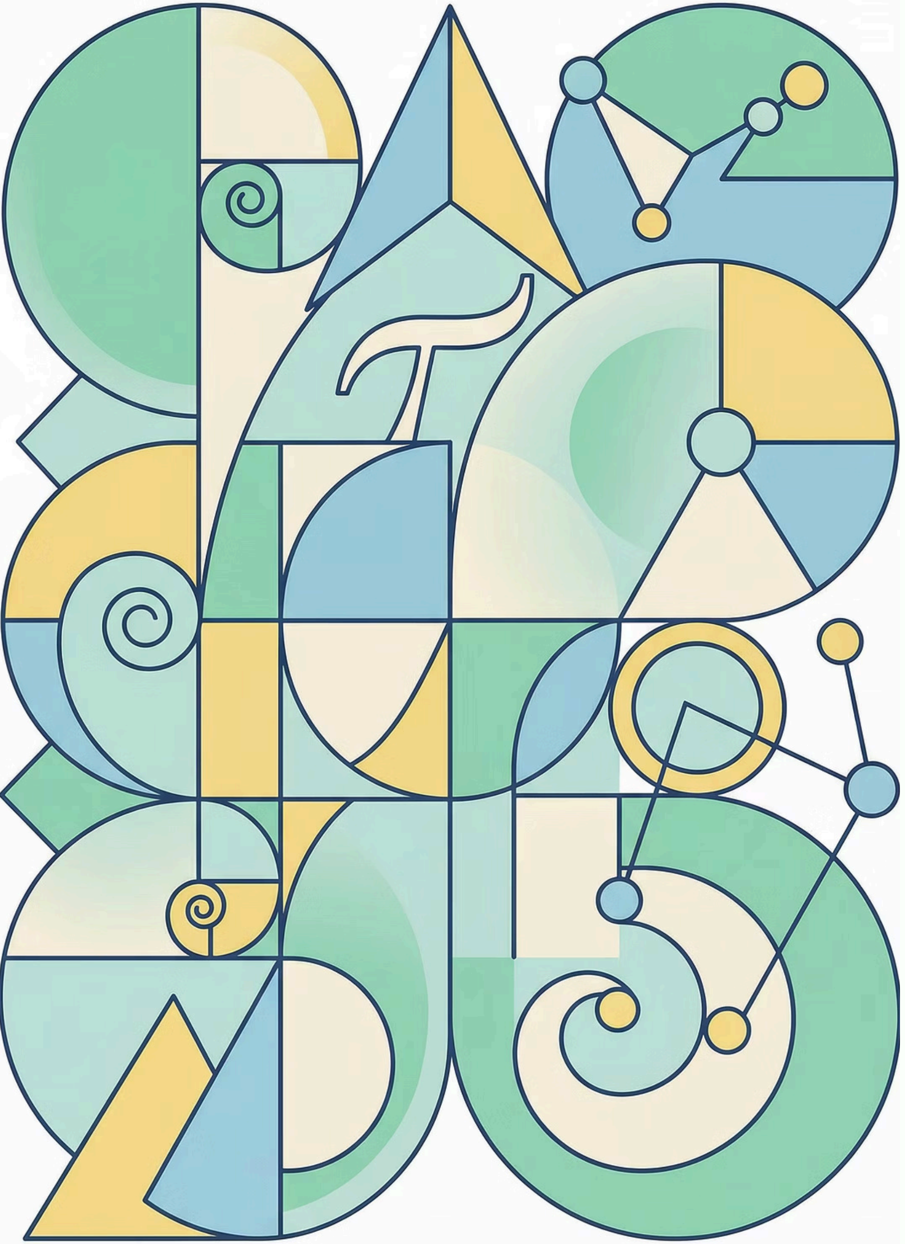


Linear Algebra Made Simple

A friendly introduction to scalars, vectors, matrices, and array operations with easy examples you can work through in your head.

Understanding the Building Blocks

Scalars

Just a single number on its own

$$a = 5$$

Vectors

A list of numbers arranged in order

$$\mathbf{v} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

Matrices

A rectangle filled with numbers in rows and columns

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

These three types form the foundation of linear algebra. Think of scalars as single values, vectors as ordered lists, and matrices as organised grids.

Let's Try Vector Addition

The Question

Given the vector:

$$\mathbf{v} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

Add 2 to every entry.

The Answer

Simply add 2 to each number:

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

Each element increases by the same amount.

Matrix Operations Are Similar

Question

Take this matrix and add 1 to every entry:

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

The Process

Add 1 to each number in the grid, maintaining the same structure and position.

