μ Learn

Numpy

- Basically Matrix Manipulatio

for creating ndarray

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

print (a.shape) # print shape of array, here

p

[2, 3]

rows columns.

shape

→ np.zeros(2,2)

↳ Creates ~~of~~ an array of all zeros in (2x2) shape

→ np.ones((1,2))

↳ Array of all ones → [[1, 1.]]

→ np.full((2,2),7)

Constant ~~arr~~ 2x2 array with 7 as the only mem

→ np.eye(2)

2x2 identity matrix ⇒ $\begin{bmatrix}1 ; 0\\ 0 ; 1\end{bmatrix}$

Boolean Indexing.

→ a = np.array([[1,2],[3,4],[5,6]])

→ bool_idx = (a>2) #condition checking for array

→ print (bool_idx) → "False False

True True

True True"

arr → General for array.

→ np.add(array, array) ≈ array + array

→ print (x+y)

→ np.subtract(arr, arr) ≃ arr − arr

→ np.multiply(arr, arr) ≈ arr * arr # Element wise only

→ np.divide(arr, arr) ≈ arr / arr # Element wise only.

→ ## For Matrix Multiplication dot() is used

→ 1° arr. dot (2° array) ≈ np.dot(v, w)

1° arr → v, 2° arr → w

For row & column specific sum.

↳ np.sum(arr, axis=0) // sum of each column

→ np.sum(arr, axis=1) // sum of each row

Transposing a Matrix

⇒ arr.T // will give transpose