

# 小伙伴计划暑期学习营——零基础Python入门

## 第五讲：数据处理入门

张智帅 电子系

清华大学学生学业与发展指导中心  
2019-2020学年夏季学期

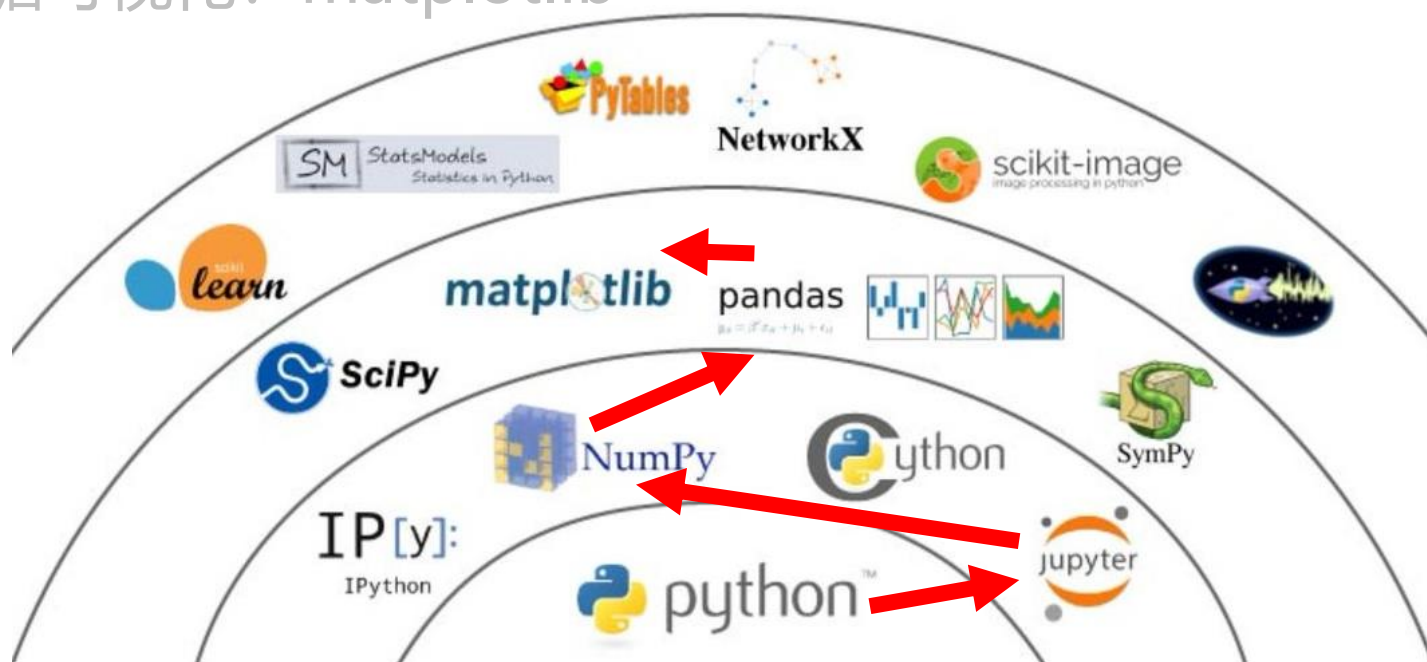
# 第五讲-数据处理入门-目录

## ■ 安装必备库

□ 科学计算: NumPy

□ 数据处理: pandas

□ 数据可视化: matplotlib



**Python数据科学生态系统**

# 数据处理基本步骤

$\infty$   $\pi$   $f_x$   
 $=$   $\neq$   $\Sigma$   
 $\cancel{+}$   $/$   $>$   
数学

 **SciPy**  
科学计算算法

 **scikit learn**  
Machine Learning with Scikit-Learn  
机器学习算法



## 1. 读取数据

(结构化数据: 矩阵、  
表格、时间序列.....)



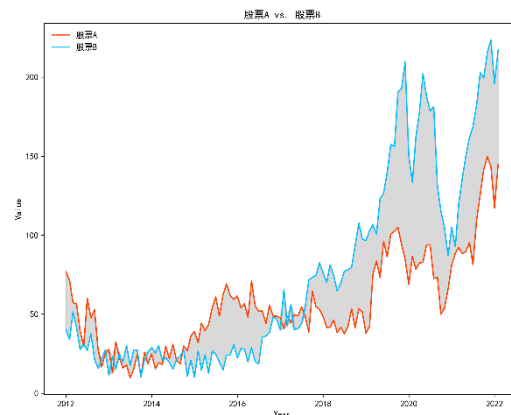
## 2. 数据预处理



## 3. 数据分析



## 4. 数据可视化



## 5. 图像、结果、结论.....

# 配环境

## ❑ 安装必备库NumPy、pandas、matplotlib

```
[3] ▶ Ml
import numpy as np

-----
ModuleNotFoundError                                Traceback (most recent call last)
in
----> 1 import numpy as np

ModuleNotFoundError: No module named 'numpy'
```



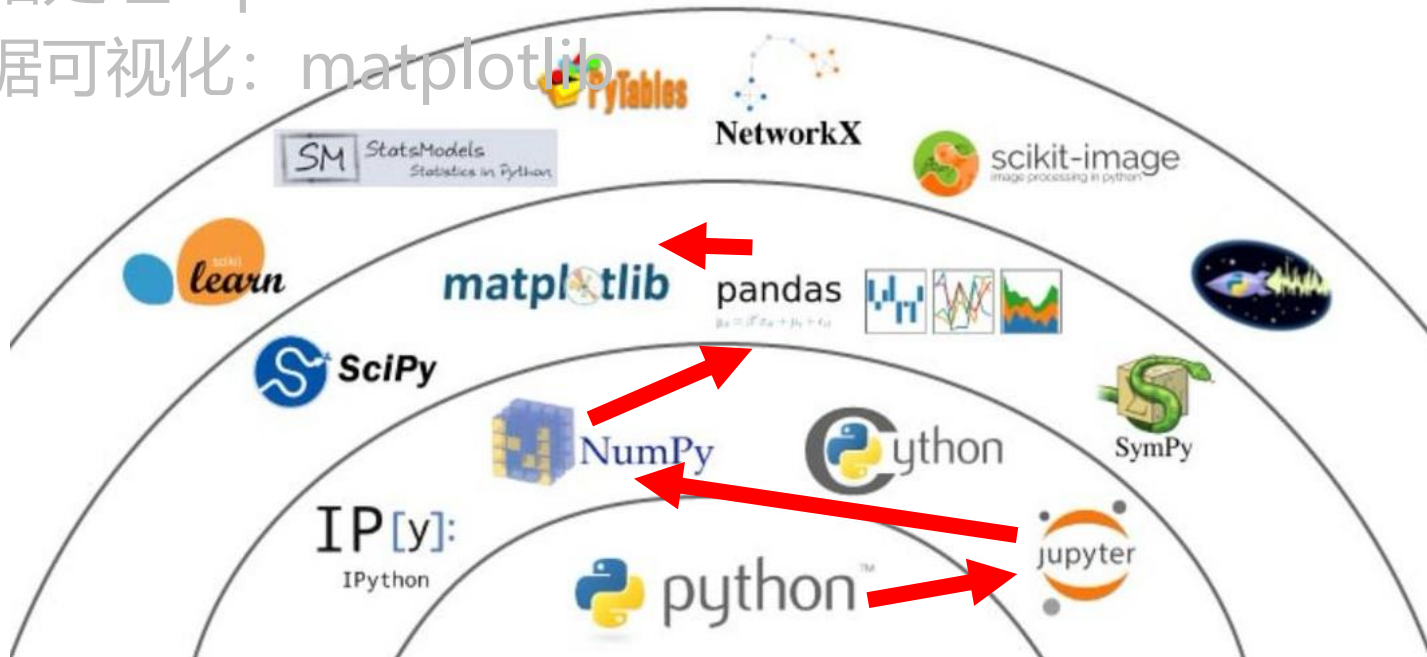
```
[2] ▶ Ml
# 导入必备库，若没有则pip install <库名>
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```



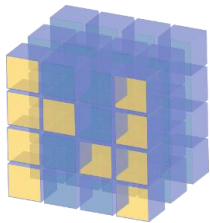
## ❑ 终端输入：pip install numpy, pip install pandas, pip install matplotlib

# 第五讲-数据处理入门-目录

- 安装必备库
- 科学计算：NumPy
  - ndarray
- 数据处理：pandas
- 数据可视化：matplotlib



**Python数据科学生态系统**



# NumPy

□ Numerical Python, 科学计算的基础包

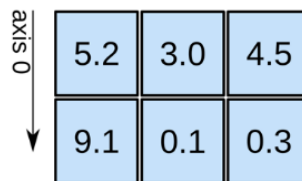
➤ 快速、高效的**高维数组对象ndarray**

1D array



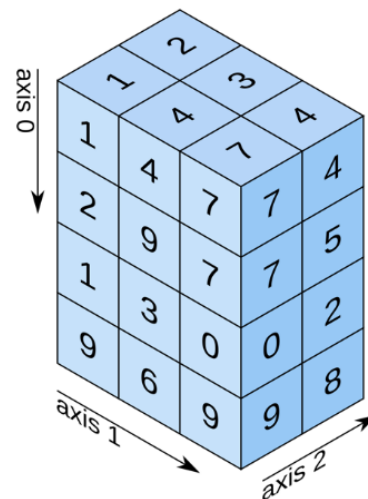
shape: (4,)

2D array



shape: (2, 3)

3D array



shape: (4, 3, 2)

➤ 从序列创建数组

