# 22649011-ngohongthong-xla-lab3

August 19, 2024

Ngô Hồng Thông

[2]:
```
import numpy as np
```

[3]:
```
# Create a 1D NumPy array
list_of_values = [1, 2, 3]
x = np.array(list_of_values)
x
```

[3]:
```
array([1, 2, 3])
```

[4]:
```
more_values = [[[4], [5], [6]]]
y = np.array(more_values)
y
```

[4]:
```
array([[[4],
        [5],
        [6]]])
```

[5]:
```
# tạo mảng 0 với 2 chiều
zeros = np.zeros((20, 2), dtype=np.int32)
zeros
```

[5]:
```
array([[0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
       [0, 0],
```

```
        [0, 0],
        [0, 0],
        [0, 0],
        [0, 0],
        [0, 0]])
```

[6]:
```
#
one = np.ones((5, 2, 1), dtype=np.float64)
one
```

[6]:
```
array([[[1.],
        [1.]],

       [[1.],
        [1.]],

       [[1.],
        [1.]],

       [[1.],
        [1.]],

       [[1.],
        [1.]]])
```

[7]:
```
x = np.array([5, 5])
zeros = np.zeros_like(x)
zeros
```

[7]:
```
array([0, 0])
```

[8]:
```
ones = np.ones_like(x, dtype=np.float32)
ones
```

[8]:
```
array([1., 1.], dtype=float32)
```

[9]:
```
x = np.arange(5)
print(x)

y = np.arange(5, 10)
print(y)

z = np.arange(5, 10, 2)
print(z)
```

```
[0 1 2 3 4]
[5 6 7 8 9]
```

```
[5 7 9]
```

[10]:
```
np.eye(5)
```

[10]:
```
array([[1., 0., 0., 0., 0.],
       [0., 1., 0., 0., 0.],
       [0., 0., 1., 0., 0.],
       [0., 0., 0., 1., 0.],
       [0., 0., 0., 0., 1.]])
```

[11]:
```
x = np.random.random((2, 2))
x
```

[11]:
```
array([[0.51252411, 0.17530716],
       [0.01225259, 0.79547777]])
```

[12]:
```
x = np.full((2, 2), 7)
x
```

[12]:
```
array([[7, 7],
       [7, 7]])
```

[13]:
```
x = np.ones((2, 2)) * 7
x
```

[13]:
```
array([[7., 7.],
       [7., 7.]])
```

[14]:
```
x = np.array([1, 2, 3, 4])
y = np.array([5, 6, 7, 8], dtype=np.float32)
print(x.dtype)
print(y.shape)
```

```
int32
(4,)
```

[15]:
```
x = np.array([1, 2, 3, 4])
y = np.ones((2, 4, 1, 2, 3))
print(x.shape)
print(y.shape)
```

```
(4,)
(2, 4, 1, 2, 3)
```

[17]:
```
x = np.full((2, 2, 3), 7)
print(x)

x = np.expand_dims(x, 0)
```

```
print(x)
assert x.shape == (1, 2, 2, 3)
print(x)

x = np.expand_dims(x, -1)
print(x)
assert x.shape == (1, 2, 2, 3, 1)
print(x)
```

```
[[[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]]
[[[[7 7 7]
   [7 7 7]]

  [[7 7 7]
   [7 7 7]]]]
[[[[7 7 7]
   [7 7 7]]

  [[7 7 7]
   [7 7 7]]]]
[[[[[7]
    [7]
    [7]]

   [[7]
    [7]
    [7]]]


  [[[7]
    [7]
    [7]]

   [[7]
    [7]
    [7]]]]]
[[[[[7]
    [7]
    [7]]

   [[7]
    [7]
    [7]]]
```

```
[[[7]
  [7]
  [7]]

 [[7]
  [7]
  [7]]]]
```

[22]:
```
x = np.arange((10))
print(x)
y = np.reshape(x, (2, 5))
print(y)
assert y.shape == (2, 5)
```

```
[0 1 2 3 4 5 6 7 8 9]
[[0 1 2 3 4]
 [5 6 7 8 9]]
```

