



DAY 09 — NumPy Advanced

Goal : Write powerful array logic with minimal code

1 Broadcasting (Most Important Concept)

What is Broadcasting?

Broadcasting lets NumPy **perform operations on arrays of different shapes**.

Simple Example:

```
import numpy as np

arr = np.array([1,2,3])
print(arr + 10)

# output
[11 12 13]
```

👉 NumPy “stretches” 10 to match the array.

2D Broadcasting Example:

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

vector = np.array([10,20,30])

print(matrix + vector)
```

Each row gets the vector added.

2 Boolean Masking (Filtering Data)

Why this matters

Used in:

- Data cleaning
- Removing outliers

- Feature selection

Example:

```
arr = np.array([10, 20, 5, 30, 15])  
  
mask = arr > 15  
print(mask)  
  
# output  
[False True False True False]
```

Apply mask:

```
filtered = arr[arr > 15]  
print(filtered)
```

3 Axis Explained (Very Important)

Think of axis like this:

- `axis=0` → down(columns)
- `axis=1` → across(rows)

Example:

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

Column-wise sum

```
data.sum(axis=0) # [5,7,9]
```

Row-wise sum

```
data.sum(axis=1) # [6 15]
```

4 `np.where` (Conditional Logic)

```
arr = np.array([10, 20, 30])  
  
result = np.where(arr > 15, 1, 0)  
print(result)  
  
# [0 1 1]
```

Used to create labels in ML.

5 Fancy Indexing

```
arr = np.array([10, 20, 30, 40])  
  
indices = [0, 2]  
print(arr[indices])  
  
  
# output  
[10 30]
```

6 Stacking Arrays

```
a = np.array([1,2])  
b = np.array([3,4])  
  
np.vstack((a, b))  
np.hstack((a, b))  
  
print(np.vstack((a, b)))  
print(np.hstack((a, b)))  
  
# output  
[[1 2],  
 [3 4]]  
[1 2 3 4]
```

7 Real ML Example

Normalize data:

```
X = np.array([10, 20, 30])  
  
X_norm = (X - X.mean()) / X.std()  
print(X_norm)  
  
# output  
[-1.22474487  0.          1.22474487]
```

This uses:

- ✓ broadcasting
 - ✓ aggregation
 - ✓ vectorization
-