# PS200C Section: Basic Linear Algebra Review

Haotian (Barney) Chen

2024-04-03

This document is a slight modification of notes from the previous TAs, Doeun Kim and Soonhong Cho.

Section 1

# Definitions

## Definition (Matrix)

A **matrix** is a rectangular array of numbers. An $(n \times k)$ matrix has $n$ rows and $k$ columns.

$$\underset{(n\times k)}{\mathbf{A}} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1k} \\ a_{21} & a_{22} & \cdots & a_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nk} \end{bmatrix}$$

- We'll often organize our data such that each row is an observation, and each column is a covariate.
- Type `mathbf{X}` is Latex.
- We denote each element of a matrix with the lower case and its location, like $a_{12}$.

## Definition (Row Vector)

A $(1 \times k)$ matrix is a **row vector**.

$$\underset{(1\times k)}{\mathbf{y}} = \begin{bmatrix} y_{11} & \cdots & y_{1k} \end{bmatrix}$$

## Definition (Column Vector)

A $(n \times 1)$ matrix is a **column vector**.

$$\underset{(n\times 1)}{\mathbf{y}} = \begin{bmatrix} y_{11} \\ \vdots \\ y_{n1} \end{bmatrix}$$

## Definition (Scalar)

A scalar is a $(1 \times 1)$ matrix/vector (i.e., a constant).

## Definition (Square Matrix)

A **square matrix** is one where $n = k$: the same number of rows and columns.

$$\underset{(n\times n)}{\mathbf{A}} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

## Definition (Diagonal Matrix)

A **diagonal matrix** is a square matrix where $a_{ij} = 0$ if $i \neq j$.

$$\underset{(n\times n)}{\mathbf{A}} = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{bmatrix}$$

## Definition (Identity Matrix)

An **identity matrix** is a diagonal matrix whose diagonal elements ($a_{ij}$ where $i = j$) are one, and off-diagonal elements ($a_{ij}$ where $i \neq j$) are zero.

$$\underset{(n \times n)}{\mathbf{I}} \equiv \mathbf{I}_n \equiv \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

The identity matrix is the matrix version of "1." That is, any matrix multiplied by the identity matrix is itself.

Section 2

# Matrix Operations

# Addition (A+B)

We can add two matrices $\mathbf{A}$ and $\mathbf{B}$ if they have the same dimensions. It is conducted element by element, and represented by $\mathbf{A} + \mathbf{B} = [a_{ij} + b_{ij}]$.

$$\underset{n \times k}{\mathbf{A}} + \underset{n \times k}{\mathbf{B}} = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1k} + b_{1k} \\ a_{21} + b_{21} & a_{22} + b_{22} & \cdots & a_{2k} + b_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} + b_{n1} & a_{n2} + b_{n2} & \cdots & a_{nk} + b_{nk} \end{bmatrix}$$

For example: if $\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$, $\mathbf{B} = \begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}$, then $\mathbf{A} + \mathbf{B} = \begin{bmatrix} 3 & 6 \\ 9 & 12 \end{bmatrix}$

# Scalar multiplication (cA)

For any scalar $c \in \mathbb{R}^1$, $c\mathbf{A} = [c \cdot a_{ij}]$:

$$c\mathbf{A} = \begin{bmatrix} ca_{11} & ca_{12} & \cdots & ca_{1k} \\ ca_{21} & ca_{22} & \cdots & ca_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ ca_{n1} & ca_{n2} & \cdots & ca_{nk} \end{bmatrix}$$

# Matrix multiplication

The matrix multiplication $\underset{(n\times m)}{\mathbf{A}}\underset{(m\times k)}{\mathbf{B}}$ is only possible if they are **conformable**: the number of columns of the left one $\mathbf{A}$ must be equal to the number of rows of the right one $\mathbf{B}$. The dimension of resulting matrix $\mathbf{AB}$ is $(n \times m)(m \times k) = (n \times k)$. However, $\underset{(m\times k)}{\mathbf{B}}\underset{(n\times m)}{\mathbf{A}}$ is undefined, because the matrices are not conformable.

