Numpy'nin işlevleri numpy, python'da bilimsel hesaplamanın temel kütüphanesidir, bu nedenle veri yaparsanız veri bilimi makine öğrenimi veya derin öğrenme ile ilgili herhangi bir şey ve aynı zamanda temel pakettir Matplotlib ve daha pek çok şey artık numpy'deki merkezi nesnedir.

Çok boyutlu dizi böylece numpy yüksek performans sağlar
İşlemlerin gerçekten hızlı olmasını sağlayan bir dizi nesnesi var ve aynı zamanda
Bu dizilerle matematiksel işlemler için tüm araçları sağlar ve bir yan not olarak kodun çoğunu sağlar

Numpy'nin tamamı c ile yazılır ve python fonksiyonları sadece bu c'lerin etrafındaki sarmalayıcılardır.
Numpy code :

```
import numpy as np
print(np.____version___)
```

1.26.3

```
import numpy as np

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

```
[1 2 3]
(3,)
int64
```

```
import numpy as np

a=np.array([1,2,3,4])
print(a)
print(a.shape)
print(a.dtype)
print(a.ndim)
print(a.size)
print(a.itemsize)
```

```
[1 2 3 4]
(4,)
int64
1
4
8
import numpy as np

a=np.array([1,2,3,4])
print(a)
print(a[0])
a[0]=10
print(a)
import numpy as np

a=np.array([1,2,3,4])
print(a)
print(a[0])
a[0]=10
print(a)
```

```
[1 2 3 4]
1
[10  2  3  4]
```

```
import numpy as np


a=np.array([1,2,3,])
print(a)
print(a[0])
a[0]=10
print(a)

b=a=np.array([2,0,2])
print(b)
import numpy as np
```

```
[10  2  3]
[20,0,6]
```

Array

```
import numpy as np
l=[1,2,3]
a=np.array([1,2,3])

l=l+[4]
print(l)
a=a+np.array([4])
print(a)
```

```
[1, 2, 3, 4]
[5 6 7]
```

```
import numpy as np
l=[1,2,3]
a=np.array([1,2,3])
a=np.sqrt(a)
print(a)
```

```
[1.         1.41421356 1.73205081]
```

Dot Product

```
#dot pruduct
dot=0
for i in range(len(l1)):
  dot+=l1[i] * l2[i]
  print(dot)
  dot=np.dot(a1,a2)
  print(dot)
  sum1=a1*a2
  dot=(a1*a2).sum()
  print (dot)
  dot=a1@a2
  print(dot)
32
```

32
32
32

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

b=a[0,1]
print(b)
```

```
[[1 2]
 [3 4]]
2
```

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

b=a[0,1:3]
print(b)
```

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

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

b=a[-1,-1]
print(b)
```

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

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

bool_idx=a<2
print(bool_idx)
print(a[bool_idx])
```

```
[[1 2]
 [3 4]
 [5 6]]
[[ True False]
 [False False]
 [False False]]
[1]
```

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

print(a[a>2])
b=np.where(a>2,a,-1)
```

```
print(b)
[[1 2]
 [3 4]
 [5 6]]
[3 4 5 6]
[[-1 -1]
 [ 3  4]
 [ 5  6]]
import numpy as np
a=np.arrange(1,7)
print(a)
print(a.shape)
b=a[np.newaxis,:]

print(b)
[1 2 3 4 5 6]
(6,)
[(1 2 3 4 5 6 ]

import numpy as np
a=np.array([1,2,3,4])
b=np.array([5,6,7,8])
c=np.vstack((a,b))
print(c)

[[1 2 3 4]
 [5 6 7 8]]

import numpy as np
x=np.array([[1,2,3],[4,5,6],[1,2,3],[4,5,6]])

a=np.array([1,0,1])
y=x+a
print(y)
[[2 2 4]
 [5 5 7]
 [2 2 4]
 [5 5 7]]

import numpy as np
a=np.array([[7,8,9,10,11,12,13],[17,18,19,20,21,22,23]])
print(a)
print(a.sum(axis=None))
[[ 7  8  9 10 11 12 13]
 [17 18 19 20 21 22 23]]
210

import numpy as np
a=np.array([1,2,3])
b=a.copy()
b[0]=42
print(b)
print(a)
[42  2  3]
[1 2 3]

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