[29]:
```
x = np.arange(20)
element = x[10]
element
```

[29]: 10

[28]:
```
x = np.arange(20)
elements = x.item(10)
elements
```

[28]: 10

[31]:
```
x = np.arange(20)
elements = x[10:15]
elements
```

[31]: array([10, 11, 12, 13, 14])

[34]:
```
more_elements = x[10:-7]
more_elements
```

[34]: array([10, 11, 12])

[35]:
```
x = x[::3]
x
```

[35]: array([ 0,  3,  6,  9, 12, 15, 18])

[40]:
```
x = np.full((5, 2), 3)
y = np.full((5, 1), 4)
z = np.concatenate([x, y], axis=1)
print(z)
assert z.shape == (5, 3)
print('--------------')
print(z)
```

```
[[3 3 4]
 [3 3 4]
 [3 3 4]
 [3 3 4]
 [3 3 4]]
--------------
[[3 3 4]
 [3 3 4]
 [3 3 4]
 [3 3 4]
 [3 3 4]]
```

[41]:
```
x = np.full((4, 2, 3), 8)
print(x)
y = np.ones_like(x)
print(y)
```

```
[[[8 8 8]
  [8 8 8]]

 [[8 8 8]
  [8 8 8]]

 [[8 8 8]
  [8 8 8]]

 [[8 8 8]
  [8 8 8]]]
[[[1 1 1]
  [1 1 1]]

 [[1 1 1]
  [1 1 1]]

 [[1 1 1]
  [1 1 1]]

 [[1 1 1]
  [1 1 1]]]
```

[45]:
```
array_sub = x - y
print(array_sub)
print("-----------------")
assert np.array_equal(array_sub, np.ones_like(x) * 7)
print(array_sub)
```

```
[[[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]]
-----------------
[[[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]]
```

[48]:
```
x = np.full((4, 2, 3), 8)
y = 1

array_sum = x + y
print(array_sum)
print("-----------------")
assert np.array_equal(array_sum, np.ones_like(x) * 9)
print(array_sum)
print("-----------------")

array_sub = x - y
assert np.array_equal(array_sub, np.ones_like(x) * 7)
print(array_sub)
```

```
[[[8 8 8]
  [8 8 8]]

 [[8 8 8]
```

```
  [8 8 8]]

 [[8 8 8]
  [8 8 8]]

 [[8 8 8]
  [8 8 8]]]
[[[9 9 9]
  [9 9 9]]

 [[9 9 9]
  [9 9 9]]

 [[9 9 9]
  [9 9 9]]

 [[9 9 9]
  [9 9 9]]]
------------------
[[[9 9 9]
  [9 9 9]]

 [[9 9 9]
  [9 9 9]]

 [[9 9 9]
  [9 9 9]]

 [[9 9 9]
  [9 9 9]]]
------------------
[[[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]

 [[7 7 7]
  [7 7 7]]]
```

[50]:
```
x = np.full((4, 2, 3), 8)
y = np.ones_like(x) * 2
print(y)
print("------------------")
```

```
mul = x * y
assert np.array_equal(mul, np.ones_like(x) * 16)
print(mul)
print("-----------------")
mul2 = np.multiply(x, y)
assert np.array_equal(mul2, np.ones_like(x) * 16)
print(mul2)
```

```
[[[2 2 2]
  [2 2 2]]

 [[2 2 2]
  [2 2 2]]

 [[2 2 2]
  [2 2 2]]

 [[2 2 2]
  [2 2 2]]]
-----------------
[[[16 16 16]
  [16 16 16]]

 [[16 16 16]
  [16 16 16]]

 [[16 16 16]
  [16 16 16]]

 [[16 16 16]
  [16 16 16]]]
-----------------
[[[16 16 16]
  [16 16 16]]

 [[16 16 16]
  [16 16 16]]

 [[16 16 16]
  [16 16 16]]

 [[16 16 16]
  [16 16 16]]]
```

