

Lesson 07: numpy 3 ~ 5

2023-01-30

ref

[https://guebin.github.io/IP2022/2022/04/11/\(6\)-4-11.html](https://guebin.github.io/IP2022/2022/04/11/(6)-4-11.html)

[https://guebin.github.io/IP2022/2022/04/18/\(7\)-4-18.html#matplotlib](https://guebin.github.io/IP2022/2022/04/18/(7)-4-18.html#matplotlib)

imports

```
import numpy as np
```

3

- : numpy .

2 1

-

$$\begin{cases} y + z + w = 3 \\ x + z + w = 3 \\ x + y + w = 3 \\ x + y + z = 3 \end{cases}$$

- ?

$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \\ 3 \\ 3 \end{bmatrix}$$

-

```
A = np.array([[0,1,1,1],[1,0,1,1],[1,1,0,1],[1,1,1,0]])  
A
```

```
array([[0, 1, 1, 1],  
       [1, 0, 1, 1],  
       [1, 1, 0, 1],  
       [1, 1, 1, 0]])
```

```
b= np.array([3,3,3,3]).reshape(4,1)  
b
```

```
array([[3],  
       [3],  
       [3],  
       [3]])
```

```
np.linalg.inv(A) @ b
```

```
array([[1.],  
       [1.],  
       [1.],  
       [1.]])
```

-

b .

```
b=np.array([3,3,3,3])  
b
```

```
array([3, 3, 3, 3])
```

```
b.shape # b.shape      1      .
```

```
(4,)
```

```
np.linalg.inv(A) @ b
```

```
array([1., 1., 1., 1.])
```

@

- . - (2,2) @ (2,1) => (2,1) - (1,2) @ (2,2) =>
(1,2)

```
A = np.array([1,2,3,4]).reshape(2,2)
b = np.array([1,2]).reshape(2,1)
A@b
```

```
array([[ 5],
       [11]])
```

```
A.shape, b.shape, (A@b).shape
```

```
((2, 2), (2, 1), (2, 1))
```

```
A = np.array([1,2,3,4]).reshape(2,2)
b = np.array([1,2]).reshape(1,2)
b@A
```

```
array([[ 7, 10]])
```

```
A.shape, b.shape, (b@A).shape
```

```
((2, 2), (1, 2), (1, 2))
```

-

```
A = np.array([1,2,3,4]).reshape(2,2)
b = np.array([1,2]).reshape(2,1)
b@A
```

ValueError: matmul: Input operand 1 has a mismatch in its core dimension 0, with gufunc signature

```
A = np.array([1,2,3,4]).reshape(2,2)
b = np.array([1,2]).reshape(1,2)
A@b
```

ValueError: matmul: Input operand 1 has a mismatch in its core dimension 0, with gufunc signature

- ? ? \rightarrow ! - $(2,) @ (2,2) = (2,) - (2,2) @ (2,) =$
 $(2,)$

```
A = np.array([1,2,3,4]).reshape(2,2)
b = np.array([1,2])
A@b
```

```
array([ 5, 11])
```

```
A.shape, b.shape, (A@b).shape
```

```
((2, 2), (2,), (2,))
```

- b $(2,1)$ $(2,)$.

```
b@A
```

```
array([ 7, 10])
```

```
A.shape, b.shape, (b@A).shape
```

```
((2, 2), (2,), (2,))
```

- $b \quad (1,2) \quad (2,) \quad .$

- ?

```
b1 = np.array([1,2,3,4])
```

```
b2 = np.array([1,2,3,4])
```

```
b1@b2
```

30

```
b1.shape, b2.shape, (b1@b2).shape
```

```
((4,), (4,), ())
```

- $(1,4) @ (4,1) = (1,1)$

- .

```
b1 = np.array([1,2,3,4]).reshape(1,4)
```

```
b2 = np.array([1,2,3,4]).reshape(4,1)
```

```
b1@b2
```

```
array([[30]])
```

```
b1.shape, b2.shape, (b1@b2).shape
```

```
((1, 4), (4, 1), (1, 1))
```

- $(4,1) @ (1,4)$

```
b1 = np.array([1,2,3,4]).reshape(4,1)
b2 = np.array([1,2,3,4]).reshape(1,4)
b1@b2
```

```
array([[ 1,  2,  3,  4],
       [ 2,  4,  6,  8],
       [ 3,  6,  9, 12],
       [ 4,  8, 12, 16]])
```