```
▶ M4  
a = np.array(1) # 0维数组 (标量)  
b = np.array([1, 2]) # 1维数组 (向量)  
c = np.array([[1.0, 2.0, 3.0], [4.0, 5.0, 6.0]]) # 2维数组 (矩阵)  
d = np.array([[[1, 2, 3], [4, 5, 6]], [[4, 5, 6], [1, 2, 3]]]) # 3维数组 (张量)  
# 多维数组需要用嵌套的列表创建
```

# NumPy-ndarray

## ➤ 随机数组

```
a = np.random.rand(10)# 均匀随机数  
b = np.random.randn(2,10)# 正态分布的随机数  
c = np.random.randint(0,100,10)# 随机整数
```

## ➤ 数组索引

```
c = np.random.randint(0,100,[4,3])# 二维随机整数  
print(c)  
print("\n",c[3]) # 默认按照行索引  
print("\n",c[:,2]) # 按照列索引
```

## ➤ 数组属性与函数

```
print(data_array.ndim) #维度数  
print(data_array.shape) #形状（尺寸）  
print(data_array.dtype) #元素数据类型
```

```
print(arr.sum()) # 求和  
print(arr.max()) # 最大值  
print(arr.min()) # 最小值
```

```
print(arr.mean()) # 平均值  
print(arr.std()) # 标准差  
print(arr.var()) # 方差
```

# NumPy-矢量运算

- 相同尺寸：逐元素运算

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} + - \times \div \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} = ?$$

- 不同尺寸：

- 标量×向量：向量的数乘

$$10 \times (0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6) = (0 \ 10 \ 20 \ 30 \ 40 \ 50 \ 60)$$

- $(4 \times 3)$ 矩阵 +  $(1 \times 3)$ 向量？

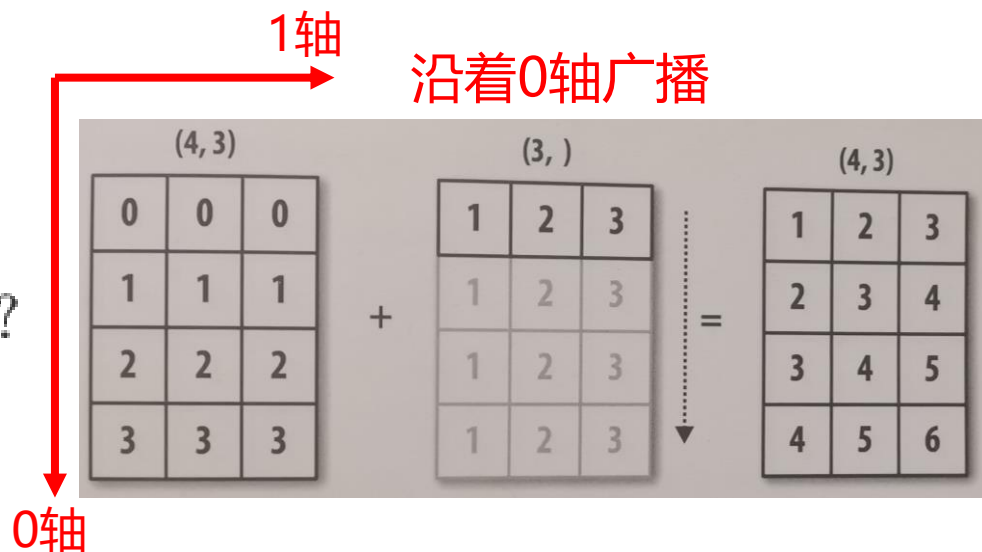
$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \end{pmatrix} + (1 \ 2 \ 3) = ?$$



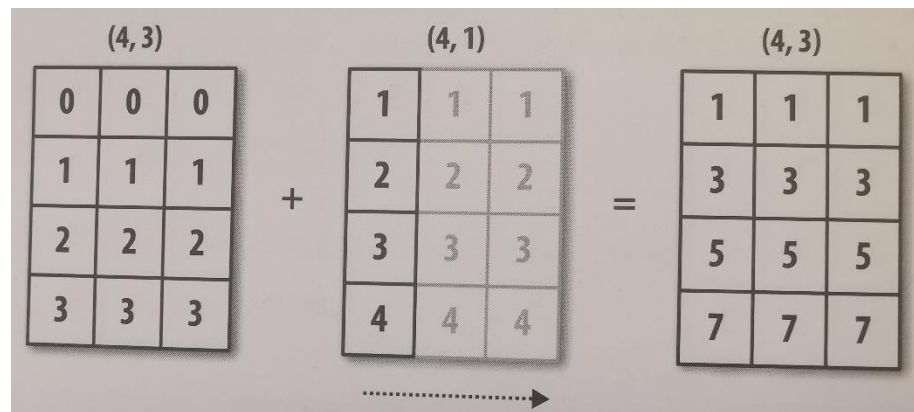
# NumPy-数组广播

□ 这能运行吗?

$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \end{pmatrix} + (1 \ 2 \ 3) = ?$$



$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \end{pmatrix} + \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} = ?$$



□ 广播维必须为1，广播维之外的维度相同

# NumPy-线性代数运算

## □ 矩阵乘法

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \end{pmatrix} = \begin{pmatrix} 70 & 80 & 90 \\ 158 & 184 & 210 \end{pmatrix}$$

## □ 转置

```
[171] ▶ MI
arr1 = np.arange(1,9).reshape(2,4)
arr2 = np.arange(1,13).reshape(4,3)
print(arr1)
print(arr2)
|
arr1.dot(arr2) # 矩阵乘法

[[1 2 3 4]
 [5 6 7 8]]
[[ 1  2  3]
 [ 4  5  6]
 [ 7  8  9]
 [10 11 12]]

