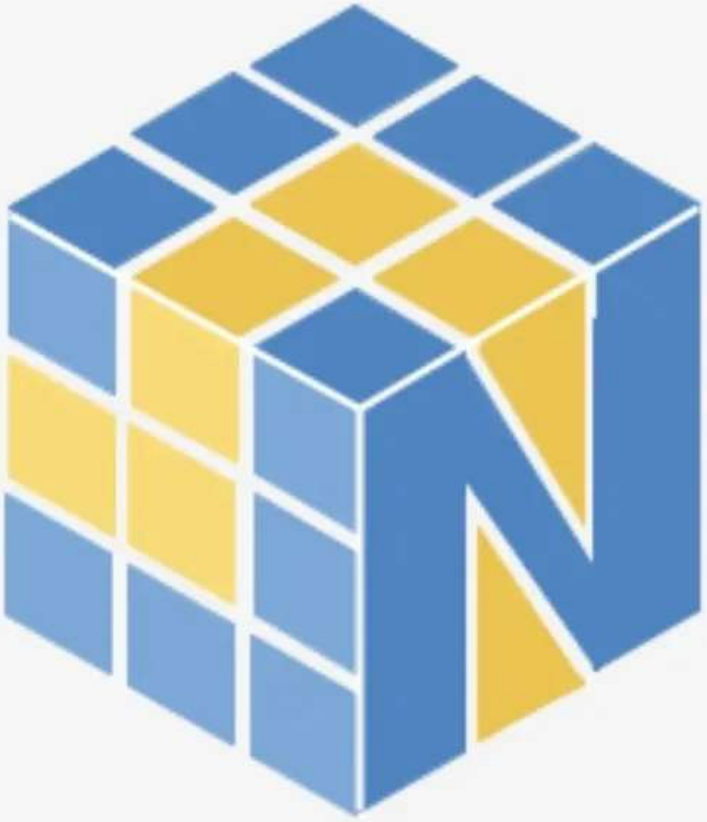
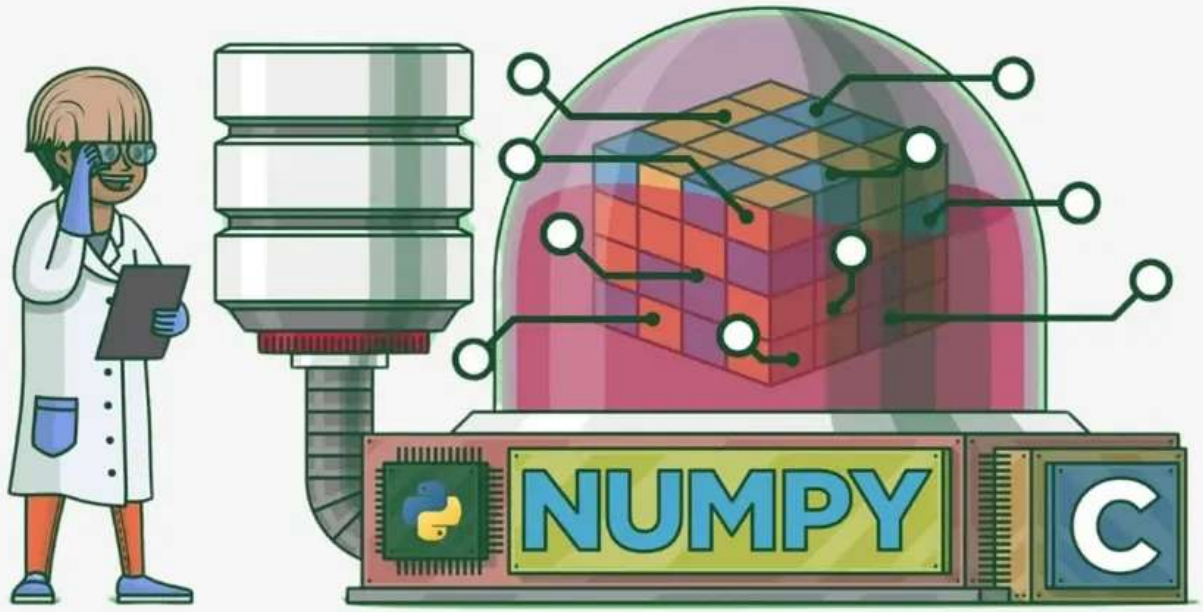
Explaining Numpy

swipe



Introduction

NumPy (**Num**erical **Py**thon) is a powerful Python library used for **numerical computing**. It provides support for **large**, **multi-dimensional arrays** and **matrices**, along with a **collection of mathematical functions** to operate on these arrays efficiently.



Why do I need Numpy?

Imagine you're doing some serious number crunching in **Python**. You're adding, subtracting, multiplying, dividing, and working with big sets of numbers.

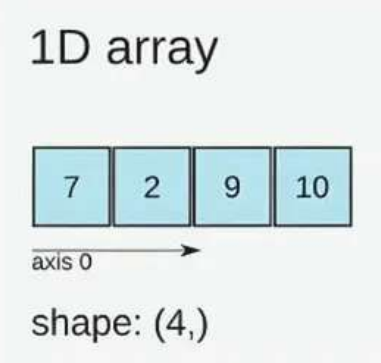
Python can handle all that, but sometimes **it's not as fast or as convenient** as you'd like, especially when you're dealing with **large amounts of data.**

Foundation of NumPy

The central figure in NumPy is the ndarray (n-dimensional array).

