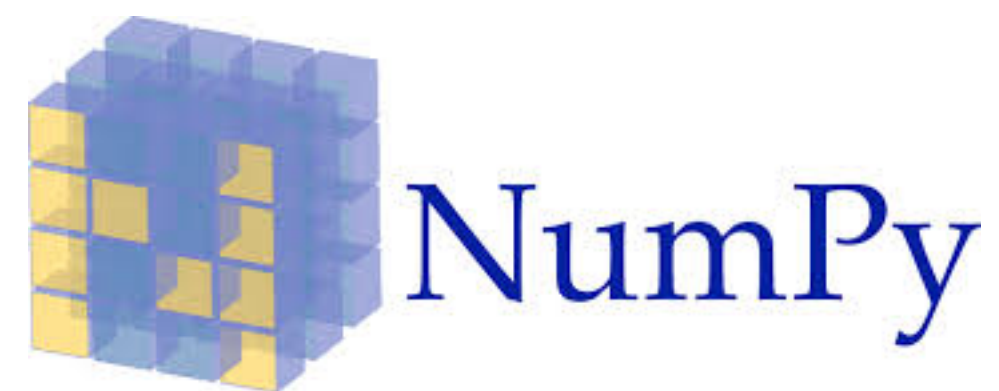
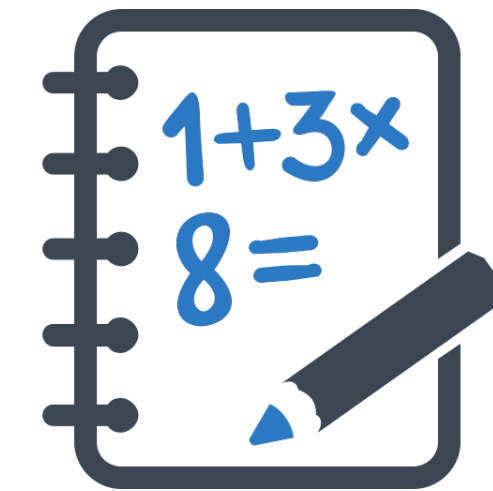
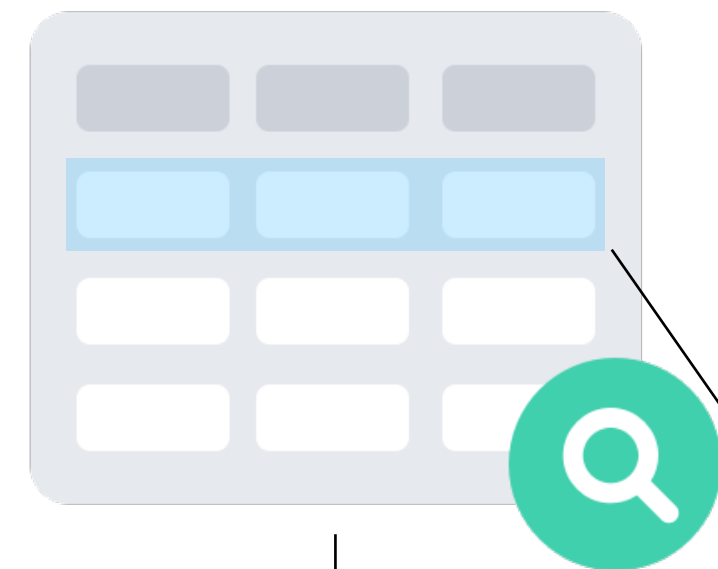