- .shape

- .

```
a=np.array(3.14) # , 0d array
a, a.shape
```

```
(array(3.14), ())
```

```
a=np.array([3.14]) # , 1d array
a, a.shape
```

```
(array([3.14]), (1,))
```

```
a=np.array([[3.14]]) # , 2d array
a, a.shape
```

```
(array([[3.14]]), (1, 1))
```

```
a=np.array([[[3.14]]]) # , 3d array
a, a.shape
```

```
(array([[[3.14]]]), (1, 1, 1))
```

4

- : axis .

np.concatenate

-

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

```
np.concatenate([a,b])
```

```
array([ 1,  2, -1, -2])
```

-

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

```
np.concatenate([a,b,c])
```

```
array([ 1,  2, -1, -2,  3,  4,  5])
```

-
- $a+b+c$?

- 2d array .

```
a=np.arange(4).reshape(2,2)  
b=-a
```

```
np.concatenate([a,b])
```

```
array([[ 0,  1],
       [ 2,  3],
       [ 0, -1],
       [-2, -3]])
```

- ?

```
np.concatenate([a,b],axis=1)
```

```
array([[ 0,  1,  0, -1],
       [ 2,  3, -2, -3]])
```

- axis=1 ? axis=0,2 ?

```
np.concatenate([a,b],axis=0)
```

```
array([[ 0,  1],
       [ 2,  3],
       [ 0, -1],
       [-2, -3]])
```

- np.concatenate([a,b]) .
- np.concatenate([a,b]) np.concatenate([a,b],axis=0)
?

```
np.concatenate([a,b],axis=2)
```

AxisError: axis 2 is out of bounds for array of dimension 2

- .

- axis .

```
a=np.array(range(2*3*4)).reshape(2,3,4)
a
```



```
array([[[ 0,  1,  2,  3],
        [ 4,  5,  6,  7],
        [ 8,  9, 10, 11]],

      [[12, 13, 14, 15],
        [16, 17, 18, 19],
        [20, 21, 22, 23]]])
```

```
b=-a
b
```

```
array([[[ 0, -1, -2, -3],
        [-4, -5, -6, -7],
        [-8, -9, -10, -11]],

      [[-12, -13, -14, -15],
        [-16, -17, -18, -19],
        [-20, -21, -22, -23]]])
```

```
np.concatenate([a,b],axis=0)
```

```
array([[[ 0,  1,  2,  3],
        [ 4,  5,  6,  7],
        [ 8,  9, 10, 11]],

      [[ 12,  13,  14,  15],
        [ 16,  17,  18,  19],
        [ 20,  21,  22,  23]],

      [[ 0, -1, -2, -3],
        [-4, -5, -6, -7],
        [-8, -9, -10, -11]],

      [[-12, -13, -14, -15],
        [-16, -17, -18, -19],
        [-20, -21, -22, -23]]])
```

```
np.concatenate([a,b],axis=1)
```

```
array([[ 0,  1,  2,  3],
       [ 4,  5,  6,  7],
       [ 8,  9, 10, 11],
       [ 0, -1, -2, -3],
       [-4, -5, -6, -7],
       [-8, -9, -10, -11]],

      [[12, 13, 14, 15],
       [16, 17, 18, 19],
       [20, 21, 22, 23],
       [-12, -13, -14, -15],
       [-16, -17, -18, -19],
       [-20, -21, -22, -23]])
```

```
np.concatenate([a,b],axis=2)
```

```
array([[ 0,  1,  2,  3,  0, -1, -2, -3],
       [ 4,  5,  6,  7, -4, -5, -6, -7],
       [ 8,  9, 10, 11, -8, -9, -10, -11]],

      [[12, 13, 14, 15, -12, -13, -14, -15],
       [16, 17, 18, 19, -16, -17, -18, -19],
       [20, 21, 22, 23, -20, -21, -22, -23]])
```

- axis=2 ?

```
np.concatenate([a,b],axis=3)
```

AxisError: axis 3 is out of bounds for array of dimension 3

- axis=3 ?

- ?

(1) np.concatenate([a,b],axis=0)

```
a=np.array(range(2*3*4)).reshape(2,3,4)
b=-a
```

```
a.shape, b.shape, np.concatenate([a,b],axis=0).shape
```

((2, 3, 4), (2, 3, 4), (4, 3, 4))

• => => axis=0 (0 !)

(2) np.concatenate([a,b],axis=1)

```
a=np.array(range(2*3*4)).reshape(2,3,4)
b=-a
```

```
a.shape, b.shape, np.concatenate([a,b],axis=1).shape
```

((2, 3, 4), (2, 3, 4), (2, 6, 4))

• => => axis=1

(3) np.concatenate([a,b],axis=2)

```
a=np.array(range(2*3*4)).reshape(2,3,4)
b=-a
```

```
a.shape, b.shape, np.concatenate([a,b],axis=2).shape
```

((2, 3, 4), (2, 3, 4), (2, 3, 8))

• => => axis=2

(4) np.concatenate([a,b],axis=3)

```
a=np.array(range(2*3*4)).reshape(2,3,4)
b=-a
```

```
a.shape, b.shape, np.concatenate([a,b],axis=3).shape
```

AxisError: axis 3 is out of bounds for array of dimension 3

- $\Rightarrow \Rightarrow \text{axis}=3$.

(1)

```
a=np.array(range(2*3*4)).reshape(2,3,4)
b=-a
```

```
np.concatenate([a,b],axis=-1)
```

```
array([[[ 0,  1,  2,  3,  0, -1, -2, -3],
        [ 4,  5,  6,  7, -4, -5, -6, -7],
        [ 8,  9, 10, 11, -8, -9, -10, -11]],

       [[ 12, 13, 14, 15, -12, -13, -14, -15],
        [ 16, 17, 18, 19, -16, -17, -18, -19],
        [ 20, 21, 22, 23, -20, -21, -22, -23]])
```

```
a.shape, b.shape, np.concatenate([a,b],axis=-1).shape
```

((2, 3, 4), (2, 3, 4), (2, 3, 8))

- $\Rightarrow \Rightarrow \text{axis} = -1$

(2)

```
a=np.array(range(2*3*4)).reshape(2,3,4)
b=-a
```

```
np.concatenate([a,b],axis=-2)
```

```
array([[[ 0,  1,  2,  3],
        [ 4,  5,  6,  7],
        [ 8,  9, 10, 11],
        [ 0, -1, -2, -3],
        [-4, -5, -6, -7],
```

```

[ -8,  -9, -10, -11]],

[[ 12,  13,  14,  15],
 [ 16,  17,  18,  19],
 [ 20,  21,  22,  23],
 [-12, -13, -14, -15],
 [-16, -17, -18, -19],
 [-20, -21, -22, -23]])

```

```
a.shape, b.shape, np.concatenate([a,b],axis=-2).shape
```

```
((2, 3, 4), (2, 3, 4), (2, 6, 4))
```

• 2 => 2 => axis = -2

```
( 3)
```

```
a=np.array(range(2*3*4)).reshape(2,3,4)
b=-a
```

```
np.concatenate([a,b],axis=-3)
```

```

array([[ [ 0,  1,  2,  3],
         [ 4,  5,  6,  7],
         [ 8,  9, 10, 11]],

        [[ 12, 13, 14, 15],
         [ 16, 17, 18, 19],
         [ 20, 21, 22, 23]],

        [[ 0, -1, -2, -3],
         [-4, -5, -6, -7],
         [-8, -9, -10, -11]],

        [[-12, -13, -14, -15],
         [-16, -17, -18, -19],
         [-20, -21, -22, -23]])

```

```
a.shape, b.shape, np.concatenate([a,b],axis=-3).shape
```

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

- 3 ==> 3 ==> axis = -3

```
( 3)
```

```
a=np.array(range(2*3*4)).reshape(2,3,4)
b=-a
```

```
np.concatenate([a,b],axis=-4)
```

AxisError: axis -4 is out of bounds for array of dimension 3

- 4 ==> 4 ==> axis = -4 .

- 0 concatenate .

```
a= np.array(1)
b= np.array(-1)
```

```
a.shape, b.shape
```

```
(( ), ( ))
```

```
np.concatenate([a,b])
```

ValueError: zero-dimensional arrays cannot be concatenated

- a,b .

```
a=np.array(range(4)).reshape(2,2)
b=np.array(range(2)).reshape(2,1)
```

```
np.concatenate([a,b],axis=1)
```

```
array([[0, 1, 0],  
       [2, 3, 1]])
```

```
a.shape, b.shape, np.concatenate([a,b],axis=1).shape
```

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

np.stack

- ?

