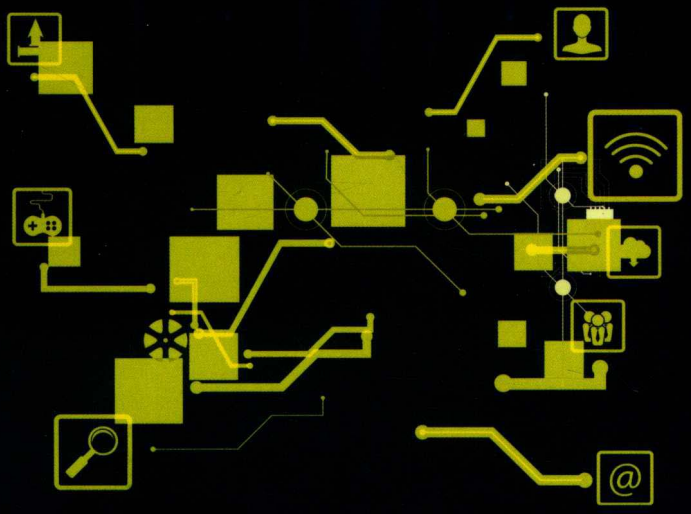
普通高等院校应用型人才培养“十三五”规划教材

Python 数据分析

微课版

Python Shuju Fenxi

吴道君 朱家荣 主编



中国铁道出版社有限公司
CHINA RAILWAY PUBLISHING HOUSE CO., LTD.

普通高等院校应用型人才培养“十三五”规划教材

Python 数据分析

吴道君　朱家荣◎主　编
毛凤翔　郭洪涛　宋　毅　孙海龙◎副主编
王庆喜◎主　审

中国铁道出版社有限公司
CHINA RAILWAY PUBLISHING HOUSE CO., LTD.

内 容 简 介

本书全面讲解 Python 数据分析的相关知识和技术，内容包括 Python 数据分析概述、NumPy 数值计算、Matplotlib 数据可视化、Pandas 数据分析、数据预处理、Sklearn 机器学习。本书以培养学生编程能力和数据分析能力为目标，注重技术应用能力的培养。

本书内容充实、结构合理、实用性强，具有明确的应用能力培养目标，易于接受和理解，学完本书后，可以具备数据分析的基本能力。

本书适合作为普通高等院校人工智能、数据科学与大数据以及计算机相关专业课程的教材，也可以作为相关从业人员的技术参考用书。

图书在版编目（CIP）数据

Python 数据分析/吴道君，朱家荣主编. —北京：中国铁道出版社有限公司，2019.9

普通高等院校应用型人才培养“十三五”规划教材

ISBN 978-7-113-25871-9

Ⅰ.①P… Ⅱ.①吴… ②朱… Ⅲ.①软件工具-程序设计-高等学校-教材 Ⅳ.①TP311.561

中国版本图书馆 CIP 数据核字（2019）第 149952 号

书　　名：Python 数据分析

作　　者：吴道君　朱家荣

策　　划：韩从付　　**编辑部电话**：010-63589185 转 2019

责任编辑：周海燕　彭立辉

封面设计：穆　丽

责任校对：张玉华

责任印制：郭向伟

出版发行：中国铁道出版社有限公司（100054，北京市西城区右安门西街 8 号）

网　　址：http://www.tdpress.com/51eds/

印　　刷：北京柏力行彩印有限公司

版　　次：2019 年 9 月第 1 版　2019 年 9 月第 1 次印刷

开　　本：787 mm×1 092 mm　1/16　**印张**：13　**字数**：322 千

书　　号：ISBN 978-7-113-25871-9

定　　价：45.00 元

前　言

PREFACE

数据的价值越来越被公众认可和推崇，而数据分析的作用就是通过一定的方法找出数据的价值。

近年来，随着大数据技术和人工智能技术的发展，Python 已经成为数据科学领域最为重要的语言和工具。Python 是一种面向对象、解释型的计算机程序设计语言，其语法简洁清晰、成熟稳定。

