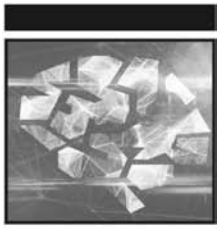

Python 机器学习

[新加坡] 李伟梦(Wei-Meng Lee) 著
李周芳 译

清华大学出版社
北 京

北京市版权局著作权合同登记号 图字：01-2019-4942

Wei-Meng Lee
Python Machine Learning
EISBN：978-1-119-54563-7

图书在版编目(CIP)数据

Python 机器学习 / (新加坡)李伟梦 著；李周芳 译. —北京：清华大学出版社，2020.5
书名原文：Python Machine Learning
ISBN 978-7-302-55197-3

Ⅰ. ①P…　Ⅱ. ①李… ②李…　Ⅲ. ①软件工具—程序设计　Ⅳ. ①TP311.561

中国版本图书馆 CIP 数据核字(2020)第 050468 号

责任编辑：王　军　韩宏志
装帧设计：孔祥峰
责任校对：牛艳敏
责任印制：刘海龙

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 装 者：三河市吉祥印务有限公司
经　　销：全国新华书店
开　　本：170mm×240mm　　印　　张：18.75　　字　　数：368 千字
版　　次：2020 年 6 月第 1 版　　印　　次：2020 年 6 月第 1 次印刷
定　　价：68.00 元

产品编号：084810-01

译 者 序

机器学习(Machine Learning，ML)是一门多领域交叉学科，涉及概率论、统计学、逼近论、凸分析、算法复杂度理论等多门学科。专门研究计算机怎样模拟或实现人类的学习行为，以获取新的知识或技能，重新组织已有的知识结构，使之不断改善自身的性能。机器学习是继专家系统之后人工智能应用的又一重要研究领域，也是人工智能和神经计算的核心研究课题之一。对机器学习的讨论和机器学习研究的进展，必将促使人工智能和整个科学技术进一步发展。

机器学习是人工智能的核心，是使计算机具有智能的根本途径，其应用遍及人工智能的各个领域，例如，数据挖掘、计算机视觉、自然语言处理、生物特征识别、搜索引擎、医学诊断、检测信用卡欺诈、证券市场分析、DNA 序列测序、语音和手写识别、战略游戏和机器人运用。

按学习形式分类，机器学习可以分为监督学习和非监督学习。监督学习主要应用于分类和预测，是从给定的训练数据集中分析出一个函数，当新的数据到来时，可以根据这个函数预测结果。而非监督学习又称归纳性学习，利用 K 方式、建立中心，通过循环和递减运算来减小误差，达到分类的目的。

市场上的大多数书要么太肤浅，要么过于深奥让初学者望而生畏。本书摒弃那种展现机器学习中核心算法和理论的方式，只简单介绍 Python 中一些使机器学习成为可能的基本库，例如如何使用 NumPy 库操作数字数组，如何使用 Pandas 库处理表格数据，如何使用 matplotlib 库可视化数据。

然后讨论进行机器学习的准备工作，例如获取样例数据集、生成自己的数据集、执行数据清理以及从数据集中删除异常值等。接着通过示例展示常用的机器学习算法，如分类、回归、聚类；主要讲解了线性回归、逻辑回归、SVM(包括线性内核和 RBF 内核)、kNN 等算法。

本书还包含一章，主要介绍如何使用 Microsoft Azure Machine Learning Studio，通过拖放操作来构建机器学习模型，而不需要编写代码。最后讨论如何部署所构建的模型，以使运行在移动和桌面设备上的客户机应用程序可以使用这些模型。

本书的主要意图是让尽可能多的开发人员能够阅读本书。本书不要求读者具有深厚的知识背景，而是在必要时介绍其他一些学科的基本概念，但读者应该具备一些 Python 编程的基本知识，以及一些基本的统计知识。

本书可作为计算机科学与工程、统计学和社会科学等专业的大学生或研究生的教材，也可作为软件研究人员或从业人员的参考资料。

在这里要感谢清华大学出版社的编辑，他们为本书的翻译投入了巨大的热情并付出了很多心血。没有他们的帮助和鼓励，本书不可能顺利付梓。

对于这本经典之作，译者本着“诚惶诚恐”的态度，在翻译过程中力求“信、达、雅”，但是鉴于译者水平有限，错误和失误在所难免，如有任何意见和建议，请不吝指正。

译　者

作者简介

Wei-Meng Lee 是一名技术专家，也是 Developer Learning Solutions 公司(http://www.learn2development.net)的创始人，该公司专门从事最新技术的实践培训。

