→ NumPy Library

import numpy as np

- reduce computation time

array → x = np.array([ ]) must be sametype

type n[redacted] dimensional array

dtype elements type & size

shape returns tuple (row size, column size)

ndim returns array dimensions

store array in a file

np.save('my-array', x)

load file

np.load('___')

file → my-array.npy

→ Built-in Functions

np.zeros(tuple) creates an array of zeros

shape-array

np.ones(tuple) " " " " ones

" " " " with the same value

np.save('my-array', x) np.load('___')

Built-in Functions

np.zeros(tuple) creates an array of zeros
shape_array

np.ones(tuple) " " " " ones

np.full(tuple, constant to fill) full an array with the same value

np.eye(int) creates a matrix with 1s as a diagonal

np.diag([, , ,]) fills a matrix diagonal with specific values

np.arrange(start, stop, step) 1-D array from start to stop → 1

np.linspace(start, stop, n) 1-D array from start to stop evenly spaced by (n)

if you want exclusive [redacted] stop → end point → false

np.reshape(array, shape) converts 1D → 2D

must have same size

np.diag([, , ,]) fills a matrix diagonal with specific values

np.arrange (start , stop , step) 1-D array from start to stop → 1

np.linspace (start , stop , n) 1-D array from start to stop evenly spaced by (n)

if you want exclusive [redacted] stop → end point → false

np.reshape (array , shape) converts 1D → 2D

must have [redacted] same size

np.random.random ((shape ,)) random numbers (floats)

np.random.randint (from, to , (shape ,)) random integers

np.random.normal (specific mean , specific STD , shape)

→ Access, Delete, Insert

index can +ve or -ve

np.delete (array , elements to be deleted)

np.delete (array , elements to be deleted , axis)

1 → rows , 0 → columns (axis)

np.append (array , elements)

np.append (array , element , axis)

np.insert (array , index , elements , axis)

→ Copy

x.copy ()

np.vstack must be similar

np.hstack

np.hstack

→ Slicing creates a view of the original array

x [start : end]
Inclusive exclusive

x [s : e , s : e]
rows columns

→ Boolean indexing

x [(boolean expression)]

np.sort(x) function doesn't affect original array

|— x.sort() method affects " "

→ Arithmetic Operation mean median max min

→ Broadcasting adjusting sizes