

经济金融数据分析及其环境

数据分析是指用适当的统计与计量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,从而对数据加以详细研究和概括总结的过程。这一过程也是质量管理体系的支持过程。在实用中,数据分析可帮助人们作出判断,以便采取适当行动。

大数据分析是指对规模巨大的数据进行分析。大数据可以概括为 5 个 V,数据量大(volume)、速度快(velocity)、类型多(variety)、价值(value)、真实性(veracity)。大数据作为时下最火热的 IT 行业的词汇,随之而来的数据仓库、数据安全、数据分析、数据挖掘等围绕大数据的商业价值的利用逐渐成为行业人士争相追捧的利润焦点。随着大数据时代的来临,大数据分析也应运而生。

本章简要介绍经济金融数据的类型、来源,简要评述主要的数据分析软件包,掌握目前流行的经济金融数据分析 Python 语言及其环境。

1.1 经济金融数据类型

经济金融中需要处理的数据类型主要有三类:时间序列数据、横截面数据和面板数据。

1. 横截面数据

它是同一时间(时期或时点)某一指标在不同空间的观测数据。如某一时点中国 A 股市场的平均收益率、2022 年所有 A 股上市公司的净资产收益率。在利用横截面数据作分析时,由于单个或多个解释变量观测值起伏变化会对被解释产生不同的影响,因而导致异方差问题。因此在数据整理时必须消除异方差。

2. 时间序列数据

即按时间序列排列的数据,也称为动态序列数据。时间序列数据是按照一定时间间隔对某一变量或不同时间的取值进行观测所得到的一组数据,例如每一季度的 GDP 数据、每一天的股票交易数据或债券收益率数据等。在经济金融数据分析中,时间序列数据是常见的一类数据类型。

3. 面板数据

即时间序列数据和横截面数据相结合的数据。

金融领域以时间序列数据分析(如金融市场)与面板数据分析(如公司金融)为主。

1.2 经济金融数据来源

1. 专业性网站

如国家统计局网站、中国人民银行网站、中国证监会网站、世界银行网站、国际货币基金组织网站等。

2. 专业数据公司和信息公司

国外数据库主要有芝加哥大学商学院的证券价格研究中心(CRSP)、路透(Reuters)终端、彭博(Bloomberg)终端、雅虎财经等。国内提供经济金融数据库主要有：万德(Wind)数据库、CCER 中国经济金融数据库、国泰安(GTA)数据库、锐思数据库、挖地兔数据库等。如表 1-1 所示。

表 1-1 经济金融数据库

数据来源名称	网 址
CRSP	www.chicagobooth.edu
路透	www.reuters.com
彭博	www.bloomberg.com
雅虎财经	www.finance.yahoo.com
万得(Wind)经济金融数据库	www.wind.com.cn
国泰安 GTA 经济金融数据库	www.gtadata.com
CCER 中国经济金融数据库	www.ccer.edu.cn
锐思经济金融数据库	www.resset.cn
天相经济金融数据库	www.txsec.com/zqsc/tx_data.asp
挖地兔财经数据库	www.waditu.cn
证券之宝财经数据库	www.baostock.com
Akshare 财经数据库	www.akshare.xyz

3. 抽样调查

抽样调查是针对某些专门的研究开展的一类获取数据的方式。比如,要对中国的投资者信心进行建模,就必须通过设计调查问卷,对不同的投资群体进行数据采集。

1.3 经济金融数据分析工具简介

1.3.1 Python 数据分析工具简介

Python(英语发音: /ˈpaɪθən/), 是一种面向对象、解释型计算机程序设计语言, 由 Guido van Rossum 于 1989 年底发明, 第一个公开发行人版发行于 1991 年, Python 源代码同样遵循 GPL(GNU General Public License)协议。Python 语法简洁而清晰, 具有丰富和强

大的类库。它常被昵称为胶水语言,能够把用其他语言制作的各种模块(尤其是 C/C++)很轻松地联结在一起。常见的一种应用情形是,使用 Python 快速生成程序的原型(有时甚至是程序的最终界面),然后对其中有特别要求的部分,用更合适的语言改写,比如 3D 游戏中的图形渲染模块,性能要求特别高,就可以用 C/C++ 重写,而后封装为 Python 可以调用的扩展类库。需要注意的是在您使用扩展类库时可能需要考虑平台问题,某些可能不提供跨平台的实现。

Python 需要安装 Pandas、NumPy、SciPy、Statsmodels、Matplotlib 等一系列的程序包,还需要安装 iPython 交互环境,目前有包括这些程序包的套装软件可供下载。

详细内容请登录: <https://www.python.org/> 查询。

1.3.2 R 数据分析工具简介

R 是统计领域广泛使用的诞生于 1980 年左右的 S 语言的一个分支。可以认为 R 是 S 语言的一种实现。而 S 语言是由 AT&T 贝尔实验室开发的一种用来进行数据探索、统计分析和作图的解释型语言。最初 S 语言的实现版本主要是 S-PLUS。S-PLUS 是一个商业软件,它基于 S 语言,并由 MathSoft 公司的统计科学部进一步完善。后来奥克兰(Auckland)大学的 Robert Gentleman 和 Ross Ihaka 及其他志愿人员开发了一个 R 系统。由“R 开发核心团队”负责开发。R 是基于 S 语言的一个 GNU 项目,所以也可以当作 S 语言的一种实现,通常用 S 语言编写的代码都可以不加修改地在 R 环境下运行。R 的语法来自 Scheme。R 的使用与 S-PLUS 有很多类似之处,这两种语言有一定的兼容性。S-PLUS 的使用手册,只要稍加修改就可作为 R 的使用手册。所以有人说:R,是 S-PLUS 的一个“克隆”。

