



# Introduction to Numpy

---

Diego Useche - [dh.useche@uniandes.edu.co](mailto:dh.useche@uniandes.edu.co)

Métodos Computacionales I

Physics Department, Universidad de los Andes, Bogotá



# Numpy



**Numpy Array  
Special Math  
Operations**


# Numpy

github.com/numpy/numpy

Gmail YouTube Google Scholar Maps Consulta de Proces... Panel - Oficina Virtual Ekogui 5722 PIEZAS CRÉDI... Iniciar Sesión - TMS.

pavement.py	REL: Prepare main for NumPy 2.0.0 development	9 months ago
pyproject.toml	ENH: add a pkg-config file and a numpy-config script	last week
pytest.ini	MAINT: Remove a now-redundant warning ignore related to...	2 months ago

README Code of conduct License Security



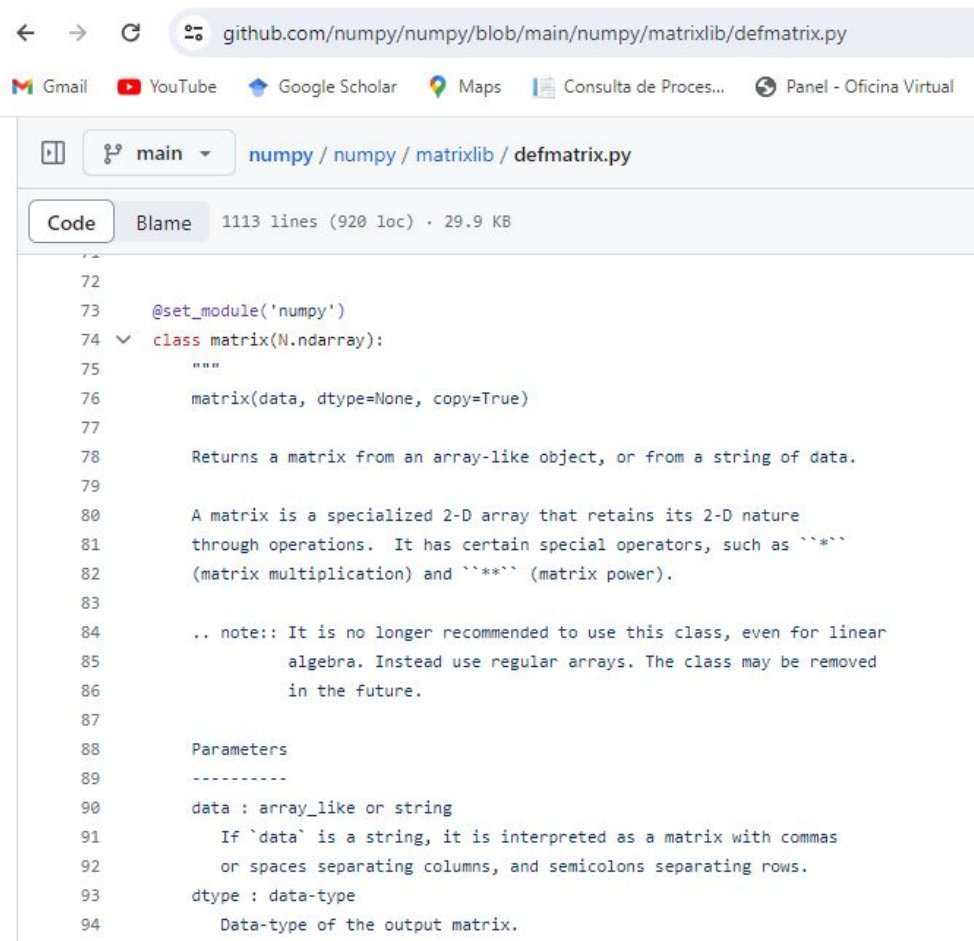
powered by NumFOCUS PyPI downloads 222M/month Conda downloads 70M stackoverflow Ask questions

DOI 10.1038/s41586-020-2649-2 openssf scorecard 8.7

NumPy is the fundamental package for scientific computing with Python.

