



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌

# 第1章 数据可视化与matplotlib



- 数据可视化概述
- 常见的数据可视化库
- 初识matplotlib
- 使用matplotlib绘制第一个图表

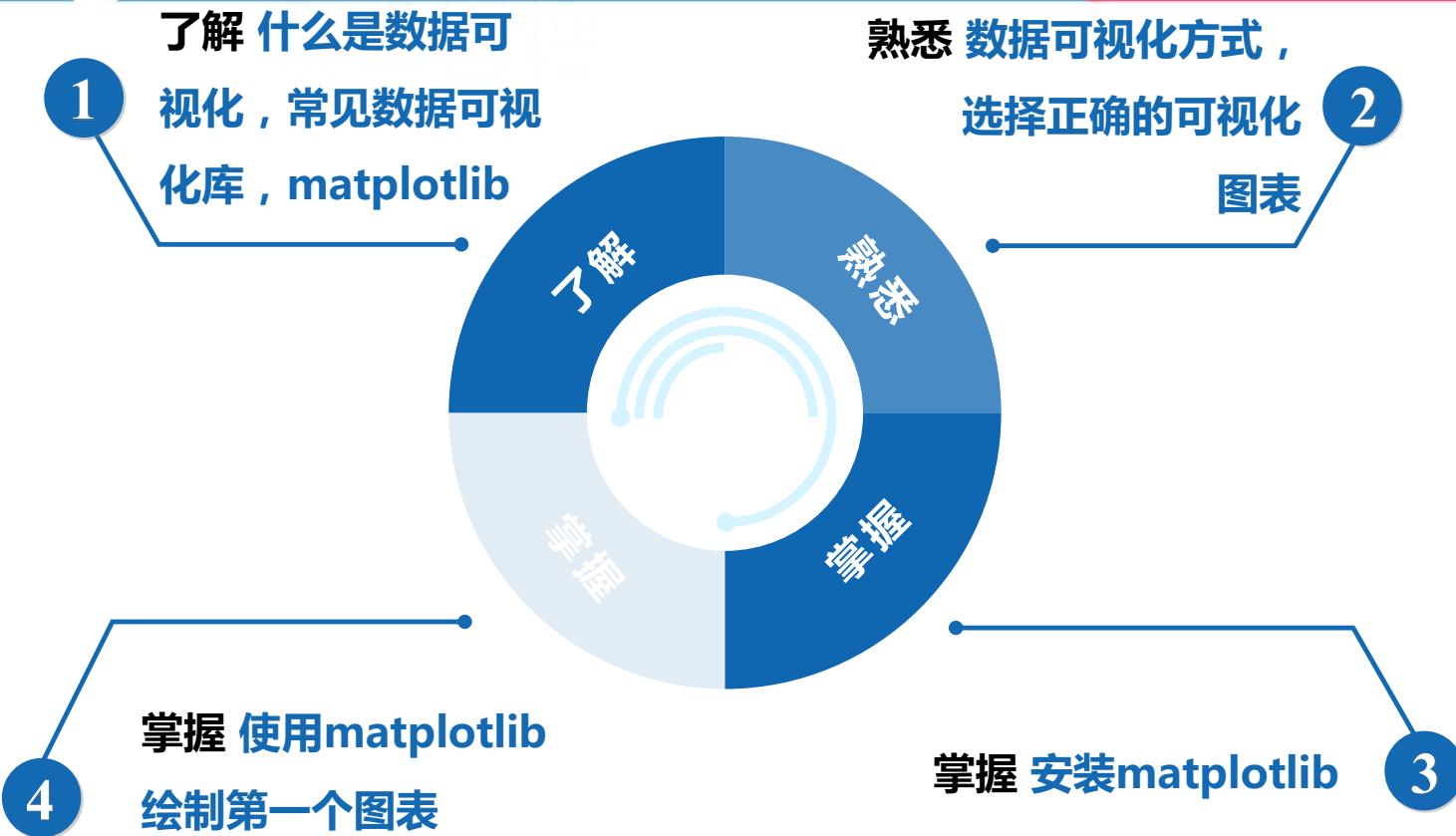


# 学习目标



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌





# 目录页



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌



- 1.1 数据可视化概述
- 1.2 常见的数据可视化库
- 1.3 初识**matplotlib**
- 1.4 使用**matplotlib**绘制图表





# 目录页



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌



## 1.1 数据可视化概述

## 1.2 常见的数据可视化库

## 1.3 初识**matplotlib**

## 1.4 使用**matplotlib**绘制图表

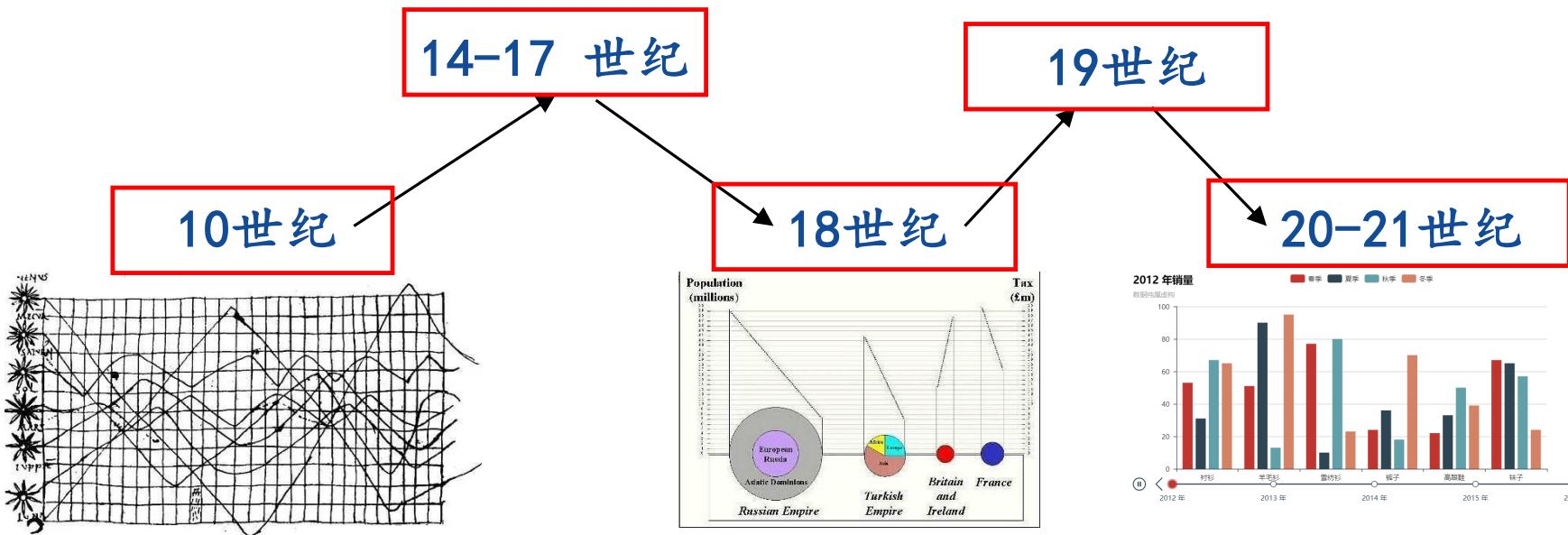




## 1.1.1 什么是数据可视化



数据可视化有着非常久远的历史，最早可以追溯至10世纪，至今已经应用和发展了数百年。





## 1.1.1 什么是数据可视化



- **什么是数据可视化？**

数据可视化旨在借助图形化的手段，将**一组数据以图形的形式表示**，并利用数据分析和开发工具发现其中未知信息的处理过程。

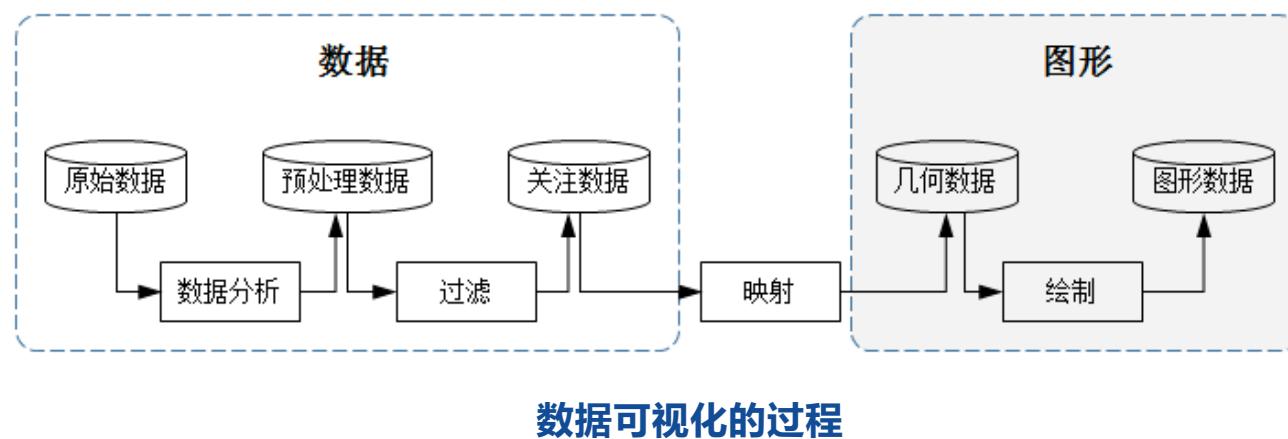




## 1.1.1 什么是数据可视化



可视化其实是一个**抽象**的过程，它可以简单地理解为将一个不易描述的事物形成一个可感知画面的过程，也就是从数据空间到图形空间的**映射**。





### 1.1.1 什么是数据可视化



无论原始数据被映射为哪种图形数据，最终要达到的目的只有一个，便是准确地、高效地、全面地传递信息，进而建立起数据间的关系，使人们发现数据间的规律和特征，并挖掘出有价值的信息，提高数据沟通的效率。