详细内容请登录: <http://cran.r-project.org> 查询。

1.3.3 Stata 数据分析工具简介

Stata 由美国计算机资源中心(Computer Resource Center)1985 年研制。其特点是采用命令行/程序操作方式,程序短小精悍,功能强大。Stata 是一套提供其使用者数据分析、数据管理以及绘制专业图表的完整及整合性统计软件。它提供许许多多功能,包含线性混合模型、均衡重复反复及多项式普罗比模式。新版本的 Stata 采用最具亲和力的窗口接口,使用者自行建立程序时,软件能提供具有直接命令式的语法。Stata 提供完整的使用手册,包含统计样本建立、解释、模型与语法、文献等出版品。

除此之外,Stata 工具可以通过网络实时更新每天的最新功能。使用者还可以得知世界各地的使用者对于 Stata 公司提出的问题与解决之道,也可以通过 Stata Journal 获得许许多多的相关信息以及书籍介绍等。另外一个获取庞大资源的渠道就是 Statalist,它是一个独立的 listserver,每月交替提供使用者 1000 多个信息以及 50 多个程序。

详细内容请登录: <http://www.stata.com> 查询。

1.3.4 Matlab 数据分析工具简介

Matlab 是由美国 Mathworks 公司推出的用于数值计算和图形处理的科学计算系统。在 Matlab 工具环境下,用户可以集成地进行程序设计、数值计算、图形绘制、输入输出、文件

管理等各项操作。它提供的是一个人机交互的数学系统环境,与利用 C 语言做数值计算的程序设计相比,利用 Matlab 可以节省大量的编程时间,且程序设计自由度大。最大的特点是给用户带来的是最直观、最简洁的程序开发环境,语言简洁紧凑,使用方便灵活,库函数与运算符极其丰富,另外具有强大的图形功能。

在国际学术界,Matlab 已经被确认为准确、可靠的科学计算标准软件,许多国际一流学术刊物上,都可以看到 Matlab 的应用。

详细内容请登录: <http://www.mathworks.com> 查询。

1.3.5 EViews 数据分析工具简介

EViews 是美国 GMS 公司 1981 年发行第 1 版的 Micro TSP 的 Windows 版本,通常称为计量经济学软件包。EViews 是 Econometrics Views 的缩写,它的本意是对社会经济关系与经济活动的数量规律,采用计量经济学方法与技术进行“观察”。计量经济学的核心是设计模型、收集资料、估计模型、检验模型、运用模型进行预测、求解模型和运用模型。EViews 是完成上述任务必不可少的得力工具。正是由于 EViews 等计量经济学软件包的出现,使计量经济学取得了长足的进步,发展成为实用与严谨的经济学科。使用 EViews 软件包可以对时间序列和非时间序列的数据进行分析,建立序列(变量)间的统计关系式,并用该关系式进行预测、模拟等。虽然 EViews 是由经济学家开发的,并且大多数被用于经济学领域,但并不意味着必须限制该软件包只用于处理经济方面的时间序列。EViews 处理非时间序列数据照样得心应手。实际上,相当大型的非时间序列(截面数据)的项目也能在 EViews 中进行处理。

详细内容请登录: <http://www.eviews.com> 查询。

1.3.6 SAS 数据分析工具简介

SAS 是美国 SAS 研究所研制的一套大型集成应用软件系统,具有完备的数据存取、数据管理、数据分析和数据展现功能。尤其是创业产品统计分析系统部分,由于其具有强大的数据分析能力,一直为业界著名软件,在数据处理和统计分析领域,被誉为国际上的标准软件和最权威的优秀统计软件包,广泛应用于政府行政管理、科研、教育、生产和金融等不同领域,发挥着重要的作用。SAS 系统中提供的主要分析功能包括统计分析、经济计量分析、时间序列分析、决策分析、财务分析和全面质量管理工具等。

详细内容请登录: <http://www.sas.com> 查询。

1.3.7 SPSS 数据分析工具简介

SPSS(statistical package for the social science)——社会科学统计软件包是世界上著名的统计分析软件之一。20 世纪 60 年代末,美国斯坦福大学的三位研究生研制开发了最早的统计分析软件 SPSS,同时成立了 SPSS 公司,并于 1975 年在芝加哥组建了 SPSS 总部。20 世纪 80 年代以前,SPSS 统计软件主要应用于企事业单位。1984 年 SPSS 总部首先推出了世界第一个统计分析软件微机版本 SPSS/PC+,开创了 SPSS 微机系列产品的开发方向,从而确立了个人用户市场第一的地位。2009 年 IBM 收购 SPSS 公司后,现在在中国国内市场上推出的最新产品,是 IBM SPSS Statistics 21.0 多国语言版。SPSS/PC+ 的推出,极大

