

# Python 应用程序设计

黄炜钦



5

## 爬虫05 图表展示



## 一、 matplotlib 绘制图表

1.1、 折线图绘制

1.2、 柱状图绘制

1.3、 饼状图绘制

## 二、 wordcloud 绘制词云



# 一、matplotlib 绘制图表

## ■ Matplotlib 简介

matplotlib 是 Python 的绘图库。Matplotlib 的安装如下：

安装命令： pip install matplotlib

或者： pip install -i https://pypi.tuna.tsinghua.edu.cn/simple Matplotlib

```
(venv) F:\DemoProjects\Python>pip install matplotlib  
Collecting matplotlib
```

```
Installing collected packages: six, cycler, pyparsing, numpy, kiwiso  
Successfully installed cycler-0.10.0 kiwisolver-1.1.0 matplotlib-3.1
```



# 一、matplotlib 绘制图表

## ■ NumPy 简介

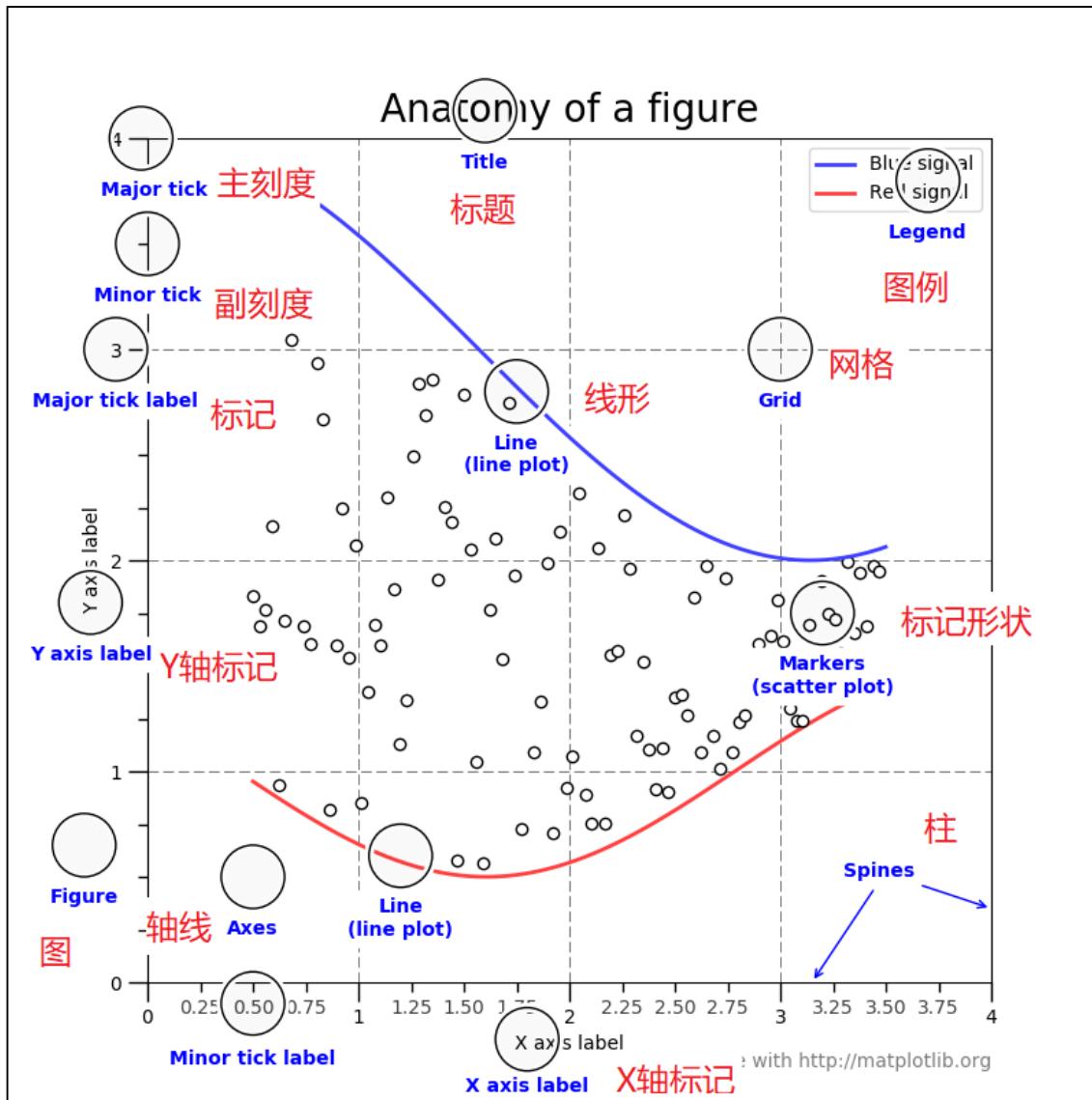
NumPy 通常与 Matplotlib（绘图库）一起使用，这种组合广泛用于替代 MatLab，是一个强大的科学计算环境，有助于我们通过 Python 学习数据科学或者机器学习。

NumPy(Numerical Python) 是 Python 语言的一个扩展程序库，支持大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。

pip install NumPy

pip install -i <https://pypi.tuna.tsinghua.edu.cn/simple> NumPy

# 一、matplotlib 绘制图表





# 一、matplotlib 绘制图表

plt.figure(): 自定义窗口（名字，大小，背景颜色等）

plt.subplot(): 设置画布划分以及图像在画布上输出的位置

plt.axis([xmin, xmax, ymin, ymax]) # 设置x, y轴刻度的范围

plt.xticks([0,1,2]): 设置x轴刻度的显示

plt.xlim(0,1000) # 设置x轴刻度范围，从0~1000

plt.ylim(0,20) # 设置y轴刻度的范围，从0~20

plt.xlabel('x-value') # 设置x轴标记

plt.ylabel('y=2\*x+1')

plt.title('y=2\*x +1 直线') #添加标题

plt.plot(x,y,color='red',linewidth=2.5,linestyle='-', label="cos") #  
color设置颜色， linewidth宽度， linestyle风格， label图例显示名

