

# CLASS 2 : NUMPY

## 1. Vectorization in NumPy

Vectorization in NumPy refers to performing operations on entire arrays or vectors at once instead of using explicit loops. NumPy internally implements these operations in optimized low-level code, which significantly improves execution speed and efficiency.

### Key Benefits:

- Faster computations compared to traditional loops
- Cleaner and more readable code
- Efficient use of CPU and memory

Vectorization is especially important in data science and artificial intelligence, where large numerical datasets are processed frequently.

## 2. Difference Between `reshape()` and `resize()`

Both `reshape()` and `resize()` are used to change the shape of NumPy arrays, but they behave differently.

- `reshape()` returns a new view of the array with the desired shape without changing the original data. It does not modify the size of the array.
- `resize()` changes the shape and size of the original array itself. If the new size is larger, it fills additional entries with repeated values; if smaller, it truncates the data.

### Key Difference:

`reshape()` is safer and non-destructive, while `resize()` permanently alters the array.

## 3. NumPy Dimensions and Axes

The **dimension** of a NumPy array refers to the number of directions or levels needed to access its elements (also called the array's rank).

- A 1-D array has one dimension
- A 2-D array has rows and columns

- A 3-D array extends to multiple layers

An **axis** specifies the direction along which operations are performed:

- Axis 0 typically represents rows
- Axis 1 typically represents columns

Understanding axes is essential for applying aggregation functions like sum, mean, and maximum correctly.

## 4. Slicing in NumPy Arrays

Slicing is the process of extracting a portion of an array by specifying a range of indices. It allows selective access to elements, rows, columns, or sub-arrays without copying data unnecessarily.

### Advantages of Slicing:

- Efficient data access
- Enables quick data manipulation
- Reduces memory overhead

Slicing is widely used in preprocessing and feature selection tasks in machine learning.

## 5. Role of NumPy in Mathematical Computations for AI

NumPy provides a powerful foundation for numerical and mathematical computations required in artificial intelligence. It supports fast array operations, linear algebra functions, statistical calculations, and random number generation.

### Importance in AI:

- Enables efficient handling of large datasets
- Supports matrix operations used in neural networks
- Forms the backbone of AI libraries like TensorFlow, PyTorch, and Scikit-learn

NumPy's performance and mathematical capabilities make it an essential tool for AI model development and data analysis.