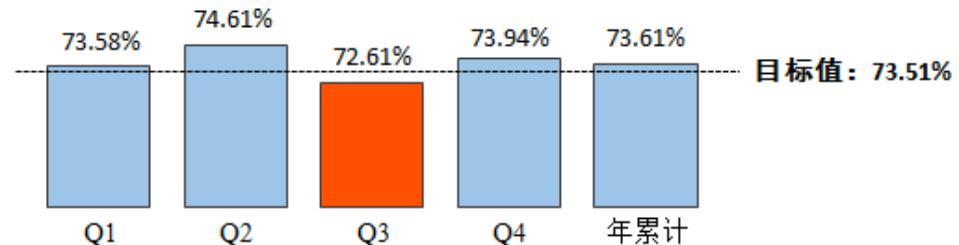
### 1.1.1 什么是数据可视化



假设某公司员工在整理全年KPI报告时准备了表格和图形这两种形式的数据。

季度	实际值	目标值	差距值
Q1	73.58%	73.51%	0.07%
Q2	74.61%	73.51%	1.10%
Q3	72.61%	73.51%	-0.90%
Q4	73.94%	73.51%	0.43%
年累计	73.61%	73.51%	0.10%

表格可以帮助公司领导快速地知道各季度的具体数值，但无法快速地了解各季度之间的比较情况。



图形可以帮助公司高层准确地了解各季度之间的比较情况，方便对公司下一年的工作做出有效地决策。



### 1.1.1 什么是数据可视化



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌

总

结

综上所述，数据可视化是数据分析工作中重要的一环，对数据潜在价值的挖掘有着深远的影响。随着数据可视化**平台的拓展、表现形式的变化**，以及实时**动态效果、用户交互**使用等功能的增加，数据可视化的内涵正在不断扩大，相信数据可视化的应用领域会越来越广泛。

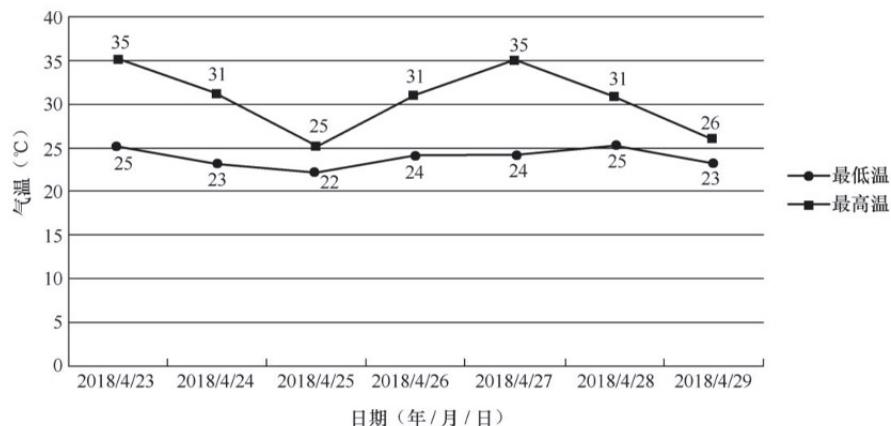


## 1.1.2 常见的数据可视化方式



- 折线图

折线图是将数据标注成点，并通过直线将这些点按某种顺序连接而成的图表，它以折线的倾斜程度来形象地反映事物沿某一维度的变化趋势，能够清晰地展示数据增减的趋势、速率、规律及峰值等特征。



海口市4月23—29日的最高气温和最低气温的变化情况

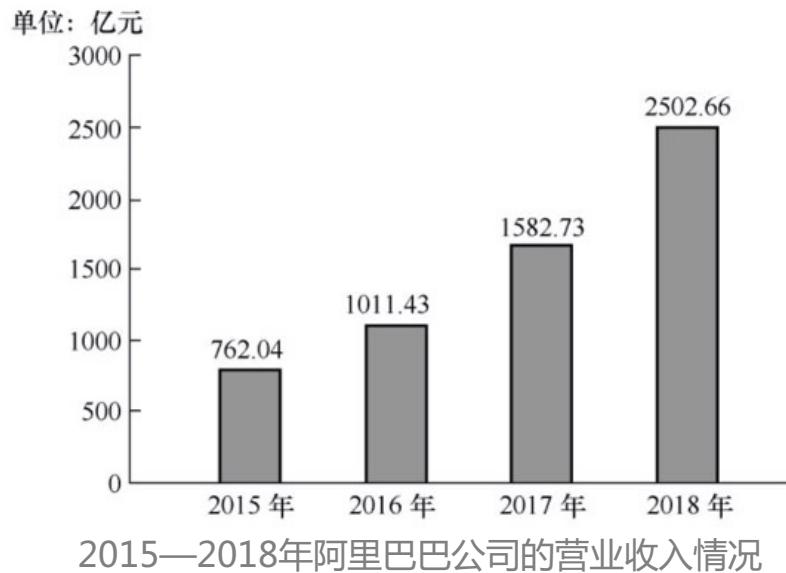


## 1.1.2 常见的数据可视化方式



### • 柱形图

柱形图是由一系列宽度相等的纵向矩形条组成的图表，它使用矩形条的高度表示数值，以此反映不同分类数据之间的差异。



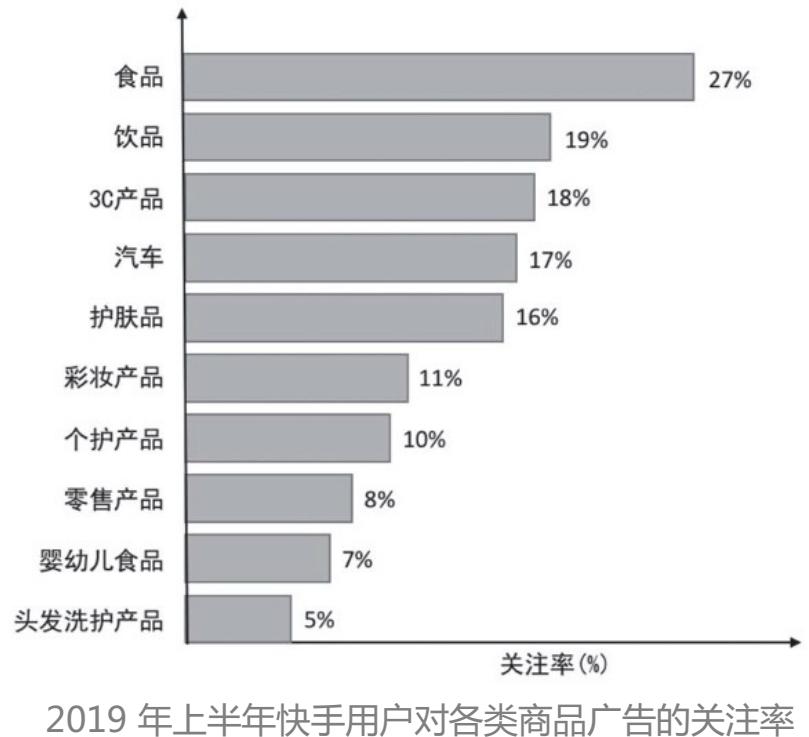


## 1.1.2 常见的数据可视化方式



### • 条形图

条形图是横置的柱形图，由一系列高度相等、长短不一的横向矩形条组成。

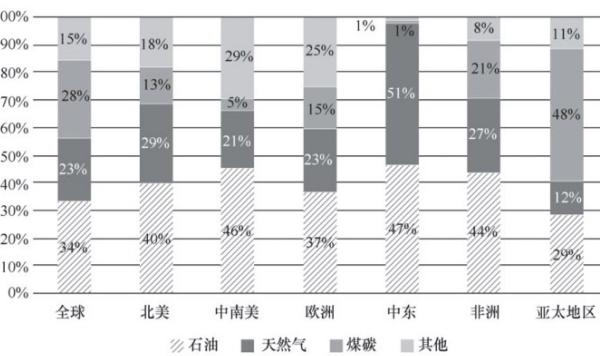
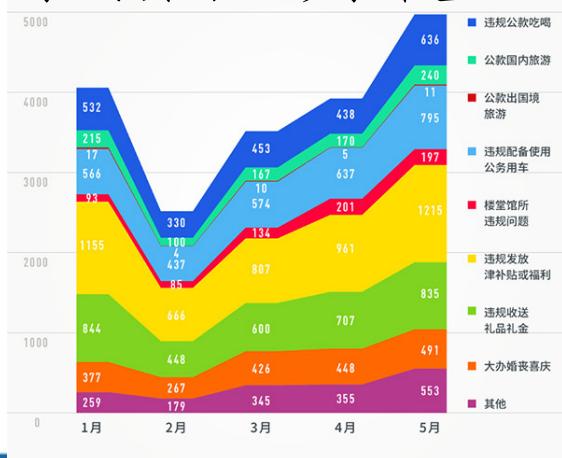




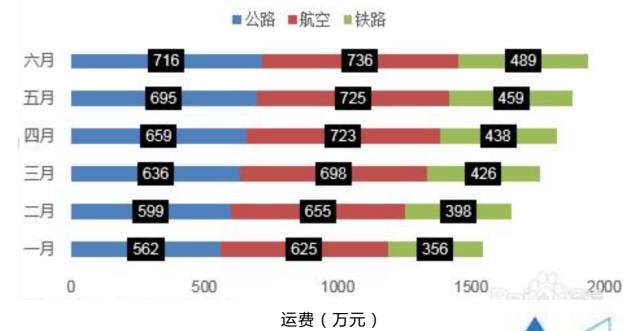
## 1.1.2 常见的数据可视化方式

- ### 堆积图

堆积图可分为**堆积面积图**、**堆积柱形图**和**堆积条形图**。其中堆积面积图是由若干条折线与折线或水平坐标轴之间的填充区域组成的图表，用于强调每个部分变化的趋势；堆积柱形图和堆积条形图是由若干个以颜色或线条填充、高度不一的纵向矩形条或横向矩形条堆叠而成的图表，主要用于反映各构成部分在总体中的比重。