plt.legend(loc='upper left') # 图例显示位置



# 一、matplotlib 绘制图表 折线图

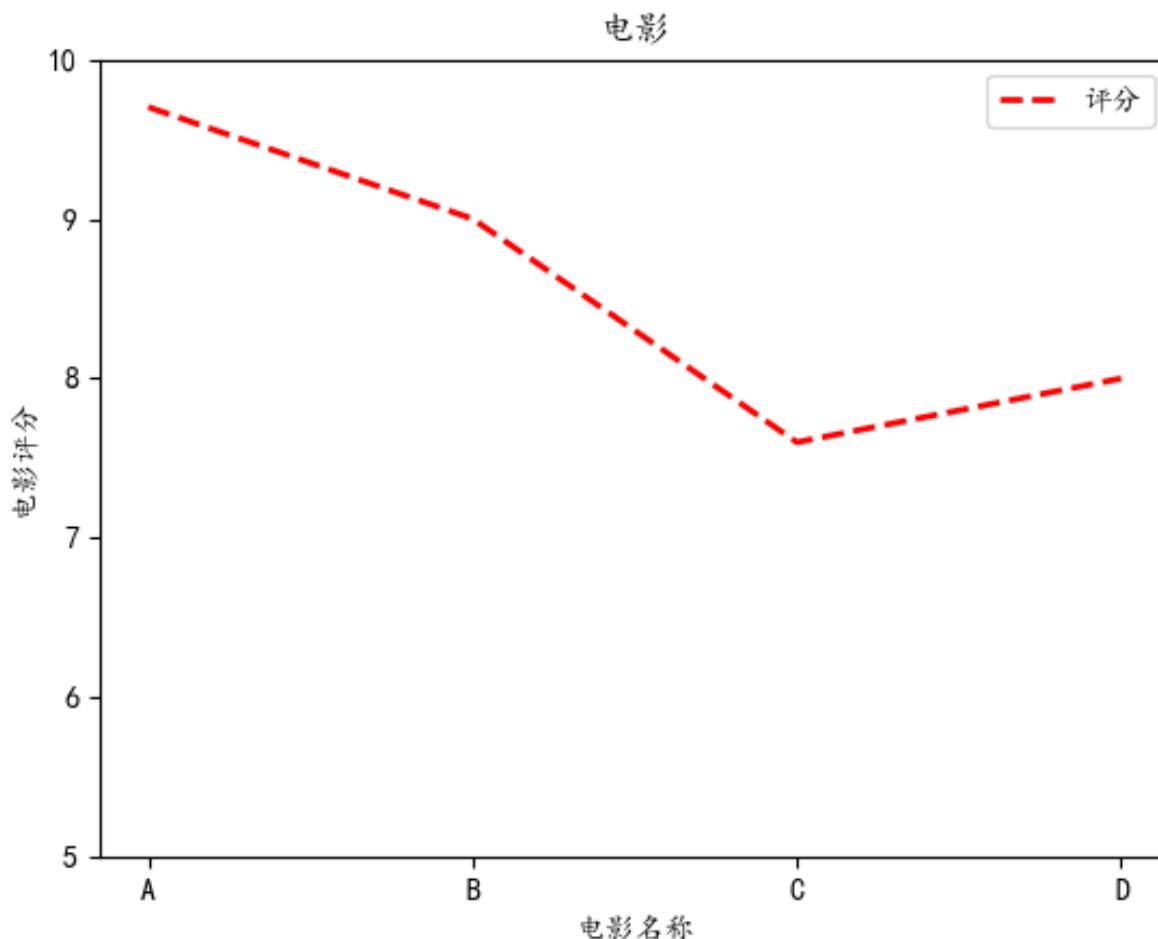
## ◆ 折线图绘制

matplotlib.pyplot.plot(\*args, \*\*kwargs)

```
import matplotlib.pyplot as plt
# 解决中文显示问题 指定默认字体
plt.rcParams['font.sans-serif'] = ['KaiTi']
plt.title("电影") # 设置图的标题
plt.xlabel("电影名称") # x轴标签
plt.ylabel("电影评分") # y轴标签
plt.ylim(5, 10) # y轴取值, 针对数值
x = ['A', 'B', 'C', 'D'] # x轴的值
y = [9.7, 9, 7.6, 8] # y轴的值
plt.plot(x,y,color="red", linewidth=2.0, linestyle="--")
plt.legend(labels=["评分"], loc="best") # 设置图例
plt.show() # 展示绘图
```

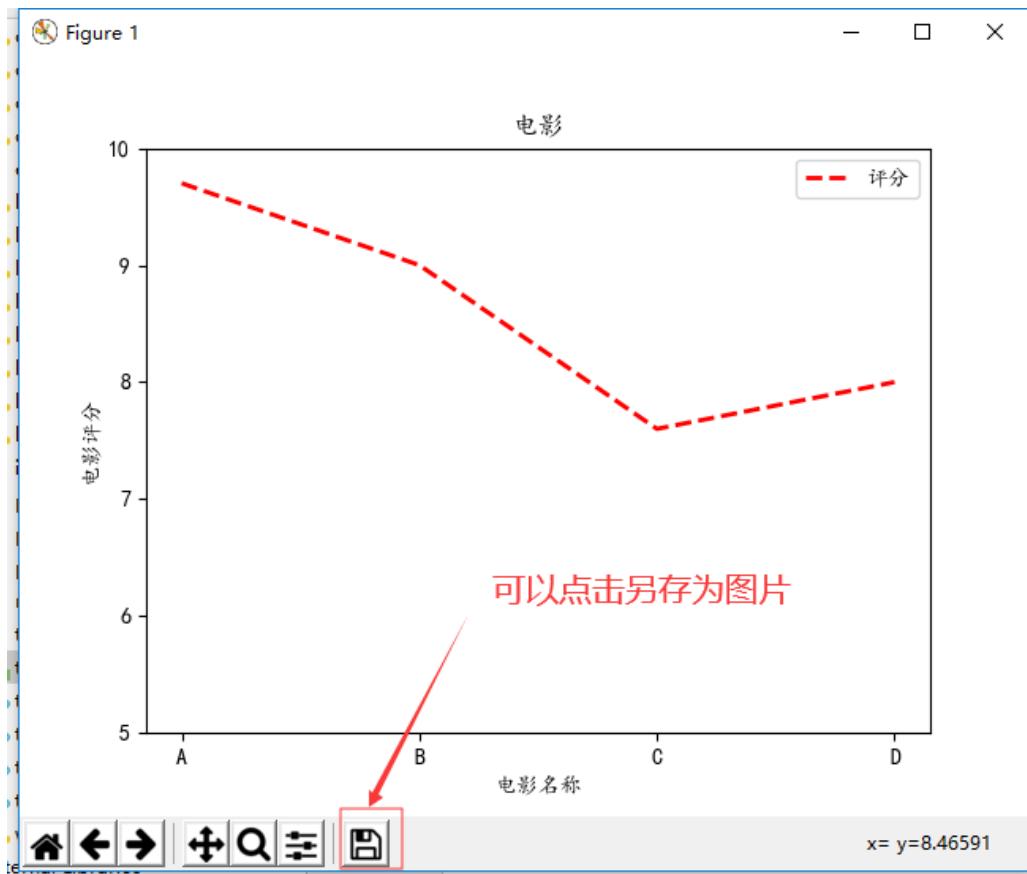
# 一、matplotlib 绘制图表 折线图

## ◆ 折线图绘制——结果



# 一、matplotlib 绘制图表 折线图

## ◆ 折线图绘制——结果





# 一、matplotlib 绘制图表 折线图

## ◆ 折线图绘制——部分属性值说明

### 线条类型 linestyle 常见属性值

字 符	描 述	字 符	描 述
'—'	实线	's'	方块符号
'---'	虚线	'p'	五角形符号
'-. '	点横虚线	'*'	星星符号
' :'	点线	'h'	六边形符号
' .'	点符号	'H'	六边形符号
', '	像素符号	'+'	加号
'o'	圆圈符号	'x'	X 符号
'v'	下三角符号	'D'	菱形符号
'^'	上三角符号	'd'	细菱形符号
'<'	左三角符号	' '	竖线符号
'>'	右三角符号	'_'	横线符号



# 一、matplotlib 绘制图表 折线图

## ◆ 折线图绘制——部分属性值说明

### 颜色color常见属性值

符 号	颜 色	符 号	颜 色
'b'	blue	'm'	magenta
'g'	green	'y'	yellow
'r'	red	'k'	black
'c'	cyan	'w'	white

### 图例位置 loc常见属性值

位置字符串	位 置 代 码	位置字符串	位 置 代 码
'best'	0	'center left'	6
'upper right'	1	'center right'	7
'upper left'	2	'lower center'	8
'lower left'	3	'upper center'	9
'lower right'	4	'center'	10
'right'	5		



