A CS Guide to Linear Algebra from Scratch

Strings, zach.waffle

September 2025

Abstract

This guide was made to help programmers learn linear algebra concepts with a CS perspective with as little background knowledge as possible. The main focus will be on understanding the underlying principles behind practical applications as well as how they interact with the theoretical ones.

Chapter 1

Introduction

Linear algebra is a section of math dedicated to solving systems of linear equations and understanding geometric concepts such as planes and lines. The underlying foundation of linear algebra is vectors and matrices (the plural of "matrix"). From a CS perspective, the use of linear algebra can often be seen in graphics, machine learning, and cryptography. This paper will be used to build from the basic fundamentals to an above surface knowledge of the topic from the math and its computational implementation.

Chapter 2

Scalars and Vectors

2.1 Scalar

A scalar refers to a normal number used in a vector operation. They are typically *real numbers ($\mathbb{R}$)* or *complex numbers ($\mathbb{C}$)*

2.2 Vectors

Vectors are one of the biggest building blocks of linear algebra. Vectors are often represented with an arrow (such as $\vec{v}$) or with bold (such as **v** At its core, a vector is just a list of objects. For example, consider the vector:

$$\vec{v} = \begin{bmatrix} 2 \\ 5 \\ -1 \end{bmatrix} \tag{2.1}$$

This vector has 3 components: 2, 5, -1.
Vectors are used to store information; a common use of a vector is to store a location.

2.2.1 Vector Sets

A *vector set* is a collection of vectors, a bundle of vectors put in a set. You think of them as "the vectors we're allowed to consider" *Example:*

$$S = \left\{ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\} \tag{2.2}$$

Here, the set S contains three vectors in 2D space. Anything not in S is not considered part of this particular set.

Later, we will see a **vector space**, which is a special type of vector set. In a vector space, you can combine vectors or stretch them, and you will never leave the set - everything still stays inside the set.

Set theory is a large field within mathematics, and there are many resources available online to help you learn it!

2.2.2 Vector Space

Now that we understand what a **vector set** is (a collection of vectors), let's talk about a **vector space** It is a special type of vector set where you can do two things freely:

1. **Add any two vectors** from the set, and the result is still in the set.
2. **Multiply any vector by a number** (scalar), and the result is still in the set.

In other words, a vector space is a 'magic basket' of vectors: - You can combine vectors or stretch/shrink them. - No matter what you do, you never leave the basket.

Example: All 2D vectors form a vector space:

$$\mathbb{R}^2 = \left\{ \begin{bmatrix} x \\ y \end{bmatrix} \middle| x, y \in \mathbb{R} \right\} \tag{2.3}$$

In non-Klingon, this means that a 2D vector contains x and y, and both x and y are real numbers.

- Add two vectors:

$$\begin{bmatrix}1\\2\end{bmatrix} + \begin{bmatrix}3\\4\end{bmatrix} = \begin{bmatrix}4\\6\end{bmatrix} \in \mathbb{R}^2 \tag{2.4}$$

- Multiply a vector by a scalar:

$$2 \cdot \begin{bmatrix}1\\2\end{bmatrix} = \begin{bmatrix}2\\4\end{bmatrix} \in \mathbb{R}2 \tag{2.5}$$

Notice how the results stay inside the set, that is, the defining feature of a vector space.

2.3 Representing Vectors

There are multiple ways to represent vectors. This section does not attempt to explain them all.

2.3.1 Algebraically

Algebraically, a vector is an element of a *vector space*.
For example, all 3D vectors form $\mathbb{R}^3$, the three-dimensional vector space.

The primary method of representing a vector is simply as a list of its components:

$$\vec{v} = \begin{bmatrix}3\\4\end{bmatrix} \tag{2.6}$$

- You can add component-wise:

$$\begin{bmatrix}1\\2\\3\end{bmatrix} + \begin{bmatrix}4\\5\\6\end{bmatrix} = \begin{bmatrix}5\\7\\9\end{bmatrix} \tag{2.7}$$

- And you can multiply by a scalar:

$$2 \cdot \begin{bmatrix}3\\-1\end{bmatrix} = \begin{bmatrix}6\\-2\end{bmatrix} \tag{2.8}$$

