第 1 章
Python 语言、数据分析与可视化概述

1.1 Python 语言

1.1.1 Python 语言简介

Python 语言是一个开源的解释型、面向对象的编程语言，拥有丰富的库。由荷兰人吉多·范罗苏姆(Guido van Rossum)于 1989 年年底发明，被广泛应用于数据处理、网络爬虫、科学计算以及开发各种应用程序。

1.1.2 Python 的特点

Python 的设计秉承“优雅”“明确”“简单”的理念，具有以下特点。

1. 简单、易学

Python 是一种推崇简单主义的语言。它能使学习者更加专注于解决问题本身。同时 Python 容易上手，而且它有极其简单的说明文档。

2. 速度快

Python 的底层是基于 C 语言的，很多标准库和第三方库也都是用 C 语言编写的，运行速度非常快。

3. 免费、开源

Python 是 FLOSS(自由/开放源码软件)之一。使用者可以自由地发布这个软件的副本，阅读它的源代码，对它做改动或把它的一部分用于新的自由软件中。

4. 高层语言

用 Python 语言编写程序时无须考虑诸如如何管理程序使用的内存等一类的底层细节。

5. 可移植性

由于它的开源本质，Python 已经被移植在许多平台上(经过改动使它能够在不同平台上工作)。这些平台包括 Linux、Windows、VMS、Solaris 以及 Google 基于 linux 开发的 android 平台等。

6. 解释性

使用 Python 语言编写的程序不需要编译成二进制代码，可以直接从源代码运行程

序。在计算机内部，Python 解释器会把源代码转换成称为字节码的中间形式，然后把它翻译成计算机使用的机器语言并运行。这使得使用 Python 更加简单，也使得 Python 程序更加易于移植。

7. 面向对象

Python 既支持面向过程的编程也支持面向对象的编程。在面向过程语言中，程序是由过程或仅仅是可重用代码的函数构建起来的。而在面向对象语言中，程序是由数据和功能组合而成的对象构建起来的。

8. 可扩展性与可嵌入性

如果需要一段关键代码运行得更快或者希望某些算法不公开，可以将部分程序用 C 或 C++ 语言编写，然后在 Python 程序中使用它们。同时也可以把 Python 嵌入 C/C++ 程序，从而向程序用户提供脚本功能。

9. 丰富的库

Python 有很庞大的库，利用这些库可以帮助处理各种工作，包括正则表达式、文档生成、单元测试、线程、数据库、网页浏览器、CGI、FTP、电子邮件、XML、XML-RPC、HTML、WAV 文件、密码系统、GUI(图形用户界面)和其他与系统有关的操作。这被称作 Python 的“功能齐全”理念。

1.1.3 Python 的应用领域

随着 Python 语言的盛行，它应用的领域越来越广泛，如 Web 开发、网络爬虫、人工智能、数据分析、自动化运维、图形处理、数学运算、数据库编程、多媒体应用等。

1.2 数据分析与数据可视化概述

1.2.1 数据分析

数据分析是指采用适当的统计分析方法对收集来的大量数据进行分析，提取有用的信息和作出结论，从而对数据加以详细研究和概括总结的过程。

数据分析的应用范围很广。比较典型的数据分析主要包括以下三个步骤。

(1) 探索性数据分析。当数据刚取得时，可能杂乱无章，看不出规律，通过作图、造表、用各种形式的方程拟合，以及计算某些特征量等手段探索规律性的可能形式，即往什么方向和用何种方式去寻找和揭示隐含在数据中的规律性。

(2) 模型选定分析。在探索性分析的基础上提出一类或几类可能的模型，然后通过进一步的分析从中挑选一定的模型。

(3) 推断分析。通常使用数理统计方法对所定模型或估计的可靠程度和精确程度作出推断。

1.2.2 数据可视化

数据可视化是指将大型数据集中的数据以图形图像形式表示，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。