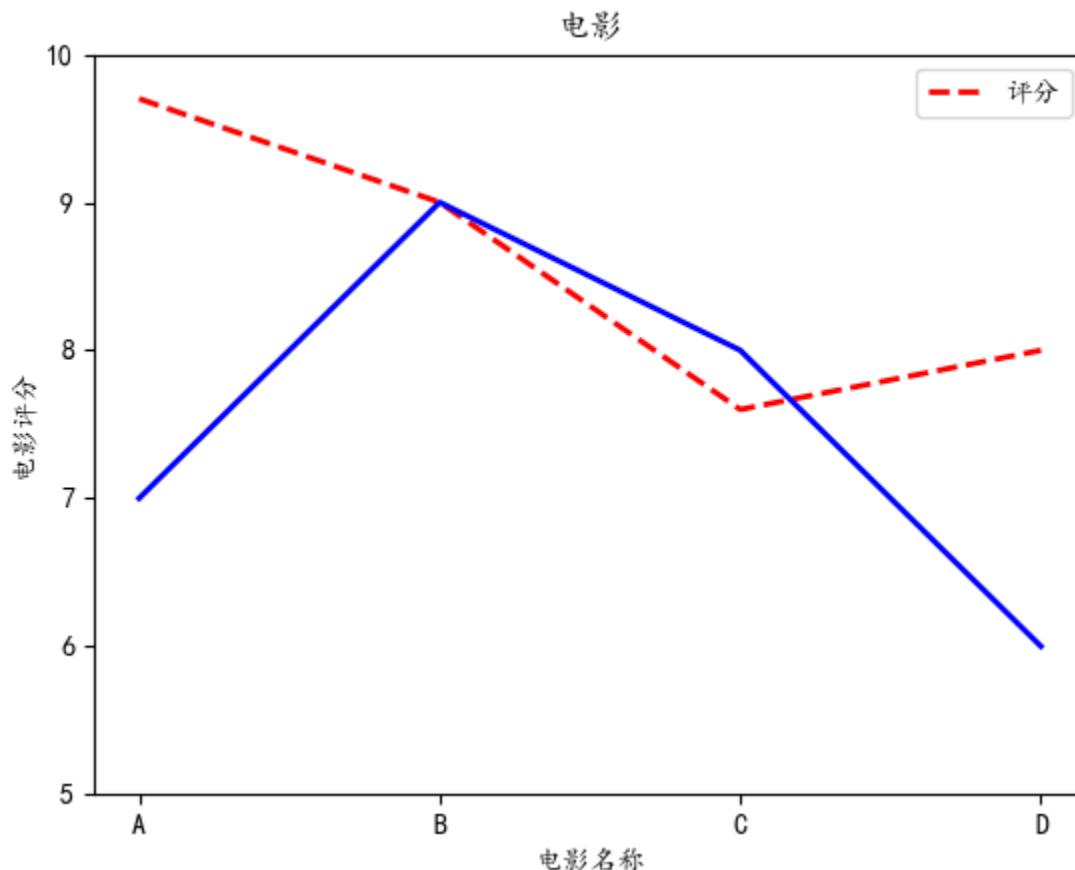
# 一、matplotlib 绘制图表 折线图

## ◆ 折线图绘制——双折线图绘制

```
import matplotlib.pyplot as plt  
...  
plt.rcParams['font.sans-serif'] = ['KaiTi']  
plt.title("电影")# 设置图的标题  
plt.xlabel("电影名称") # x轴标签  
plt.ylabel("电影评分") # y轴标签  
plt.ylim(5, 10) # y轴取值, 针对数值  
x = ['A', 'B', 'C', 'D']# x轴的值  
y = [9.7, 9, 7.6, 8]# y轴的值  
y2 = [7, 9, 8, 6] # 第二条线的y值  
plt.plot(x,y,color="red", linewidth=2.0, linestyle="--")  
plt.plot(x,y2,color="blue", linewidth=2.0, linestyle="--")  
plt.legend(labels=[ "评分"], loc="best") # 图例  
plt.show()
```

# 一、matplotlib 绘制图表 折线图

## ■ 折线图绘制——双折线图绘制





# 一、matplotlib 绘制图表 折线图

## ■ 折线图绘制——双折线图绘制

双折线图例问题

```
import matplotlib.pyplot as plt
plt.rcParams['font.sans-serif'] = ['KaiTi']
plt.title("电影")
plt.xlabel("电影名称")
plt.ylabel("电影评分")
plt.ylim(5, 10)
x = ['A', 'B', 'C', 'D']
y = [9.7, 8.6, 8, 7]
y2 = [7, 9, 8, 6]
plt.plot(x, y, color="red", linewidth=2, linestyle='-.')
plt.plot(x, y2, color="blue", linewidth=2, linestyle='--')
plt.legend(labels=['评分1', '评分2'], loc="lower center")
# 等价如下代码
# plt.plot(x, y, color="red", linewidth=2, linestyle='-.', label='评分1')
# plt.plot(x, y2, color="blue", linewidth=2, linestyle='--', label='评分2')
# plt.legend(loc="lower center")
plt.show()
```



# 一、matplotlib 绘制图表 柱状图

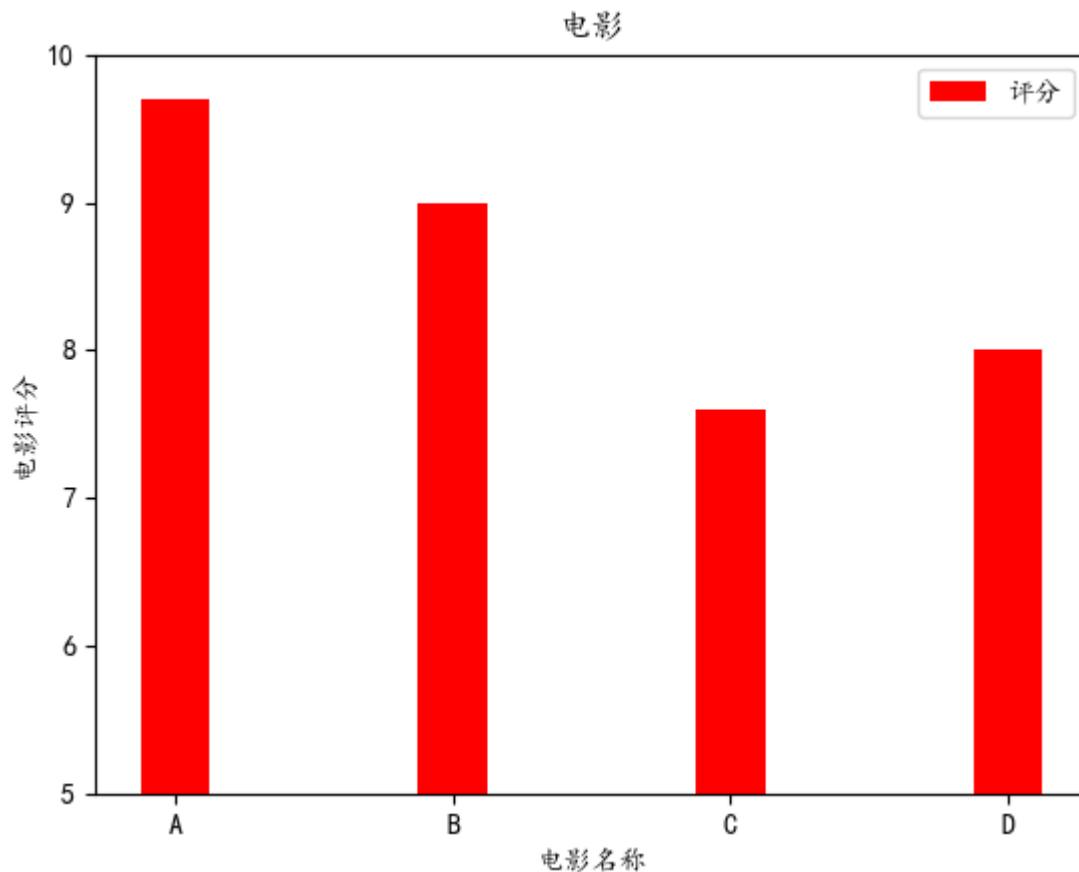
## ■ 柱状图绘制

matplotlib.pyplot.bar(\*args, \*\*kwargs)

```
import matplotlib.pyplot as plt  
...  
plt.rcParams['font.sans-serif'] = ['KaiTi']  
plt.title("电影")  
plt.xlabel("电影名称") # x轴标签  
plt.ylabel("电影评分") # y轴标签  
plt.ylim(5, 10) # y轴取值, 针对数值  
x = ['A', 'B', 'C', 'D']  
y = [9.7, 9, 7.6, 8]  
color="red" # 如果要设置不同颜色color=['r', 'g', 'b', 'c']  
plt.bar(x,y,color=color,width=0.25,linewidth=2.0,linestyle="--")  
plt.legend(labels=["评分"],loc="best")  
plt.show()
```

