



Matplotlib数据可视化

西安科技大学 牟琦
muqi@xust.edu.cn



6.1 Matplotlib绘图基础

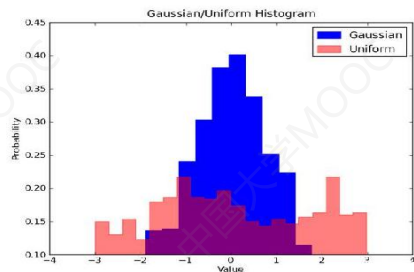
■ 数据可视化

- **数据分析**阶段：理解和洞察数据之间的关系
- **算法调试**阶段：发现问题，优化算法
- **项目总结**阶段：展示项目成果

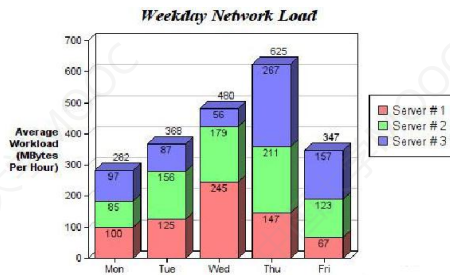
■ **Matplotlib**：第三方库，可以快速方便地生成高质量的图表



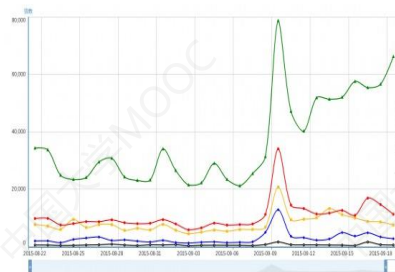
06 Matplotlib绘图库(1)



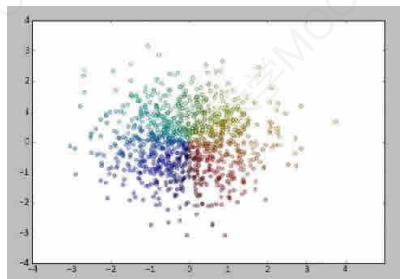
直方图



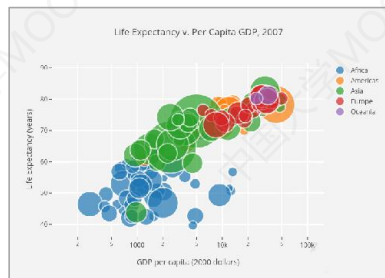
柱形图



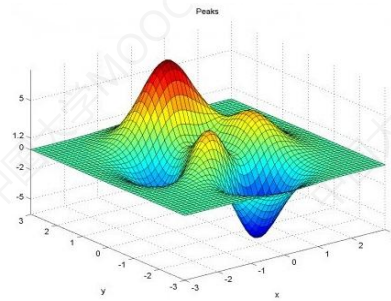
折线图



散点图



气泡图



三维图



■ 安装Matplotlib库

- Anaconda: 安装了anaconda之后, Matplotlib就已经被安装好了
- pip安装

```
pip install matplotlib
```

■ 导入Matplotlib库

```
import matplotlib.pyplot as plt
```



■ Figure 对象：创建画布

```
figure( num,figsize,dpi,facecolor,edgecolor,frameon )
```

- num: 图形**编号**或**名称**，取值为数字/字符串。
- figsize: 绘图对象的**宽**和**高**，单位为英寸。
- dpi: 绘图对象的**分辨率**，缺省值为80。
- facecolor: **背景颜色**。
- edgecolor: **边框颜色**。
- frameon: 表示**是否显示边框**。



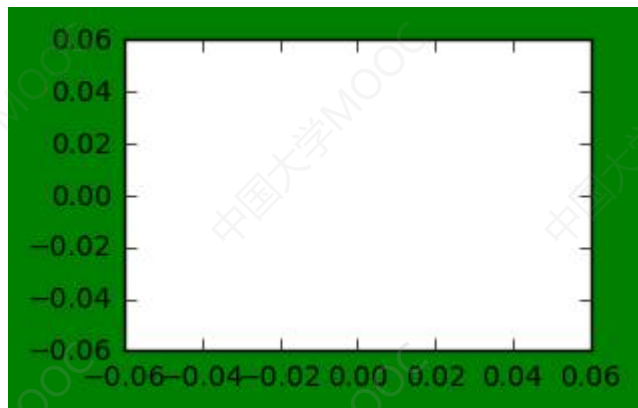
06 Matplotlib绘图库(1)

```
1 import matplotlib.pyplot as plt
2
3 plt.figure(figsize=(3,2),facecolor="green")
4 plt.plot()
5 plt.show()
```

