DATA SCIENCE

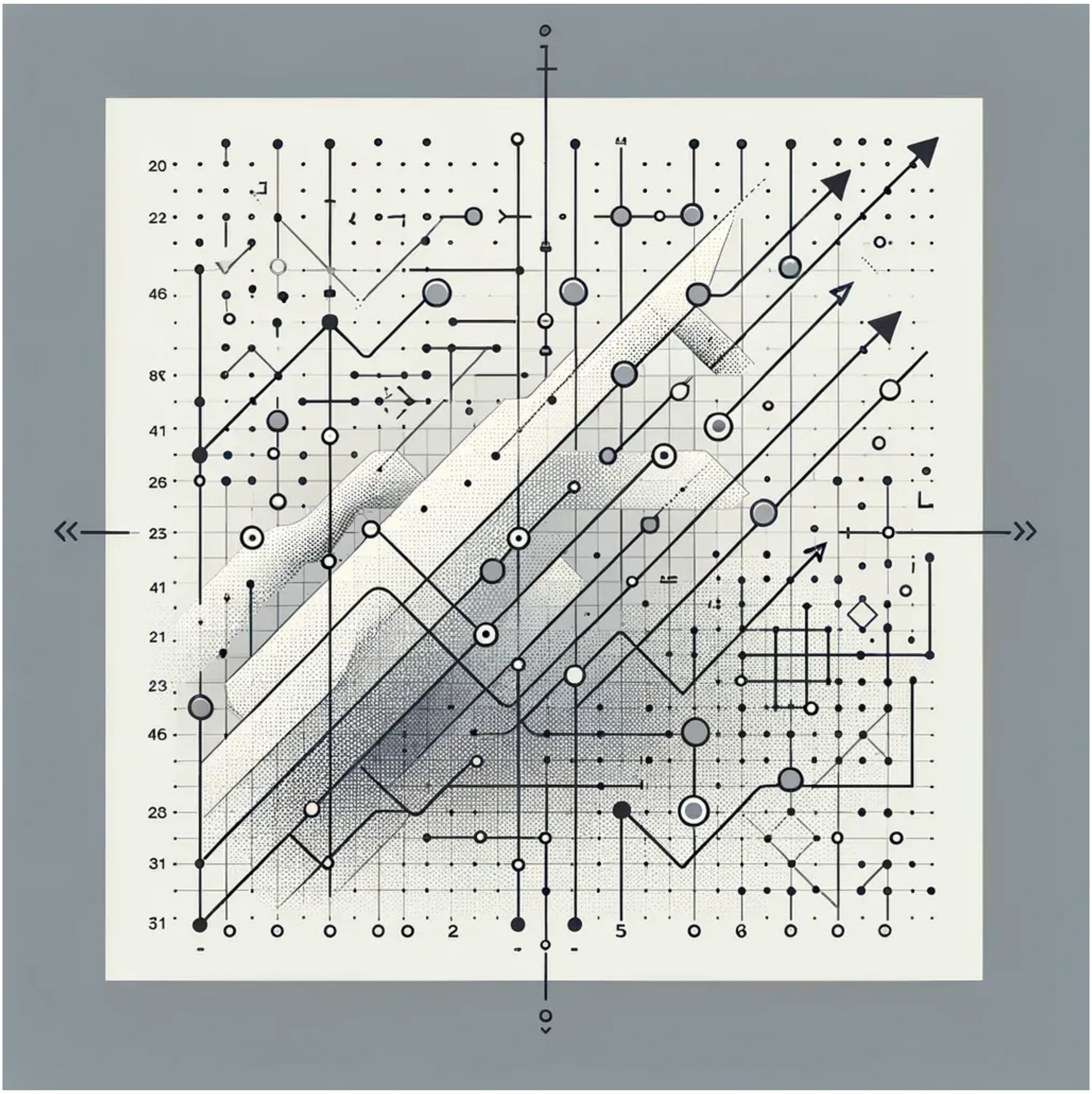
Linear Algebra Concepts Every Data Scientist Should Know

Do you know Linear Algebra well enough?

Benedict Neo · Follow

Published in bitgrit Data Science Publication

10 min read · Jun 18, 2024



Linear algebra is a bedrock for all data science and machine learning tasks.

It is the language that transforms theoretical models into practical solutions.

It embodies principles that allow algorithms to learn from data.

xkcd

They're used for:

1. **Representation of data**: a structured way to organize and manipulate data, allowing complex datasets to be represented as matrices
2. **dimensionality reduction**: techniques like PCA rely on linear algebra to reduce the number of variables to enhance model efficiency without losing important information
3. **optimization**: gradient descent, the core engine for ML, uses linear algebra to find the minimum of a function.
4. **Feature engineering**: linear transformation and matrix operations create new features from existing data
5. **similarity measures**: embeddings are stored as vectors and are used in recommendation systems and AI chatbots today.
6. and many more!

Table of contents

This article will look at some linear algebra concepts, visual explanations, and code examples.

Let's dive right in!

Code → **Deepnote Notebook**

Vector

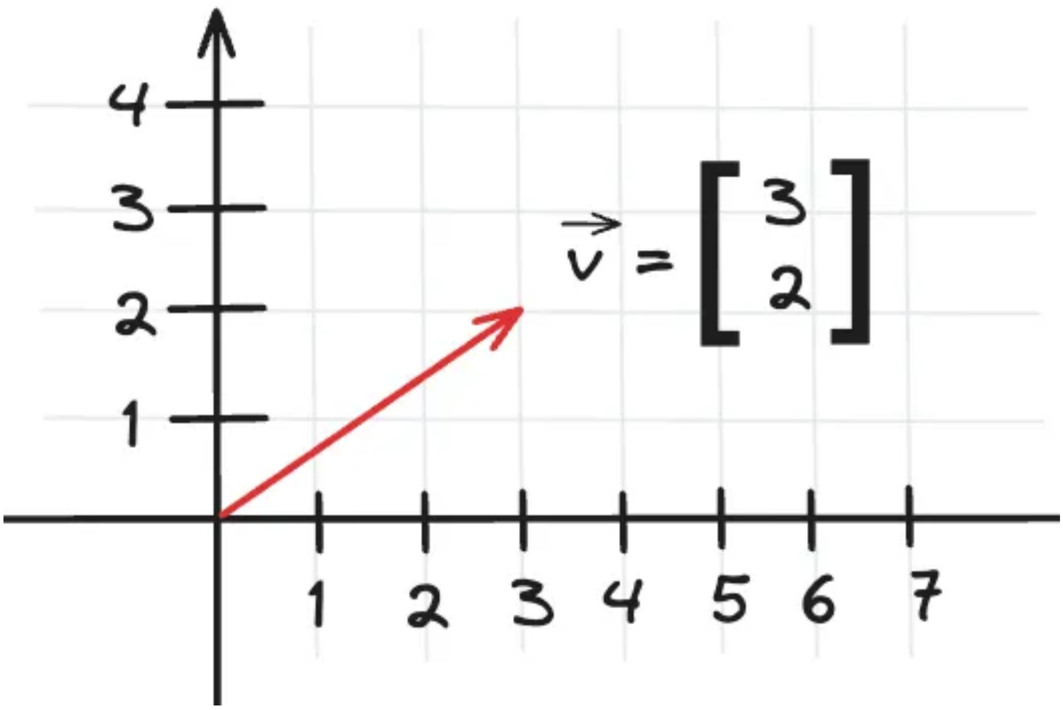


image by author

This is the fundamental building block of linear algebra.