2017 年全球及各地区一次性能源的消费结构



某物流公司2014年各运输方式的运费情况

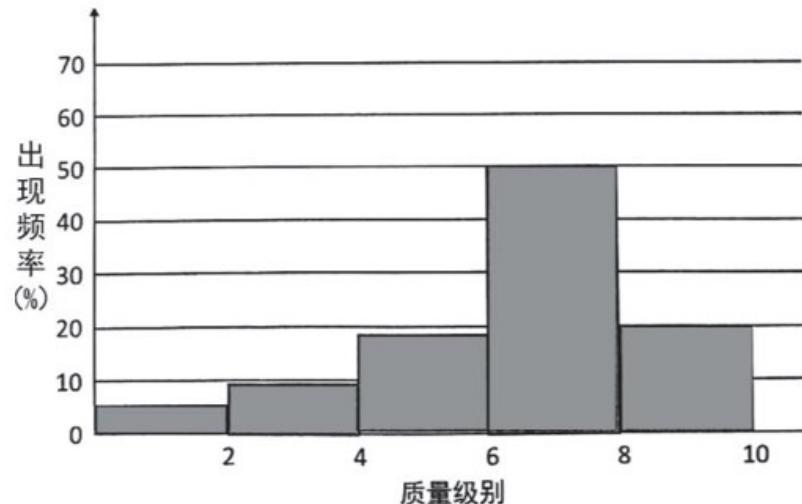


## 1.1.2 常见的数据可视化方式



### • 直方图

直方图又称质量分布图，是由一系列高低不等的纵向矩形条或线段组成的图表，用于反映数据的分布和波动情况。



某厂商对 100 个抽样产品的质量级别评定情况



## 1.1.2 常见的数据可视化方式



**柱形图与直方图的区别**包括以下两点：

- 柱形图用于展示离散型数据的分布，而直方图用于展示连续型数据的分布；
- 柱形图的各矩形条之间具有固定的间隙，而直方图的各矩形条之间没有任何间隙。



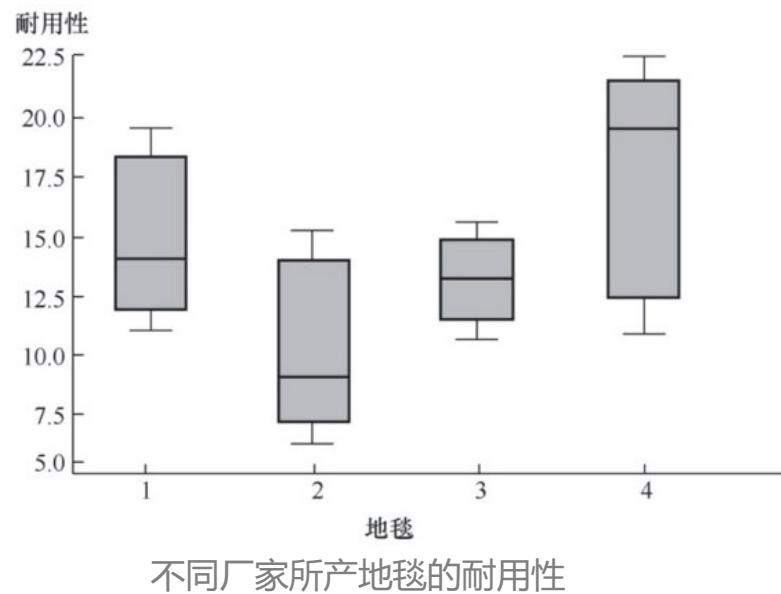


## 1.1.2 常见的数据可视化方式



### • 箱形图

箱形图又称盒须图、箱线图，是一种利用数据中的5个统计量——最小值、下四分位数、中位数、上四分位数和最大值——描述数据的图表，主要用于反映一组或多组数据的对称性、分布程度等信息，因形状如箱子而得名。

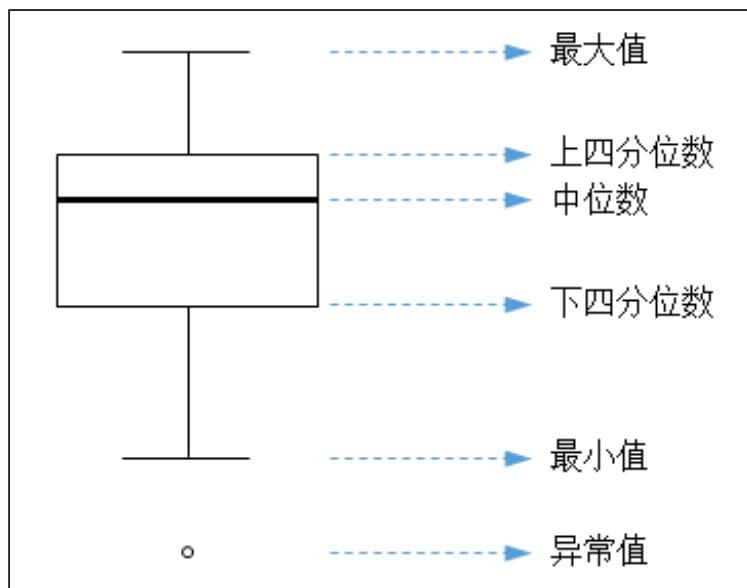




## 1.1.2 常见的数据可视化方式



箱形图中每个图形的结构是相同的，包括一个矩形**箱体**、上下**两条竖线**、上下**两条横线**。



- 箱体代表数据的集中范围；
- 上下两条竖线分别代表数据向上和向下的延伸范围；
- 上下两条横线分别代表最大值和最小值；
- 圆圈代表异常值（也称为离群值）。

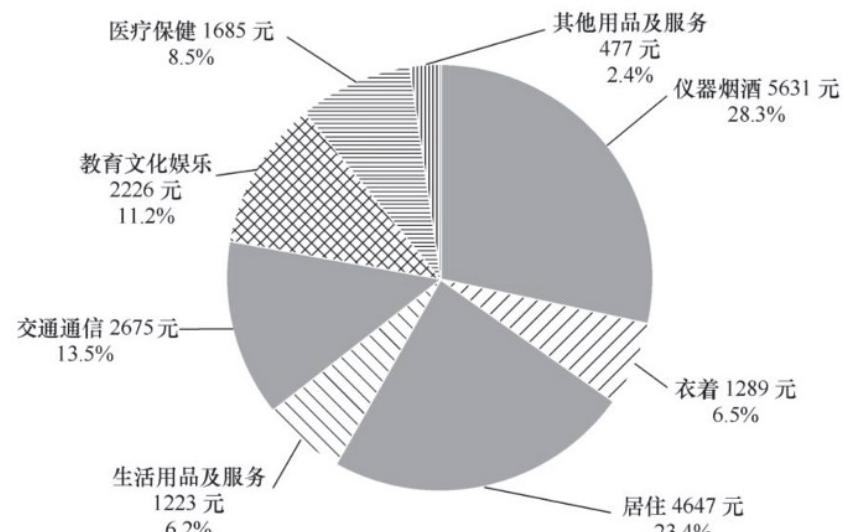


## 1.1.2 常见的数据可视化方式



### • 饼图

饼图是由若干个面积大小不一、颜色不同的**扇形**组成的圆形图表，它使用圆表示数据的总量，组成圆的每个扇形表示数据中各项占总量的比例大小，主要用于显示数据中**各项大小与各项总和的比例**。



