Python与金融数据挖掘(11)

文欣秀

wenxinxiu@ecust.edu.cn

案例分析

英为财情数据显示，4月20日当天，德国宝马汽车公司(BMWG)收盘大跌3.62%，报100.02欧元/股，流通市值蒸发24.21亿欧元，折合人民币约183亿元，有网友戏称这是史上最贵的一杯冰淇淋。

据了解，事发当日网友视频中的冰淇淋分为单球和双球，售价分别为35元和50元。而据相关预测，上海车展总访客量约为100万人，即便是访客全体每个人都拿一份冰淇淋，总价约为3500万元，这和宝马蒸发的183亿元相比，宝马明显是亏大了。有媒体尖锐的指出，这本身是次不错的营销，却因一杯冰淇淋搞砸了。

舆情数据评分系统搭建

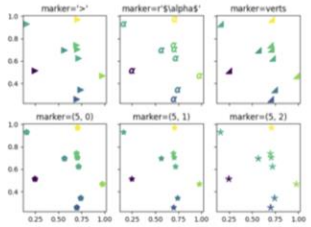
- 创建窗体和控件，用于输入新闻主题
- 编写爬虫模块，用于数据采集和清洗
- 编写舆情分析模块，用于数据的评分
- 编写数据库模块，用于存储统计数据
- 编写绘图模块，用于展示及相关性分析
- …

Matplotlib

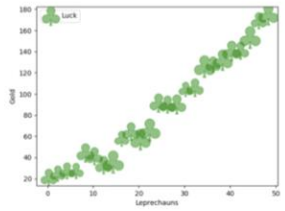
Matplotlib： 一个用来绘制二维图形的Python 模块。它可以绘制多种图形，如**直方图**、**散点图**以及**误差线图**等；可以方便地定制图形的各种属性，如类型、颜色、粗细、字体等，还可以美观地显示图中数学公式。

官网：https://matplotlib.org/

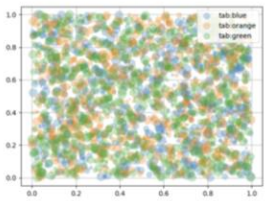
Matplotlib



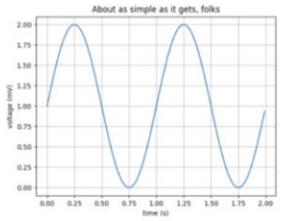
Marker examples



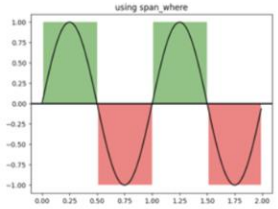
Scatter Symbol



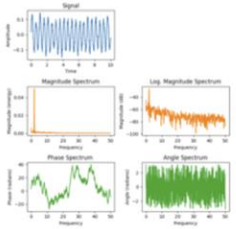
Scatter plots with a legend



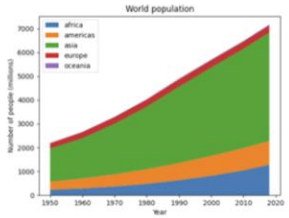
Simple Plot



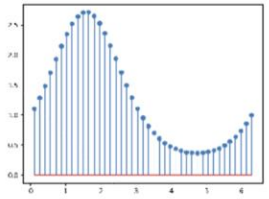
Using span_where



Spectrum Representations



Stackplots and streamgraphs



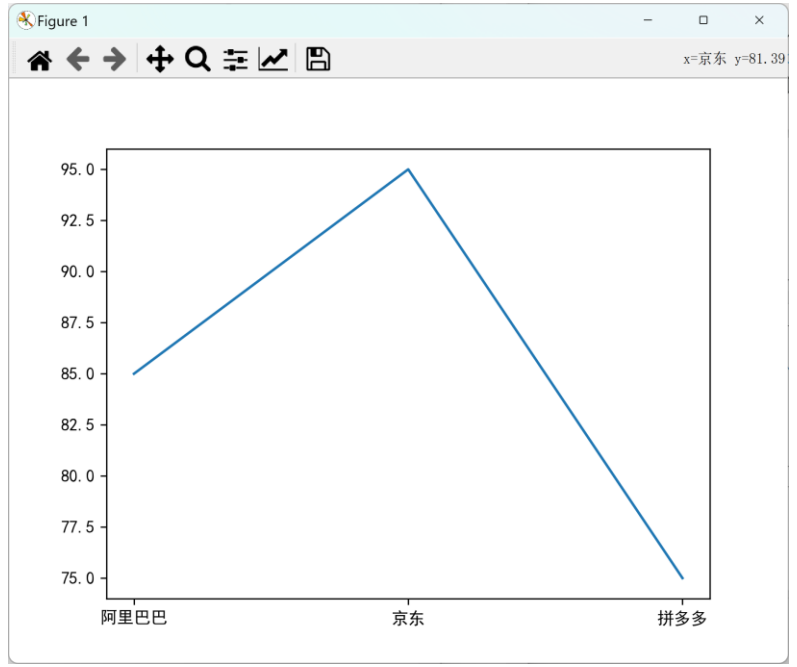
Stem Plot

Matplotlib常用函数

函数名称	函数作用
plot()	绘图折线图
show()	在本机显示图形

绘制折线图

```
import matplotlib.pyplot as plt
name=["阿里巴巴","京东","拼多多"]
grade=[85, 95, 75] #虚构数据仅为举例
plt.rcParams['font.sans-serif']=['SimHei']
plt.plot(name, grade)
plt.show()
```



