

# What is a simple explanation of NumPy axes and how are they used in data manipulation?

Authored by  
**stats writer**

June 28, 2024

## RECOMMENDED CITATION

stats writer (2024). *What is a simple explanation of NumPy axes and how are they used in data manipulation?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=155865>

NumPy axes refer to the dimensions or directions along which data is organized in a multi-dimensional array in the NumPy library. These axes are essential in data manipulation as they allow for efficient and flexible operations on the data. For example, in a 2-dimensional array, the first axis represents the rows and the second axis represents the columns. By specifying the desired axis, various mathematical and statistical functions can be applied to specific sections of the data. This allows for easy slicing, merging, and reshaping of data, making data manipulation tasks more efficient and intuitive.

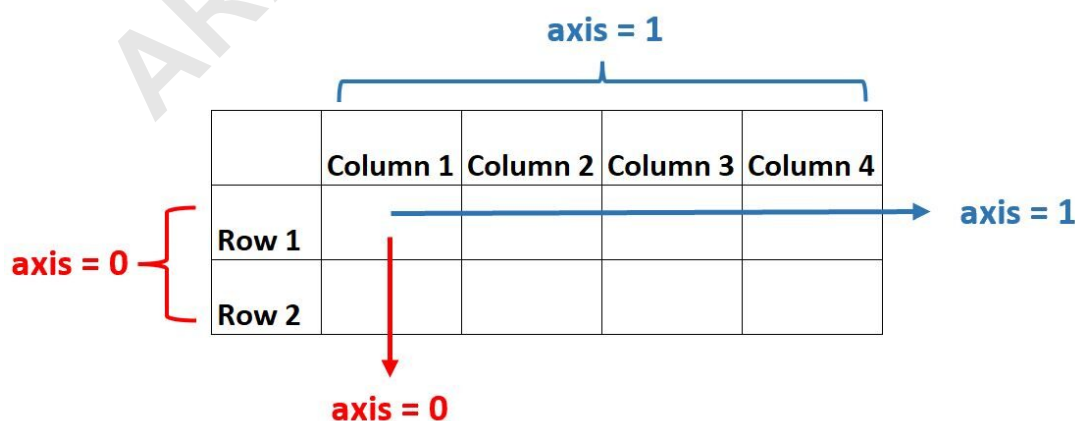
## A Simple Explanation of NumPy Axes (With Examples)

Many functions in require that you specify an axis along which to apply a certain calculation.

Typically the following rule of thumb applies:

**axis=0:** Apply the calculation "column-wise"  
**axis=1:** Apply the calculation "row-wise"

The following image shows a visual representation of the axes on a NumPy matrix with 2 rows and 4 columns:



The following examples show how to use the axis argument in different scenarios with the following NumPy matrix:

```
import numpy as np

#create NumPy matrix
my_matrix = np.matrix(, )

#view NumPy matrix
my_matrix

matrix(,
])
```

**Example 1: Find Mean Along Different Axes**

We can use `axis=0` to find the mean of each column in the NumPy matrix:

```
#find mean of each column in matrix
np.mean(my_matrix, axis=0)

matrix(])
```

The output shows the mean value of each column in the

**matrix.**

**For example:**

**The mean value of the first column is  $(1 + 5) / 2 = 3$ . The mean value of the second column is  $(4 + 10) / 2 = 7$ .**

**And so on.**

**We can also use `axis=1` to find the mean of each row in the matrix:**

```
#find mean of each row in matrix  
np.mean(my_matrix, axis=1)
```

```
matrix(  
])
```

**The output shows the mean value of each row in the matrix.**

**The mean value in the first row is  $(1+4+7+8) / 4 = 5$ . The mean value in the second row is  $(5+10+12+14) / 4 = 10.25$ .**

## Example 2: Find Sum Along Different Axes

We can use `axis=0` to find the sum of each column in the matrix:

```
#find sum of each column in matrix
```

```
np.sum(my_matrix, axis=0)
```

```
matrix([
```

The output shows the sum of each column in the matrix.

For example:

The sum of the first column is  $1 + 5 = 6$ . The sum of the second column is  $4 + 10 = 14$ .

And so on.

We can also use `axis=1` to find the sum of each row in the matrix:

```
#find sum of each row in matrix
```

```
np.sum(my_matrix, axis=1)
```

```
matrix(,
```

])

**The output shows the sum of each row in the matrix.**

**For example:**

**The sum of the first row is  $1+4+7+8 = 20$ . The sum of the second row is  $5+10+12+14 = 41$ .**

**Additional Resources**

**The following tutorials explain how to perform other common operations in NumPy:**