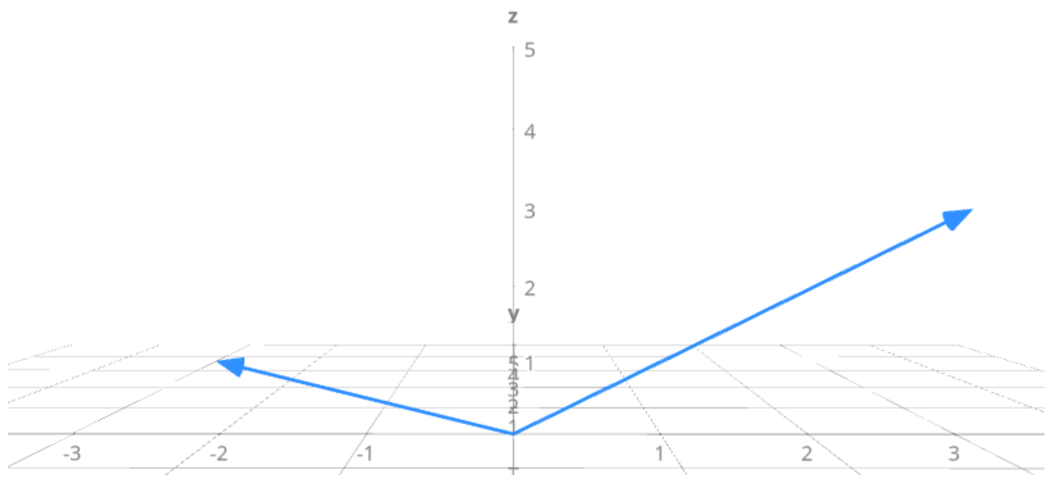

Figure 2.1: Geometric representation of 2D Vectors.

2.3.2 Geometrically

Geometrically, a vector can be visualized as an arrow going from the origin to a point. Example:

$$\vec{v}_1 = \begin{bmatrix} 3 \\ 3 \end{bmatrix} \vec{v}_2 = \begin{bmatrix} -2 \\ 1 \end{bmatrix} \tag{2.9}$$

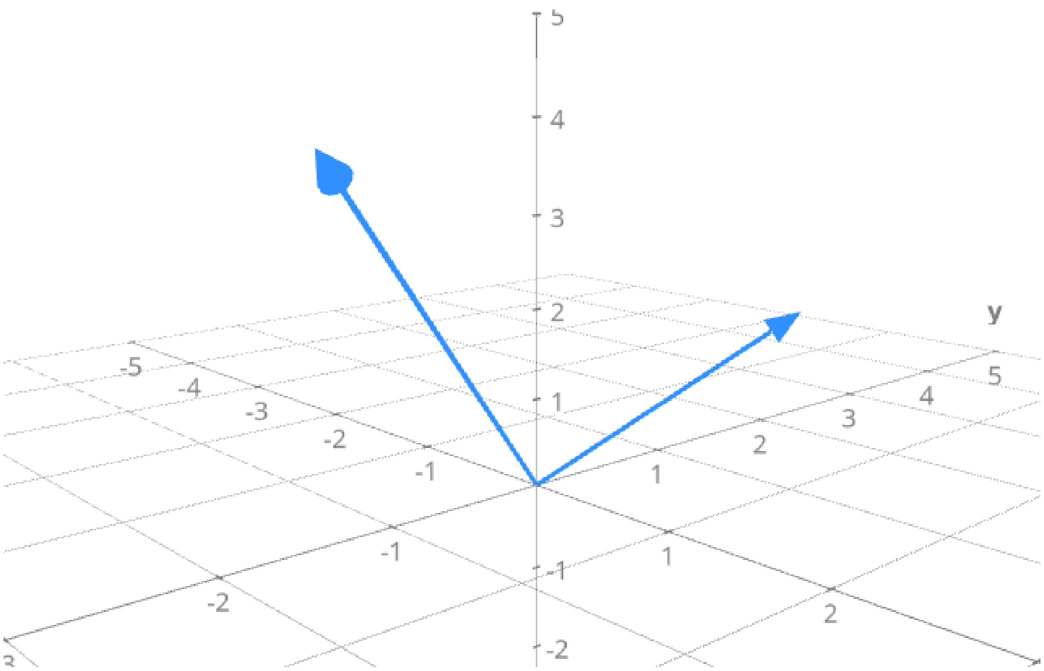

Figure 2.2: Geometric representation of 3D Vectors.

The same principle works with a 3D vector but pointing to (x, y, z)
Example:

$$\vec{v}_1 = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \vec{v}_2 = \begin{bmatrix} 1 \\ -2 \\ 4 \end{bmatrix} \tag{2.10}$$

Geometrically, vectors can also be thought of as the sum of scalar multiples of the *unit vectors*: $\hat{\imath} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$, $\hat{\jmath} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$, and $\hat{k} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$.