常用函数及其属性

plt.figure(figsize=(w, h))：创建绘图对象，并设置宽度w和高度h

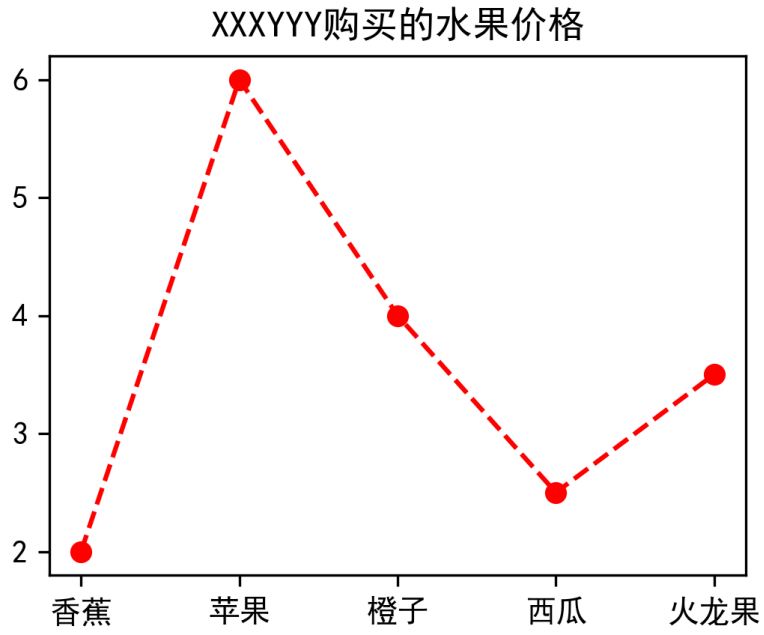
plt.title()：为图表添加标题

plt.plot()参数主要包括：

- 常见的颜色字符：'r'、'g'、'b'、'y'、'w'等
- 常见的线型字符：'-'（直线）、'--'（虚线）、':'（点线）等
- 常用的描点标记：'o'（圆圈）、's'（方块）、'^'（三角形）等

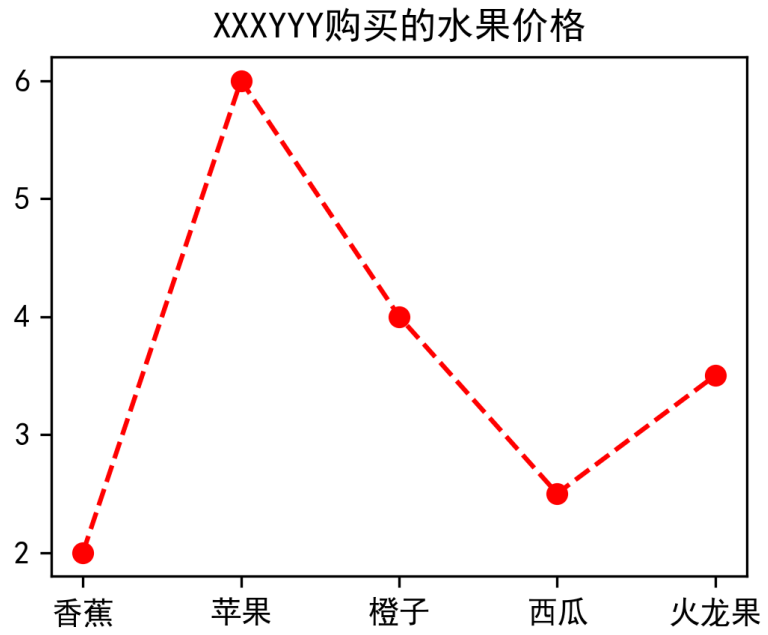
Matplotlib应用案例

编写程序：将你最近购买的水果及价格（或股票及购入价）存在字典中，使用matplotlib绘制出物品价格折线图（要求：窗口大小为4：3，标题中XXXYYY分别为你自己的学号和姓名）。



Matplotlib应用案例

```
import matplotlib.pyplot as plt
plt.figure(figsize=(4,3))
fruits={"香蕉": 2, "苹果": 6, "橙子": 4, "西瓜":2.5, "火龙果": 3.5}
name=list(fruits.keys())
money=list(fruits.values())
plt.plot(name, money,"r--o")
plt.rcParams['font.sans-serif']=['SimHei']
plt.title("XXXYYY购买的水果价格")
plt.show()
```