# 一、matplotlib 绘制图表 柱状图

## ■ 柱状图绘制





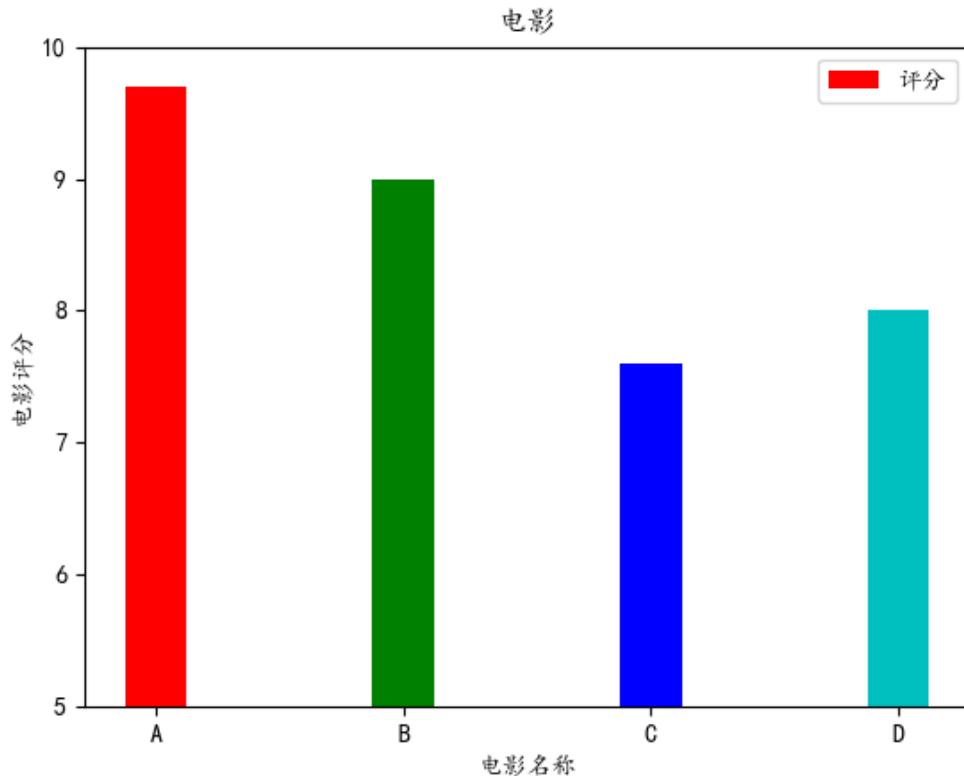
# 一、matplotlib 绘制图表 柱状图

## ■ 柱状图绘制——不同颜色的柱状图

```
import matplotlib.pyplot as plt  
...  
plt.rcParams['font.sans-serif'] = ['KaiTi']  
plt.title("电影")  
plt.xlabel("电影名称") # x轴标签  
plt.ylabel("电影评分") # y轴标签  
plt.ylim(5, 10) # y轴取值, 针对数值  
x = ['A', 'B', 'C', 'D']  
y = [9.7, 9, 7.6, 8]  
# color="red"  
color=['r','g','b','c']# 如果要设置不同颜色  
plt.bar(x,y,color=color,width=0.25,linewidth=2.0,linestyle="--")  
# plt.Legend(Label=[“评分”],Loc="best")  
plt.show()
```

# 一、matplotlib 绘制图表 柱状图

## ■ 柱状图绘制——不同颜色的柱状图



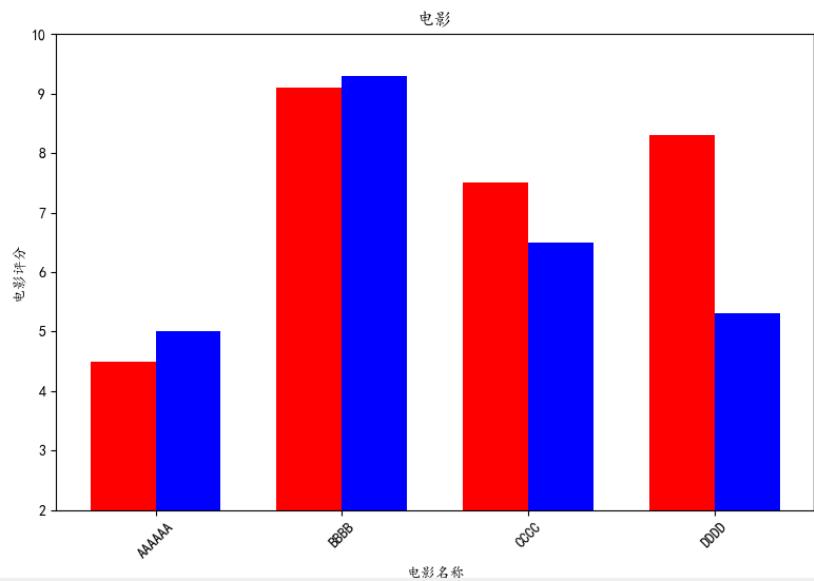
如何让标签旋转？

`plt.xticks(rotation=45)`, 其中45表示旋转角度

# 一、matplotlib 绘制图表 柱状图

## ■ 柱状图绘制——双柱状图

```
#双柱状图
import matplotlib.pyplot as plt
import numpy as np
plt.rcParams['font.sans-serif']=['KaiTi']
plt.title("电影")
plt.xlabel("电影名称")# x轴标签名
plt.ylabel("电影评分") #y轴标签名
plt.ylim(2,10) #y的取值范围
x=['AAAAAA','BBBB','CCCC','DDDD']#x轴的值
y=[4.5,9.1,7.5,8.3]#y轴的值
y2 = [5, 9.3, 6.5, 5.3] # y轴的值
bar_w=0.35
r1=np.arange(len(x))
r2=[i+bar_w for i in r1]
print(r1)
print(r2)
plt.bar(r1,y,color="red",width=bar_w,label="官方评分")
plt.bar(r2,y2,color="blue",width=bar_w,label="我的评分")
plt.xticks([i+bar_w/2 for i in r1],x,rotation=45)
# plt.legend(labels=["官方评分"])
plt.show()
```