There are 3 ways to think of a vector.

The first is the **physics perspective:** vectors are arrows pointing in space, defined by length and direction. Vectors on a flat plane are 2-dimensional, and those in the space we live in are 3-dimensional.

The second is the **computer science perspective:** vectors are ordered lists of numbers. The length of this list determines the dimension.

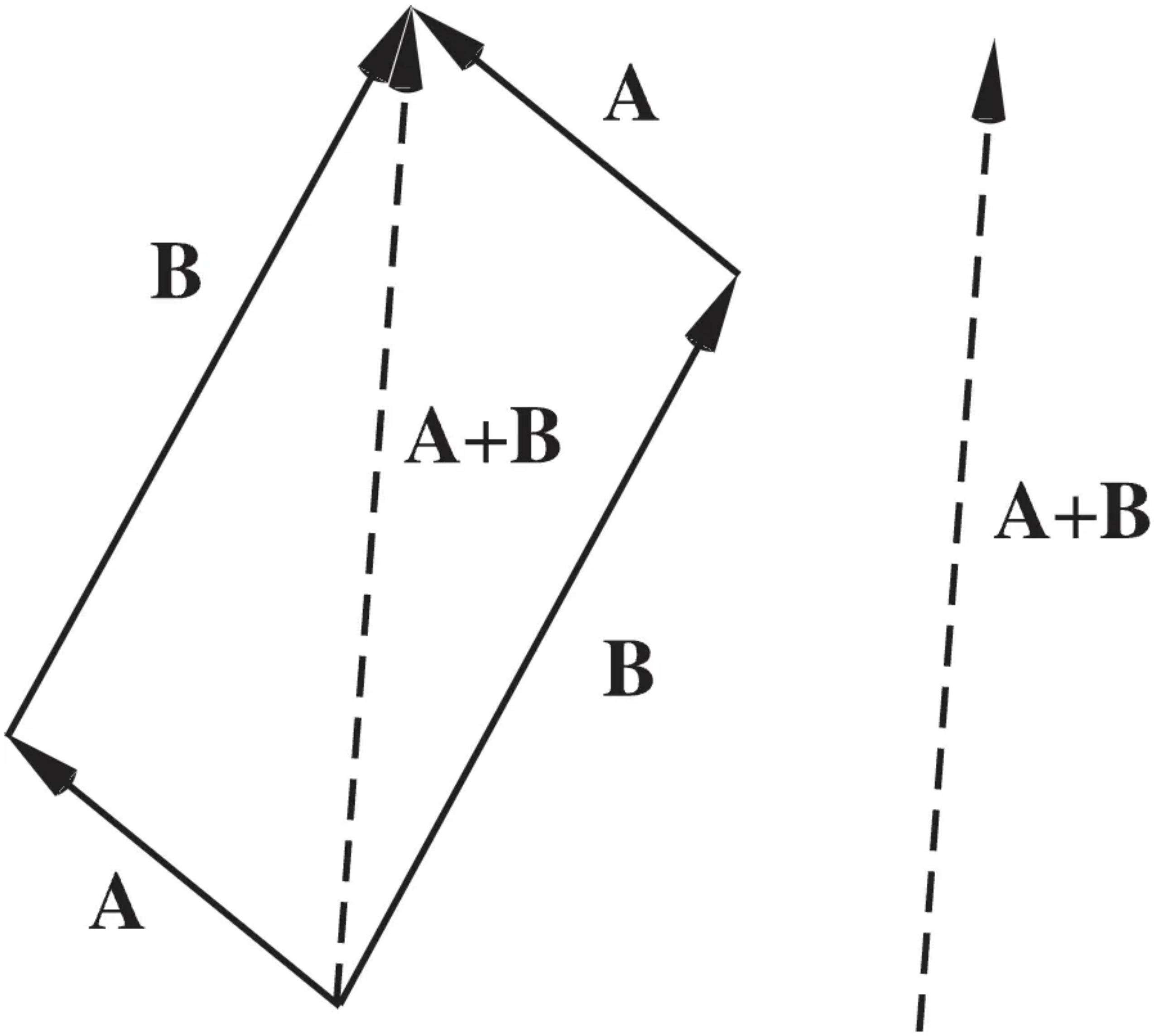
The third is the **mathematician's perspective**: vectors can be anything where two vectors are added and multiplied by a number.

Unit vector

A unit vector is a vector with a magnitude of 1. It is often used to represent the direction of a vector without regard to its magnitude.

Vector operations

Vector addition



source

The addition of two vectors to form a new vector, component-wise.

$$\vec{a} + \vec{b} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} + \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = \begin{bmatrix} a_x + b_x \\ a_y + b_y \\ a_z + b_z \end{bmatrix}$$

Scalar multiplication

Scalar multiplication is the multiplication of a vector by a scalar (a number) that results in the vector with the same direction (or opposite if the scalar is negative) as the original vector but with a magnitude that is scaled by the absolute value of the scalar.

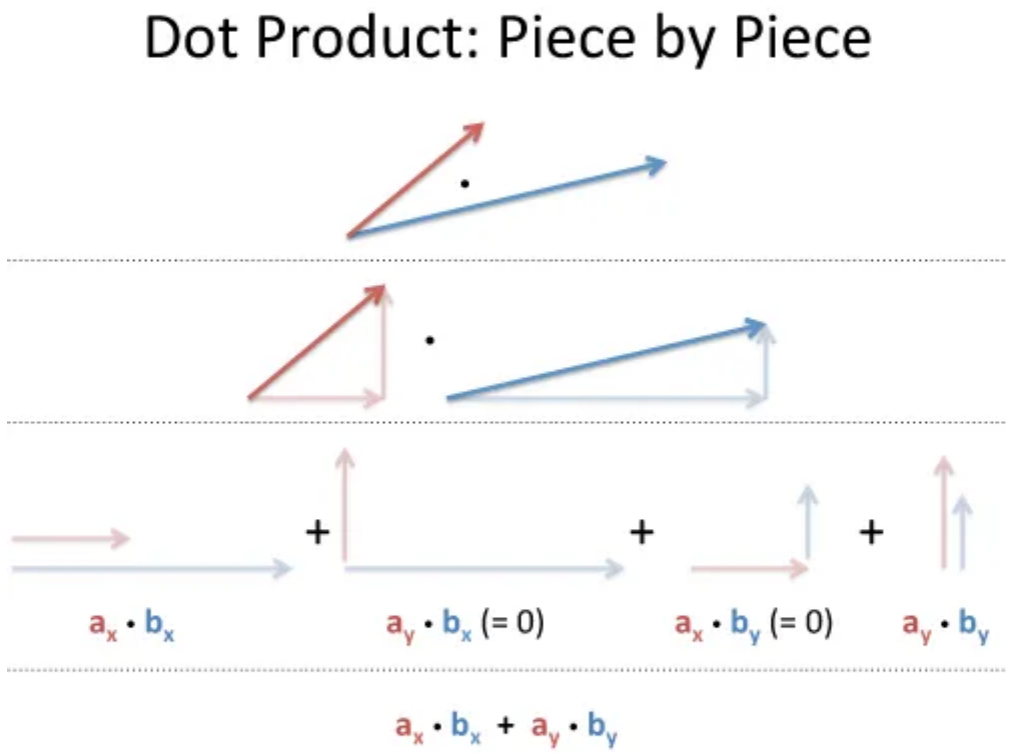
Dot product

Formally, it is the product of the Euclidian magnitudes of two vectors and the cosine of the angle between them, reflecting both the length of the vectors and their directional relationship.