2018年全国居民的人均消费支出情况



## 1.1.2 常见的数据可视化方式



圆环图也能反映各项与整体之间的关系，但它使用圆环表示整体，组成圆环的每个楔形表示各项占整体的比例大小，外形像空心的圆饼。

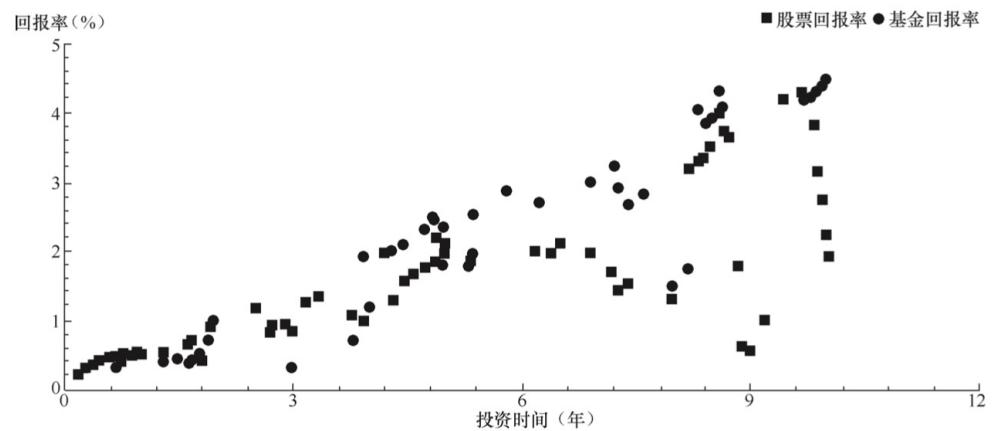


## 1.1.2 常见的数据可视化方式



### • 散点图

散点图又称X-Y图，是由若干个数据点组成的图表，主要用于判断两变量之间是否存在某种关联，或者总结数据点的分布模式。





## 1.1.2 常见的数据可视化方式



散点图中数据点的分布情况可以体现**变量之间的相关性**。

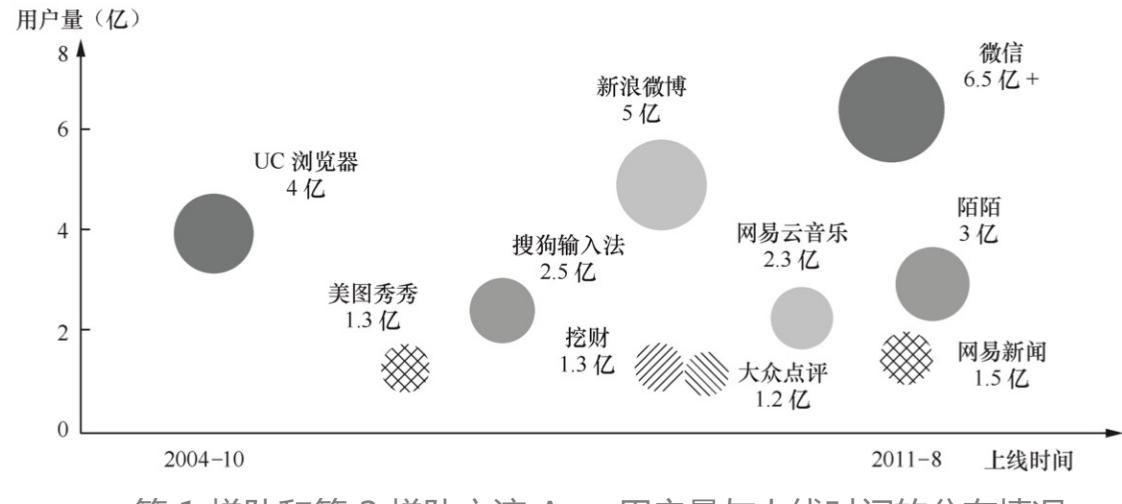
- 若所有的数据点在一条直线附近呈波动趋势，说明变量之间是线性相关的。
- 若数据点在曲线附近呈波动趋势，说明变量之间是非线性相关的。
- 若数据点没有显示任何关系，说明变量之间是不相关的。



## 1.1.2 常见的数据可视化方式

### • 气泡图

气泡图是散点图的变形，它是一种能够展示**多变量关系**的图表。气泡图一般使用两个变量标注气泡在坐标系中的位置，使用第3个变量标注气泡的面积，适用于分类数据对比、多变量相关性等情况。



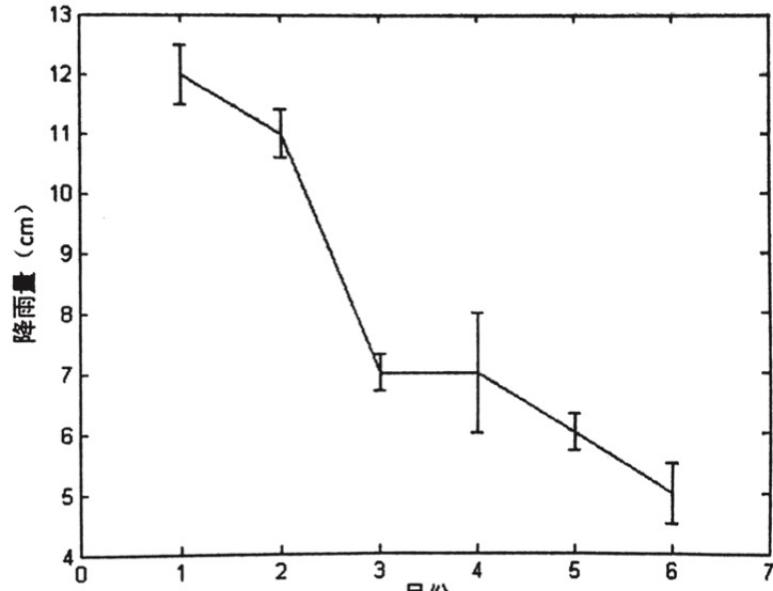


## 1.1.2 常见的数据可视化方式



### • 误差棒图

误差棒图是使用**误差棒**注明被测量数据的不确定度大小的图表，用于表示测量数据中客观存在的**测量偏差**。误差棒是在表示测量值大小的方向上的一条线段，它以被测量数据的平均值为中点，线段长度的一半为不确定度。



