# ENGS 96: Mathematical Foundations for Machine Learning
# Fall 2025

**Lecturer: Professor Peter Sang Chin**

Notes by: Farhan Sadeek

Last updated: November 13, 2025

## Introduction

The syllabus, as well as any additional class postings, can be found on the Canvas page for this course. Office hours are Tuesdays and Thursdays from 9 am to 11 am.

We will have bi-weekly homework assignments which will be worth 50% of the final grade, a midterm (20%), and a final exam (30%). The midterm and final exams would be open note and open book but NOT open internet.

## Contents

# 1 Importance of Mathematics in Machine Learning

We still don't know how the machine learning model works, for example, when we prompt large language models like ChatGPT or Gemini, it generates responses based on patterns in the data it was trained on but no one really knows how it learnt to generate those responses. The goal of this course is to understand some mathematical foundations of machine learning.

There are two types of machine learning in general, **supervised** and **unsupervised** learning. For example, we have the following:

- For **supervised** learning, we have labeled data and the model learns to predict the output from the input.
- For **unsupervised** learning, we have unlabeled data and the model tries to find patterns or groupings in the data.
- There is also **reinforcement** learning, where the model learns to make decisions by receiving rewards or penalties based on its actions. In reinforcement learning, an agent interacts with an environment and learns to make decisions by receiving feedback in the form of rewards or penalties. The goal is to learn a policy that maximizes the cumulative reward over time.

# 2 17 September 2025

The basic unit of this class is a **vector**, which can be represented as $v \in \mathbb{R}^n$, where $n$ is the dimension of the vector, and we can write our problems as

$$\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

**Definition** (Rank)
The **rank** of a matrix $M$ is the maximum number of linearly independent row vectors or column vectors in the matrix.

**Fact 0.1**
The number of linearly independent rows is equal to the number of linearly independent columns.

## 1 Matrix Multiplication

Let $M$ be an $m \times n$ matrix and $N$ be an $r \times s$ matrix. The product $MN$ is defined if and only if $n = r$, and the resulting matrix will be of size $m \times s$. The entry in the $i$-th row and $j$-th column of the product matrix is given by

$$(MN)_{ij} = \sum_{k=1}^{n} M_{ik} N_{kj}$$

**Example 1.1**

$$\begin{bmatrix} 2 & 3 \\ -1 & 4 \end{bmatrix} \times \begin{bmatrix} 0 & 5 \\ -2 & 3 \end{bmatrix} = \begin{bmatrix} (2 \cdot 0 + 3 \cdot -2) & (2 \cdot 5 + 3 \cdot 3) \\ (-1 \cdot 0 + 4 \cdot -2) & (-1 \cdot 5 + 4 \cdot 3) \end{bmatrix} = \begin{bmatrix} -6 & 19 \\ -8 & 7 \end{bmatrix}$$

**Fact 1.2**

When we multiply two matrices, we are considering the first matrix as columns and the second matrix as rows. The number of columns of the first matrix must be equal to the number of rows of the second matrix.

We can think of matrix multiplication as linear combinations of the columns of the first matrix weighted by the entries of the rows of the second matrix.

$$\begin{bmatrix} \vec{c_1} & \dots & \vec{c_n} \end{bmatrix}^{m \times n} \begin{bmatrix} r_1 \\ \vdots \\ r_n \end{bmatrix}^{r \times q} = \begin{bmatrix} r_1 \vec{c_1} + \dots + r_n \vec{c_n} \end{bmatrix}^{m \times n}$$

We can think of matrix mutliplication as a **linear mapping** or **transformation** from $\mathbb{R}^n$ to $\mathbb{R}^m$, and we write this as

$$T : \mathbb{R}^n \longrightarrow \mathbb{R}^m$$

In machine learning, we often we are given the $n$ dimensions and we have to transform into $m$, and often times we don't know what the relationship between them is and that's what we are trying to figure out.

$$\mathbb{R}^n \xrightarrow{h} \mathbb{R}^m$$

For simplicity, suppose we think that the relationship between $m$ and $n$ is linear and we know that matrix multiplication is a linear transformation, so we can write

$$\mathbb{R}^n \xrightarrow{M} \mathbb{R}^m$$

where $M$ is an $m \times n$ matrix. The goal of machine learning is to find the matrix $M$ that best describes the relationship between $m$ and $n$.

$$\begin{bmatrix} x_1^{(1)} & \dots & x_n^{(1)} \\ x_1^{(2)} & \dots & x_n^{(2)} \\ \vdots & & \vdots \\ x_1^{(n)} & \dots & x_n^{(n)} \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(n)} \end{bmatrix}$$

In machine learning we are trying to figure out what $\vec{\theta}$ is.

## 2 Matrix Inverses

**Lemma 2.1**
If we have a matrix

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

then the inverse of this matrix is given by

$$\frac{1}{ad - bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

provided that $ad - bc \neq 0$.

**Lemma 2.2**
The set of invertible matrices is larger than the set of singular matrices, we can think of that as this. A matrix is invertible if and only if the determinant is non-zero and a matrix is singular if and only if the determinant is zero.

**Fact 2.3**
A set of invertible matrices forms a closed set but non-invertible matrices forms an open set.

We can transform a matrix into its inverse by using the following

$$\left[\begin{matrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{matrix} \quad \Big| \quad \begin{matrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{matrix}\right]$$

As we perform gaussian elimination on the left side, we perform the same row operations on the right side. If we can reduce the left side to the identity matrix, then the right side will become the inverse of the original matrix. It works because we can write $A \cdot I = A$, and if we perform row operations on $A$ to get $I$, we are essentially multiplying $A$ by some matrix $E$ such that $E \cdot A = I$. Therefore, $E$ is the inverse of $A$.

In **supervised learning**, we are often given a

# 3 19 September 2025

Here for this supervised machine learning model, we are trying to are given let's say an input and output dataset, and we are trying to figure out the relattionship between the input and output. We can think of this as a function or linear transformation like following:

$$X \longrightarrow Y \longrightarrow \hat{y}$$

Where $X$ is the input data, $Y$ is the true output data, and $\hat{y}$ is the predicted output data. We want to find a function $f$ such that $f(X) = \hat{y}$ and we want to minimize the difference between $Y$ and $\hat{y}$.

Now, we want to

**Definition** (Generalized Inverse)

The **generalized inverse** of a matrix $A$ is a matrix $A^+$ that satisfies the following properties:

- $AA^+A = A$
- $A^+AA^+ = A^+$
- $(AA^+)^T = AA^+$
- $(A^+A)^T = A^+A$

Generalized inverses are usually very useful for things like these.