# What is NumPy?



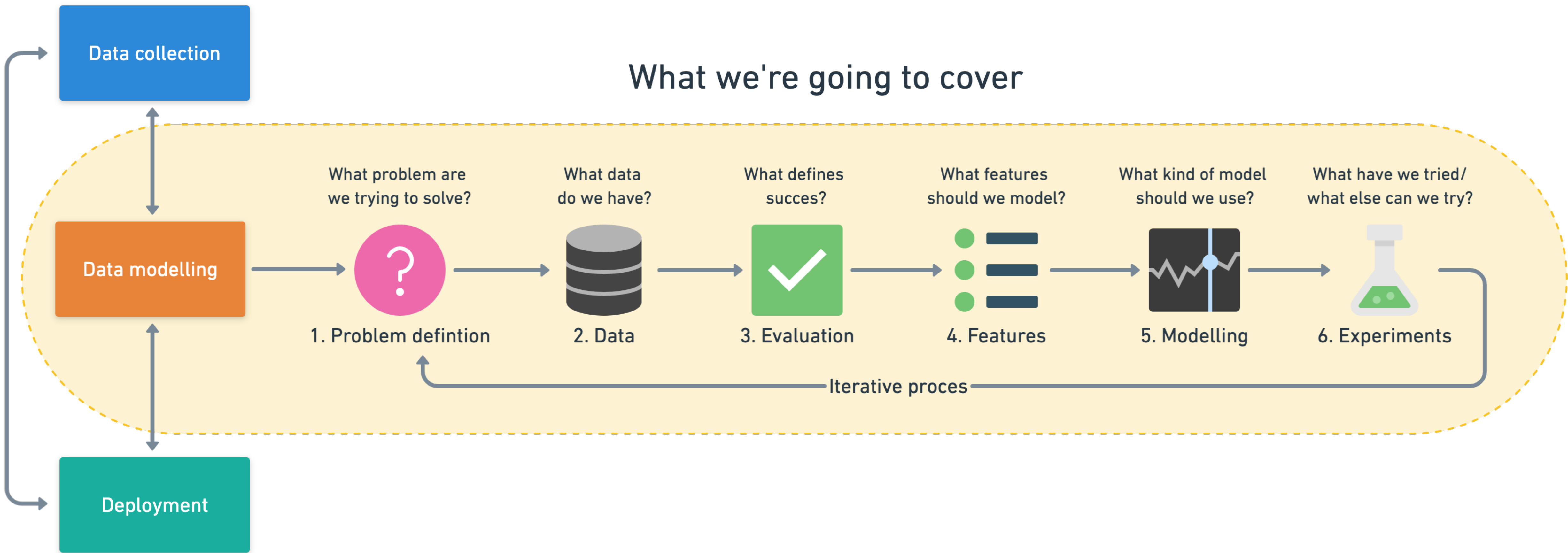
Data



5	0	3
3	7	9
3	5	2

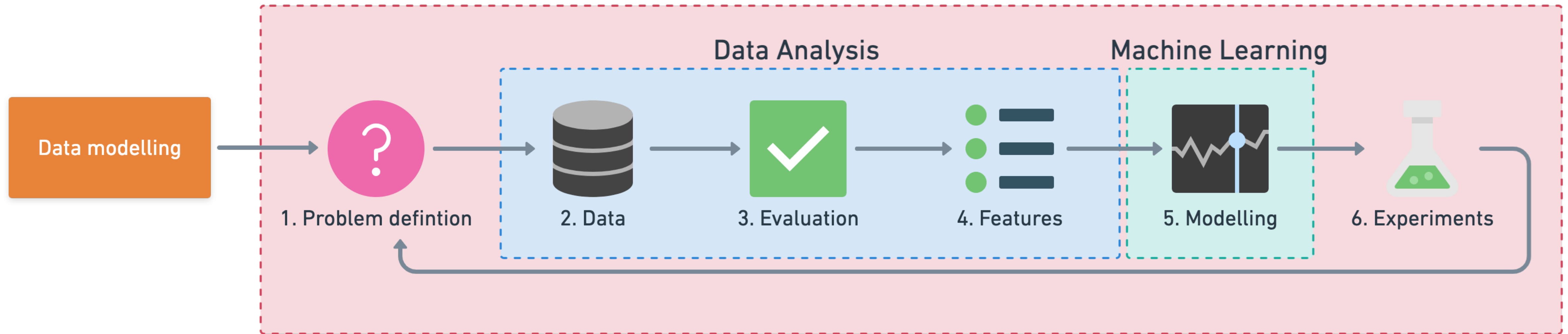
5	0	3
---	---	---

Steps in a full machine learning project



# Tools you can use

## Data Science



# Why NumPy?

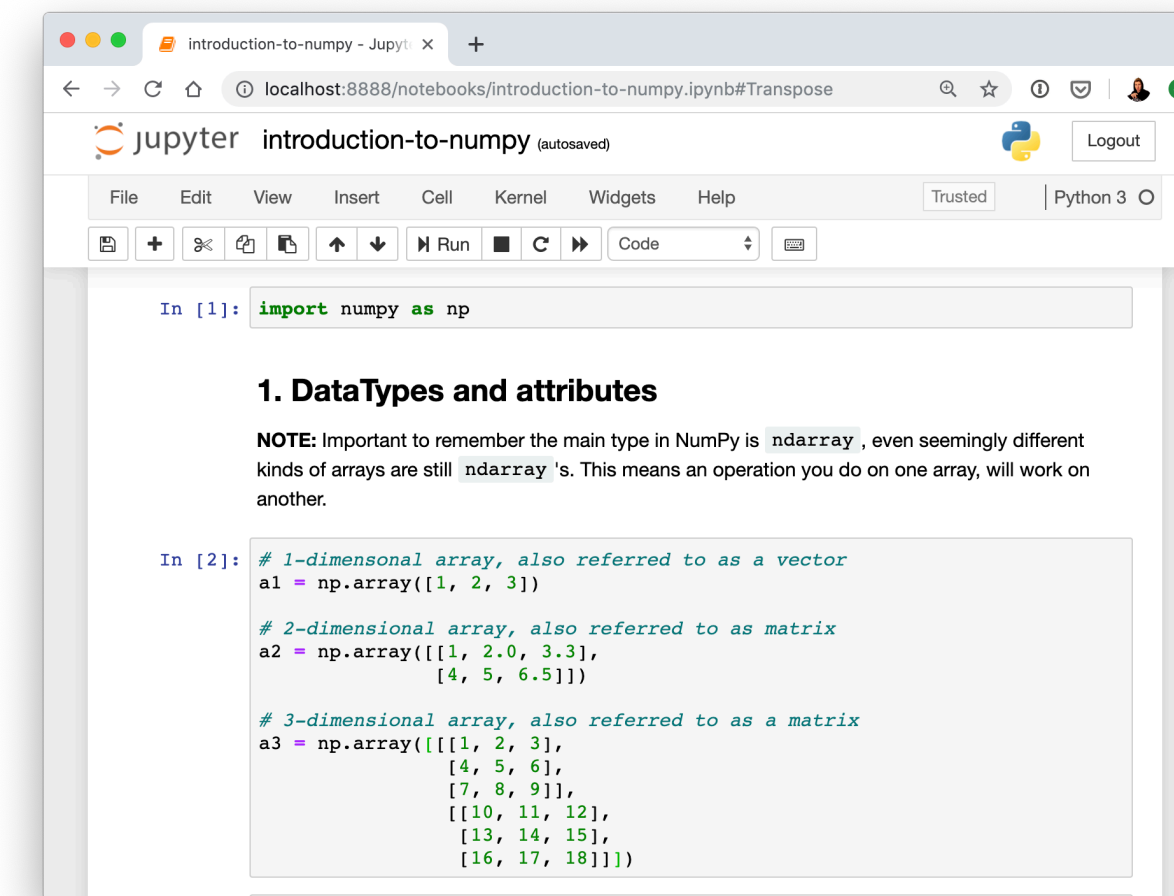
- **It's fast**
- **Behind the scenes optimizations written in C**
- **Vectorization via broadcasting (avoiding loops)**
- **Backbone of other Python scientific packages**

# What are we going to cover?

- **Most useful functions**
- **NumPy datatypes & attributes (ndarray)**
- **Creating arrays**
- **Viewing arrays & matrices**
- **Manipulating & comparing arrays**
- **Sorting arrays**
- **Use cases**