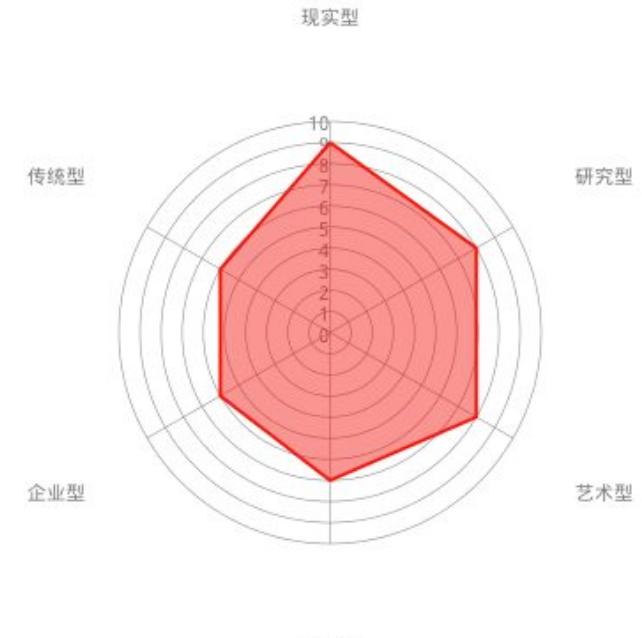


## 1.1.2 常见的数据可视化方式



### • 雷达图

雷达图又称蜘蛛网图、星状图、极区图，由一组坐标轴和多个等距同心圆或多边形组成，是一种表现多维（4维以上）数据的图表。



某人通过霍兰德职业兴趣测试的结果

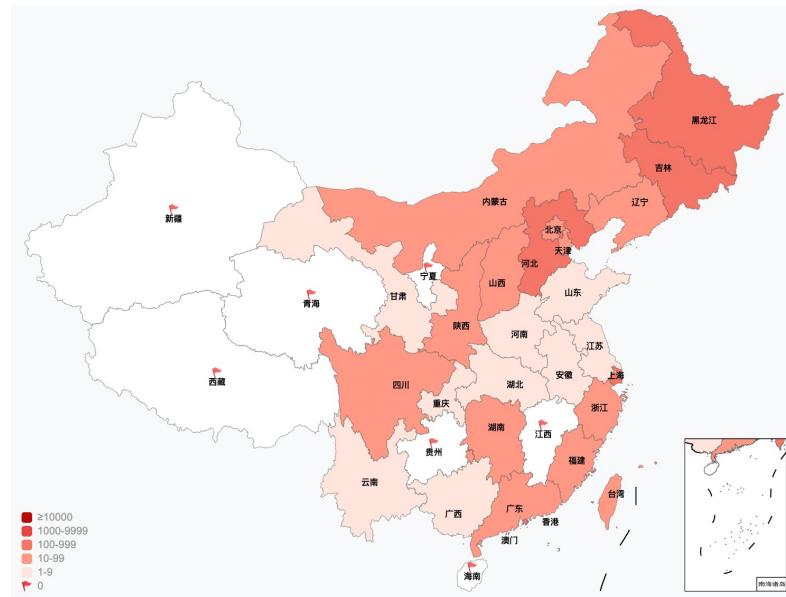


## 1.1.2 常见的数据可视化方式



### • 统计地图

统计地图是一种以地图为背景，使用各种线纹、色彩、几何图形或实物形象标注指标数值的大小及其在不同地理位置的分布状况的图表。



2021年2月1日中国新型冠状病毒肺炎疫情地图

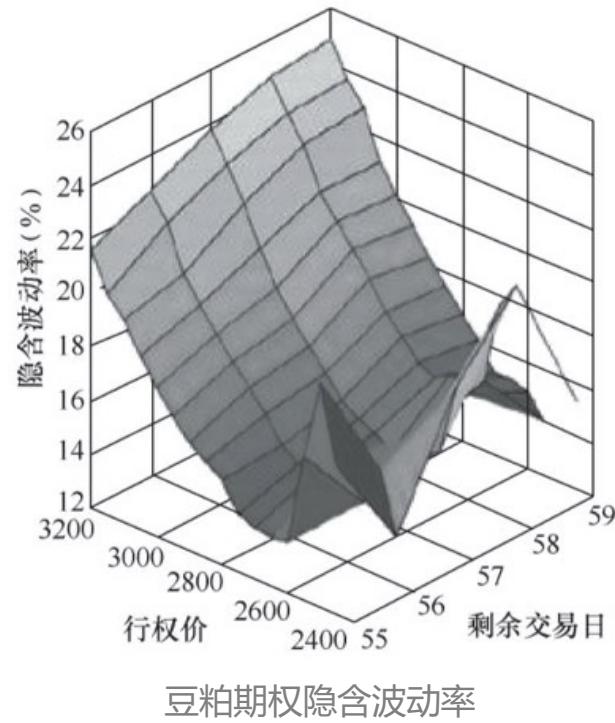


## 1.1.2 常见的数据可视化方式



### • 3D图表

3D图表是一类在三维坐标系中呈现数据的图表，常用的图表包括3D散点图、3D折线图、3D曲面图、3D直方图、3D柱形图等。





### 1.1.3 选择正确的可视化图表



数据通常包含4种关系：**比较、分布、构成和联系。**

比较关系主要关注数据中各类别或时间变化的情况。

比较

分布关系主要关注不同数值范围内包含多少数据的情况。

分布

构成关系主要关注各部分与整体占比的情况。

构成

联系关系主要关注两个及两个以上的变量之间关系的情况。

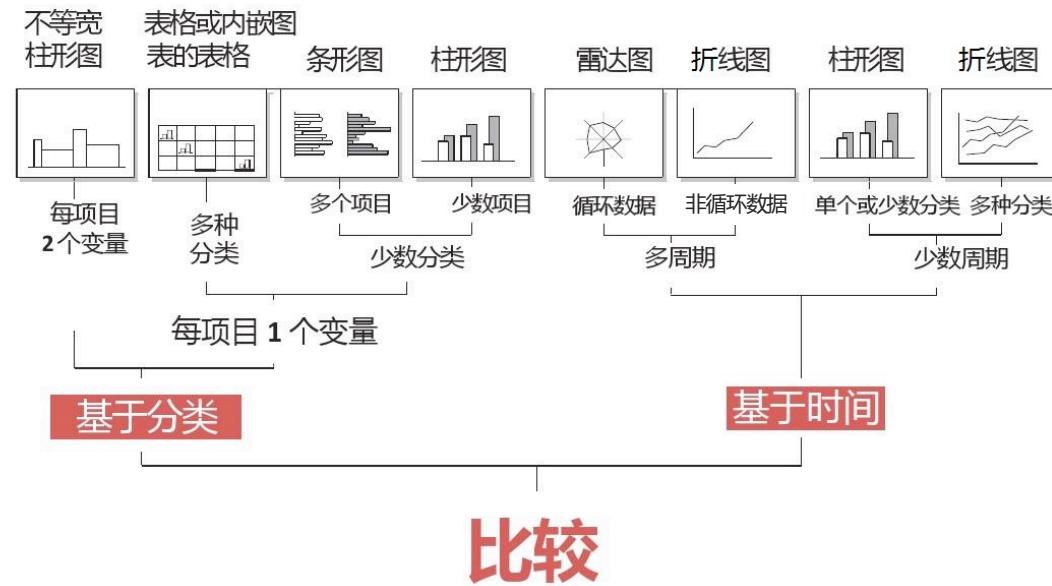
联系



### 1.1.3 选择正确的可视化图表



- 基于比较关系可选择的图表

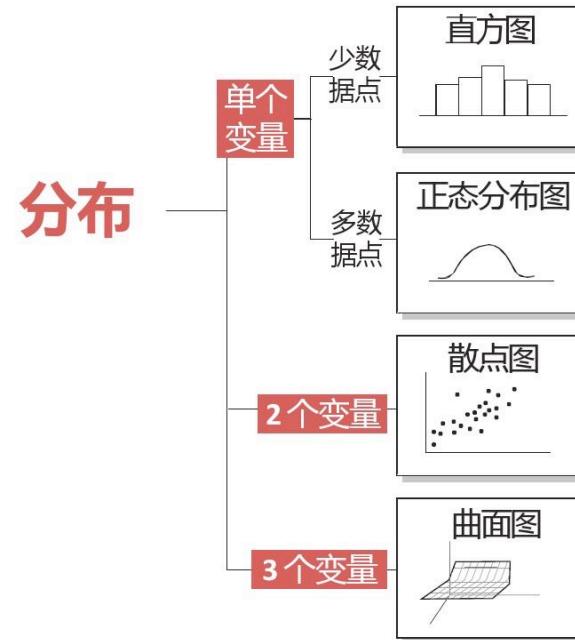




### 1.1.3 选择正确的可视化图表



- 基于分布关系可选择的图表

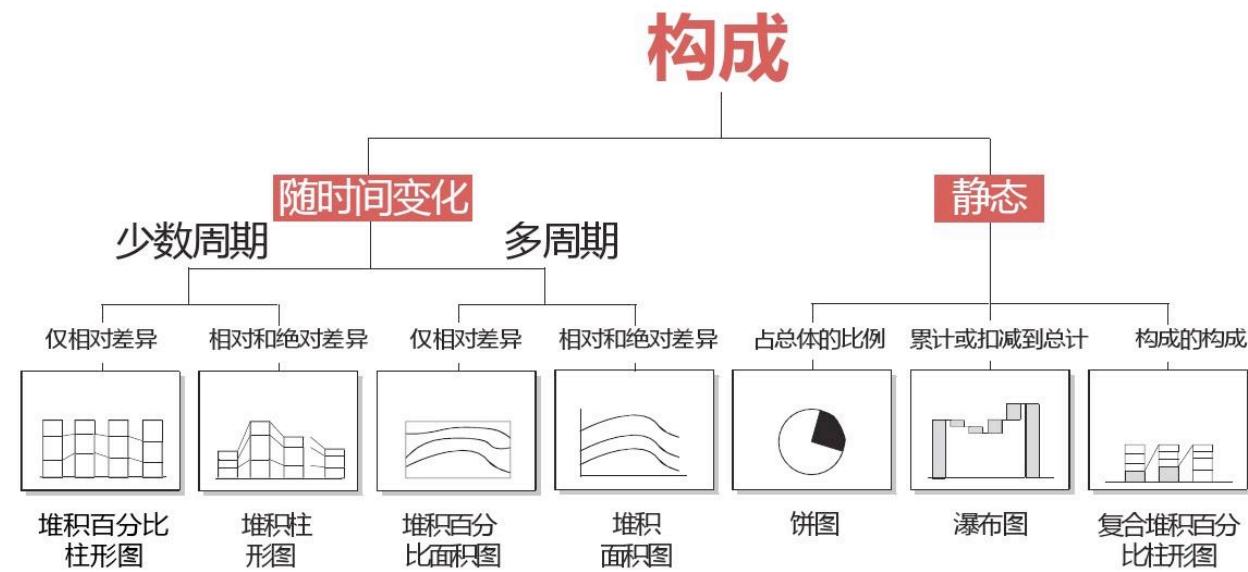




### 1.1.3 选择正确的可视化图表



- 基于构成关系可选择的图表



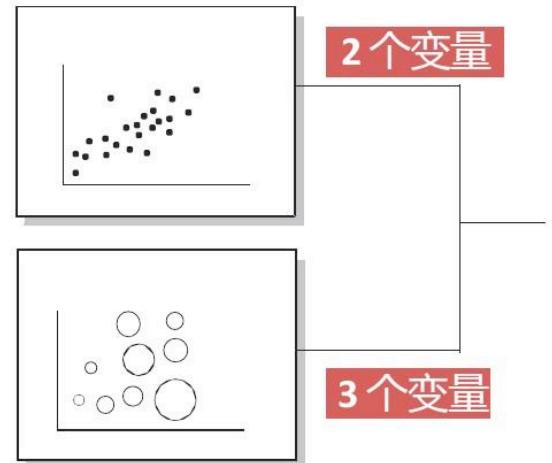


### 1.1.3 选择正确的可视化图表



- 基于联系关系可选择的图表

散点图



联系

气泡图



# 目录页



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌



**1.1 数据可视化概述**

**1.2 常见的数据可视化库**

**1.3 初识matplotlib**

**1.4 使用matplotlib绘制图表**





## 1.2 常见的数据可视化库



Python作为数据分析的首选语言，它针对数据分析的每个环节都提供了很多库。常见的数据可视化库包括matplotlib、seaborn、ggplot、bokeh、pygal、pyecharts。



## 1.2 常见的数据可视化库



matplotlib是众多Python数据可视化库的鼻祖，其设计风格与20世纪80年代设计的商业化程序语言MATLAB十分接近，具有很多强大且复杂的可视化功能。matplotlib包含多种类型的API，可以采用多种方式绘制图表并对图表进行定制。





## 1.2 常见的数据可视化库



- **seaborn**是基于matplotlib进行高级封装的可视化库，它支持**交互式界面**，使得绘制图表的功能变得愈加容易，且图表的色彩更具吸引力，可以画出丰富多样的统计图表。
- **ggplot**是基于matplotlib并旨在以简单方式提高matplotlib可视化感染力的库，它采用**叠加图层的形式绘制图形**，例如先绘制坐标轴所在的图层，再绘制点所在的图层，最后绘制线所在的图层，但其并不适用于个性化定制图形。



## 1.2 常见的数据可视化库



- **bokeh**是一个交互式的可视化库，它支持**使用Web 浏览器**展示，可使用快速简单的方式将大型数据集转换成高性能的、可交互的、结构简单的图表。
- **pygal**是一个可缩放矢量图表库，用于生成可在浏览器中打开的SVG格式的图表，这种图表可以在不同比例的屏幕上自动缩放，方便用户交互。
- **pyecharts**是一个生成Echarts Enterprise Charts（商业产品图表）图表的库，它生成的Echarts图表凭借良好的交互性、精巧的设计得到了众多开发者的认可。



# 目录页



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌



**1.1 数据可视化概述**

**1.2 常见的数据可视化库**

**1.3 初识matplotlib**

**1.4 使用matplotlib绘制图表**





## 1.3.1 matplotlib概述



matplotlib是一个由John D.Hunter等人员开发的、主要用于绘制2D图表的Python库。

matplotlib支持numpy、pandas的数据结构，具有绘制丰富的图表、定制图表元素或样式的能力。matplotlib还可用于绘制一些3D图表。





## 1.3.1 matplotlib概述



matplotlib官网提供了三种API：**pyplot API**、**object-oriented API**、**pylab API**。

**pyplot API**是使用**pyplot**模块开发的接口，该接口封装了一系列与**MATLAB**命令同名的函数，使用这些函数可以像使用**MATLAB**命令一样快速地绘制图表。

**pyplot API**

**object-oriented API**是面向对象的接口，该接口封装了一系列对应图表元素的类，只有创建这些类的对象并按照隶属关系组合到一起才能完成一次完整的绘图。

**object-oriented API**

**pylab API**是使用**pylab**模块开发的接口，它最初是为了模仿**MATLAB**的工作方式而设计的，包括**pyplot**、**numpy**模块及一些其它附加功能，适用于Python交互环境中。

**pylab API**



## 1.3.1 matplotlib概述



matplotlib官方不建议使用pylab API进行开发，并在最新的版本中**弃用了 pylab API**。用户在使用时可以根据自身的实际情况进行选择，若只是需要快速地绘制图表，可以选择pyplot API进行开发；若需要**自定义图表**，可以选择object-oriented API进行开发。