[53]:
```
x1 = np.full((4, 2, 3), 8)
x2 = np.full((3, 3), 7)
y = np.eye(3)
```

```
print(y)
print("-----------------")

mul = np.matmul(x1, y)
assert np.array_equal(mul, x1)

print(mul)
print("-----------------")

mul2 = np.matmul(x2, y)
assert np.array_equal(mul2, x2)
print(mul2)
```

```
[[1. 0. 0.]
 [0. 1. 0.]
 [0. 0. 1.]]
-----------------
[[[8. 8. 8.]
  [8. 8. 8.]]

 [[8. 8. 8.]
  [8. 8. 8.]]

 [[8. 8. 8.]
  [8. 8. 8.]]

 [[8. 8. 8.]
  [8. 8. 8.]]]
-----------------
[[7. 7. 7.]
 [7. 7. 7.]
 [7. 7. 7.]]
```

[54]:
```
x = np.arange(5)
y = np.where(x < 2, 0, 255)
print(y)
```

```
[  0   0 255 255 255]
```

[56]:
```
data = np.loadtxt('data.txt', dtype=np.float32,delimiter=',')
data
```

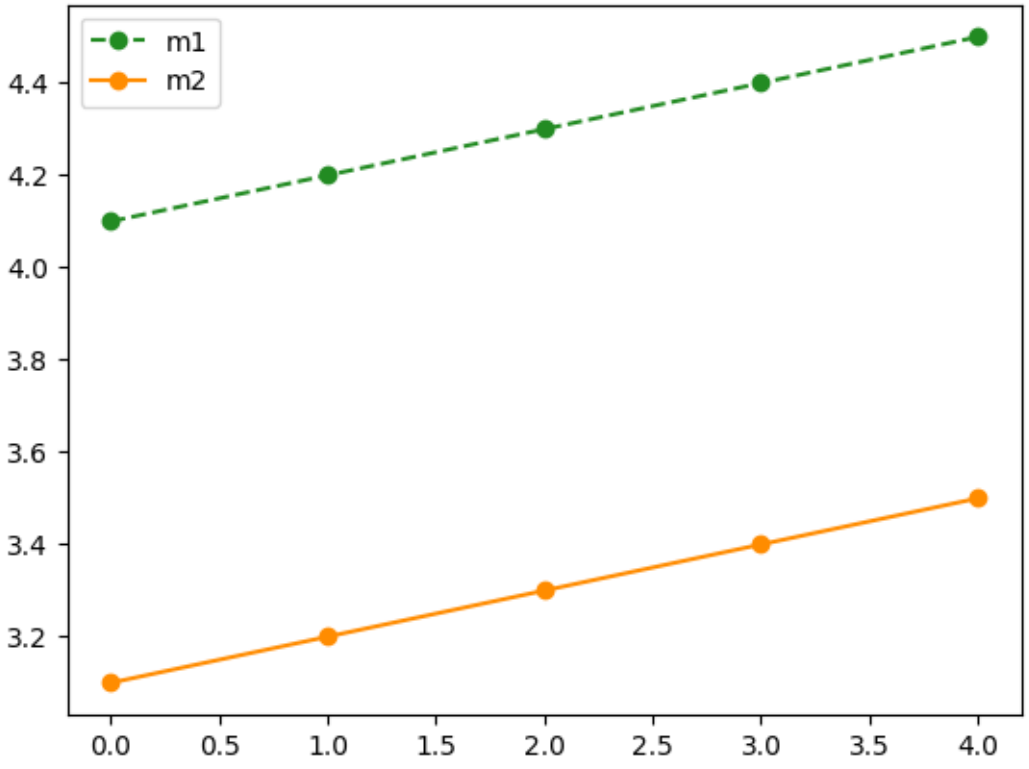
[56]:
```
array([[ 1.,  2.,  3.,  4.,  5.,  6.],
       [ 7.,  8.,  9., 10., 11., 12.]], dtype=float32)
```

[58]:
```
import matplotlib.pyplot as plt
```

```
labels = [7, 8, 9, 10, 11]
x = np.arange(len(labels))
m1 = [4.1, 4.2, 4.3, 4.4, 4.5]
m2 = [3.1, 3.2, 3.3, 3.4, 3.5]

plt.plot(x, m1, color='forestgreen', label='m1', linestyle='dashed', marker='o')
plt.plot(x, m2, color='darkorange', label='m2', linestyle='-', marker='o')

plt.legend()
plt.savefig("chart.png")
```




[ ]: