

FSDS Gen A1

Data Science Libraries

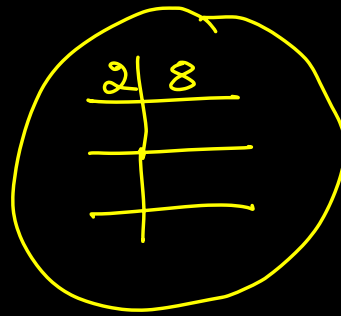
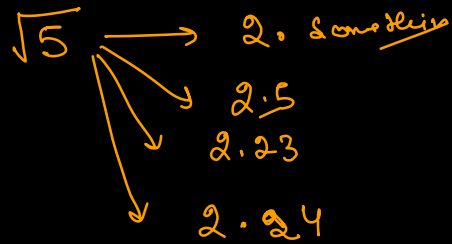
- ✓ 1. Why libraries are required
- ✓ 2. DS libraries →
- ✓ 3. How to install them
- ✓ 4. Import & usage
- ✓ 5. ML Algo ⇒ Code + Algo

Numpy

Pandas

Matplotlib +

Seaborn



$$2\sqrt{9}$$

$$35 \times 35 =$$

$$85 \times 85 = 7225$$

$$\underline{95} \times \underline{95} = 9025$$

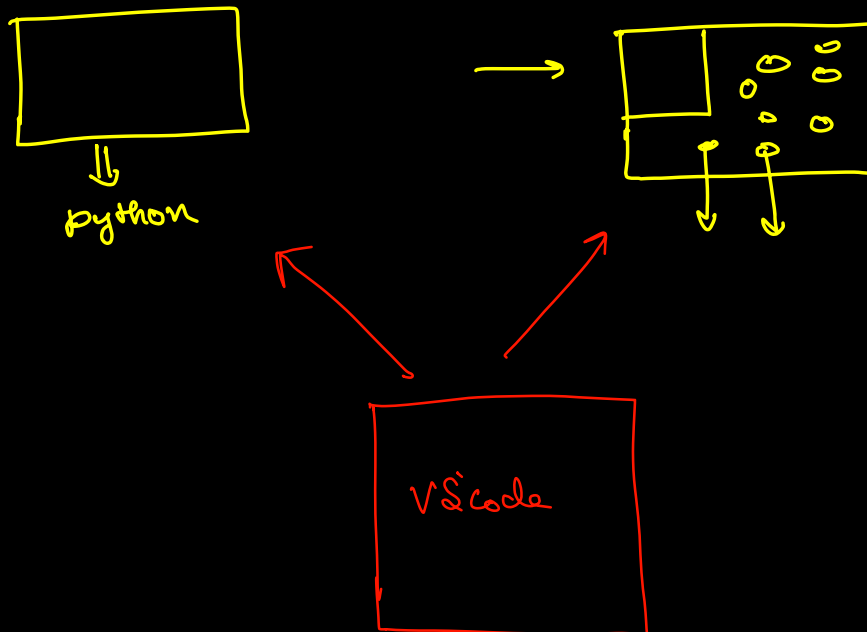
$$\underbrace{\quad}_{9 \times (9+1)} = 9025$$

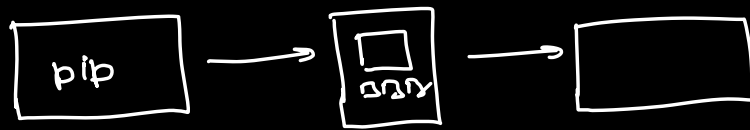
$$125 \times 125 = 156$$

- very complex
- implementation can be made much more efficient
- Our ~~new~~ implementation might not be best.

5& ⇐

pip install <package-name>





Environments

→ python

1000
resources

Numpy library

matlab }
Java } \Rightarrow python
C++ }

Numerical python \Rightarrow

Fast computation with n -dimⁿ array

Numpy

\rightarrow it has just one data structure
nd arrays [n -dimⁿ]

\rightarrow import numpy as np
np.

stats, linear algebra \Rightarrow arrays

List = [
 [1], [1], [1],]

very slow

numpy \Rightarrow they are getting executed
with C++ in background
and are a lot faster.

