbf{B}+\mathbf{C}) = \mathbf{AB}+\mathbf{AC}$$
$$(\mathbf{A}+\mathbf{B})\mathbf{C} = \mathbf{AC}+\mathbf{BC}$$

We have the following rules for transposes.

$$\begin{aligned}(\mathbf{A}+\mathbf{B})^T &= \mathbf{A}^T+\mathbf{B}^T\\ (\mathbf{A}\mathbf{B})^T &= \mathbf{B}^T\mathbf{A}^T\end{aligned}$$

Some special matrices

There are some matrices which have particular structure or properties of interest. We will use the following matrices often in this course.

Identity Matrix: An identity matrix (of any size), is a diagonal matrix with 1 as each diagonal entry. For example, $\mathbf{I_3}$ is defined as

$$\mathbf{I_3} = \begin{pmatrix} 1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}.$$

```
diag(3)
```

```
##      [,1] [,2] [,3]
## [1,]    1    0    0
## [2,]    0    1    0
## [3,]    0    0    1
```

Ones: We also need to define a vector of ones; $\mathbf{1}_p$, a $p \times 1$ matrix containing only the value 1. There is no inbuilt function in {R} to create this vector, it is easily added:

```
ones <- rep(1, 3)
ones
```

```
## [1] 1 1 1
```

Zero matrix: Finally, $\mathbf{0}$ denotes the zero matrix, a matrix of zeros. Unlike the previously mentioned matrices this matrix can be any shape you want. So, for example:

$$\mathbf{0_{2\times 3}} = \begin{pmatrix} 0 & 0 & 0\\ 0 & 0 & 0\end{pmatrix}$$

```
matrix(0, nrow=2,ncol=3)
```

```
##      [,1] [,2] [,3]
## [1,]    0    0    0
## [2,]    0    0    0
```

Diagonal Matrices: A diagonal matrix is a square matrix in which all the ``off diagonal" elements are zero. An example of diagonal matrix is

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix}.$$

```
diag(c(1:3))
```

```
##      [,1] [,2] [,3]
## [1,]    1    0    0
## [2,]    0    2    0
## [3,]    0    0    3
```

Symmetric matrices: A matrix A is called a matrix if $A_{ij} = A_{ji}$, that is, $A = A^T$. As a consequence, symmetric matrix has to square, that is, they has to have the same number of rows and columns. For example, the following is a symmetric matrix:

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 6 \end{pmatrix}.$$

Idempotent matrices: A matrix $\boldsymbol{H}$ is called idempotent if $\boldsymbol{H} = \boldsymbol{H}^2$.

```
H <- 0.5*matrix(1, ncol = 2, nrow = 2)
H
```

```
##      [,1] [,2]
## [1,]  0.5  0.5
## [2,]  0.5  0.5
```

```
## Check that HH = H
H%*%H
```

```
##      [,1] [,2]
## [1,]  0.5  0.5
## [2,]  0.5  0.5
```

Determinants

The determinant of a **square** $p \times p$ matrix $\boldsymbol{A}$ is denoted as $|\boldsymbol{A}|$. Determinant of a 2×2 matrix is computed as

$$|\boldsymbol{A}| = det\begin{pmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{pmatrix} = a_{11}a_{22} - a_{12}a_{21}$$

In general, the determinant of $\boldsymbol{A}$ can be calculated through a more complex formula.

In R, `det()` is used to find the determinant of a matrix.

```
D <- matrix(c(5,3,9,6),2,2)
D
```

```
##      [,1] [,2]
## [1,]    5    9
## [2,]    3    6
```

```
det(D)
```

```
## [1] 3
```

```
E <- matrix(c(1,2,3,6, 6,7,8,9, 10),3,3)
E
```

```
##      [,1] [,2] [,3]
## [1,]    1    6    8
## [2,]    2    6    9
## [3,]    3    7   10
```

```
det(E)
```

```
## [1] 7
```

Some useful properties of determinants:

1. The determinant of a diagonal matrix is the product of the diagonal elements.
2. For any scalar k, $|k\boldsymbol{A}| = k^n|\boldsymbol{A}|$, where $\boldsymbol{A}$ has size $n \times n$.
3. If two rows (or columns) of a matrix are interchanged, the sign of the determinant changes.
4. If two rows or columns are equal or proportional (see material on rank later), the determinant is zero.
5. The determinant is unchanged by adding a multiple of some column (row) to any other column (row).

6. If all the elements of a column / row are zero then the determinant is zero.
7. If $\boldsymbol{A}$ and $\boldsymbol{B}$ are $n \times n$ matrices then $|\boldsymbol{AB}| = |\boldsymbol{A}||\boldsymbol{B}|$.

Rank of a matrix

Rank denotes the number of linearly independent rows or columns. For example:

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

is 3×3 matrix with rank 2 since the first column can be found from the other two columns as $\boldsymbol{a_1} = \boldsymbol{a_2} + \boldsymbol{a_3}$.

If all the rows and columns of a square matrix are linearly independent it is said to be of full rank and non-singular. If $\boldsymbol{A}$ is singular, then $|\boldsymbol{A}| = 0$.

Matrix inversion

Suppose $\boldsymbol{A}$ is a non-singular (full rank) $p \times p$ matrix. There is a unique matrix $\boldsymbol{B}$ such that $\boldsymbol{AB} = \boldsymbol{BA} = \boldsymbol{I}_p$. We call the matrix $\boldsymbol{B}$ the inverse of $\boldsymbol{A}$, and denote by $\boldsymbol{A}^{-\mathbf{1}}$.

The inverse of a 2×2 matrix $\boldsymbol{A}$ can be calculated as follows:

$$\text{If } \boldsymbol{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \text{then } \boldsymbol{A}^{-1} = \frac{1}{|\boldsymbol{A}|} \begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix}$$

The general formula is more complicated.

A singular matrix has no inverse since its determinant is 0.

In R, we use `solve()` to invert a matrix.

```
D <- matrix(c(5,3,9,6),2,2)
D
```

```
##      [,1] [,2]
## [1,]    5    9
## [2,]    3    6
```

```
solve(D)
```

```
##      [,1]      [,2]
## [1,]    2 -3.000000
## [2,]   -1  1.666667
```

Some properties of inverses:

1. The inverse of a symmetric matrix is also symmetric.
2. The inverse of the transpose of $\boldsymbol{A}$ is the transpose of $\boldsymbol{A}^{-1}$.
3. The inverse of the product of several square matrices is a little more subtle: $(\boldsymbol{ABC})^{-1} = \boldsymbol{C}^{-1}\boldsymbol{B}^{-1}\boldsymbol{A}^{-1}$. If c is a non-zero scalar then $(c\boldsymbol{A})^{-1} = c^{-1}\boldsymbol{A}^{-1}$.
4. The inverse of a diagonal matrix is another diagonal matrix with the reciprocals of the original elements as its diagonal elements.

Main page: **ST 437/537: Applied Multivariate and Longitudinal Data Analysis (https://maityst537.wordpress.ncsu.edu/)**