Python 最为重要的是具有丰富和强大的库，例如在数据分析领域的 NumPy、Matplotlib、Pandas 和 Sklean 等，这些库基本上包含了数据分析的所有方面，为数据分析提供了强大的功能支持。有了这些数据分析库，就可以非常容易地对数据进行分析，不再需要从基础做起，大大降低了数据分析的难度和复杂度。

本书主要讲解使用 Python 以及 Python 的库进行数据分析的技术，全书共分为 6 章，主要内容如下：

第 1 章 Python 数据分析概述，主要讲解数据分析的相关概念及其应用、Python 在数据分析领域的优势、Python 数据分析的第三方类库、Python 数据分析环境库的安装、Jupyter Notebook 工具的基本使用。

第 2 章 NumPy 数值计算，主要讲解 NumPy 数组的概念，NumPy 数组的创建方法、属性和数据类型，常用数组操作方法的使用，数组的切片和索引方法，数组的各类运算方法和使用，NumPy 的线性代数运算函数，数组的存取操作方法。

第 3 章 Matplotlib 数据可视化，主要讲解线形图的绘制，线形图的线的颜色、线型、坐标点、线宽设置；散点图、柱状图、条形图、饼图、直方图、箱线图的绘制；图例、坐标网格、坐标系、样式的设置，样式、RC 设置和文本注解；子图的绘制、子图坐标系的设置、图形嵌套；三维图形的绘制。

第 4 章 Pandas 数据分析，主要讲解 Pandas 的数据结构，常用的 DataFrame 数据结构；DataFrame 的基本功能，DataFrame 的行操作与列操作；Pandas 操作外部数据的方法，读取 CVS、数据库数据的方法；DataFrame 的重建索引、更换索引和层次化索引的使用；Series、DataFrame 的数据运算，函数应用与映射、排序、迭代方法；描述性统计函数，协方差、相关性等的计算方法；分组与聚合的概念、分组聚合的方法使用；透视表、交叉表的方法。

第 5 章数据预处理，主要讲解数据清洗的概念和方法，重复值、缺失值和异常值的检测

与处理；DataFrame 对象的合并连接与重塑方法；数据变换的种类、常用的数据变换方法。

第 6 章 Sklearn 机器学习，主要讲解机器学习的有关概念，Sklearn 数据集，Sklearn 数据预处理，降维、回归、聚类和分类算法，模型的选择、训练、预测和评估等。

本书配有完善的教学资源，包括教学课件、电子教案、教学大纲、教学计划、实验参考、习题答案等，可以在 http://www.tdpress.com/51eds 中下载。在教学过程中如果遇到任何问题，可以通过电子邮箱 qingxiwang1111@163.com 与作者进行交流。

本书由广东岭南职业技术学院吴道君、广西民族师范学院朱家荣任主编，信阳学院毛凤翔、洛阳师范学院郭洪涛、哈尔滨华德学院宋毅和孙海龙任副主编，其中宋毅编写了第 1 章，吴道君编写了第 2 章，朱家荣编写了第 3 章，毛凤翔编写了第 4 章，孙海龙编写了第 5 章，郭洪涛编写了第 6 章。全书由王庆喜主审。

本书得到相关领导、同事和有关学生的热情帮助和支持，在此向他们表示衷心的感谢。

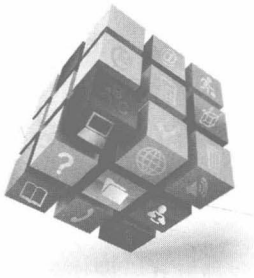
由于时间仓促，编者水平有限，书中难免存在疏漏和不足之处，敬请读者批评指正。

编　者

2019 年 5 月

目　录

CONTENTS



第1章

Python 数据分析概述

学习目标

- 熟悉数据分析的相关概念。
- 了解数据分析的应用。
- 了解 Python 在数据分析领域的优势。
- 熟悉 Python 数据分析第三方的类库。
- 掌握 Python 数据分析的类库安装。
- 掌握 Jupyter Notebook 的基本使用。