创建画布

绘制空白图形

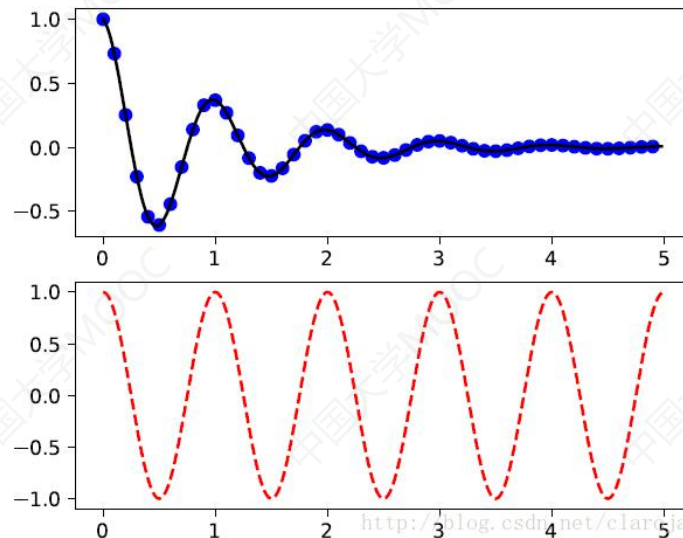
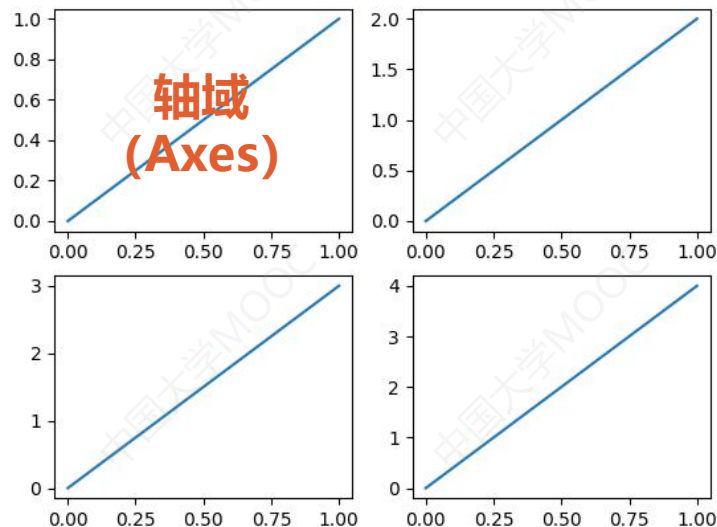
显示绘图



颜色	缩略字符	颜色	缩略字符
blue	b	black	k
green	g	white	w
red	r	cyan	c
yellow	y	magenta	m



□ 划分子图



□ subplot()函数——划分子图

subplot(行数, 列数, 子图序号)

1
2

1	2
3	4

1	2	3
4	5	6

当subplot()函数中的3个参数都**小于10**时，可以省略参数间的逗号，用一个3位数来表示

```
1 plt.subplot(2,2,1)
2 plt.subplot(2,2,2)
3 plt.subplot(2,2,3)
4 plt.subplot(2,2,4)
```

```
1 plt.subplot(221)
2 plt.subplot(222)
3 plt.subplot(223)
4 plt.subplot(224)
```

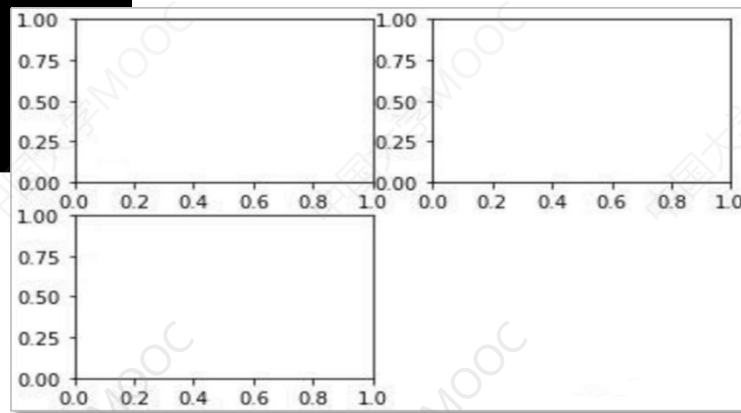
每个subplot()函数只创建一个子图。
要创建4个子图，就需要4条语句



例：将画布划分为 2×2 的子图区域，并绘制3个子图

```
1 import matplotlib.pyplot as plt
2
3 fig = plt.figure()
4
5 plt.subplot(221)
6 plt.subplot(222)
7 plt.subplot(223)
8
9 plt.show()
```

运行结果：



■ 设置中文字体

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

run configuration Params

字体

中文黑体

运行配置参数：指定所绘制图表中的各种**默认属性**，是matplotlib中的**全局变量**

中文字体	英文描述	中文字体	英文描述
宋体	SimSun	楷体	KaiTi
黑体	SimHei	仿宋	FangSong
微软雅黑	Microsoft YaHei	隶书	LiSu
微软正黑体	Microsoft JhengHei	幼圆	YouYuan