Wei-Meng 具有多年的培训经验，他的培训课程特别强调“边做边学”。他动手学习编程的方法使理解这个主题比仅阅读书籍、教程和文档容易得多。

Wei-Meng 这个名字经常出现在网上和印刷出版物，如 DevX.com、MobiForge.com 和 *CoDe* 杂志。

技术编辑简介

Doug Mahugh 是一名软件开发人员，他于 1978 年作为波音公司的 Fortran 程序员开始了他的职业生涯。Doug 自 2005 年以来一直在微软工作，承担各种工作，包括开发人员宣传、标准制订和内容开发。自 2008 年学习 Python 以来，Doug 编写了一些示例和教程，主题涉及缓存、持续集成乃至 Azure Active Directory 身份验证和 Microsoft Graph。Doug 曾在 20 多个国家的行业活动上发言，他是微软在 ISO/IEC、Ecma International、OASIS、CalConnect 等标准组织的技术代表。

Doug 目前和他的妻子 Megan 一起居住在西雅图。他还养了两只萨摩耶犬杰米和爱丽丝。

致　谢

撰写书籍总是令人兴奋的，但随之而来的是长时间的艰苦工作，以求把事情做得准确无误。为使本书面世，许多无名英雄不知疲倦地在幕后工作。为此，我想借此机会感谢一些特殊的人，是他们使本书成功面世。

首先，我要感谢组稿编辑 Devon Lewis，他是我撰写本书的第一个联系人。谢谢 Devon 给我这个机会，谢谢你对我的信任!

接下来，非常感谢我的项目编辑 Gary Schwartz，他一直是我的合作伙伴。Gary 总是与他人保持着联系，即使他在机场！Gary 对我很有耐心，尽管我好几次错过了撰写本书的最后期限。我知道这对他的计划是个障碍，但他总是乐于助人。和他一起工作，我知道我的书得到了很好的处理。非常感谢你，Gary!

同样重要的是技术编辑 Doug Mahugh。Doug 一直非常敏锐地编辑和测试我的代码，如果事情没有按照预期进行，他总是让我知道。谢谢你发现我的错误，让本书变得更好，Doug！我也想借此机会感谢制作编辑 Barath Kumar Rajasekaran。如果没有他的努力，本书就不可能出版。谢谢你，Barath!

最后，但并非最不重要的，要感谢我的父母和妻子 Sze Wa，他们给了我所有的支持。当我在撰写本书的时候，他们无私地调整了时间表来适应我繁忙的日程。我爱你们所有人!

前　言

本书介绍了机器学习，这是近年来最热门的话题之一。目前设备的计算能力呈指数级数增长，同时价格在不断下降，这是了解机器学习的最佳时机。机器学习任务通常需要非常强大的处理能力，但现在可以在台式机上完成。然而，机器学习并不适合胆小的人——你需要具备良好的数学、统计学基础和编程知识。市场上的大多数书要么太肤浅，要么过于深奥让初学者望而生畏。

本书将对这个问题采取温和的态度。首先，本书介绍 Python 中使用的一些使机器学习成为可能的基本库。特别是，学习如何使用 NumPy 库操作数字数组，如何使用 Pandas 库处理表格数据。完成这些之后，学习如何使用 matplotlib 库可视化数据，它允许绘制不同类型的图表和图形，以便轻松地可视化数据。

一旦牢固地掌握了基础知识，就可以开始使用 Python 和 Scikit-learn 库进行机器学习。这样可以深入了解各种机器学习算法幕后的工作原理。

本书将介绍常用的机器学习算法，如回归、聚类和分类。

本书还包含一章，介绍如何使用 Microsoft Azure Machine Learning Studio 进行机器学习，该工具允许开发人员开始使用拖放操作来构建机器学习模型，而不需要编写代码。最重要的是，不需要深入掌握机器学习知识。

最后讨论如何部署所构建的模型，以便运行在移动和桌面设备上的客户机应用程序可以使用这些模型。

本书的主要意图是让尽可能多的开发人员能够阅读本书。要从本书中得到最大的收获，应该具备一些 Python 编程的基本知识，以及一些基本的统计知识。就像永远不可能仅通过阅读一本书就学会游泳一样，强烈建议在阅读章节时尝试一下示例代码。继续修改代码，看看输出是如何变化的，你常会对自己能做的工作感到惊讶。

本书中的所有样例代码都可用于 Jupyter Notebook。要下载样例代码，可访问本书的支持页面 http://www.tupwk.com.cn/downpage，然后输入本书 ISBN 或中文名。另外，也可扫本书封底二维码下载。下载后，可马上试用。

不要拖延了，欢迎阅读本书!

目　录

第 1 章　机器学习简介 …… 1
1.1　什么是机器学习? …… 2
1.1.1　在本书中机器学习将解决什么问题? …… 3
1.1.2　机器学习算法的类型 …… 4
1.2　可得到的工具 …… 7
1.2.1　获取 Anaconda …… 8
1.2.2　安装 Anaconda …… 8
1.3　本章小结 …… 17

第 2 章　使用 NumPy 扩展 Python …… 19
2.1　NumPy 是什么? …… 19
2.2　创建 NumPy 数组 …… 20
2.3　数组索引 …… 22
2.3.1　布尔索引 …… 22
2.3.2　切片数组 …… 23
2.3.3　NumPy 切片是一个引用 …… 25
2.4　重塑数组 …… 26
2.5　数组数学 …… 27
2.5.1　点积 …… 29
2.5.2　矩阵 …… 30
2.5.3　累积和 …… 31
2.5.4　NumPy 排序 …… 32
2.6　数组赋值 …… 34

2.6.1 通过引用复制 …… 34
2.6.2 按视图复制(浅复制) …… 35
2.6.3 按值复制(深度复制) …… 37
2.7 本章小结 …… 37

第 3 章 使用 Pandas 处理表格数据 …… 39
3.1 Pandas 是什么? …… 39
3.2 Pandas Series …… 40
3.2.1 使用指定索引创建 Series …… 41
3.2.2 访问 Series 中的元素 …… 41
3.2.3 指定 Datetime 范围作为 Series 的索引 …… 42
3.2.4 日期范围 …… 43
3.3 Pandas DataFrame …… 44
3.3.1 创建 DataFrame …… 45
3.3.2 在 DataFrame 中指定索引 …… 46
3.3.3 生成 DataFrame 的描述性统计信息 …… 47
3.3.4 从 DataFrame 中提取 …… 48
3.3.5 选择 DataFrame 中的单个单元格 …… 54
3.3.6 基于单元格值进行选择 …… 54
3.3.7 转置 DataFrame …… 54
3.3.8 检查结果是 DataFrame 还是 Series …… 55
3.3.9 在 DataFrame 中排序数据 …… 55
3.3.10 将函数应用于 DataFrame …… 57
3.3.11 在 DataFrame 中添加和删除行和列 …… 60
3.3.12 生成交叉表 …… 63
3.4 本章小结 …… 64

第 4 章 使用 matplotlib 显示数据 …… 67
4.1 什么是 matplotlib? …… 67
4.2 绘制折线图 …… 67
4.2.1 添加标题和标签 …… 69
4.2.2 样式 …… 69
4.2.3 在同一图表中绘制多条线 …… 71
4.2.4 添加图例 …… 72
4.3 绘制柱状图 …… 73
4.3.1 在图表中添加另一个柱状图 …… 74

4.3.2　更改刻度标签 …… 76
4.4　绘制饼图 …… 77
4.4.1　分解各部分 …… 79
4.4.2　显示自定义颜色 …… 79
4.4.3　旋转饼状图 …… 80
4.4.4　显示图例 …… 81
4.4.5　保存图表 …… 83
4.5　绘制散点图 …… 83
4.5.1　合并图形 …… 84
4.5.2　子图 …… 85
4.6　使用 Seaborn 绘图 …… 86
4.6.1　显示分类图 …… 87
4.6.2　显示 lmplot …… 89
4.6.3　显示 swarmplot …… 90
4.7　本章小结 …… 92

第 5 章　使用 Scikit-learn 开始机器学习 …… 93
5.1　Scikit-learn 简介 …… 93
5.2　获取数据集 …… 93
5.2.1　使用 Scikit-learn 数据集 …… 94
5.2.2　使用 Kaggle 数据集 …… 97
5.2.3　使用 UCI 机器学习存储库 …… 97
5.2.4　生成自己的数据集 …… 97
5.3　Scikit-learn 入门 …… 100
5.3.1　使用 LinearRegression 类对模型进行拟合 …… 101
5.3.2　进行预测 …… 101
5.3.3　绘制线性回归线 …… 102
5.3.4　得到线性回归线的斜率和截距 …… 103
5.3.5　通过计算残差平方和检验模型的性能 …… 104
5.3.6　使用测试数据集评估模型 …… 105
5.3.7　持久化模型 …… 106
5.4　数据清理 …… 108
5.4.1　使用 NaN 清理行 …… 108
5.4.2　删除重复的行 …… 110
5.4.3　规范化列 …… 112
5.4.4　去除异常值 …… 113