# Where can you get help?

- Follow along with the code
- Try it for yourself
- Search for it
- Try again
- Ask



A screenshot of a Jupyter Notebook interface. The browser address bar shows 'localhost:8888/notebooks/introduction-to-numpy.ipynb#Transpose'. The notebook title is 'introduction-to-numpy (autosaved)'. The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar. The code cell shows the following content:

```
In [1]: import numpy as np
```

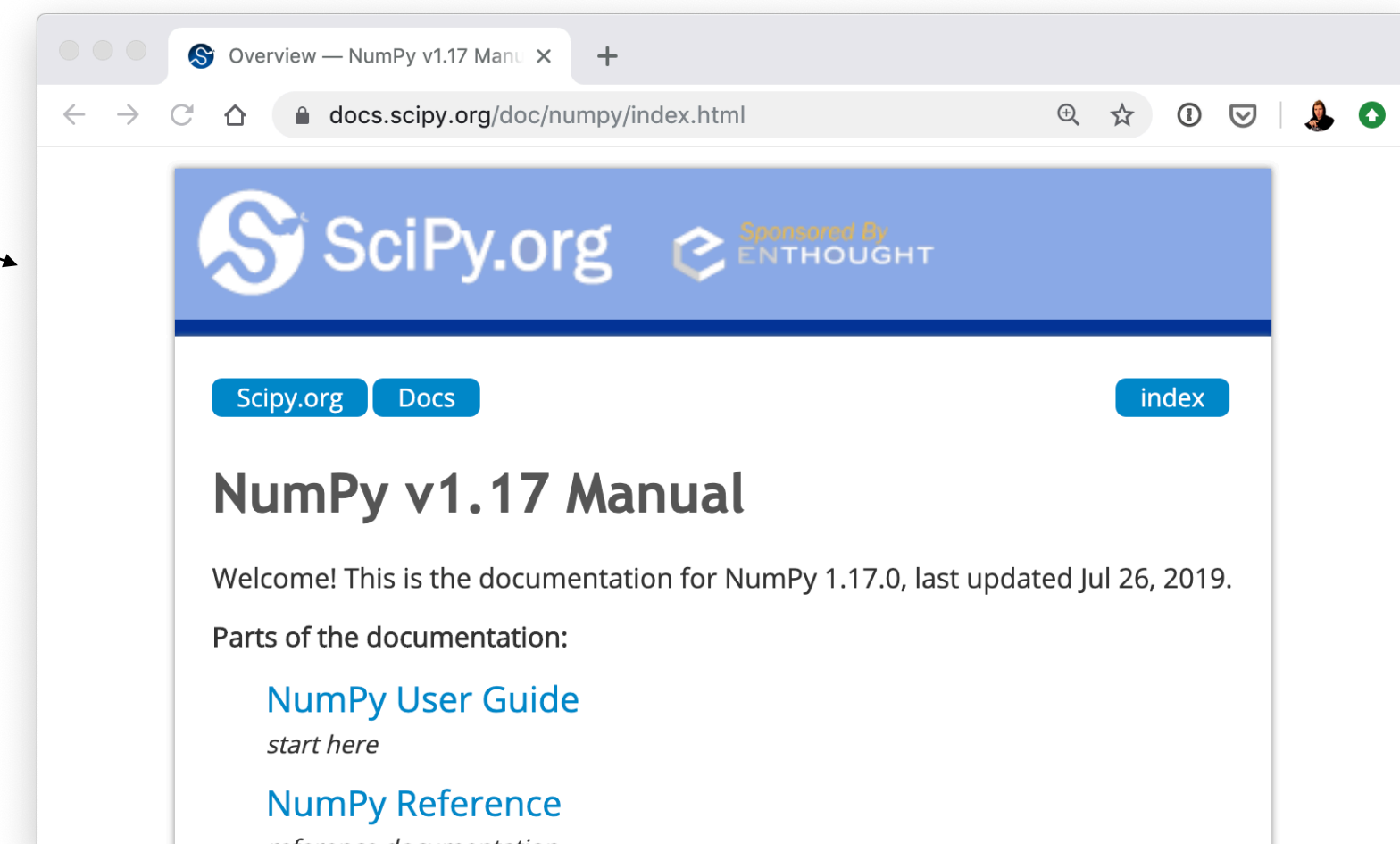
**1. DataTypes and attributes**

**NOTE:** Important to remember the main type in NumPy is `ndarray`, even seemingly different kinds of arrays are still `ndarray`'s. This means an operation you do on one array, will work on another.

```
In [2]: # 1-dimensional array, also referred to as a vector
a1 = np.array([1, 2, 3])

# 2-dimensional array, also referred to as matrix
a2 = np.array([[1, 2.0, 3.3],
               [4, 5, 6.5]])

# 3-dimensional array, also referred to as a matrix
a3 = np.array([[[1, 2, 3],
                [4, 5, 6],
                [7, 8, 9]],
               [[10, 11, 12],
                [13, 14, 15],
                [16, 17, 18]]])
```



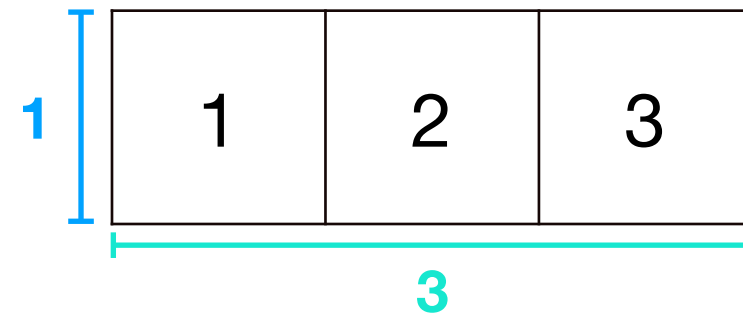
# Let's code!



