Python TUTORIAL

NumPy reshape()

Summary: in this tutorial, you'll learn how to use the numpy `reshape()` function to change the shape of an array.

Introduction to the numpy reshape() function

A shape of an array (https://www.pythontutorial.net/python-numpy/create-numpy-array/) stores the number of dimensions (or axes) and the number of elements on each dimension. The `shape` property returns a tuple that describes the shape of an array.

The `reshape()` function changes the shape of an array without changing its elements. Here's the syntax of the `reshape()` function:

```
numpy.reshape(a, newshape, order='C')
```

In this syntax, the `reshape()` function changes the shape of the array `a` to the `newshape` but keep the number of elements the same.

The `reshape()` function is equivalent to calling the `reshape()` method on the array a:

```
a.reshape(newshape, order='C')
```

NumPy reshape() function examples

Let's take some examples of using the `reshape()` function.

1) Using numpy reshape() function with 1-D array example

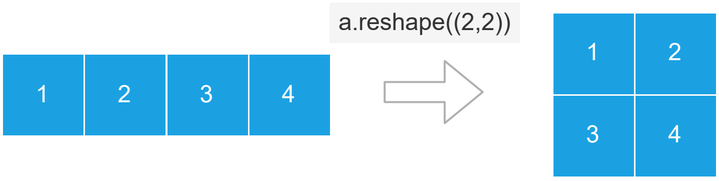
The following example uses the numpy `reshape()` function to change a 1-D array with 4 elements to a 2-D array:

```
import numpy as np

a = np.arange(1, 5)
print(a)

b = np.reshape(a, (2, 2))
print(b)
```

Output:

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

How it works.



First, create a 1-D array with four numbers from 1 to 4 by using the `arange()` `(https://www.pythontutorial.net/python-numpy/numpy-arange/)` function:

```
a = np.arange(1, 5)
print(a)
```

Second, change the shape of array a to an array with two dimensions, each dimension has 2 elements:

```
b = np.reshape(a, (2, 2))
print(b)
```

2) Numpy reshape() returns a view

Note that array b is a view of array a. It means that if you change an element of array b, the change is reflected in array a. For example:

```
import numpy as np

a = np.arange(1, 5)
b = np.reshape(a, (2, 2))

# change the element [0,0]
b[0, 0] = 0

print(b)
print(a)
```

Output:

```
[[0 2]
 [3 4]]
[0 2 3 4]
```

In this example, we change the element at index [0,0] in the array b. The change is also reflected in the array a.

Summary

- Use the numpy `reshape()` function to change the shape of an array without changing its elements.

- You can change the shape of an array as long as the number of elements is the same.