5.5 本章小结 117

第 6 章 有监督的学习——线性回归 119
6.1 线性回归的类型 119
6.2 线性回归 120
6.2.1 使用 Boston 数据集 120
6.2.2 数据清理 125
6.2.3 特征选择 126
6.2.4 多元回归 129
6.2.5 训练模型 131
6.2.6 获得截距和系数 133
6.2.7 绘制三维超平面 134
6.3 多项式回归 136
6.3.1 多项式回归公式 138
6.3.2 Scikit-learn 中的多项式回归 138
6.3.3 理解偏差和方差 142
6.3.4 对 Boston 数据集使用多项式多元回归 145
6.3.5 绘制三维超平面 146
6.4 本章小结 149

第 7 章 有监督的学习——使用逻辑回归进行分类 151
7.1 什么是逻辑回归? 151
7.1.1 理解概率 153
7.1.2 logit 函数 153
7.1.3 sigmoid 曲线 155
7.2 使用威斯康星乳腺癌诊断数据集 156
7.2.1 检查特征之间的关系 157
7.2.2 使用一个特征训练 161
7.2.3 使用所有特性训练模型 164
7.3 本章小结 174

第 8 章 有监督的学习——使用支持向量机分类 175
8.1 什么是支持向量机? 175
8.1.1 最大的可分性 176
8.1.2 支持向量 177
8.1.3 超平面的公式 178
8.1.4 为 SVM 使用 Scikit-learn 179

8.1.5　绘制超平面和边距 …… 182
8.1.6　进行预测 …… 183
8.2　内核的技巧 …… 184
8.2.1　添加第三个维度 …… 185
8.2.2　绘制三维超平面 …… 187
8.3　内核的类型 …… 189
8.3.1　C …… 193
8.3.2　径向基函数(RBF)内核 …… 195
8.3.3　gamma …… 196
8.3.4　多项式内核 …… 198
8.4　使用 SVM 解决实际问题 …… 199
8.5　本章小结 …… 202

第 9 章　有监督的学习——使用 k-近邻(kNN)分类 …… 203
9.1　k-近邻是什么? …… 203
9.1.1　用 Python 实现 kNN …… 204
9.1.2　为 kNN 使用 Scikit-learn 的 KNeighborsClassifier 类 …… 209
9.2　本章小结 …… 218

第 10 章　无监督学习——使用 k-means 聚类 …… 219
10.1　什么是无监督学习? …… 219
10.1.1　使用 k-means 的无监督学习 …… 220
10.1.2　k-means 中的聚类是如何工作的 …… 220
10.1.3　在 Python 中实现 k-means …… 223
10.1.4　在 Scikit-learn 中使用 k-means …… 228
10.1.5　利用 Silhouette 系数评价聚类的大小 …… 230
10.2　使用 k-means 解决现实问题 …… 234
10.2.1　导入数据 …… 234
10.2.2　清理数据 …… 235
10.2.3　绘制散点图 …… 236
10.2.4　使用 k-means 聚类 …… 236
10.2.5　寻找最优尺寸类 …… 238
10.3　本章小结 …… 239

第 11 章　使用 Azure Machine Learning Studio …… 241
11.1　什么是 Microsoft Azure Machine Learning Studio? …… 241
11.1.1　以泰坦尼克号实验为例 …… 241

11.1.2 使用 Microsoft Azure Machine Learning Studio ····· 243
11.1.3 训练模型 ····· 254
11.1.4 将学习模型作为 Web 服务发布 ····· 258
11.2 本章小结 ····· 263

第 12 章 部署机器学习模型 ····· 265
12.1 部署 ML ····· 265
12.2 案例研究 ····· 266
12.2.1 加载数据 ····· 267
12.2.2 清理数据 ····· 267
12.2.3 检查特征之间的相关性 ····· 269
12.2.4 绘制特征之间的相关性 ····· 270
12.2.5 评估算法 ····· 273
12.2.6 训练并保存模型 ····· 275
12.3 部署模型 ····· 277
12.4 创建客户机应用程序来使用模型 ····· 279
12.5 本章小结 ····· 281

第 1 章

机器学习简介

你正在阅读本书，这清楚地表明你关注机器学习这个非常有趣、令人兴奋的话题。

本书涵盖了近年来最热门的编程主题之一——机器学习。机器学习(Machine Learning，ML)是一组算法和技术的集合，用于设计从数据中学习的系统。然后，这些系统能根据所提供的数据进行预测或推断模式。

目前设备的计算能力呈指数级数增长，同时价格在不断下降，这是了解机器学习的最佳时机。机器学习任务通常需要非常强大的处理能力，但现在可在台式机上完成。然而，机器学习并不适合胆小的人——你需要具备良好的数学、统计学基础和编程知识。市面上大多数关于机器学习的书籍都过于强调细节，这常让初学者望而生畏。大多数关于机器学习的讨论都是围绕着统计理论和算法展开的，所以除非是数学家或博士研究生，否则你可能发现它们很难理解。对于大多数人，特别是开发人员，他们想要的是对机器学习的工作原理有一个基本的了解，最重要的是，明白如何在应用程序中应用机器学习。这就是撰写本书的动机。

