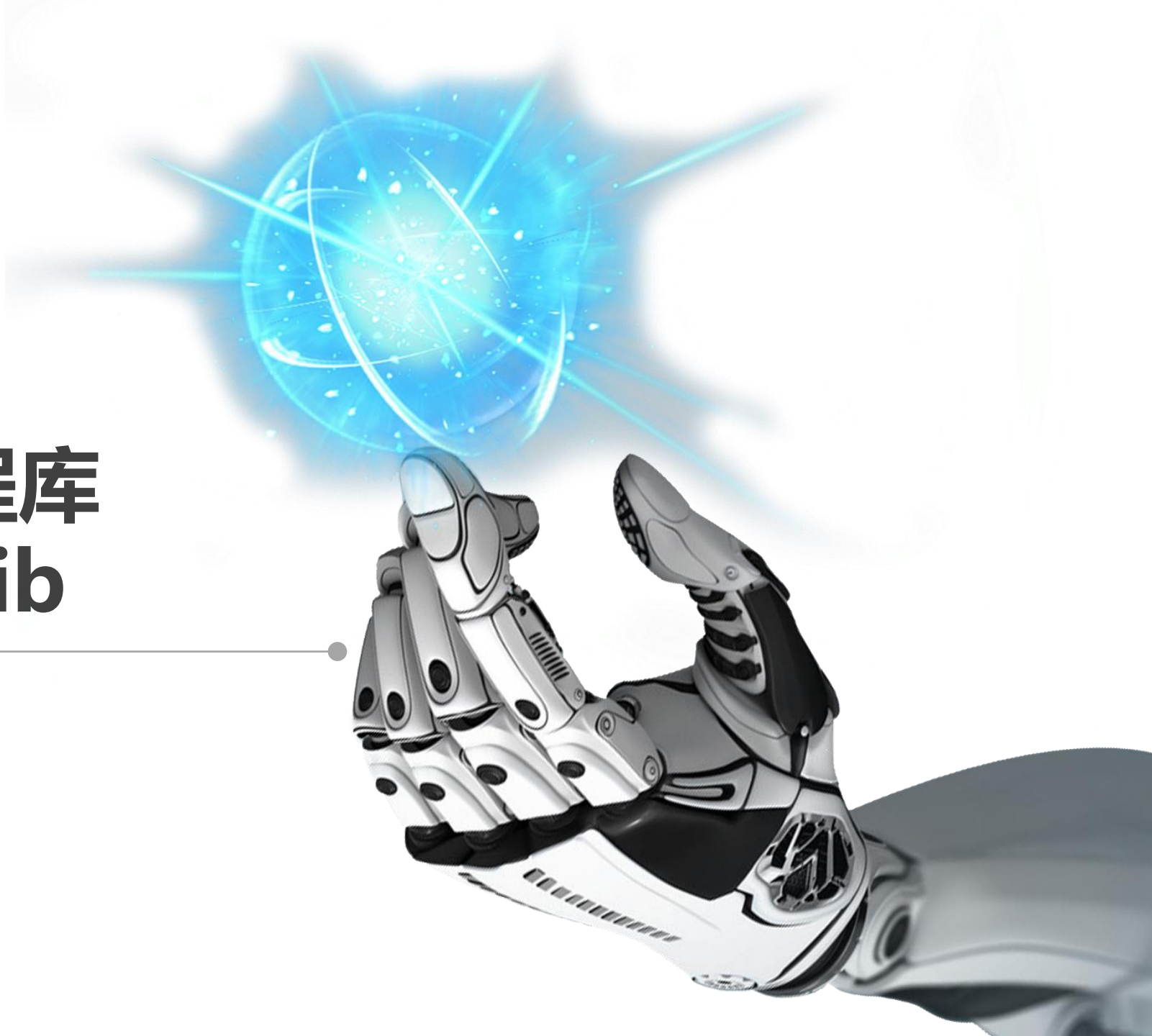


# Python编程库 --Matplotlib

---



# 引入—Numpy

- 数组的创建
- 占位符
- 打印数组
- 基本操作
- 矩阵相乘
- 总和、最小、最大
- axis参数
- 索引、切片、迭代
- 更改数组形状
- 修改数组本身
- 数组堆叠
- 数组拆分