地扩充了它的应用范围,使其能很快地应用于自然科学、技术科学、社会科学的各个领域,世界上许多有影响的报纸杂志纷纷就 SPSS 的自动统计绘图、数据的深入分析、使用方便、功能齐全等方面给予了高度的评价与称赞。目前 SPSS 已经在我国逐渐流行起来。它使用 Windows 的窗口方式展示各种管理和分析数据方法的功能,使用对话框展示出各种功能选择项,只要掌握一定的 Windows 操作技能,粗通统计分析原理,就可以使用该软件为特定的科研工作服务。

详细内容请登录: <http://www.spss.com> 查询。

还有一些统计和计量经济学软件,如 Statistica、S-PLUS 等,但相对来说没有上面 6 种软件流行。各软件网站列表如表 1-2 所示。

表 1-2 常见的经济金融数据分析工具网站

工具名称	网 址
Python	www.python.org
R	www.cran.r-project.org
Stata	www.stata.com
Matlab	www.mathworks.com
EViews	www.eviews.com
SAS	www.sas.com
SPSS	www.spss.com

1.4 Python 数据分析工具的下載

1. 下载安装 Python 执行文件

可以在网站 <https://www.python.org/downloads/> 下载 Python 执行文件。如图 1-1 所示,目前最新版是 Python3.10.0。

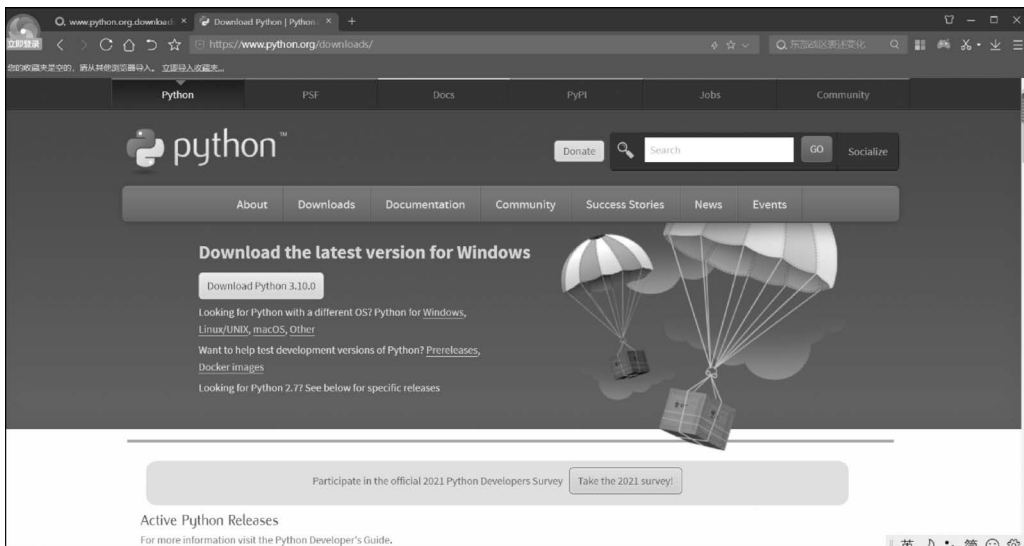


图 1-1 网站 <https://www.python.org/downloads/> 下载 Python3.10.0

在图 1-1 中点击 Download Python3.10.0,即可下载相应的 Python 执行文件,可选择你所需要的下载目录。如图 1-2 所示。

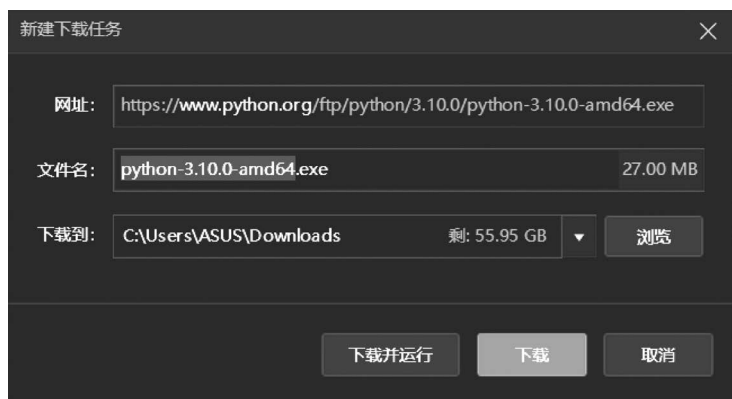


图 1-2 下载的 python-3.10.0-amd64.exe 执行文件

双击已下载目录中的 python-3.10.0-amd64.exe 执行文件,按照相应提示操作即可安装 Python。

Python 自身环境内置很多函数和模块,不过这些函数和模块功能有限,Python 的强大功能更多的是通过第三方库或者其他模块来实现。如果函数库或者模块没有内置于 Python 环境中,则需要先下载安装该函数库或模块,然后才能使用。一般通过 pip 指令来安装包,安装指令为: pip install name(如 NumPy)。

2. Python 工具 Anaconda 的下载

Python 执行文件,需要安装许多库,安装起来比较复杂,容易出现版本不兼容的问题。如果专注于科学计算功能,可直接安装 Anaconda。Anaconda 是 Python 的科学计算环境,内置 Python 安装程序,其主要功能如下:

(1) 安装简单,下载 Anaconda 的.exe 执行文件,双击执行文件,即可。

(2) 配置众多科学计算包,Anaconda 集合了 400 个以上的科学计算与数据分析功能的包,如 NumPy、Pandas、SciPy、Matplotlib 和 iPython,安装 Anaconda,这些包都被成功安装。

支持多种操作系统,兼容 Python 多种版本(2.X 和 3.X 版本可相互切换)。

在如下网站可下载 Anaconda: <https://www.anaconda.com/download/>(下载速度慢); <https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/?C=M&O=D>(下载速度快); <https://repo.anaconda.com/archive/>(下载速度快)。

Anaconda 是一个用于科学计算的 Python 发行版的套装软件,支持 Unix、Linux、Mac、Windows 等操作系统,包含了众多流行的科学计算、数据分析的 Python 包。其中包括 Pandas、NumPy、SciPy、Statsmodels、Matplotlib 等一系列的程序包以及 iPython 交互环境。界面如图 1-3 所示。

在图 1-3 中点击 Download,即可下载 Anaconda,出现图 1-4 所示的界面。

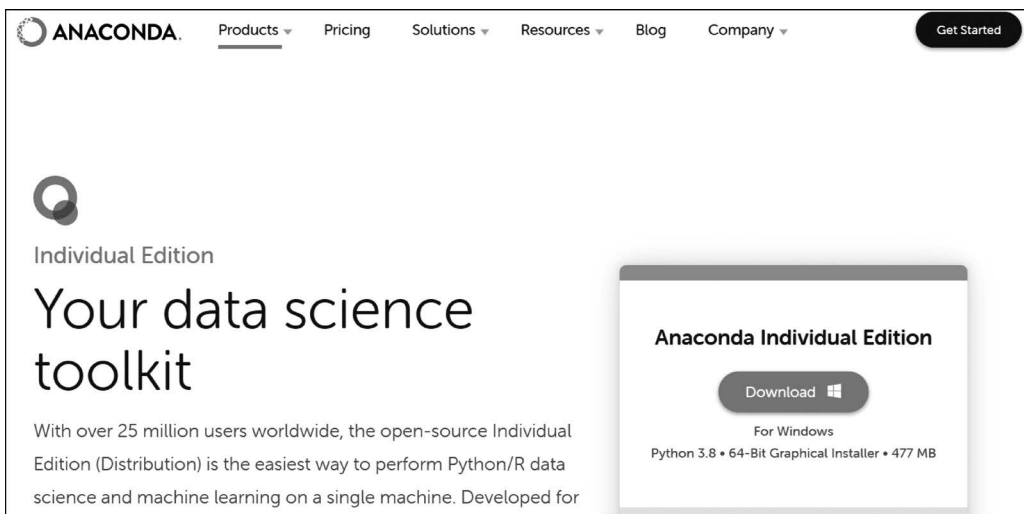


图 1-3 Anaconda 安装包界面

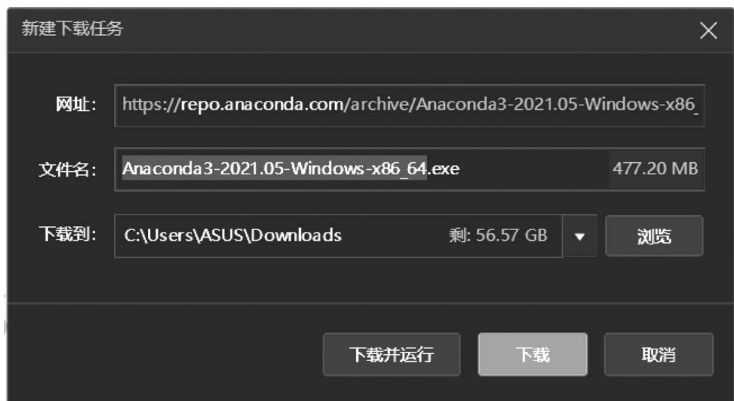


图 1-4 Anaconda 安装包下载界面

在图 1-4 所示界面中点击下载 Anaconda3-2021.05-Windows-x86_64.exe,即可得到用 Python 做量化投资的套装软件工具。

Anaconda3-2021.05-Windows-x86_64.exe 工具中提供了 Python 做量化投资的丰富资源:包括 Pandas、NumPy、SciPy、Statsmodels、Matplotlib 等一系列的程序包以及 Python 用户开发工作环境。要了解 Python 的其他程序包,可到 <https://anaconda.org> 网站上去搜索你所需要的程序包进行安装。

1.5 Python 数据分析工具的安裝

Python 在 Windows 环境中安装有很多版本。如:(1) Anaconda2-2.4.1-Windows-x86.exe(32 位)版本;(2)最新的 Anaconda3-2021.05-Windows-x86_64.exe。本书使用的 Anaconda3-2021.05-Windows-x86_64.exe 版本。

