

## 1、plot简介:

### pyplot 简介

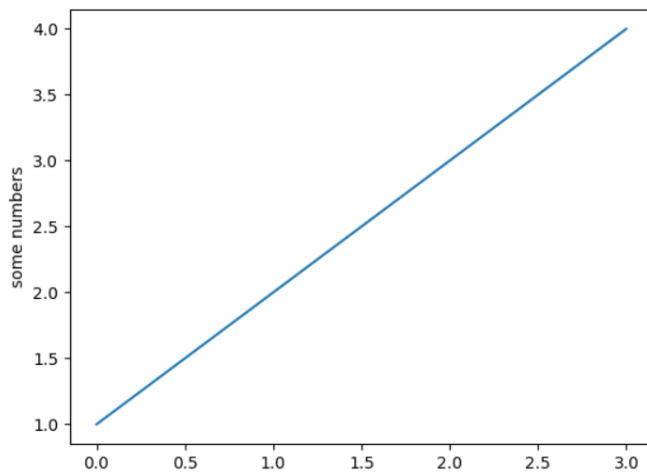
matplotlib.pyplot 是命令样式函数的集合，使matplotlib像MATLAB一样工作。每个pyplot函数对图形进行一些更改：例如，创建图形，在图形中创建绘图区域，在绘图区域中绘制一些线条，用标签装饰图形等。

在matplotlib.pyplot中，各种状态在函数调用中保留，以便跟踪当前图形和绘图区域等内容，并且绘图函数指向当前轴（请注意“轴”在此处以及在大多数位置 文档是指图形的轴部分，而不是多个轴的严格数学术语。

**注意:** pyplot API通常不如面向对象的API灵活。您在此处看到的大多数函数调用也可以作为Axes对象中的方法调用。我们建议您浏览教程和示例以了解其工作原理。

使用pyplot生成可视化非常快速：

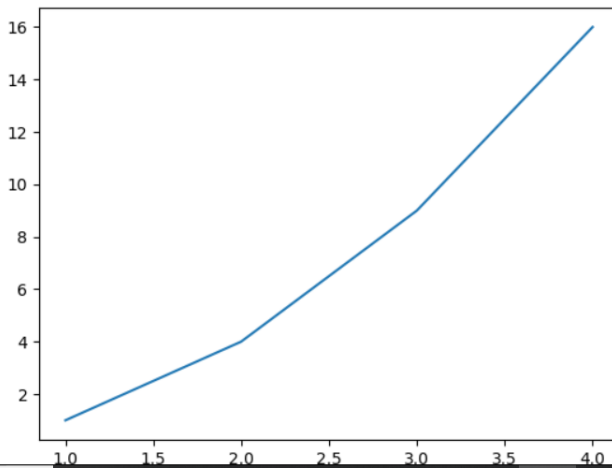
```
import matplotlib.pyplot as plt
plt.plot([1, 2, 3, 4])
plt.ylabel('some numbers')
plt.show()
```



您可能想知道为什么x轴的范围是0-3，y轴的范围是1-4。如果为plot()命令提供单个列表或数组，则matplotlib假定它是一系列y值，并自动为您生成x值。由于python范围以0开头，因此默认的x向量与y具有相同的长度，但从0开始。因此x数据为 `[0,1,2,3]`。

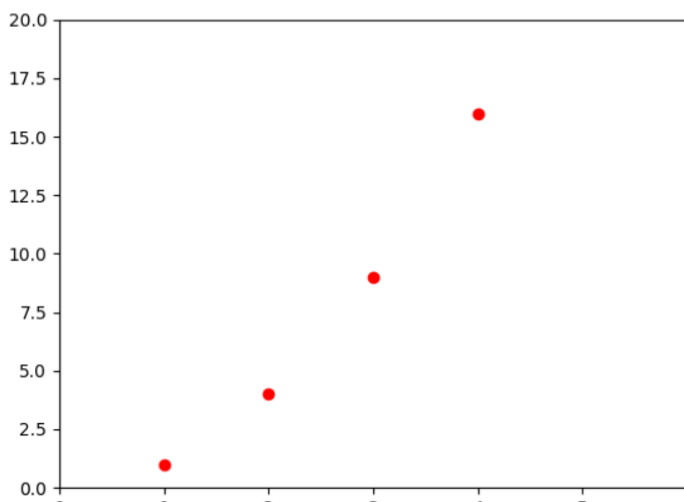
plot() 是一个多功能命令，将采用任意数量的参数。例如，要绘制x与y的关系，您可以发出命令：

```
plt.plot([1, 2, 3, 4], [1, 4, 9, 16])
```



