

explain these topics one by one try to cover hidden aspects of them

Linear Algebra Basics

Vectors, matrices, operations

Linear equations & Gaussian elimination

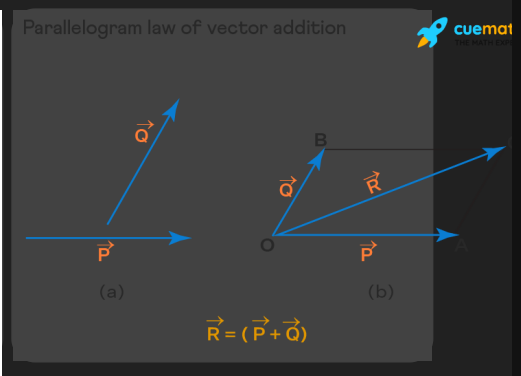
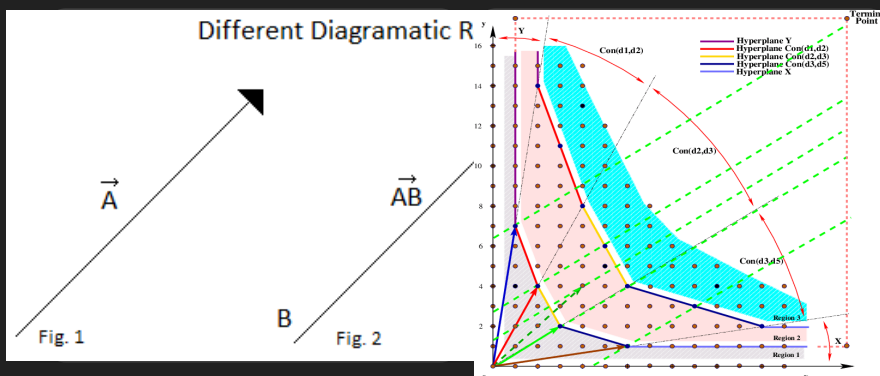
Determinants, rank, inverses

Intro to eigenvalues & eigenvectors

Great — now we start **Linear Algebra Basics**, but in a way that makes every concept *crystal clear* and deeply connected, with hidden aspects revealed.

I will explain **one topic at a time**, with intuitive geometry + formal math + connections to later ML/AI topics.

1. VECTORS — The True Foundation



1.1 What a Vector *Really* Is

A vector is **not** just an arrow.

A vector is:

- a direction
- a magnitude
- a list of numbers representing a point direction in coordinate space
- an element of a **vector space** (this is the hidden part)

Deep meaning:

A vector has only one requirement:

If you scale it and add it to other vectors, the result must still be a vector.

This abstract view is used everywhere in ML:

- word embeddings \rightarrow vectors
 - images \rightarrow high-dim vectors
 - weights in neural networks \rightarrow vectors
 - gradients \rightarrow vectors
-

1.2 Vector Representation

As coordinates

$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

As an arrow from origin

Same information, interpreted geometrically.

1.3 Vector Operations (with hidden insights)

Addition

$$\mathbf{u} + \mathbf{v} = \begin{bmatrix} u_1 + v_1 \\ u_2 + v_2 \end{bmatrix}$$

Hidden meaning:

Vector addition is *translation* in space. Everything in 3D games, robotics, and physics uses this.

Scalar Multiplication

$$c\mathbf{v} = \begin{bmatrix} cv_1 \\ cv_2 \end{bmatrix}$$

Hidden meaning:

Scaling changes magnitude but not direction.

Negative scaling flips direction.

Dot Product (VERY IMPORTANT)

$$\mathbf{u} \cdot \mathbf{v} = \sum u_i v_i = \|\mathbf{u}\| \|\mathbf{v}\| \cos \theta$$

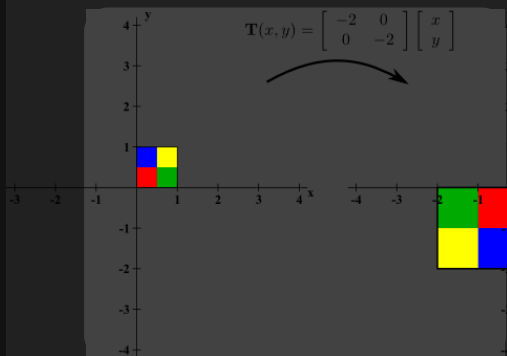
Hidden insights:

- Measures **alignment** ($\cos \theta$)
- $= 0 \rightarrow$ vectors are perpendicular
- Basis of:
 - cosine similarity (NLP)
 - projections (PCA)
 - gradients (optimization)

Cross Product (3D only)

Gives a vector perpendicular to both \rightarrow used in rotation, 3D graphics, physics.