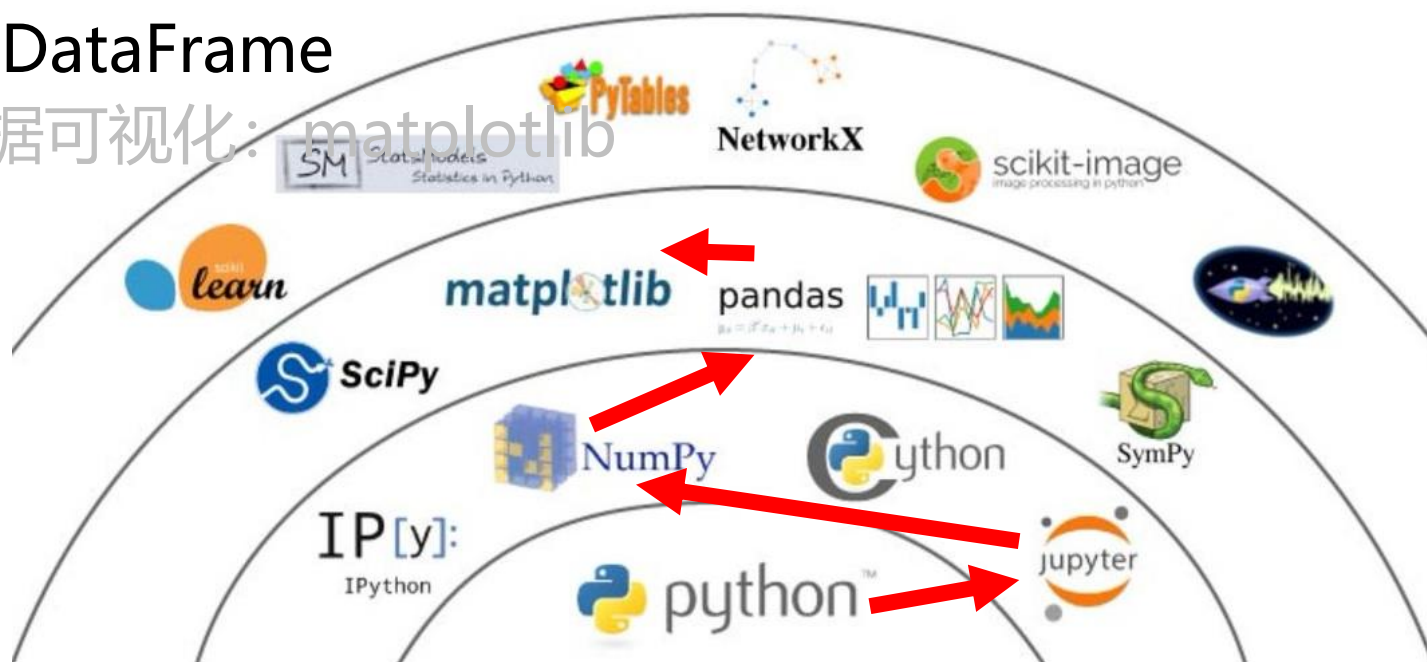
array([[ 70,  80,  90],
       [158, 184, 210]])
```

```
[172] ▶ MI
arr1.T #矩阵转置

array([[1, 5],
       [2, 6],
       [3, 7],
       [4, 8]])
```

# 第五讲-数据处理入门-目录

- 安装必备库
- 科学计算: NumPy
- 数据处理: pandas
  - Series
  - DataFrame
- 数据可视化: matplotlib



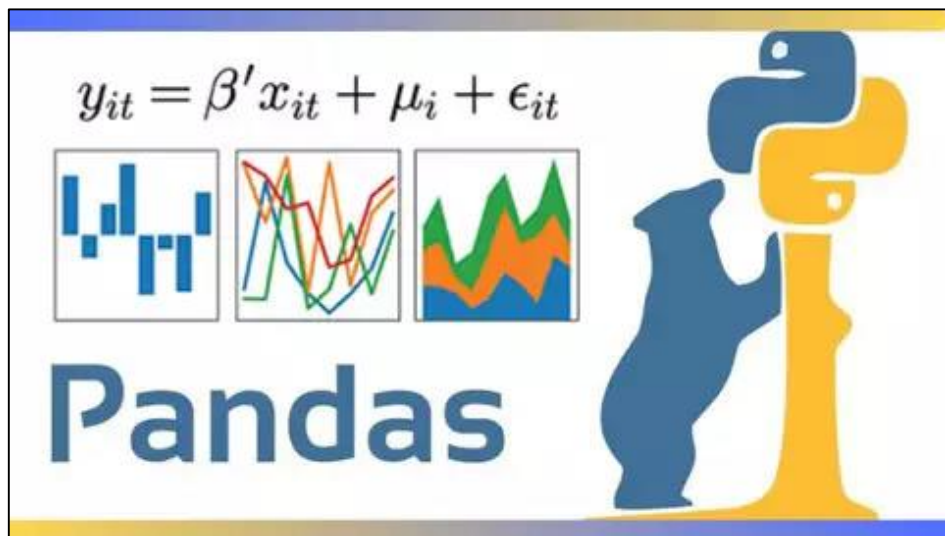
Python数据科学生态系统

# Pandas

- ❑ Panel data或Python data analysis, 基于NumPy, 是数据分析的基础包
  - 源自金融数据应用, 支持时间序列分析&非时间序列分析
  - 快速、便捷地处理结构化数据 (索引、数据对齐)
  - 灵活处理缺失数据



真正的熊猫

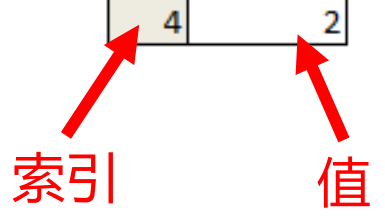


虚假的熊猫

# Pandas基本数据结构

- Series: 序列型数据结构 (一维) ; 索引+值
- DataFrame: 表格型数据结构 (二维或高维) ; 行/列+值

Series 1			Series 2			Series 3			DataFrame			
Mango			Apple			Banana			Mango	Apple	Banana	
0	4		0	5		0	2		0	4	5	2
1	5		1	4		1	3		1	5	4	3
2	6		2	3		2	5		2	6	3	5
3	3		3	0		3	2		3	3	0	2
4	1		4	2		4	7		4	1	2	7



索引      值

➤ 官方文档:

<https://pandas.pydata.org/pandas-docs/stable/reference/series.html>

<https://pandas.pydata.org/pandas-docs/stable/reference/frame.html>

# Pandas基本数据结构

## □ 构造DataFrame的办法有很多

### ➤ 常用：直接传入一个由等长列表或NumPy数组组成的字典