## 1.3.2 安装matplotlib



安装matplotlib的方式有多种，既可以直接使用pip命令安装，也可以使用Anaconda工具进行安装。

Anaconda是一个开源的Python发行版本，包括  
conda、Python环境，以及诸如numpy、pandas、  
matplotlib、scipy等180多个科学计算包，既  
可以在同一台计算机上安装不同版本的软件包  
和依赖项，也能在不同环境之间进行切换，非  
常适合初学者使用。

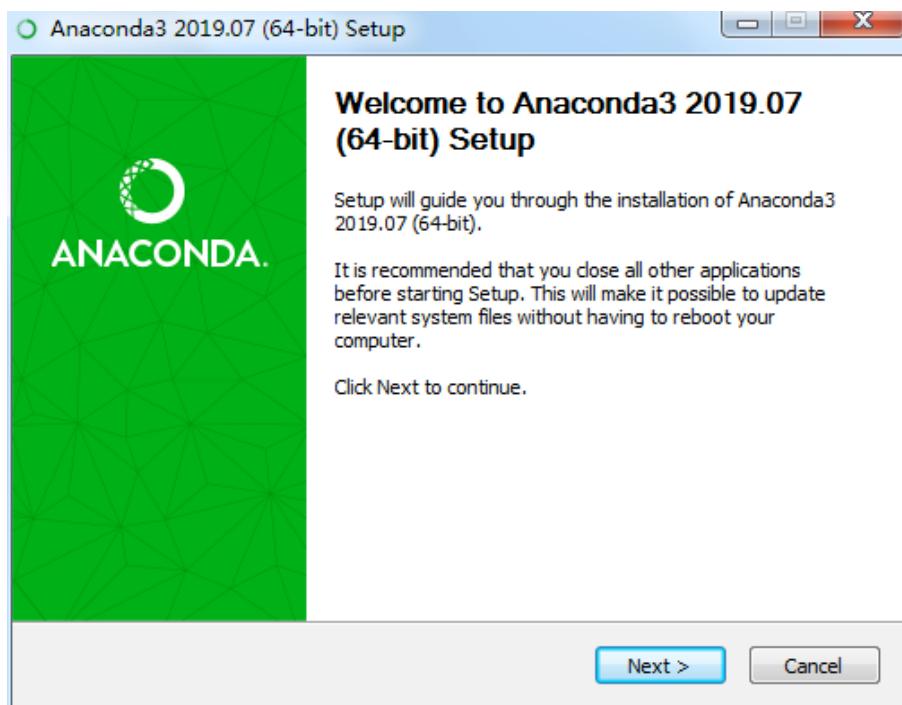




## 1.3.2 安装matplotlib

- 安装Anaconda工具

(1) 从Anaconda官网下载的安装文件，双击启动安装程序，进入**欢迎使用Anaconda3**的界面，如右图所示。



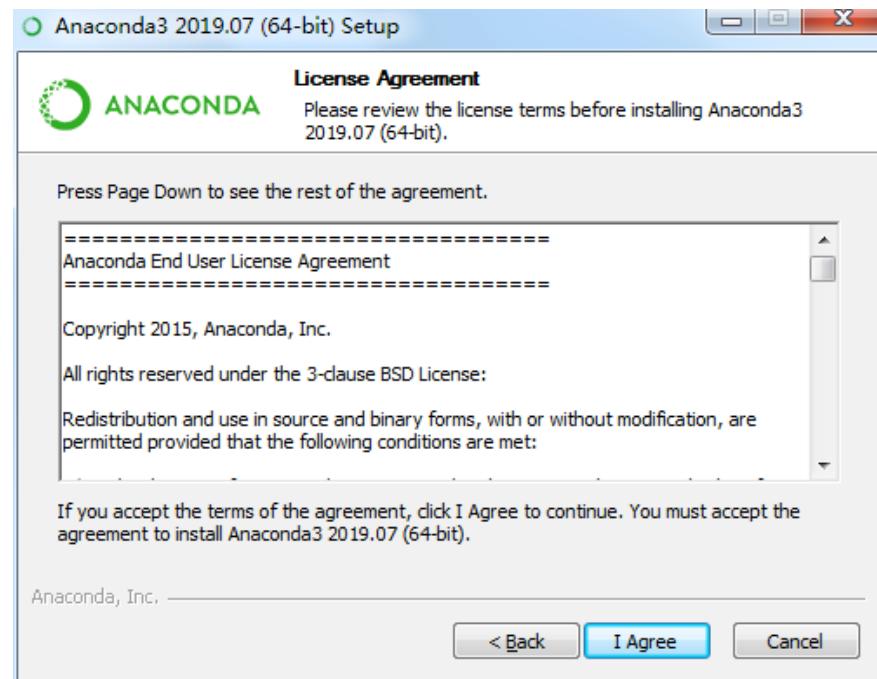


## 1.3.2 安装matplotlib



- **安装Anaconda工具**

(2) 单击【Next】按钮进入要求用户接受许可协议的界面，如右图所示。



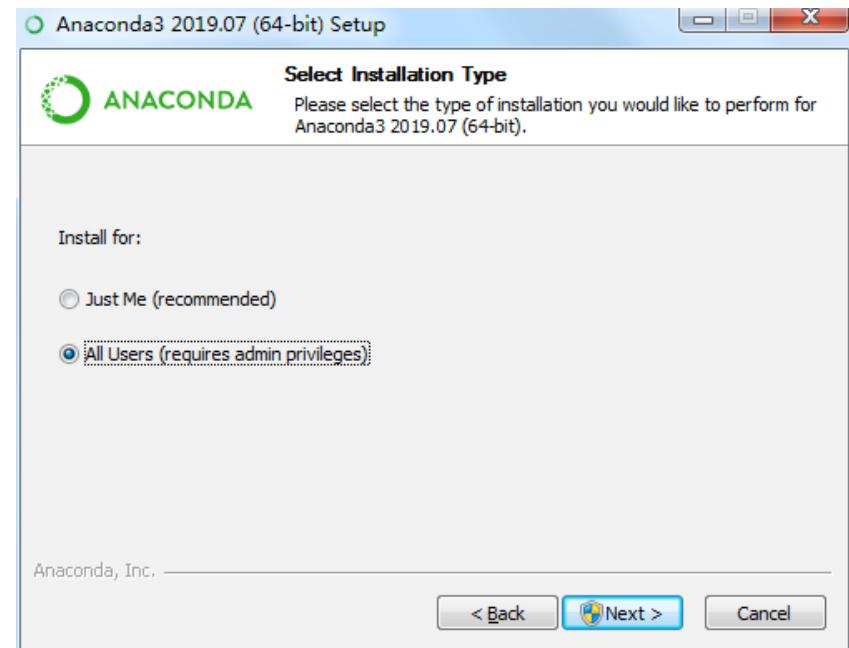


## 1.3.2 安装matplotlib



- 安装Anaconda工具

(3) 单击【I Agree】按钮进入用户选择安装类型的界面，如右图所示。这里选择采用“Just Me”方式进行安装。

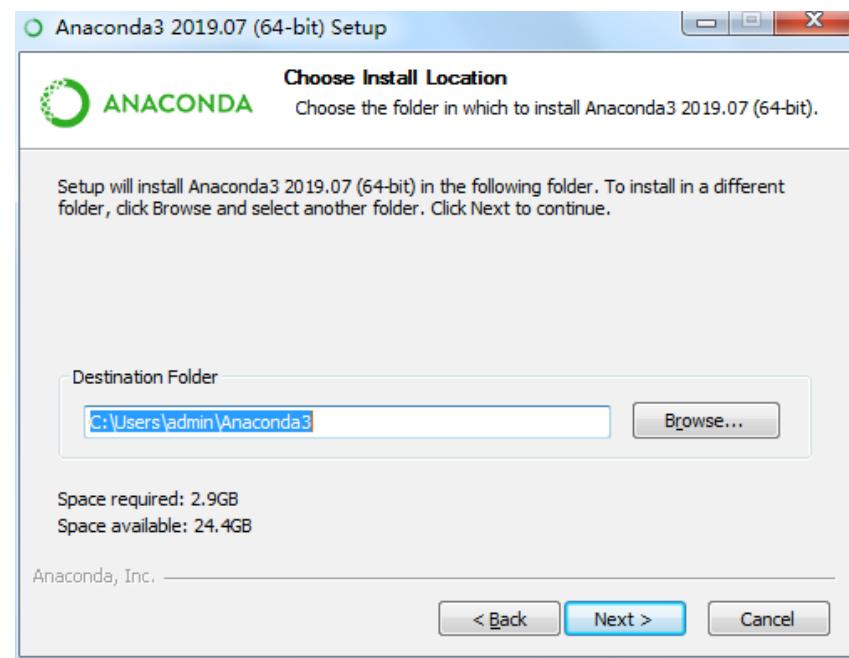




## 1.3.2 安装matplotlib

- 安装Anaconda工具

(4) 单击【Next】按钮进入用户选择Anaconda安装位置的界面，  
默认安装路径为  
“C:\Users\admin\Anaconda3” ，  
如右图所示。