# Matrix multiplication

One trick for easier life is thinking column-wise way: the rows of **B** give a "recipe" to combine the columns of **A**. Here is one example: say we want to compute **AB** for

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{n1} & a_{n2} \end{bmatrix}, \mathbf{B} = \begin{bmatrix} {\color{blue}b_{11}} & {\color{red}b_{12}} \\ {\color{blue}b_{21}} & {\color{red}b_{22}} \end{bmatrix}.$$

They are conformable and the dimension of the resulting matrix must be $(3 \times 2) \times (2 \times 2) = (3 \times 2)$. We have:

$$\underset{(3\times{\color{blue}2})({\color{blue}2}\times 2)}{\mathbf{A}\ \mathbf{B}} = \begin{bmatrix} {\color{blue}b_{11}} \begin{pmatrix} a_{11} \\ a_{21} \\ a_{31} \end{pmatrix} + {\color{blue}b_{21}} \begin{pmatrix} a_{12} \\ a_{22} \\ a_{32} \end{pmatrix} & {\color{red}b_{12}} \begin{pmatrix} a_{11} \\ a_{21} \\ a_{31} \end{pmatrix} + {\color{red}b_{22}} \begin{pmatrix} a_{12} \\ a_{22} \\ a_{32} \end{pmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} {\color{blue}b_{11}} a_{11} + {\color{blue}b_{21}} a_{12} & {\color{red}b_{12}} a_{11} + {\color{red}b_{22}} a_{12} \\ {\color{blue}b_{11}} a_{21} + {\color{blue}b_{21}} a_{22} & {\color{red}b_{12}} a_{21} + {\color{red}b_{22}} a_{22} \\ {\color{blue}b_{11}} a_{31} + {\color{blue}b_{21}} a_{32} & {\color{red}b_{12}} a_{31} + {\color{red}b_{22}} a_{32} \end{bmatrix},$$

where the first equality is our "trick," and the second one is the definition.

# Matrix multiplication

A more specific example: say $\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$ and $\mathbf{B} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$. Then, we have:

$$\mathbf{AB} = \begin{bmatrix} 1\begin{pmatrix} 1 \\ 2 \end{pmatrix} + 0\begin{pmatrix} 3 \\ 4 \end{pmatrix} & 1\begin{pmatrix} 1 \\ 2 \end{pmatrix} + 1\begin{pmatrix} 3 \\ 4 \end{pmatrix} \end{bmatrix} = \begin{bmatrix} 1 & 4 \\ 2 & 6 \end{bmatrix}.$$

In `R`, do not use the usual multiplication mark $*$ as it's for element-wise multiplication (in the above example, it will return $\begin{bmatrix} 1 & 3 \\ 0 & 4 \end{bmatrix}$). You have to use `%*%` for matrix multiplication!

```r
A <- matrix(c(1, 2, 3, 4), byrow=FALSE, ncol=2)
A
```

```
##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4
```

```r
B <- cbind(c(1, 0), c(1, 1)) #"column vector" way
B
```

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

```r
A*B #element-wise multiplication ("Hadamard product")
```

```
##      [,1] [,2]
## [1,]    1    3
## [2,]    0    4
```

```r
A %*% B #matrix multiplication
```

```
##      [,1] [,2]
## [1,]    1    4
## [2,]    2    6
```

# Properties of matrix multiplication

1. The commutative property does not hold! Notice that **BA** is NOT conformable:

$$\underset{(n\times m)}{\mathbf{A}}\underset{(m\times k)}{\mathbf{B}} \neq \underset{(m\times k)}{\mathbf{B}}\underset{(n\times m)}{\mathbf{A}}$$

2. Associative property:

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

3. Distributive properties (both left and right hold):

$$\mathbf{A}(\mathbf{B}+\mathbf{C}) = \mathbf{AB}+\mathbf{AC} \text{ and } (\mathbf{B}+\mathbf{C})\mathbf{A} = \mathbf{BA}+\mathbf{CA}$$

4. Multiplicative identity property:

$$\mathbf{IA} = \mathbf{A} \text{ and } \mathbf{AI} = \mathbf{A}$$

5. Dimension property:

The product of an $(n\times m)$ matrix and an $(m\times k)$ matrix is an $(n\times k)$ matrix.

# Transpose

## Definition (Transpose)

Let $\mathbf{A} = [a_{ij}]$ be an $(n \times k)$ matrix. The **transpose** of $\mathbf{A}$, denoted $\mathbf{A}'$ or $\mathbf{A}^T$ is the $k \times n$ matrix whose elements are $\mathbf{A}' \equiv [a_{ji}]$

When you transpose, you essentially "mirror" the matrix about its diagonal. This is the same as saying that column $i$ of a matrix is row $i$ of its transpose. For example, if $\mathbf{A} = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$, then $\mathbf{A}' = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$.