A vector $\vec{v} = \begin{bmatrix} 2 \\ -3 \end{bmatrix}$ can also be written as $\vec{v} = 2\hat{\imath} - 3\hat{\jmath}$

Programmatically

In Computer Science, common usages of vectors can be:

- A list/array

- Method to represent states (e.g., pixel colors, probability, features in ML)
- A direction in computer graphics/physics
- A velocity or force
- Rotation

Example 2.3.1 (C++ Representation)**.** *Here is an example script using the 'vector' class of the C++ standard library.*

upquote

```
#include <iostream>
#include <vector>

int main() {
    std::vector<int> v1 = {3, 4};
    std::vector<int> v2 = {-2, 1};

    std::vector<int> sum(2);
    for (size_t i = 0; i < 2; ++i) {
        sum[i] = v1[i] + v2[i];
    }

    std::cout << "v1 + v2 = [";
    for (size_t i = 0; i < 2; ++i) {
        std::cout << sum[i];
        f (i < 1) std::cout << ", ";
    }
    std::cout << "]" << std::endl;
    return 0;
}

```

2.4 Vector Operations

2.4.1 Magnitude

The magnitude of a vector $\vec{v}$, written as $\|\vec{v}\|$, is simply the square root of the squares of each of its elements.

$$||\vec{v}|| = \sqrt{v_1^2 + v_2^2 \cdots + v_n^2} \tag{2.11}$$

Example:

$$|| \begin{bmatrix} 3 \\ 4 \end{bmatrix} || = \sqrt{3^2 + 4^2} = 5 \tag{2.12}$$

Note: The letter n is commonly used to denote the length of a vector, which is the amount of elements it contains.
Also worth noting: A unit vector is a vector with a magnitude of 1.

2.4.2 Vector Addition

If both vectors are the same length, you can combine 2 vectors by adding their *components*.

$$a \vec{+} b = \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ . \\ a_n \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ . \\ . \\ . \\ b_n \end{bmatrix} = \begin{bmatrix} a_1 + b_2 \\ a_2 + b_2 \\ . \\ . \\ . \\ a_n + a_b \end{bmatrix} \tag{2.13}$$

Example:

$$a \vec{+} b = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} + \begin{bmatrix} 5 \\ 6 \\ 7 \\ 8 \end{bmatrix} = \begin{bmatrix} 6 \\ 8 \\ 10 \\ 12 \end{bmatrix} \tag{2.14}$$

2.4.3 Vector Subtraction

If both vectors are the same length, you can subtract 2 vectors by subtracting their *components*.

$$a \vec{-} b = \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ . \\ a_n \end{bmatrix} - \begin{bmatrix} b_1 \\ b_2 \\ . \\ . \\ . \\ b_n \end{bmatrix} = \begin{bmatrix} a_1 - b_2 \\ a_2 - b_2 \\ . \\ . \\ . \\ a_n - a_b \end{bmatrix} \tag{2.15}$$

Example:

$$a \vec{-} b = \begin{bmatrix} 5 \\ 6 \\ 7 \\ 8 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 4 \\ 4 \end{bmatrix} \tag{2.16}$$

2.4.4 Scalar Multiplication

Scalar multiplication is the act of multiplying a vector by a scalar. You multiply the scalar to all *components* inside the vector.

$$c \cdot \vec{a} = c \cdot \begin{bmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ a_n \end{bmatrix} = \begin{bmatrix} c \cdot a_1 \\ c \cdot a_2 \\ \cdot \\ \cdot \\ \cdot \\ c \cdot a_n \end{bmatrix} \text{ where } c \in \mathbb{R} \tag{2.17}$$

Example:

$$5 \cdot \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 5 \\ 10 \\ 15 \\ 20 \end{bmatrix} \tag{2.18}$$

2.4.5 Dot Product

The dot product (also known as the *inner product* is the sum of all components between two vectors.

Algebraically it can be used to measure the similarity between 2 vectors. If you have 2 vectors of the same size:

$$\vec{u} = \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ \cdot \\ u_n \end{bmatrix}, \vec{v} = \begin{bmatrix} v_1 \\ v_2 \\ \cdot \\ \cdot \\ \cdot \\ v_n \end{bmatrix} \tag{2.19}$$

Then the **dot product** would be:

$$\vec{u} \cdot \vec{v} = u_1 v_1 + u_2 v_2 + \cdots + u_n v_n \tag{2.20}$$

Example:

$$\vec{u} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \vec{v} = \begin{bmatrix} 4 \\ -1 \end{bmatrix} \tag{2.21}$$

$$\vec{u} \cdot \vec{v} = (2 \cdot 4) + (4 \cdot -1) = 8 - 3 = 5 \tag{2.22}$$

The dot product can also relate to angles between vectors:

$$\vec{u} \cdot \vec{v} = \|\vec{u}\| \|\vec{v}\| cos\theta \tag{2.23}$$

- $\|\vec{u}\|$ is the magnitude (length) of $\vec{u}$
- θ is the angle between $\|\vec{u}\|$ and $\|\vec{v}\|$