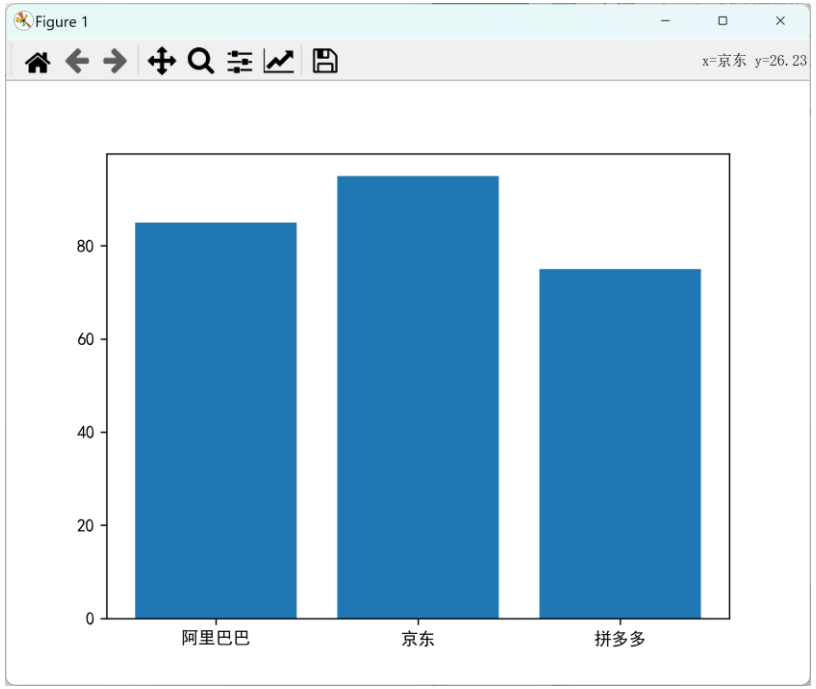


Matplotlib常用函数

函数名称	函数作用
plot()	绘图折线图
show()	在本机显示图形
bar()	绘制垂直条形图
scatter()	绘制散点图

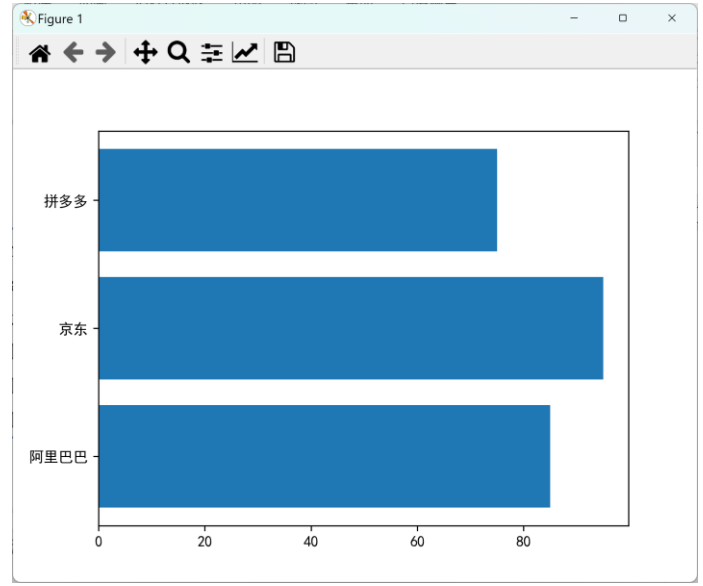
绘制垂直条形图

```
import matplotlib.pyplot as plt
name=["阿里巴巴","京东","拼多多"]
grade=[85, 95, 75]  #虚构数据仅为举例
plt.rcParams['font.sans-serif']=['SimHei']
plt.bar(name, grade)
plt.show()
```



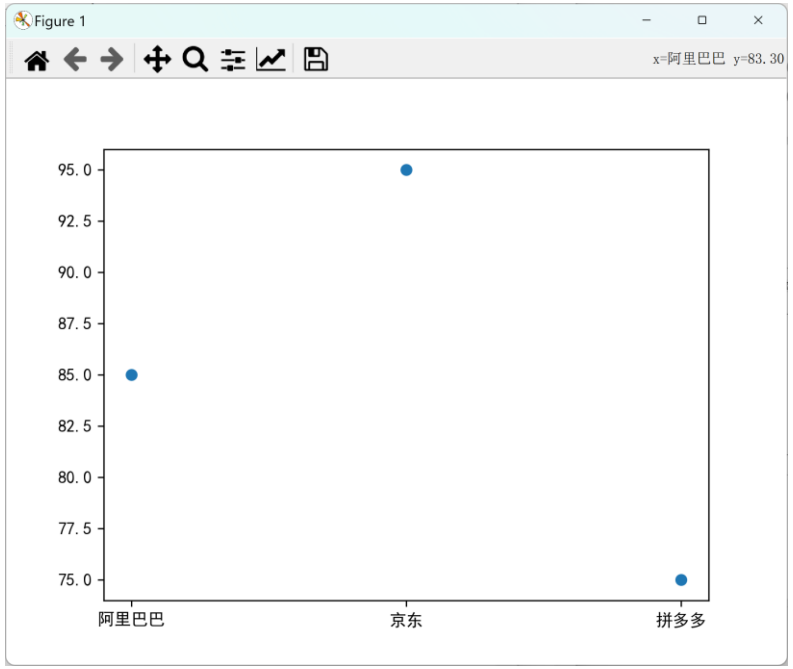
绘制水平条形图

```
import matplotlib.pyplot as plt
name=["阿里巴巴","京东","拼多多"]
grade=[85, 95, 75]  #虚构数据仅为举例
plt.rcParams['font.sans-serif']=['SimHei']
plt.barh(name, grade)
plt.show()
```



绘制散点图

```
import matplotlib.pyplot as plt
name=["阿里巴巴","京东","拼多多"]
grade=[85, 95, 75] #虚构数据仅为举例
plt.rcParams['font.sans-serif']=['SimHei']
plt.scatter(name, grade)
plt.show()
```