本书采用温和的方法来介绍机器学习，努力做到以下几点：

- 涵盖为机器学习奠定基础的 Python 库，即 NumPy、Pandas 和 matplotlib。
- 讨论使用 Python 和 Scikit-learn 库进行机器学习。如果可能的话，本书将使用 Python 手工实现相关的机器学习算法，以便了解各种机器学习算法如何在后台工作。之后展示如何使用 Scikit-learn 库，它很容易将机器学习集成到自己的应用程序中。
- 涵盖了常见的机器学习算法——回归、聚类和分类。

提示：

本书不打算深入讨论机器学习算法。虽然有一些章节讨论了算法背后的一些数学概念，但其意图是使这个主题易于理解，并希望能激励读者进一步学习。

机器学习确实是一个非常复杂的话题。但是，本书不讨论它背后复杂的数学理论，而是使用易于理解的示例来介绍它，并给出大量代码示例。本书中的代码很多，鼓励读者试用各个章节中的大量示例，这些章节相互独立、结构紧凑、易于遵循和理解。

1.1 什么是机器学习?

只要编写过程序，就会熟悉图 1.1 中所示的关系图。编写一个程序，输入一些数据，就会得到输出。例如，编写一个程序来执行公司的一些会计任务。这种情况下，收集的数据将包括销售记录、库存清单等。然后，该程序将接收数据，并根据销售记录计算利润或亏损。也可制作一些漂亮的图表来展示销售业绩。这种情况下，输出是损益表以及其他图表。

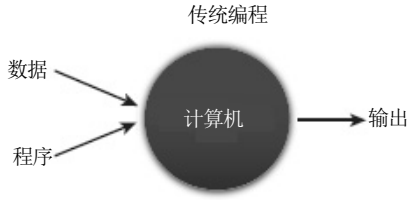

图 1.1　在传统编程中，数据和程序产生输出

多年来，传统的桌面和 Web 编程一直占据着主导地位，许多算法和方法都在不断发展，以提高程序的运行效率。然而，近年来，机器学习已经接管了编程界。机器学习将图 1.1 中的范例转换为一个新范例，如图 1.2 所示。现在不是将数据提供给程序，而是使用收集到的数据和输出来派生程序(也称为模型)。使用前面的会计示例，在机器学习范例中，将获取详细的销售记录(是数据和输出的统称)，并使用它们派生出一组规则来进行预测。可用这个模型来预测明年最受欢迎的商品，或者哪些商品不那么受欢迎。

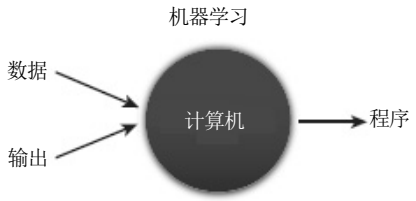

图 1.2　在机器学习中，数据和输出产生程序

提示：

机器学习就是在数据中寻找模式。

1.1.1　在本书中机器学习将解决什么问题?

那么，机器学习到底是什么？机器学习(ML)是算法和技术的集合，用于设计从数据中学习的系统。了解 ML 算法需要有很强的数学和统计基础，但不大需要区域知识。ML 由以下学科组成：

- 科学计算
- 数学
- 统计

机器学习一个很好的应用是试图确定某个特定的信用卡交易是否存在欺诈。给定过去的事务记录，数据科学家的工作是根据区域知识清理和转换数据，以便应用正确的 ML 算法来解决问题(在本例中，确定事务是否存在欺诈)。数据科学家需要知道哪种机器学习方法最有助于完成这项任务，以及如何应用它。数据科学家不一定需要知道这种方法是如何工作的，但知道这一点总是有助于建立更精确的学习模型。

本书想用机器学习来解决三种主要类型的问题。这些问题类型如下。

(1) 分类：这是 A 还是 B?

(2) 回归：多少?

(3) 聚类：这是如何组织的?

