

# 数据挖掘的day\_1

## 环境安装

matplotlib  
numpy  
pandas  
TA-Lib  
tables  
jupyter

## Jupyter notebook

当python的一个IDE用就好了  
运行: jupyter notebook

## Matplotlib

Scripting (脚本)  
Artist (美工)  
Backend (后端)

## 折线图

1. 导入模块: `import matplotlib.pyplot as plt`
2. 中文及负值显示: `from matplotlib.pyplot import mpl.rcParams['font.sans-serif'] = ['arial unicode ms']`  
`mpl.rcParams['axes.unicode_minus'] = False`
3. 指定图形大小: `plt.figure(figsize=(x,y), dpi=z)`  
x: 长; y: 宽; z: 清晰度
4. 构造数据: `x = range(A); y=[B for i in x]`
5. 绘图: `plt.plot(x, y, label="标题名")`
6. 自定义坐标轴刻度值: `plt.xticks(x, "A"); plt.yticks(B, "B")`
7. 构造中文: `x_ch = ["xxxx"]*len(x).format(i) for i in x`
8. 修改坐标刻度: `plt.xticks(x[:5], x_ch[:5])`  
`plt.yticks(y_ticks[:5])`
9. 增加坐标轴标签:  
`plt.xlabel("A")`  
`plt.ylabel("B")`  
`plt.title("C")`
10. 添加图例: `plt.legend(loc="best")`
11. 保存图片: `plt.savefig(path)`

## 多坐标轴显示折线图

`fig, ax = plt.subplots(nrows=1, ncols=2, figsize=(A,B))`  
fig是画布, ax是坐标系, 通过`ax[0].ax[1]`获取  
`ax[0].plot(...)`  
`ax[0].set_xticks`  
`ax[0].set_xlabel`  
`ax[0].set_title`  
`ax[0].legend`

## 柱状图

1. 导入模块: `import matplotlib.pyplot as plt`
2. 中文及负值显示: `from matplotlib.pyplot import mpl.rcParams['font.sans-serif'] = ['arial unicode ms']`  
`mpl.rcParams['axes.unicode_minus'] = False`
3. 指定图形大小: `plt.figure(figsize=(x,y), dpi=z)`  
x: 长; y: 宽; z: 清晰度
4. 构造数据: `x = range(len(A)); y=[A,B,C...]`
5. 绘图: `plt.bar(x, y, label="标题名")`
6. 自定义坐标轴刻度值: `plt.xticks(x, "A"); plt.yticks(B, "B")`
7. 构造中文: `x_ch = ["xxxx"]*len(x).format(i) for i in x`
8. 修改坐标刻度: `plt.xticks(x[:5], x_ch[:5])`  
`plt.yticks(y_ticks[:5])`
9. 增加坐标轴标签:  
`plt.xlabel("A")`  
`plt.ylabel("B")`  
`plt.title("C")`
10. 添加图例: `plt.legend(loc="best")`
11. 保存图片: `plt.savefig(path)`

## 多个柱状图及坐标位置

`plt.bar(x, first_day, width=0.2, color='r', label="A")`  
`plt.bar([i+0.2 for i in x], first_weekend, width=0.2, label="B")`  
`plt.xticks([i+0.1 for i in x], movie_name)`

## 直方图

1. 导入模块: `import matplotlib.pyplot as plt`
2. 中文及负值显示: `from matplotlib.pyplot import mpl.rcParams['font.sans-serif'] = ['arial unicode ms']`  
`mpl.rcParams['axes.unicode_minus'] = False`
3. 指定图形大小: `plt.figure(figsize=(x,y), dpi=z)`  
x: 长; y: 宽; z: 清晰度
4. 构造数据: `data = [...]`  
设置组距: `bins = 2(num)`  
设置组数: `group = int((max(data) - min(data))/bins)`
5. 绘图: `plt.hist(data, group)`  
添加网格显示: `plt.grid(True, linestyle='-', alpha=0.6)`
6. 自定义坐标轴刻度值: `plt.xticks(x, "A"); plt.yticks(B, "B")`
7. 构造中文: `x_ch = ["xxxx"]*len(x).format(i) for i in x`
8. 修改坐标刻度: `plt.xticks(x[:5], x_ch[:5])`  
`plt.yticks(y_ticks[:5])`
9. 增加坐标轴标签:  
`plt.xlabel("A")`  
`plt.ylabel("B")`  
`plt.title("C")`
10. 添加图例: `plt.legend(loc="best")`
11. 保存图片: `plt.savefig(path)`

## 饼图

1. 导入模块: `import matplotlib.pyplot as plt`
2. 中文及负值显示: `from matplotlib.pyplot import mpl.rcParams['font.sans-serif'] = ['arial unicode ms']`  
`mpl.rcParams['axes.unicode_minus'] = False`
3. 指定图形大小: `plt.figure(figsize=(x,y), dpi=z)`  
x: 长; y: 宽; z: 清晰度
4. 构造数据: `data = [...]`
5. 绘图: `plt.pie(data, labels=AAA, autopct='%1.2f%%', colors=['A'])`  
`plt.axis('equal') # 让饼保持圆形`
10. 添加图例: `plt.legend(loc="best")`
11. 保存图片: `plt.savefig(path)`