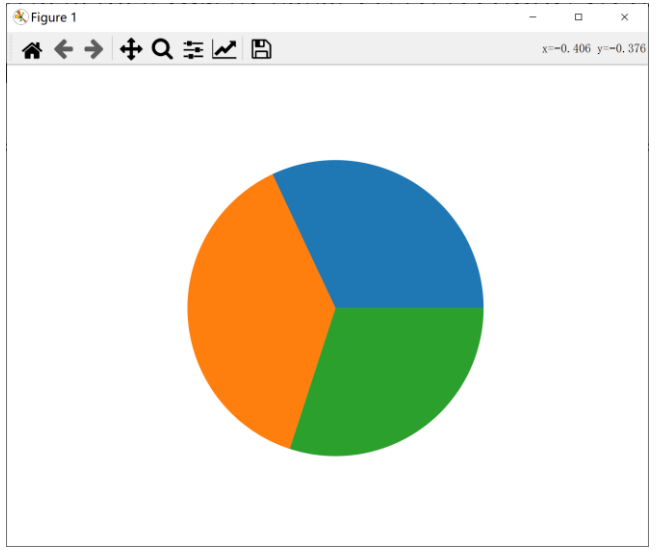


Matplotlib常用函数

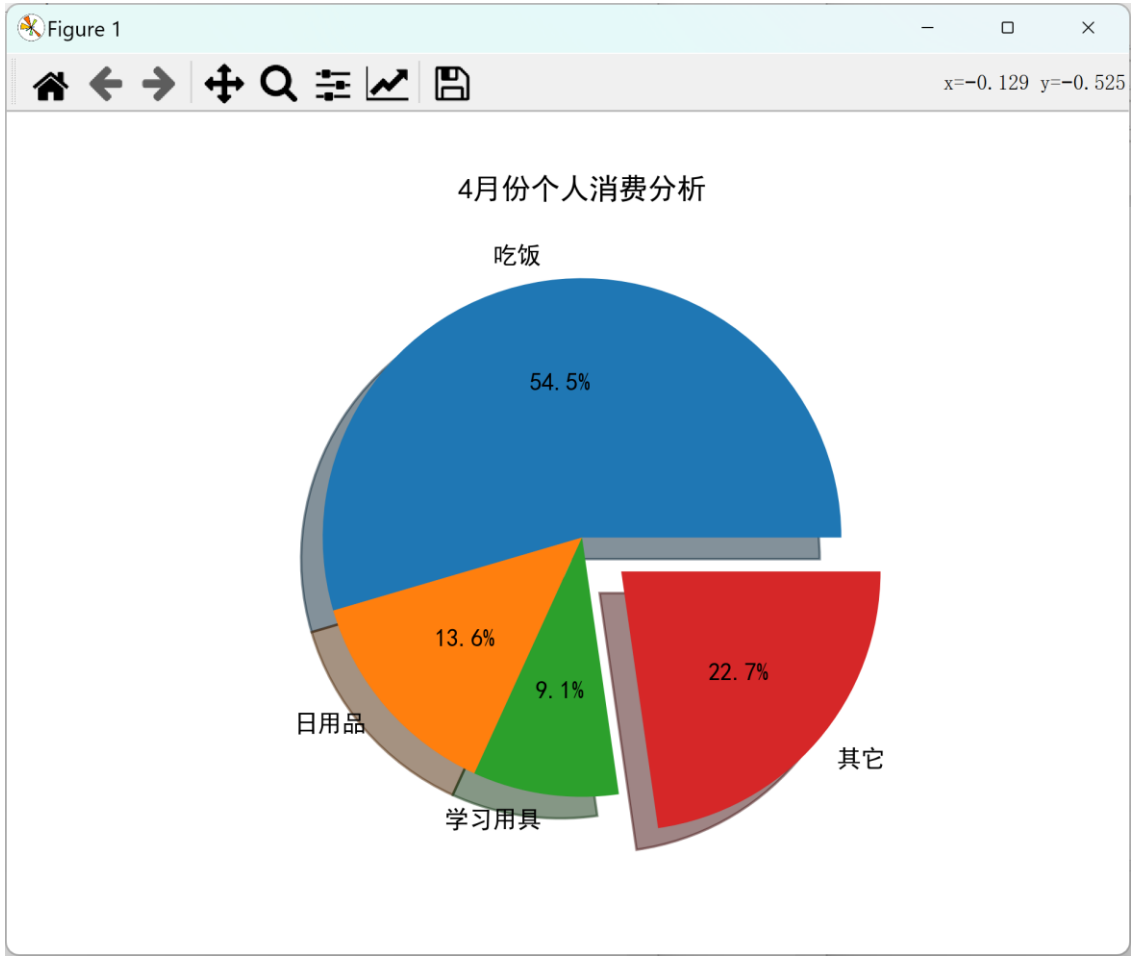
函数名称	函数作用
plot()	绘图折线图
show()	在本机显示图形
bar()	绘制垂直条形图
scatter()	绘制散点图
pie()	绘制饼图

绘制饼图

```
import matplotlib.pyplot as plt
score=[85, 95, 75]
plt.pie(score)
plt.show()
```

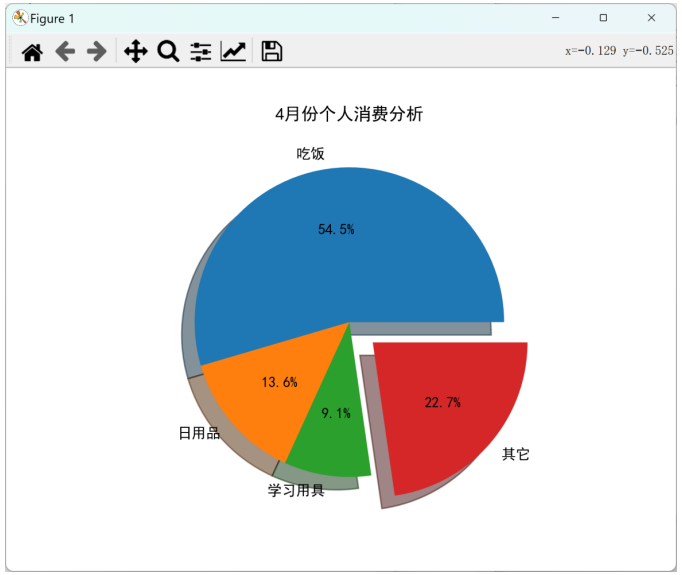


制作个人消费饼图



制作个人消费饼图

```
import matplotlib.pyplot as plt
plt.rcParams['font.sans-serif']=['SimHei']
labels = ['吃饭','日用品','学习用具','其它']
sizes = [1200,300,200,500]
explodes = (0,0,0,0.2)
plt.pie(sizes,explode=explodes,labels=labels,
 autopct='%.1f%%', shadow=True)
plt.title("4月份个人消费分析")
plt.show()
```



Matplotlib常用函数

函数名称	函数作用
plot()	绘图折线图
show()	在本机显示图形
bar()	绘制垂直条形图
scatter()	绘制散点图
pie()	绘制饼图
subplot()	绘制子图