1. 分类

在机器学习中，分类是根据所观察到的类别中包含的训练数据集，确定一个新观察到的数据集属于哪一组类别。以下是一些分类问题的例子：

- 预测 2020 年美国总统大选的获胜者
- 预测肿瘤是否癌变
- 区分不同类型的花

具有两个类的分类问题称为两个类的分类问题。具有两个以上类的问题称为

多类的分类问题。

分类问题的结果是一个离散值，表示预测观察值所在的类。分类问题的结果也可以是一个连续值，表示观察值属于特定类的可能性。例如，预测候选人 A 赢得选举的概率为 0.65(或 65%)。这里，0.65 是表示预测置信度的连续值，通过选择概率最高的预测，可将其转换为一个类值(本例中为“赢得选举”)。

第 7～9 章将详细讨论分类。

2. 回归

回归通过估计变量之间的关系来帮助预测未来。与分类不同，回归返回一个连续的输出变量。下面是一些回归问题的例子：

- 预测某一特定产品下季度的销售数字
- 预测下周的气温
- 预测特定型号轮胎的使用寿命

第 6 章将详细讨论回归。

3. 聚类

聚类有助于将相似的数据点分组成直观的组。给定一组数据，通过将它们分组为自然块，聚类有助于发现它们是如何组织的。

聚类问题的例子如下：

- 哪些观众喜欢同一类型的电影
- 哪些型号的硬盘驱动器会以同样的方式失败

为在数据中发现特定模式，聚类非常有用。第 10 章将详细讨论聚类。

1.1.2 机器学习算法的类型

机器学习算法分为两大类：

- 监督学习算法使用标注的数据进行训练。换句话说，这种数据包含带有期望答案的示例。例如，识别欺诈性信用卡使用情形的模型，会利用数据点标有已知欺诈性和有效收费的数据集进行训练。大多数机器学习都是有监督的。
- 无监督学习算法适用于没有标签的数据，其目标是发现数据中的关系。例如，可能希望找到具有类似购买习惯的客户统计数据组。

1. 有监督的学习

在有监督的学习中，使用有标记的数据集。有标记的数据集意味着一组数据已被标记。这个标记为数据提供了信息意义。使用标记，可以预测未标记的数据，来获得一个新标记。例如，数据集可能由一系列包含以下字段的记录组成，这些字段记录了不同房屋的面积和售价：

房子面积，售价

在这个非常简单的例子中，“售价”就是标记。当绘制在图表上时(见图 1.3)，这个数据集可以帮助预测尚未售出的房子的价格。预测房价是一个回归问题。

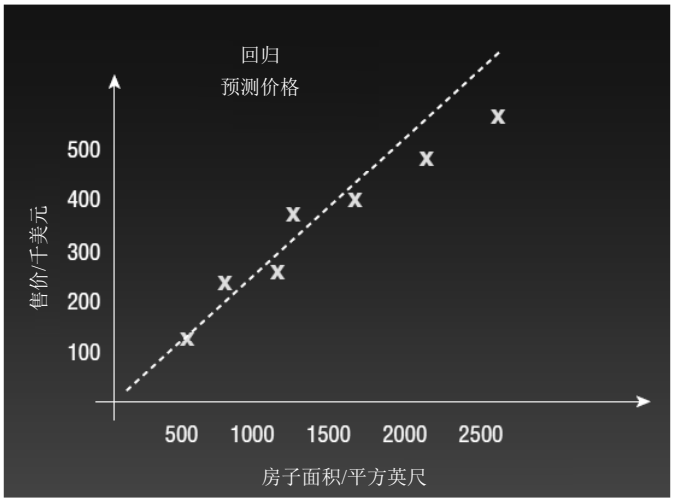

图 1.3　运用回归方法预测房屋的预期售价

在另一个例子中，假设有一个包含以下内容的数据集：

肿瘤大小，年龄，恶性

“恶性”字段是一个标记，表明肿瘤是否癌变。在图表中绘制数据集时(见图 1.4)，就能将数据集分为两组，一组包含癌性肿瘤，另一组包含良性肿瘤。使用这个分组，现在可预测新的肿瘤是否癌变。这类问题称为分类问题。