- Website: <https://www.numpy.org>
- Documentation: <https://numpy.org/doc>

# Numpy



```
72
73     @set_module('numpy')
74     class matrix(N.ndarray):
75         """
76         matrix(data, dtype=None, copy=True)
77
78         Returns a matrix from an array-like object, or from a string of data.
79
80         A matrix is a specialized 2-D array that retains its 2-D nature
81         through operations. It has certain special operators, such as ``*``
82         (matrix multiplication) and ``**`` (matrix power).
83
84         .. note:: It is no longer recommended to use this class, even for linear
85                   algebra. Instead use regular arrays. The class may be removed
86                   in the future.
87
88         Parameters
89         -----
90         data : array_like or string
91             If `data` is a string, it is interpreted as a matrix with commas
92             or spaces separating columns, and semicolons separating rows.
93         dtype : data-type
94             Data-type of the output matrix.
```

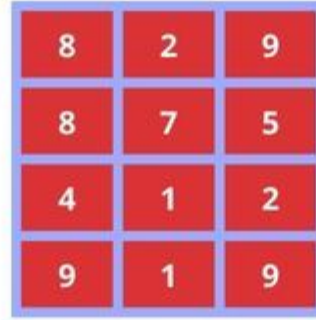
# Numpy arrays

We can have one dimensional, two dimensional, three dimensional, nth dimensional arrays.



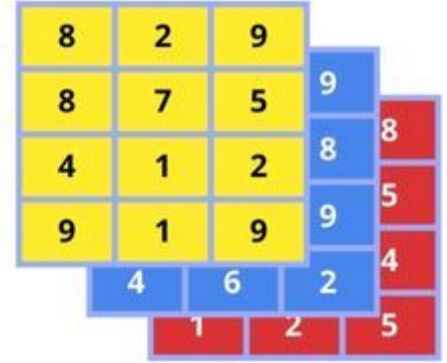
1

```
my_np.array([4,5,9])
```



2

```
my_np.array([[8,2,9],  
            [8,7,5],  
            [4,1,2],  
            [9,1,9]])
```



3

```
my_np.array([[[8,2,9],  
            [8,7,5],[4,1,2],[9,1,9]],  
            [[4,5,9],[4,2,8],  
            [2,1,9],[4,6,2]],  
            [[6,8,8],[2,4,5],  
            [7,8,4],[1,2,5]]])
```