2. MATRICES — Transformations and Feel



$$\text{Rot}_{135^\circ} \left(\begin{bmatrix} 3 \\ 2 \end{bmatrix} \right) = \begin{bmatrix} -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

$$\text{Rot}_{135^\circ} \left(\begin{bmatrix} 3 \\ 2 \end{bmatrix} \right) = \begin{bmatrix} -\frac{3\sqrt{2}}{2} & -\frac{2\sqrt{2}}{2} \\ \frac{3\sqrt{2}}{2} & -\frac{2\sqrt{2}}{2} \end{bmatrix}$$

$$\text{Rot}_{135^\circ} \left(\begin{bmatrix} 3 \\ 2 \end{bmatrix} \right) = \begin{bmatrix} -\frac{5\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}$$

Understanding the Matrix of a Linear Map

$$\mathcal{M}(T) = \begin{pmatrix} w_1 & \dots & w_k & \dots & w_n \\ \vdots & & \vdots & & \vdots \\ w_m & & A_{m,k} & & \end{pmatrix}$$

The k^{th} column of $\mathcal{M}(T)$ consists of the scalars needed to write Tv_k as a linear combination of w_1, \dots, w_m :

$$Tv_k = \sum_{j=1}^m A_{j,k} w_j$$

The picture above should remind you that Tv_k can be computed from $\mathcal{M}(T)$ by multiplying each entry in the k^{th} column by the corresponding w_j from the left column, and then adding up the resulting vectors.

2.1 What a Matrix Actually Represents

A matrix is:

- a table of numbers
- a transformer of vectors
- a system of linear equations
- a mapping between spaces

Hidden meaning:

Every matrix is a **linear transformation** — it stretches, rotates, shears, or reflects space.

This is the core of neural networks:

- weight matrices transform inputs between layers
- convolutions are structured matrices
- PCA uses matrix rotations

2.2 Matrix Multiplication

$$Ax$$

Hidden meaning:

Matrix multiplication is **function composition**.

If:

- A = rotate
- B = scale

Then AB = scale then rotate.

This is why order matters.

2.3 Matrix Operations

Addition and Scalar multiplication — elementwise.

Matrix Multiplication — not elementwise (hidden rule):

Rows interact with columns.

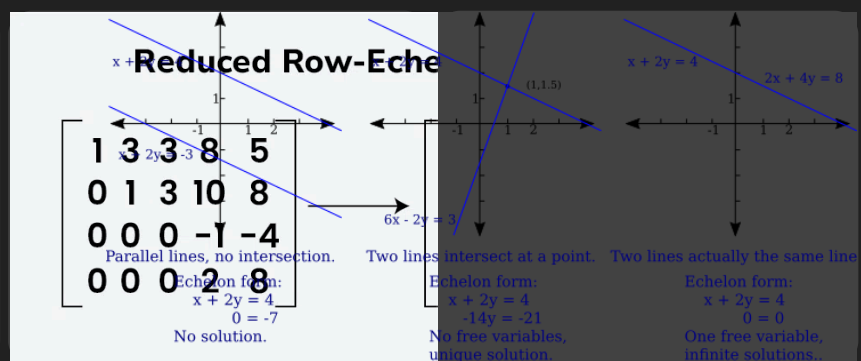
This is why we multiply weights with inputs.

3. LINEAR EQUATIONS & GAUSSIAN ELIMINATION

Gaussian Elimination

$$\begin{aligned} -z &= -2 \\ x + z &= 5 \\ x + 2z &= 1 \end{aligned} \rightarrow \left[\begin{array}{ccc|c} 1 & 1 & -1 & -2 \\ 0 & 1 & 1 & 5 \\ 0 & 0 & 2 & 1 \end{array} \right]$$

$$\begin{aligned} -z &= 2 \\ -z &= -3 \\ z &= 2 \end{aligned} \leftarrow \left[\begin{array}{ccc|c} 1 & 1 & -1 & 2 \\ 0 & 1 & -1 & -3 \\ 0 & 0 & 1 & 2 \end{array} \right]$$



3.1 System of Equations

Example:

$$\begin{aligned}x + 2y &= 5 \\ 3x + y &= 4\end{aligned}$$

Written in matrix form:

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

3.2 Gaussian Elimination

Goal: turn A into a simpler matrix using **row operations**.

Row operations (hidden meaning):

- swap → reorganize equations
- multiply row → scale equation
- add a multiple → eliminate variable

Resulting form:

- **Row Echelon Form (REF)**
- **Reduced Row Echelon Form (RREF)**

Used to compute:

- solutions
- inverses
- rank
- consistency of system

Hidden connection:

This is exactly what happens inside **LU decomposition** and many ML solvers.