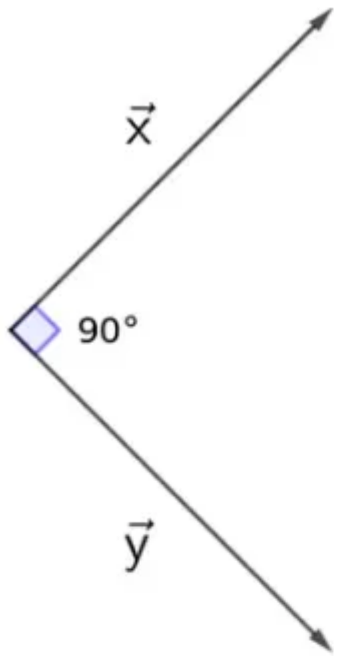
$$\vec{a} \cdot \vec{b} = a_x \cdot b_x + a_y \cdot b_y = |\vec{a}||\vec{b}| \cos(\theta)$$

dot product formula

Intuitively, think of it as applying the directional growth of one vector to another or “How much push/energy is one vector giving to the other?”. The result is how much stronger we’ve made the original vector (positive, negative, or zero)



If the dot product is 0, it tells us that the vectors are orthogonal.



$$\vec{x} \cdot \vec{y} = |\vec{x}||\vec{y}|\cos(90^\circ) = 0$$

source

A fun analogy by betterexplained

Imagine the red vector is your speed, and the blue vector is the orientation of the boost pad. Larger numbers = more power. The dot product is how much boost you will get.

Using the equation, |a| is your incoming speed, |b| is the max boost, the percentage of boost you get is cos(**θ**), for an overall boost of |a| |b| cos(**θ**)

betterexplained

Vector space

A vector (or linear) space is any collection of vectors that can be added together and multiplied ("scaled") by numbers, called scalars in this context.

A list of axioms must be satisfied for V to be called a vector space.

Definition of Vector Space

Let V be a set on which two operations (**vector addition** and **scalar multiplication**) are defined. If the listed axioms are satisfied for every $\mathbf{u}$, $\mathbf{v}$, and $\mathbf{w}$ in V and every scalar (real number) c and d, then V is called a **vector space.**

Addition:

1. $\mathbf{u} + \mathbf{v}$ is in V.	Closure under addition
2. $\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$	Commutative property
3. $\mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$	Associative property
4. V has a **zero vector** $\mathbf{0}$ such that for every $\mathbf{u}$ in V, $\mathbf{u} + \mathbf{0} = \mathbf{u}$.	Additive identity
5. For every $\mathbf{u}$ in V, there is a vector in V denoted by $-\mathbf{u}$ such that $\mathbf{u} + (-\mathbf{u}) = \mathbf{0}$.	Additive inverse

Scalar Multiplication:

6. $c\mathbf{u}$ is in V.	Closure under scalar multiplication
7. $c(\mathbf{u} + \mathbf{v}) = c\mathbf{u} + c\mathbf{v}$	Distributive property
8. $(c + d)\mathbf{u} = c\mathbf{u} + d\mathbf{u}$	Distributive property
9. $c(d\mathbf{u}) = (cd)\mathbf{u}$	Associative property
10. $1(\mathbf{u}) = \mathbf{u}$	Scalar identity

source

Null space (kernel)

The null space is a set of vectors that, when multiplied by the matrix, results in the zero vector.

It represents the solution to the equation Ax = 0, where A is the given matrix.

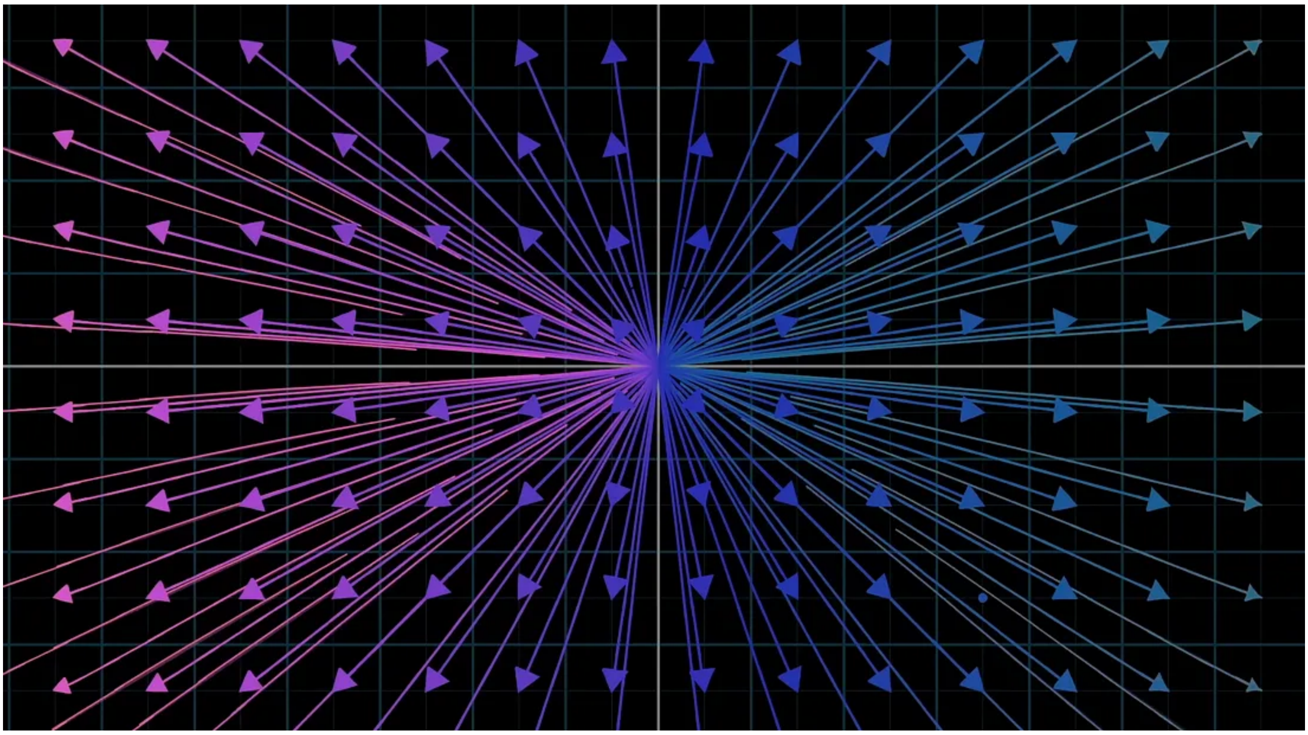
Imagine a 2d space with two vectors; the null space of a matrix can be visualized as a subspace that collapses these vectors to the origin (zero vector) when multiplied by the matrix.

Span

The set of all possible vectors you can reach given a linear combination of a given pair of vectors, *v* and *w*,

$av + bw$, and let a and b be all real numbers.

For most pairs of vectors, it can reach every point in the 2d vector plane



3blue1brown video on span

When the two vectors happen to line up, it is limited to the single line that passes through the origin.

Matrix

Matrices are a way to organize inputs and operations in rows and columns.

The idea of span underlies the idea of basis.