# Numpy array axes

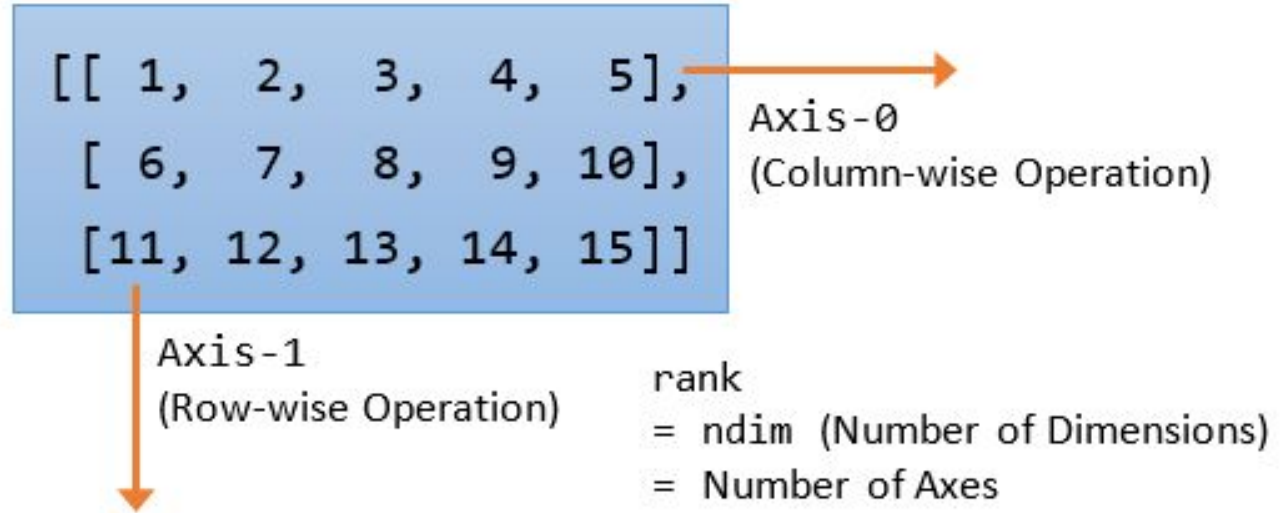


Image taken from

[https://www3.ntu.edu.sg/home/ehchua/programming/webprogramming/Python4\\_DataAnalysis.html](https://www3.ntu.edu.sg/home/ehchua/programming/webprogramming/Python4_DataAnalysis.html)

# Numpy array shape

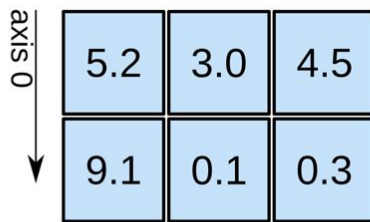
1D array



axis 0 →

shape: (4,)

2D array

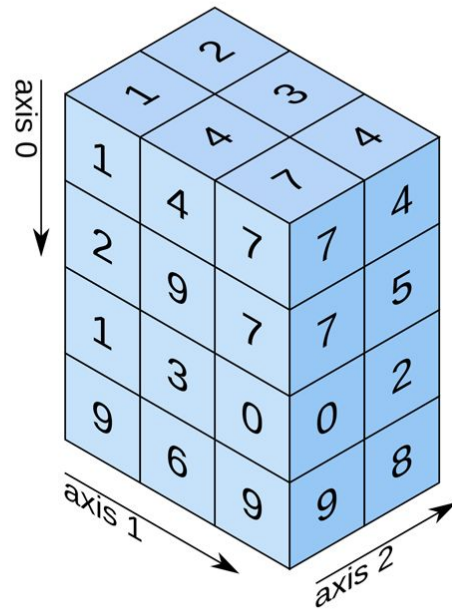


axis 0 ↓

axis 1 →

shape: (2, 3)

3D array



axis 0 ↓

axis 1 →

axis 2 →

shape: (4, 3, 2)

# Numpy indexing

A 3x3 grid representing a NumPy array. The columns are indexed 0, 1, 2 from left to right. The rows are indexed 0, 1, 2 from top to bottom. Each cell contains a coordinate pair (row, column). An arrow labeled 'Column Index' points to the column headers. An arrow labeled 'Row Index' points to the row headers.

	0	1	2
0	(0,0)	(0,1)	(0,2)
1	(1,0)	(1,1)	(1,2)
2	(2,0)	(2,1)	(2,2)



# Numpy indexing

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Row one, columns two to four

```
>>> arr[1, 2:4]
array([7, 8])
```