```
# 从“等长列表组成的字典”构造DataFrame
```

```
df = pd.DataFrame(  
    {  
        "Subject": ["军训", "思修", "史纲", "马原", "毛概", "体育1", "体育2"],  
        "Score": [65, 80, 75, 83, 77, 100, 98],  
        "Credit": [3, 3, 3, 4, 4, 1, 1],  
        "Year": [2018, 2018, 2019, 2019, 2020, 2019, 2020]  
    }  
)  
df
```

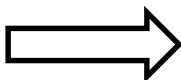
	Subject	Score	Credit	Year
0	军训	65	3	2018
1	思修	80	3	2018
2	史纲	75	3	2019
3	马原	83	4	2019
4	毛概	77	4	2020
5	体育1	100	1	2019
6	体育2	98	1	2020

# Pandas基本操作

▣ Series和DataFrame的基本操作：**花式**索引与切片、增删查改、排序.....

➤ 操作对象：索引、列名、单元格

	Subject	Score
0	军训	65
1	思修	80
2	史纲	75
3	马原	83
4	毛概	77
5	体育1	100
6	体育2	98



	Subject	Score	credit
0	军训	65	3
1	思修	80	3
2	线性代数	75	3
4	毛概	77	4
5	体育1	100	1
6	体育2	98	1

➤ 作业1：10 Minutes to pandas

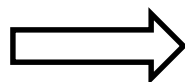
➤ 根据需要，现用现查

# Pandas函数应用与映射

- 分类求平均分、最高分？ **分组与聚合**
- 批量转换求GPA？ **应用自定义函数**

	subject	year	score	credits	property
0	机械制图D7	2018	90.0	5	限选
1	程序设计基础C9	2019	81.0	1	任选
2	电工电子技术C4	2017	77.0	1	必修
3	大学物理C3	2018	59.0	5	任选
4	程序设计基础A8	2017	93.0	5	限选
...	...	...	...	...	...
96	大学物理D7	2017	99.0	4	任选
97	电工电子技术B9	2018	73.0	1	限选
98	微积分A8	2018	67.0	4	必修
99	电工电子技术B5	2019	80.0	5	限选
100	大学物理A9	2018	81.0	2	必修

101 rows x 5 columns



	subject	year	score	credits	property	grade	GPA