Arrays

Fixed size
(new created
if changed)

homogeneous

Good for mathematical
fn

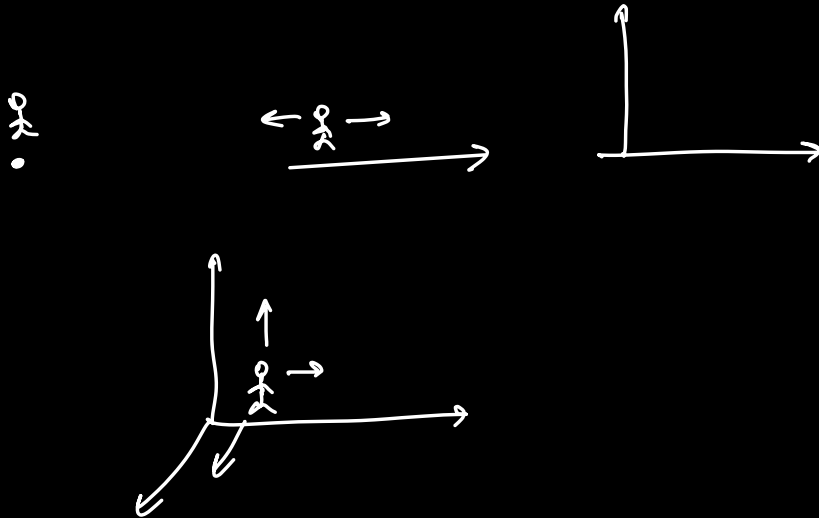
List

list are mutable

heterogeneous

we don't have
that
here.

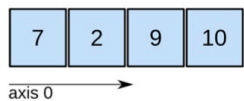
Creating numpy arrays



Creation of NumPy Arrays

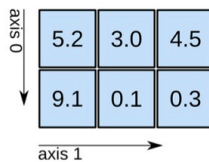
- `Np.array (Sequence)`

1D array



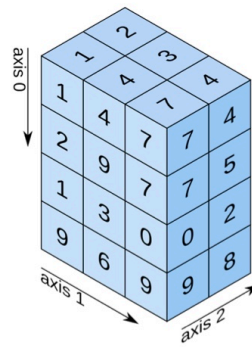
shape: (4,)

2D array



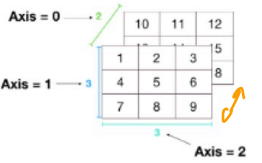


shape: (2, 3)

3D array



shape: (4, 3, 2)

Data	NumPy	Details
	<code>array([1, 2, 3])</code>	<ul style="list-style-type: none"> Names: Array, vector 1-dimensional Shape = (1, 3)
	<code>array([[1. , 2. , 3.3], [4. , 5. , 6.5]])</code>	<ul style="list-style-type: none"> Names: Array, matrix More than 1-dimension Shape = (2, 3)
	<code>array([[[1, 2, 3], [4, 5, 6], [7, 8, 9]], [[10, 11, 12], [13, 14, 15], [16, 17, 18]])</code>	<ul style="list-style-type: none"> Names: Array, matrix More than 1-dimension Shape = (2, 3, 3)

`np.array`

→ type

→ without using type

`np.arange`

`np.ones`

`np.zeros`

`np.random.random`

`np.linspace`

`np.identity` `np.eye`

int 8
↓
2 bytes

int 32
↓
32

u1
↓
u

Data Types in NumPy

NumPy has some extra data types, and refer to data types with one character, like `i` for integers, `u` for unsigned integers etc.

Below is a list of all data types in NumPy and the characters used to represent them.

- `i` - integer
- `b` - boolean
- `u` - unsigned integer
- `f` - float
- `c` - complex float
- `m` - timedelta
- `M` - datetime
- `O` - object
- `S` - string
- `U` - unicode string
- `V` - fixed chunk of memory for other type (void)