文本形式的数据总是显得很混乱、不直观，而可视化的数据可以帮助人们快速、轻松地提取数据中的含义。因此，用可视化方式可以充分展示数据的模式、趋势和相关性，而假如采用其他呈现方式则可能难以被发现。

数据可视化可以是静态的或交互的。几个世纪以来，人们一直在使用静态数据可视化，如图表和地图。交互式的数据可视化则相对更为先进，能够使用电脑和移动设备深入到这些图表和图形的具体细节，然后用交互的方式改变他们看到的数据及数据的处理方式。

1.2.3 数据可视化首选工具 Python

数据分析与可视化工具有很多，如 Microsoft Excel、PHP、JavaScript、SPSS、R、Matlab、Python 等。那为什么要首选 Python 进行数据分析与可视化？原因至少有以下三点。

1. 数据爬取需要 Python

Python 是目前最流行、最受青睐的数据爬取语言。Python 拥有许多支持爬取数据的扩展库，如 requests、bs4-beautifulsoup 4、Portia、Crawley、Scrapy 等。使用 Python 可以爬取 Internet 上公开的大部分数据。

2. 数据分析处理需要 Python

在获取数据之后要对数据进行清洗与预处理，之后还要对数据进行分析和可视化。Python 提供了很多对数据分析处理的扩展库，如 Numpy、Pandas、Matplotlib、Seaborn、Pyecharts 等，利用这些库可方便地进行科学计算、数据处理、图形绘制等。

3. Python 语言简洁、灵活、高效

Python 语法简单、易学易用、可移植性强，这不仅让学习者感受到语法学习的轻松，对于数据分析处理的专业人员来说也摆脱了其语言语法和跨平台的困扰，从而能够更快地对数据进行分析处理。

1.2.4 Python 数据分析与可视化的常用扩展库

1. Numpy 库

NumPy 是 Python 语言的一个扩展库，可支持多维数组与矩阵运算，此外也针对数组运算提供了大量的数学函数库。

2. Pandas 库

Pandas 是一个基于 Numpy 的 Python 库，可专门用于解决数据分析任务，提供了大量便于数据处理的函数和方法，被广泛应用于经济、统计、分析等领域。

3. Matplotlib 库

Matplotlib 是一套面向对象的绘图库，主要使用了 Matplotlib.pyplot 工具包，其绘制的图表中的每个绘制元素（如线条、文字等）都是对象。Matplotlib 库配合 NumPy 库使用，可以实现科学计算结果的可视化显示。

4. Seaborn 库

Seaborn 是基于 Matplotlib 的 Python 数据可视化库。它提供了一个高级界面，用于绘制内容丰富的统计图形，只是在 Matplotlib 上进行了更高级的 API 封装，从而使绘制

图形变得更加容易。

5. Pyecharts 库

Pyecharts 是一个用于生成 Echarts 图表的类库，Echarts 是百度开源的一个数据可视化 JavaScript 库，主要用于数据可视化。Pyecharts 主要基于 Web 浏览器进行显示，可绘制的图形比较多，包括折线图、柱状图，以及饼图、漏斗图、地图、词云图及极坐标图等。

1.3 Python 开发环境及工具

Python 是一种开源、免费的程序语言，它并没有提供一个官方的开发环境，需要用户自主来选择编辑工具。目前，Python 的开发环境有很多种，如 IDLE、Anaconda 等。

1.3.1 IDLE 开发工具

IDLE 是 Python 内置的集成开发环境（Integrated Development and Learning Environment，IDLE），它由 Python 安装包来提供，也就是 Python 自带的文本编辑器。

IDLE 为开发人员提供了许多有用的功能，如自动缩进、语法高亮显示、单词自动完成以及命令历史等，在这些功能的帮助下，用户能够有效地提高开发效率。

1.3.2 Anaconda 开发工具

Anaconda 是可以便捷获取包且对包能够进行管理，同时对环境可以统一管理的发行版本。Anaconda 包含了 conda、python 在内的超过 180 个科学包及其依赖项。

