Linear algebra

1. Linear Transformation

🔷 What is a Linear Transformation?

A **linear transformation** (or linear map) is a function T that transforms vectors from one vector space to another, while preserving the structure of vector operations.

If $T : \mathbb{R}^n \to \mathbb{R}^m$, then for all vectors $\vec{u}, \vec{v} \in \mathbb{R}^n$ and any scalar $c \in \mathbb{R}$:

- **Additivity**: $T(\vec{u} + \vec{v}) = T(\vec{u}) + T(\vec{v})$
- **Homogeneity**: $T(c\vec{u}) = cT(\vec{u})$

🔷 Matrix Representation

Every linear transformation can be represented as multiplication by a **matrix**.

Suppose $T : \mathbb{R}^2 \to \mathbb{R}^2$ is defined as:

$$T(x, y) = (2x + y, x - y)$$

Then it can be written in matrix form as:

$$T\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}$$

🔷 Geometric Interpretations

A linear transformation can **rotate**, **reflect**, **scale**, or **shear** a vector space.

- **Scaling**: Multiply the vector by a constant (e.g., $T(x, y) = (2x, 2y)$)
- **Rotation**: Rotating vectors in the plane (using rotation matrices)
- **Reflection**: Across a line or plane
- **Projection**: Flattening a space into a subspace (e.g., projecting 3D onto a plane)

🔷 Important Properties

- If $T(\vec{x}) = A\vec{x}$, then:
  - The **null space** of A represents vectors that become zero after transformation.
  - The **column space** of A gives all possible outputs of T.

---

2. Rank, Spaces, and Span

Let's break this into parts:

🔷 Span

Definition: The span of a set of vectors is the collection of all **linear combinations** of those vectors.

If $\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_k$ are vectors, then their span is:

$$\text{Span}(\vec{v}_1, \ldots, \vec{v}_k) = \{c_1\vec{v}_1 + c_2\vec{v}_2 + \cdots + c_k\vec{v}_k \mid c_i \in \mathbb{R}\}$$

Example:

- Vectors $(1, 0)$ and $(0, 1)$ span $\mathbb{R}^2$ — they can generate any 2D vector.

Use: To determine if vectors are sufficient to describe a whole space.

🔷 Column Space (Image)

The **column space** of a matrix A is the set of all possible linear combinations of its **columns**.

- Also known as the **range** or **image** of the matrix.
- It tells you what outputs the transformation $A\vec{x}$ can produce.

If the matrix is:

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

Then the column space is the span of columns:

$$\text{Col}(A) = \text{Span}\left\{ \begin{bmatrix} 1 \\ 3 \end{bmatrix}, \begin{bmatrix} 2 \\ 4 \end{bmatrix} \right\}$$

🔷 Null Space (Kernel)

The **null space** is the set of all vectors $\vec{x}$ such that $A\vec{x} = 0$.

- It contains all solutions to the homogeneous system of equations.
- Represents input vectors that get **"annihilated"** (mapped to 0 vector) by the transformation.

Example:

$$A = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}, \quad A\vec{x} = 0 \Rightarrow \vec{x} \in \text{Null}(A)$$

This matrix has a null space because its rows are linearly dependent.

🔷 Row Space

The **row space** of a matrix is the span of its row vectors (as vectors in $\mathbb{R}^n$).

- It reflects the number of **linearly independent equations** in the system $A\vec{x} = \vec{b}$.

🔷 Rank

The **rank** of a matrix is the dimension of the column space (or row space) — i.e., the number of **linearly independent columns**.

- Rank tells us the number of directions in which the matrix transformation has an effect.
- A full-rank matrix transforms the space without collapsing it (no dimension is lost).

Example:

$$A = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \Rightarrow \text{Rank}(A) = 1$$

Because the second column is just twice the first — they're linearly dependent.