- $(3,) \quad (3,) \Rightarrow (3,2)$

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

```
a,b
```

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

```
np.concatenate([a,b],axis=1)
```

AxisError: axis 1 is out of bounds for array of dimension 1

-

-

```
a=np.array([1,2,3]).reshape(3,1)  
b=-a
```

```
a,b
```

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

```
np.concatenate([a,b],axis=1)
```

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

- : (3) (3) => (3,1) (3,1) => (3,1) concat (3,1)

-

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

```
np.stack([a,b],axis=1)
```

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

-

```
np.stack([a,b],axis=0)
```

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

-

```
( 1)
```



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

```
a.shape, b.shape, np.stack([a,b],axis=0).shape
```

((3,), (3,), (2, 3))

- (3) (3) => (axis=0) => (1,3) (1,3) => (2,3)

(2)

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

```
a.shape, b.shape, np.stack([a,b],axis=1).shape
```

((3,), (3,), (3, 2))

- (3) (3) => (axis=1) => (3,1) (3,1) => (3,2)

-

```
a=np.arange(3*4*5).reshape(3,4,5)
b=-a
```

```
a.shape, b.shape
```

((3, 4, 5), (3, 4, 5))

```
np.stack([a,b],axis=0).shape # (3,4,5) => (1,3,4,5) //
```

(2, 3, 4, 5)

```
np.stack([a,b],axis=1).shape # (3,4,5) => (3,1,4,5) //
```

(3, 2, 4, 5)

```
np.stack([a,b],axis=2).shape # (3,4,5) => (3,4,1,5) //
```

(3, 4, 2, 5)

```
np.stack([a,b],axis=3).shape # (3,4,5) => (3,4,5,1) //
```

(3, 4, 5, 2)

```
np.stack([a,b],axis=-1).shape # axis=-1 <=> axis=3
```

(3, 4, 5, 2)

```
np.stack([a,b],axis=-2).shape # axis=-2 <=> axis=2
```

(3, 4, 2, 5)

np.concatenate , *np.stack*

sum

- 1

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

array([1, 2, 3])

```
a.sum()
```

6

```
a.sum(axis=0)
```

6

- 2

```
a=np.array(range(6)).reshape(2,3)
a
```

```
array([[0, 1, 2],
       [3, 4, 5]])
```

```
a.sum() #
```

15

```
a.sum(axis=0)
```

```
array([3, 5, 7])
```

```
a.sum(axis=1)
```

```
array([ 3, 12])
```

- 2

```
a.shape, a.sum(axis=0).shape
```

```
((2, 3), (3,))
```

- \Rightarrow axis=0

```
a.shape, a.sum(axis=1).shape
```

((2, 3), (2,))

- \Rightarrow axis=1

-

```
a=np.array(range(10)).reshape(5,2)
a
```

```
array([[0, 1],
       [2, 3],
       [4, 5],
       [6, 7],
       [8, 9]])
```

(1) 1 , 2 ?

() (5,2) \Rightarrow (2,) . (?)

```
a.sum(axis=0)
```

```
array([20, 25])
```

(2) 1 , 2 , ... , 5 ?

() (5,2) \Rightarrow (5,) . (?)

```
a.sum(axis=1)
```

```
array([ 1,  5,  9, 13, 17])
```

(3) a ?

() (5,2) \Rightarrow () . (, ?)

```
a.sum(axis=(0,1))
```

45

```
a.sum() # a.sum(axis=(0,1))
```

45

mean, std, max, min, prod

- sum

```
a=np.array(range(10)).reshape(5,2)
a
```

```
array([[0, 1],
       [2, 3],
       [4, 5],
       [6, 7],
       [8, 9]])
```

```
a.mean(axis=0), a.std(axis=0), a.max(axis=0), a.min(axis=0), a.prod(axis=0)
```

```
(array([4., 5.]),
 array([2.82842712, 2.82842712]),
 array([8, 9]),
 array([0, 1]),
 array([ 0, 945]))
```

```
a.mean(axis=1), a.std(axis=1), a.max(axis=1), a.min(axis=1), a.prod(axis=1)
```

```
(array([0.5, 2.5, 4.5, 6.5, 8.5]),
 array([0.5, 0.5, 0.5, 0.5, 0.5]),
 array([1, 3, 5, 7, 9]),
 array([0, 2, 4, 6, 8]),
 array([ 0,  6, 20, 42, 72]))
```

- std n .

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

1.118033988749895

```
np.sqrt(sum((a-a.mean())**2)/4)
```

1.118033988749895

- n-1 ?