引言

随着科技的发展，各行各业产生的数据量呈现指数级增长，如何管理和使用这些数据，逐渐成为数据科学领域中的一个重要课题。近年来，Python 语言发展迅猛，为数据分析提供了极其优秀的工具，并快速成为数据科学领域的主要语言之一，越来越多的数据分析师在工作中采用 Python 技术。

1.1 数据分析的概念、流程和应用

数据分析作为数据科学与大数据技术的重要组成部分，近年来成为了数据科学领域中数据从业人员必须具备的技能，越来越被重视。

1.1.1 数据分析的概念

数据分析是指选用适当的分析方法对收集来的大量数据进行分析、提取有用信息和形成结论，对数据加以详细研究和概括总结的过程。

广义的数据分析包括狭义数据分析和数据挖掘两部分。狭义数据分析是指根据分析目的，采用对比分析、分组分析、交叉分析和回归分析等分析方法，对收集的数据进行处理与分析，提取有价值的信息，发挥数据的作用，得到一个特征统计量结果的过程。数据挖掘则是从大量的、不完全的、有噪声的、模糊的、随机的实际应用数据中，通过应用聚类模型、分类模型、回归和关联规则等技术，挖掘潜在价值的过程。

数据分析的目的是把隐藏在一大批看起来杂乱无章的数据中的信息集中、萃取和提炼出来，以找出所研究对象的内在规律，并加以利用，从而创建经济和社会价值。

1.1.2 数据分析的流程

数据分析已经逐渐演化为一种解决问题的过程，典型的数据分析流程如下：

1. 需求分析

需求分析的主要内容是根据数据分析需求方的要求和实际情况，结合现有的数据情况，提出数据分析需求的整体分析方向、分析内容，最终和需求方达成一致意见。

2. 数据获取

数据获取是根据需求分析的结果提取、收集数据。数据获取主要有两种方式：网络数据与本地数据。网络数据是指存储在互联网中的各类视频、图片、语言和文字等信息；本地数据则是指存储在本地数据库中的数据。本地数据按照数据时间又可以划分为两部分：历史数据和实时数据。历史数据是指系统在运行过程中遗存下来的数据，其数据随系统运行时间的增加而增长；实时数据是指最近一个单位周期内产生的数据。

3. 数据预处理

数据预处理是指对数据进行数据合并、数据清洗和数据变换，并直接用于分析建模的这一过程的总称。其中，数据合并可以将多张相互关联的表格合并成为一张；数据清洗可以处理重复值、缺失值和异常值；数据变换可以通过一定规则把原始数据转换为适合分析的形式，满足后期分析与建模的数据要求。

4. 分析与建模

分析与建模是指通过对比分析、分组分析、交叉分析、回归分析等分析方法，以及聚类模型、分类模型、关联模型等模型与算法，发现数据中有价值信息，并得出结论的过程。

分析与建模的方法按照目标不同可以划分几大类。如果分析目标是描述行为模式的，可采用描述性数据分析方法，同时还可以考虑关联规则、序列规则和聚类模型等。如果分析目标是量化未来一段时间内某个时间发生概率的，则可以使用分类预测模型和回归预测模型。

5. 模型评价与优化

模型评价是指对于已经建立的模型，根据其模型的类别，使用不同指标评价其性能优劣的过程。常用的聚类模型评价方法有 ARI 评价法（兰特系数）、AMI 评价（互信息）、V-measure 评分等。常用的分类模型评价方法有准确率（Accuracy）、精确率（Precision）、召回率（Recall）等。常用的回归模型评价指标有平均绝对误差、均方误差、中值绝对误差等。