29	电工电子技术D7	2018	100.0	1	限选	A+	4.0
51	体育C9	2017	100.0	3	必修	A+	4.0
57	线性代数C8	2020	100.0	1	限选	A+	4.0
96	大学物理D7	2017	99.0	4	任选	A	4.0
46	电工电子技术A8	2019	99.0	3	必修	A	4.0
...	...	...	...	...	...	...	...
10	程序设计基础C1	2018	55.0	3	任选	F	0.0
5	线性代数B9	2020	0.0	5	任选	F	0.0
31	机械制图A7	2019	0.0	5	限选	F	0.0
76	微积分D3	2019	0.0	1	限选	F	0.0
91	微积分D3	2020	0.0	5	限选	F	0.0

101 rows x 7 columns

- Pandas数据分析教程——超好用的Groupby用法详解：  
<https://www.jianshu.com/p/b50941b6d229>
- Pandas数据分析三板斧——map、apply、applymap详解：  
<https://www.jianshu.com/p/e76861ed1815>



# 第五讲-数据处理入门-目录

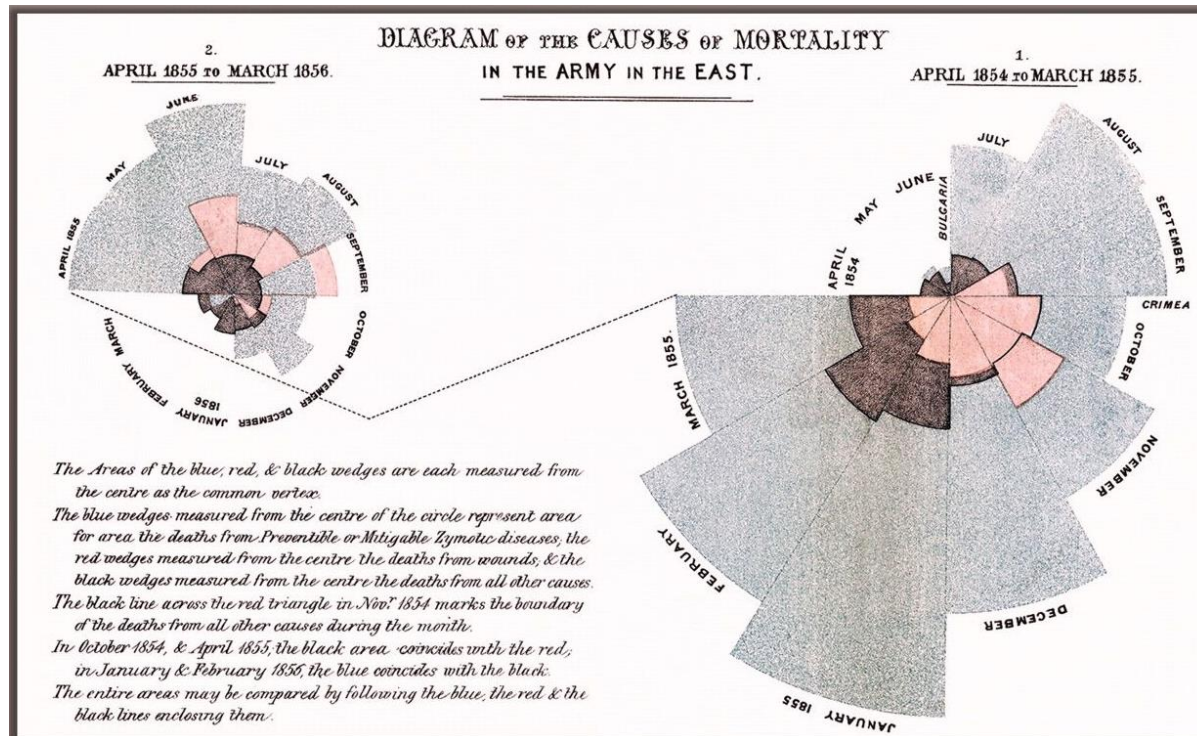
- 安装必备库
- 科学计算：NumPy
- 数据处理：pandas
- 数据可视化：matplotlib



**Python数据科学生态系统**

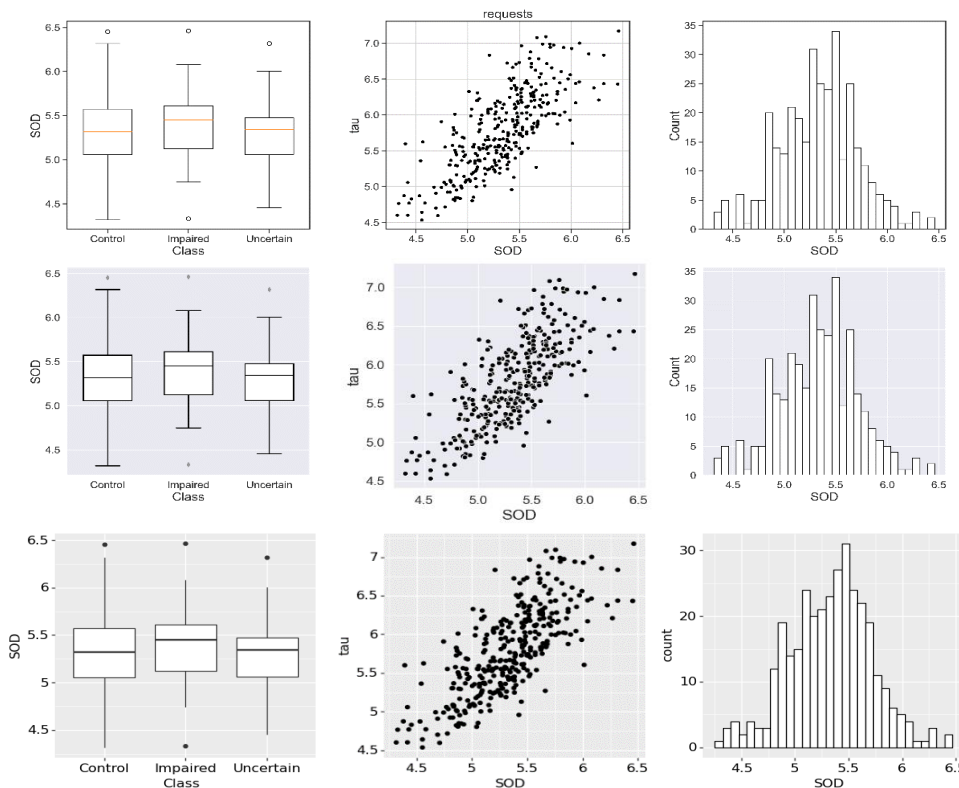
# 数据可视化基础

- 一图抵千言, A picture is worth a thousand words.
- 可视化的作用: 真实、准确地展示数据; 揭示数据的关系、规律
- 故事: 南丁格尔玫瑰图



# 数据可视化常用工具

- Python: **matplotlib**, Seaborn, plotline
- R: ggplot2
- 软件或在线工具: Excel, Power BI, Echart, Tableau



matplotlib

seaborn

plotline

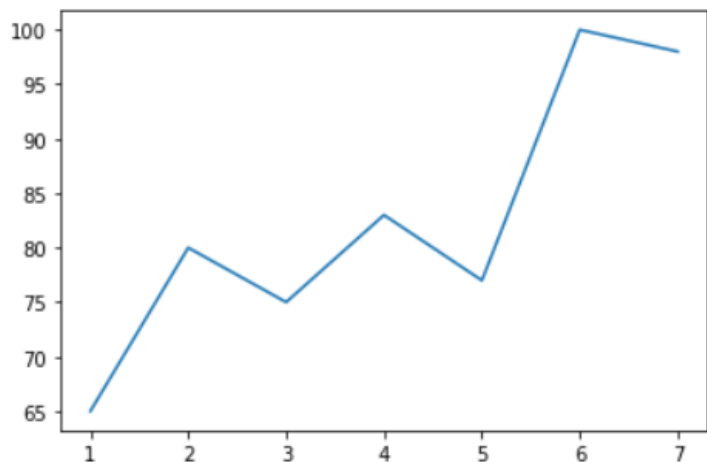
# Matplotlib绘图

## □ 基本操作：从序列画图

▶ MI

# 直接从列表画图

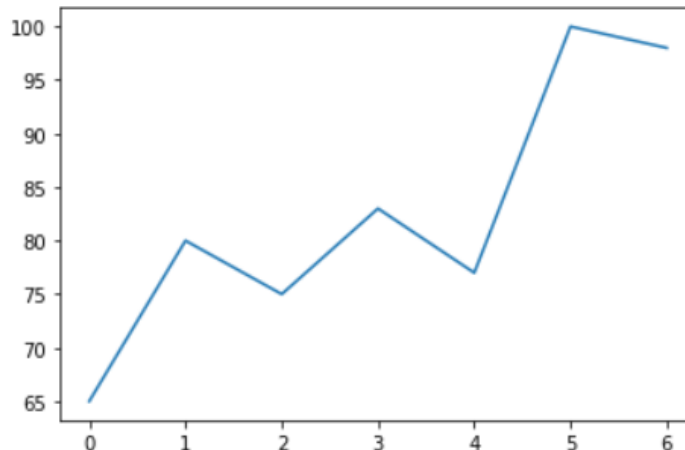
```
x = [1,2,3,4,5,6,7]
y = [65, 80, 75, 83, 77, 100, 98]
plt.plot(x,y)
plt.show()
```



▶ MI

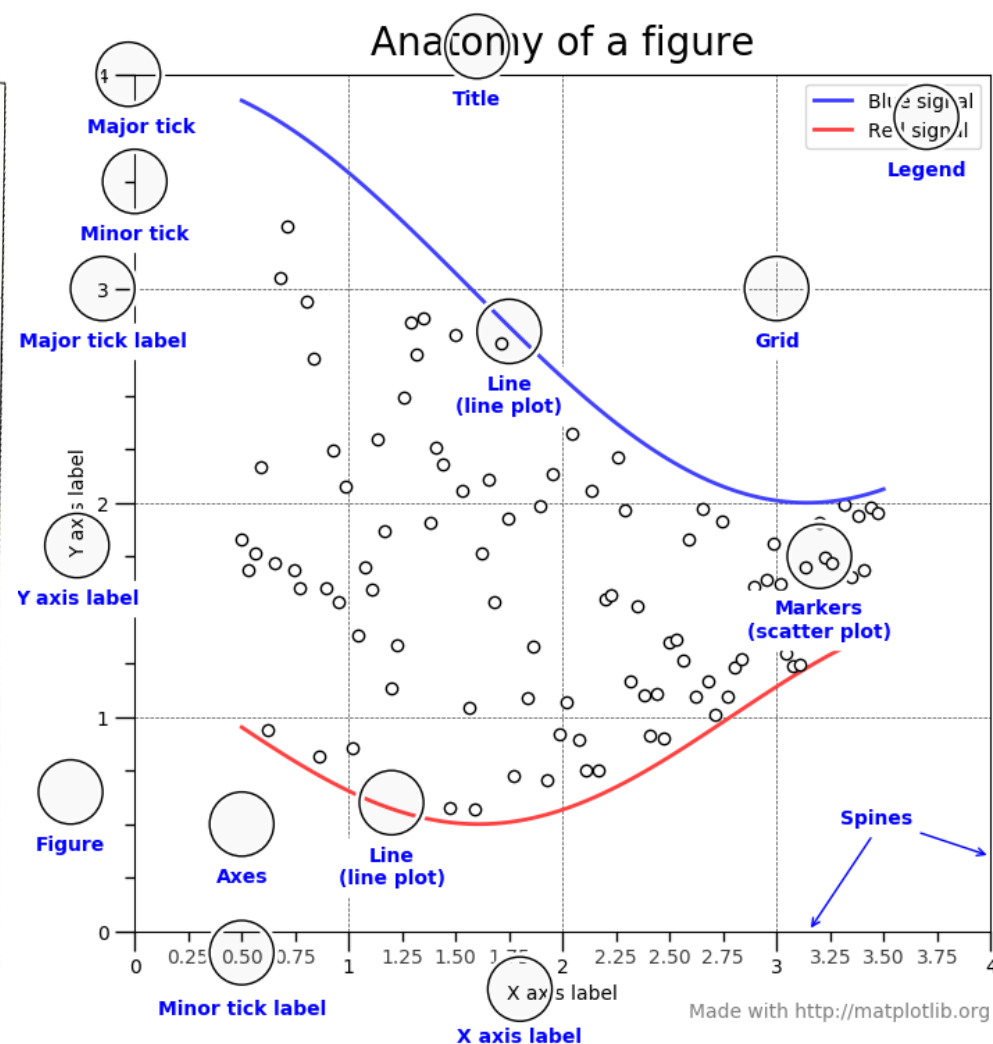
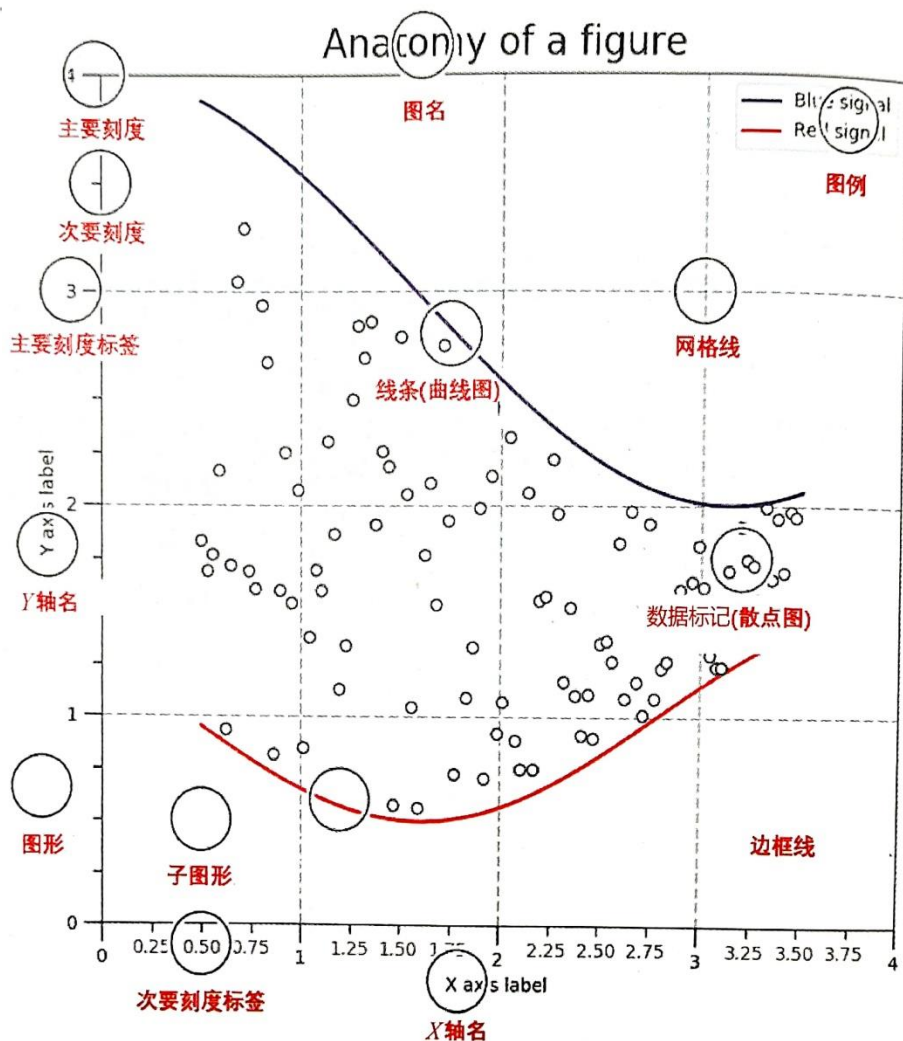
# 从NumPy画图