# 一、matplotlib 绘制图表 饼状图

## ■ 饼状图绘制

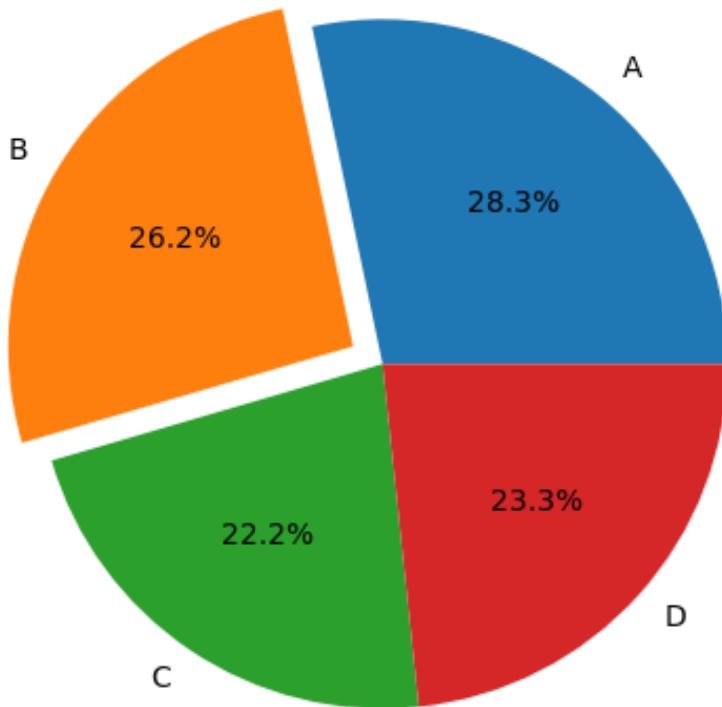
```
import matplotlib.pyplot as plt  
...  
x = ['A', 'B', 'C', 'D']  
y = [9.7, 9, 7.6, 8]  
explodevalue = (0,0.1,0,0) # 突出分片设置  
fig1,ax1 = plt.subplots()  
ax1.pie(y,labels=x,explode=explodevalue,autopct='%.1f%%')  
# autopct 数据格式 %%打印%  
ax1.axis('equal')  
plt.show()
```