双击已下载的 Anaconda3-2021.05-Windows-x86_64.exe 应用程序,即可得到如图 1-5 所示界面。

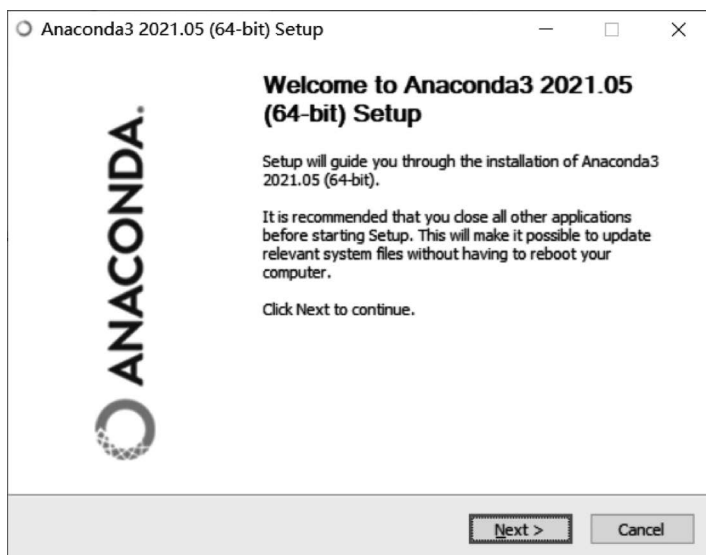


图 1-5 安装界面

在图 1-5 中点击 Next 按钮,得到如图 1-6 所示的界面。

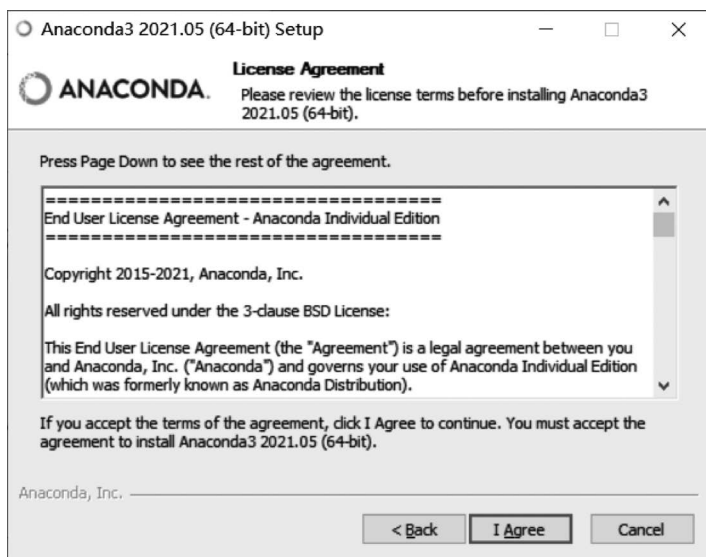


图 1-6 安装界面

在图 1-6 中点击 I Agree 按钮,得到如图 1-7 所示界面。

点击图 1-7 中 Next 按钮,得到如图 1-8 所示界面。

点击图 1-8 中 Next 按钮,即可完成 Python 套装软件的安装,得到如图 1-9 所示的界面。

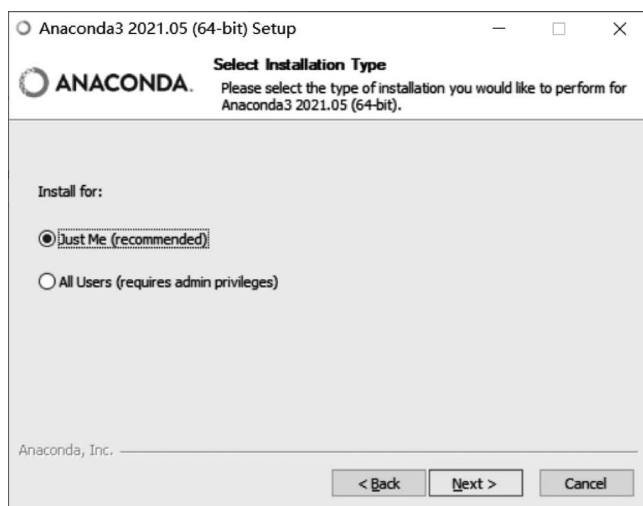


图 1-7 安装向导

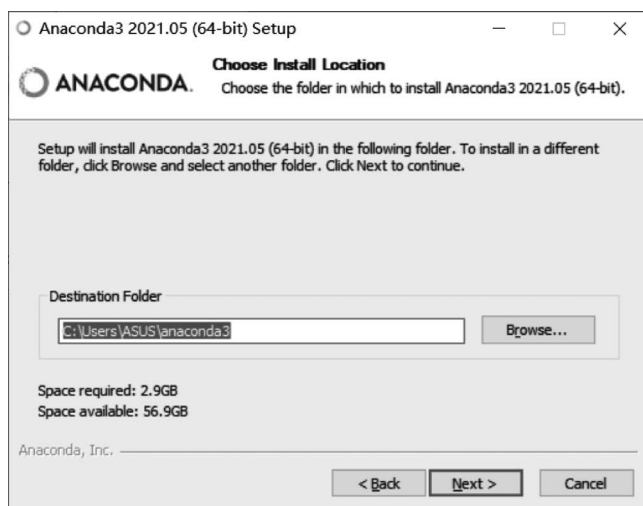


图 1-8 安装向导

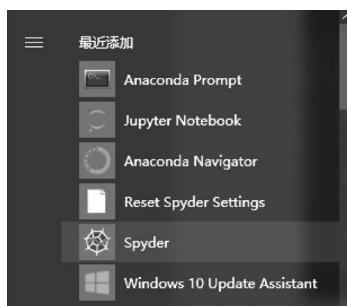


图 1-9 安装完后界面

1.6 Python 的启动和退出

1. Python 工具的启动

点击图 1-9 中 Spyder 图标,即可启动 Python 的用户界面。最后得到如图 1-10 所示的界面。

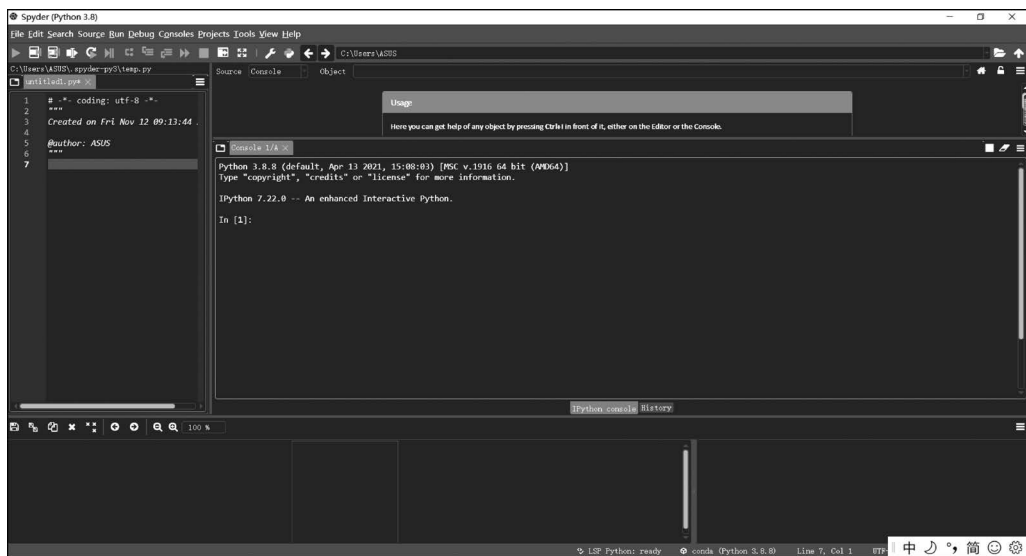


图 1-10 Python 的用户界面

2. Python 的退出

在图 1-10 中点击 Python 的用户界面中的“File”下的“Quit”菜单,即可退出 Python。

1.7 Python 数据分析相关的程序包

Python 进行数据分析时,具有获取数据、整理数据,模型计算,数据图形化等功能,相关的 Python 数据分析程序包如表 1-3 所示。

表 1-3 Python 数据分析程序包

程 序 包	简 介
NumPy	提供数组支持
SciPy	提供矩阵支持,以及矩阵相关的数值计算,优化和统计模块
Pandas	强大、灵活的数据分析和探索工具
Matplotlib	强大的数据可视化工具、作图库
StatsModels	统计建模和计量经济学,包括描述统计、统计模型估计和推断
Scikit-learn	支持回归、分类、聚类等的强大机器学习库
Keras	深度学习库,用于建立神经网络以及深度学习模型

续表

程 序 包	简 介
Gensim	用来做文本主题模型的库,文本挖掘可能用到
Pillow	涉及图片处理
OpenCV	涉及视频处理
GMPY2	涉及高精度运算

1.7.1 Statsmodels 程序包

表 1-3 的 Statsmodels 程序包是 Python 进行统计建模和计量经济学工具,提供一些互补 SciPy 统计计算的功能,包括描述性统计、统计模型估计和推断等。主要如下。