个人消费对比分析

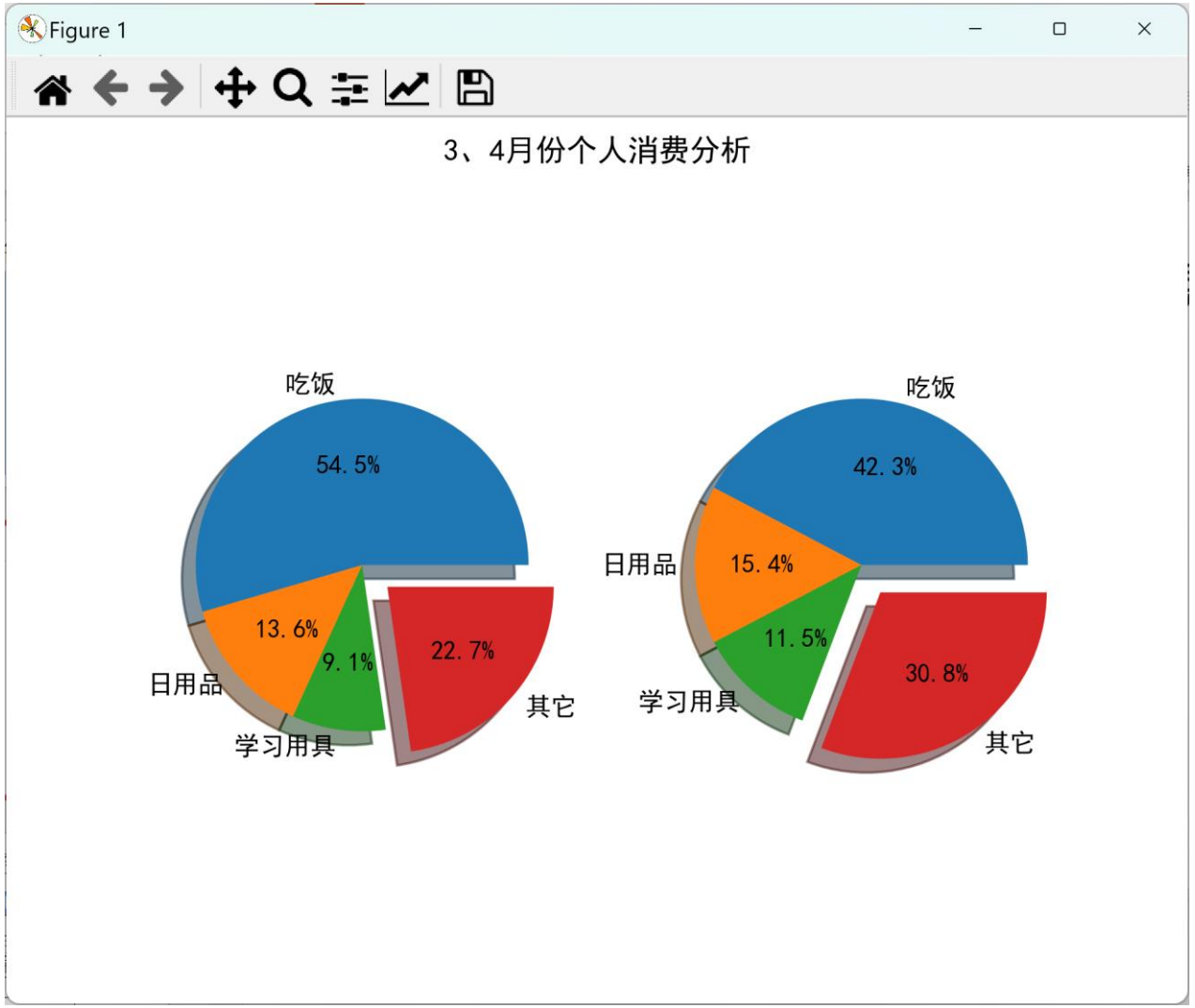
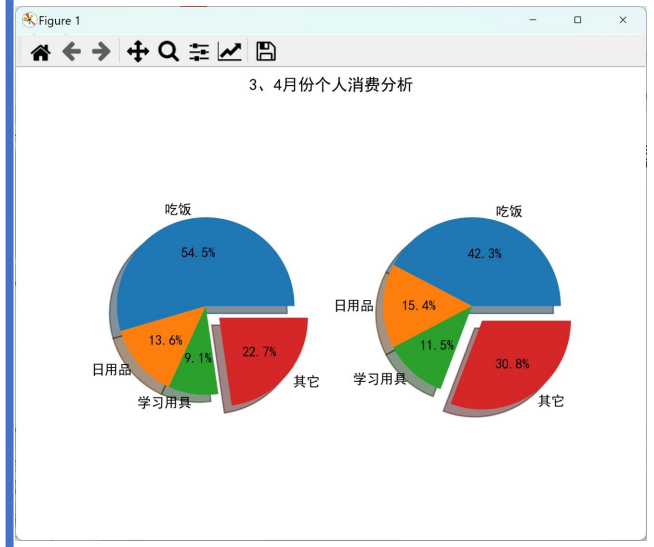


Figure 1
3、4月份个人消费分析
吃饭
54. 5%
13. 6%
9. 1%
22. 7%
日用品
学习用具
其它
吃饭
42. 3%
日用品
15. 4%
11. 5%
30. 8%
学习用具
其它

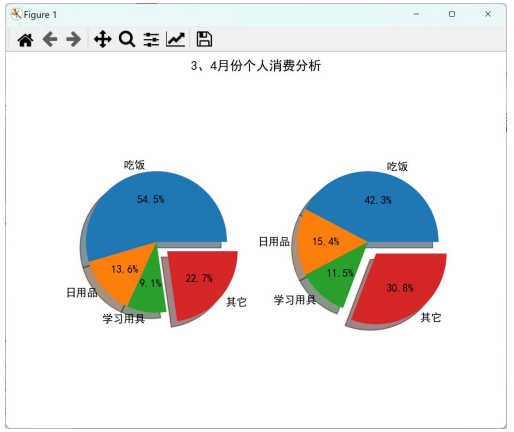
个人消费对比分析

```
import matplotlib.pyplot as plt
plt.rcParams['font.sans-serif']=['SimHei']
p1=plt.subplot(121)
p2=plt.subplot(122)
labels = ['吃饭','日用品','学习用具','其它']
sizes1 = [1200,300,200,500]
sizes2 = [1100,400,300,800]
```



