


NumPy

Introdução

Transposição de arrays e troca de eixos

Transposição de arrays e troca de eixos

Em matemática, matriz transposta é a matriz que se obtém da troca de linhas por colunas de uma dada matriz. Desta forma, transpor uma matriz é a operação que leva à obtenção de sua transposta.


$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^T = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}.$$

$$\begin{bmatrix} 1 & 2 \end{bmatrix}^T = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

Transposição de arrays e troca de eixos

A transposição de um array retorna uma visualização dos dados sem realização de cópias.

Podemos usar os métodos *transpose* ou o atributo especial *T* de um *array* para realizar transposição.

Transposição de arrays e troca de eixos

Exemplo usando o atributo especial *T*.

```
In [7]: import numpy as np
In [8]: arr = np.arange(15).reshape((3, 5))
In [9]: arr
Out[9]:
array([[ 0,  1,  2,  3,  4],
       [ 5,  6,  7,  8,  9],
       [10, 11, 12, 13, 14]])

In [10]: arr.T
Out[10]:
array([[ 0,  5, 10],
       [ 1,  6, 11],
       [ 2,  7, 12],
       [ 3,  8, 13],
       [ 4,  9, 14]])
```

Transposição de arrays e troca de eixos

Exemplo usando o método transpose.

```
In [13]: import numpy as np

In [14]: arr = np.arange(15).reshape((3, 5))

In [15]: arr
Out[15]:
array([[ 0,  1,  2,  3,  4],
       [ 5,  6,  7,  8,  9],
       [10, 11, 12, 13, 14]])

In [16]: arr.transpose()
Out[16]:
array([[ 0,  5, 10],
       [ 1,  6, 11],
       [ 2,  7, 12],
       [ 3,  8, 13],
       [ 4,  9, 14]])
```

Transposição de arrays e troca de eixos

Vamos agora usar o transpose com um array com mais dimensões. Usaremos o shape para visualizar as dimensões.

```
In [183]: arr = np.arange(24).reshape((2,3,4))
In [184]: arr
Out[184]: array([[[ 0,  1,  2,  3],
                  [ 4,  5,  6,  7],
                  [ 8,  9, 10, 11]],
                 [[12, 13, 14, 15],
                  [16, 17, 18, 19],
                  [20, 21, 22, 23]]])
In [185]: arr.shape
Out[185]: (2, 3, 4)
```

Nosso array possui duas dimensões, três linhas e quatro colunas.

```
In [187]: x = arr.transpose(2,0,1)
In [188]: x
Out[188]: array([[[ 0,  4,  8],
                  [12, 16, 20]],
                 [[ 1,  5,  9],
                  [13, 17, 21]],
                 [[ 2,  6, 10],
                  [14, 18, 22]],
                 [[ 3,  7, 11],
                  [15, 19, 23]]])
In [189]: x.shape
Out[189]: (4, 2, 3)
```

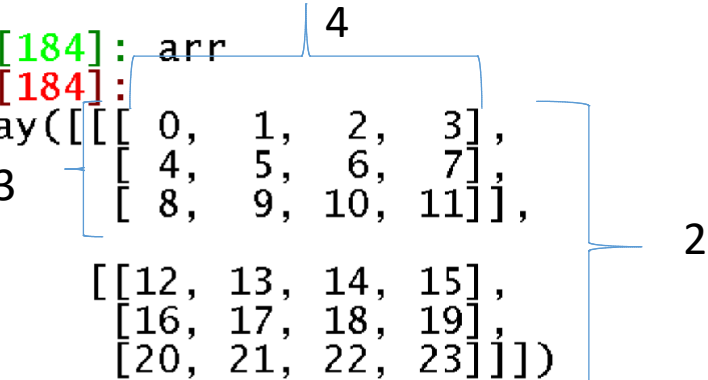
O transpose recebe como parâmetros o índice dos eixos. Vamos alterar passando os índices 2, 0 e 1, nesta ordem.

Transposição de arrays e troca de eixos

Podemos trocar os eixos usando o método `swapaxes` que recebe por parâmetro dois inteiros que são os números dos eixos que serão trocados.

```
In [183]: arr = np.arange(24).reshape((2,3,4))
```

```
In [184]: arr
Out[184]: array([[[ 0,  1,  2,  3],
                  [ 4,  5,  6,  7],
                  [ 8,  9, 10, 11]],
                 [[12, 13, 14, 15],
                  [16, 17, 18, 19],
                  [20, 21, 22, 23]]])
```




```
In [185]: arr.shape
Out[185]: (2, 3, 4)
```

```
In [190]: x = arr.swapaxes(2, 1)
```

```
In [191]: x
Out[191]: array([[[ 0,  4,  8],
                  [ 1,  5,  9],
                  [ 2,  6, 10],
                  [ 3,  7, 11]],
                 [[12, 16, 20],
                  [13, 17, 21],
                  [14, 18, 22],
                  [15, 19, 23]]])
```

Trocando os eixos (linhas e colunas).



```
In [192]: x.shape
Out[192]: (2, 4, 3)
```

FIM