

80.02 np.diag

原贴: https://blog.csdn.net/qq_35277038/article/details/80766746

NumPy包中的内置`diag`函数很有意思。

假设创建一个**1维数组a**，和一个**3*3数组b**：

```
import numpy as np
a = np.arange(1, 4)
b = np.arange(1, 10).reshape(3, 3)
```

结果如下：

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

使用`diag`函数，看一看结果：

```
>>> np.diag(a)
array([[1, 0, 0],
       [0, 2, 0],
       [0, 0, 3]])
>>> np.diag(b)
```

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

可以发现，当 **np.diag(array)** 中

array是一个**1维数组**时，结果形成一个以一维数组为对角线元素的矩阵

array是一个**二维矩阵**时，结果输出矩阵的对角线元素

进一步，还有第二个参数 k

原贴：https://blog.csdn.net/The_Time_Runner/article/details/89918383

numpy.diag(v, k=0) [官方文档](#)

以一维数组的形式返回方阵的对角线（或非对角线）元素，或将一维数组转换成方阵（非对角线元素为0）。两种功能角色转变取决于输入的v。¹

更深层的见[numpy.diagnal\(\)](#)

参数详解：

v : array_like.

如果v是2D数组，返回k位置的对角线。

如果v是1D数组，返回一个v作为k位置对角线的2维数组。

k : int, optional

对角线的位置，大于零位于对角线上面，小于零则在下面。

示例

```
>>> x = np.arange(9).reshape((3,3))
```

```

>>> x
array([[0, 1, 2],
       [3, 4, 5],
       [6, 7, 8]])

>>> np.diag(x)
array([0, 4, 8])
>>> np.diag(x, k=1)
array([1, 5]) // k=1, 取对角线上一层; 若k=2, 取对角线上二层
>>> np.diag(x, k=-1)
array([3, 7])

>>> np.diag(np.diag(x))
array([[0, 0, 0],
       [0, 4, 0],
       [0, 0, 8]])

```

```

arr2D = np.diag(1+np.arange(4), k=-1)
print(arr2D)

```

```

[[0 0 0 0 0]
 [1 0 0 0 0]
 [0 2 0 0 0]
 [0 0 3 0 0]
 [0 0 0 4 0]]

```

```

arr2D = np.diag(1+np.arange(4), k=-2)
print(arr2D)

```

```

[[0 0 0 0 0 0]
 [0 0 0 0 0 0]
 [1 0 0 0 0 0]
 [0 2 0 0 0 0]
 [0 0 3 0 0 0]
 [0 0 0 4 0 0]]

```