Anaconda 具有开源、安装过程简单、高性能使用 Python 和 R 语言以及免费的社区支持等特点，其特点主要基于 Anaconda 拥有 conda 包、环境管理器以及 1000 多个开源库。

Anaconda 可以在 Windows、macOS、Linux（x86/Power 8）等系统平台中安装使用，且系统要求是 32 位或 64 位，其下载文件大小约 500MB，所需存储空间大小约 3GB。

1.3.3 Jupyter 编辑平台

Jupyter Notebook 是基于网页的用于交互计算的应用程序，支持运行几十种编程语言。Jupyter Notebook 的本质是一个 Web 应用程序，便于创建和共享流程化程序文档，支持实时代码、数学方程、可视化和 markdown。Jupyter Notebook 的主要特点如下。

（1）编程时具有语法高亮、缩进、Tab 补全的功能。

（2）可直接通过浏览器运行代码，同时在代码块下方展示运行结果。

（3）以富媒体格式展示计算结果。富媒体格式包括 HTML、LaTeX、PNG、SVG 等。

（4）对代码编写说明文档或语句时，支持 Markdown 语法。

（5）支持使用 LaTeX 编写数学性说明。

1.3.4 库的安装与管理

Python 库分为标准库和扩展库（第三方库），Python 的标准库是随着 Python 安装时默认自带的库，Python 的第三方库，需要下载或在线安装到 Python 的安装目录下。

Python有两个基本的库管理工具easy_install和pip。目前大部分使用者都采用pip来进行对扩展库的查看、安装与卸载。下面介绍几个常用的pip命令。

1. 查看扩展库

```
cmd> pip list
```

例如：X:\Programs Files\Python38\Scripts>pip list。

2. 查看当前安装的库

```
cmd> pip show Package
```

例如：X:\Programs Files\Python38\Scripts>pip show jieba。

3. 安装指定版本的扩展库

```
cmd> pip install Package ==版本号
```

例如：X:\Programs Files\Python38\Scripts>pip install django==1.9.7。

4. 离线安装扩展库文件whl

```
cmd> pip install Package.whl
```

例如：X:\Programs Files\Python38\Scripts>pip install numpy-1.15.4+vanilla-cp35-cp35m-win_amd64.whl。

5. 卸载扩展库

```
cmd> pip uninstall Package
```

例如：X:\Programs Files\Python38\Scripts>pip install django。

6. 更新扩展库

```
cmd> pip install -U Package
```

例如：X:\Programs Files\Python38\Scripts>pip install -U jieba。

说明：U为大写字母，Package为库名称。

1.4 任务实现

1. Python的下载、安装与使用

(1) 打开Python的官方网站(https://www.python.org)，如图1-1所示，在Downloads菜单下选择要安装的操作系统类型，以Windows为例，如图1-2所示，单击选择Windows命令，在打开的窗口中找到需要的版本(如python-3.8.0-amd64.exe)，下载该版本文件即可。

(2) 双击下载的程序文件，如python-3.8.0-amd64.exe，打开如图1-3所示的对话框。其中Install Now选项为直接安装，Customize installation选项为自定义安装，若选择Install launcher for all users(recommended)复选框表示为所有用户执行安装(推荐)，若选择Add Python 3.8 to PATH复选框表示添加Python 3.8到系统环境路径中。

