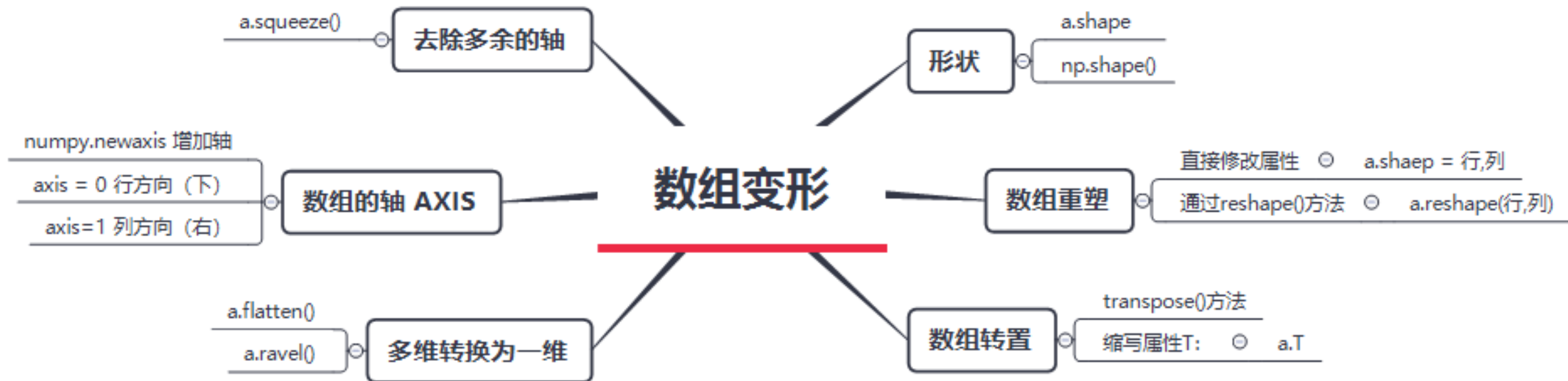


Python科学计算库Numpy之

03-数组形状





数组重塑

```
In [2]:
```

```
a = arange(6)  
a
```

```
Out[2]:
```

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

修改数组的shape属性

与之对应的方法是 `reshape`，但它不会修改原来数组的值，而是返回一个新的数组

注意：`shape` 和 `reshape` 方法不能改变数组中元素的总数

将形状修改为2乘3：

```
In [3]:
```

```
a.shape = 2,3  
a
```

```
Out[3]:
```

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

```
In [4]:
```

```
a.reshape(3,2)
```

```
Out[4]:
```

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

数组转置

data

1	2
3	4
5	6

data.T

1	3	5
2	4	6

```
a = np.arange(1,7).reshape(3,2)
a
array([[1, 2],
       [3, 4],
       [5, 6]])
```

a.transpose()

使用 transpose()方法 返回数组的转置，
本质上是将所有维度反过来

或者使用缩写属性T:

data

1
2
3
4
5
6

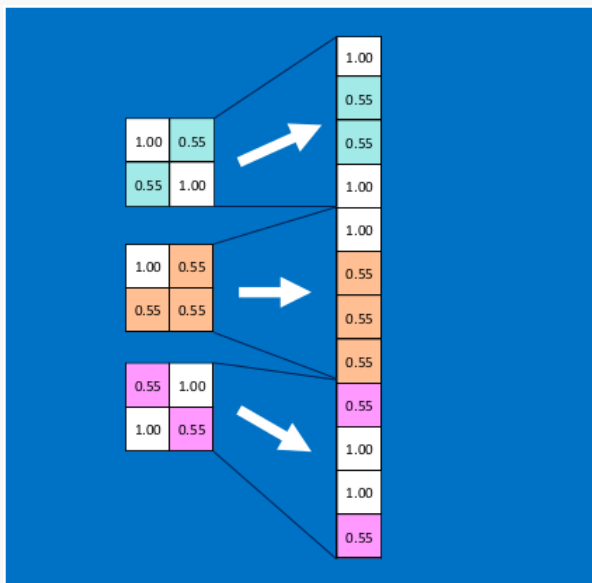
data.reshape(2,3)

1	2	3
4	5	6

data.reshape(3,2)

1	2
3	4
5	6

多维数组变成一维数组



将多维数组转化为1维数组

```
b = a.flatten()
```

```
b = a.ravel()
```

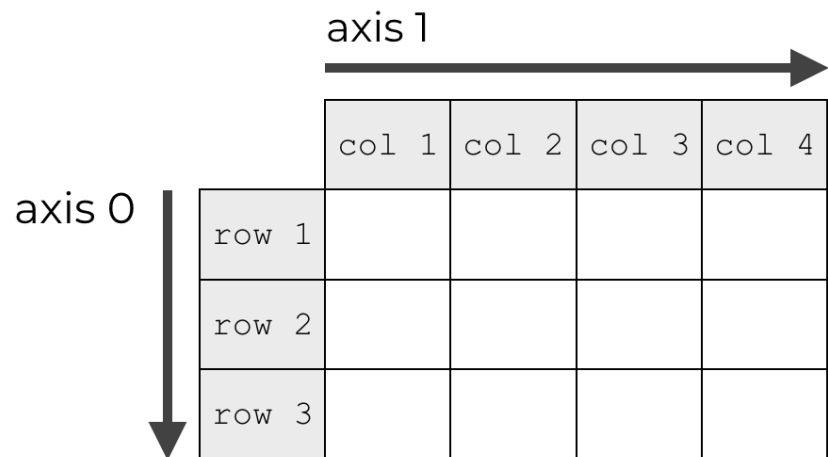
后者修改b会改变a，前者不会

```
a = np.array([[0, 1],  
              [2, 3]])  
  
b = a.flatten()  
b  
  
array([0, 1, 2, 3])
```

```
a = np.array([[0, 1],  
              [2, 3]])  
b = a.ravel()  
b  
  
array([0, 1, 2, 3])
```

增加数组维数

```
>> import numpy as np
>> a = np.arange(4)    #创建一个一维数组
>> a.shape
(4,)
>> b=a[np.newaxis, :]  #在行处使用
>> b.shape
(1, 4)
>> c=a[:, np.newaxis]  #在列处使用
>> c.shape
(4, 1)
```



`numpy.newaxis`

首先理解“轴”的概念！

在二维NumPy数组中，轴是沿行和列的方向。

AXIS 0 轴是沿着行（rows）的方向

axis 0 是第一轴。对于二维或多维数组，axis 0 是沿行（row）向下的轴。

（一维数组是特例，只有一个轴（即axis=0））

AXIS 1 轴是沿着列（columns）的方向

在NumPy数组中，axis 1 是第2根轴。对于二维或多维数组，axis 1 是沿列（columns）横穿的轴。

去除多余的轴

`squeeze()`

`squeeze` 返回一个将所有长度为1的维度去除的新数组。

```
y = a[np.newaxis, np.newaxis, :]  
y.shape
```

```
(1, 1, 5)
```

```
y.squeeze()
```

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

谢谢!

