Center for Statistics and the Social Sciences Math Camp 2020

Matrix Algebra

Jessica Godwin & Emily Finchum-Mason

Department of Statistics & Evans School
University of Washington

September 21, 2020

Outline

Matrix Algebra

- Definitions, notation
- Matrix Operations
- Determinants - existence of an inverse
- Linear equations
- Least Squares and Regression with matrices

Motivation

Matrix algebra provides concise notation and rules for manipulating matrices (arrays of numbers).

Matrix algebra will be important for computing linear regression estimates.

Motivation

Example `data.frame` in `R`:

	region	years	u5m	lower	upper
1	All	80-84	0.1691030	0.1573394	0.1815566
2	All	85-89	0.1603335	0.1490694	0.1722763
3	All	90-94	0.1208087	0.1079371	0.1349829
4	tanga	80-84	0.1810487	0.1369700	0.2354425
5	tanga	85-89	0.2230574	0.1677716	0.2902086

- `region`: Regions in Tanzania
- `years`: time, measured in 5-year periods
- `u5m`: estimated under-five mortality rate
- `lower`: lower end of confidence band
- `upper`: upper end of confidence band

A **matrix** is an array of number is a rectangular form.
Examples:

$$A = \begin{bmatrix} 1 & 2 & 6 & 4 \\ 5 & 8 & 12 & 8 \\ 4 & 3 & 2 & 1 \end{bmatrix} B = \begin{bmatrix} 4 & 3 & 2 \\ 1 & 2 & 4 \end{bmatrix}$$

where A is a 3×4 matrix and B is a 2×3 matrix. Note: matrix **dimensions**, $(n \times m)$are always listed as rows × columns.

- **Notation:** Often A is written $A_{n \times m}$.

In mathematical notation, a matrix is written

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}$$

Where x_{ij} is the value in the ith row and the jth column of matrix X.

A **vector** is a matrix that has *n* rows and 1 column (or 1 row and *n* columns).
Examples:

$$\begin{bmatrix} 1 & 2 & 6 & 4 \end{bmatrix} \text{ or } \begin{bmatrix} 4 \\ 5 \\ 1 \end{bmatrix}$$

A **square** matrix has the same number of rows and columns.
Example:

$$\begin{bmatrix} 4 & 3 \\ 1 & 2 \end{bmatrix}$$

A **symmetric** matrix has elements such that $x_{ij} = x_{ji}$.
Example:

$$\begin{bmatrix} 1 & 4 & 5 \\ 4 & 2 & 3 \\ 5 & 3 & 7 \end{bmatrix}$$

A symmetric matrix must also be a square matrix.

A **diagonal** matrix is a matrix that is zero everywhere except on the diagonal. Where the diagonal is defined as all elements for which the row number is equal to the column number $\{(1,1),(2,2),(3,3),...\}$.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 7 \end{bmatrix}$$

A special case of a diagonal matrix is the **identity** matrix. Its diagonal elements are all ones.

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

Clearly, the identity matrix (or any other diagonal matrix) is also symmetric.

Matrix Equality: Two matrices A, B are equal if and only if, for all elements, each $a_{ij} = b_{ij}$. (Note: this means they must have the same dimensions.)

Matrix Transpose: The **transpose** of a matrix is found by interchanging the corresponding rows and columns of a matrix. The first row becomes the first column, the second row becomes the second column, etc. The dimensions are then switched and the element a_{ij} becomes the element a_{ji}. The transposed matrix is often denoted A^t (or A'). You can find the transpose of a matrix in `R` by using the `t()` function.

$$A = \begin{bmatrix} 1 & 2 & 6 \\ 3 & 5 & 9 \end{bmatrix} \quad A^t = \begin{bmatrix} 1 & 3 \\ 2 & 5 \\ 6 & 9 \end{bmatrix}$$

Matrix Operations

Addition & Subtraction

Two matrices can be added or subtracted only if their dimensions are the same (both rows and columns). The corresponding elements are then added or subtracted.

