

» Python第一讲

Numpy 简介

龚怡

2167570874@qq.com

计算机学院





导入numpy

Numpy是Python的一个很重要的第三方库，很多其他科学计算的第三方库都是以Numpy为基础建立的。

Numpy的一个重要特性是它的数组计算。

在使用Numpy之前，我们需要导入numpy包：

```
from numpy import *
```



使用前一定要先导入 Numpy 包，导入的方法有以下几种：

```
import numpy
import numpy as np
from numpy import *
from numpy import array, sin
```

事实上，在ipython中可以使用magic命令来快速导入Numpy的内容。

```
%pylab
```

Using matplotlib backend: Qt4Agg
Populating the interactive namespace from
numpy and matplotlib



数组上的数学操作

假如我们想将列表中的每个元素增加1，但列表不支持这样的操作（报错）：

```
a = [1, 2, 3, 4]
a + 1
```

TypeError

Traceback

(most recent call last)

<ipython-input-3-068856d2a224> in <module>()
1 a = [1, 2, 3, 4]

----> 2 a + 1

TypeError: can only concatenate list (not "int") to list

转成 array :



```
a = array(a)
a
```

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

array 数组支持每个元素加 1 这样的操作:

```
a + 1
```

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

与另一个 array 相加, 得到对应元素相加的结果:

```
b = array([2, 3, 4, 5])
a + b
```

```
array([3, 5, 7, 9])
```



对应元素相乘：

```
a * b
```

```
array([ 2,  6, 12, 20])
```

对应元素乘方：

```
a ** b
```

```
array([ 1,  8, 81, 1024])
```



提取数组中的元素

提取第一个元素：

```
a[0]
```

1

提取前两个元素：

```
a[:2]
```

```
array([1, 2])
```



最后两个元素：

```
a[-2:]
```

```
array([3, 4])
```

将它们相加：

```
a[:2] + a[-2:]
```

```
array([4, 6])
```




修改数组形状

查看 array 的形状:

```
a.shape
```

```
(4, )
```

修改 array 的形状:

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

```
a
```

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



多维数组

a 现在变成了一个二维的数组，可以进行加法：

```
a + a
```

```
array([[2, 4],  
       [6, 8]])
```

乘法仍然是对应元素的乘积，并不是按照矩阵乘法来计算：

```
a * a
```

```
array([[1, 4],  
       [9, 16]])
```



画图

linspace 用来生成一组等间隔的数据：

```
a = linspace(0, 2*pi, 21)


```
%precision 3
a
```


```

```
array([ 0.      ,  0.314,  0.628,  0.942,  1.257,
        1.571,  1.885,  2.199,
        2.513,  2.827,  3.142,  3.456,  3.77 ,
        4.084,  4.398,  4.712,
        5.027,  5.341,  5.655,  5.969,  6.283])
```



三角函数:

```
b = sin(a)
```

```
b
```

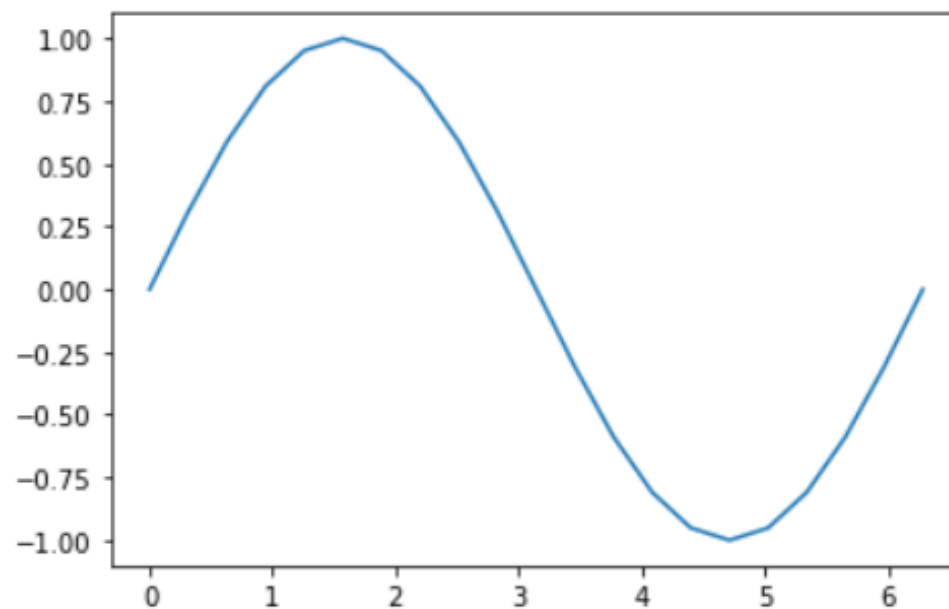
```
array([ 0.000e+00,  3.090e-01,  5.878e-01,  
       8.090e-01,  9.511e-01,  
        1.000e+00,  9.511e-01,  8.090e-01,  
       5.878e-01,  3.090e-01,  
        1.225e-16, -3.090e-01, -5.878e-01,  
      -8.090e-01, -9.511e-01,  
       -1.000e+00, -9.511e-01, -8.090e-01,  
      -5.878e-01, -3.090e-01,  
       -2.449e-16])
```



画出图像：

```
%matplotlib inline  
plot(a, b)
```

[<matplotlib.lines.Line2D at 0xa128ba8>]





从数组中选择元素

假设我们想选取数组b中所有非负的部分，首先可以利用b产生一组布尔值：

```
b >= 0
```

```
array([ True,  True,  True,  True,  True,  True,
        True,  True,  True,
           True,  True, False, False, False, False,
        False, False, False,
           False, False, False], dtype=bool)
```

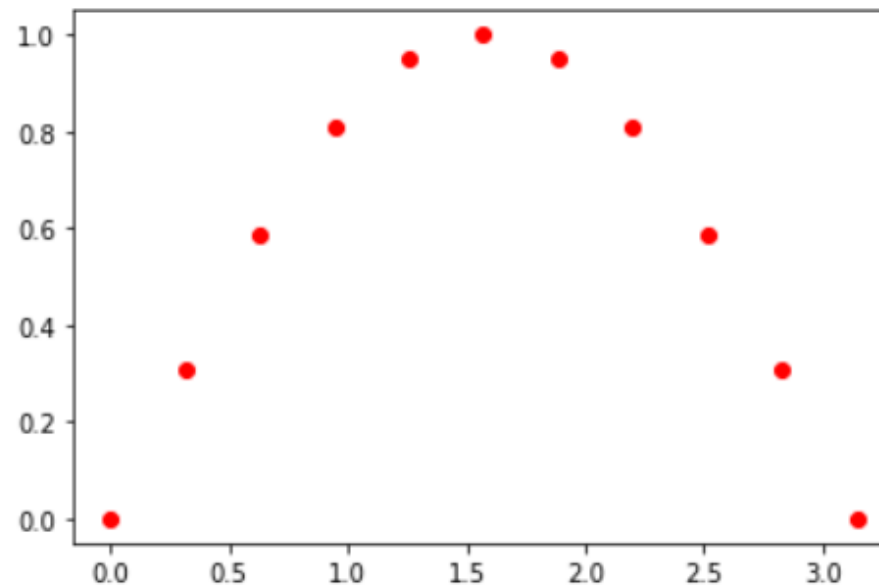


```
mask = b >= 0
```

画出所有对应的非负值对应的点：

```
plot(a[mask], b[mask], 'ro')
```

[<matplotlib.lines.Line2D at 0xa177be0>]



Thank You !

