Cheat Sheet

NumPy

Python Numerical Computing Library

VISHNU VARDHAN

A guide to essential NumPy functions for numerical computing, data manipulation, and scientific computing in Python.

VISHNU VARDHAN

Array Creation

Create Arrays

```
import numpy as np

# Basic arrays
arr1 = np.array([1, 2, 3, 4, 5])
arr2 = np.array([[1, 2], [3, 4]])
```

Special Arrays

```
zeros = np.zeros((3, 4))
ones = np.ones((2, 3))
identity = np.eye(3)
empty = np.empty((2, 3))
```

Sequence Arrays

```
range_arr = np.arange(10)
spaced = np.linspace(0, 1, 5)
log_space = np.logspace(0, 2, 5)
```

Random Arrays

```
random = np.random.random((2, 3))
normal = np.random.normal(0, 1, (2, 3))
int_random = np.random.randint(0, 10, (2, 3))
```

Array Properties & Indexing

Array Properties

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

shape = arr.shape
ndim = arr.ndim
size = arr.size
dtype = arr.dtype
```

Basic Indexing

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

element = arr[1, 2]
row = arr[0]
col = arr[:, 1]
```

Boolean Indexing

```
arr = np.arange(10)

mask = arr > 5
filtered = arr[mask]
filtered2 = arr[(arr > 5) & (arr < 8)]
```

VISHNU VARDHAN

Array Operations

Math Functions

```
a     =     np.array([0,     np.pi/2,     np.pi])

sin_values = np.sin(a)
exp_values = np.exp(a)
log_values = np.log(np.array([1, np.e, np.e**2]))
sqrt_values = np.sqrt(np.array([1, 4, 9]))
```

Comparison Operations

```
a  =  np.array([1,  2,  3])
b = np.array([3, 2, 1])
equal = a := b
greater = a > b
any_true = np.any(a > 2)
all_true = np.all(a < 4)
```

Aggregation

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

sum_all = np.sum(arr)
min_val = np.min(arr)
max_val = np.max(arr)
mean_val = np.mean(arr)
```

VISHNU VARDHAN

Array Manipulation

Reshaping Arrays

```
arr = np.arange(12)

reshaped = arr.reshape(3, 4)
flattened = reshaped.flatten()
raveled = reshaped.ravel()
```

Joining Arrays

```
concat = np.concatenate((a, b))
vstack = np.vstack((a, b))
hstack = np.hstack((a.reshape(-1, 1), b.reshape(-1, 1)))
```

Splitting Arrays

```
arr = np.arange(12).reshape(3, 4)

split = np.split(arr, 3)
hsplit = np.hsplit(arr, 2)
vsplit = np.vsplit(arr, 3)
```

Transposing & Axes

```
arr = np.arange(12).reshape(3, 4)

transposed = arr.T
swapped = np.swapaxes(arr, 0, 1)
```

VISHNU VARDHAN

Broadcasting

Broadcasting Basics

```
arr = np.ones((3, 4))

arr_plus_5 = arr + 5
arr_times_vec = arr * np.arange(4)
```

Broadcasting Example

```
a = np.arange(3).reshape(3, 1)
b = np.arange(4)

c = a + b
```

Normalizing Data

```
data = np.random.random((5, 3))

means = data.mean(axis=0)
normalized = data - means
```

Outer Product

```
a = np.array([1, 2, 3])
b = np.array([4, 5])

outer = np.outer(a, b)
```

VISHNU VARDHAN

Linear Algebra

Matrix Operations

```
a = np.array([[1, 2], [3, 4]])
b = np.array([[5, 6], [7, 8]])

dot = np.dot(a, b)
matmul = a @ b
```

Matrix Decomposition

```
a = np.array([[1, 2], [2, 1]])

eigenvalues, eigenvectors = np.linalg.eig(a)
u, s, vh = np.linalg.svd(a)
```

Solving Linear Systems

```
A = np.array([[3, 1], [1, 2]])
b = np.array([9, 8])

x = np.linalg.solve(A, b)
```

Matrix Properties

```
a = np.array([[1, 2], [3, 4]])

det = np.linalg.det(a)
rank = np.linalg.matrix_rank(a)
trace = np.trace(a)
```

VISHNU VARDHAN

Statistics

Basic Statistics

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

mean = np.mean(arr)
median = np.median(arr)
std = np.std(arr)
var = np.var(arr)
```

Percentiles

```
arr = np.random.normal(0, 1, 1000)

percentile_50 = np.percentile(arr, 50)
percentiles = np.percentile(arr, [25, 50, 75])
```

Histogram

```
arr = np.random.normal(0, 1, 1000)

hist, bins = np.histogram(arr, bins=10)
```

Correlation

```
x = np.random.normal(0, 1, 100)
y = x + np.random.normal(0, 0.5, 100)
corr_matrix = np.corrcoef(x, y)
```

VISHNU VARDHAN

Advanced Indexing

Boolean Masks

```
arr = np.arange(12).reshape(3, 4)

mask = arr > 5
filtered = arr[mask]
arr[arr % 2 := 0] = -1
```

Fancy Indexing

```
arr = np.arange(10, 20)
indices = [2, 4, 6, 8]

selected = arr[indices]
```

Where Function

```
x = np.arange(10) condition = x < 5

result = np.where(condition, x, 10*x)
```

Masked Arrays

```
from        numpy        import        ma
arr = np.arange(10) mask = arr % 3 := 0

masked_arr = ma.masked_array(arr, mask)
```

VISHNU VARDHAN

File I/O

Binary Files

```
arr = np.arange(10)

np.save('array.npy', arr)
loaded = np.load('array.npy')
```

Multiple Arrays

```
x = np.arange(10)
y = np.ones(5)

np.savez('arrays.npz', x=x, y=y)
loaded = np.load('arrays.npz')
```

Text Files

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

np.savetxt('array.txt', arr)
np.savetxt('array.csv', arr, delimiter=',')
```

Loading Text

```
loaded = np.loadtxt('array.txt')
csv_loaded = np.loadtxt('array.csv', delimiter=',')
```

Advanced Features

Structured Arrays

```
dt = np.dtype([('name', 'U10'), ('age', 'i4')])
s = np.array([('Ann', 25), ('Bob', 32)], dtype=dt)

names = s['name']
ages = s['age']
```

Polynomials

```
from numpy.polynomial import polynomial as poly

p = poly.Polynomial([1, 2, 3])
val = p(2)
```

FFT

```
x = np.linspace(0, 10, 100)
y = np.sin(x)

fft = np.fft.fft(y)
freqs = np.fft.fftfreq(len(y), d=x[1]-x[0])
```

FOLLOW

VISHNU VARDHAN

VISHNU VARDHAN