**np.array([1,2,3,4])**

**np.ones((3,4)) np.zeros((3,4)) .....**

**print(np.arange(10000))**

**b=np.arange(4) c=a-b a\*=3**

**A.dot(B) np.dot(A,B)**

**a.sum() a.min() a.max()**

**列b.sum(axis=0) 行b.min(axis=1)**

**a[2:5] a[:6:2]=-1000**

**a.reshape(6,2) a.ravel()**

**a.resize((2,6))**

**np.vstack((a,b)) np.hstack((a,b))**

**np.hsplit(a,3) vsplit**

# 复习

```
import numpy as np
a = np.array([[11, 12, 13, 14, 15],
              [16, 17, 18, 19, 20],
              [21, 22, 23, 24, 25],
              [26, 27, 28, 29, 30],
              [31, 32, 33, 34, 35]])

print(a[2, 4])           #输出“25”
print(a[0, 1:4])        #[12 13 14]
print(a[1:4, 0])        #[16 21 26]
print(a[:, 1])          #[12 17 22 27 32]
```

# 目录

CONTENTS



Matplotlib简介



Matplotlib示例

折线图、条形图、散点图

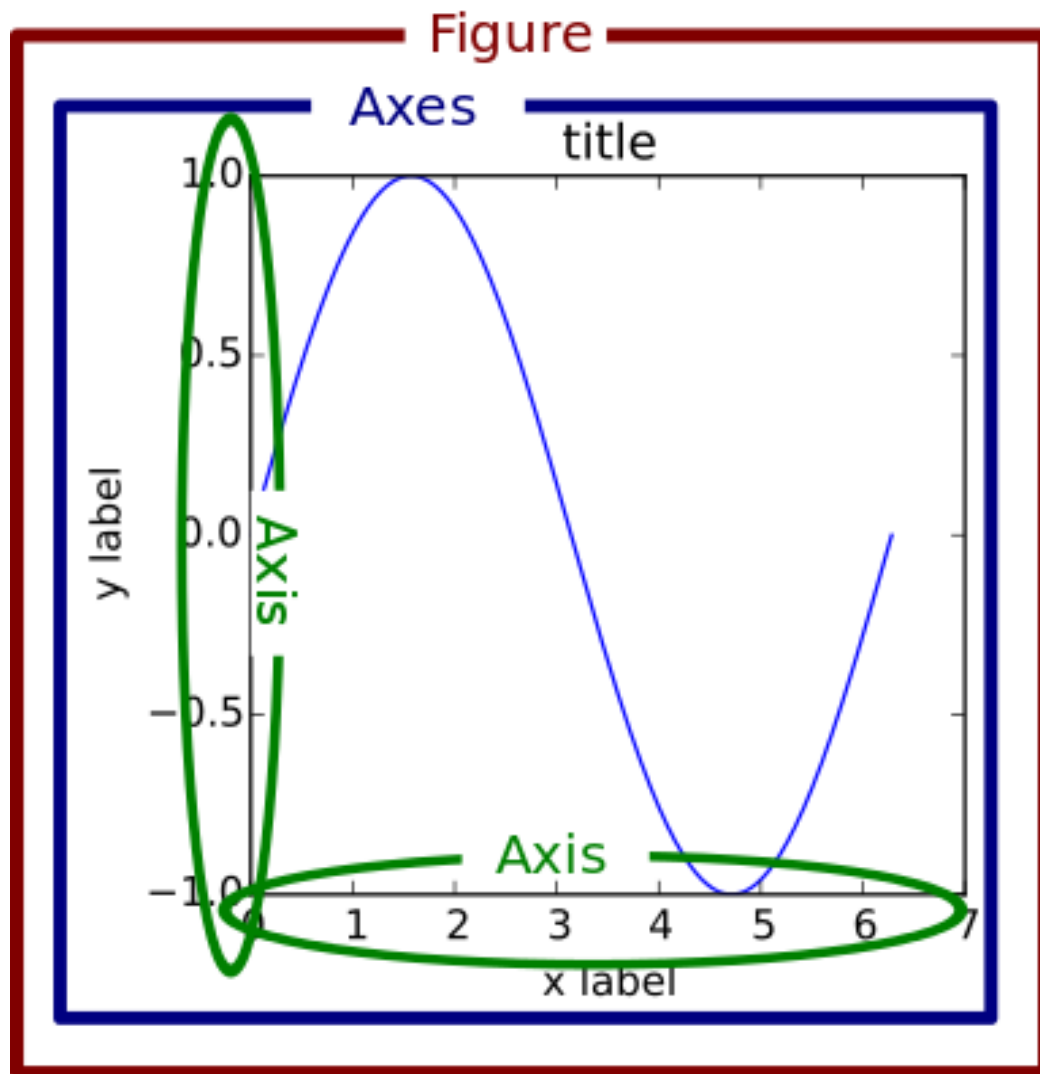
# Matplotlib简介

- 官网: <https://matplotlib.org/>
- 中文文档: <https://www.matplotlib.org.cn>
- Matplotlib是一个 Python 的 2D绘图库。通过Matplotlib, 开发者可以仅需要几行代码, 便可以生成绘图、直方图、功率谱、条形图、错误图、散点图等。