🔷 Rank-Nullity Theorem

For a matrix A with n columns:

$$\text{Rank}(A) + \text{Nullity}(A) = n$$

Where:

- Rank = dimension of column space (output space)
- Nullity = dimension of null space (solutions mapped to zero)

Column Space – Super Intuitive Explanation 🚀

Column Space – Intuition

The **column space** of a matrix A is one of the most fundamental concepts in linear algebra. It provides deep insight into the structure and properties of a linear transformation represented by A. Here's an intuitive breakdown:

---

1. What is the Column Space?

- The column space of a matrix A (denoted $\mathrm{Col}(A)$) is the set of all possible linear combinations of its column vectors.
- In other words, it's the **span** of the columns of A.

Example:
Consider a matrix A:

$$A = \begin{bmatrix} 1 & 0 \\ 2 & 3 \end{bmatrix}$$

- The columns are $\mathbf{c}_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$ and $\mathbf{c}_2 = \begin{bmatrix} 0 \\ 3 \end{bmatrix}$.
- The column space is all vectors of the form:

$$x\mathbf{c}_1 + y\mathbf{c}_2 = x\begin{bmatrix} 1 \\ 2 \end{bmatrix} + y\begin{bmatrix} 0 \\ 3 \end{bmatrix} = \begin{bmatrix} x \\ 2x + 3y \end{bmatrix}$$

- This spans the entire $\mathbb{R}^2$ plane (since the columns are linearly independent).

---

2. Why is the Column Space Important?

- **Represents the Output Space:**
  When A is viewed as a linear transformation $T(\mathbf{x}) = A\mathbf{x}$, the column space is the **range** of T.
  → It tells us all possible outputs $A\mathbf{x}$ for any input vector $\mathbf{x}$.
- **Solutions to $A\mathbf{x} = \mathbf{b}$:**
  The system $A\mathbf{x} = \mathbf{b}$ has a solution **if and only if $\mathbf{b}$** is in the column space of A.
- **Dimension = Rank:**
  The dimension of $\mathrm{Col}(A)$ is the **rank** of A, which tells us the number of linearly independent columns.

---

3. Key Properties

- If A is $m \times n$, then $\mathrm{Col}(A)$ is a subspace of $\mathbb{R}^m$.
- The pivot columns of A (from its reduced row echelon form) form a basis for $\mathrm{Col}(A)$.
- If $\mathrm{Col}(A) = \mathbb{R}^m$, then A is **full row rank**, meaning $A\mathbf{x} = \mathbf{b}$ has at least one solution for every $\mathbf{b}$.

---

4. Visualizing the Column Space

- For a 2×2 matrix, the column space is either:
  - A **plane** (if the columns are linearly independent).
  - A **line** (if one column is a multiple of the other).
  - Just the **zero vector** (if both columns are zero).
- For a 3×2 matrix, the column space could be:
  - A **plane** in $\mathbb{R}^3$ (if the two columns are independent).
  - A **line** in $\mathbb{R}^3$ (if they are dependent).

---

5. Connection with Other Concepts

- **Null Space:** The null space $\mathrm{Nul}(A)$ contains all $\mathbf{x}$ such that $A\mathbf{x} = \mathbf{0}$.
  → The column space and null space are orthogonal complements in $\mathbb{R}^n$ (by the Fundamental Theorem of Linear Algebra).
- **Row Space:** The row space is the span of A's rows (or A^T's columns).
  → For any matrix, $\mathrm{rank}(A) = \dim(\mathrm{Col}(A)) = \dim(\mathrm{Row}(A))$.

---

6. Practical Implications

- **Data Science:** In PCA (Principal Component Analysis), the column space helps identify the directions of maximum variance.
- **Engineering:** In control theory, the column space determines reachable states of a system.
- **Computer Graphics:** The column space defines the possible transformations of vectors (e.g., rotations, scaling).

Summary

The column space captures **all possible outputs** of A when multiplied by any vector $\mathbf{x}$. Understanding it helps in solving linear systems, analyzing transformations, and grasping the rank and structure of a matrix.