Axix('equal'): 为了让显示的饼图保持圆形

%.1f%%: 保留1位小数的百分比形式

# 一、matplotlib 绘制图表 饼状图

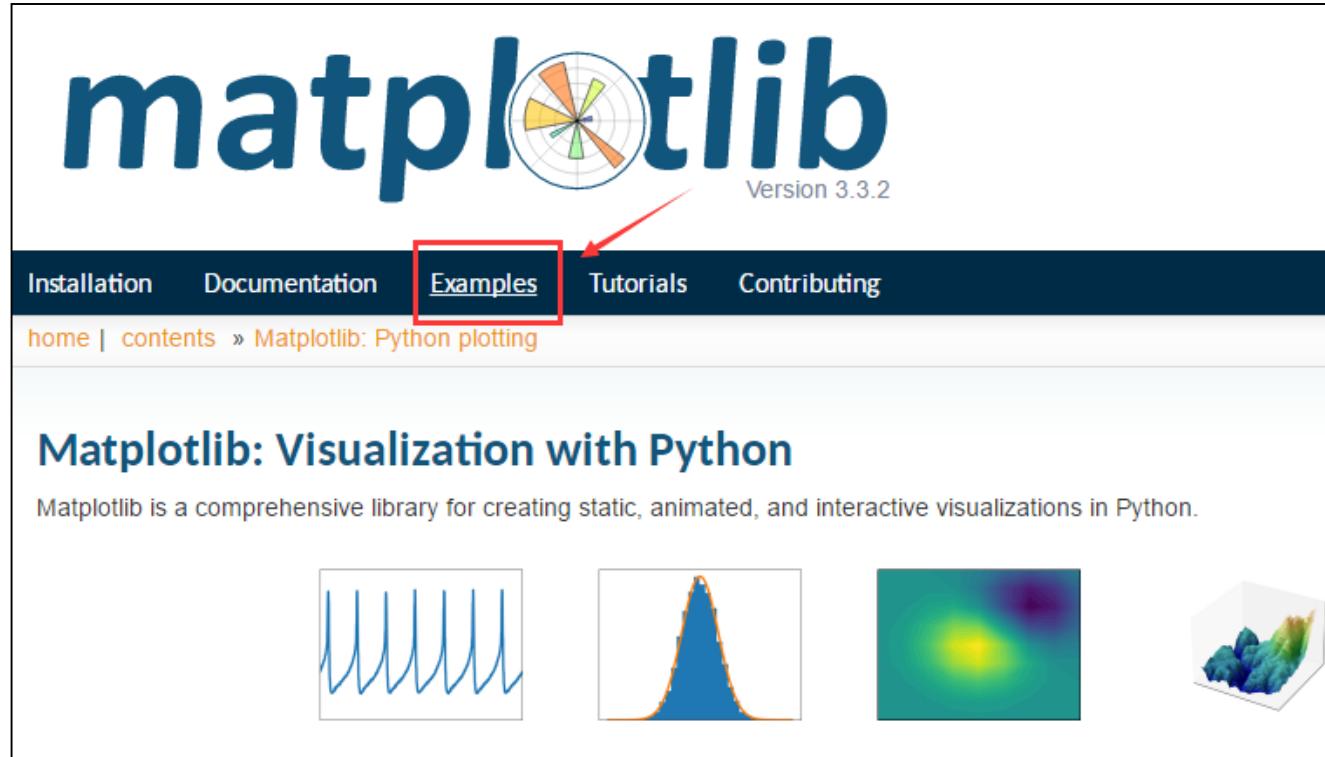
## ■ 饼状图绘制



# 一、matplotlib 绘制图表 更多

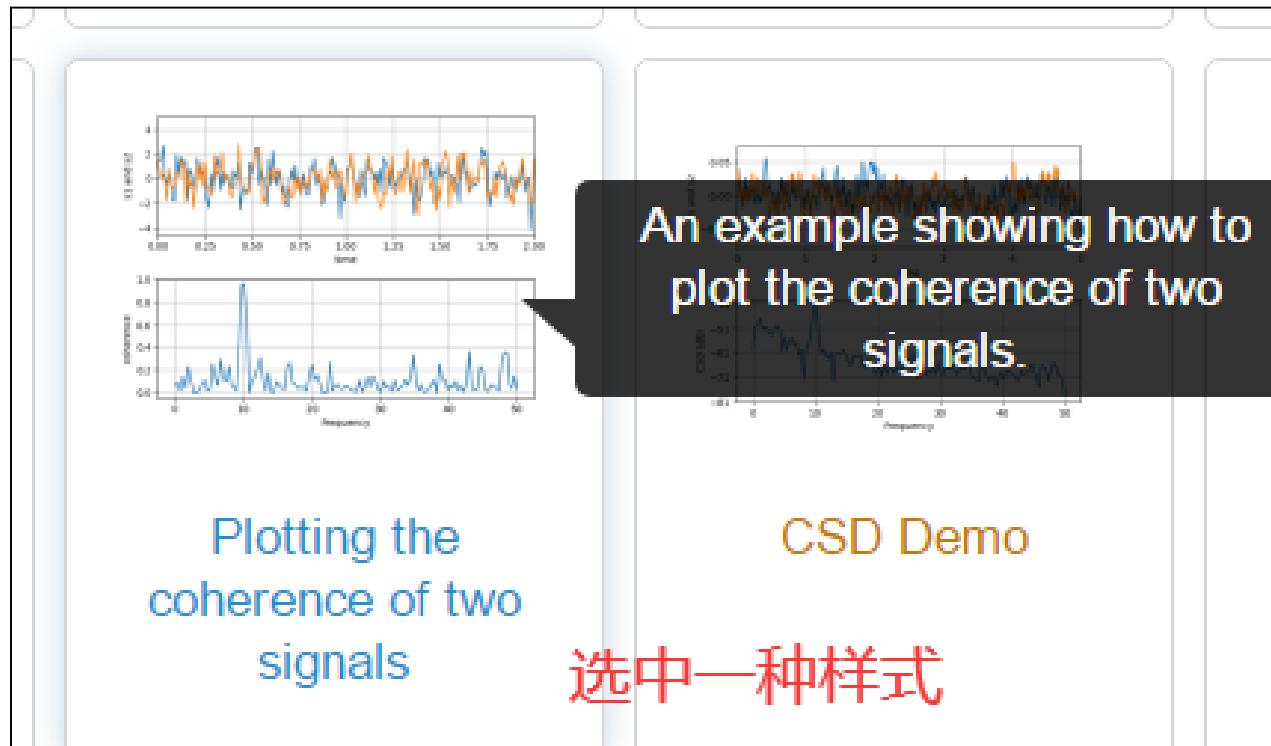
- 如何做出更加酷炫的图标效果呢？

Matplotlib官网 <https://matplotlib.org/index.html>



# 一、matplotlib 绘制图表 更多

■ 如何做出更加酷炫的图标效果呢？





# 一、matplotlib 绘制图表 更多

## ■ 如何做出更加酷炫的图标效果呢？

```
import numpy as np
import matplotlib.pyplot as plt

# Fixing random state for reproducibility
np.random.seed(19680801)

dt = 0.01
t = np.arange(0, 30, dt)
nse1 = np.random.randn(len(t))                               # white noise 1
nse2 = np.random.randn(len(t))                               # white noise 2

# Two signals with a coherent part at 10Hz and a random part
s1 = np.sin(2 * np.pi * 10 * t) + nse1
s2 = np.sin(2 * np.pi * 10 * t) + nse2

fig, axs = plt.subplots(2, 1)
axs[0].plot(t, s1, t, s2)
axs[0].set_xlim(0, 2)
axs[0].set_xlabel('time')
axs[0].set_ylabel('s1 and s2')
axs[0].grid(True)

cxy, f = axs[1].cohere(s1, s2, 256, 1. / dt)
axs[1].set_ylabel('coherence')

fig.tight_layout()
plt.show()
```

Copied!

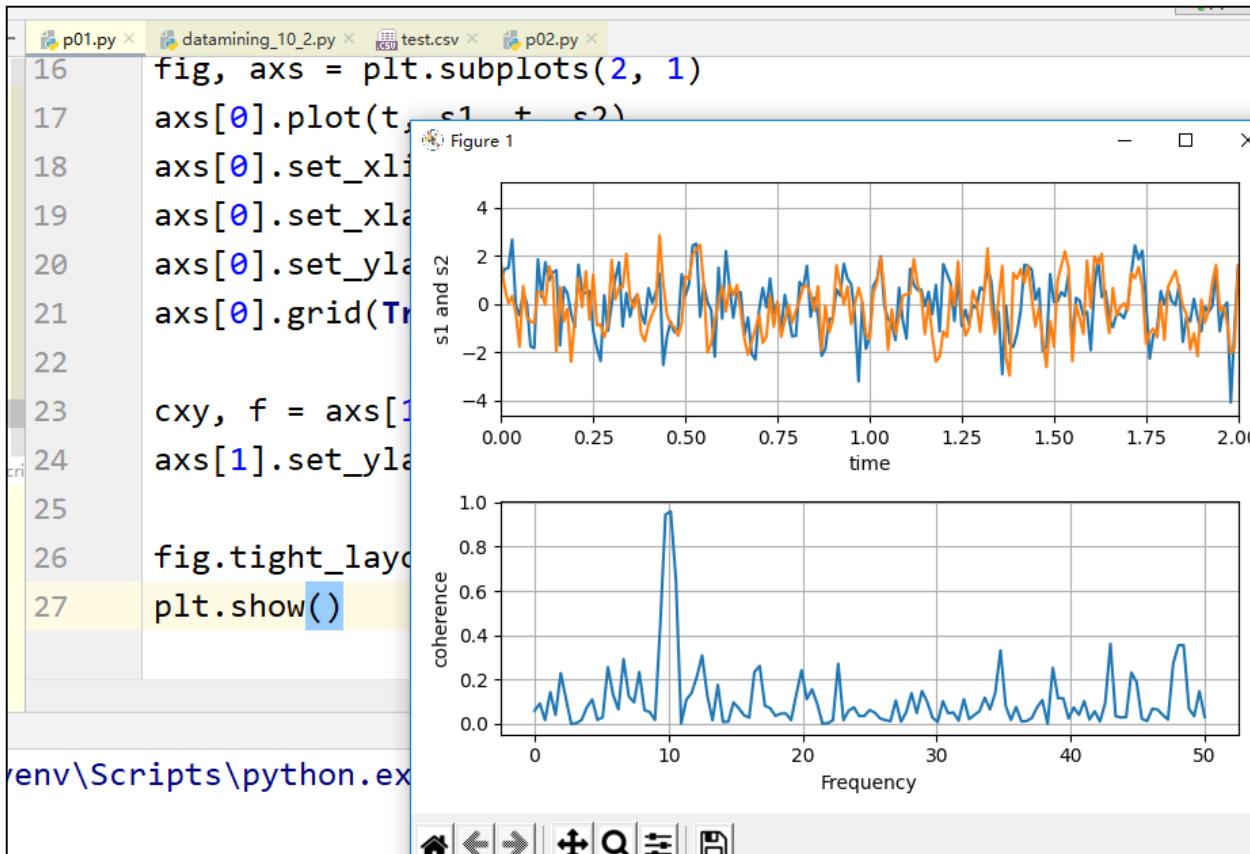
复制代码

```
import numpy as np
import matplotlib.pyplot as plt
```

```
# Fixing random state for reproducibility
np.random.seed(19680801)
```

# 一、matplotlib 绘制图表 更多

## ■ 如何做出更加酷炫的图标效果呢？



根据自己的数据完成数据的替换。



# 本次课主要内容

## 一、 matplotlib 绘制图表

1.1、 折线图绘制

1.2、 柱状图绘制

1.3、 饼状图绘制

## 二、 wordcloud 绘制词云



## 二、 wordcloud词云绘制

### ■ wordcloud 简介

wordcloud是Pyhon 中用于绘制词云的模块。 wordcloud 的安装如下：

安装命令： pip install wordcloud

或者： pip install -i https://pypi.tuna.tsinghua.edu.cn/simple wordcloud

```
(venv) F:\pythonTest\week1>pip install wordcloud
Collecting wordcloud
```



## 二、wordcloud词云绘制

### ■ wordcloud 简介

词云生成使用过程中需要安装的其他模块

**jieba** 模块是一个用于中文分词的模块

```
(venv) F:\pythonTest\week1>pip install jieba  
Collecting jieba
```

**imageio** 模块是一个图像输入输出模块

```
(venv) F:\pythonTest\week1>pip install imageio  
Collecting imageio
```



## 二、wordcloud词云绘制

### ■ wordcloud 简介

参数	描述
width	指定词云对象生成图片的宽度，默认400像素。 <code>w=wordcloud.WordCloud(width=600)</code>
height	指定词云对象生成图片的高度，默认200像素。 <code>w=wordcloud.WordCloud(height=400)</code>
min_font_size	指定词云中字体的最小字号，默认4号。 <code>w=wordcloud.WordCloud(min_font_size=10)</code>
max_font_size	指定词云中字体的最大字号，根据高度自动调节。 <code>w=wordcloud.WordCloud(max_font_size=20)</code>
font_step	指定词云中字体字号的步进间隔，默认为1。 <code>w=wordcloud.WordCloud(font_step=2)</code>