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} \\ a_{21}+b_{21} & a_{22}+b_{22} \end{bmatrix}$$

Example:

$$\begin{bmatrix} 1 & 2 & 6 \\ 3 & 5 & 9 \end{bmatrix} - \begin{bmatrix} 1 & 3 & 8 \\ 6 & 9 & 6 \end{bmatrix} = \begin{bmatrix} 0 & -1 & -2 \\ -3 & -4 & 3 \end{bmatrix}$$

To multiply a matrix by a **scalar** (a constant value; any $a \in \mathbb{R}$), multiply each element by that number.
Example:

$$A = \begin{bmatrix} 1 & 3 & 8 \\ 6 & 9 & 6 \end{bmatrix} \quad 3A = \begin{bmatrix} 3 & 9 & 24 \\ 18 & 27 & 18 \end{bmatrix}$$

Two matrices $A_{n_A \times m_A}$ and $B_{n_B \times m_B}$ can be multiplied only if the number of columns of the first matrix, m_A, equals the number of rows of the second matrix, n_B, i.e. the "inside numbers".

The resulting matrix, $(A \cdot B)_{n_A \times m_B}$ or $(AB)_{n_A \times m_B}$ has n_A rows and m_B columns, i.e. the "outside numbers".

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix}$$

A is (2×3); B is (3×2).
$B \cdot A$ is computable and has dimension $3 \times 2 \cdot 2 \times 3 = 3 \times 3$.
$A \cdot B$ is computable and has dimension $2 \times 3 \cdot 3 \times 2 = 2 \times 2$.

To compute $A_{2\times 3} \cdot B_{3\times 2}$, we find each element $(ab)_{ij}$by summing the crossproducts of the ith row of A and the jth column of B.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix}$$

$$A{\cdot}B = \begin{bmatrix} a_{11} \cdot b_{11} + a_{12} \cdot b_{21} + a_{13} \cdot b_{31} & a_{11} \cdot b_{12} + a_{12} \cdot b_{22} + a_{13} \cdot b_{32} \\ a_{21} \cdot b_{11} + a_{22} \cdot b_{21} + a_{23} \cdot b_{31} & a_{21} \cdot b_{12} + a_{22} \cdot b_{22} + a_{23} \cdot b_{32} \end{bmatrix}$$

Matrix Operations

Multiplication Examples

Examples:

$$A = \begin{bmatrix} 1 & 3 & 8 \\ 6 & 9 & 6 \\ 2 & 1 & 3 \end{bmatrix} \quad B = \begin{bmatrix} 3 & 9 \\ 2 & 1 \\ 3 & 2 \end{bmatrix}$$

$$AB = \begin{bmatrix} 1 \cdot 3 + 3 \cdot 2 + 8 \cdot 3 & 1 \cdot 9 + 3 \cdot 1 + 8 \cdot 2 \\ 6 \cdot 3 + 9 \cdot 2 + 6 \cdot 3 & 6 \cdot 9 + 9 \cdot 1 + 6 \cdot 2 \\ 2 \cdot 3 + 1 \cdot 2 + 3 \cdot 3 & 2 \cdot 9 + 1 \cdot 1 + 3 \cdot 2 \end{bmatrix} = \begin{bmatrix} 33 & 28 \\ 54 & 75 \\ 17 & 25 \end{bmatrix}$$

Matrix Multiplication

Order Matters

$$A = \begin{bmatrix} red1 & red3 & red8 \\ 6 & 9 & 6 \\ 2 & 1 & 3 \end{bmatrix} \quad B = \begin{bmatrix} blue3 & 9 \\ blue2 & 1 \\ blue3 & 2 \end{bmatrix}$$

- $A \cdot B$ is not necessarily equal to $B \cdot A$, as with scalar multiplication.

  This is called the **commutative property**: $4 \times 2 = 2 \times 4 = 8$.

- $B \cdot A$ cannot be computed as the **dimensions are not compatible:** $3 \times 2 \cdot 3 \times 3$.

  The "inside numbers" are not equal: $m_B \neq n_A$.

We need something that "looks like" scalar division.

The **multiplicative inverse** of a scalar, $a \in \mathbb{R}$, is the number, a^{-1} such that $a \times a^{-1}$ equals the **multiplicative identity**, e.g.

$$a \times a^{-1} = 1.$$

We know then that, $a^{-1} = \frac{1}{a}$, or

$$a \times \frac{1}{a} = 1.$$

This gives us the notion of division or multiplying by a fraction. For example,

$$4 \cdot 1/4 = 1$$

$$10 \div 5 = 10 \times \frac{1}{5} = 2 \times 5 \times \frac{1}{5} = 2.$$

The **inverse** of a matrix $A_{n\times n}$ is the matrix $A^{-1}_{n\times n}$ that satisfies

$$A \cdot A^{-1} = I$$

.

