

Basic Concepts and Array Creation

NumPy (Numerical Python) is the fundamental package for scientific computing in Python, providing powerful N-dimensional array objects.

- Import: `import numpy as np`
- Creation from List: `a = np.array([1, 2, 3])`
- Special Arrays:
 - `np.zeros((3, 4))`: Array filled with zeros.
 - `np.ones((2, 3))`: Array filled with ones.
 - `np.full((2, 2), 7)`: Array filled with a specific value.
 - `np.eye(3)`: 3x3 Identity matrix.
 - `np.arange(0, 10, 2)`: Sequence [0, 2, 4, 6, 8].
 - `np.linspace(0, 1, 5)`: 5 evenly spaced points between 0 and 1.
- Random Arrays:
 - `np.random.rand(3, 2)`: Random values in [0, 1).
 - `np.random.randn(3, 3)`: Samples from standard normal distribution.
 - `np.random.randint(0, 10, (2, 2))`: Random integers.

Array Properties

- `a.shape`: Tuple representing dimensions (rows, cols).
- `a.ndim`: Number of dimensions.
- `a.dtype`: Data type of elements (e.g., `int64`, `float64`).
- `a.size`: Total number of elements.
- `a.astype(np.float32)`: Cast array to a different type.

Selection and Indexing

- Slicing: `a[start:stop:step]` (e.g., `a[1:3]`).
- 2D Selection: `a[row, col]` (e.g., `a[0, 1]`, `a[:, 1]` for all rows in 2nd column).
- Boolean Indexing: `a[a > 5]` (Select elements satisfying a condition).
- Fancy Indexing: `a[[0, 2, 4]]` (Select elements at specific indices).

Array Reshaping and Manipulation

- `a.reshape(3, 2)`: Change shape without changing data.
- `a.flatten()` / `a.ravel()`: Convert to 1D array.
- `a.T` or `a.transpose()`: Transpose matrix.
- `np.concatenate([a, b], axis=0)`: Join arrays.
- `np.vstack([a, b])`, `np.hstack([a, b])`: Vertical/Horizontal stack.
- `np.split(a, 3)`: Split array into sub-arrays.

Mathematical Operations

- Element-wise: `+`, `-`, `*`, `/`, `**`, `np.sqrt(a)`, `np.exp(a)`, `np.sin(a)`.
- Matrix Multiplication: `a @ b` or `np.dot(a, b)`.
- Statistics:
 - `np.sum(a)`, `np.mean(a)`, `np.median(a)`.
 - `np.std(a)`, `np.var(a)`.
 - `np.min(a)`, `np.max(a)`.
 - `np.argmax(a)`, `np.argmin(a)` (Indices of min/max).
 - Sum along axis: `a.sum(axis=0)` (columns), `a.sum(axis=1)` (rows).

Broadcasting

The mechanism that allows NumPy to work with arrays of different shapes during arithmetic operations (e.g., adding a scalar to a vector).

Sorting

- `np.sort(a)`: Return a sorted copy.
- `a.sort()`: Sort in-place.
- `np.argsort(a)`: Return indices that would sort the array.

Universal Functions (ufuncs)

Functions that operate element-wise on whole arrays, significantly faster than Python loops.

- `np.add`, `np.subtract`, `np.multiply`, `np.divide`.
- `np.unique(a)`: Find unique elements.