Ndarray vs List

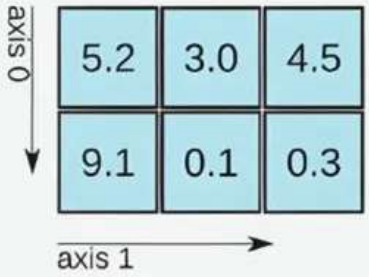
Ndarray is much **faster** and more efficient than Python lists, but unlike lists all **elements** of ndarray needs to be **of same type**



One dimension arrays (1D) represent **vectors**

Matrices and Beyond

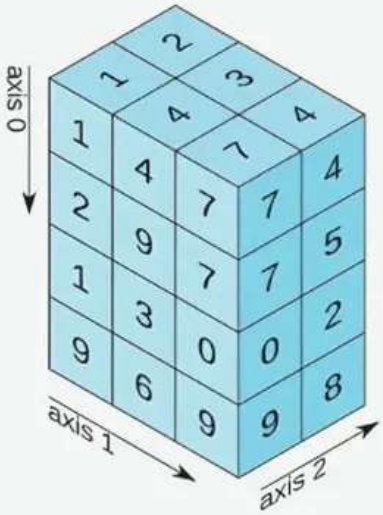
2D array



shape: (2, 3)

Two-dimensional arrays (2D) represent **matrices**

3D array



shape: (4, 3, 2)

Higher dimensional arrays represent **tensors**

Getting started

Installing: pip install numpy
Importing: import numpy as np

Creating arrays:

```
array1 = np.array([1, 2, 3, 4, 5]) #1-dimensional
array2 = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]]) #2-dim

zeros_array = np.zeros(5) - #result: [0,0,0,0,0]
range_array = np.arange(0, 10, 2) #result: [0,2,4,6,8]
```

```
arr = np.array([3, 1, 4, 1, 5, 9, 2, 6, 5])

# Adding elements
arr = np.append(arr, [7, 8])

# Removing elements
arr = np.delete(arr, [0, 1])

# Sorting elements
arr = np.sort(arr)
```

Command

```
np.array([1,2,3])
```

NumPy Array

1
2
3

Effortless Data Extraction

Indexing: allows you to access specific elements of an array using their position

	data
0	1
1	2
2	3

data[0]
1

data[1]
2

Slicing: allows you to create a subarray by specifying a range of indices, making it easy to work with subsets of an array.

data[0:2]
1
2

data[1:]
2
3

data[-2:]
2
3

	data	
0	1	
1	2	-2
2	3	-1
3		

Why is Numpy so cool?

Instead of writing a bunch of for loops to do math with each element of a list, you can use **NumPy** to **perform those operations on entire arrays at once.**

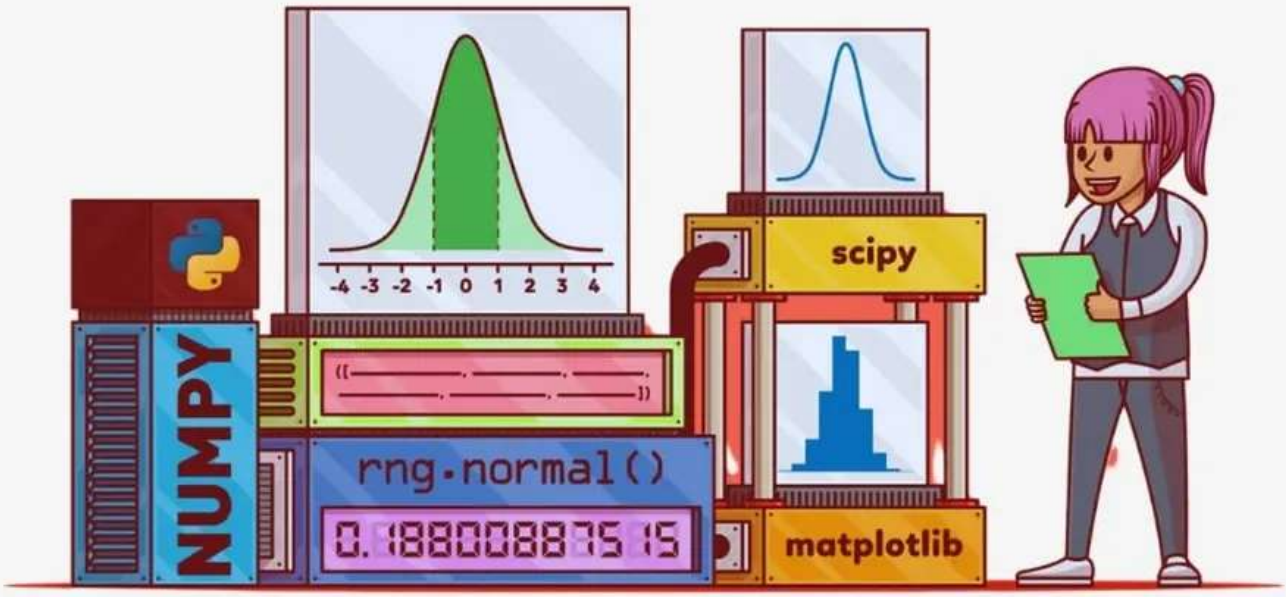
```python
# Using Python lists
python_list1 = [1, 2, 3, 4, 5]
python_list2 = [6, 7, 8, 9, 10]
result_list = [x + y for x, y in zip(python_list1, python_list2)]
print("Result using Python lists:", result_list)

# Using NumPy arrays
import numpy as np

numpy_array1 = np.array([1, 2, 3, 4, 5])
numpy_array2 = np.array([6, 7, 8, 9, 10])
result_array = numpy_array1 + numpy_array2
print("Result using NumPy arrays:", result_array)
```

Just add them directly
Cool, right?

This not only makes your code cleaner and easier to read, but it also makes it way faster.

Who uses Numpy?

NumPy is widely used by **data scientists**, researchers, engineers, and developers for **numerical computing, data analysis, machine learning, scientific computing,** and more.



It's an essential tool for anyone working with numerical data in Python.