numpy intro

ndarray → support for math operations

scalars in numpy can be uint8, int8, uint16, ...

example:

- s = np.array(5) → only holds a scalar
- s.shape → () → 0-dim
- x = s+3

- v = np.array([1,2,3])
- v.shape → (3,)
- v[1] → 2, v[1:] → [2,3]

- m = np.array([[1,2,3],[4,5,6],[7,8,9]])
- m.shape → (3,3)

reshaping:

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

X = v.reshape(1,4) (1,4)

element-wise matrix operations :

$$2 + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 2+1 & 2+2 \\ 2+3 & 2+4 \end{bmatrix}$$

`x = np.multiply(some_array, 5)` == `x = some_array*5`

`m *= 0` → no matter the dims of m, all of the elements become Zero.

`m * m` → element-wise multiplication

Matrix product :

`np.matmul(A,B)`

$AB \neq BA$

Matrix transpose :

`np.transpose(a)` or `a.T`