(1) 线性回归模型: 广义最小二乘法 (generalized least squares), 普通最小二乘法 (ordinary least squares)。

(2) glm: 广义线性模型。

(3) discrete: 离散变量的回归, 基于最大似然估计。

(4) rlm: 稳健线性模型。

(5) tsa: 时间序列分析模型

(6) nonparametric: 非参数估计。

(7) datasets: 数据集合。

(5) stats: 常用统计检验。

(6) iolib: 读 Stata 的 .dta 格式, 输出 ascii、latex 和 html。

Statsmodels 程序包详细内容参见 <https://github.com/statsmodels/statsmodels> 主页。

1.7.2 Scikit-learn 程序包

表 1-3 的 Scikit-Learn 程序包的功能如下。

(1) 所有模型提供的接口有: `model.fit()`: 训练模型, 对于监督模型来说是 `fit(X, y)`, 对于非监督模型是 `fit(X)`。

(2) 监督模型提供的接口有: ① `model.predict(X_new)`: 预测新样本; ② `model.predict_proba(X_new)`: 预测概率, 仅对某些监督模型有用 (比如 LR); ③ `model.score()`: 得分越高, fit 越好。

(3) 非监督模型提供的接口有: ① `model.transform()`: 从数据中学到新的“基空间”; ② `model.fit_transform()`: 从数据中学到新的基并将这个数据按照这组“基”进行转换。

1.7.3 Keras 程序包

虽然 Scikit-learn 足够强大, 但是它并没有包含一种强大的模型——人工神经网络。在语言处理、图像识别等领域有着重要的作用。

但要注意的是 Windows 环境下 Keras 的速度会大打折扣, 因此, 要研究神经网络和深度学习方面的内容, 需要在 Linux 下搭建环境。

1.8 Python 数据分析快速入门

1.8.1 数据导入

这是很关键的一步,为了后续的分析,首先需要导入数据。通常来说,数据一般是 CSV 格式,就算不是,至少也可以转换成 CSV 格式。

为了读取本地 CSV 文件,我们需要 pandas 这个数据分析库中的相应模块。其中的 read_csv 函数能够读取本地和 web 数据。

在 Python 中,操作如下:

```
import pandas as pd
# 读取本地数据
df = pd.read_csv('F:/2glkx/data/Advertising.csv')
```

df 中的变量:

TV——指把产品推向市场在电视上的广告费用(以万元为单位);

radio——在广播媒体上的广告费用(单位同上);

newspaper——用于报纸媒体的广告费用(单位同上);

sales——对应产品的销售金额(单位同上)。

1.8.2 数据变换

有了数据,接下来就是数据变换。统计学家和科学家们通常会在这一步移除分析中的非必要数据。先看看网上读取数据的前 5 行和最后 5 行。操作如下:

```
# df 数据的前 5 行
print(df.head())
      TV  radio  newspaper  sales
0  230.1   37.8     69.2   22.1
1   44.5   39.3     45.1   10.4
2   17.2   45.9     69.3    9.3
3  151.5   41.3     58.5   18.5
4  180.8   10.8     58.4   12.9
# df 数据的后 5 行
print(df.tail())
      TV  radio  newspaper  sales
195  38.2    3.7     13.8    7.6
196  94.2    4.9      8.1    9.7
197 177.0    9.3      6.4   12.8
198 283.6   42.0     66.2   25.5
199 232.1    8.6      8.7   13.4
```

对 R 语言程序员来说,上述操作等价于通过 print(head(df)) 来打印数据的前 6 行,以及通过 print(tail(df)) 来打印数据的后 6 行。当然 Python 中,默认打印是 5 行,而 R 则是 6 行。因此 R 的代码 head(df, n=10), 在 Python 中就是 df.head(n=10), 打印数据尾部也是同样道理。

在 R 语言中,数据列和行的名字通过 colnames 和 rownames 来分别进行提取。在

Python 中,则使用 columns 和 index 属性来提取,操作如下:

```
# 提取列名
print (df.columns)
Index(['TV', 'radio', 'newspaper', 'sales'], dtype='object')
# 提取行名或索引 Extracting row names or the index
print (df.index)
RangeIndex(start = 0, stop = 200, step = 1)
```

数据转置使用 T 方法,操作如下:

```
print (df.T)
      0      1      2      3      4      5      6      7      8      9  \
TV      230.1  44.5  17.2  151.5  180.8  8.7  57.5  120.2  8.6  199.8
radio    37.8  39.3  45.9  41.3  10.8  48.9  32.8  19.6  2.1  2.6
newspaper 69.2  45.1  69.3  58.5  58.4  75.0  23.5  11.6  1.0  21.2
sales    22.1  10.4  9.3  18.5  12.9  7.2  11.8  13.2  4.8  10.6
      190    191    192    193    194    195    196    197    198  \
TV      39.5  75.5  17.2  166.8  149.7  38.2  94.2  177.0  283.6
radio    41.1  10.8  4.1  42.0  35.6  3.7  4.9  9.3  42.0
newspaper 11.5  6.0  31.6  3.6  6.0  13.8  8.1  6.4  66.2
sales     10.8  9.9  5.9  19.6  17.3  7.6  9.7  12.8  25.5
      199
TV      232.1
radio      8.6
newspaper  8.7
sales     13.4
[4 rows x 200 columns]
```

其他变换,例如排序就是用 sort 属性。现在我们提取特定的某列数据。Python 中,可以使用 iloc。例如需数据第一列的前 5 行,操作如下:

```
print (df.iloc[:, 0].head())
0      230.1
1      44.5
2      17.2
3     151.5
4     180.8
Name: TV, dtype: float64
```

要注意的是,Python 的索引是从 0 开始而非 1。为了取出从 11 到 20 行的前 3 列数据,我们有:

```
print (df.iloc[10:20, 0:3])
      TV  radio  newspaper
10  66.1   5.8     24.2
11 214.7  24.0      4.0
12  23.8  35.1     65.9
13  97.5   7.6      7.2
14 204.1  32.9     46.0
15 195.4  47.7     52.9
16  67.8  36.6    114.0
17 281.4  39.6     55.8
18  69.2  20.5     18.3
19 147.3  23.9     19.1
```

为了舍弃数据中的列,如列 1(TV)和列 2(radio),可使用 drop 属性,操作如下:

```
print(df.drop(df.columns[[2, 3]], axis = 1).head())
```

	TV	radio
0	230.1	37.8
1	44.5	39.3
2	17.2	45.9
3	151.5	41.3
4	180.8	10.8

axis 参数告诉函数到底舍弃列还是行。如果 axis 等于 0,那么就舍弃行。

1.8.3 统计描述

下一步就是通过 describe 属性,对数据的统计特性进行描述:

```
print(df.describe())
```

	TV	radio	newspaper	sales
count	200.000000	200.000000	200.000000	200.000000
mean	147.042500	23.264000	30.554000	14.022500
std	85.854236	14.846809	21.778621	5.217457
min	0.700000	0.000000	0.300000	1.600000
25 %	74.375000	9.975000	12.750000	10.375000
50 %	149.750000	22.900000	25.750000	12.900000
75 %	218.825000	36.525000	45.100000	17.400000
max	296.400000	49.600000	114.000000	27.000000