# Matplotlib简介

- 一般步骤:
  - 创建一个图纸 (figure)
  - 在图纸上创建一个或多个绘图 (plotting) 区域
  - 在plotting区域上描绘点、线等各种marker
  - 为plotting添加修饰标签 (绘图线上的或坐标轴上的)
  - 其他各种DIY

# Matplotlib简介



变量  
函数  
图纸 (figure) 和轴域 (axes)

# Matplotlib示例1

```
import matplotlib.pyplot as plt # 调用绘图库
'''figure和axes的声明可以省略(选用默认情况)'''
plt.plot([1, 2, 3, 4])
plt.xlabel('x numbers')
plt.ylabel('y numbers')
plt.title('a straight line')
plt.show()
```

**头脑风暴：**

**为什么y是1-4，而x是0-3？**



# plt.plot

- `plt.plot(*args, scalex=True, scaley=True, data=None, **kwargs)`

## 常用参数及说明：

参数	接收值	说明
<code>x, y</code>	array	表示 x 轴与 y 轴对应的数据；
<code>color</code>	string	表示折线的颜色；
<code>marker</code>	string	表示折线上数据点处的类型；
<code>linestyle</code>	string	表示折线的类型；
<code>linewidth</code>	数值	线条粗细： <code>linewidth=1.=5.=0.3</code>
<code>alpha</code>	0~1之间的小数	表示点的透明度；
<code>label</code>	string	数据图例内容： <code>label= '实际数据'</code>

# Matplotlib示例2

```
import matplotlib.pyplot as plt
plt.plot([1, 2, 3, 4], [1, 4, 9, 16])
plt.axis([0, 6, 0, 20])
plt.show()
```