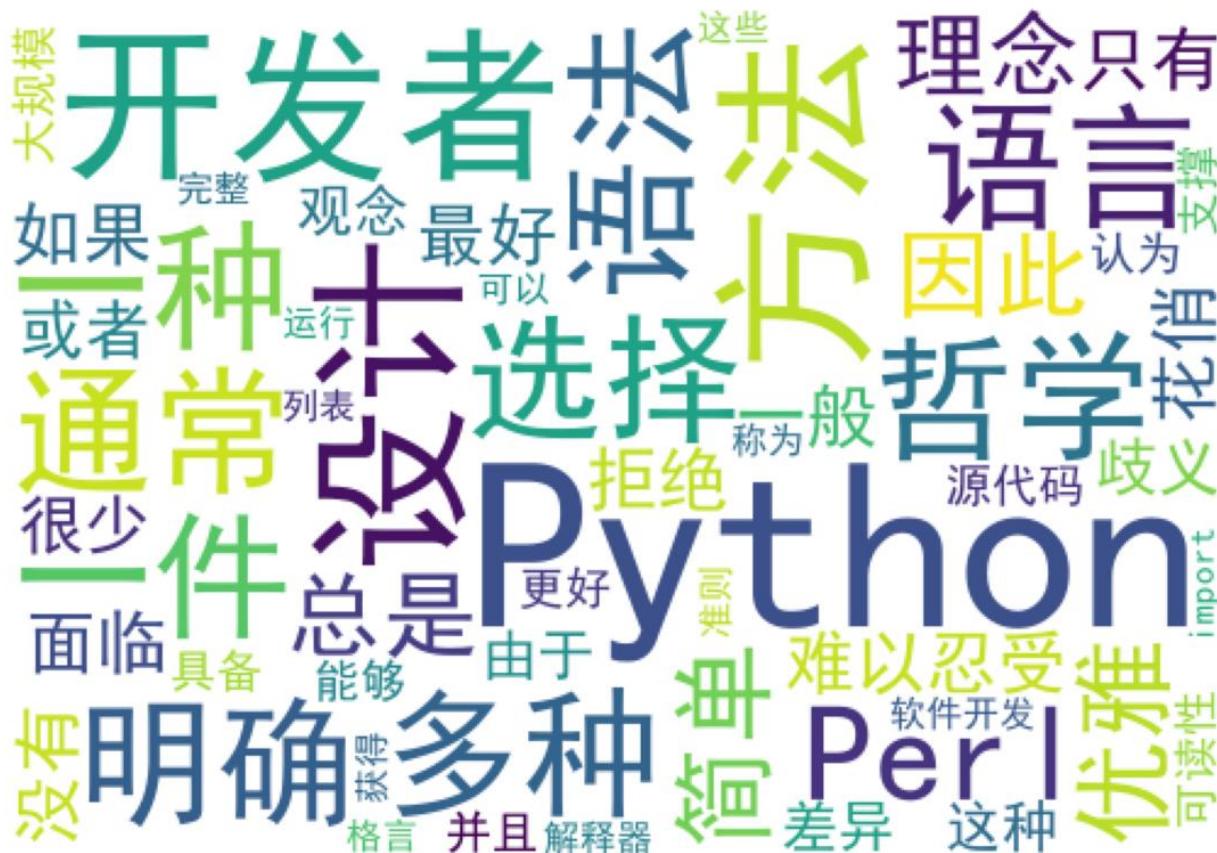
## 二、wordcloud词云绘制

### ■ wordcloud 简介

参数	描述
font_path	指定字体文件的路径,默认None。 w=wordcloud.WordCloud(font_path="msyh.ttc")
max_words	指定词云显示的最大单词数量,默认200。 w=wordcloud.WordCloud(max_words=20)
stop_words	指定词云的排除词列表,即不显示的单词列表。 w=wordcloud.WordCloud(stop_words={"Python"})
mask	指定词云形状,默认为长方形,需要引用imread()函数。 from scipy.misc import imread mk=imread("pic.png") w=wordcloud.WordCloud(mask=mk)
background_color	指定词云图片的背景颜色,默认为黑色。 w=wordcloud.WordCloud(background_color="white")

## 二、wordcloud词云绘制

### ■ wordcloud 实例





## 二、wordcloud词云绘制

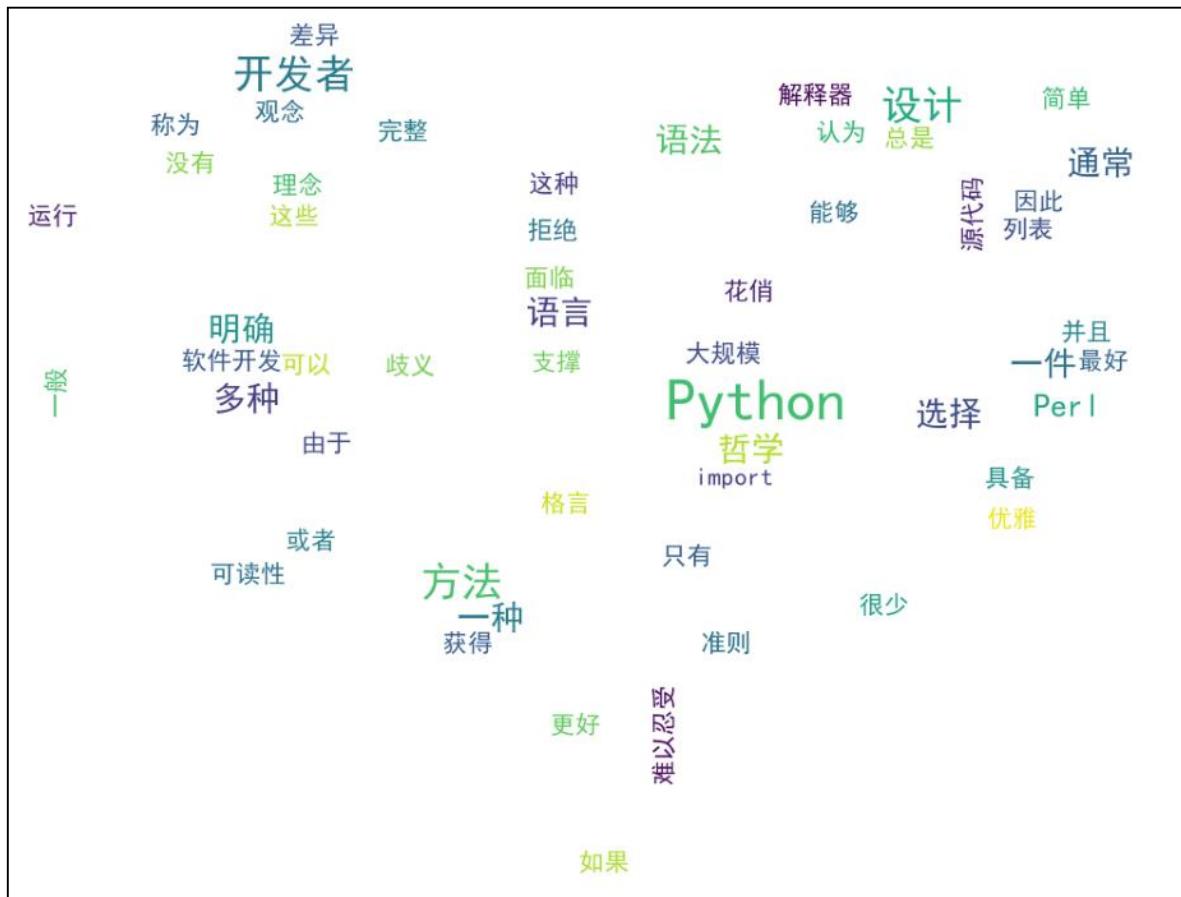
### ■ wordcloud 实例

```
import matplotlib.pyplot as plt
import jieba
from wordcloud import WordCloud
text = open(r'6.txt', "r", encoding="utf-8").read()# 读入txt
cut_text = jieba.cut(text)# 2.结巴中文分词
result = " ".join(cut_text)# 分词结果空格连接
wc = WordCloud(
    # 中文需要设置字体，避免乱码，英文不必
    font_path='./fonts/simhei.ttf',
    background_color='white',# 设置背景色
    width=500, # 设置背景宽
    height=350, # 设置背景高
    max_font_size=100,# 最大字体
    min_font_size=10, # 最小字体
    mode='RGBA' #colormap='pink'
)
wc.generate(result)# 产生词云
wc.to_file(r"wordcloud.png")# 保存图片
plt.figure("Python介绍")# 指定所绘图名称
plt.imshow(wc)# 以图片的形式显示词云
plt.axis("off")# 关闭图像坐标系
plt.show()
```

## 二、wordcloud词云绘制

### ■ wordcloud 实例

如何修改词云的形状?