$I_{n\times n}$ is the **identity matrix**. It has ones along the diagonal and zeroes everywhere else.

$$I_{3\times 3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Like the multiplicative identity, any matrix multiplied by I is itself:

$$A \times I = I \times A = A.$$

How do we find the inverse? How do we know if the inverse exists?

The **determinant** is a measure, in a sense, of the "volume" of the matrix.
For a 2×2 matrix,

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

the determinant is $D(A) = a \cdot d - b \cdot c$.

- If $D(A) = 0$, A^{-1} does not exist. A is **singular**.
  There is no "volume" to the matrix.
- If $D(A) \neq 0$, A^{-1} exists. A is **nonsingular**.

Examples:

$$A = \begin{bmatrix} 4 & 12 \\ 3 & 6 \end{bmatrix}$$

$D(A) = 4 \cdot 6 - 12 \cdot 3 = -12$. Inverse exists. Matrix is nonsingular.

$$A = \begin{bmatrix} 2 & 4 \\ 1 & 2 \end{bmatrix}$$

$D(A) = 2 \cdot 2 - 4 \cdot 1 = 0$. Inverse does not exist. Matrix is singular.

If the inverse, A^{-1}, exists for $A_{2\times 2}$ computing it easy.

For higher dimensions let a computer do it.

The function `solve()` computes matrix inverses in `R`.

Inverting big matrices can take **a lot** of computing power.

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad A^{-1} = \frac{1}{D(A)} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Recall: $D(A) = a \cdot d - b \cdot c$.

Matrix Operations

Inverse Example

$$A^{-1} = \frac{1}{D(A)} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Example:

$$A = \begin{bmatrix} 4 & 12 \\ 3 & 6 \end{bmatrix}, \quad A^{-1} = \frac{1}{-12} \begin{bmatrix} 6 & -12 \\ -3 & 4 \end{bmatrix} = \begin{bmatrix} -1/2 & 1 \\ 1/4 & -1/3 \end{bmatrix}$$

Linear Equations

Let's go back to thinking about systems of two equations:

$$ax + by = g$$

$$cx + dy = f$$

Previously we solved this system by eliminating the y variable, solving for x, and then substituting back in for y.

No we can write this system in matrix notation:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \quad z = \begin{bmatrix} x \\ y \end{bmatrix}, \quad w = \begin{bmatrix} g \\ f \end{bmatrix}$$

Solving our system of equations is the same as solving for z in the matrix equation:

$$A \cdot z = w$$

Solving our system of equations is the same as solving for z in the matrix equation:

So how do we solve for z?

$$\begin{aligned} A \cdot z =& w \\ A^{-1} \cdot A \cdot z =& A^{-1} \cdot w & \text{[Left-multiply by } A^{-1}] \\ I \cdot z =& A^{-1} \cdot w & [A^{-1} \times A = I] \\ z =& A^{-1} \cdot w. \end{aligned}$$

The solution to our system is $z = A^{-1} \cdot w = \begin{bmatrix} x \\ y \end{bmatrix}$.

Linear Equations

Examples

$$2x + y = 1$$
$$4x + 3y = 8$$

$$A = \begin{bmatrix} 2 & 1 \\ 4 & 3 \end{bmatrix}, \quad z = \begin{bmatrix} x \\ y \end{bmatrix}, \quad w = \begin{bmatrix} 1 \\ 8 \end{bmatrix}$$

$$A^{-1} = \frac{1}{2 \cdot 3 - 4 \cdot 1} \begin{bmatrix} 3 & -1 \\ -4 & 2 \end{bmatrix} = \begin{bmatrix} 3/2 & -1/2 \\ -2 & 1 \end{bmatrix}$$

$$z = A^{-1} \cdot w = \begin{bmatrix} 3/2 & -1/2 \\ -2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 8 \end{bmatrix} = \begin{bmatrix} 3/2 \cdot 1 + -1/2 \cdot 8 \\ -2 \cdot 1 + 1 \cdot 8 \end{bmatrix} = \begin{bmatrix} -5/2 \\ 6 \end{bmatrix}$$

Linear Regression and Least Squares

The goal of **linear regression** is estimate the intercept and slope in a linear relationship between an independent variable or covariate X and a dependent variable or outcome, Y.