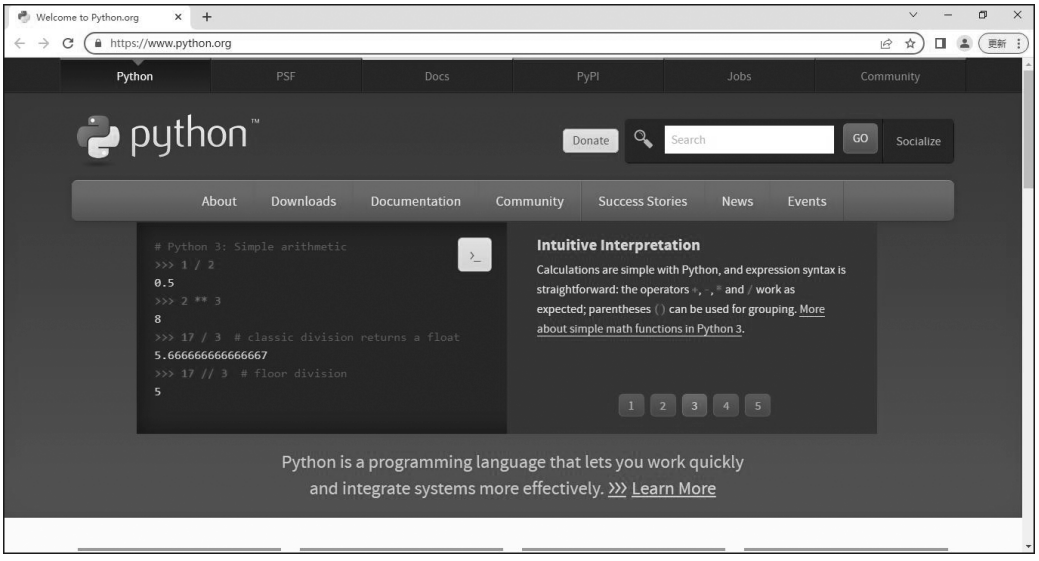


图 1-1　Python 官方网站主页

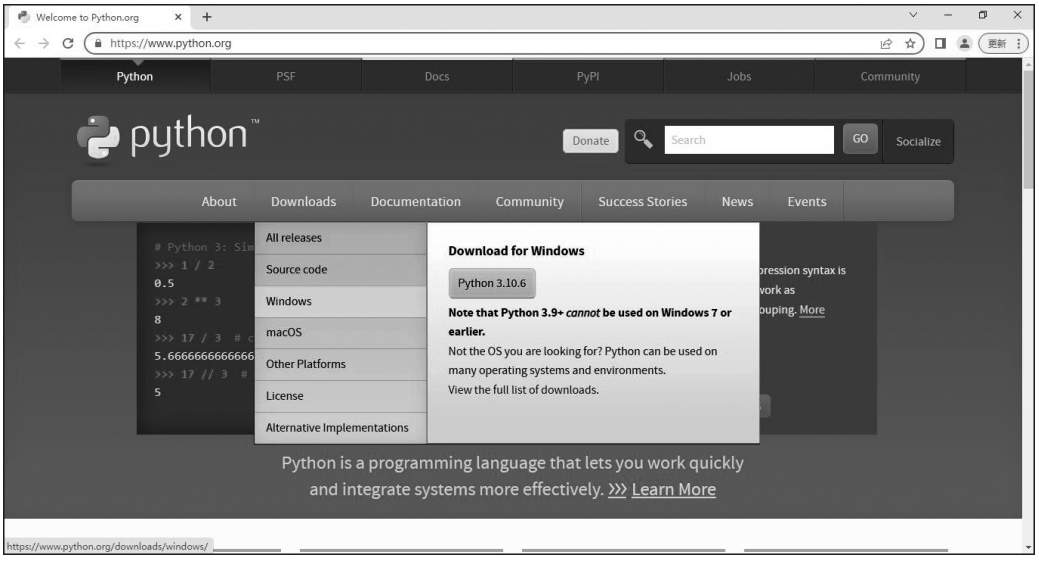


图 1-2　选择 Windows 命令下载所需版本

在此，选择自定义安装，并选中 Add Python 3.8 to PATH 复选框，然后单击 Customize installation 进行自定义安装，打开如图 1-4 所示对话框。

(3) 使用默认设置，单击 Next 按钮，打开如图 1-5(a)所示的对话框，根据需要进行相应的设置，如选中 Install for all users 选项，如图 1-5(b)所示。