```
x = np.arange(0,7)
y = np.array([65, 80, 75, 83, 77, 100, 98])
plt.plot(x,y)
plt.show()
```



## □ 完善细节：调整线条、坐标轴、图框，添加标注

# Matplotlib图像构成





# Matplotlib图像构成

## □ 常用设置:

- 线条: 颜色、标记、线型.....
- 图框: 刻度、标签、图例、网格、注释.....

ID	函数	核心参数说明	功能
1	figure()	figsize ( 图表尺寸 )、dpi ( 分辨率 )	设置图表的大小与分辨率
2	title()	str ( 图名 )、fontdict ( 文本格式, 包括字体大小、类型等 )	设置标题
3	xlabel()、ylabel()	xlabel ( $X$ 轴名 ) 或 ylabel ( $Y$ 轴名 )	设置 $X$ 轴和 $Y$ 轴的标题
4	axis()、xlim()、ylim()	xmin、xmax 或 ymin、ymax	设置 $X$ 轴和 $Y$ 轴的范围
5	xticks()、yticks()	ticks ( 刻度数值 )、labels ( 刻度名称 )、fontdict	设置 $X$ 轴和 $Y$ 轴刻度
6	grid()	b ( 有无网格线 )、which ( 主/次网格线 )、axis ( $X$ 轴和 $Y$ 轴网格线 )、color、linestyle、linewidth、alpha ( 透明度 )	设置 $X$ 轴和 $Y$ 轴的主要和次要网格线
7	legend()	loc ( 位置 )、edgecolor、facecolor、fontsize	控制图例显示

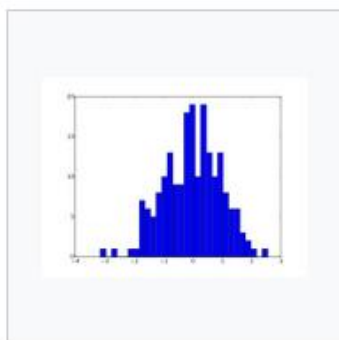
- 根据需要, 现用现查

# Matplotlib图像种类

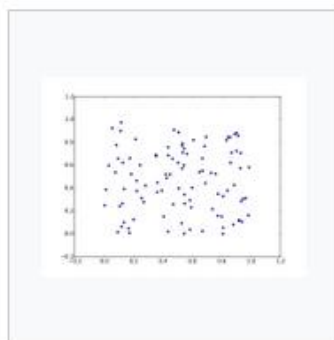
□ 根据需求，确定图像的形式



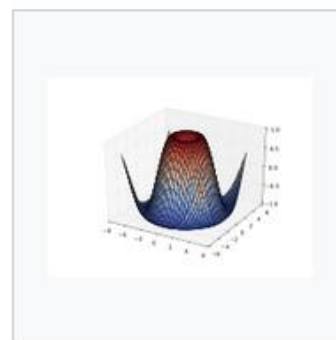
Line plot



Histogram



Scatter plot



3D plot

