Intro to Vectors and Matrix Operations

QL1.1

Learning Outcomes

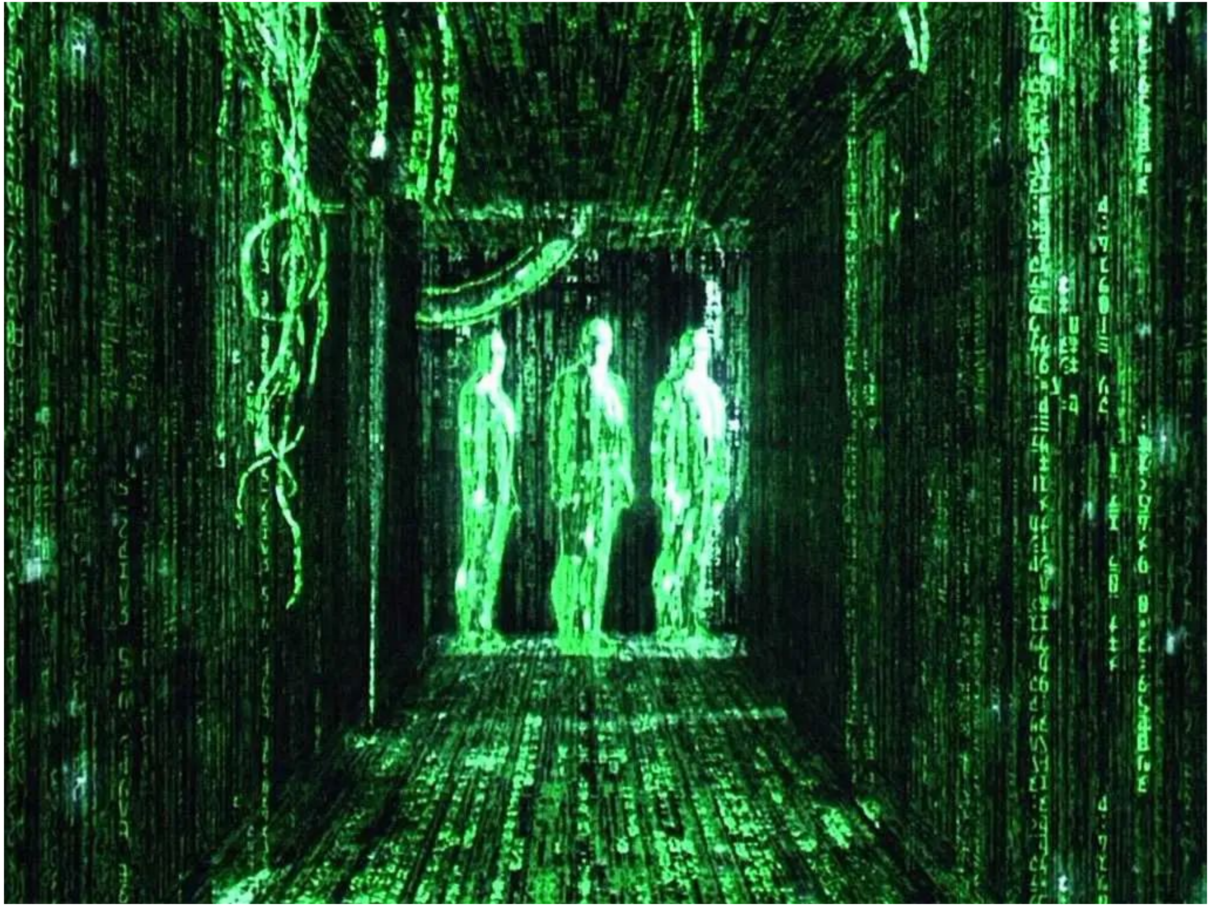
By the end of today, you should be able to...

1. Describe vectors and Matrices
2. Compare and contrast different types of matrix multiplication
3. Implement matrix multiplication using numpy
4. Apply matrix operation to solve various problems

Why we need Linear Algebra

- Whatever you see in your display is achieved by linear algebra
- Airplanes, trains, satelites, GPS, Video Games, etc all use Matrix operations.
- Most of machine learning models and all most all deep learning models need it

Everything is in Matrix!



Matrix Definition

- Matrix: a rectangular array or table of numbers, symbols, or expressions, arranged in rows and columns.

$$
\begin{array}{c c}
 & \begin{array}{cccc} 1 & 2 & \cdots & n \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ \vdots \\ m \end{array} &
\begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
a_{31} & a_{32} & \cdots & a_{3n} \\
\vdots & \vdots & \vdots & \vdots \\
a_{m1} & a_{m2} & \cdots & a_{mn}
\end{bmatrix}
\end{array}
$$

Vector

Vector: Vector is a matrix with 1 dimension.