Column Space – Super Intuitive Explanation 🚀

Imagine you have a matrix A as a **"machine"** that takes in vectors and spits out new vectors. The **column space** is simply **all the possible outputs** this machine can produce.

---

1. The Matrix as a "Vending Machine" 🍫

- Think of a matrix A like a vending machine with **columns as buttons**.
- Each column is a **fixed output vector** (like a snack).
- Pressing buttons (scaling columns) and combining them (adding) gives you **all possible snacks** the machine can dispense.

Example:
Let $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$.

- **Column 1** = $\begin{bmatrix} 1 \\ 3 \end{bmatrix}$ (Chips)
- **Column 2** = $\begin{bmatrix} 2 \\ 4 \end{bmatrix}$ (Chocolate)

The **column space** = All combos of Chips + Chocolate:

$$a \cdot \text{Chips} + b \cdot \text{Chocolate}$$

→ This fills the entire 2D plane (since the snacks aren't redundant).

But if $A = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$, the 2nd column is just **2× the 1st**. Now:

- The machine **only dispenses snacks along a single line** (no variety!).
  → Column space = a line.

---

2. Why Does This Matter? 🔍

- **Can we solve $A\mathbf{x} = \mathbf{b}$?**
  → Only if $\mathbf{b}$ is in the column space!
  (i.e., Can the machine produce $\mathbf{b}$? If yes, solution exists.)
- **Is the matrix "useful"?**
  → If column space is tiny (e.g., just a line in 3D), the matrix is **"lame"** (low rank).
  → If it fills the whole space, it's **"powerful"** (full rank).

---

3. Visualizing Column Space 🎨

- **In 2D:**
  - If columns point in different directions → fills **entire plane**.
  - If columns are parallel → fills **just a line**.
- **In 3D:**
  - 2 columns → fills a **plane** (if they're not parallel).
  - 3 columns → fills the **whole 3D space** (if they're independent).

---

4. Punchline 🎯

The column space is **all the stuff the matrix can generate**.

- Big column space = more "creative" matrix.
- Small column space = "limited" matrix.

It's like asking: **"What can this machine do?"**
The answer lies in its columns!

How to Find the Column Space (with Numerical Example)

To find the **column space** of a matrix A, follow these steps:

1. **Identify the columns** of A.
2. **Determine which columns are linearly independent** (the "pivot columns").
3. **The column space is the span of these independent columns**.

Example: Find the Column Space of

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{bmatrix}$$

Step 1: Write the Columns

The matrix A has 3 columns:

$$\mathbf{c}_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \quad \mathbf{c}_2 = \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix}, \quad \mathbf{c}_3 = \begin{bmatrix} 3 \\ 6 \\ 9 \end{bmatrix}$$

Step 2: Check Linear Independence

- Notice that $\mathbf{c}_2 = 2\mathbf{c}_1$ and $\mathbf{c}_3 = 3\mathbf{c}_1$.
- **Only $\mathbf{c}_1$ is independent**; the others are just scaled copies.

Step 3: Column Space = Span of Independent Columns

$$\mathrm{Col}(A) = \mathrm{Span}\left(\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}\right)$$

- This is a **line** in $\mathbb{R}^3$ along the vector $\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$.

Verification via RREF (Row Reduction)

Let's confirm by row-reducing A:

$$\text{RREF}(A) = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- **Only the first column has a pivot**, so the column space is spanned by the **first original column** of A.

Another Example (Full Column Space)

$$B = \begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix}$$

Step 1: Write the Columns

$$\mathbf{b}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{b}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{b}_3 = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix}$$

Step 2: Check Linear Independence

- No column is a multiple of another.
- **All 3 columns are independent** (you can check with RREF).

Step 3: Column Space

$$\text{Col}(B) = \text{Span}\left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} \right) = \mathbb{R}^3$$

- Since all 3 columns are independent, they span the **entire 3D space**.

🔷 What is Column Space?