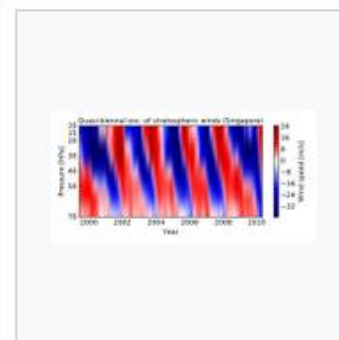
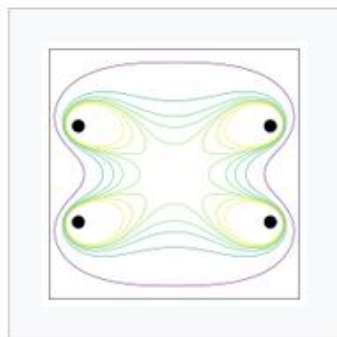
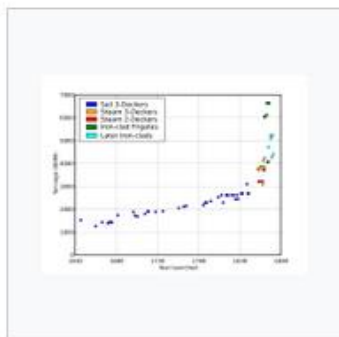


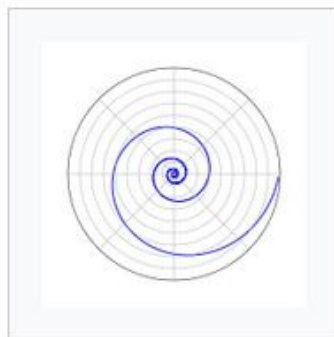
Image plot



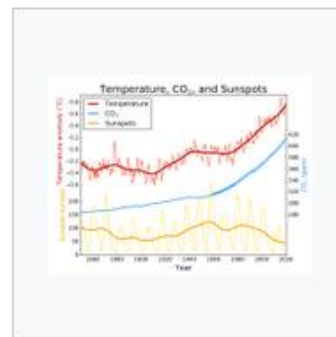
Contour plot



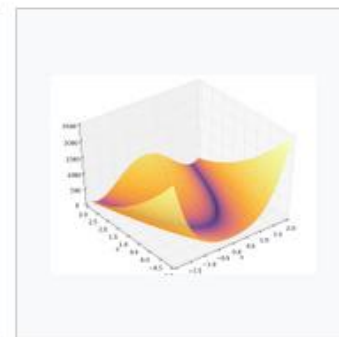
Scatter plot



Polar plot



Line plot



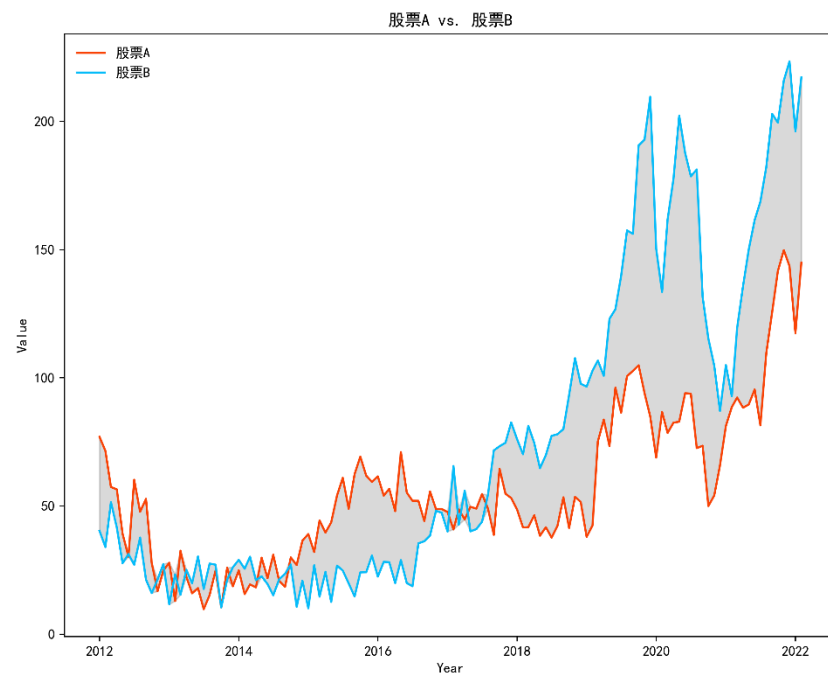
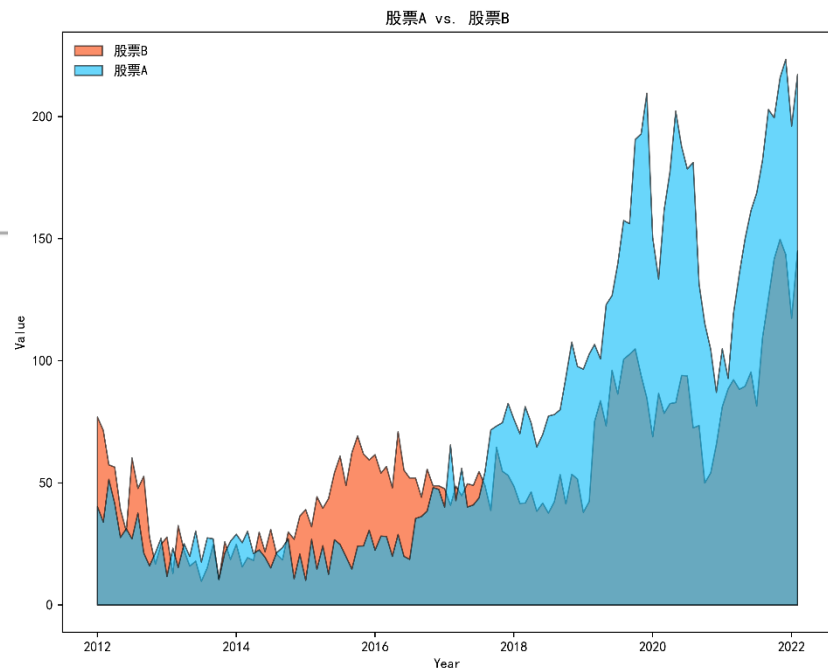
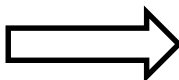
3-D plot

# 时间序列绘制举例

## □ Pandas整理数据→Matplotlib绘图

	股票A	股票B
date		
2012-1-1	76.985291	40.301222
2012-2-1	71.506866	33.981832
2012-3-1	57.410031	51.475666
2012-4-1	56.485801	41.560597
2012-5-1	39.225449	27.668570
...	...	...
2021-10-1	141.698789	199.537386
2021-11-1	149.804607	215.963365
2021-12-1	143.574939	223.464030
2022-1-1	117.443904	196.151518
2022-2-1	144.976472	217.230120

122 rows x 2 columns

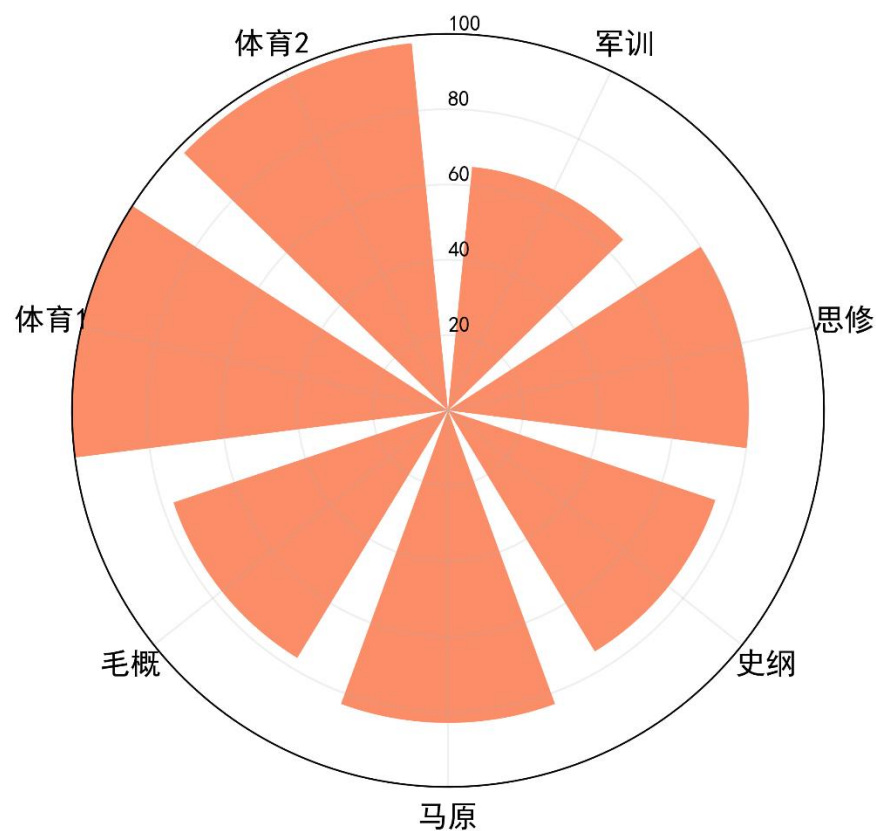
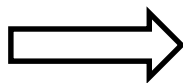


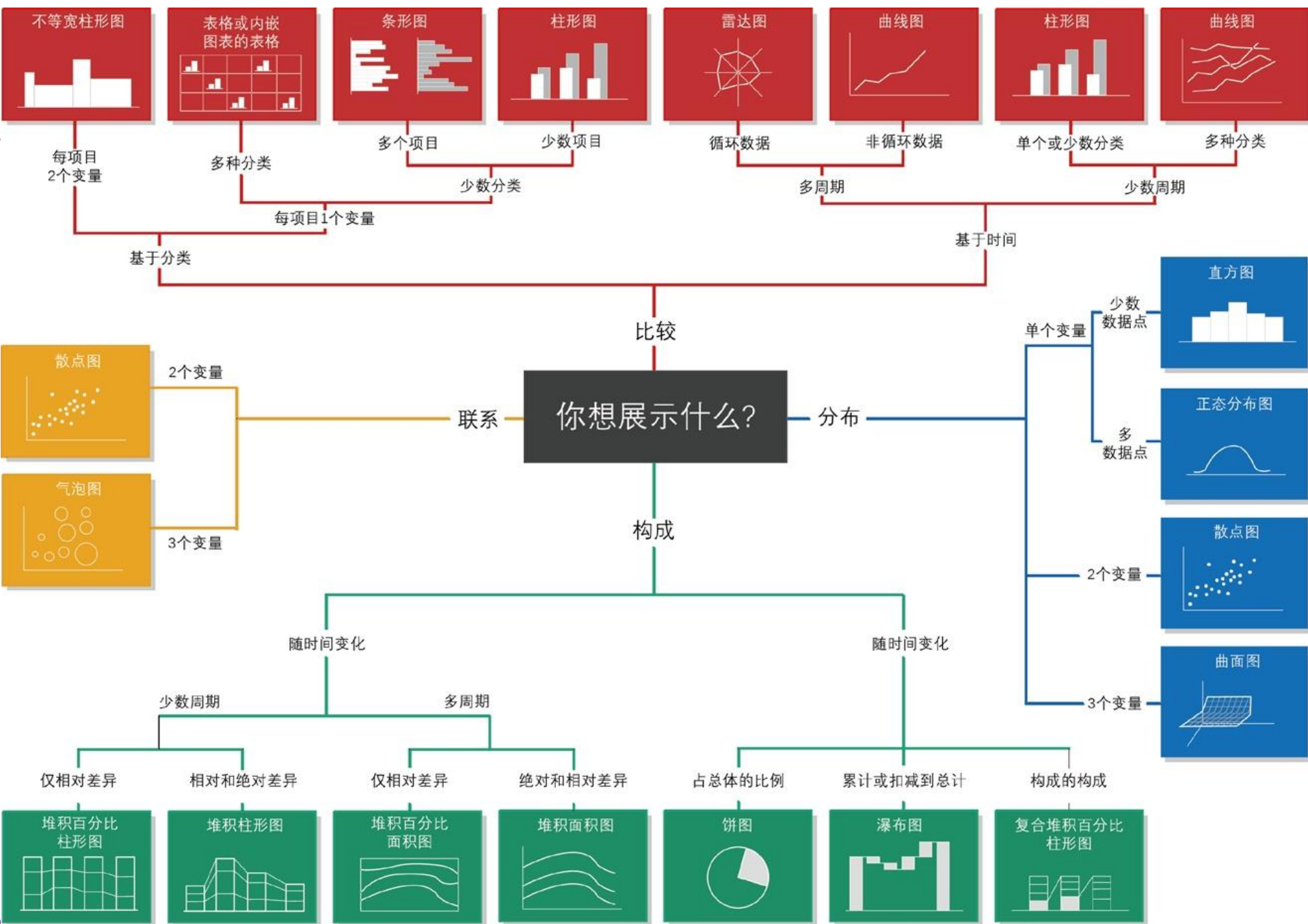