模型优化则是指模型在经过模型评价后已经达到了要求，但在实际生产环境应用中，发现模型并不理想，继而对模型进行重构与优化的过程。

6. 部署

部署是指将数据分析结果与结论应用至实际生产系统的过程。

1.1.3 数据分析的应用

数据分析可以解决大量的实际问题，已经应用于各行各业，并取得了很好的效果。

1. 客户与营销分析

客户分析是根据客户的基本数据进行的商业行为分析，例如，根据客户的需求、所处行业的特征以及客户的经济情况等，使用统计分析方法和预测验证法分析目标客户，提高销售

效率；根据已有的客户特征进行客户特征分析、忠诚度分析和客户收益分析等。

营销分析囊括了产品分析、价格分析、渠道分析、广告与促销分析。产品分析主要是竞争产品分析，通过对竞争产品分析制定自身产品策略。价格分析又可以分为成本分析和售价分析。成本分析的目的是降低不必要的成本；售价分析的目的是制定符合市场的价格。渠道分析是指对产品的销售渠道进行分析，确定最优的渠道配比。广告与促销分析则能够结合客户分析，实现销量的提升、利润的增加。

2．业务流程优化

数据分析可以帮助企业优化业务流程，例如，可以通过业务系统和 GPS 定位系统获得数据，使用数据构建交通状况预测分析模型，有效预测实时路况、物流状况、车流量、客流量和货物吞吐量，进而提前补货，制定库存管理策略和路线优化；人力资源业务可以通过数据分析来优化人才招聘；交通部门可以在数据分析的基础上建立智能化交管方案降低高峰时段的路线拥堵情况。

3．完善执法

利用传感器、闭路电视安装并接入中央云数据库、车牌识别、语音识别、犯罪嫌疑人及罪犯 GPS 追踪等数据分析，实现智能警务；监控并识别异常活动、行为或事故，加快决策制定速度并防止及减少犯罪事件；通过分类模型分析方法对非法集资和洗钱的逻辑路径进行分析，找到其行为特征；通过聚类模型分析方法可以分析相似价格的运动模式，可能发现关联交易及内幕交易的可疑信息；通过关联规则分析方法可以监控多个用户的关联交易行为，为发现跨账号协同的金融欺骗行为提供依据。

4．网络安全

新型的病毒防御系统可使用数据分析技术，建立潜在攻击识别分析模型，检测大量网络活动数据和相应的访问行为，识别可能进行入侵的可疑模式，做到未雨绸缪。

5．优化机器和设备性能

通过物联网技术收集和分析设备上的数据流，包括连续用电、零部件温度、环境湿度和污染物颗粒等多种潜在特征，建立设备管理模型，从而预测设备故障，合理安排预防性的维护，以确保设备正常作业，降低因设备故障带来的安全风险。

6．改善日常生活

利用穿戴的装备生成最新的数据，根据热量的消耗以及睡眠模式来进行追踪；交友网站利用数据分析工具来帮助需要的人匹配合适的对象；基于城市实时交通信息，利用社交网络和天气数据来优化最新的交通情况。

7．医疗卫生与生命科学

利用远程医疗监控能够简化医护人员访问并分析病患医疗记录的流程，从而确保病人得到有效诊疗并降低不必要的成本；临床数据流分析能够顺利识别出异常或者预料之外的行为或者表现，从而辅助做出更准确的诊断意见；实时传感器数据分析有助于检测传染病暴发的可能性，并通过早期预警系统提示预防及准备；数据分析应用能够在几分钟内解码整个 DNA，从而制定出更科学的治疗方案，甚至对疾病进行预测，达到疾病预防的目的。

