

# G4G Linear Algebra

<https://www.geeksforgeeks.org/machine-learning/ml-linear-algebra-operations/>

- Data in ML is represented as vectors (features) and matrices (datasets).
- Operations like dot product, matrix multiplication and transformations power ML algorithms.
- Key concepts such as eigenvalues, eigenvectors and decompositions simplify dimensionality reduction, optimization and training.
- Algorithms like PCA, SVD, regression, SVMs and neural networks rely heavily on linear algebra.

## \*1. Vectors

Vectors are quantities that have both magnitude and direction, often represented as arrows in space.

## \*2. Matrices

Matrices are rectangular arrays of [numbers](#), arranged in rows and columns.

Machine learning:

- represent linear transformations
- systems of linear equations
- data transformations

## 3. Scalars

Scalars are single numerical values, without direction, magnitude only. Scalars are just single numbers that can multiply vectors or matrices.

In machine learning:

- to adjust things like the weights in a model or the learning rate during training

### Operation in Linear Algebra

**Dot Product:** Measures similarity of directions by multiplying matching elements and summing.

**\*\*Example:\*\***  $u \cdot v = u_1v_1 + u_2v_2 + u_3v_3$

**Cross Product:** For 3D vectors, produces a new vector perpendicular to both.

**\*\*Example:\*\***  $u \times v = [u_2v_3 - u_3v_2, u_3v_1 - u_1v_3, u_1v_2 - u_2v_1]$

### Linear Transformation

[Linear transformations](#) are basic operations in linear algebra that change vectors and matrices while keeping important properties like straight lines and proportionality.

In machine learning:

- preparing data
- creating features
- training models.

**Definition:** A transformation  $T$  is linear if it satisfies:

- **Additivity:**  $T(u+v) = T(u) + T(v)$
- **Homogeneity:**  $T(kv) = k T(v)$

\*\*Common Types in ML

- **Translation** : Centering data by subtracting the mean.
- **Scaling** : Normalizing features so no single feature dominates.
- **Rotation\*** : Turning data, often used in computer vision and robotics.

# Matrix Operations

Matrix operations are central to linear algebra and widely used in machine learning for data handling, transformations and model training. The most common ones are:

- **Matrix Multiplication:** Combines two matrices by taking the dot product of rows and columns. Used in feature transformations, parameter computation and neural network operations.

**Example:**  $A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ ,  $B = \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix}$ ,  $A \times B = \begin{bmatrix} 7 & 2 \\ 5 & 4 \end{bmatrix}$

- **Transpose:** Flips a matrix across its diagonal (rows become columns). Denoted by  $A^T$ .
- **Inverse:** The matrix  $A^{-1}$  satisfies  $A \cdot A^{-1} = I$ .  
Exists only if  $\det(A) \neq 0$ . Used in solving equations and optimization.
- **Determinant:** A scalar value indicating whether a matrix is invertible. If  $\det(A) = 0$ , the matrix cannot be inverted.

# Eigenvalues and Eigenvectors

Eigenvalues and eigenvectors describe how matrices transform space, making them fundamental in many ML algorithms.

- **Eigenvalues ( $\lambda$ ):** Scalars showing how much a transformation stretches or compresses along a direction.
- **Eigenvectors ( $v$ ):** Non-zero vectors that only scale (not change direction) under transformation.

**Example:** For  $A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$

solving  $\det(A - \lambda I) = 0$  gives  $\lambda_1 = 1, \lambda_2 = 3$ .

- $\lambda_1 = 1 \rightarrow v_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$
- $\lambda_2 = 3 \rightarrow v_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

**Eigen Decomposition:**  $A = Q\Lambda Q^{-1}$

where  $Q$  holds eigenvectors and  $\Lambda$  is diagonal with eigenvalues.

**Applications in ML:**

- **Dimensionality Reduction (PCA):** Keeps directions with largest eigenvalues (most variance).
- **Matrix Factorization (SVD, NMF):** Breaks large

datasets into smaller, structured parts for feature extraction.

## Eigenvector of a $3 \times 3$ Matrix

An example of eigenvector of  $3 \times 3$  matrix is,

**Example:** Find the eigenvalues and the eigenvector for

the matrix  $A = \begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix}$

**Solution:**

If eigenvalues are represented using  $\lambda$  and the

eigenvector is represented as  $v = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$

Then the eigenvector is calculated by using the equation,

$$\begin{aligned} |A - \lambda I| &= 0 \\ \begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 2 - \lambda & 2 & 2 \\ 2 & 2 - \lambda & 2 \\ 2 & 2 & 2 - \lambda \end{bmatrix} &= 0 \end{aligned}$$

*Simplifying the above determinant we get*

$$\Rightarrow (2 - \lambda)(\lambda^2) + 2\lambda^2 + 2\lambda^2 = 0 \Rightarrow (-\lambda^3) + 6\lambda^2 = 0$$

$$\Rightarrow \lambda^2(6 - \lambda) = 0$$

$$\Rightarrow \lambda = 0, \lambda = 6, \lambda = 0$$

## **Linear Independence**

It is necessary for determining the size of a vector space and finding solutions for optimization problems.