要用红色圆圈绘制上述内容，您将发出：

```
plt.plot([1, 2, 3, 4], [1, 4, 9, 16], 'ro')  
plt.axis([0, 6, 0, 20])  
plt.show()
```



有关线型和格式字符串的完整列表，请参阅 plot() 文档。上例中的 axis() 命令采用 [xmin, xmax, ymin, ymax] 列表并指定轴的视口。

## 2、plt.figure()的使用：

[https://blog.csdn.net/m0\\_37362454/article/details/81511427](https://blog.csdn.net/m0_37362454/article/details/81511427)

## 3、plt.xticks():

[https://matplotlib.org/api/\\_as\\_gen/matplotlib.pyplot.xticks.html](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.xticks.html)

xticks的主要作用就是将坐标轴设置刻度 并将刻度转化为想要的刻度。

rotation表示刻度名称旋转的角度

## 4、plt.legend()函数的用法:

[https://blog.csdn.net/you\\_are\\_my\\_dream/article/details/53440964](https://blog.csdn.net/you_are_my_dream/article/details/53440964)

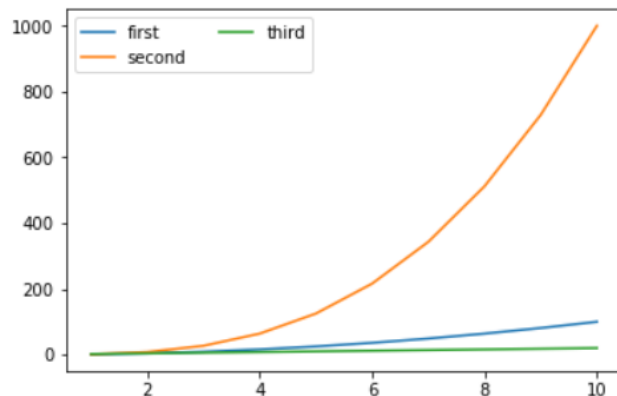
```

1: x=np.arange(1,11,1)

1: label=['first','second','third']

1: plt.plot(x,x**2,label='first')
plt.plot(x,x**3,label='second')
plt.plot(x,2*x,label='third')
plt.legend(loc=0,ncol=2)
plt.show()

```



**legend里的参数:**

参数:

loc(设置图例显示的位置)

```

'best'      : 0, (only implemented for axes legends)(自适应方式)
'upper right': 1,
'upper left' : 2,
'lower left' : 3,
'lower right': 4,
'right'      : 5,
'center left': 6,
'center right': 7,
'lower center': 8,
'upper center': 9,
'center'     : 10,

```

ncol(设置列的数量, 使显示扁平化, 当要表示的线段特别多时会有用)

**ncol的意思是设置标签显示的列数, 上面的loc和ncol参数需要自己写出来。**

## 5、plt.ticke\_params()函数的用法

[https://matplotlib.org/api/\\_as\\_gen/matplotlib.axes.Axes.tick\\_params.html#matplotlib.axes.Axes.tick\\_params](https://matplotlib.org/api/_as_gen/matplotlib.axes.Axes.tick_params.html#matplotlib.axes.Axes.tick_params)

里面有该函数的参数

```

39]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import matplotlib as pl

```

```

]: t=np.arange(0.01,10.0,0.01)
   data1=np.exp(t)
   data2=np.sin(2*np.pi*t)
   fig,ax1=plt.subplots()
   color='red'
   ax1.set_xlabel('time(s)')

```

```

]: <matplotlib.text.Text at 0x17a50edb9b0>

```

```

]: ax1.set_xlabel('exp',color=color)
   ax2=ax1.twinx()
   ax1.plot(data1,t,color=color)
   ax1.tick_params(axis='x',labelcolor=color)