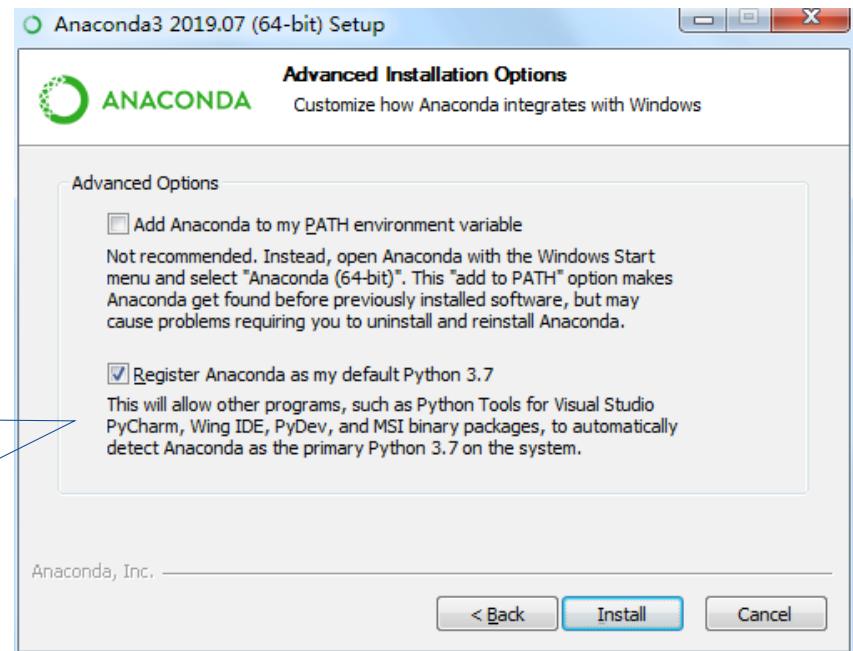


## 1.3.2 安装matplotlib

- **安装Anaconda工具**

(5) 保持默认配置。单击【Next】按钮进入设置高级安装选项的界面，如右图所示。

第1个选项为将Anaconda添加到计算机的环境变量中，第2个选项为是否允许Anaconda使用Python 3.7，这里保持默认配置。

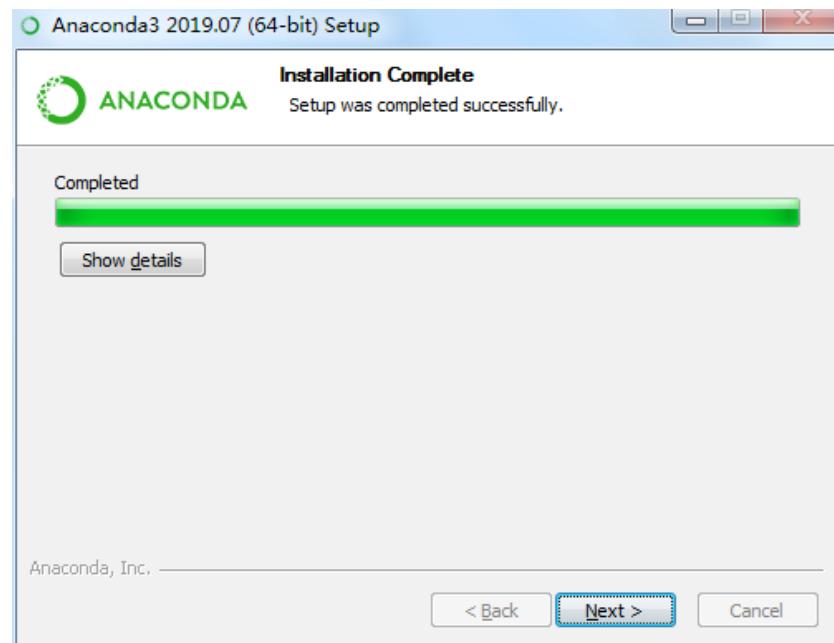




## 1.3.2 安装matplotlib

- 安装Anaconda工具

(6) 单击【Install】按钮进入开始安装的界面，待安装完成后直接进入安装完成的界面，如右图所示。

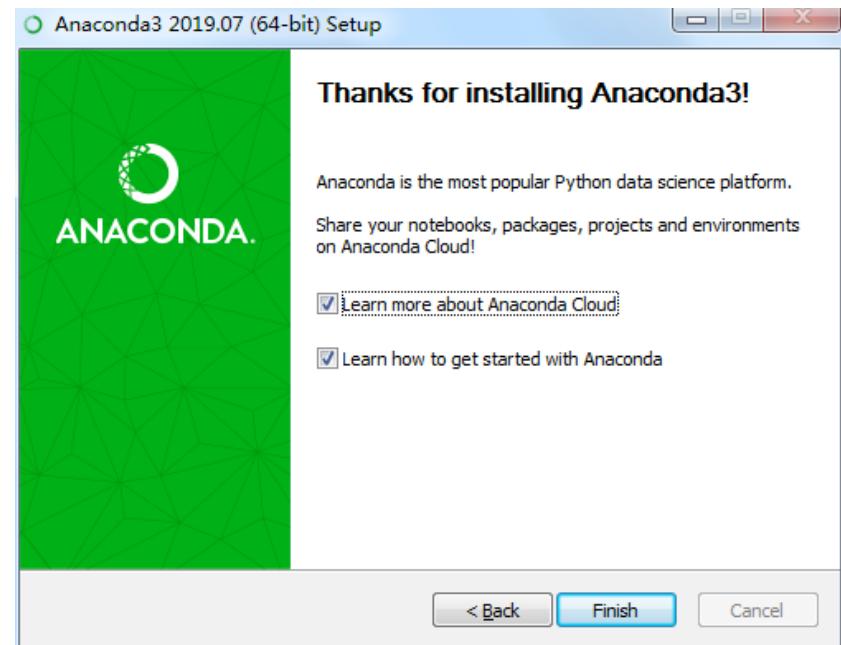




## 1.3.2 安装matplotlib

- **安装Anaconda工具**

(7) 单击【Next】按钮，进入介绍Anaconda3信息的界面，再次单击【Next】按钮进入谢谢安装Anaconda3的界面，如右图所示。





## 1.3.2 安装matplotlib



- **安装Anaconda工具**

(8) 单击【Finish】按钮，之后在默认的浏览器中打开了登录或注册“Anaconda Cloud”的界面，大家直接关闭即可。

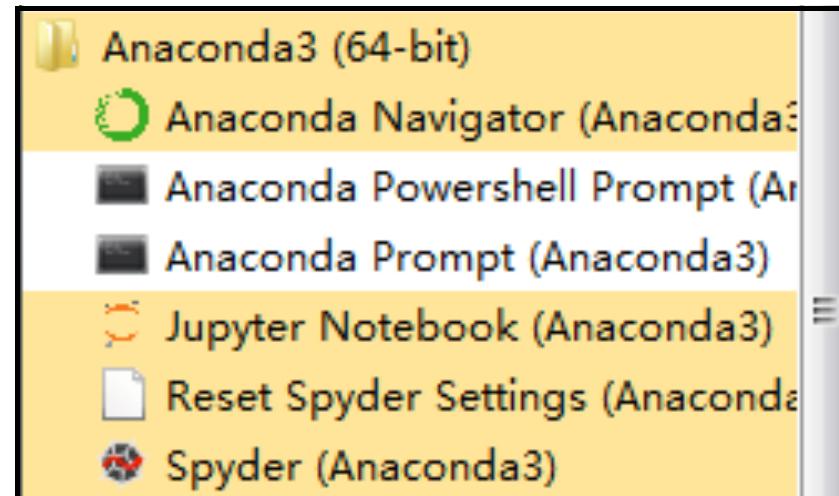




## 1.3.2 安装matplotlib

- 验证matplotlib是否安装成功

(1) 单击计算机的【开始】→【所有程序】→【Anaconda3(64-bit)】，可以看到“Anaconda3(64-bit)”目录中包含多个组件。





## 1.3.2 安装matplotlib



- 验证matplotlib是否安装成功

Anaconda发行版中包含的图形用户界面，允许用户在不使用命令的情况下启动程序并轻松管理包。

Navigator

Anaconda发行版中自带的命令行工具，允许用户使用conda命令管理包。

Prompt

基于Web网页的交互计算的应用程序，支持实时代码、数学方程和可视化。

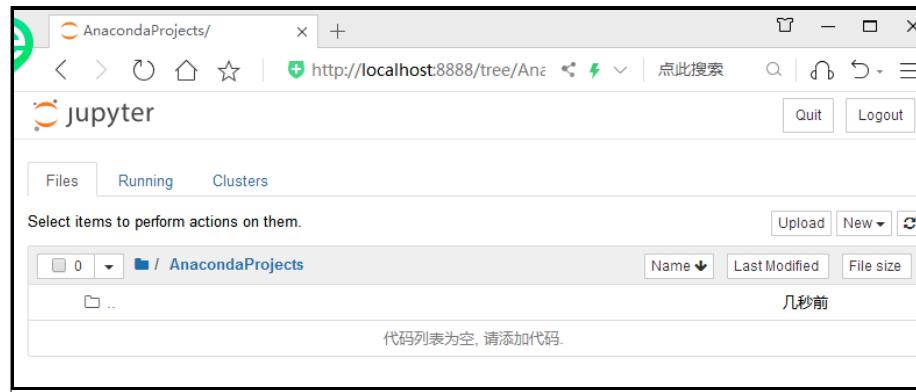
Jupyter Notebook



## 1.3.2 安装matplotlib

- 验证matplotlib是否安装成功

(2) 单击“Jupyter Notebook (Anaconda3)”启动程序，并在默认的浏览器中打开Jupyter Notebook工具，再次单击“AnacondaProjects”进入存放程序文件的目录，此时该目录中还没有任何程序文件。