Basis

The basis is a set of **linearly independent** vectors that span the entire vector space. This means every vector in the vector space can be expressed as a **linear**

combination of the basis vector.

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

image by author

Here's a matrix with 2 rows and 2 columns.

They're a mathematical tool that can solve problems in a structured manner.

Matrices as functions

You can think of matrices as functions. Just as a Python function takes input parameters, processes them, and returns output, a matrix transformation transforms input vectors into output vectors through linear transformation.

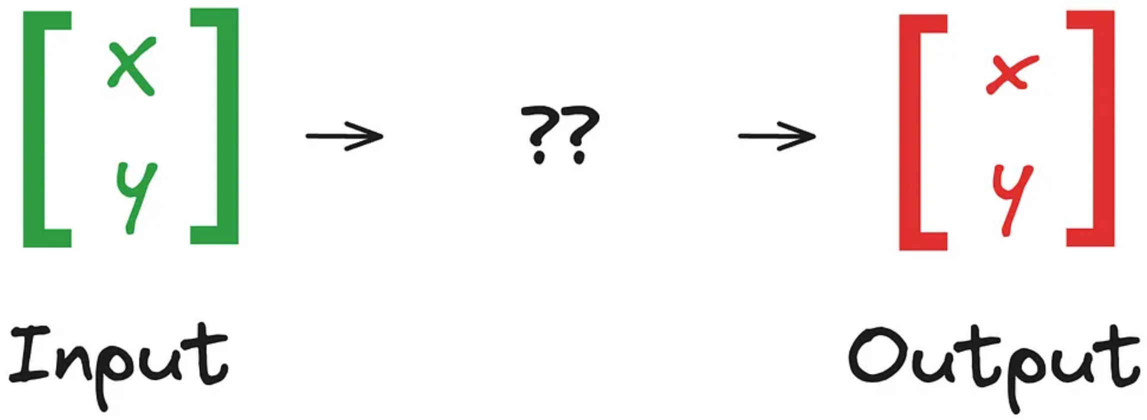
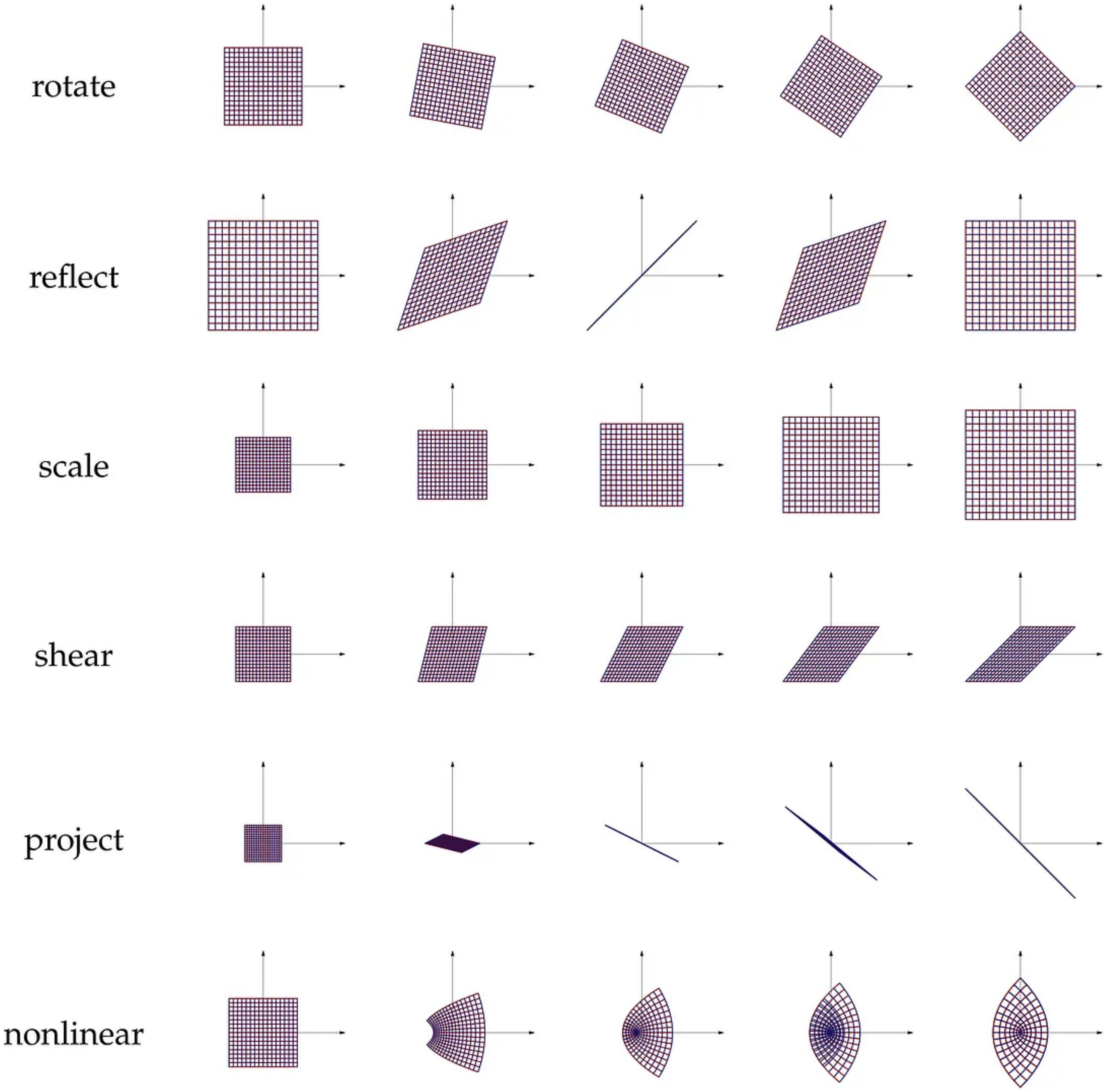


image by author

Linear Transformation



source

A linear transformation is a mapping $V \to W$ between two vector spaces that preserves the operations of vector addition and scalar multiplication.

In practical terms, applying a matrix A to a vector x to get another vector y (via the operation $Ax = y$) is a linear transformation.

This is used heavily in data science:

- **dimensionality reduction:** PCA uses linear transformation to map high-dimensional data into lower-dimensional space

- **data transformation:** normalizing or standardizing a dataset is a linear transformation
- **feature engineering:** creating new features through combinations of existing ones.

Below are a few forms of matrices

Inverse Matrix

A matrix, when multiplied by its inverse, results in the identity matrix.

Singular Matrix

A singular matrix is a square matrix that does not have an inverse. This is equivalent to saying the matrix's determinant is zero or its rank is less than its size.

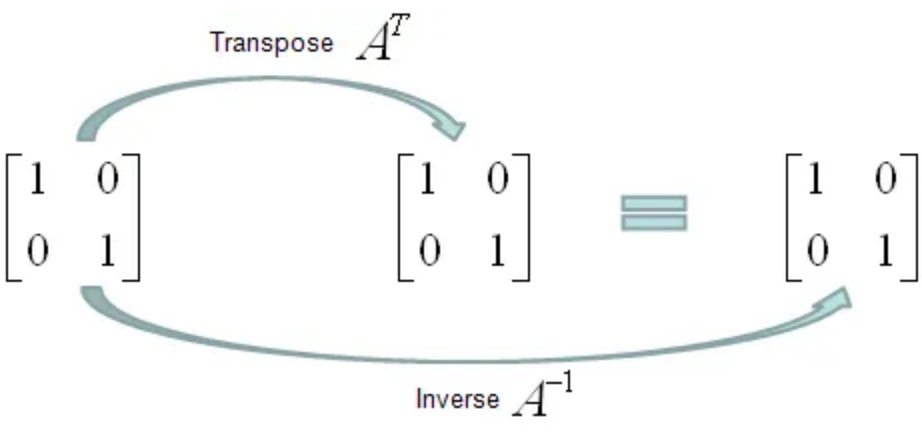
Identity matrix.