□ 恢复标准默认配置

```
plt.rcParamsDefaults( )
```



■ 添加标题

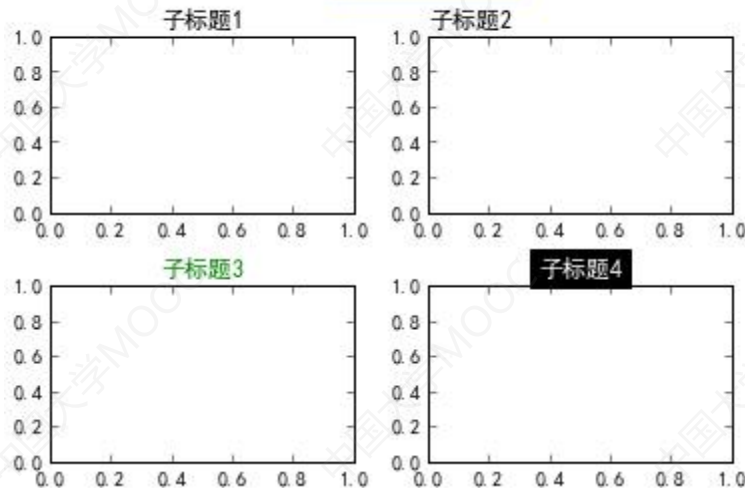
□ 添加全局标题

suptitle (标题文字)

□ 添加子标题

title (标题文字)

全局标题



□ **suptitle()**函数的主要参数

参 数	说 明	默认值
x	标题位置的x坐标	0.5
y	标题位置的y坐标	0.98
color	标题颜色	黑色
backgroundcolor	标题背景颜色	12
fontsize	标题的字体大小	
fontweight	字体粗细	normal
fontstyle	设置字体类型	
horizontalalignment	标题水平对齐方式	center
verticalalignment	标题的垂直对齐方式	top

fontsize:

xx-small
x-small
small
medium
large
x-large
xx-large

fontweight:

light
normal
medium
semibold
bold
heavy
black

normal / italic / oblique

left / right / center

center / top / bottom / baseline



□ title()函数的主要参数

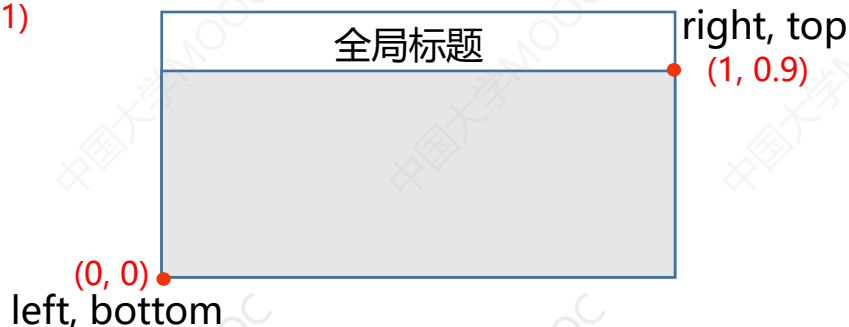
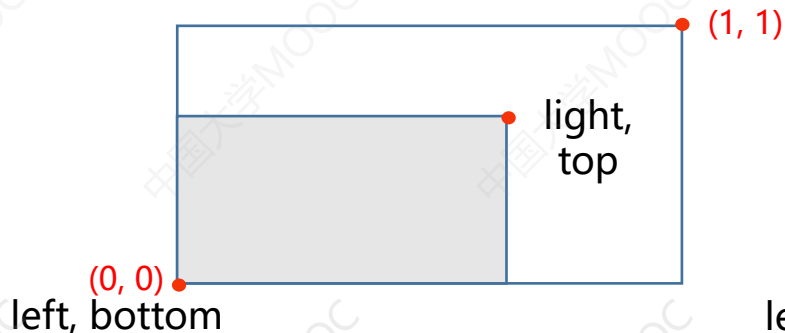
参 数	说 明	取 值
loc	标题位置	left, right
rotation	标题文字旋转角度	
color	标题颜色	黑色
fontsize	标题的字体大小	
fontweight	字体粗细	normal
fontstyle	设置字体类型	
horizontalalignment	标题水平对齐方式	center
verticalalignment	标题的垂直对齐方式	top
fontdict	设置参数字典	



□ tight_layout()函数

检查坐标轴标签、刻度标签、和子图标题，**自动调整子图**，使之**填充整个绘图区域**，并**消除子图之间的重叠**。

```
tight_layout( rect=[left, bottom, right, top])
```



6.1 Matplotlib绘图基础

```
1 import matplotlib.pyplot as plt
2
3 plt.rcParams["font.family"] = "SimHei"
4
5 fig = plt.figure(facecolor="lightgrey")
6
7 plt.subplot(2,2,1)
8 plt.title('子标题1')
9 plt.subplot(2,2,2)
10 plt.title('子标题2',loc="left",color="b")
11 plt.subplot(2,2,3)
12 myfontdict = {"fontsize":12, "color":"g", "rotation":30}
13 plt.title('子标题3',fontdict=myfontdict)
14 plt.subplot(2,2,4)
15 plt.title('子标题4',color='white', backgroundcolor="black")
16
17 plt.suptitle("全局标题", fontsize=20, color="red",backgroundcolor="yellow")
18
19 plt.tight_layout(rect=[0,0,1,0.9])
20
21 plt.show()
```