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

1.2909944487358056

```
np.sqrt(sum((a-a.mean())**2)/3)
```

1.2909944487358056

argmax, argmin

- 1

```
a= np.array([1,-2,3,10,4])
a
```

array([1, -2, 3, 10, 4])

```
a.argmax() #
```

3

```
a.argmin() #
```

1

- 2

```
np.random.seed(43052)
a=np.random.randn(4*5).reshape(4,5)
a
```

```
array([[ 0.38342049,  1.0841745 ,  1.14277825,  0.30789368,  0.23778744],
       [ 0.35595116, -1.66307542, -1.38277318, -1.92684484, -1.4862163 ],
       [ 0.00692519, -0.03488725, -0.34357323,  0.70895648, -1.55100608],
       [ 1.34565583, -0.05654272, -0.83017342, -1.46395159, -0.35459593]])
```

```
a.argmin(), a.min()
```

(8, -1.9268448358915802)

```
a.argmax(), a.max()
```

(15, 1.3456558341738827)

```
a.argmin(axis=0), a.argmin(axis=1)
```

(array([2, 1, 1, 1, 2]), array([4, 3, 4, 3]))

```
a.argmax(axis=0), a.argmax(axis=1)
```

```
(array([3, 0, 0, 2, 0]), array([2, 0, 3, 0]))
```

cumsum, cumprod

- 1

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

```
array([1, 2, 3, 4])
```

```
a.cumsum()
```

```
array([ 1,  3,  6, 10])
```

```
a.cumprod()
```

```
array([ 1,  2,  6, 24])
```

- 2

```
a=np.array(range(3*4)).reshape(3,4)  
a
```

```
array([[ 0,  1,  2,  3],  
       [ 4,  5,  6,  7],  
       [ 8,  9, 10, 11]])
```

```
a.cumsum(axis=0), a.cumsum(axis=1)
```



```
(array([[ 0,  1,  2,  3],
       [ 4,  6,  8, 10],
       [12, 15, 18, 21]]),
 array([[ 0,  1,  3,  6],
       [ 4,  9, 15, 22],
       [ 8, 17, 27, 38]]))
```

```
a.cumprod(axis=0), a.cumprod(axis=1)
```

```
(array([[ 0,  1,  2,  3],
       [ 0,  5, 12, 21],
       [ 0, 45, 120, 231]]),
 array([[ 0,  0,  0,  0],
       [ 4, 20, 120, 840],
       [ 8, 72, 720, 7920]]))
```

diff

- 1

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

```
array([1, 2, 4, 6, 7])
```

```
np.diff(a)
```

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

- 2

```
np.diff(np.diff(a))
```

```
array([ 1,  0, -1])
```

- prepend, append

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

```
array([1, 2, 4, 6, 7])
```

```
np.diff(a,prepend=100)
#np.diff(np.array([100]+a.tolist())) )
```

```
array([-99,  1,  2,  2,  1])
```

- [1,2,4,6,7] -> [100,1,2,3,4,6] -> np.diff

```
np.diff(a,append=100)
#np.diff(np.array(a.tolist()+[100])) )
```

```
array([ 1,  2,  2,  1, 93])
```

() a=[1,2,4,6,7] 1 .

```
np.diff(a,prepend=a[0])
#np.diff(a,prepend=1)
```

```
array([0, 1, 2, 2, 1])
```

() a=[1,2,4,6,7] 7 .

```
np.diff(a,append=a[-1])
#np.diff(a,append=7)
```

```
array([1, 2, 2, 1, 0])
```

- 2 array

```
a=np.arange(24).reshape(4,6)
a
```

```
array([[ 0,  1,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10, 11],
       [12, 13, 14, 15, 16, 17],
       [18, 19, 20, 21, 22, 23]])
```

```
np.diff(a,axis=0)
```

```
array([[6, 6, 6, 6, 6, 6],
       [6, 6, 6, 6, 6, 6],
       [6, 6, 6, 6, 6, 6]])
```

```
np.diff(a,axis=1)
```

```
array([[1, 1, 1, 1, 1],
       [1, 1, 1, 1, 1],
       [1, 1, 1, 1, 1],
       [1, 1, 1, 1, 1]])
```