The identity matrix is a square matrix with values of one on the diagonals and zero everywhere else. It acts as a multiplicative identity in matrix multiplication, leaving any matrix unchanged by it, just like the number 1.

Diagonal Matrix

A diagonal matrix is a square matrix where all entries outside the main diagonal are zero. It is used in finding eigenvalues, and for calculating the determinant.

Orthogonal matrix



A square matrix with real elements is considered orthogonal if its **transpose** equals its **inverse**.

Formally, a matrix A is orthogonal if $A^T A = AA^T = I$, where I is the identity matrix.

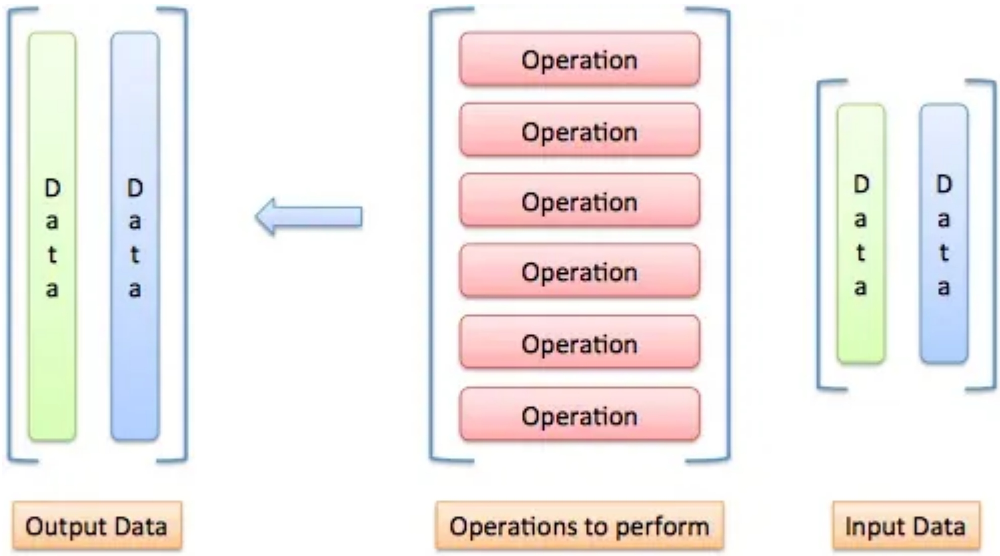
Geometrically, a matrix is orthogonal if its columns and rows are orthogonal unit vectors, a.k.a. they are mutually perpendicular and have a magnitude of 1.

Recall that two vectors are orthogonal if they are perpendicular to each other (90 degrees) and the dot product between them is 0.

Matrix multiplication

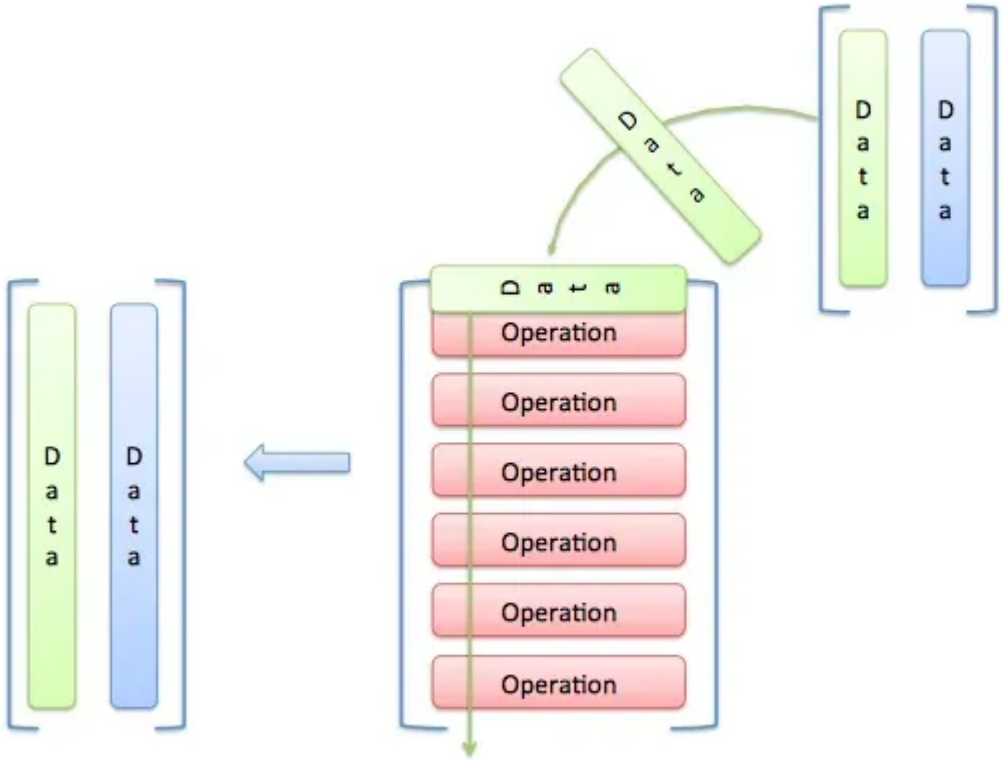
You use matrices to perform **matrix multiplication**.

Here's a nice visualization from An Intuitive Guide to Linear Algebra

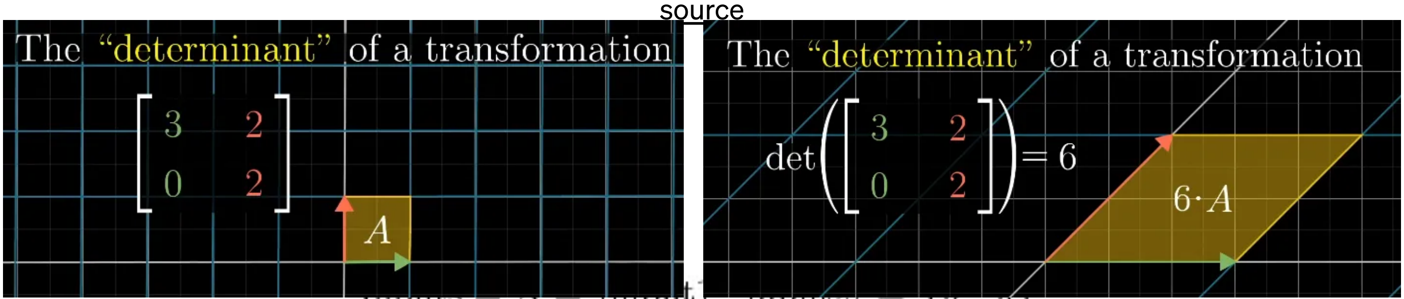


source

Imagine you're pouring each input data through each operation.