Formal Definition:
The **column space** of a matrix A is the set of all **possible vectors** you can get by multiplying A by any input vector $\vec{x}$.

$$\text{Column space of } A = \text{span of the columns of } A$$

That means:
Column space = All the **linear combinations** of the columns of the matrix.

🧠 Intuition: What Is It Really?

Imagine a matrix A as a **machine** made up of its **columns**.

- You give it some weights (a vector $\vec{x}$).
- The machine uses those weights to **combine** its columns.
- The output is a new vector — that's inside the **column space**.

So, the column space is:

> "Everything the matrix can **reach** or **create** using its columns."

Null Space

📊 Example:

Let’s say we have:

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

We want to find $\vec{x}$ such that:

$$A\vec{x} = \vec{0} \Rightarrow \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Solving this gives:

$$x_1 + 2x_2 = 0 \Rightarrow x_1 = -2x_2$$

So null space = all vectors of the form:

$$\begin{bmatrix} -2t \\ t \end{bmatrix} \text{ for all real } t$$

This means:

- There is **one full direction** that gets squashed to zero.
- That direction lies along the line $[-2, 1]$.

✨ Geometric Intuition

If you're in:

- **2D**, and the null space is a **line** → matrix squashes 2D into 1D.
- **3D**, and null space is a **plane** → matrix flattens 3D into a line.
- If the null space is **only the zero vector**, the matrix doesn’t erase anything — it's **full rank**.

Orthogonality & Projections – Intuitive Explanation with Examples

---

1. Orthogonality (Perpendicularity)

Two vectors are **orthogonal** if they meet at a **90° angle**. Mathematically:

$$\mathbf{u} \cdot \mathbf{v} = 0$$

Why?
The dot product $\mathbf{u} \cdot \mathbf{v} = \|\mathbf{u}\|\|\mathbf{v}\| \cos \theta$. If $\theta = 90^\circ$, $\cos \theta = 0$.

Example:
Let $\mathbf{u} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ and $\mathbf{v} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$.

$$\mathbf{u} \cdot \mathbf{v} = (1)(-1) + (1)(1) = 0$$

→ They're orthogonal!

📘 Projections in Linear Algebra – Complete Concept

🔷 Definition:

In **Linear Algebra**, a **projection** refers to the process of mapping a vector onto another vector or a subspace in such a way that the **difference between the original and projected vector is orthogonal** to the subspace.

🔷 Projection of a Vector onto Another Vector:

Let $\vec{a}$ and $\vec{b}$ be vectors in $\mathbb{R}^n$, and $\vec{a} \neq 0$.

The **projection of $\vec{b}$ onto $\vec{a}$** is:

$$\text{proj}_{\vec{a}} \vec{b} = \frac{\vec{a} \cdot \vec{b}}{\vec{a} \cdot \vec{a}} \vec{a}$$

- Result is a **vector** in the **direction of $\vec{a}$**.
- Represents the component of $\vec{b}$ that lies **along** $\vec{a}$.

📊 Example:

Let's say:

$$\vec{a} = \begin{bmatrix} 3 \\ 0 \end{bmatrix}, \quad \vec{b} = \begin{bmatrix} 4 \\ 5 \end{bmatrix}$$

Project $\vec{b}$ onto $\vec{a}$:

$$\text{proj}_{\vec{a}}\vec{b} = \frac{\vec{a} \cdot \vec{b}}{\vec{a} \cdot \vec{a}}\vec{a} = \frac{(3)(4) + (0)(5)}{(3)^2} \cdot \vec{a} = \frac{12}{9} \cdot \vec{a} = \frac{4}{3} \cdot \begin{bmatrix} 3 \\ 0 \end{bmatrix} = \begin{bmatrix} 4 \\ 0 \end{bmatrix}$$

So, the **shadow of $\vec{b}$** on $\vec{a}$ is $\begin{bmatrix} 4 \\ 0 \end{bmatrix}$.