Name	Size	Example	
Row vector	$1 \times n$	$\begin{bmatrix} 3 & 7 & 2 \end{bmatrix}$	A matrix with one row, sometimes used to represent a vector
Column vector	$n \times 1$	$\begin{bmatrix} 4 \\ 1 \\ 8 \end{bmatrix}$	A matrix with one column, sometimes used to represent a vector

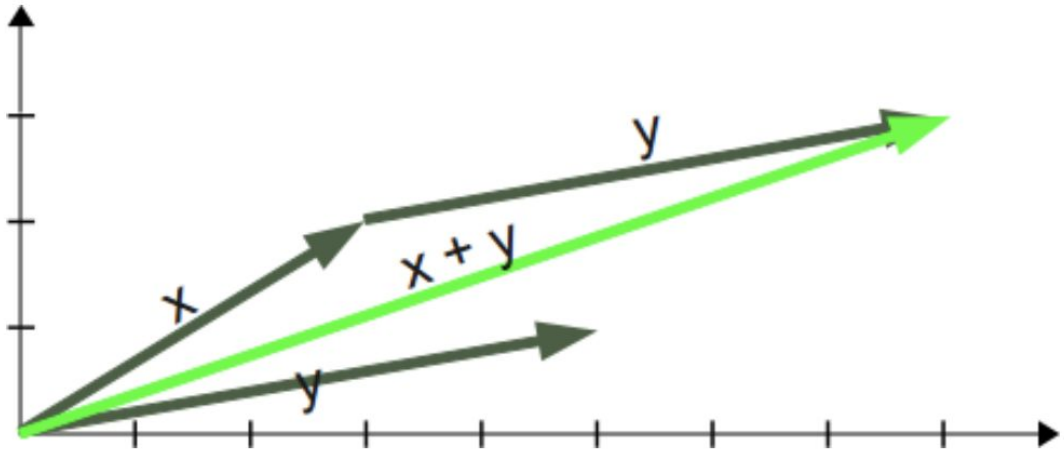
Addition, Scalar multiplication and Transposition

Operation	Definition	Example
Addition	The *sum* **A**+**B** of two *m*-by-*n* matrices **A** and **B** is calculated entrywise: $(\mathbf{A}+\mathbf{B})_{i,j} = \mathbf{A}_{i,j} + \mathbf{B}_{i,j}$, where $1 \le i \le m$ and $1 \le j \le n$.	$\begin{bmatrix}1 & 3 & 1\\1 & 0 & 0\end{bmatrix} + \begin{bmatrix}0 & 0 & 5\\7 & 5 & 0\end{bmatrix} = \begin{bmatrix}1+0 & 3+0 & 1+5\\1+7 & 0+5 & 0+0\end{bmatrix} = \begin{bmatrix}1 & 3 & 6\\8 & 5 & 0\end{bmatrix}$
Scalar multiplication	The product *c***A** of a number *c* (also called a scalar in the parlance of abstract algebra) and a matrix **A** is computed by multiplying every entry of **A** by *c*: $(c\mathbf{A})_{i,j} = c \cdot \mathbf{A}_{i,j}$. This operation is called *scalar multiplication*, but its result is not named "scalar product" to avoid confusion, since "scalar product" is sometimes used as a synonym for "inner product".	$2 \cdot \begin{bmatrix}1 & 8 & -3\\4 & -2 & 5\end{bmatrix} = \begin{bmatrix}2\cdot 1 & 2\cdot 8 & 2\cdot -3\\2\cdot 4 & 2\cdot -2 & 2\cdot 5\end{bmatrix} = \begin{bmatrix}2 & 16 & -6\\8 & -4 & 10\end{bmatrix}$
Transposition	The *transpose* of an *m*-by-*n* matrix **A** is the *n*-by-*m* matrix $\mathbf{A}^{\mathrm{T}}$ (also denoted $\mathbf{A}^{\mathrm{tr}}$ or ${}^{\mathrm{t}}\mathbf{A}$) formed by turning rows into columns and vice versa: $(\mathbf{A}^{\mathrm{T}})_{i,j} = \mathbf{A}_{j,i}$.	$\begin{bmatrix}1 & 2 & 3\\0 & -6 & 7\end{bmatrix}^{\mathrm{T}} = \begin{bmatrix}1 & 0\\2 & -6\\3 & 7\end{bmatrix}$