source

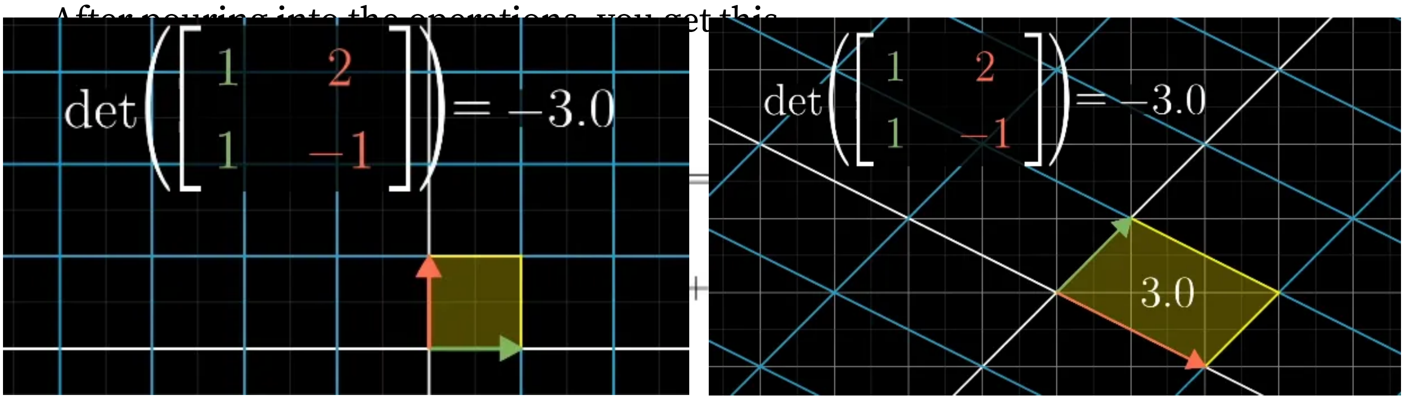


source

A negative determinant tells us that the entire space was flipped. A transformation of this is like turning a set of paper onto the other side.

$$\text{Operations} = M = \begin{bmatrix}\text{operation1}\\\text{operation2}\end{bmatrix} = \begin{bmatrix}3 & 4 & 5\\3 & 0 & 0\end{bmatrix}$$

After pouring into the operations, you get this



source

The input was a [3 x **2]** matrix, and our operation matrix is **[2** x 3]; the result is [2 x

3] [3 x 2] = [2 x 2].

Notice how the orientation of the red and green axes was reversed.
The size of the input has to match the size of the operation.

A determinant of 0 means the matrix is "destructive" and cannot be reversed. Similar to multiplying by zero, information is lost.

Determinants can tell us whether a matrix is invertible if `det(A)` is 0, the inverse does not exist; the matrix is singular.

Rank

The maximum number of linearly independent column/row vectors in a matrix. It represents the dimension of the vector space spanned by its rows or columns.

It also tells us the number of output dimensions after a linear transformation.

When the output of a transformation is a single line (it is one-dimensional), we say the transformation has a rank of 1.

If all vectors land on some two-dimensional plane, we say the transformation has a rank of 2.

For a 2x2 matrix, a rank of 2 is the best that it can be. This is known as a full rank. It means the basis vectors can span the entire 2d space and the non-zero determinant.

But for 3x3 matrices, a rank of 2 means it collapsed, but not as much as a rank of 1.

Eigenvectors and Eigenvalues

Eigenvectors and eigenvalues represent the “axes” of transformation.

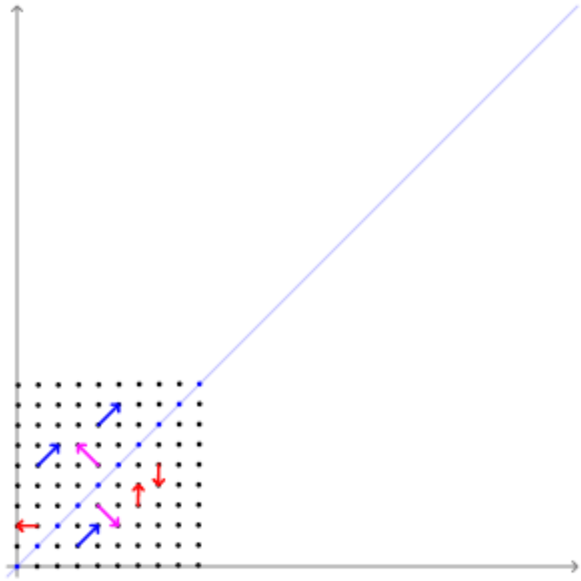
Eigenvectors are inputs that don’t change direction after a linear transformation. Even though the direction doesn’t change, the size might. This size, the amount that the eigenvector is scaled up or down, is the eigenvalue.

Think about when you spin a globe; every location faces a new direction except the poles. Their direction doesn’t change.

Here’s a visual example of eigenvectors.

Trace

The trace of a matrix i It is invariant under the change of basis an e matrix, i.e., the trace is the sum of the



Eigenvectors

Formally, for a matrix A and a vector v, if $Av = \lambda v$, then λ is an eigenvalue, and v is an eigenvector of A.

Anther way of saying this is the eigenvectors of a square matrix A are vectors for which **matrix multiplication = scalar multiplication**.

Determinant

Determinant is the size of the output transformation.

If the input was the unit vector (area or volume of 1), the determinant is the size of the transformed area or volume.

Take this matrix, for example. If the area of A was scaled by 6, the determinant of the transformation is 6.

Thanks for reading!

Resources

Hackers approach

- Computational Linear Algebra for Coders
- Introduction to Linear Algebra for Applied Machine Learning with Python

Visualize

- Graphical Linear Algebra — a new way of doing LA
- Essence of linear algebra by 3Blue1Brown — amazing animations, visualize concepts
- inVectorize
- Intuitive Math

Papers/Courses/Textbooks

- The Matrix Calculus You Need For Deep Learning

- Matrix Methods in Data Analysis, Signal Processing, and Machine Learning | Mathematics | MIT OpenCourseWare
- Linear Algebra Done Right
- linear_algebra_in_4_pages.pdf

Stay connected!

Be sure to follow the **bitgrit Data Science Publication** to keep updated!

Want to discuss the latest developments in Data Science and AI with other data scientists? **Join our discord server**!

Follow Bitgrit below to stay updated on workshops and upcoming competitions!