5

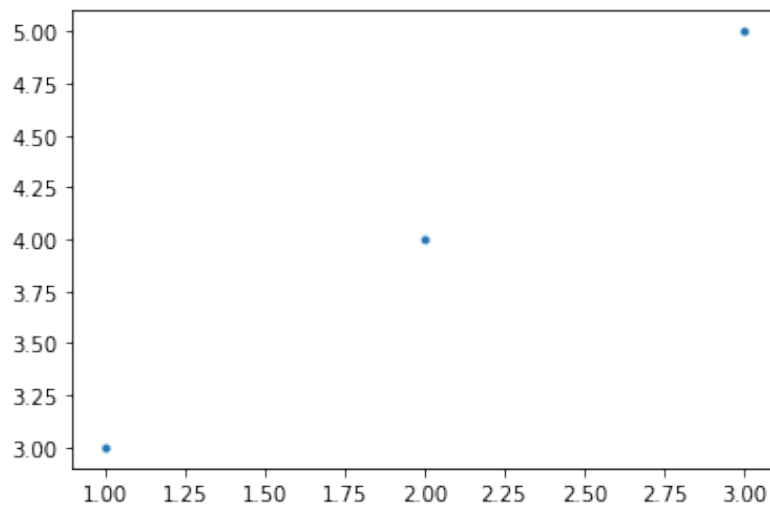
- : matplotlib .

```
import matplotlib.pyplot as plt
```

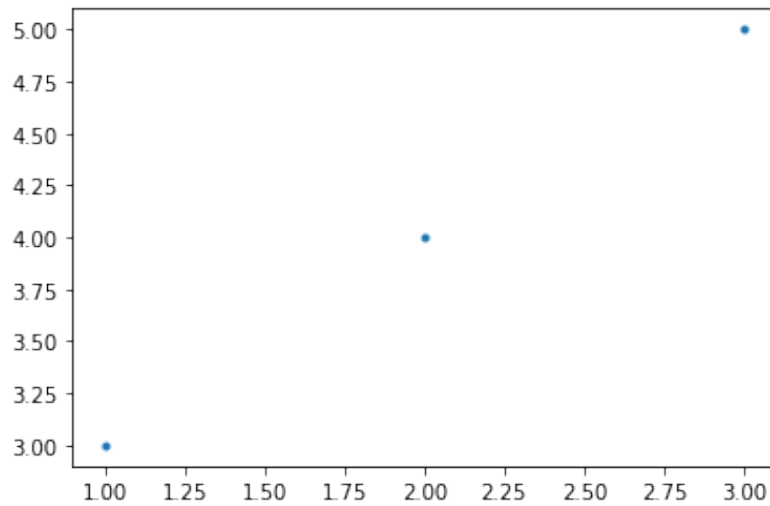
plt.plot

-

```
plt.plot([1,2,3],[3,4,5],'.')
```

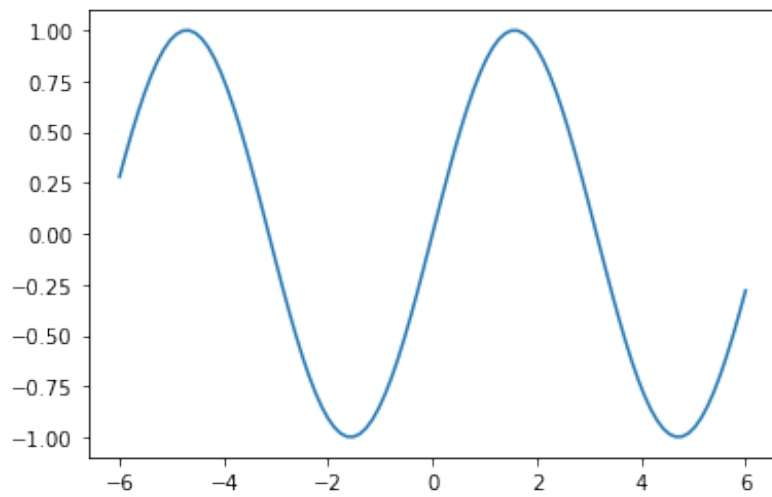


```
plt.plot(np.array([1,2,3]),np.array([3,4,5]),'.')
```