个人消费对比分析

```
explodes = (0,0,0,0.2)
p1.pie(sizes1,explode=explodes,labels=labels,
 autopct='%1.1f%%', shadow=True)
p2.pie(sizes2,explode=explodes,labels=labels,
 autopct='%1.1f%%', shadow=True)
plt.suptitle("3、4月份个人消费分析")
plt.show()
```



Matplotlib常用函数

函数名称	函数作用
plot()	绘图折线图
show()	在本机显示图形
bar()	绘制垂直条形图
scatter()	绘制散点图
pie()	绘制饼图
subplot()	绘制子图
hist()	绘制直方图

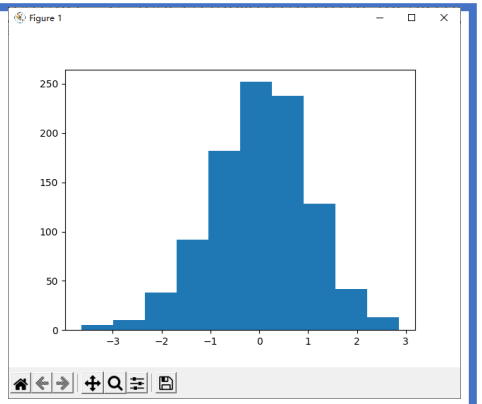
绘制直方图

```
import matplotlib.pyplot as plt

import numpy as np

plt.hist(x=np. random. normal(size=1000))#正态分布

plt.show()
```



24

Numpy

NumPy(Numerical Python的缩写)：是一个开源的Python科学计算库，NumPy数组在数值运算方面的效率优于列表。它是数据分析、机器学习和科学计算的主力军。

官网：**https://numpy.org/doc/stable/**

创建Numpy数组

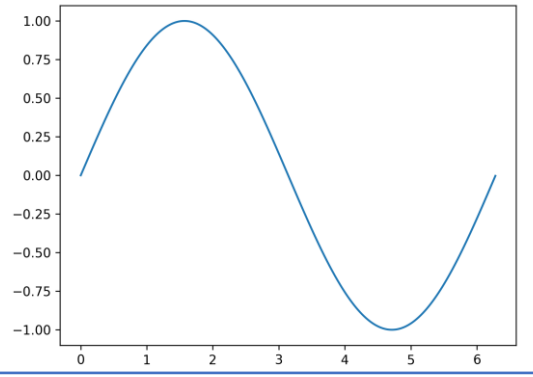
```
>>> import numpy as np          #一般以np作为别名
>>> score=np.array([80,91,78])      # 创建一维数组
>>> print(score+5)
>>> b = np.array([[10,5],[30,6]])    # 创建二维数组
>>> print(b*b)
```

Numpy重要函数

```
>>> import numpy as np
>>> a = np. arange(0,10, 0.1)          #[0, 10)，步长为0.1
>>> b = np. linspace(0,10,100)         #[0,10]，分成100份
>>> c=a. reshape(20,5)                 #变为20行5列
>>> result=a. reshape(-1,1)            #变成1列
>>> test=result. flatten()  #返回一个折叠成一维的数组
```

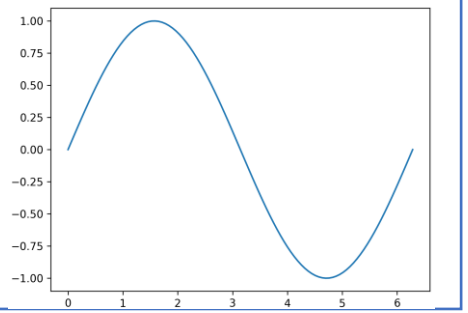
Numpy绘制函数图

```
import matplotlib.pyplot as plt
import numpy as np
x=np.arange(0,2*np.pi,0.01)  #x从0到2π，步长0.01
y=np.sin(x)
plt.plot(x,y)
plt.show()
```



Numpy绘制函数图

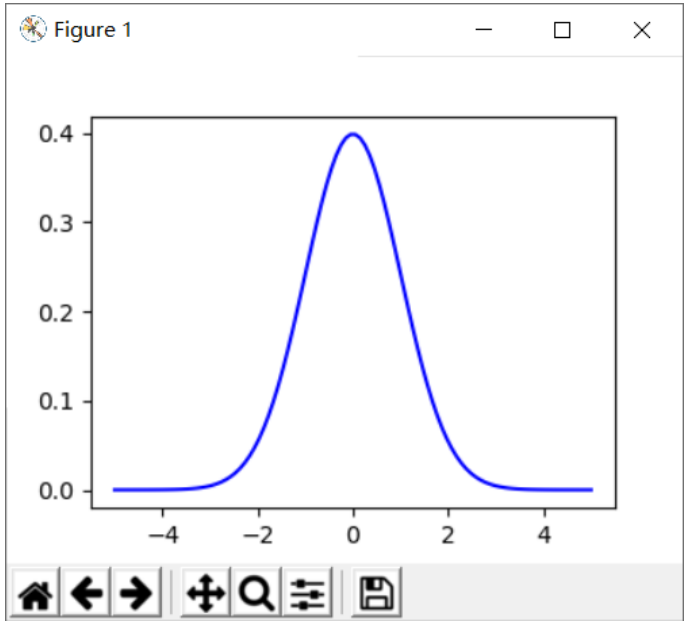
```
import matplotlib.pyplot as plt
import numpy as np
x=np.linspace(0,2*np.pi,100)  #x从0到2π分成100份
y=np.sin(x)
plt.plot(x,y)
plt.show()
```