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

All rows in column one

```
>>> arr[:, 1]
array([2, 6, 10, 14])
```

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

All rows after row two,  
all columns after column two

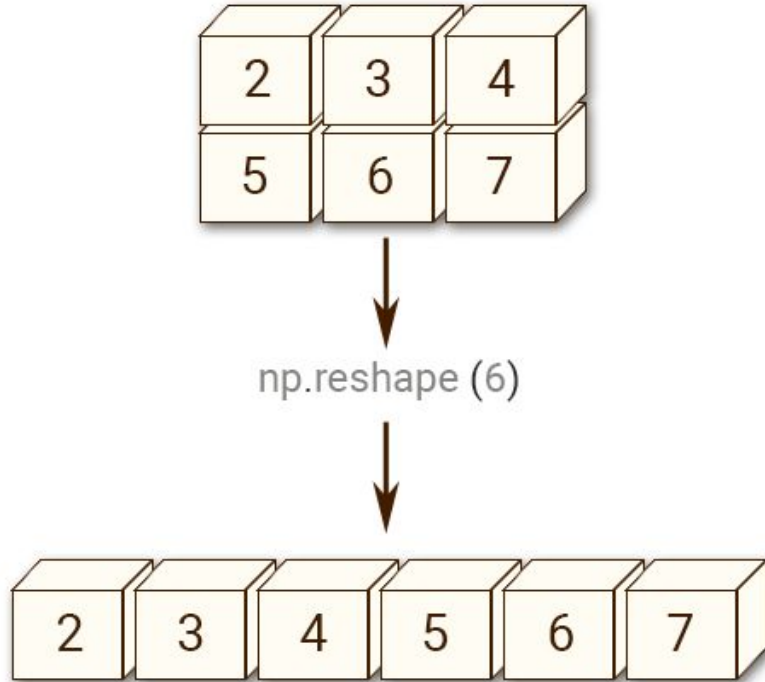
```
>>> arr[2:, 2:]
array([[11, 12],
       [15, 16]])
```

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

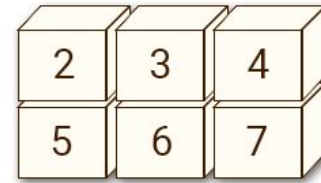
Every other row after row one,  
every other column

```
>>> arr[1::2, ::2]
array([[5, 7],
       [13, 15]])
```