```

```

]: color='green'
   ax2.set_xlabel('sin',color=color)
   ax2.plot(data2,t,color=color)
   ax2.tick_params(axis='x',labelcolor=color)
   fig.tight_layout()

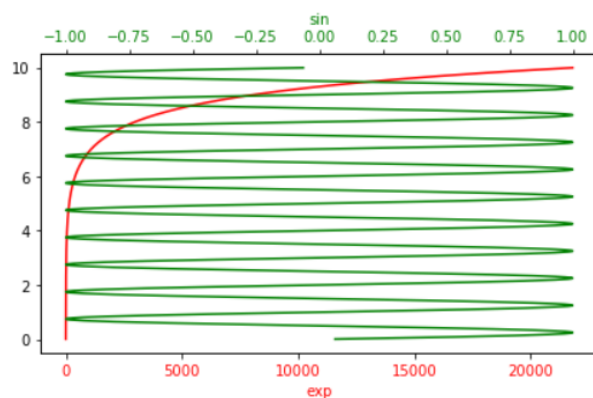
```

```

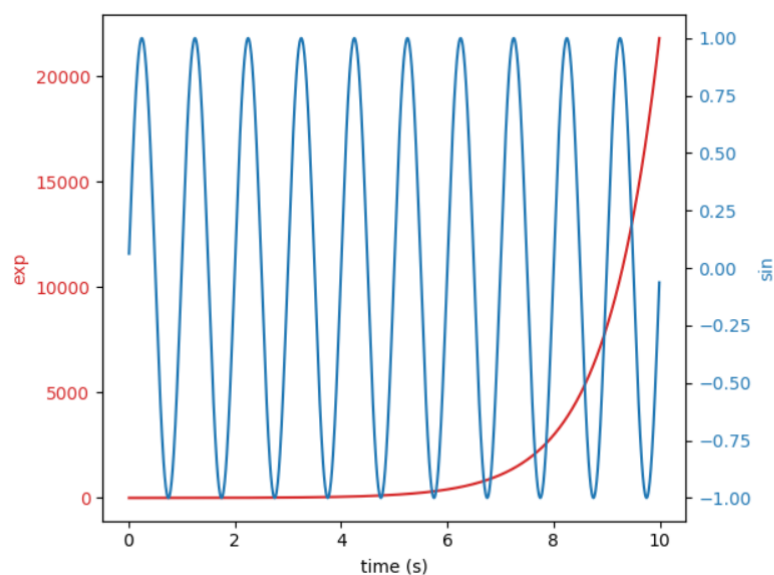
]: plt.show()

```

In [44]: plt.show()



画该图的时候需要用到plt.twinx()函数 其实原图是plt.twinx()函数 是为了了解twinx()函数而改过的；  
这两个函数的具体解释可以去网上找资料看 比较好理解。



这是twinx()函数的结果

## 6、python中的默认颜色：

[https://matplotlib.org/gallery/color/color\\_demo.html#sphx-glr-gallery-color-color-demo-py](https://matplotlib.org/gallery/color/color_demo.html#sphx-glr-gallery-color-color-demo-py)

## 7、颜色教程：

<https://matplotlib.org/tutorials/colors/colors.html>

## 8、plt.setp () :

生，如果循环不已活颜色，则默认刀黑色。

8. 其中一个'是'tab10'分类调色板中的Tableau颜色 (这是默认的颜色循环) ;{'tab:blue', 'tab:orange', 'tab:green', 'tab:red', 'tab:purple', 'tab:brown', 'tab:pink', 'tab:gray', 'tab:olive', 'tab:cyan'}

为画出来的图设置属性。

[https://matplotlib.org/api/\\_as\\_gen/matplotlib.pyplot.setp.html](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.setp.html)

## 9、填充图和线条的实例：

图： [https://www.matplotlib.org.cn/gallery/lines\\_bars\\_and\\_markers/fill.html](https://www.matplotlib.org.cn/gallery/lines_bars_and_markers/fill.html)

线条： [https://www.matplotlib.org.cn/gallery/lines\\_bars\\_and\\_markers/fill\\_between\\_demo.html](https://www.matplotlib.org.cn/gallery/lines_bars_and_markers/fill_between_demo.html)

## 10、自定义虚线样式：

[https://www.matplotlib.org.cn/gallery/lines\\_bars\\_and\\_markers/line\\_demo\\_dash\\_control.html](https://www.matplotlib.org.cn/gallery/lines_bars_and_markers/line_demo_dash_control.html)

## 11、tight\_layout()用来自动调整参数

## 12、在图中的标注可以显示中文：

```
3 plt.rcParams['font.sans-serif']='SimHei'  
4 plt.rcParams['axes.unicode_minus']=False
```

加上这两行就行

```
plt.rcParams['font.sans-serif']='SimHei'
```

```
plt.rcParams['axes.unicode_minus']=False
```