思考题

编写程序，绘制正态分布的密度函数：$f(x)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$

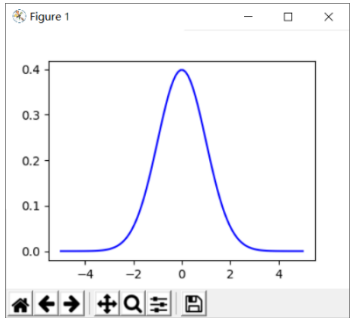
其中： $\mu=0,\sigma=1$ $x\in[-5,5]$



30

正态分布密度函数

```
import matplotlib.pyplot as plt
from numpy import *
plt.figure(figsize=(4,3))
x=linspace(-5,5,100)   #x从-5到5分成100份
y=(1/(sqrt(2*pi)))*exp(-(x*x)/2)
plt.plot(x,y,'-b')
plt.show()
```



Numpy元素取值

```
>>> import numpy as np
>>> a = np. arange(10). reshape(2,5)
>>> a[0]                                  #打印第1行
>>> a[1][2]或者a[1, 2]                    #打印第2行第3列
>>> a[: , 1]                              #打印第2列
>>> a[: , [1,3]]                          #打印第2、4列
```

随机整数

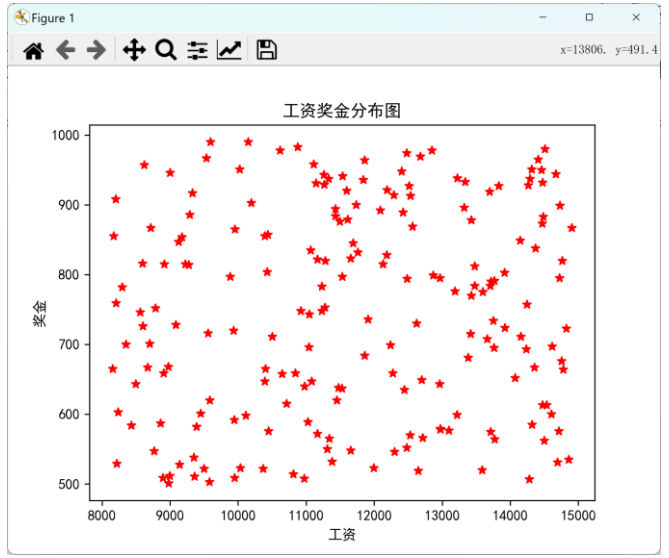
numpy.random. randint(low, high, size, dtype=int)：返回范围为[low, high)随机整数，size为数组尺寸

```
>>> import numpy as np
>>> one=np. random. randint(2) # 产生1个[0,2)之间随机整数
>>> grade=np. random. randint(1,5,size=10) # 产生10个[1,5)之间随机整数
>>> salary=np. random. randint(2000,3000,size=(2,4)) #2行4列
```

工资奖金散点图

```
import numpy as np
import matplotlib.pyplot as plt
plt.rcParams['font.family']=['SimHei']
salary=np. random. randint(8000,15000,size=200)
bonus=np. random. randint(500,1000,size=200)
plt.scatter(salary,bonus,c="r",marker="*")
plt.xlabel("工资")
plt.ylabel("奖金")
plt.title('工资奖金分布图')
plt.show()
```

如何产生浮点数工资及奖金?



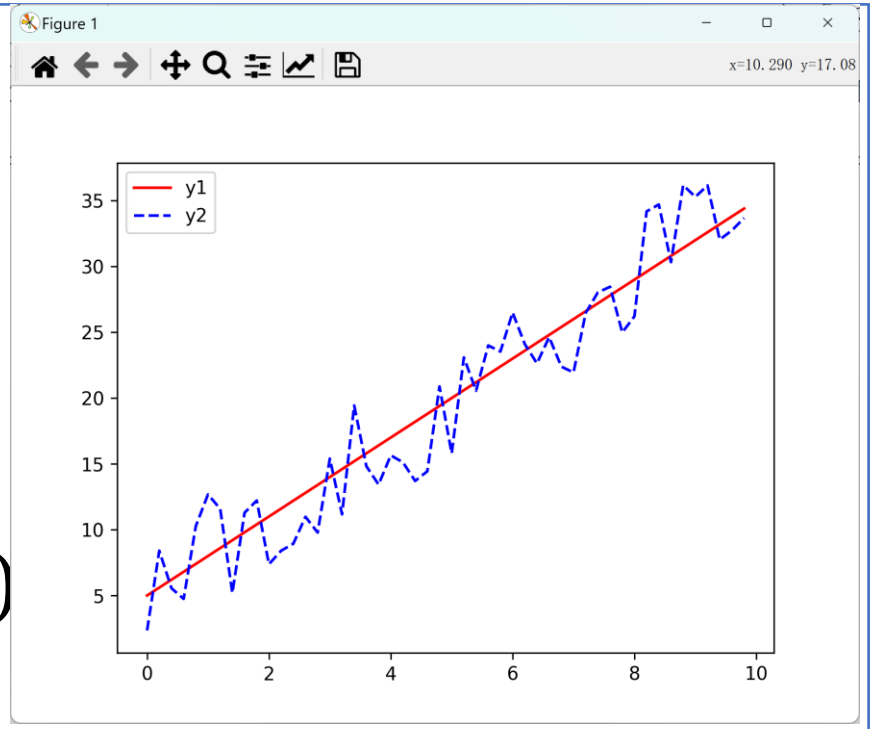
随机浮点数

numpy.random.uniform(low,high,size)： 从一个均匀分布[low,high)中随机采样，size为样本数目

```
>>> import numpy as np
>>> test=np. random. uniform()   # 产生1个[0,1)之间随机浮点数
>>> score= np. random. uniform(0, 100, size=3) #产生 3个0-99的随机浮点数
>>> s= np. random. uniform(200,300,size=(2 ,4)) #产生2行4列200-299的浮点数
```