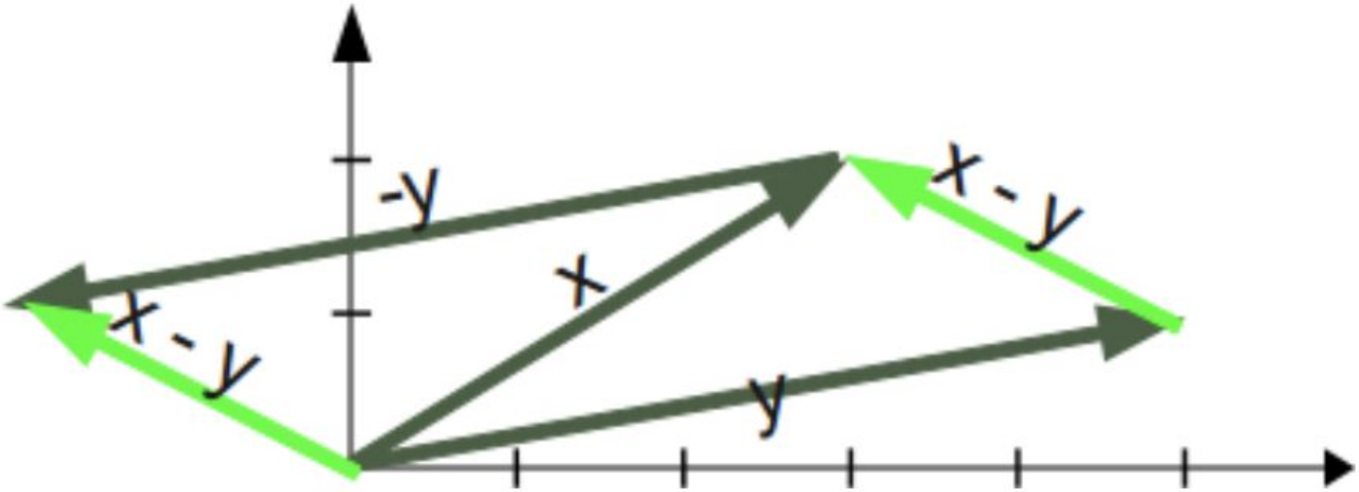
Numpy Arrays

8 min

1. Create two vectors a and b in numpy and
   a. Add them together and put the result in c.
   b. Multiply them together (element-wise)
   c. Divide a by b (element-wise)
   d. Obtain the modulo (element-wise)




```
>>> x = np.array([3,2])
>>> y = np.array([5,1])
>>> z = x + y
>>> z
array([8, 3])
```




```
>>> x = np.array([3,2])
>>> y = np.array([5,1])
>>> z = x - y
>>> z
array([-2, 1])
```

Teacher Demo: Matrix Multiplication

MAKE SCHOOL

1. Vector times a vector
2. Vector times a matrix
3. Matrix times a matrix
4. $AB \neq BA$

Matrix Multiplication

5 mins

On paper, Calculate:

$$\begin{bmatrix} 1 & 2 \\ 0 & 1 \\ 2 & 3 \end{bmatrix} \times \begin{bmatrix} 2 \\ 6 \end{bmatrix} =$$

$$\begin{bmatrix} 1 & 2 \\ 0 & 1 \\ 2 & 3 \end{bmatrix} \times \begin{bmatrix} 2 & 5 \\ 6 & 7 \end{bmatrix} =$$

Numpy Matrix Multiplication

```
a = np.array([[1, 0],
              [0, 1]])
b = np.array([[4, 1],
              [2, 2]])
np.matmul(a, b)
array([[4, 1],
       [2, 2]])
```

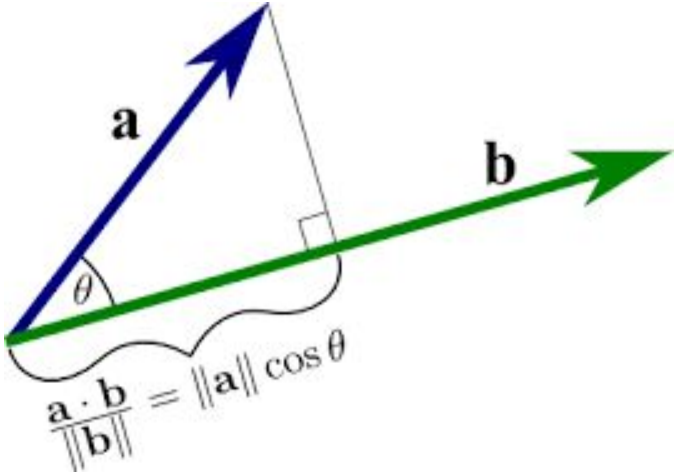
Note: Don't confuse a * b with np.matmul(a,b). The first one is element-wise multiplication not a matrix multiplication!

Break 10 mins

Scalar Product = Dot Product = Inner Product

The dot product of two vectors a = [a1, a2, ..., an] and b = [b1, b2, ..., bn] is defined as:

$$\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$



Meaning of Dot Product

Reference

1. https://www.python-course.eu/matrix_arithmetic.php