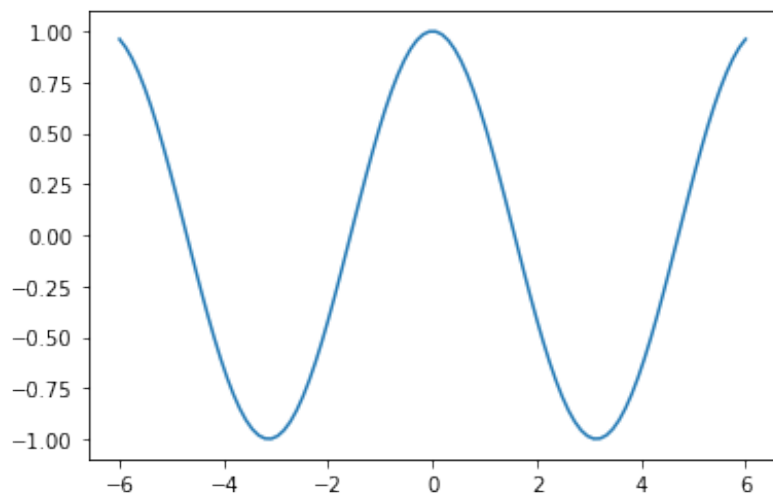


```
-  
  
t=np.linspace(-6,6,100)  
x=np.sin(t)  
y=np.cos(t)
```

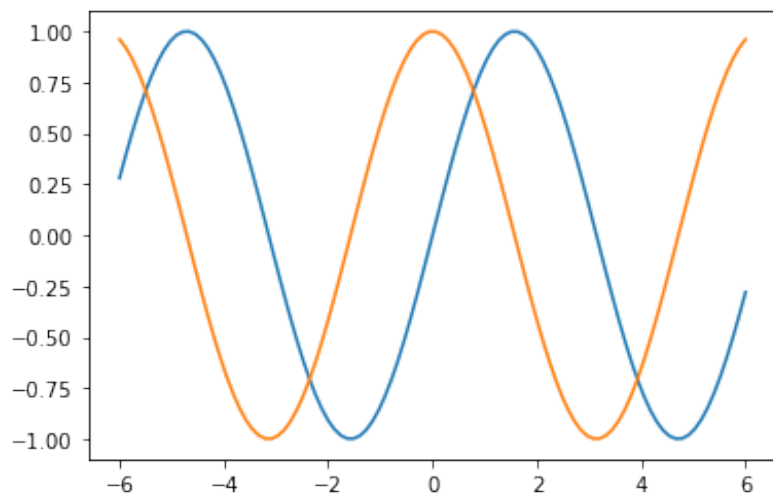
```
plt.plot(t,x)
```



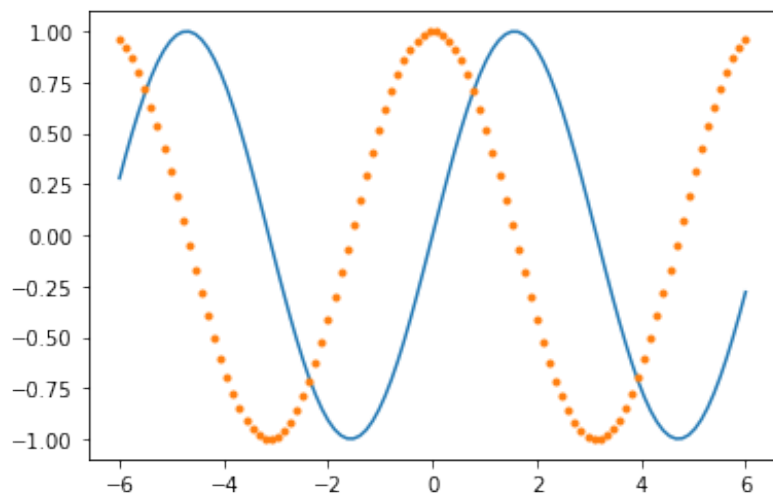
```
plt.plot(t,y)
```



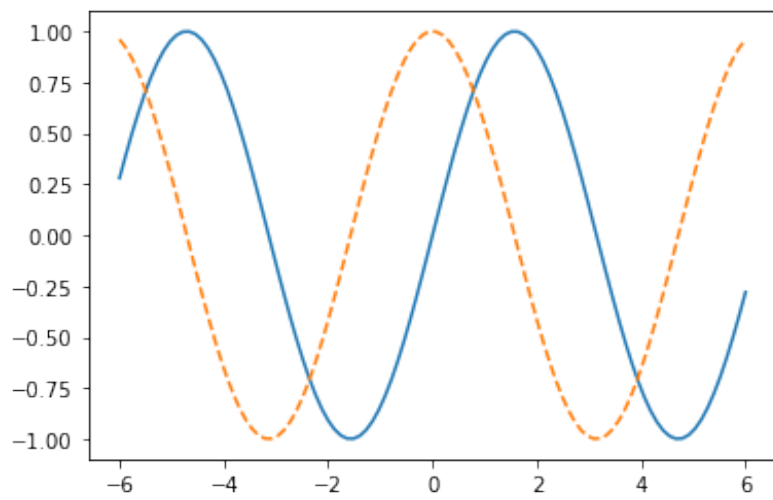
```
plt.plot(t,x)  
plt.plot(t,y)
```



```
plt.plot(t,x)  
plt.plot(t,y,'.')'
```



```
plt.plot(t,x)  
plt.plot(t,y,'--')
```

