1.5 布尔索引

2020年4月4日 11:50

一维数组

```
import numpy as np
数组 = np.arange(10)
print(数组)
筛选 = 数组 > 5
print(筛选) # 返加False和True
print(数组[筛选]) # 返回6 7 8 9
```

实例1：把一维数组进行01化处理

假设这10个数字，我想让大于5的数字变成1，小于等于5的数字变成0

```
数组[数组<=5] = 0    # 小于5的重新赋值为0
数组[数组>5] = 1     #  大于5的重新赋值为1
print(数组)
```

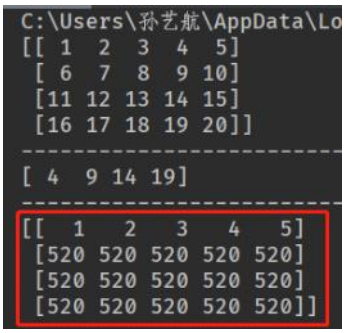
实例2：进行自增量的操作，给大于5的加上520

```
数组[数组>5] += 520
print(数组)
```

二维数组

```
import numpy as np
数组 = np.arange(1,21).reshape(4,5)
print(数组)
筛选 = 数组>10
print(筛选)       # 返回一个布尔数组，即有行又有列
print(数组[筛选])   # 返回所有为True的对应数字组成的数组，以一维数组展现
```

```
# 例：把第3例大于5的行筛选出来并重新赋值为520
import numpy as np
数组 = np.arange(1,21).reshape(4,5)
print(数组)
print("-"*30)
print(数组[:,3]) # 所有行，第3列
print("-"*30)
筛选 = 数组[:,3] > 5  # 所有行第3列，大于5的
数组[数组[:,3]>5] = 520
print(数组)
```



一、所的行第3列改成520

```
import numpy as np
数组 = np.arange(1,21).reshape(4,5)
数组[:,3] = 520
print(数组)
```

二、**所有行第3列，大于5的改成520**

```
import numpy as np
数组 = np.arange(1,21).reshape(4,5)
筛选 = 数组[:,3]>5  # 所有行第3列，大于5的
数组[:,3][筛选] = 520
print(数组)
```

```
import numpy as np
数组 = np.array([[10,20,30],[50,40,10],[10,1,10]])
print(数组)
筛选 = 数组>25
print(筛选)
print(数组[筛选])
```

条件组合：找出偶数或小于7的数

```
import numpy as np
数组 = np.arange(10)
print(数组)
print("-"*30)
条件 = (数组%2==0) | (数组<7)
print(条件)
print("-"*30)
print(数组[条件])
```