# Properties of Transpose

1. $(\mathbf{A}')' = \mathbf{A}$
2. $\alpha' = \alpha$, where $\alpha$ is a scalar
3. $(\alpha\mathbf{A})' = \alpha\mathbf{A}'$
4. $(\mathbf{A}+\mathbf{B})' = \mathbf{A}' + \mathbf{B}'$
5. $(\mathbf{AB})' = \mathbf{B}'\mathbf{A}'$, where $\mathbf{A}$ is $(n \times k)$ and $\mathbf{B}$ is $(k \times m)$. Note that they become $(m \times n)$ matrices.
6. Inner product: let $\mathbf{x}$ be a $(n \times 1)$ vector. Then $\mathbf{x}'\mathbf{x} = \sum_i^n x_i^2$. For example, if $\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$, then

$$\mathbf{x}'\mathbf{x} = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = x_1^2 + x_2^2 + x_3^2 = \sum_{i=1}^{3} x_i^2.$$

Expanding this result, you will likely run into the construction $\mathbf{X}'\mathbf{X}$ (hint: in OLS estimator); this is the matrix world equivalent of "squaring and summing".

```
A
```

```
##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4
```

```
t(A) #transpose: function `t`
```

```
##      [,1] [,2]
## [1,]    1    2
## [2,]    3    4
```

# Symmetric Matrix

**Definition (Symmetric Matrix)**

A square matrix $\mathbf{A}$ is **symmetric** if and only if $\mathbf{A}' = \mathbf{A}$. If $\mathbf{A}$ is a $(n \times k)$ non-square matrix, then $\mathbf{A}'\mathbf{A}$ is always defined and is symmetric.

Let's see if $\mathbf{A}'\mathbf{A}$ is symmetric using the definition and some properties:

$$(\mathbf{A}'\mathbf{A})' = \mathbf{A}'(\mathbf{A}')' = \mathbf{A}'\mathbf{A},$$

so $\mathbf{A}'\mathbf{A}$ is symmetric.

# Trace

## Definition (Trace)

For any $(n \times n)$ symmetric matrix $\mathbf{A}$, the trace of $\mathbf{A}$, denoted $tr(\mathbf{A})$, is the sum of its diagonal elements: $tr(\mathbf{A}) = \sum_{i=1}^{n} a_{ii}$.

- Some useful properties include:
  1. $tr(\mathbf{I}_n) = n$
  2. $tr(\mathbf{A} + \mathbf{B}) = tr(\mathbf{A}) + tr(\mathbf{B})$
  3. $tr(\alpha\mathbf{A}) = \alpha tr(\mathbf{A})$
  4. $tr(\mathbf{AB}) = tr(\mathbf{BA})$, where $\mathbf{A}$ is $(n \times k)$ and $\mathbf{B}$ is $(k \times n)$. This one is really useful when you learn more advanced matrix algebra in the context of statistics.

# Inverse

## Definition (Inverse)

An $(n \times n)$ square matrix $\mathbf{A}$ has an **inverse** $\mathbf{A}^{-1}$ if and only iff $\mathbf{A}^{-1}\mathbf{A} = \mathbf{A}\mathbf{A}^{-1} = \mathbf{I_n}$.

- In this case $\mathbf{A}$ is said to be *invertible* or *non-singular*. Otherwise, $\mathbf{A}$ is called *non-invertible* or *singular*.
- An inverse is **only defined for square matrices**, but not all square matrices has an inverse.

# Rank

**Definition (Rank)**

The rank of a matrix is the maximum number of linearly independent rows/columns.

- To get the rank, transform the matrix to row echelon form and count nonzero rows. A matrix has rank $min(m, n)$ is called full-rank.

# (Back to) Inverse

- Intuitively, if a matrix is full-rank, it means that every row/column provides unique and new information that we can't get from any of the other rows/columns or their some combinations.
- This is why OLS does not work under "perfect multicollinearity": because the OLS estimator $\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$ needs to invert the quantity $(\mathbf{X}'\mathbf{X})$, it requires that $\mathbf{X}$ to be full-rank.

# Inverse

- Example: are they invertible?

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 3 & 4 & 1 \\ 1 & 3 & 2 \\ 4 & 6 & 2 \end{bmatrix}$$

```
solve(A) #R function for inverse is `solve`
```

```
##      [,1] [,2]
## [1,]   -2  1.5
## [2,]    1 -0.5
```

```
C <- matrix(c(3,1,4,4,3,6,1,2,2), nrow=3)
solve(C) #not invertible ("singular")
```

```
## Error in solve.default(C): system is computationally singular: recip
```

Why C is not invertible? Check the sum of column 1 and column 3.

# Properties of the Inverse

1. $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$, if they are invertible.
2. $(\mathbf{A}')^{-1} = (\mathbf{A}^{-1})'$
3. The inverse of a matrix is *unique*, if exists.