# Matplotlib 色彩和样式

## 色彩和样式

19

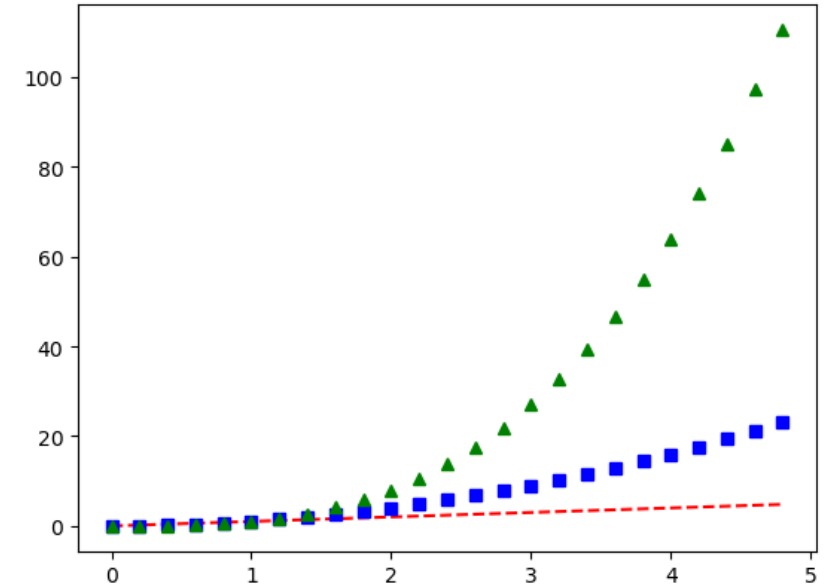
符号	颜色
b	blue
g	green
r	red
c	cyan
m	magenta
Y	yellow
k	black
w	white

线型	描述
'-'	solid
'--'	dashed
'-.'	dash_dot
':'	dotted
'None'	draw nothing
''	draw nothing
''	draw nothing

标记	描述
"o"	circle
"v"	triangle_down
"s"	square
"p"	pentagon
"*"	star
"h"	hexagon1
"+"	plus
"D"	diamond
...	...

# Matplotlib示例3

```
# coding=utf-8  
import numpy as np  
import matplotlib.pyplot as plt  
# 0到5之间每隔0.2取一个数  
t = np.arange(0., 5., 0.2)  
# 红色的破折号, 蓝色的方块, 绿色的三角形  
plt.plot(t, t, 'r--', _____)  
plt.show()
```



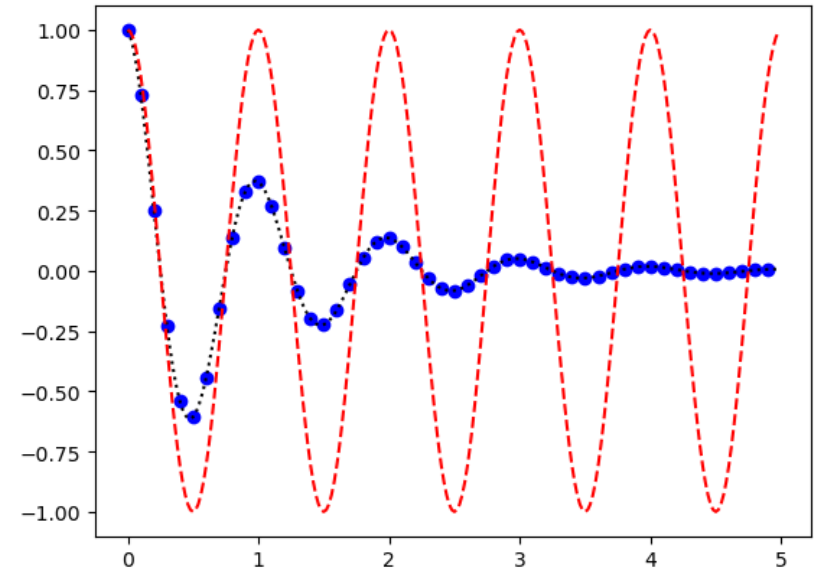
二次方（蓝色方块），三次方（绿色三角形）

```
t, t**2, 'bs', t, t**3, 'g^'
```