1.2 数据分析工具

随着云计算、大数据以及人工智能技术的快速发展，Python 及其开发生态环境正在受

到越来越多的关注。Python 已经成为计算机世界最重要的语言之一，更是数据分析的首选语言。

1.2.1 常用工具

主流数据分析语言有 Python、R 和 MATLAB。

Python 具有丰富和强大的类库，能够把其他语言模块很轻松地连接在一起，是一门易学、易用的程序设计语言。

R 语言主要用于统计分析、绘图等，它属于 GNU 系统的一个自由、免费、源代码开放的软件。

MATLAB 的作用是进行矩阵运算、回执函数与数据、实现算法、创建用户界面和连接其他编程语言的程序等，主要应用于工程计算、控制设计、信号处理与通信、图像处理、信号检测、金融建模设计与分析等领域。

Python、R 和 MATLAB 数据分析工具对比如表 1-1 所示。

表 1-1 Python、R 和 MATLAB 对比

项目 \ 语言	Python	R	MATLAB
难易程度	接口统一，学习曲线平缓	接口众多，学习曲线陡峭	自由度大，学习曲线较为平缓
使用场景	数据分析、机器学习、矩阵运算、科学可视化、数字图像处理、Web 应用、网络爬虫、系统运维等	统计分析、机器学习、科学数据可视化	矩阵预算、数值分析、科学数据可视化、机器学习、符号计算、数字图像处理、数字信号处理、仿真模拟等
第三方支持	拥有大量的第三方库，能够简便地调用 C、C++、Java 等其他语言	拥有大量的包，能够调用 C、C++、Java 等其他语言	拥有大量专业的工具箱，在新版本中加入了对 C、C++、Java 的支持
流行领域	工业界>学术界	工业界≈学术界	工业界≤学术界
软件成本	开源免费	开源免费	商业收费

1.2.2 Python 数据分析

Python 是一门应用十分广泛的计算机编程语言，在数据科学领域具有无可比拟的优势，逐渐成为数据科学领域的主流语言。Python 数据分析具有五方面优势：

① 语法简单精练。比起其他编程语言，Python 更容易学习和使用。

② 功能强大的库。大量优秀好用的第三方库，扩充了 Python 功能，提升了 Python 的能力，使 Python 如虎添翼。

③ 功能强大。Python 是一个混合体，丰富的工具使它介于传统的脚本语言和系统语言之间。Python 不仅具备简单易用的特点，还提供了编译语言所具有的软件工程能力。

④ 不仅适用于研究和原型构建，同时也适用于构建生产系统。研究人员和工程技术人员使用同一种编程工具，可给企业带来显著的组织效益，并降低企业的运营成本。

⑤ Python 是一门“胶水”语言。Python 程序能够以多种方式轻易地与其他语言的组件“粘接”在一起，例如 Python 的 C 语言 API 可以帮助 Python 程序灵活地调用 C 程序。因此，可以根据需要给 Python 程序添加功能，或者其他环境系统中使用 Python。

Python 数据分析除了使用 Python 基础外，还需要第三方类库。

1. NumPy

NumPy 是 Numerical Python 的简称，是 Python 语言的一个科学计算的扩展程序库，支持

大量的多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。NumPy主要提供以下内容：

① 快速高效的多维数组对象ndarray。

② 广播功能函数，广播是一种对数组执行数学运算的函数，其执行的是元素级计算。广播提供了算术运算期间处理不同形状的数组的能力。

③ 读/写硬盘上基于数组的数组集的工具。

④ 线性代数运算、傅里叶变换及随机数生成功能。

⑤ 将C、C++、Fortran代码集成到Python的工具。

除了为Python提供快速的数组处理能力外，NumPy在数据分析方面还有另外一个主要作用，即作为算法之间传递数据的容器。对于数值型数据，使用NumPy数组存储和处理数据要比使用内置的Python数据结构高效得多。此外，由其他语言（如C语言）编写的库可以直接操作NumPy数组中数据，无须进行任何数据复制工作。

2. Pandas

Pandas是Python的数据分析核心库，最初被作为金融数据分析工具而开发出来。Pandas为时间序列分析提供了很好的支持。Pandas纳入了大量库和一些标准的数据模型，提供了高效地操作大型数据集所需的工具，提供一系列能够快速、便捷地处理结构化数据的结构和函数。Python之所以成为强大而高效的数据分析环境与它息息相关。