Useful notes:

- If $\vec{u} \cdot \vec{v} > 0$: Vectors are pointing in a similar direction.
- If $\vec{u} \cdot \vec{v} < 0$: vectors are pointing in opposite directions.
- If $\vec{u} \cdot \vec{v} = 0$: vectors are perpendicular

2.4.6 Cross Product

The cross product of two vectors is another vector that is perpendicular to both factors.

$$\vec{u} \times \vec{v} = \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix} \tag{2.24}$$

2.4.7 Normalization

Normalization converts a vector to a *unit vector*.

$$\hat{a} = \frac{\vec{a}}{||\vec{a}||} \tag{2.25}$$

When a vector is *hatted* (e.g. $\hat{a}, \hat{b}$), it is a unit vector.

Chapter 3

Matrices

3.1 What is a Matrix?

Matrices are also one of the biggest building blocks of linear algebra. A matrix is an array of values with m rows and n columns ($m \times n$). Matrices are similar to vectors; a vector could be considered a single-column matrix (in fact, when programming with them, that is often the case!). For example, consider the matrix:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \tag{3.1}$$

This matrix has 3 rows and 2 columns (3×2).

Matrix values are often represented using a subscript, with the first value being m and the 2nd being n. So the value of the 5th row, 2nd column could be written as $\mathbf{a_{5,2}}$. See here:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} \tag{3.2}$$

i and j are index values. i = row numbers, j = column numbers.

3.2 Matrix Operations

3.2.1 Addition

If both matrices have the same dimensions, you can combine them by adding their *components*.

$$A + B = \begin{bmatrix} 2 & 4 & 1 \\ 0 & -3 & 5 \end{bmatrix} + \begin{bmatrix} 1 & 0 & -1 \\ 7 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 2+1 & 4+0 & 1+(-1) \\ 0+7 & -3+2 & 5+3 \end{bmatrix} = \begin{bmatrix} 3 & 4 & 0 \\ 7 & -1 & 8 \end{bmatrix} \tag{3.3}$$

3.2.2 Subtraction

If both matrices have the same dimensions, you can subtract them by adding their *components*.

$$A - B = \begin{bmatrix} 2 & 4 & 1 \\ 0 & -3 & 5 \end{bmatrix} + \begin{bmatrix} 1 & 0 & -1 \\ 7 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 2-1 & 4-0 & 1-(-1) \\ 0-7 & -3-2 & 5-3 \end{bmatrix} = \begin{bmatrix} 1 & 4 & 2 \\ -7 & -5 & 2 \end{bmatrix} \tag{3.4}$$

Scalar Multiplication

To multiply a matrix by a scalar, simply multiply each component by that scalar.

$$c \cdot A = c \cdot \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ . & \\ . & \\ . & \\ a_{m1} & a_{m2} \end{bmatrix} = \begin{bmatrix} c \cdot a_{11} & c \cdot a_{21} \\ c \cdot a_{21} & c \cdot a_{22} \\ . & \\ . & \\ . & \\ c \cdot a_{m1} & c \cdot a_{m2} \end{bmatrix} \text{ where } c \in \mathbb{R} \tag{3.5}$$

Transpose

A function that "flips" a matrix. The transpose of a matrix with m rows and n columns has n rows and m columns.

$$A^T = \begin{bmatrix} a_{ji} \end{bmatrix} \tag{3.6}$$

Example:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \tag{3.7}$$

$$A^T = \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \\ a_{13} & a_{23} \end{bmatrix} \tag{3.8}$$

Here is an example that uses actual numbers.

$$A = \begin{bmatrix} 2 & 1 & -3 \\ 5 & 9 & 4 \end{bmatrix} \tag{3.9}$$

$$A^T = \begin{bmatrix} 2 & 5 \\ 1 & 9 \\ -3 & 4 \end{bmatrix} \tag{3.10}$$

3.3 Matrix Types

3.3.1 Identity Matrix

An identity matrix is a square matrix with 1s on the diagonal and 0s everywhere else. Here's what a 5×5 identity matrix would look like:

$$I_5 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.11}$$

The identity matrix is important because it is the multiplicative identity: any matrix times the appropriately sized identity matrix is that matrix!

3.3.2 Zero Matrix

A zero matrix is a matrix with all zeros. It's pretty simple lol:

$$0_{5,5} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.12}$$

Similarly to how any number plus zero is that number, any matrix plus the appropriately sized zero matrix is that matrix.