The Result

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

Understanding Shapes and Dimensions

01

Reading Matrix Shape

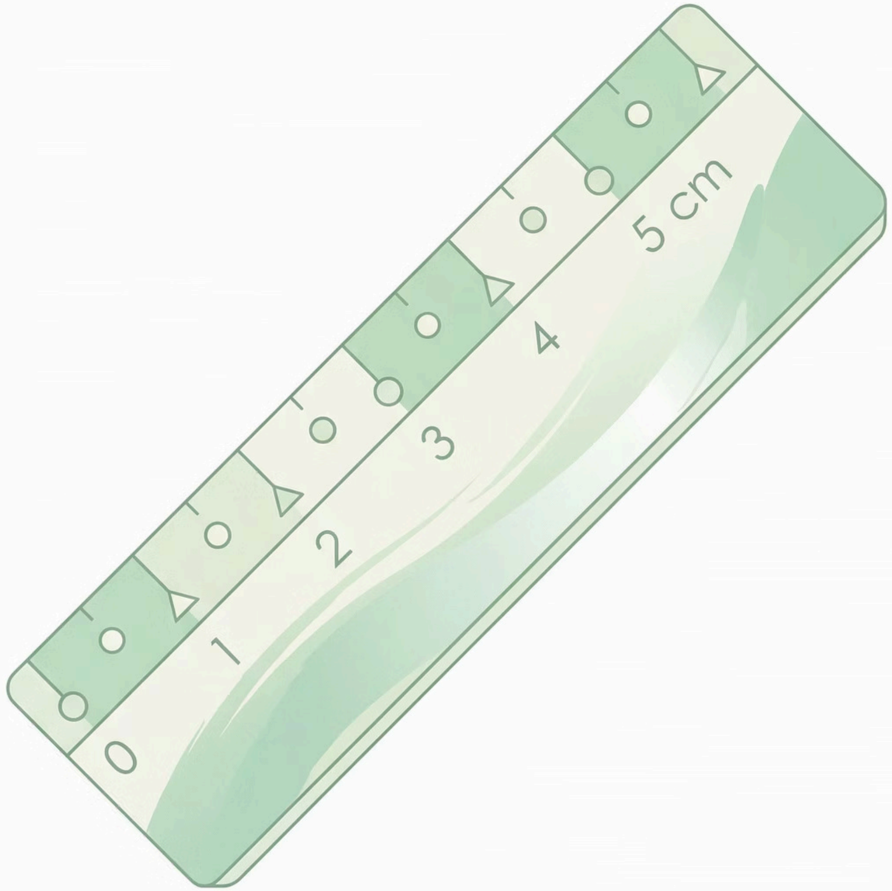
For matrix $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$, count rows first (2), then columns (3). The shape is 2×3.

02

Multi-dimensional Shapes

A batch of 10 images, each 28×28 pixels with 1 colour channel, has shape $(10, 28, 28, 1)$. The first number (10) represents the batch size.

Tip: The shape tells you the dimensions at each level. Think of it like giving directions: batch → height → width → channels.



Element-wise Operations

Vector Addition

Given two vectors:

$$\mathbf{x} = \begin{bmatrix} 1 \\ 3 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 4 \\ 5 \end{bmatrix}$$

Add them element by element:

$$\mathbf{x} + \mathbf{y} = \begin{bmatrix} 1+4 \\ 3+5 \end{bmatrix} = \begin{bmatrix} 5 \\ 8 \end{bmatrix}$$

Scalar Multiplication

For matrix:

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

Multiply every entry by 2:

$$2A = \begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}$$

Element-wise operations work on corresponding positions. Each element operates independently with its counterpart.

The Dot Product Explained

1

Start with Two Vectors

$$\begin{bmatrix}1\\2\end{bmatrix}, \begin{bmatrix}3\\4\end{bmatrix}$$

2

Multiply Pairs

$1 \cdot 3 = 3$

$2 \cdot 4 = 8$

3

Add Results

$$3 + 8 = 11$$

That's your dot product!

The dot product combines two vectors into a single number by multiplying corresponding entries and summing them up.

Matrix-Vector Multiplication

Example Problem

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

Find AB

First Row Calculation

Dot product of first row with vector: $1 \cdot 1 + 2 \cdot 1 = 3$

Second Row Calculation

Dot product of second row with vector: $3 \cdot 1 + 4 \cdot 1 = 7$

Combine the Results

$$AB = \begin{bmatrix} 3 \\ 7 \end{bmatrix}$$

Broadcasting: NumPy's Clever Trick

Broadcasting automatically expands smaller arrays to match larger ones, making operations more intuitive.

Adding a Scalar

$$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + 10 = \begin{bmatrix} 11 \\ 12 \\ 13 \end{bmatrix}$$

The single number 10 "broadcasts" to match each element.

Adding a Row Vector

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 10 & 20 \end{bmatrix}$$

Result: $\begin{bmatrix} 11 & 22 \\ 13 & 24 \end{bmatrix}$

The row broadcasts across each matrix row.

Reshaping and Transposing

Reshaping Arrays

Take 6 numbers $[1, 2, 3, 4, 5, 6]$ and reshape to 2×3:

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

The total count stays the same (6 elements), just arranged differently.

The Transpose Operation

For matrix $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$, the transpose $A^\top$ flips rows and columns:

$$A^\top = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$$

Row 1 becomes column 1, and row 2 becomes column 2.

These operations change how we view data without changing the underlying values. They're essential tools in linear algebra and machine learning.