Pandas兼具NumPy高性能的数组计算功能以及电子表格和关系型数据库（如SQL）的灵活数据处理功能，它提供了复杂精细的索引功能，以便便捷地完成重塑、切片和切换、聚合及选取数据子集等操作。

3. Matplotlib

Matplotlib是最流行的用于绘制数据图形的Python库，它以各种硬拷贝格式和跨平台的交互式环境生成出高质量的图形。Matplotlib最初由John D.Hunter创建，目前由一个庞大的开发团队维护。Matplotlib的操作比较容易，只需要几行代码即可生成线形图、散点图、直方图、条形图和箱线图等，甚至可以绘制三维图形。

4. Sklearn

Sklearn（Scikit-Learn）是一个简单高效的数据挖掘和数据分析工具，可以供用户在各种环境下重复使用。而且Sklearn建立在NumPy、SciPy和Matplotlib基础之上，对一些常用的算法进行了封装。目前，Sklearn的基本模块主要有数据预处理、模型选择、分类、聚类、数据降维和回归6个。在数据量不大的情况下，Sklearn可以解决大部分问题。对算法不精通的用户在执行建模任务时，并不需要自行编写所有算法，只需要简单地调用Sklearn库中的模块即可。

5. 其他

xlrd和openpyxl是读取Excel文件需要的类库；Seaborn与Matplotlib类似，主要作用是绘制图形，但是Seaborn自带了一些数据集，可以用来练习。

1.3 Python数据分析环境

Python数据分析环境的搭建包括Python安装以及多个第三方库的安装。

先安装Python，再分别安装需要的第三方库。读者如果想省事，也可以采用安装Anaconda

的方式简化安装。Anaconda 包含了本书使用的所有第三方库，有兴趣的读者也可以自行安装 Anaconda。因为本书使用的开发环境并不复杂，因此没有使用 Anaconda。

注意：安装过程需要网络，因为需要先下载再安装。

1. 安装 Python

Python 数据分析环境搭建

本书读者应该具备 Python 基础，因此不再赘述 Python 的安装。

注意：在安装 Python 时，一定要同时安装 PIP，否则下边的安装都无法进行。

2. 安装数据分析库

（1）安装第三方数据分析库

第三方库的安装使用 pip3 命令，如下所示：

```
pip3 install numpy
pip3 install scipy
pip3 install matplotlib
pip3 install sklearn
pip3 install xlrd
pip3 install openpyxl
pip3 install seaborn
```

（2）检查安装

安装后，可以在 Python 环境中使用导入检查是否安装成功。

```
import numpy as np
import matplotlib as plt
import pandas as pd
import sklearn.datasets import ds
```

如果需要的类库没有安装，则会提示模块不存在，如果没有错误提示，则说明安装成功。

3. Jupyter Notebook 的使用

Jupyter Notebook 的使用

Jupyter Notebook 是 IPython Notebook 的继承者，是一个交互式笔记本，支持运行 40 多种编程语言。它本质上是一个支持实施代码、数学方程、可视化和 Markdown 的 Web 应用程序。对于数据分析，Jupyter Notebook 最大的优点是可以重现整个分析过程，并将说明文字、代码、图表、公式和结论都整合在一个文档中。用户可以通过电子邮件、Dropbox、GitHub 和 Jupyter Notebook Viewer 将分析结果分享给其他人。

Jupyter Notebook 是一个非常强大的工具，常用于交互式地开发和展示数据科学项目。它将代码和它的输出集成到一个文档中，并且结合了可视的叙述性文本、数学方程和其他丰富的媒体。它直观的工作流促进了迭代和快速开发，使得 Jypyter Notebook 在当代数据科学分析和越来越多的科学研究中越来越受欢迎。最重要的是，作为开源项目，它是完全免费的。