# Matplotlib 示例4

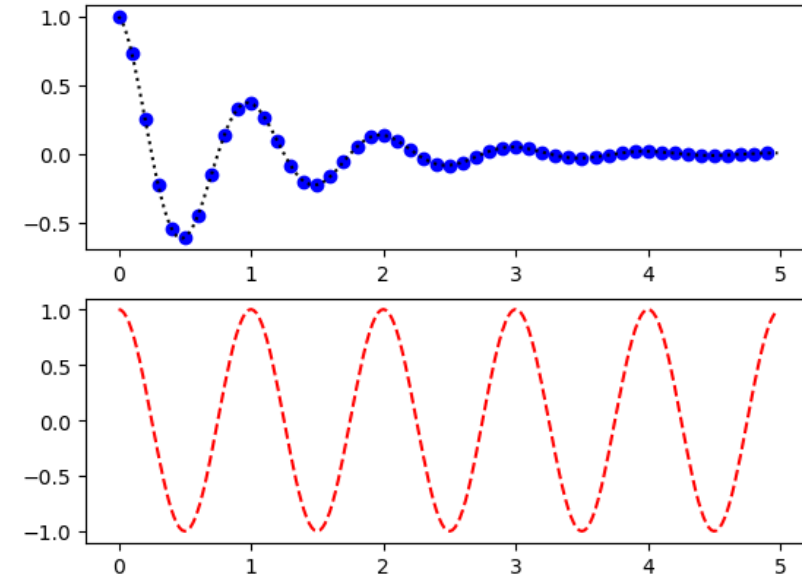
```
import numpy as np
import matplotlib.pyplot as plt
def f(t):
    return np.exp(-t) * np.cos(2*np.pi*t)
t1 = np.arange(0.0, 5.0, 0.1)
t2 = np.arange(0.0, 5.0, 0.02)
plt.plot(t1, f(t1), 'bo', t2, f(t2), 'k:')
plt.plot(t2, np.cos(2*np.pi*t2), 'r--')
plt.show()
```

t1, f(t1)      t2, f(t2)



# Matplotlib 示例5

```
import numpy as np
import matplotlib.pyplot as plt
def f(t):
    return np.exp(-t) * np.cos(2*np.pi*t)
t1 = np.arange(0.0, 5.0, 0.1)
t2 = np.arange(0.0, 5.0, 0.02)
plt.figure("2subplot")
plt.subplot(2, 1, 1)
plt.plot(t1, f(t1), 'bo', t2, f(t2), 'k:')
plt.subplot(2, 1, 2)
plt.plot(t2, np.cos(2*np.pi*t2), 'r--')
plt.show()
```



头脑风暴：使用了哪个函数绘制子图？

subplot()中的参数分别指定了numrows、numcols、**fignum**

若子图数小于10，那么参数之间有没有逗号都无所谓。 1- numrows\*numcols

# Matplotlib示例6

- 加入中文

```
import matplotlib.pyplot as plt
from matplotlib.font_manager import FontProperties
#导入宋体字体文件
font_set = FontProperties(\
fname=r"c:\windows\fonts\simsum.ttc", size=15)
dataX = [1, 2, 3, 4]
dataY = [1, 2, 3, 1]
#plot还有很多参数, 可以查API修改, 如颜色, 虚线等
plt.plot(dataX, dataY)
plt.title("绘制直线", FontProperties=font_set)
plt.xlabel("x轴", FontProperties=font_set)
plt.ylabel("y轴", FontProperties=font_set)
plt.show()
```

# Matplotlib示例7

- 条形图 

```
import matplotlib.pyplot as plt
from matplotlib.font_manager import FontProperties
#导入宋体字体文件
font_set = FontProperties(\
fname=r"c:\windows\fonts\simSun.ttc", size=15)
# 条形图
x = [0, 1, 2, 4, 5, 6]
y = [1, 2, 3, 2.1, 4, 1]
#竖的条形图
plt.bar(x, y)
#plt.barh(x, y)#横的条形图, 注意x, y坐标
plt.title("条形图", FontProperties=font_set)
plt.xlabel("x轴", FontProperties=font_set)
plt.ylabel("y轴", FontProperties=font_set)
plt.show()
```



# Matplotlib

## • 散点图

```
# cc
```

```
import
```

```
# 散
```

```
x =
```

```
y =
```

```
plt.
```

```
plt.
```

```
84.matplotlib.pyplot.scatter(x, y, s=20, c='b', marker='o', cmap=None, norm=None, vmin=None, vmax=None, alpha=None, linewidths=None, verts=N
```

```
one, hold=None, **kwargs)
```