Discord | Website | Twitter | LinkedIn | Instagram | Facebook | YouTube

Data Science Mathematics Machine Learning Math Python

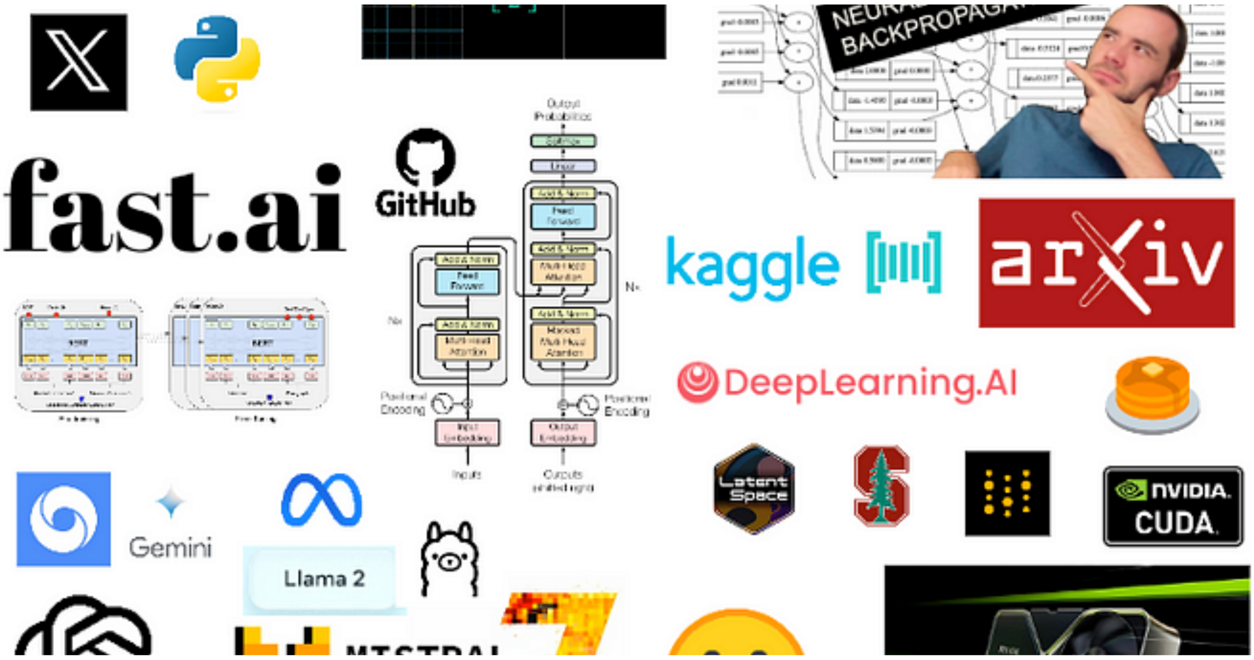
Follow

Written by Benedict Neo

42K Followers · Editor for bitgrit Data Science Publication

Python ∩ Data Science ∩ AI https://www.linkedin.com/in/benedictneo/

More from Benedict Neo and bitgrit Data Science Publication

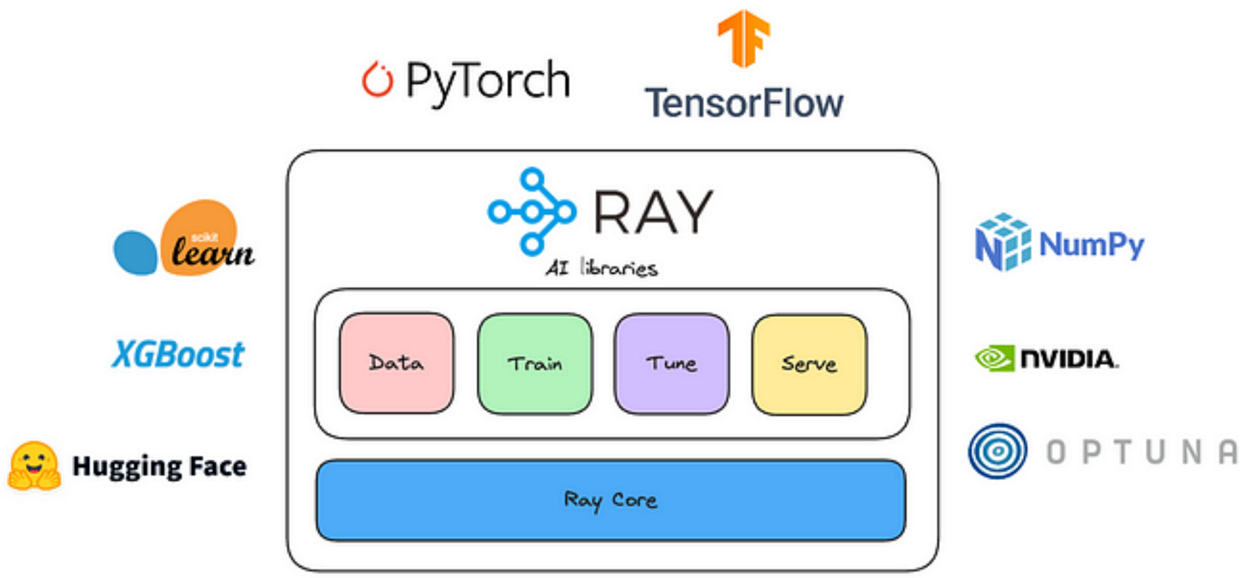


Benedict Neo in bitgrit Data Science Publication

Roadmap to Learn AI in 2024

A free curriculum for hackers and programmers to learn AI

Mar 11 · 13.8K · 180



Benedict Neo in bitgrit Data Science Publication

The Modern AI Stack: Ray

283 2

Benedict Neo in bitgrit Data Science Publication

10 Best Practices for Data Science

Lessons from 100+ data science projects with new startups to Fortune 50 companies

Jun 14 902 10

Benedict Neo in bitgrit Data Science Publication

Forget `pip install`, Use This Instead

Install Python packages up to 100x ⚡ faster than before.

Mar 27 2.5K 26

See all from Benedict Neo

See all from bitgrit Data Science Publication

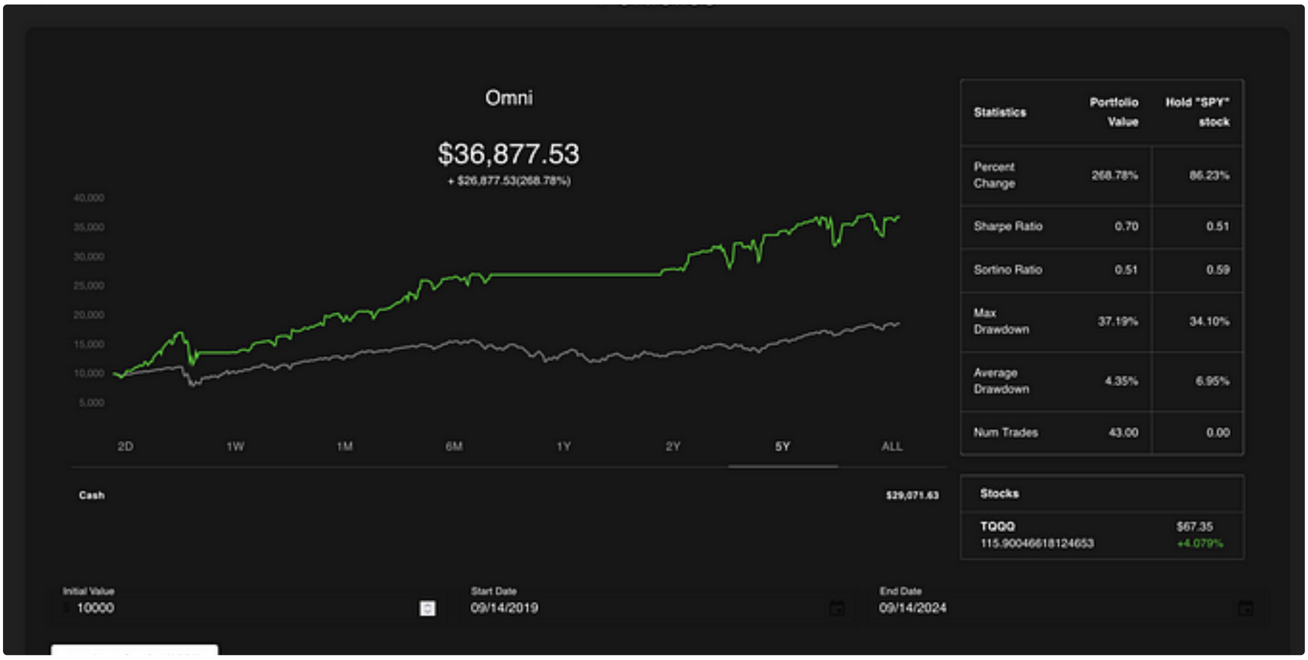
Recommended from Medium

Mahnoor Fatima

The Perfect Data Science Resume in 2023 (an 8-Step Guide)

If you want a job in data science, you'll have to get past the "Gatekeeper," better known as the recruiter or hiring manager. And that...