（1）安装 Jupyter Notebook

使用如下命令安装 Jupyter Notebook。

```
pip3 install jupyter
```

（2）启动 Jupyter Notebook

注意：Jupyter Notebook 在启动后只允许访问启动目录中包含的文件（包括子目录中包含的文件），并且在 Jupyter Notebook 中创建的文件也保存在启动目录中，在启动 Jupyter Notebook 之前需要修改当前目录。

启动 Jupyter Notebook 之前先做准备工作。

① 创建目录（文件夹）。例如，在 D 盘下创建 notebook 文件夹。

② 改变系统的当前目录，把当前目录更改为创建的目录（文件夹）。

准备工作完成后，开始启动 Jupyter Notebook。在 Windows 系统下的命令行或者在 Linux 系统下的终端输入命令 Jupyter notebook 后按【Enter】键即可启动 Jupyter Notebook。启动后会自动打开系统默认的浏览器，自动展示 Jupyter Notebook 的界面。

推荐使用 Chrome 浏览器，读者可以在启动 Jupyter Notebook 之前，设置操作系统的默认浏览器。

启动后浏览器地址栏显示 http://localhost:8888/tree。其中，localhost 不是一个网站，而是表示本地机器中服务的内容。Jupyter Notebook 是 Web 应用程序，它启动了一个本地的 Python 服务器，将这些应用程序提供给 Web 浏览器，使其从根本上独立于平台，并具有 Web 上共享的优势。

（3）新建一个 Notebook

打开 Jupyter Notebook 以后会在系统默认的浏览器中出现 Jupyter Notebook 的界面（Home）。单击右上方的 New 下拉按钮，出现下拉列表，选择 Python 3 选项，进入 Python 脚本编辑界面。

下拉列表中是创建的 Notebook 类型，其中，Text File 为纯文本型，Folder 为文件夹，Python 3 表示 Python 运行脚本，灰色字体表示不可用项目。

（4）Jupyter Notebook 界面

Jupyter NoteBook 文档由一系列单元（Cell）构成，单元有两种形式。

① 代码单元。代码单元是编写代码的地方，其左边有“In[]:”符号，编写代码后，单击界面上方工具栏中的“运行”按钮，执行程序，其结果会在对应代码单元的下方显示。

② Markdown 单元。Markdown 单元对文本进行编辑，采用 Markdown 语法规范，可以设置文本格式，插入链接、图片甚至数学公式。Markdown 也可以运行，运行后显示格式化的文本（原文本被替代）。

（5）Jupyter Notebook 的两种模式

① 编辑模式。用于编辑文本和代码，对于 Markdown 单元，选中单元并按【Enter】键（或者双击）进入编辑模式；对于代码单元，选中单元后直接进入编辑模式。编辑模式的单元左侧显示绿色竖线。

② 命令模式。用于执行键盘输入的快捷命令，在编辑模式下通过按【Esc】键进入命令模式。命令模式的单元左侧显示蓝色竖线。

（6）检查点

当创建一个新的 Notebook 时，Jupyter Notebook 都会创建一个检查点文件和一个 Notebook 文件；它将位于保存位置的隐藏子目录中，称作.ipynb_checkpoints，也是一个 .ipynb 文件。默认情况下，Jupyter 将每隔 120 s 自动保存 Notebook，而不会改变主 Notebook 文件。当“保存和检查点”时，Notebook 和检查点文件都将被更新。因此，检查点能够在发生意外事件时恢复未保存的工作，通过菜单 File→Revert to Checkpoint 恢复到检查点。

（7）Markdown

Markdown 是一种轻量级的、易于学习的、可以使用普通文本编辑器编写的标记语言，通过简单的标记语法，它可以使普通文本内容具有一定的格式。Jupyter Notebook 的 Markdown