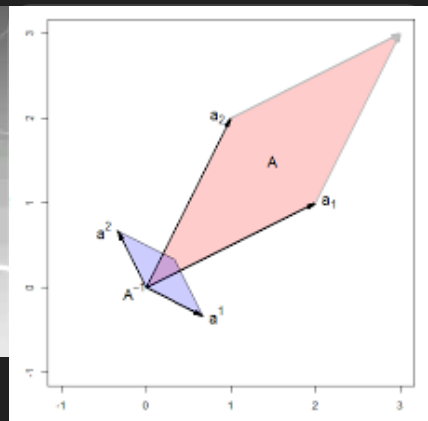
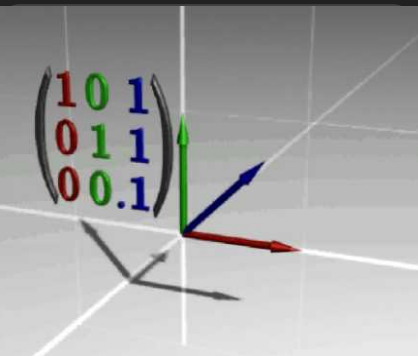
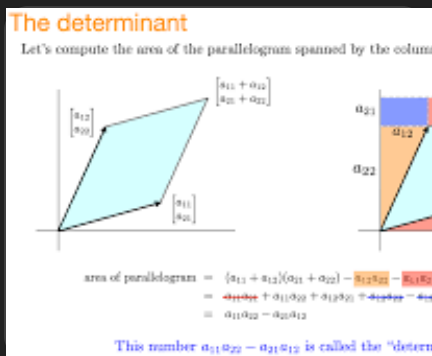
3.3 Solution Types

1. **Unique solution** → full rank
 2. **Infinite solutions** → dependent equations
 3. **No solution** → inconsistent equations
-
-

4. DETERMINANTS, RANK, INVERSE — 4 Pillars of Matrix Understanding

The determinant

Let's compute the area of the parallelogram spanned by the columns



4.1 Determinant = Area/Volume Scaling

For 2×2 :

$$\det(A) = ad - bc$$

Hidden truth:

- \det = how much the matrix scales space
- $\det = 0 \rightarrow$ matrix collapses space \rightarrow not invertible

This is crucial in:

- change of variables in integrals
- Jacobians
- probabilistic models (normalizing flows)

4.2 Rank

Rank = number of independent rows/columns.

Hidden meaning:

- dimension of output space A can reach
- number of meaningful directions

Full rank \rightarrow invertible.

Low rank \rightarrow matrix compresses space:

- PCA
- SVD
- embeddings

- dimensionality reduction

4.3 Matrix Inverse

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

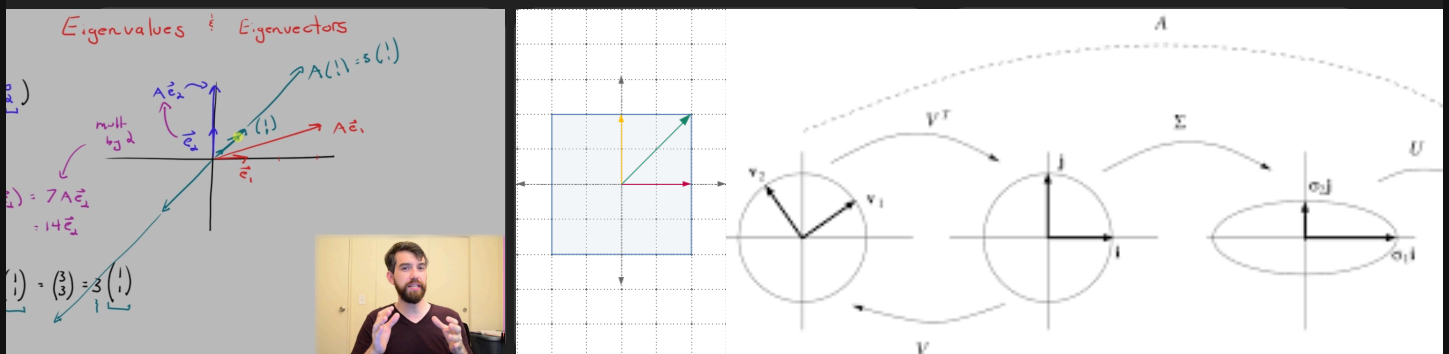
Hidden meaning:

- inverse "undoes" the transformation
- exists only when $\det \neq 0$ and matrix is full rank

Used in:

- solving equations
- optimization (Newton's method)
- covariance matrices

5. INTRO TO EIGENVALUES & EIGENVECTORS



5.1 The equation

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

Meaning:

- A transforms \mathbf{v} into a scaled version of itself
- \mathbf{v} = eigenvector
- λ = eigenvalue

Geometric hidden meaning:

Eigenvectors are directions that *do not rotate* under the transformation.

5.2 Why are they important?

They reveal:

- natural modes of a system
 - stable/unstable directions
 - principal components (PCA)
 - spectral clustering
 - Markov chain steady states
 - quantum mechanics energy levels
-

5.3 Diagonalization

If:

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

Then A can be understood as:

- change basis → diagonal → apply scaling → return basis

This simplifies:

- powers of matrices
 - differential equations
 - linear dynamical systems
-

✓ NEXT STEP

If you want, I can now go **deeper into each topic one by one**, with:

- geometric animations
- algebraic derivations
- hidden connections to ML/AI
- problems + solutions
- topic-by-topic mastery tests

Which topic should I expand next?

Vectors,

Matrices,

Linear equations, or

Eigenvalues/eigenvectors?