提示：

第 6 ~ 9 章将详细讨论有监督学习算法。

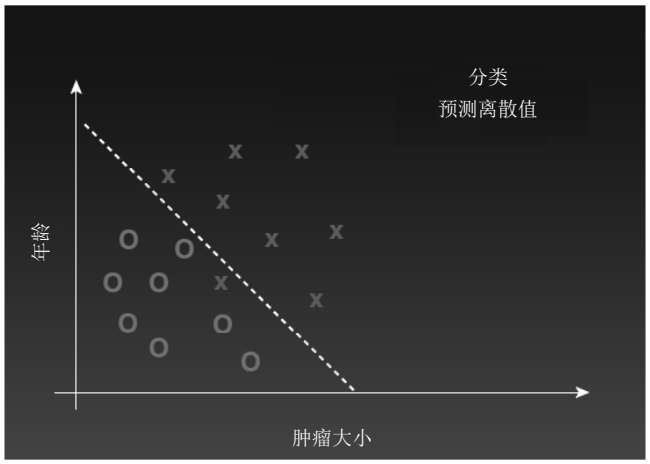

图 1.4　使用分类方法将数据分为不同的类

2. 无监督学习

在无监督学习中，使用的数据集没有标记。查看未标记数据的一个简单方法是考虑包含一组人的腰围和腿长的数据集：

腰围，腿长

使用无监督学习，需要尝试预测数据集中的模式。可在图表中绘制数据集，如图 1.5 所示。

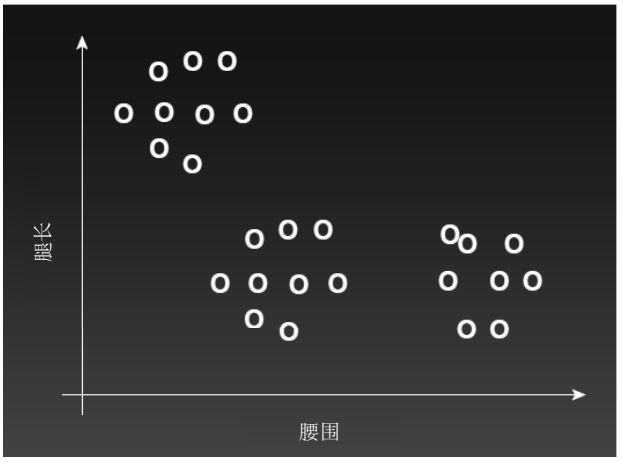

图 1.5　绘制未标记的数据

然后，可使用一些聚类算法来查找数据集中的模式。最终结果可能是在数据中发现三个不同的聚类组，如图 1.6 所示。

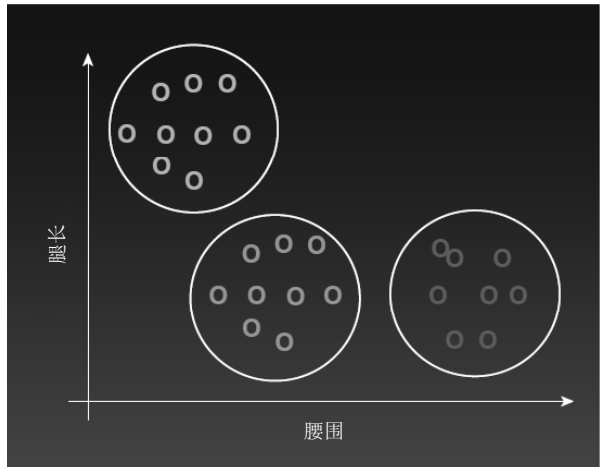

图 1.6　将这些点聚在不同的组中

提示：

第 10 章将详细讨论无监督学习算法。

1.2　可得到的工具

对于本书，所有示例都使用 Python 3 和 Scikit-learn 库进行测试，Scikit-learn 库是一个 Python 库，它实现了各种类型的机器学习算法，如分类、回归、聚类、决策树等。除了 Scikit-learn 外，还将使用一些互补的 Python 库——NumPy、Pandas 和 matplotlib。

虽然可在计算机上独立安装 Python 解释器和其他库，但安装所有这些库的无故障方法是安装 Anaconda 包。Anaconda 是一个免费的 Python 发行版，提供了创建数据科学和机器学习项目需要的所有必要库。

Anaconda 包括以下内容：

- 核心 Python 语言
- 各种 Python 包(库)
- conda(Anaconda 自己的包管理器)，用于更新 Anaconda 和包
- Jupyter Notebook(以前称为 iPython Notebook)，一个用于 Python 项目的基于 Web 的编辑器