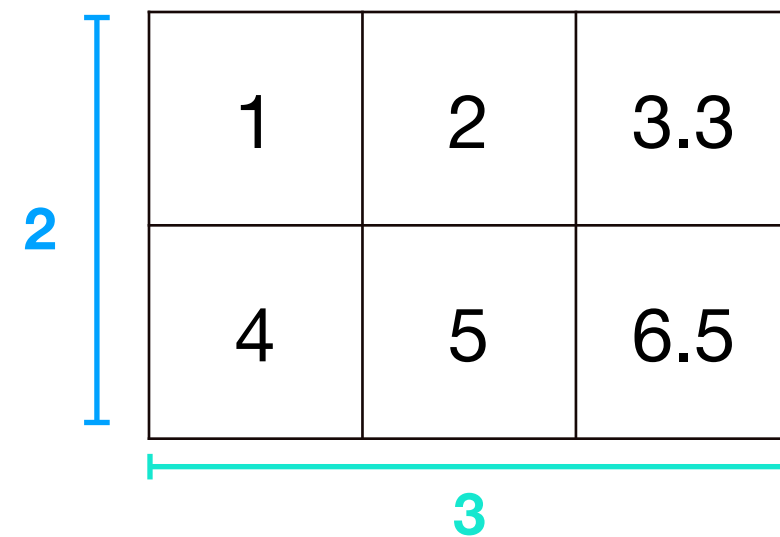


# Anatomy of a NumPy array

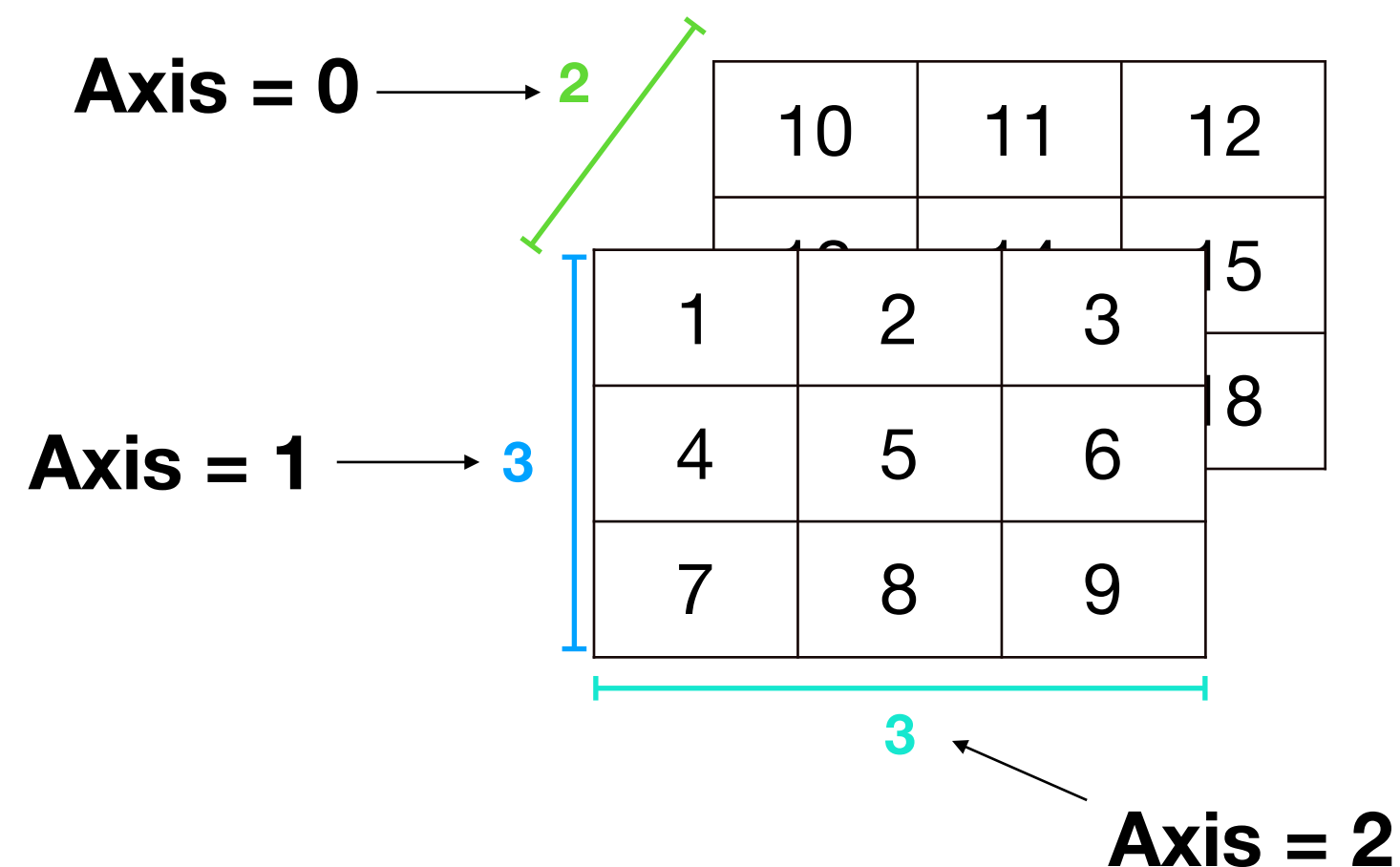
## Data



1	2	3
---	---	---



1	2	3.3
4	5	6.5



10 11 12			5
1	2	3	
4	5	6	8
7	8	9	

## NumPy

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

```
array([[1. , 2. , 3.3],  
       [4. , 5. , 6.5]])
```

```
array([[[ 1,  2,  3],  
        [ 4,  5,  6],  
        [ 7,  8,  9]],
```

```
       [[10, 11, 12],  
        [13, 14, 15],  
        [16, 17, 18]])])
```

## Details

- Names: Array, vector
- 1-dimensional
- Shape = (1, 3)