案例分析

```
import numpy as np
import matplotlib.pyplot as plt
x=np.arange(0,10,0.2)
y1=3*x+5; y2=[]
for i in y1:
    y2.append(i+np.random.uniform(-5,5))
plt.plot(x,y1,"r-",label='y1')
plt.plot(x,y2,"b--",label='y2')
plt.legend(loc='upper left')
plt.show()
```



如何将数据存入文件中？

Numpy数据存储

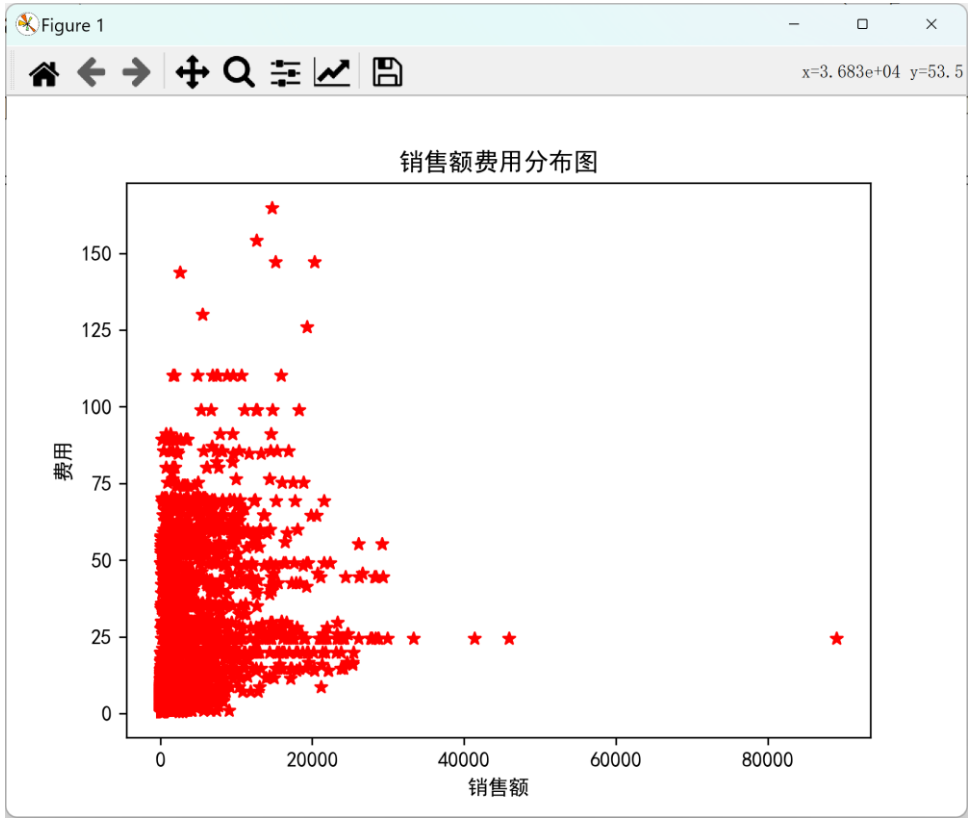
	A	B	C	D	E	F	G	H	I	J
1	5	5.6	6.2	6.8	7.4	8	8.6	9.2	9.8	10.4
2	5.2	9.3	5.4	10.2	2.5	12.3	9	12	9.4	12.1

```
import numpy as np
import matplotlib.pyplot as plt
x=np.arange(0,10,0.2)
y1=3*x+5; y2=[]
for i in y1:
    y2.append(i+np.random.uniform(-5,5))
c=[a,b]
np.savetxt("result.csv",c,fmt='%.1f',delimiter=',', newline='\n')
```

思　　考

如何从文件中读取销售额和费用并绘制图形？

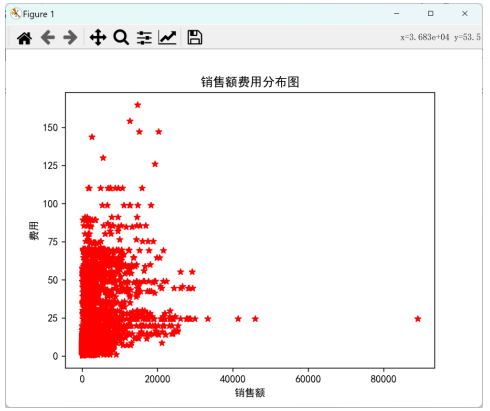
	A	B
1	261.54	35
2	6	2.56
3	2808.08	5.81
4	1761.4	89.3
5	160.2335	5.03
6	140.56	8.99
7	288.56	2.25
8	1892.848	8.99
9	2484.7455	4.2
10	3812.73	1.99
11	108.15	0.7
12	1186.06	3.92
13	51.53	0.7
14	90.05	2.58
15	7804.53	5.99



销售额与费用散点图

```
import numpy as np
import matplotlib.pyplot as plt
plt.rcParams['font.family']=['SimHei']
result=np.loadtxt("trade.csv",delimiter=",").reshape((-1,2))
money=result[:,0]
cost=result[:,1]
plt.scatter(money,cost,c="r",marker="*")
plt.xlabel("销售额")
plt.ylabel("费用")
plt.title('销售额费用分布图')
plt.show()
```

	A	B
1	261.54	35
2	6	2.56
3	2808.08	5.81
4	1761.4	89.3
5	160.2335	5.03
6	140.56	8.99
7	288.56	2.25
8	1892.848	8.99
9	2484.7455	4.2
10	3812.73	1.99
11	108.15	0.7
12	1186.06	3.92
13	51.53	0.7
14	90.05	2.58
15	7804.53	5.99



谢　谢