```
a=np.ones((2,3))
print(a)
a=np.full((2,3),5.0)
print(a)
a=np.eye(3)
print(a)
```

```
[[0. 0. 0.]
 [0. 0. 0.]]
[[1. 1. 1.]
 [1. 1. 1.]]
[[5. 5. 5.]
 [5. 5. 5.]]
[[1. 0. 0.]
 [0. 1. 0.]
 [0. 0. 1.]]
```

```
import numpy as np
a=np.random.random((3,2))
print(a)
a=np.random.randn(1000)
print(a.mean(),a.var())
a=np.random.randint(10,size=(3,3))
print(a)
a=np.random.choice(5,size=10)
print(a)
a=np.random.choice([-8,-7,-6],size=10)
print(a)
```

```
[[0.5951818  0.8685035 ]
 [0.67975063 0.02074646]
 [0.77925884 0.28703115]]
0.02547205775917928 1.0098392922197925
[[8 6 6]
 [7 7 7]
 [2 9 1]]
[3 2 2 4 1 2 4 4 1 3]
[-7 -8 -8 -7 -8 -7 -6 -7 -8 -8]
```

```
import numpy as np
a=np.array([[1,2],[3,4]])
eingevalues,eigenvectors=np,linalg.eig(a)
b=eigenvectors[:,0]*eingevalues[0]
print(b)
c=a@ eigenvectors[:,0]
print(b)

print(np.allclose(b,c))
```

```
import numpy as np
A=np.array([[1,1],[1.5,4.0]])
b=np.array([2200,5050])
x=np.linalg.inv(A).dot(b)
print(x)
x=np.linalg.solve(A,b)
print(x)
```

[1500. 700.]
[1500. 700.]

NumPy, Python programlama dilinde bilimsel hesaplamalar ve veri analizi için kullanılan güçlü bir kütüphanedir. İşte NumPy'nin temel özellikleri ve fonksiyonlarına kısa bir özet:

- **Numpy Array:**
  - NumPy'nin temel veri yapısı "numpy array"dir. Bu, homojen veri tiplerini destekleyen çok boyutlu dizilerdir.
  - Diziler, vektörler ve matrisler gibi çok boyutlu veri yapılarına olanak tanır.
- **Veri Tipi Desteği:**
  - NumPy, C ve Fortran dilindeki veri tiplerini destekler ve bu sayede bellek kullanımını optimize eder.
  - Örneğin, int, float, complex gibi farklı veri tiplerini kullanabilirsiniz.
- **Matematiksel İşlemler:**
  - NumPy, vektörel ve matris işlemleri gibi hızlı matematiksel operasyonları destekler.
  - Eleman bazlı operasyonlar, matris çarpımı, transpoz gibi işlemleri kolayca gerçekleştirebilirsiniz.
- **Broadcasting:**
  - NumPy'de broadcasting, farklı şekillerdeki diziler üzerinde işlemler yapmayı kolaylaştırır. Bu, boyutları uyumsuz diziler arasında otomatik olarak yayma (broadcasting) işlemini ifade eder.
- **Rastgele Sayı Üretimi:**
  - NumPy, rastgele sayı üretimi için çeşitli fonksiyonlar içerir. Bu, simülasyonlar ve istatistiksel analizlerde kullanışlıdır.
- **İndeksleme ve Dilimleme:**
  - NumPy, diziler üzerinde veriye erişim için güçlü indeksleme ve dilimleme özelliklerine sahiptir.
  - Veriyi seçmek, değiştirmek veya alt küme oluşturmak için kullanılır.
- **Matris İşlemleri ve Lineer Cebir:**
  - NumPy, lineer cebir operasyonları için özel fonksiyonlar içerir. Örneğin, determinant hesaplama, ters matris bulma gibi.
- **Veri Girişi ve Çıkışı:**
  - NumPy, diskten veya diğer veri kaynaklarından veri okuma ve yazma işlemlerini destekleyen fonksiyonlar içerir.
- **İstatistiksel Fonksiyonlar:**
  - NumPy, temel istatistiksel hesaplamaları kolaylaştıran bir dizi fonksiyon içerir. Ortalama, varyans, standart sapma gibi.