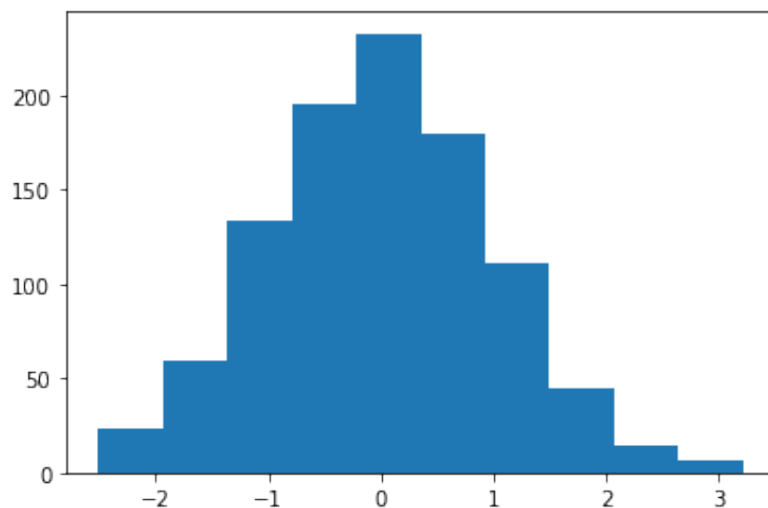


plt.hist

```
X = np.random.randn(1000)
Y = np.random.rand(1000)
```

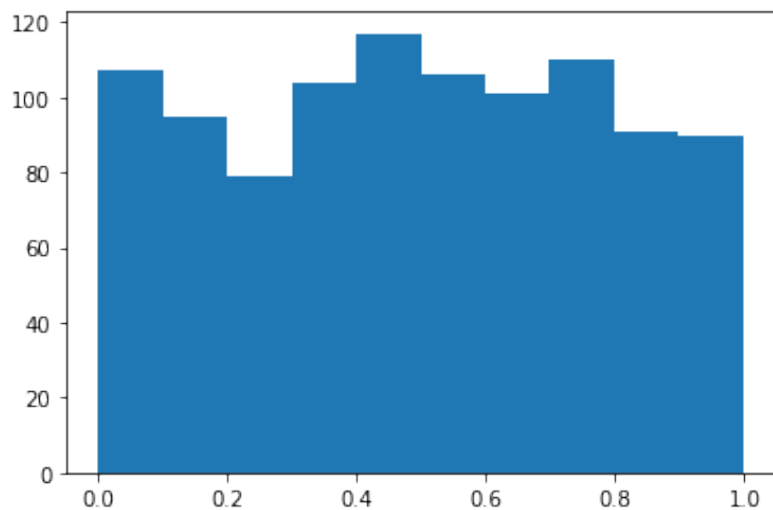
```
plt.hist(X)
```

```
(array([ 23.,  59., 134., 195., 233., 180., 111.,  45.,  14.,   6.]),
 array([-2.50630325, -1.93388828, -1.3614733 , -0.78905833, -0.21664336,
        0.35577162,  0.92818659,  1.50060157,  2.07301654,  2.64543152,
        3.21784649]),
 <BarContainer object of 10 artists>)
```



```
plt.hist(Y)
```

```
(array([107.,  95.,  79., 104., 117., 106., 101., 110.,  91.,  90.]),
 array([0.00168942, 0.10132944, 0.20096946, 0.30060948, 0.4002495 ,
        0.49988951, 0.59952953, 0.69916955, 0.79880957, 0.89844958,
        0.9980896 ]),
 <BarContainer object of 10 artists>)
```

Quiz

()

```
a=np.arange(24).reshape(4,6)
a
```

```
array([[ 0,  1,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10, 11],
       [12, 13, 14, 15, 16, 17],
       [18, 19, 20, 21, 22, 23]])
```

```
axis=1    np.diff    (4,5) array    1    column
.
```

```
array([[1, 1, 1, 1, 1, 1],
       [1, 1, 1, 1, 1, 1],
       [1, 1, 1, 1, 1, 1],
       [1, 1, 1, 1, 1, 1]])
```