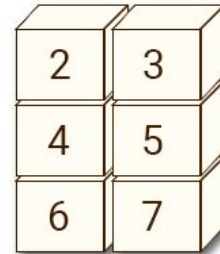
# Numpy reshape



© w3resource.com



`np.reshape(3, 2)`



© w3resource.com

# Numpy operations

1	2
0	3

+

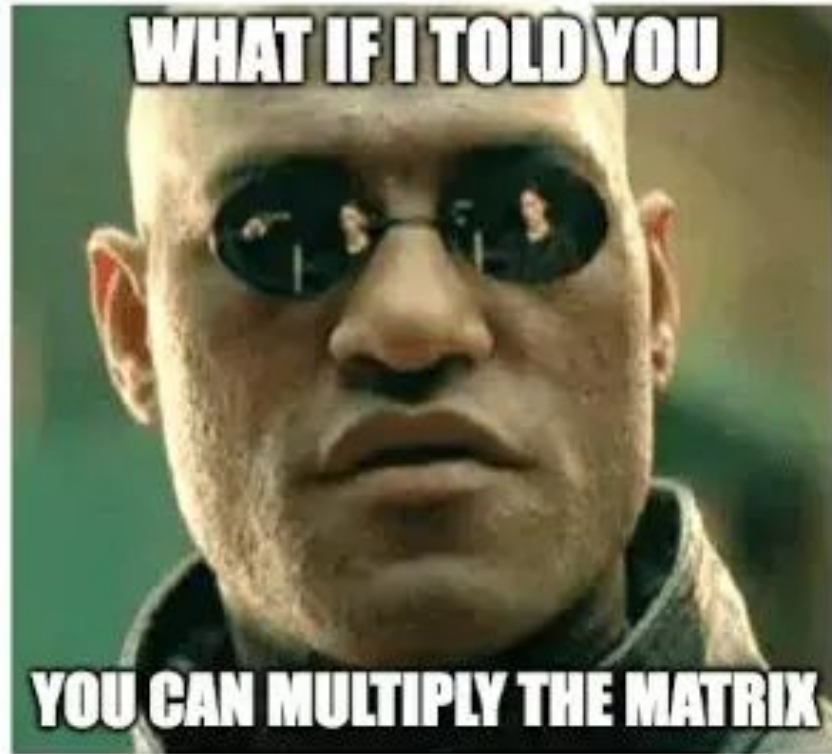
4	1
2	2

=

5	3
2	5

Elementwise Sum

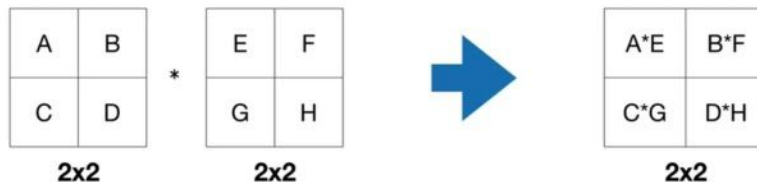
# Numpy operations



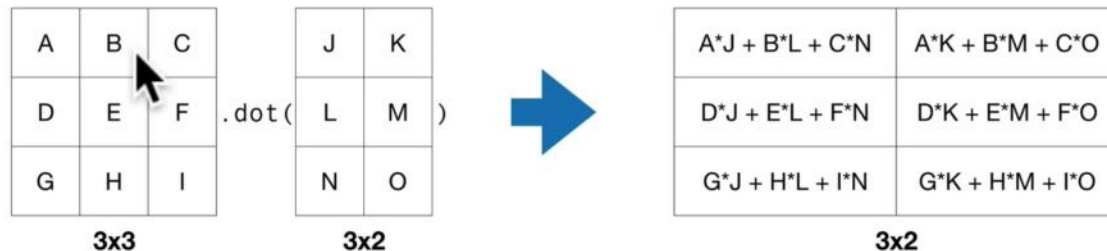
# Numpy operations

## Dot product vs. element-wise

Element-wise

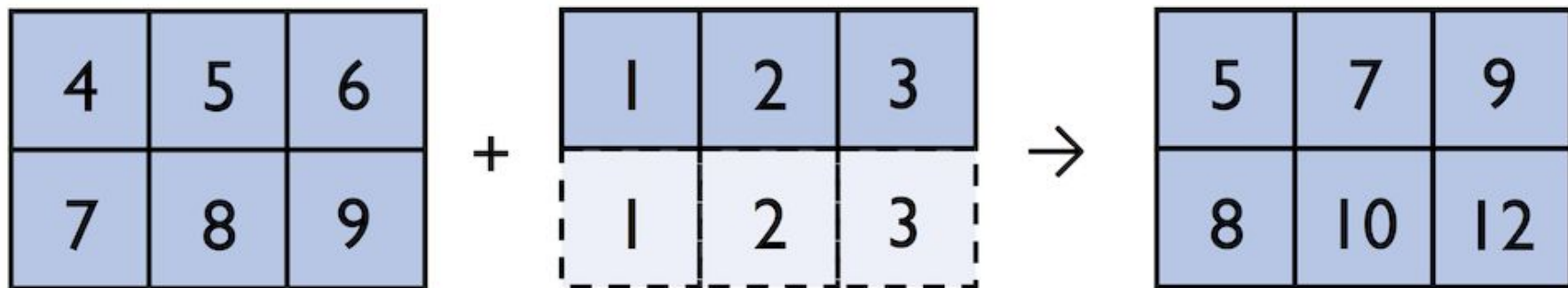


Dot product



## Numpy broadcasting

```
np.array([[4, 5, 6],  
         [7, 8, 9]]) + np.array([1, 2, 3]):
```



# Numpy broadcasting

The diagram illustrates NumPy broadcasting with the following components:

- Input Array 1:** A light blue 2x1 array containing the values 1 and 2.
- Operator:** A multiplication symbol (\*) followed by the scalar value 1.6 in purple.
- Equality:** An equals sign (=).
- Input Array 2:** A light blue 2x1 array containing the values 1 and 2.
- Operator:** A multiplication symbol (\*) followed by a 2x1 array of purple cells, each containing the value 1.6.
- Equality:** An equals sign (=).
- Output Array:** A white 2x1 array containing the values 1.6 and 3.2.