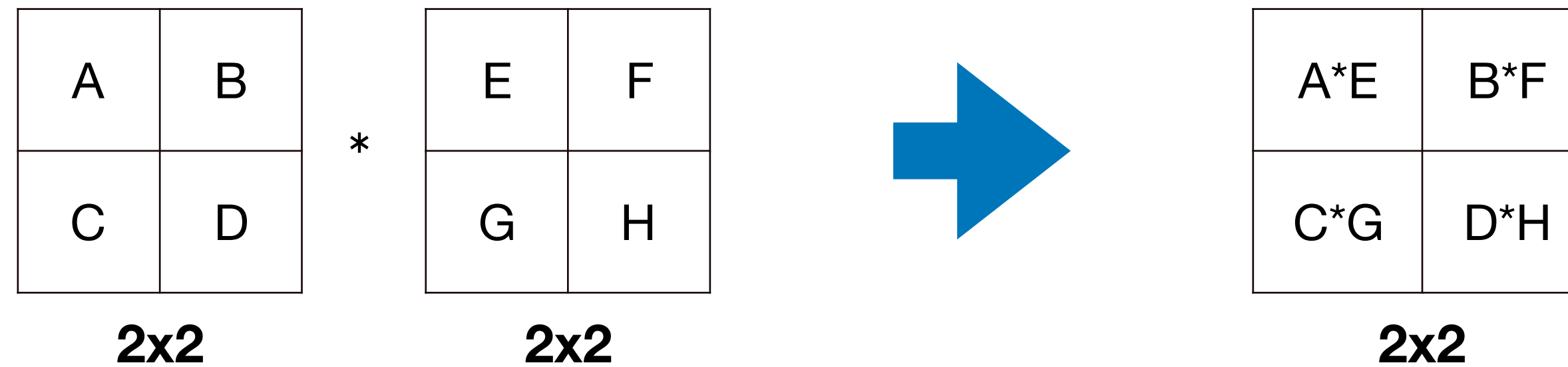
- Names: Array, matrix
- More than 1-dimension
- Shape = (2, 3)

- Names: Array, matrix
- More than 1-dimension
- Shape = (2, 3, 3)

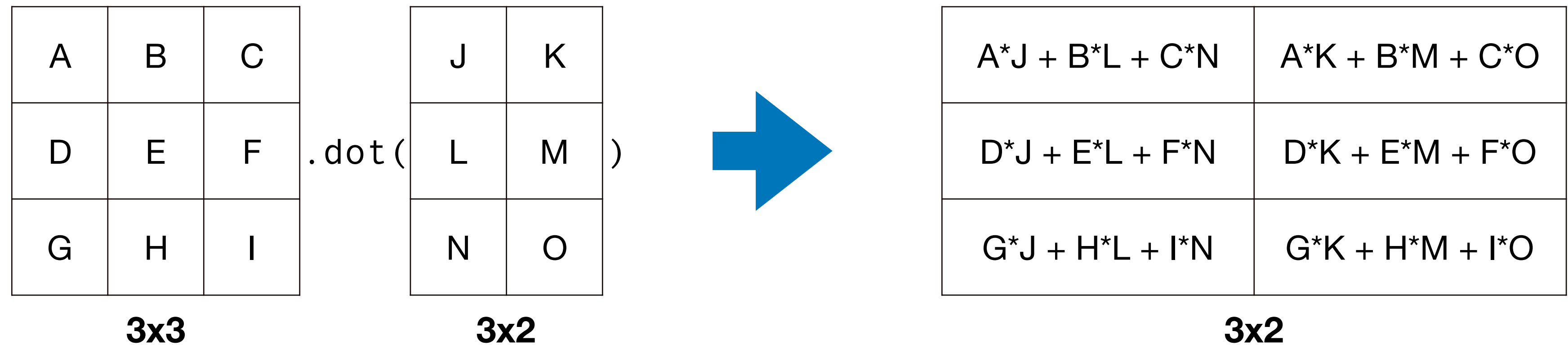


# Dot product vs. element-wise

**Element-wise**



**Dot product**



# Dot product