## 二、wordcloud词云绘制

### ■ wordcloud 实例

- 1、准备一张心形图片；
- 2、设置mask属性。



[https://blog.csdn.net/weizjn\\_49746433](https://blog.csdn.net/weizjn_49746433)



## 二、wordcloud词云绘制

### ■ wordcloud 实例

```
import matplotlib.pyplot as plt
import jieba
import imageio
from wordcloud import WordCloud
text = open(r'6.txt', "r", encoding="utf-8").read()# 读入txt
cut_text = jieba.cut(text)# 2.结巴中文分词
result = " ".join(cut_text)# 分词结果空格连接
mk=imageio.imread("1.png")
wc = WordCloud(
    # 中文需要设置字体，避免乱码，英文不必
    font_path='./fonts/simhei.ttf',
    background_color='white',# 设置背景色
    width=500, # 设置背景宽
    height=350, # 设置背景高
    max_font_size=100,# 最大字体
    min_font_size=10, # 最小字体
    mode='RGBA',#colormap='pink'
    mask=mk)
wc.generate(result)# 产生词云
wc.to_file(r"wordcloud.png")# 保存图片
plt.figure("Python介绍")# 指定所绘图名称
plt.imshow(wc)# 以图片的形式显示词云
plt.axis("off")# 关闭图像坐标系
plt.show()
```



### 三、扩展

#### ■ pyecharts

其实，Python绘图的方式很多，也有很多开源不错的模块可以选择，比如常用于科学计算的Matplotlib、Chaco、Python Google Chart、PyCha、pyOFC2、PyChart、PLplot、ReportLab、VPython等等。

下面为大家提供**pyecharts**模块。



### 三、扩展

## ■ pyecharts

首先安装pyecharts 模块

```
PS D:\pyprojects> pip install pyecharts
Collecting pyecharts
  Downloading pyecharts-1.9.0-py3-none-any.whl (135 kB)
    ████████████████████████████████████████████ | 135 kB 22 kB/s
Collecting prettytable
  Downloading prettytable-2.2.1-py3-none-any.whl (23 kB)
Collecting jinja2
```



### 三、扩展

## ■ pyecharts

Pyeahcrt中中文手册链接: <https://pyecharts.org/#/zh-cn/intro>

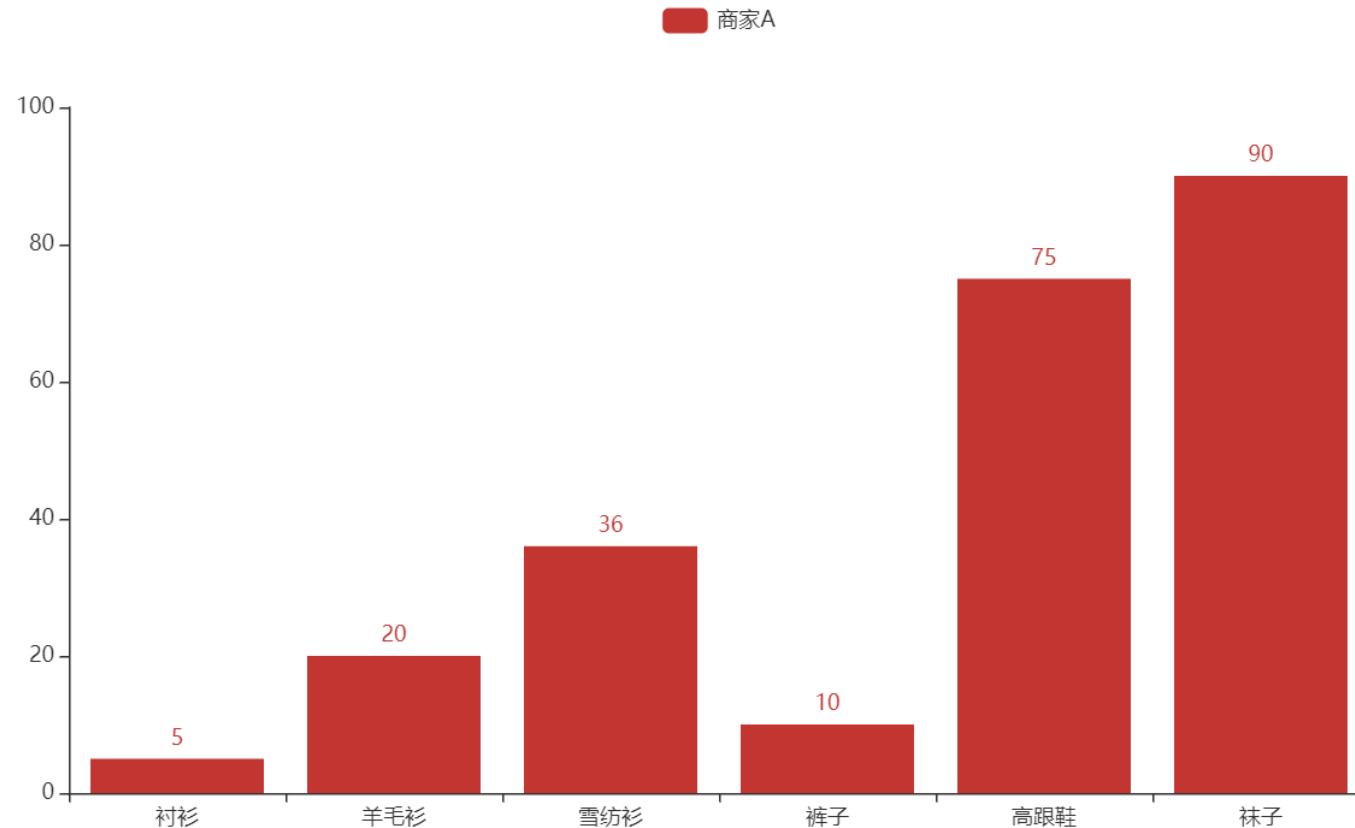
以其中一个柱状图为例

<https://pyecharts.org/#/zh-cn/quickstart>

```
main.py x
1 from pyecharts.charts import Bar
2
3 bar = Bar()
4 bar.add_xaxis(["衬衫", "羊毛衫", "雪纺衫", "裤子", "高跟鞋", "袜子"])
5 bar.add_yaxis("商家A", [5, 20, 36, 10, 75, 90])
6 # render 会生成本地 HTML 文件，默认会在当前目录生成 render.html 文件
7 # 也可以传入路径参数，如 bar.render("mycharts.html")
8 bar.render()
```

### 三、扩展

## ■ pyecharts





### 三、扩展

#### ■ pyecharts

更多demo展示

[https://gallery.pyecharts.org/#/Bar/stack\\_bar\\_percent](https://gallery.pyecharts.org/#/Bar/stack_bar_percent)

---

# 谢谢！

---