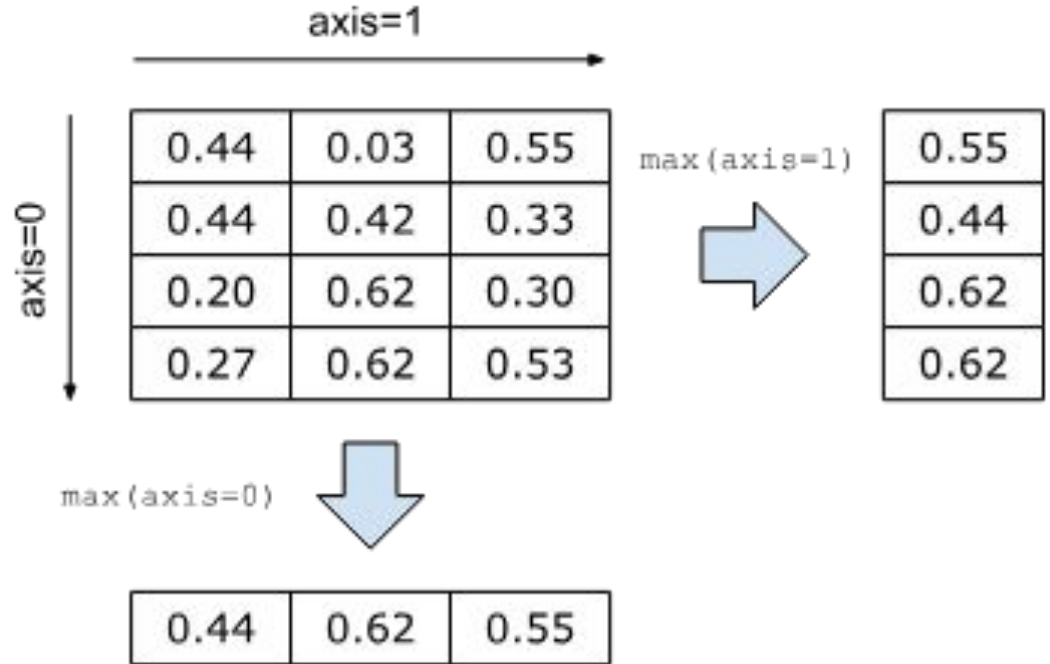
# Numpy axis operations

`my_array.max(axis = 0)`

will return the maximum value by moving along the rows

`my_array.sum(axis = 1)`

will sum the elements by moving along the columns





# References

---

<https://github.com/numpy/numpy/tree/main>

<https://www.analyticsvidhya.com/blog/2021/05/mathematical-operations-in-python-with-numpy/>

[https://www.youtube.com/watch?v=dm8fpMvRZQY&ab\\_channel=RaiGurukul](https://www.youtube.com/watch?v=dm8fpMvRZQY&ab_channel=RaiGurukul)

<https://www.plus2net.com/python/numpy-ndarray.php>

[https://www3.ntu.edu.sg/home/ehchua/programming/webprogramming/Python4\\_DataAnalysis.html](https://www3.ntu.edu.sg/home/ehchua/programming/webprogramming/Python4_DataAnalysis.html)

<https://www.learndatasci.com/tutorials/applied-introduction-to-numpy-python-tutorial/>

<https://iq.opengenus.org/2d-array-in-numpy/>

<https://www.w3resource.com/numpy/manipulation/reshape.php>

<https://sebastianraschka.com/blog/2020/numpy-intro.html>

<https://www.youtube.com/watch?v=uS3Ytql-efk>

<https://datascienceparichay.com/article/numpy-elementwise-sum-two-arrays/>

[https://numpy.org/doc/1.25/user/absolute\\_beginners.html](https://numpy.org/doc/1.25/user/absolute_beginners.html)

<https://mlbookcamp.com/article/numpy>