图 1-3 Python 安装向导

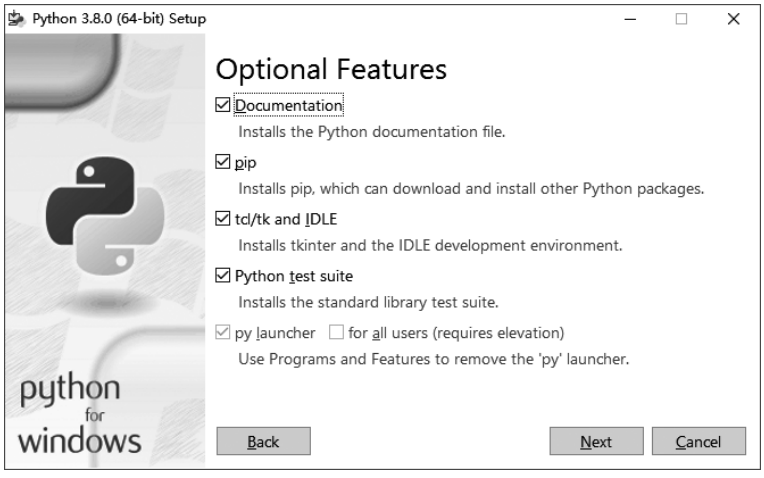
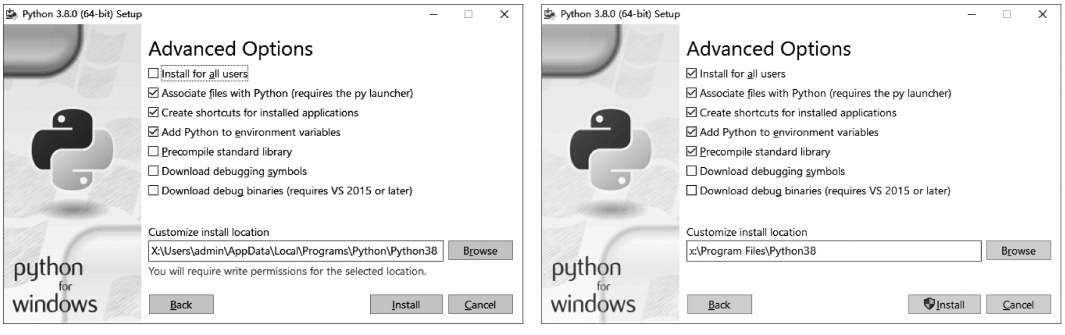


图 1-4 Python 安装自定义项



(a) (b)