# 南丁格尔图绘制举例

□ Pandas整理数据→Matplotlib绘图

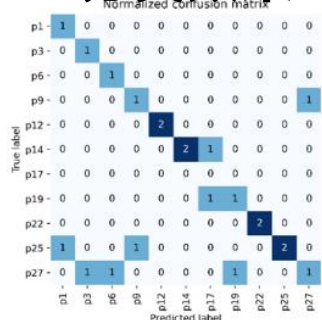
	Subject	Grade
0	军训	65
1	思修	80
2	史纲	75
3	马原	83
4	毛概	77
5	体育1	100
6	体育2	98



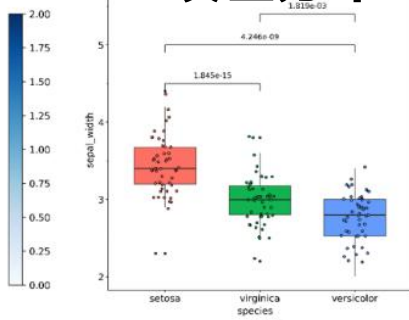


# 数据可视化应用举例

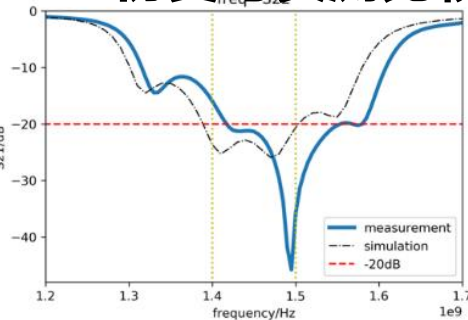
## 分类效果



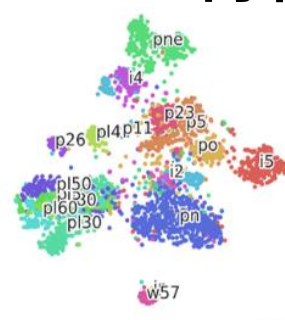
## 误差分布



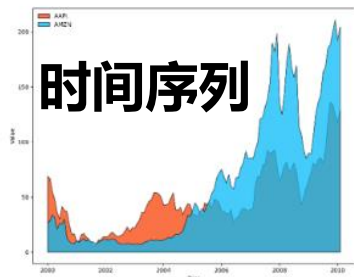
## 仿真与实测比较



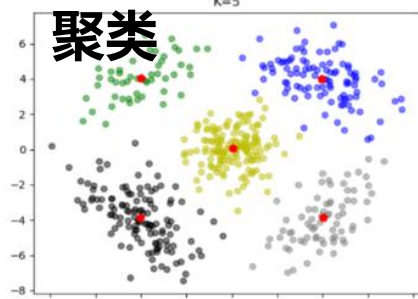
## t-sne高维特征



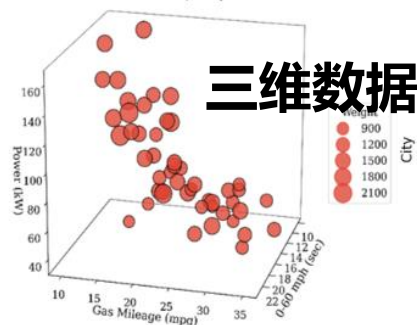
## 时间序列



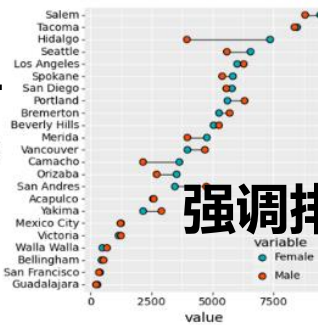
## 聚类



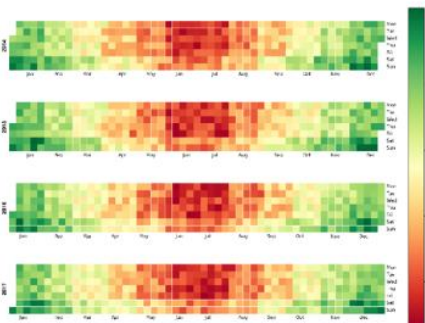
## 三维数据



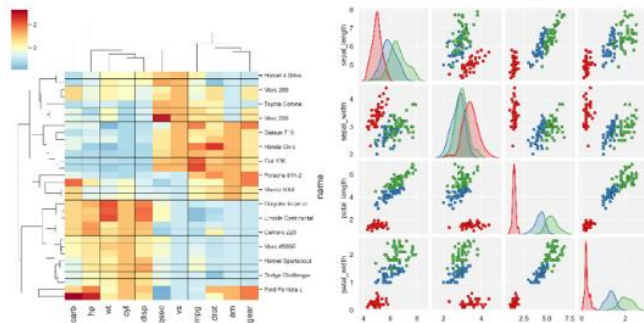
## 强调排序和差距



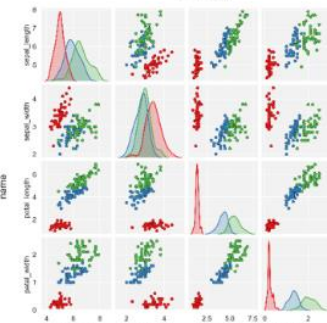
## 日期活跃情况



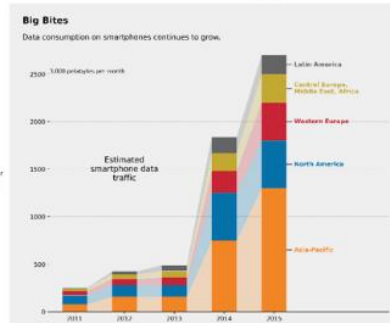
## 层次聚类



## 相关度



## 变化趋势



# 课后练习

---

- 1. (基本要求) 阅读并实现：十分钟入门 Pandas  
[https://www.py pandas.cn/docs/getting\\_started/10min.html](https://www.py pandas.cn/docs/getting_started/10min.html)
- 2. (基本要求) 用matplotlib画一幅图像，鼓励使用自己的数据

# 反馈问卷

