Introduction to NumPy

Shahkar Ahmad Nahvi

July 11, 2026

Outline

What is NumPy?

- Numerical Python library for scientific computing.
- Provides support for large, multi-dimensional arrays and matrices.
- Includes mathematical functions, linear algebra, Fourier transforms, random numbers.
- Essential for data science, machine learning, and numerical analysis.

Key Features of NumPy

- **ndarray**: Efficient N-dimensional array object.
- Vectorized operations (faster than Python lists).
- Broadcasting for operations on different-shaped arrays.
- Integration with C/C++ and Fortran code.
- Basis for libraries like Pandas, SciPy, scikit-learn.

NumPy Arrays vs Python Lists

- **Memory Efficiency:** Arrays use less memory than lists.
- **Speed:** Arrays are implemented in C, so operations are much faster.
- **Vectorization:** Arrays support element-wise operations without loops.
- **Homogeneity:** Arrays store elements of the same data type, while lists can store mixed types.
- **Convenience:** Built-in mathematical functions for arrays, unlike lists.

Data Types in Python vs NumPy

- **Python Lists:**
  - Can hold elements of different data types (heterogeneous).
  - Slower for numerical computations.
- **NumPy Arrays:**
  - Require a single data type (homogeneous).
  - Support a wide range of numeric types (`int8`, `int32`, `float64`, `complex128`, etc.).
  - Enable faster operations and efficient storage.
- Example: `np.array([1, 2, 3], dtype=np.float32)`

Creating Arrays

```
import numpy as np

# Create a 1D array
a = np.array([1, 2, 3, 4])

# Create a 2D array
b = np.array([[1, 2], [3, 4]])

# Special arrays
zeros = np.zeros((3,3))
ones  = np.ones((2,2))
range_arr = np.arange(0,10,2)
```

Array Operations

```
x = np.array([1, 2, 3])
y = np.array([4, 5, 6])

# Element-wise addition
print(x + y)   # [5 7 9]

# Element-wise multiplication
print(x * y)   # [ 4 10 18]

# Dot product
print(np.dot(x, y))  # 32
```

Applications of NumPy

- Data analysis and manipulation.
- Image and signal processing.
- Machine learning (as backbone of libraries).
- Numerical simulations and scientific computing.

Conclusion

- NumPy is the foundation of numerical computing in Python.
- Provides fast, efficient array operations.
- Arrays outperform Python lists in memory, speed, and functionality.
- Widely used in academia and industry.