图 1-5 Python 高级选项及安装路径

(4) 单击 Install 按钮开始安装,安装进度如图 1-6 所示。

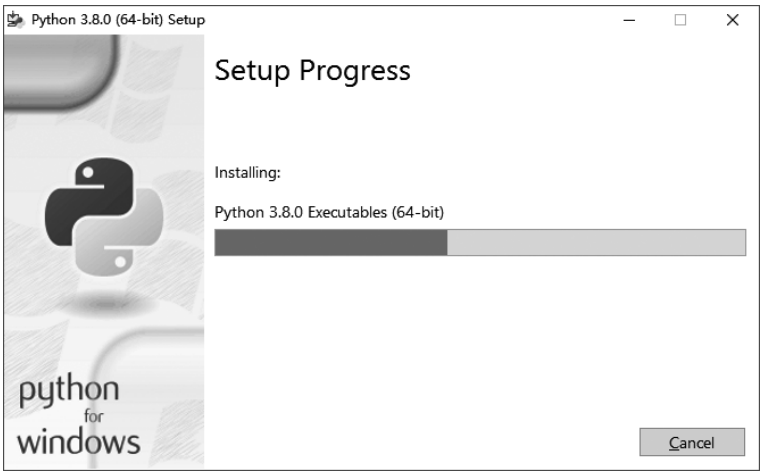


图 1-6 Python 安装进度对话框

(5) 安装完成后如图 1-7 所示。

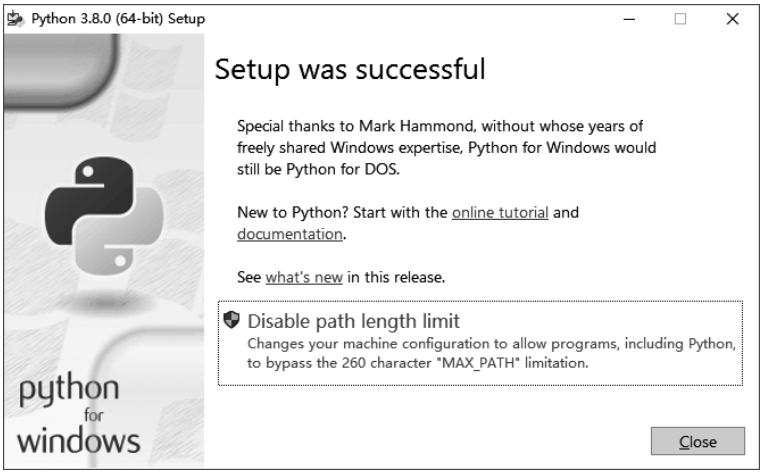


图 1-7 Python 安装完成对话框

(6) 单击 Close 按钮,完成安装。

(7) 安装完成后,打开命令行窗口,进入默认安装的文件夹 Python 3.8 输入 python 后,按 Enter 键,如图 1-8 所示,则表示安装配置成功。

(8) 启动 IDLE。安装好 Python 后,将会在 Windows 菜单中出现如图 1-9 所示的 Python 3.8 文件夹。单击选择 IDLE(Python 3.8 64-bit)命令,即可进入 IDLE 编辑环境。

2. Anaconda3 的安装与使用

(1) 打开 Anaconda 的官方网站(https://www.anaconda.com),如图 1-10 所示,单击 Download 按钮,选择需要安装的操作系统类型,然后选择需要的软件版本下载即可。

