Numpy Library in Python

Presented By

Pranab Bora

A Student of Rajiv Gandhi University, Itanagar

Functions

1D array

Declaration in numpy array or list

```
import numpy as np
a= np.array([1,2,3])
b= np.array([4,5,6])
print(a)
print(b)
```

Update list

```
a[0]=11
print(a)
```

```
#one direction
a[1:3]
```

2D array

```
c=np.array([[2,3,4],[7,8,9]])
print(c)
```

ND array

```
c[0:1]
```

Type

```
type(c)
```

Zero matric

```
a=np.zeros([3,4])
#np.zeros([rows,col])
print(a)
```

Find interval

```
a=np.arange(10,25,5)
#np.arange(start, end, interval)
print(a)
```

Find points of interval

```
b=np.linspace(6,7,10)
#np.linspace(start,end,number of points)
print(b)
```

Same number row and col

```
import numpy as np
#c=np.full((row,col),number)
c=np.full((3,4),5)
print(c)
```

Random row and col

```
import numpy as np
d=np.random.random((3,4))
print(d)
```

Return row and col number

```
import numpy as np
a=np.array([[1,2,3],[4,5,6]])
print(a)
print(a.shape)
```

Change row and col number

```
a.shape=(3,2)
print(a)
print(a.shape)
```

Returns row number

```
a.shape=(3,2)
print(a)
print(a.shape[0])
```

Set size, print size, print elements

```
a=np.arange(24)
a.size
a
#np.arange(number) prints from 0 to that number
b=np.arange(10)
print(b.size)
print(b)
```

Return dimension

```
import numpy as np
d=np.array([[1,2,3],[4,5,6]])
print(d.shape)
print(d.shape[0])
#print row
print(d.shape[1])
#prints col
print(d.ndim)
#prints dimension
print(d)
print(d.dtype)
#return data type
```

Return data type

```
a=np.linspace(4,5,5)
print(a)
print(a.dtype)
```

Sum

```
import numpy as np
a=np.array([1,3])
print(np.sum(a))
```

```
import numpy as np
a=np.array([[1,3],[1,1]])
print(np.sum(a))
```

```
import numpy as np
a=np.sum([[1,3],[1,1]], axis=0)
#Add row with row and col with col with other
print(a)
```

```
import numpy as np
a=np.sum([[0,3],[0,1]], axis=1)
#add by same
print(a)
```

Substract

```
import numpy as np
a=np.subtract(2,3)
print(a)
```

Divide

```
import numpy as np
a=np.divide(6,2)
print(a)
```

```
import numpy as np
b=np.array([6,3])
c=np.array([2,1])
a=np.divide(b,c)
print(b)
print(c)
print(a)
```

Multiply

```
a=np.multiply(2,2)
print(a)

b=np.array([6,3])
c=np.array([2,2])
a=np.multiply(b,c)
print(a)

a=np.sqrt(25)
print(a)

a=np.exp(5)
print(a)

a=np.cos(25)
print(a)

a=np.sin(25)
print(a)

a=np.log(25)
print(a)
```

Checking same element one by one

```
import numpy as np
a=[1,2,3]
b=[1,1,7]
print(np.equal(a,b))
```

```
import numpy as np
a=[1,2,3]
b=[1,2,7]
print(np.equal(a,b))
```

Checking same element whole array

```
import numpy as np
a=[1,2,3]
b=[1,2,3]
print(np.array_equal(a,b))
```

```
import numpy as np
a=[1,2,3]
b=[1,2,4]
print(np.array_equal(a,b))
```

Min and Max

```
import numpy as np
a=[1,2,3]
print(sum(a))
print(min(a))
print(np.max(a))
```

Mean Median StandardDeviation

```
import numpy as np
a=[1,2,3]
print(np.sum(a))
print(np.mean(a))
print(np.median(a))
print(np.std(a))
print(np.corrcoef(a))
```