🧠 Summary Table

Concept	Meaning	Formula / Clue
Orthogonality	Vectors are at 90°; dot product = 0	$\vec{a} \cdot \vec{b} = 0$
Projection	"Shadow" of one vector onto another	$\text{proj}_{\vec{a}}\vec{b} = \frac{\vec{a} \cdot \vec{b}}{\vec{a} \cdot \vec{a}}\vec{a}$
Use Together	Project onto a space; residual is orthogonal	Used in **least squares**, regression

🔷 Standard Definition

Let A be a **square matrix** of size $n \times n$.

A **non-zero vector** $\vec{v}$ is called an **eigenvector** of A, and a **scalar** λ is its corresponding **eigenvalue**, if:

$$A\vec{v} = \lambda\vec{v}$$

- A: linear transformation (matrix)
- $\vec{v}$: direction that doesn't change under A
- λ: amount by which $\vec{v}$ is stretched or shrunk

🧠 Intuition

Imagine applying a transformation (like rotation, scaling, etc.) to space:

- **Most vectors** change direction **and** length.
- But **eigenvectors** are **special** — they only **scale**, not rotate.
- The **eigenvalue** tells **how much** they're scaled.

> Think of pressing on a rubber sheet: some directions just stretch — those are eigenvectors.

🧮 How to Find Them

Given $A\vec{v} = \lambda\vec{v}$, rewrite as:

$$(A - \lambda I)\vec{v} = 0$$

To find λ:

$$\det(A - \lambda I) = 0$$

This is the **characteristic equation**, whose solutions are the **eigenvalues**.

Then plug each λ back into:

$$(A - \lambda I)\vec{v} = 0$$

to solve for the corresponding **eigenvectors** $\vec{v}$.

📌 Applications

- **PCA (Principal Component Analysis)**: Data directions of maximum variance
- **Differential equations**: System stability
- **Quantum mechanics**: Energy states
- **Google PageRank**: Uses eigenvectors!
- **Face recognition, Vibration analysis**, etc.

Eigen Decomposition (Diagonalizing a Matrix)

The process aims to decompose a square matrix *A* into a product of three matrices:

- **P:** A **matrix whose columns are the eigenvectors of *A*.**
- **Λ (Lambda):** A **diagonal matrix whose diagonal entries are the corresponding eigenvalues of *A*.**
- **P^{-1}: The inverse of the eigenvector matrix *P*.**

The equation you provided:

$A=P\Lambda P^{-1}$

expresses this relationship.

🔷 Example Matrix

Let's take a simple 2×2 matrix:

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

✅ Step 1: Find the Characteristic Polynomial

We want to solve:

$$\det(A - \lambda I) = 0$$

$$\begin{vmatrix} 4-\lambda & 1 \\ 2 & 3-\lambda \end{vmatrix} = 0$$

$$(4-\lambda)(3-\lambda) - 2(1) = 0$$

$$(12 - 4\lambda - 3\lambda + \lambda^2) - 2 = 0$$

$$\lambda^2 - 7\lambda + 10 = 0$$

✅ Step 2: Solve for Eigenvalues

$$\lambda^2 - 7\lambda + 10 = 0$$

Factor:

$$(\lambda - 5)(\lambda - 2) = 0$$

So the eigenvalues are:

✅ Step 3: Find Eigenvectors

For $\lambda = 5$:

Solve $(A - 5I)\vec{v} = 0$

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

This simplifies to:

$$-1x + 1y = 0 \Rightarrow y = x$$

So, one eigenvector is:

$$\vec{v}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

For $\lambda = 2$:

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

$$2x + y = 0 \Rightarrow y = -2x$$

So, one eigenvector is:

$$\vec{v}_2 = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$$

✅ Step 4: Construct the Decomposition $A = PDP^{-1}$

Let:

$$P = \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}, \quad D = \begin{bmatrix} 5 & 0 \\ 0 & 2 \end{bmatrix}$$

Now calculate:

$$P^{-1} = \frac{1}{(1)(-2) - (1)(1)} = \frac{1}{-2 - 1} = \frac{1}{-3} \begin{bmatrix} -2 & -1 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & -\frac{1}{3} \end{bmatrix}$$