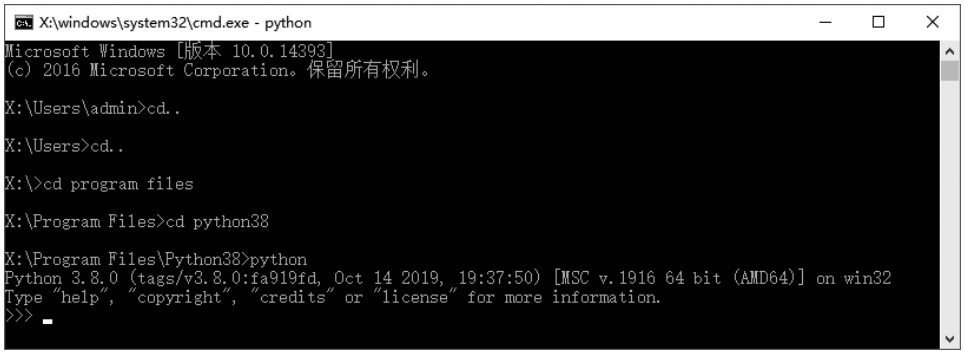


图 1-8 测试 Python 安装及配置成功

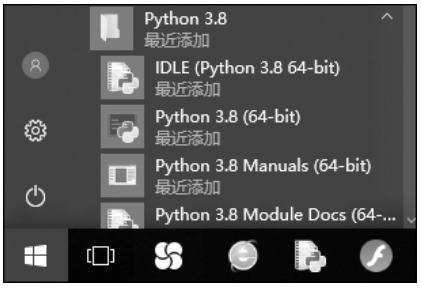


图 1-9 开始菜单中的 Python 3.8 文件夹

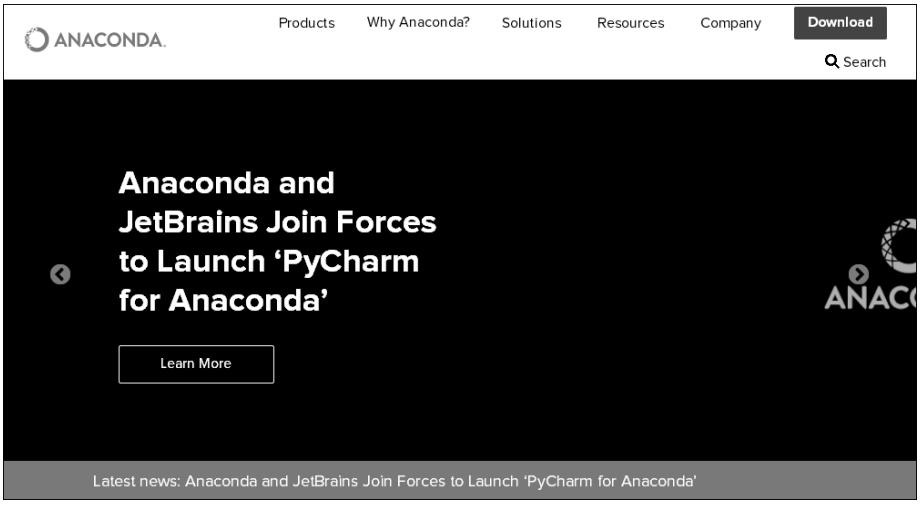


图 1-10 Anaconda 官方网站

(2) 双击下载的程序文件,如 Anaconda3-5.2.0-Windows-x86_64.exe,打开如图 1-11 所示对话框。单击“运行”按钮,打开如图 1-12 所示对话框。

(3) 单击 Next 按钮,打开如图 1-13 所示对话框。

图 1-11　Anaconda3 安全警告

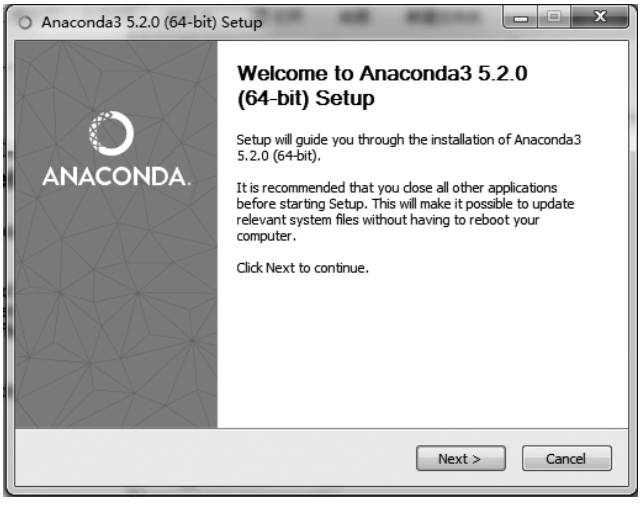


图 1-12　Anaconda3 安装对话框

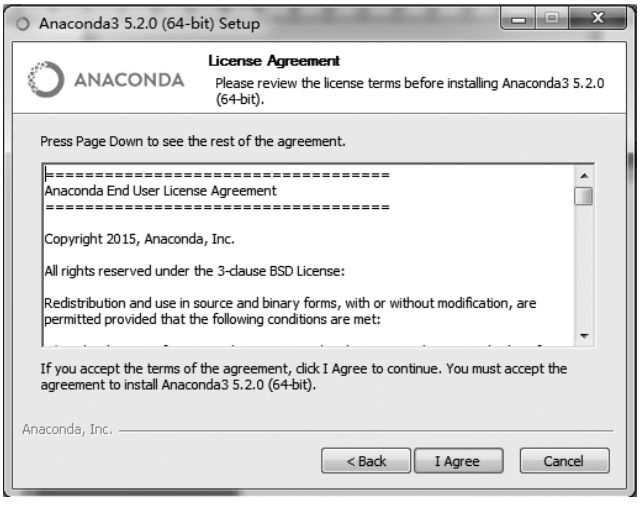


图 1-13　Anaconda3 安装许可协议对话框