1.8.4 假设检验

在 Python 中,有一个很好的统计推断包,就是 Scipy 里面的 stats。ttest_1samp 实现了单样本 t 检验。因此,如果想检验数据 TV 列的均值,通过零假设,这里我们假定总体 TV 均值为 25,我们有:

```
from scipy import stats as ss
# Perform one sample t-test using 25 as the true mean
print(ss.ttest_1samp(a = df['TV'], popmean = 25))
Ttest_1sampResult(statistic = 20.103161602049973, pvalue = 8.362542573291619e - 50)
```

返回下述值组成的元组:

t : 浮点或数组类型,为 t 统计量;

prob: 浮点或数组类型,为 two-tailed p-value 双侧概率值。

通过上面的输出,可以看到 p 值是 $8.362542573291619e-50$,远小于 $\alpha=0.05$,因此有充分的证据说 TV 平均值不是 25。将这个检验应用到所有的变量,同样假设均值为 25,我们有:

```
print(ss.ttest_1samp(a = df, popmean = 25))
Ttest_1sampResult(statistic = array([ 20.1031616 , - 1.6536043 ,  3.60653789, - 29.75497579]),
pvalue = array([ 8.36254257e - 50,  9.97845324e - 02,  3.92358536e - 04,  3.40162038e - 75]))
```

上述输出结果的第一个数组是 t 统计量,第二个数组则是相应的 p 值。

1.8.5 可视化

Python 中有许多可视化模块,最流行的是 Matplotlib 库。我们可选择功能更强的

seaborn 模块。

```
# Import the module for plotting
import matplotlib.pyplot as plt
plt.show(df.plot(kind = 'box'))
```

得到如图 1-11 所示的图形。

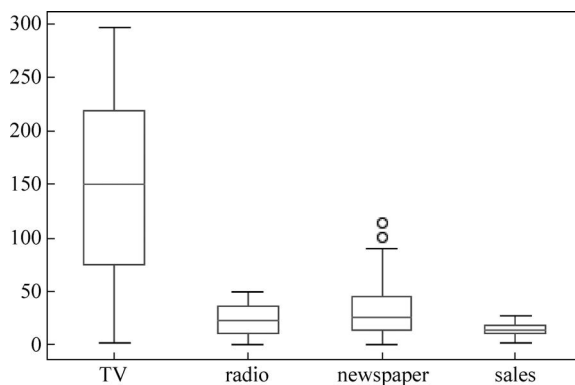


图 1-11 盒型图

有选择性的数据图输入：

```
import matplotlib.pyplot as plt
df['TV'].plot(kind = 'box')
```

这样我们就得到如图 1-12 所示的图形。

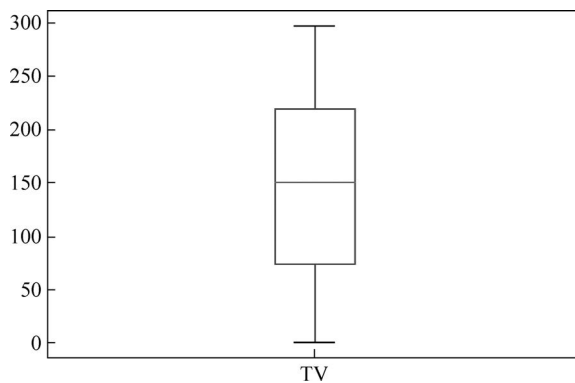


图 1-12 盒型图

可见比 matplotlib.pyplot 主题简洁太多。下面再引入功能更强大 seaborn 模块,该模块是一个统计数据可视化库。因此我们有：

```
# Import the seaborn library
import seaborn as sns
# Do the boxplot
sns.boxplot(df['TV'])
```

可得到如图 1-13 所示的图形。

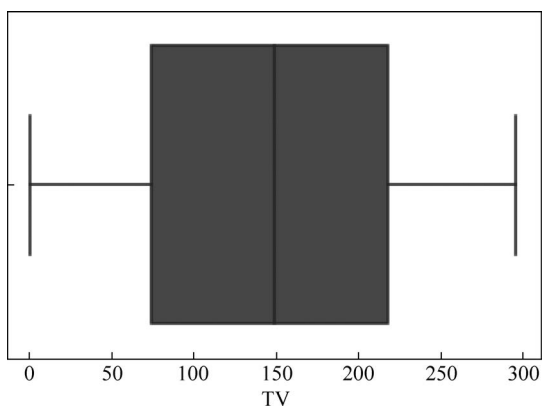


图 1-13 盒型图

1.8.6 创建自定义函数

在 Python 中,我们使用 def 函数来实现一个自定义函数。例如,如果我们要定义一个两数相加的函数,如下即可:

```
def add_2int(x, y):  
    return x + y  
print (add_2int(2, 2))  
4
```

练 习 题

1. 简述经济金融数据的类型、来源。
2. 简述经济金融数据分析的常用工具。
3. Python 与 R、Stata、Matlab、SAS、SPSS、Eviews 等数据分析工具有何区别?
4. 在网址: <https://mirrors.tuna.tsinghua.edu.cn/help/anaconda/> 下载最新 Python 工具到你指定的目录,并安装到你指定的目录,并启动 Python 软件和退出。