✅ So the eigen decomposition is:

$$A = PDP^{-1}$$

Where:

- $P = \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}$
- $D = \begin{bmatrix} 5 & 0 \\ 0 & 2 \end{bmatrix}$
- $P^{-1} = \begin{bmatrix} \frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & -\frac{1}{3} \end{bmatrix}$

🔷 Given Matrix P:

$$P = \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}$$

✅ Step-by-step Inverse of a 2×2 Matrix

For any matrix:

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

The inverse is:

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

🔶 Step 1: Compute the determinant

$$\det(P) = (1)(-2) - (1)(1) = -2 - 1 = -3$$

🔶 Step 2: Swap and negate elements

From:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}$$

We get:

$$\begin{bmatrix} -2 & -1 \\ -1 & 1 \end{bmatrix}$$

🔶 Step 3: Multiply by $\frac{1}{\det} = \frac{1}{-3}$

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

✅ Final Answer:

$$P^{-1} = \begin{bmatrix} \frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & -\frac{1}{3} \end{bmatrix}$$

3. Key Differences

Feature	Eigen Decomposition	SVD
Matrix Type	Square, diagonalizable	Any ($m \times n$)
Output	$A = P\Lambda P^{-1}$	$A = U\Sigma V^T$
Vectors	Eigenvectors (may not be orthogonal)	Singular vectors (always orthonormal)
Stability	Can fail for defective matrices	Always works numerically

---

4. Applications

Eigen Decomposition

- Solving differential equations ($e^{At} = Pe^{\Lambda t}P^{-1}$).
- Quantum mechanics (Hamiltonian diagonalization).

SVD

- **PCA** (Principal Component Analysis).
- **Image compression** (low-rank approximation).
- **Recommendation systems** (collaborative filtering).

Summary

- **Eigen Decomposition:** Great for square matrices, but limited.
- **SVD:** Universal, stable, and works on any matrix.
- **Analogy:**
  - Eigen = "Matrix's natural axes."
  - SVD = "Best way to break any matrix into stretch + rotate."

🔷 2. SVD (Singular Value Decomposition)

📘 Definition:

SVD decomposes **any** $m \times n$ matrix A as:

$$A = U\Sigma V^T$$

Where:

- $U \in \mathbb{R}^{m\times m}$: orthogonal matrix (left singular vectors)
- $\Sigma \in \mathbb{R}^{m\times n}$: diagonal matrix with **singular values**
- $V^T \in \mathbb{R}^{n\times n}$: transpose of orthogonal matrix (right singular vectors)

🧠 Intuition:

SVD breaks a matrix into **rotation/stretch/rotation**:

> "Rotate → Stretch → Rotate again"

- It tells us how the matrix transforms the space.
- Singular values tell how much stretching happens along each axis.

🎬 Analogy:

Imagine a piece of dough:

- V^T: rotate the dough
- Σ: stretch it

✅ Eigen Decomposition (Diagonalizing a Matrix)

🔍 What is it?

Eigen Decomposition is the process of expressing a **square matrix** A in terms of its **eigenvalues** and **eigenvectors**.

It writes a matrix A as:

$$A = PDP^{-1}$$

Where:

- A is the original **n × n** matrix.
- D is a **diagonal matrix** of eigenvalues.
- P is the matrix whose **columns are the eigenvectors** of A.
- P^{-1} is the inverse of matrix P.

📌 When is it possible?

Eigen decomposition is possible if:

- Matrix A is **diagonalizable**.
- A has **n linearly independent eigenvectors**.

> ✅ All **symmetric** matrices are diagonalizable.

🧠 Why is it useful?