绘制散点图，其中 X 和 Y 是相同长度的数组序列

x, y:	形如 shape (n,) 数组	输入数据
s:	标量或形如 shape (n,) 数组, 可选, 默认: 20	size in points^2
c:	色彩或颜色序列, 可选, 默认	注意 C 不应是一个单一的 RGB 数字或 RGBA 序列, 因为不便区分。C 可以是一个 RGB 或 RGBA 二维行数组
marker:	MarkerStyle, 可选, 默认: 'o'	详情参阅 markers 属性
cmap:	Colormap 可选, 默认: None	Colormap 实例
norm:	Normalize 可选, 默认: None	数据亮度 0-1, float 数据
vmin, vmax:	标量, 可选, 默认: None	亮度设置, 若 norm 实例已使用, 该参数无效
alpha:	标量, 可选, 默认: None	0-1
linewidths:	标量或数组, 默认: None	.....

```
85. matplotlib.pyplot.sci(im)
```

# Matplotlib 示例9

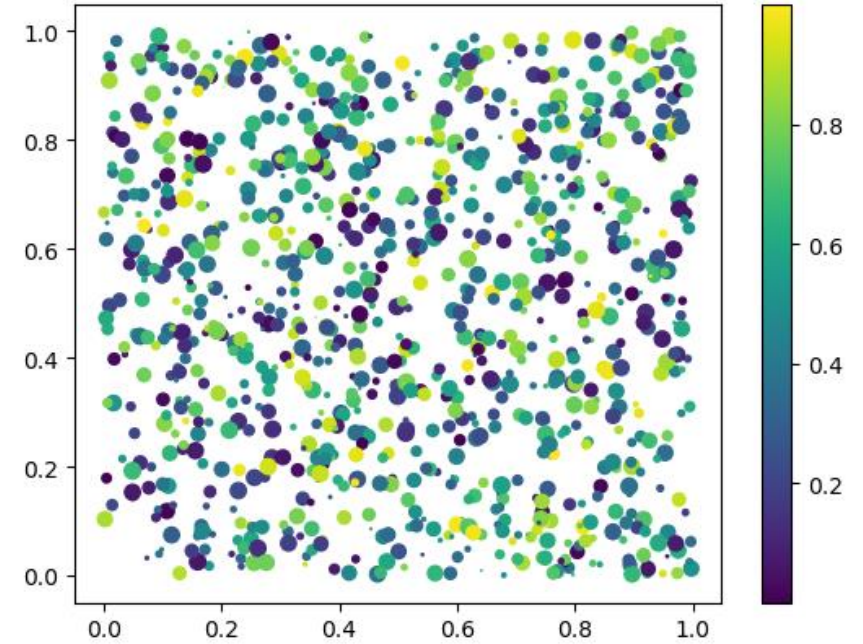
散点图:

```
import matplotlib.pyplot as plt
import numpy as np
# 彩色映射散点图
x = np.random.rand(1000)
y = np.random.rand(1000)
plt.scatter(x, y)
plt.show()
```

# Matplotlib示例10

- 散点图

```
import matplotlib.pyplot as plt
import numpy as np
# 彩色映射散点图
x = np.random.rand(1000)
y = np.random.rand(1000)
size = np.random.rand(1000) * 50
colour = np.random.rand(1000)
plt.scatter(x, y, size, colour)
plt.colorbar()          colorbar颜色渐变色条
plt.show()
```



# 小结

## CONTENTS



Matplotlib简介



Matplotlib示例

**折线图**、条形图、散点图

# 作业3

按照下列数据绘制下图，并将作业3代码提交职教云平台

	1.30	1.31	2.1	2.2	2.3	2.4	2.5	2.6	2.7	2.8
确诊	9720	11821	14411	17238	20471	24363	28060	31211	34598	37251
疑似	15238	17988	19544	21558	23214	23260	24702	26359	27657	28942