Jun 29 83 4



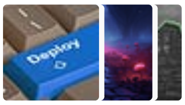
Austin Starks in DataDrivenInvestor

I used OpenAI's o1 model to develop a trading strategy. It is DESTROYING the market

It literally took one try. I was shocked.

Sep 15 · 3.7K · 103

Lists



Predictive Modeling w/ Python

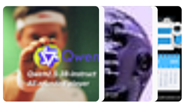
20 stories · 1594 saves

Practical Guides to Machine Learning

10 stories · 1938 saves

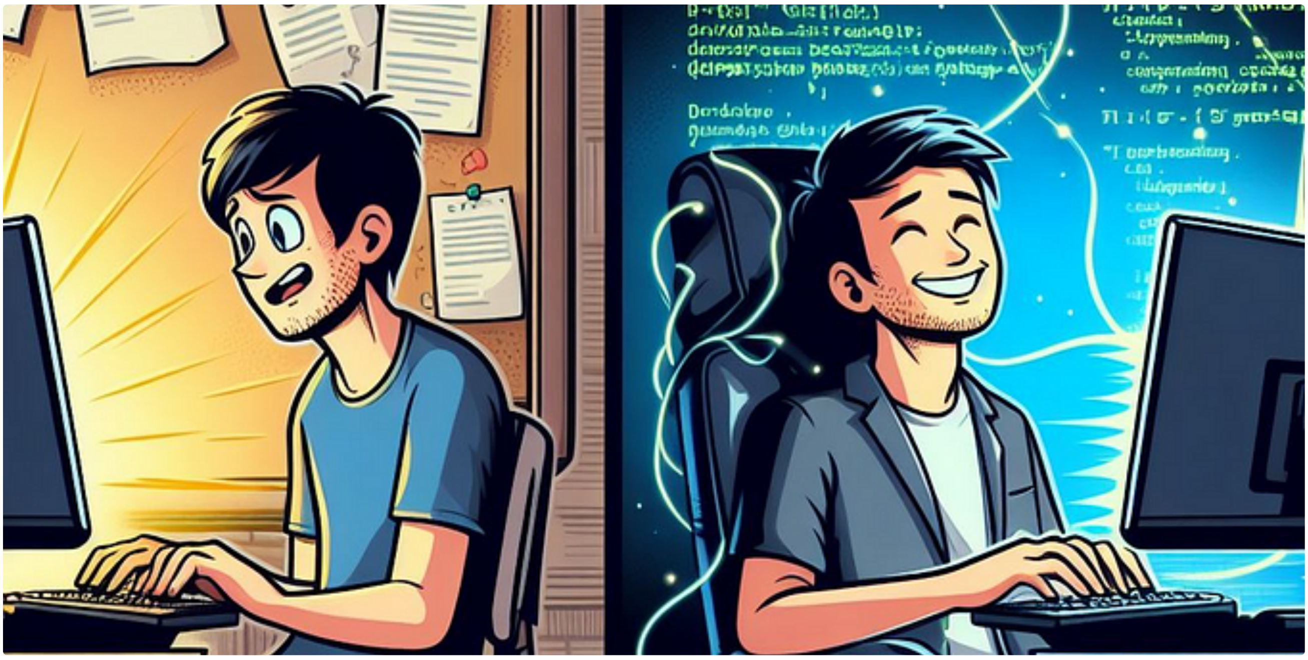
Coding & Development

11 stories · 848 saves



Natural Language Processing

1752 stories · 1352 saves



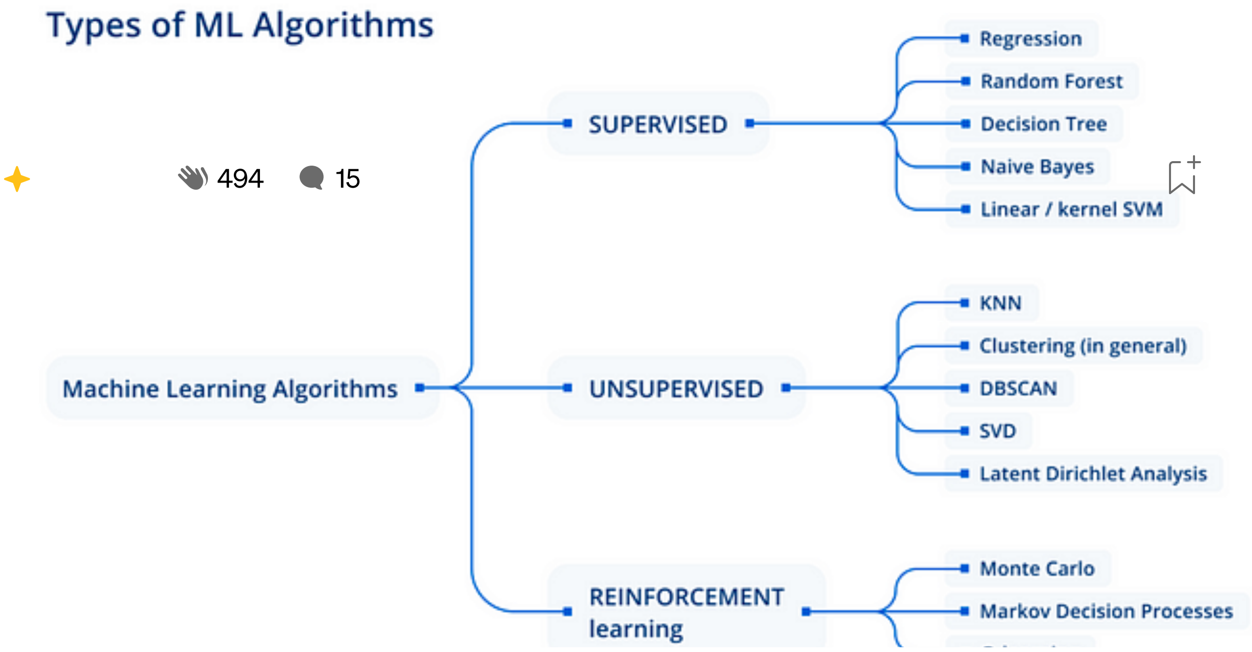
Abhay Parashar in The Pythoneers

17 Mindblowing Python Automation Scripts I Use Everyday

Scripts That Increased My Productivity and Performance

SHIVAM JINDAL in Stackademic

Why Meta is So Focused on Probability Questions in Interviews



494 15

John Vastola

10 Must-Know Machine Learning Algorithms for Data Scientists

Machine learning is the science of getting computers to act without being explicitly programmed." — Andrew Ng

Dec 6, 2022 473 7

Nathan Rosidi

Top 10 Machine Learning Algorithms for Beginner Data Scientists

Let's explore the machine learning algorithms perfect for beginners in data science. We'll explain each one and show you how to use them...

Apr 25 1.7K 34

See more recommendations