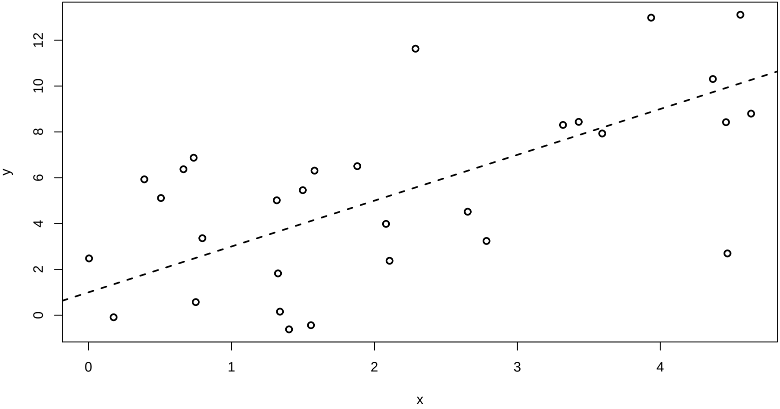
In other words, we want to fit a line through pairs of points (x_i, y_i) for observations $i = 1, \ldots, n$.

What do we do when $n > 2$? What if we have more than one independent variable?

Suppose we conduct a survey where we asked n people the same p questions. We can put that organize that data in a matrix of dimensions $n \times p$, where each row is a person and each column is the numerical response to one of the asked questions.

Least Squares

Simple Linear Regression Example



Least Squares

So how do we choose the dashed line?

We can write the equation:

$$y_i = \beta_0 + \beta_1 x_{1i} + ... + \beta_p x_{pi}$$

,

- number observations: $i = 1, \dots, n$
- number independent variables: $j = 1, .., p$
- intercept: β_0
- slope: β_j for each x_j

Least Squares

$$y_i = \beta_0 + \beta_1 x_{1i} + ... + \beta_p x_{pi}$$

In matrix notation:

$$y = X\beta = \begin{bmatrix} y_1 \\ ... \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & ... & x_{1p} \\ 1 & ... & ... & ... \\ 1 & x_{n1} & ... & x_{np} \end{bmatrix} \cdot \begin{bmatrix} \beta_0 \\ ... \\ \beta_p \end{bmatrix}$$

- $y_{n\times 1}$ is the **response**.
- $X_{n\times(p+1)}$ is the **design matrix**.
  Notice the column of 1's so that each observation's model includes a β_0.
- $\beta_{(p+1)\times 1}$ are the unknown **coefficients** we want to estimate.

Least Squares

How do we choose/estimate $\beta_{(p+1)\times 1}$?

Least squares finds the line that minimizes the squared distance between the points and the line, i.e. makes

$$[y_i - (\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi})]^2$$

as small as possible for all $i = 1, \ldots, n$.

The vector $\widehat{\beta}$ that minimizes the sum of the squared distances is

$$\widehat{\beta} = \left(X^t \cdot X\right)^{-1} X^t y.$$

Note: In statistics, once we have estimated a parameter we put a "hat" on it, e.g. $\widehat{\beta_0}$ is the estimate of the true parameter β_0.

Least Squares

$$\widehat{\beta} = \left(X^t \cdot X\right)^{-1} X^t y.$$

To see this:

$$\begin{aligned}
y_{n\times 1} =& X_{n\times(p+1)}\beta_{(p+1)\times 1} \\
X^t y =& X^t X\beta && [X \text{ isn't square, } X^{-1} \text{ doesn't exist!}] \\
\left(X^t X\right)^{-1} X^t y =& \left(X^t X\right)^{-1} X^t X\beta \\
\left(X^t X\right)^{-1} X^t y =& I \cdot \beta && [(X^T X) \text{ is square and invertible.}] \\
\beta =& \left(X^t \cdot X\right)^{-1} X^t y
\end{aligned}$$

Least Squares

Simple linear regression example in R

Truth:

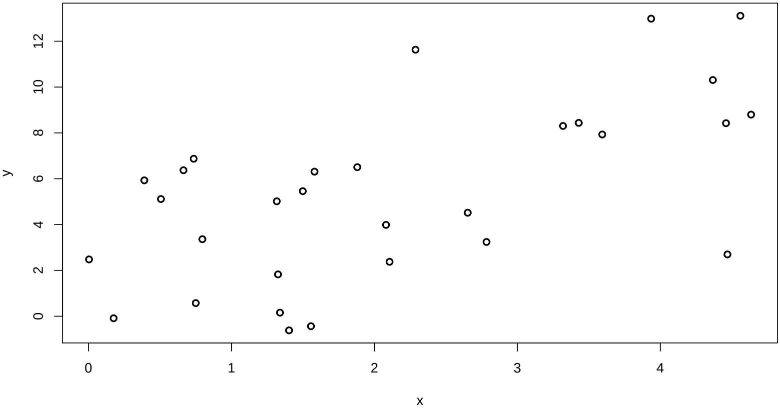
$$y_i = 1 + 2 \cdot x_i + \varepsilon_i,$$

where ε_i $N(0, 3^2)$ is thought of as **noise** or **measurement error**.

```
set.seed(1985)
beta_0<-1
beta_1<-2
n<-30
x<-runif(n,0,5)
y<-rnorm(n,mean=beta_1*x+beta_0,sd=3)
plot(x,y)
```

Least Squares

Simulated Data



Least Squares

with matrices in R

R functions and operators:

- inverse: `solve()`
- transponse: `t()`
- matrix multiplication: $\% * \%$

```
X.mat<-matrix(c(rep(1,n),x),ncol=2)
Beta.mat<-solve( t(X.mat)%*%(X.mat) ) %*% t(X.mat)%*%y
```

First two rows of design matrix, X, and coefficients, $\widehat{\beta}$, estimated via least squares.

```
X.mat[1:2,]
     [,1]     [,2]
[1,]    1 3.319174
[2,]    1 1.325468
```

```
Beta.mat
         [,1]
[1,] 1.960837
[2,] 1.590737
```

Least Squares

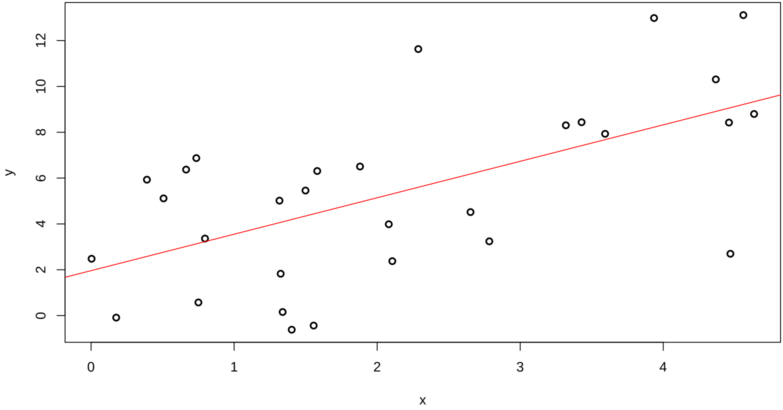


Figure: Our data with the fitted line $y = 1.59x + 1.96$.

Least Squares

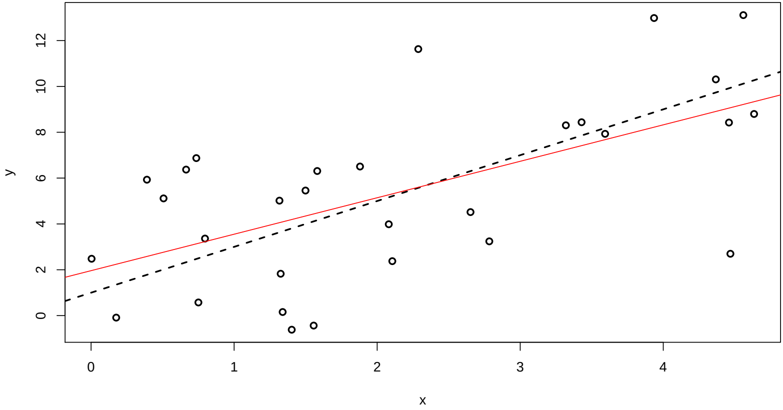


Figure: Our data with the fitted line $y = 1.96 + 1.59x$ and the true line $y = 1 + 2x$.