# Matrix Differentiation

- When we derive a regression model in matrix form, we have to use some partial derivative of a matrix. A **partial derivative** of a function of some variables is its derivative with respect to one of those variables, with the others held constant.
- Let $\mathbf{A}$ be a $(n \times n)$ symmetric matrix, $\mathbf{a}$ be a $(n \times 1)$ vector of constants (i.e., not random variables), and $\mathbf{y}$ be a $(n \times 1)$ vector of **random variables**.
- Consider a linear function $f(\mathbf{y}) = \mathbf{a}'\mathbf{y} = (a_1 y_1 + \cdots + a_n y_n)$.
- Then the partial derivative of if with regard to the vector $\mathbf{a}$ is:

$$\frac{\partial f(\mathbf{y})}{\partial \mathbf{y}} = \frac{\partial \mathbf{a}'\mathbf{y}}{\partial \mathbf{y}} = \begin{pmatrix} \frac{\partial \mathbf{a}'\mathbf{y}}{\partial y_1} \\ \vdots \\ \frac{\partial \mathbf{a}'\mathbf{y}}{\partial y_n} \end{pmatrix} = \begin{pmatrix} \frac{\partial (a_1 y_1 + \cdots + a_n y_n)}{\partial y_1} \\ \vdots \\ \frac{\partial (a_1 y_1 + \cdots + a_n y_n)}{\partial y_n} \end{pmatrix} = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} = \mathbf{a}.$$

Section 3

# Regression in matrix form

# Setting

- The (multiple) regression model in matrix form is

$$\mathbf{y} = \mathbf{X}\beta + \epsilon,$$

where

$$\underset{(n\times 1)}{\mathbf{y}} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \underset{(n\times(k+1))}{\mathbf{X}} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{nk} \end{bmatrix},$$

$$\underset{((k+1)\times 1)}{\beta} = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_k \end{bmatrix}, \underset{(n\times 1)}{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

# Setting

- Then $\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$ is the solution to the least square problem:

$$\hat{\beta}_{OLS} = \underset{\beta}{\text{argmin}}(\mathbf{Y} - \mathbf{X}\beta)'(\mathbf{Y} - \mathbf{X}\beta)$$

- $\epsilon = \mathbf{y} - \mathbf{X}\beta$ is the error term which we cannot observe. Denote $e$ as the residual.
- $\epsilon'\epsilon = (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$ - sum of squared errors.
- Note that inner product of a matrix is like "squaring" the matrix: so we call

$$\epsilon'\epsilon = \begin{pmatrix} \epsilon_1 & \cdots & \epsilon_n \end{pmatrix} \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix} = \epsilon_1^2 + \cdots + \epsilon_n^2 = \sum_{i=1}^{n} \epsilon_i$$

- $e'e = (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$ - sum of squared residuals.

The goal is to find $\hat{\beta}$ minimizes the sum of squared residuals.

$$\begin{aligned} e^{\intercal}e &= (y - \mathbf{X}\beta)^{\intercal}(y - \mathbf{X}\beta) \\ &= \text{expand the equation yourself} \\ &= (y^{\intercal}y - 2\hat{\beta}^{\intercal}\mathbf{X}^{\intercal}y + \hat{\beta}^{\intercal}X^{\intercal}X\hat{\beta}) \end{aligned}$$

$$\begin{aligned} \frac{de^{\intercal}e}{d\hat{\beta}} &= \frac{d}{d\hat{\beta}}(Y^{\intercal}Y - 2\hat{\beta}^{\intercal}\mathbf{X}^{\intercal}Y + \hat{\beta}^{\intercal}X^{\intercal}X\hat{\beta}) \\ &= -2\mathbf{X}^{\intercal}Y + 2\mathbf{X}^{\intercal}\mathbf{X}\hat{\beta} \end{aligned}$$

Set it to be zero (first order condition)

$$\begin{aligned} 2\mathbf{X}^{\intercal}\mathbf{X}\hat{\beta} &= 2\mathbf{X}^{\intercal}Y \\ \mathbf{X}^{\intercal}\mathbf{X}\hat{\beta} &= \mathbf{X}^{\intercal}Y \\ (\mathbf{X}^{\intercal}\mathbf{X})^{-1}(\mathbf{X}^{\intercal}\mathbf{X})\,\hat{\beta} &= (\mathbf{X}^{\intercal}\mathbf{X})^{-1}\mathbf{X}^{\intercal}Y \\ I\,\hat{\beta} &= (\mathbf{X}^{\intercal}\mathbf{X})^{-1}\mathbf{X}^{\intercal}Y \\ \hat{\beta} &= (\mathbf{X}^{\intercal}\mathbf{X})^{-1}\mathbf{X}^{\intercal}Y \end{aligned}$$

We can derive the variance-covariance matrix of the OLS estimator, $\hat{\beta}$:

$$\mathbb{V}(\hat{\beta}) = \mathbb{E}[(\hat{\beta} - \beta)(\hat{\beta} - \beta)']$$