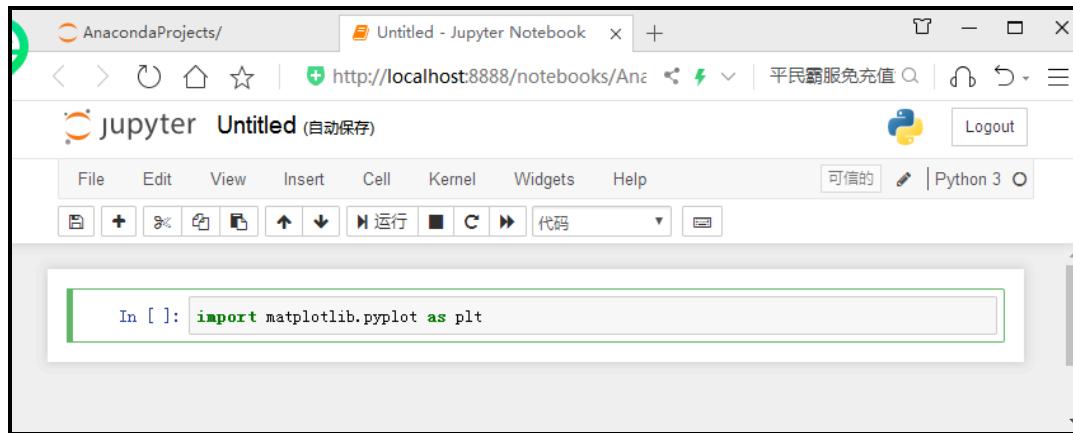




## 1.3.2 安装matplotlib

- 验证matplotlib是否安装成功

(3) 单击【New】→【Python 3】按钮，即可创建并打开一个由系统自动命名的“Untitled.ipynb”文件，在“Untitled.ipynb”文件中编写导入pyplot模块的语句。





## 1.3.2 安装matplotlib



- 验证matplotlib是否安装成功

(4) 单击【运行】按钮，程序未出现任何异常信息，表明matplotlib安装成功。

The screenshot shows a Jupyter Notebook interface. The top menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. On the right side of the menu bar, there are buttons for '可信的' (Trusted), a pencil icon, and 'Python 3'. Below the menu is a toolbar with various icons for file operations like opening, saving, and running cells. The main workspace shows two code cells. The first cell, labeled 'In [3]:', contains the Python code 'import matplotlib.pyplot as plt'. The second cell, labeled 'In []:', is currently active and has a cursor at the end of the line. A green rectangular selection box highlights the second cell.

```
In [3]: import matplotlib.pyplot as plt
```

```
In []:
```



## 1.3.2 安装matplotlib



Jupyter Notebook与其它集成开发环境相比，它可以重现整个数据分析的过程，并将代码、公式、注释、图表和结论整合到一个文档中。因此，本书后续会使用Jupyter Notebook工具进行开发，也建议初学者使用这个工具开发。



# 目录页



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌



1.1 数据可视化概述

1.2 常见的数据可视化库

1.3 初识**matplotlib**

1.4 使用**matplotlib**绘制图表





## 1.4 使用matplotlib绘制第一个图表



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌

下面结合面向对象的方式绘制一个简单的图表。

```
import numpy as np
import matplotlib.pyplot as plt
data = np.array([1, 2, 3, 4, 5])      # 准备数据
fig = plt.figure()                  # 创建画布
ax = fig.add_subplot(111)           # 添加绘图区域
ax.plot(data)                      # 绘制图表
plt.show()                         # 展示图表
```





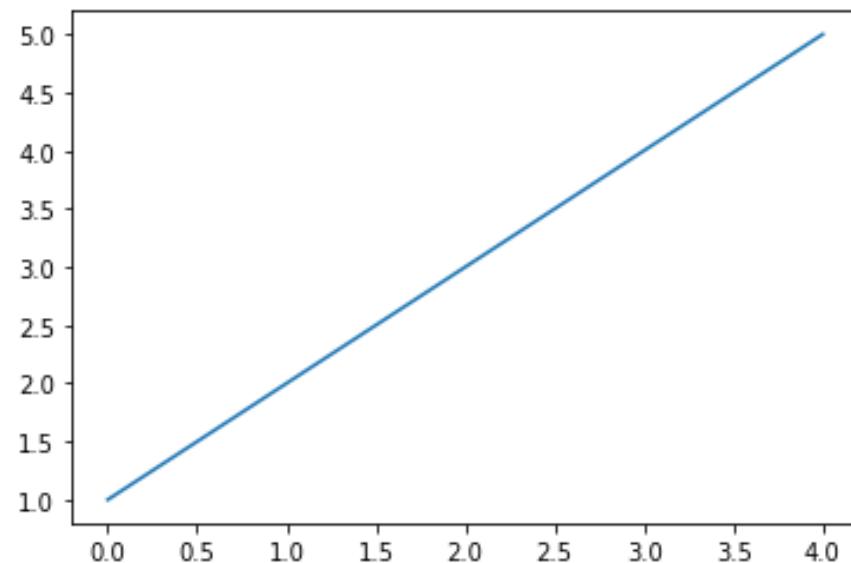
## 1.4 使用matplotlib绘制第一个图表



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌

运行效果





## 1.4 使用matplotlib绘制第一个图表



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌

下面使用pyplot的**绘图函数**快速地绘制同一个图表。

```
import numpy as np
import matplotlib.pyplot as plt
data = np.array([1, 2, 3, 4, 5]) # 准备数据
# 在当前画布的绘图区域中绘制图表
plt.plot(data)
plt.show() # 展示图表
```



## 1.4 使用matplotlib绘制第一个图表



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌



通过比较两个示例可以发现，第二个示例使用**更少的代码**便绘制了同一个图表。





## 多学一招：matplotlib所绘图形的层次结构



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌

假设大家想画一幅素描：



第1步：固定一个画板

第2步：固定一张画布

第3步：画布上作画

同理，使用 matplotlib 库绘制的图形并非只有一层结构



## 多学一招：matplotlib所绘图形的层次结构



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌

使用matplotlib绘制的图形主要由三层组成：**容器层、图像层和辅助显示层。**

容器层主要由**Canvas**对象、**Figure**对象、**Axes**对象组成。

容器层

图像层是指绘图区域  
内绘制的**图形**。

图像层

辅助显示层是指绘图  
区域内除所绘图形之  
外的**辅助元素**，包括  
坐标轴、标题、图例、  
注释文本等。

辅助显示层



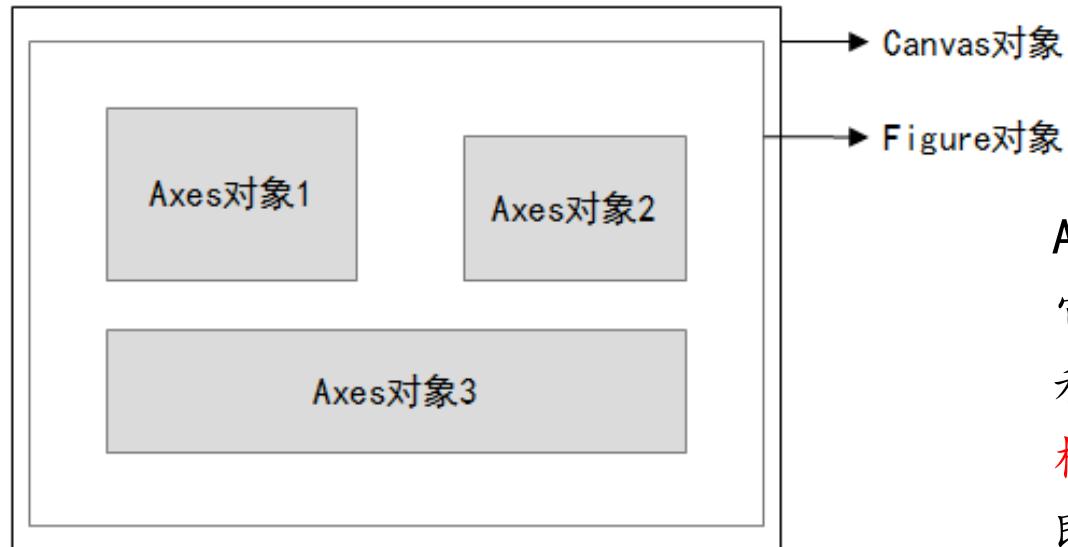
## 多学一招：matplotlib所绘图形的层次结构



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌

关于Canvas对象、Figure对象、Axes对象的**结构**如下图所示。



Axes对象拥有属于自己的坐标系，它可以是**直角坐标系**，即包含x轴和y轴的坐标系，也可以是**三维坐标系**（Axes的子类Axes3D对象），即包含x轴、y轴、z轴的坐标系。



## 多学一招：matplotlib所绘图形的层次结构



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌



图像层和辅助显示层所包含的内容都位于Axes类  
对象之上，都属于图表的元素。





# 本章小结



黑马程序员  
www.itheima.com

传智教育旗下  
高端IT教育品牌

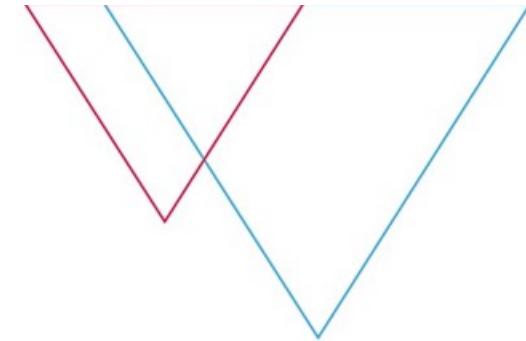


本章主要为大家介绍了数据可视化和matplotlib的入门知识，包括数据可视化概述、常见的数据可视化库、初识matplotlib库、使用matplotlib库绘制第一个图表。通过对本章的学习，希望读者可以理解可视化的过程和方式，能够独立搭建开发环境，并对matplotlib开发有一个初步的认识，为后续的学习做好铺垫。



黑马程序员  
www.iheima.com

传智教育旗下  
高端IT教育品牌



# Thank You!

**www.ityxb.com**