Array broadcasting in addition

```
a=np.array([[1,2,3],[4,5,6]])
b=np.array([1,1,1])
c=np.sum([a,b])
print(a)
print(b)
print('After adding')
print(c)
```

```
a=np.array([[1,2,3],[2,3,4],[4,5,6],[5,6,7],[6,7,8],[7,8,9]])
b=np.array([1,1,1])
print(np.sum([a,b]))
```

```
a=np.array([[1,2,3],[2,3,4],[4,5,6]])
b=np.array([1,1,1])
print(np.sum([a,b]))
could not broadcast input array from shape (3,3) into shape (3)
```

Array broadcasting in substraction

```
a=np.array([[1,2,3],[2,3,4],[4,5,6],[5,6,7],[6,7,8],[7,8,9]])
b=np.array([1,1,1])
print(np.subtract(a,b))
```

Indexing

```
import numpy as np
a=np.array([[1,2,3],[4,5,6],[7,8,9]])
print('first row')
print(a[0])
```

```
            import numpy as np
            a=np.array([[1,2,3],[4,5,6],[7,8,9]])
            print(' upto first row')
            print(a[:1])
```

Slicing

```
import numpy as np
a=np.array([[1,2,3],[4,5,6],[7,8,9]])
print('upto that row and col')
print(a[:1,:1])
```

```
            import numpy as np
            a=np.array([[1,2,3],[4,5,6],[7,8,9]])
            print('upto that row and col')
            print(a[:1,:2])
```

```
import numpy as np
a=np.array([[1,2,3],[4,5,6],[7,8,9]])
print('from row and upto col')
print(a[2:,:3])                                        #2: starting with #:2 upto that
```

```
import numpy as np
a=np.array([[1,2,3],[4,5,6],[7,8,9]])
print(a)
print(a[:2,2:])
```

Select particular element

```
import numpy as np
a=np.array([[1,2,3],[4,5,6],[7,8,9]])
print(a)
print(a[:1,2:])
```

Array Manipulation

```
import numpy as np
a=np.array([1,2,3])
b=np.array([4,5,6])
print(a)
print(b)
c=np.concatenate([a,b])
print('one dimension array')
print(c)
```

Array manipulation (concatenate)

```
import numpy as np
a=np.array([[1,2,3],[4,5,6]])
b=np.array([[7,8,9],[10,11,12]])
print(a)
print(b)
c=np.concatenate([a,b], axis=0)
print('two dimension array axis=0 means row wise')
print(c)

import numpy as np
a=np.array([[1,2,3],[4,5,6]])
b=np.array([[7,8,9],[10,11,12]])
print(a)
print(b)
c=np.concatenate([a,b], axis=1)
print('two dimension array axis=1 means col wise')
print(c)
```

Stack

row

```
a=np.array([1,2,3])
b=np.array([4,5,6])
c=np.stack((a,b), axis=0)
print(c)
```

col

```
a=np.array([1,2,3])
b=np.array([4,5,6])
c=np.stack((a,b), axis=1)
print(c)
```

Horizontal and vertical statck in 2d array

```
import numpy as np
a=np.array([[1,2,3],[4,5,6]])
b=np.array([[7,8,9],[10,11,12]])
print('Matrix a')
print(a)
print('Matrix b')
print(b)
print('horizontal stack')
print(np.hstack((a,b)))
print('vertical stack')
print(np.vstack((a,b)))
```

Diff between concatenate and stack

```
import numpy as np
a=np.array([[1,2,3],[4,5,6]])
b=np.array([[7,8,9],[10,11,12]])
print('Matrix a')
print(a)
print('Matrix b')
print(b)
print('horizontal stack')
print(np.hstack((a,b)))
print('horizontal concatenate')
print(np.concatenate((a,b),axis=0))
print('vertical stack')
print(np.vstack((a,b)))
print('vertical concatenate')
print(np.concatenate((a,b),axis=1))
```

Column statck

```
print('Column stack')
print(np.column_stack((a,b)))
```

Spliting row

```
np.split(array,split,row or col)

import numpy as np
a=np.array([[1,2,3],[4,5,6]])
print(a)
print('after spliting')
print(np.split(a,2,axis=0))
```

Spliting col

```
import numpy as np
a=np.array([[1,2,3],[4,5,6]])
print(a)
print('after spliting')
print(np.split(a,3,axis=1))
```

Thank You !!!!