单元作为基础的 Markdown 的功能更加强大，下面将从标题、列表、字体、表格和数学公式编辑五方面进行介绍。

① 标题。标题是标明文章和作品等内容的简短语句，在行前加一个“#”字符代表一级标题，加两个“#”字符代表二级标题，依此类推。

② 列表。列表是一种由数据项构成的有限序列，即按照一定的线性顺序排列而成的数据项的集合。列表一般分为两种：一种是无序列表，使用一些图标标记，没有序号，没有排列顺序；另一种是有序列表，使用数字标记，有排列顺序。Markdown 对于无序列表，可使用星号、加号或者减号作为列表标记；Markdown 对于有序列表，则使用数字“.”“（一个空格）表示。

③ 字体。文档中为了突显部分内容，一般对文字使用加粗或斜体格式，使得该部分内容变得更加醒目。对于 Markdown 排版工具而言，通常使用星号“*”和下画线“_”作为标记字词的符号。前面有两个星号或下画线表示加粗，前后有 3 个星号或下画线表示斜体。

④ 表格。使用 Markdown 同样也可以绘制表格。代码的第一行表示表头；第二行分隔表头和主体部分；从第三行开始，每一行代表一个表格行。列与列之间用符号“|”隔开，表格的一行两边也要有符号“|”。

⑤ 数学公式编辑。在 Jupyter Notebook 的 Markdown 的单元中也可以使用 LaTeX 来插入数学公式。在文本行中插入数学公式，应使用两个“$”符号。如果要插入一个数学区块，则使用两个“$$”。

（8）导出功能

Notebook 可以导出多种格式，例如 HTML、Markdown、reST、PDF 等格式。导出功能可通过选择 File→Downloads as 级联菜单中的命令实现。

（9）快捷键

为了提高编程效率，Jupyter Notebook 提供了很多快捷键，命令模式快捷键如表 1-2 所示，编辑模式快捷键如表 1-3 所示。

表 1-2　命令模式快捷键

快　捷　键	作　　用
Enter	转入编辑模式
Shift+Enter	运行本单元，选中下个单元
Ctrl+Enter	运行本单元
Alt+Enter	运行本单元，在其下插入新单元
Y	单元转入代码状态
M	单元转入 markdown 状态
R	单元转入 raw 状态
1	设置 1 级标题
2	设置 2 级标题
3	设置 3 级标题
Up	选中上方单元
Down	选中下方单元
A	在上方插入新单元
B	在下方插入新单元
Shift+M	合并选中的单元

续表

快 捷 键	作 用
Ctrl+S 或 S	保存当前 NoteBook
H	显示快捷键帮助
Shift+Space	向上滚动
Space	向下滚动

表 1-3 编辑模式快捷键

快 捷 键	作 用
Tab	代码补全或缩进
Shift+Tab	提示
Ctrl+]	缩进
Ctrl+[	解除缩进
Ctrl+A	全选
Ctrl+Z	撤销
Ctrl+Shift+Z	重做
Ctrl+Y	重做
Ctrl+Home	跳到单元开头
Ctrl+Up	跳到单元开头
Ctrl+End	跳到单元末尾
Ctrl+Down	跳到单元末尾
Ctrl+Left	跳到左边一个字首
Ctrl+Right	跳到右边一个字首
Esc	切换到命令模式
Shift+Enter	运行本单元，选中下一单元
Ctrl+Enter	运行本单元
Alt+Enter	运行本单元，在下面插入一单元
Ctrl+S	保存当前 Notebook
Shift	忽略
Up	光标上移或转入上一单元
Down	光标下移或转入下一单元
Ctrl+/	注释整行/撤销注释

小 结

本章首先介绍了数据分析的概念、流程以及应用，然后列举说明了数据分析的常用工具，并重点介绍了 Python 数据分析的第三方类库；最后介绍 Python 数据分析环境搭建，主要是第三方库的安装，特别是 Jupyter Notebook 开发工具的使用。

习 题

一、选择题

1. 数据分析第三方库包括（ ）。