- Makes **matrix powers** easier: $A^k = PD^kP^{-1}$
- Used in **PCA (Principal Component Analysis)**, **differential equations**, and **quantum mechanics**
- Helps understand the **structure and transformation** of the matrix

---

🧮 Step-by-Step Process

Given matrix A:

1. **Find Eigenvalues**: Solve the **characteristic equation**:

$$\det(A - \lambda I) = 0$$

   to find $\lambda_1, \lambda_2, \ldots, \lambda_n$

2. **Find Eigenvectors**: For each eigenvalue λ, solve:

$$(A - \lambda I)v = 0$$

3. **Form Matrices**:
   - $D =$ diagonal matrix of eigenvalues
   - $P =$ matrix with eigenvectors as columns

4. **Verify**:

$$A = PDP^{-1}$$

✅ **Example**

Let:

$$A = \begin{bmatrix} 4 & 1 \\ 0 & 3 \end{bmatrix}$$

1. **Eigenvalues:**

$$\det(A - \lambda I) = (4 - \lambda)(3 - \lambda) = 0 \Rightarrow \lambda_1 = 4, \lambda_2 = 3$$

2. **Eigenvectors:**
   - For $\lambda = 4$: $(A - 4I)v = 0 \Rightarrow v_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$
   - For $\lambda = 3$: $(A - 3I)v = 0 \Rightarrow v_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$

3. **Form matrices:**

$$P = \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix}, \quad D = \begin{bmatrix} 4 & 0 \\ 0 & 3 \end{bmatrix}$$

4. **Then:**

$$A = PDP^{-1}$$

Technique	Core Idea	Key Applications
SVD	Decompose any matrix	Compression, NLP, Recommender Systems, Denoising
Eigen Decomposition	Decompose square matrix via eigenvectors	PCA, control systems, quantum physics, facial recognition

LEAST SQUARES

We are fitting the line

$$y = mx + b$$

to the data points:

$$(1, 2), (2, 3), (3, 5)$$

This means:

- At $x = 1$, we want $y \approx 2$
- At $x = 2$, we want $y \approx 3$
- At $x = 3$, we want $y \approx 5$

Step 1: Setup in Matrix Form

We write the equation for all points:

$$\begin{aligned} b + m(1) &= 2 \\ b + m(2) &= 3 \\ b + m(3) &= 5 \end{aligned}$$

This is the same as:

$$\begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} b \\ m \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ 5 \end{bmatrix}$$

So we have:

- $A = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix}$
- $x = \begin{bmatrix} b \\ m \end{bmatrix}$ (what we want to find)
- $\mathbf{b} = \begin{bmatrix} 2 \\ 3 \\ 5 \end{bmatrix}$

Step 2: Use Least Squares (Normal Equation)

We solve using:

$$A^T A x = A^T \mathbf{b}$$

Compute $A^T A$:

$$A^T A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} = \begin{bmatrix} 3 & 6 \\ 6 & 14 \end{bmatrix}$$

Compute $A^T \mathbf{b}$:

$$A^T \mathbf{b} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 5 \end{bmatrix} = \begin{bmatrix} 10 \\ 23 \end{bmatrix}$$

Step 3: Solve the Linear System

Now solve:

$$\begin{bmatrix} 3 & 6 \\ 6 & 14 \end{bmatrix} \begin{bmatrix} b \\ m \end{bmatrix} = \begin{bmatrix} 10 \\ 23 \end{bmatrix}$$

Use any method (like substitution, elimination, or inverse). We'll solve:

From the first equation:

$$3b + 6m = 10 \quad (1)$$

From the second:

$$6b + 14m = 23 \quad (2)$$

Multiply (1) by 2:

$$6b + 12m = 20 \quad (3)$$

Now subtract (2) - (3):

$$(6b + 14m) - (6b + 12m) = 23 - 20 \Rightarrow 2m = 3 \Rightarrow m = 1.5$$

Plug back into (1):

$$3b + 6(1.5) = 10 \Rightarrow 3b + 9 = 10 \Rightarrow 3b = 1 \Rightarrow b = \frac{1}{3}$$

✅ Final Answer:

- Intercept $b = \frac{1}{3}$
- Slope $m = 1.5$

So, the best-fit line is:

$$y = 1.5x + \frac{1}{3}$$