使用 Anaconda，可灵活地安装不同的语言(R、JavaScript、Julia 等)在 Jupyter Notebook 中工作。

1.2.1 获取 Anaconda

要下载 Anaconda，请访问 https://www.anaconda.com/download/。可为这些操作系统下载 Anaconda(见图 1.7)：

- Windows
- macOS
- Linux

为正在使用的平台下载 Python 3。

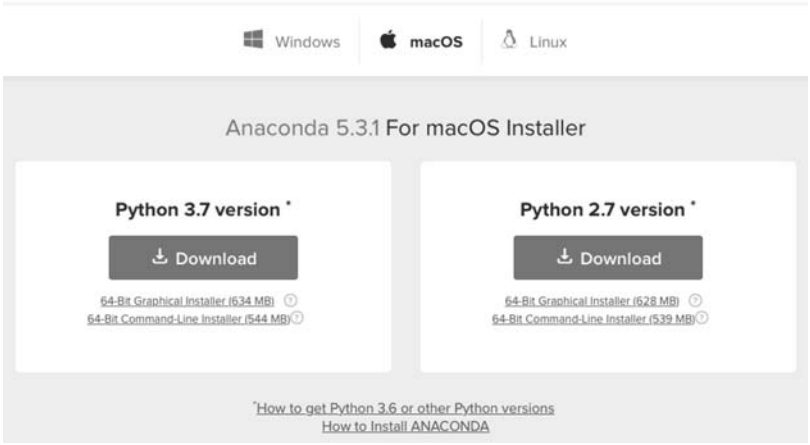

图 1.7　为 Python 3 下载 Anaconda

注意：

撰写本文时，Python 的版本是 3.7。

提示：

本书将使用 Python 3。因此，请务必下载包含 Python 3 的 Anaconda 的正确版本。

1.2.2 安装 Anaconda

安装 Anaconda 基本上是一个非事件过程。双击已下载的文件，并按照屏幕上显示的说明操作。特别是，Windows 版的 Anaconda 可以只安装给本地用户。此选项不需要管理员权限，因此对于在公司计算机上安装 Anaconda 的用户非常有用，这些计算机通常具有有限的用户权限。

一旦安装了 Anaconda，就希望启动 Jupyter Notebook。Jupyter Notebook 是一个开源 Web 应用程序，它允许创建和共享包含文档、代码等内容的文件。

1. 运行用于 macOS 的 Jupyter Notebook

要从 macOS 上启动 Jupyter，请启动终端并输入以下命令：

```
$ jupyter notebook
```

结果如下：

```
$ jupyter notebook
[I 18:57:03.642 NotebookApp] JupyterLab extension loaded from
/Users/weimenglee/anaconda3/lib/python3.7/site-packages/jupyterlab
[I 18:57:03.643 NotebookApp] JupyterLab application directory is
/Users/weimenglee/anaconda3/share/jupyter/lab
[I 18:57:03.648 NotebookApp] Serving notebooks from local directory:
/Users/weimenglee/Python Machine Learning
[I 18:57:03.648 NotebookApp] The Jupyter Notebook is running at:
[I 18:57:03.648 NotebookApp]
http://localhost:8888/?token=3700cfe13b65982612c0e1975ce3a681073
99b07f89b85fa
[I 18:57:03.648 NotebookApp] Use Control-C to stop this server and shut
down all kernels (twice to skip confirmation).
[C 18:57:03.649 NotebookApp]

    Copy/paste this URL into your browser when you connect for the first
time,
    to login with a token:
      http://localhost:8888/?token=3700cfe13b65982612c0e1975ce3a681073
99b07f89b85fa
[I 18:57:04.133 NotebookApp] Accepting one-time-token-authenticated
connection from ::1
```

实质上，Jupyter Notebook 会启动一个 Web 服务器，监听端口 8888。一段时间后，将启动 Web 浏览器(见图 1.8)。

提示：

Jupyter Notebook 的主页显示了该目录的内容。因此，在启动 Jupyter Notebook 之前，最好先切换到包含源代码的目录。

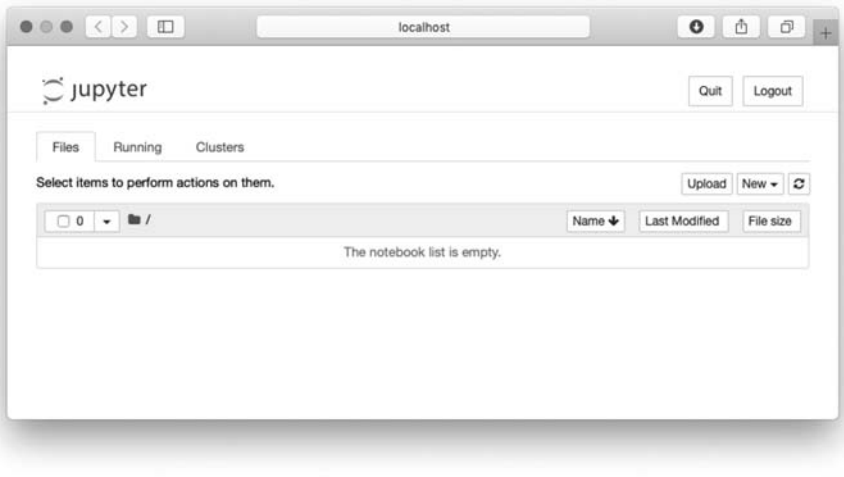

图 1.8　Jupyter Notebook 主页

2. 运行 Windows 版的 Jupyter Notebook

在 Windows 中启动 Jupyter Notebook 的最好方法是从 Anaconda 提示符启动它。Anaconda 提示符自动运行批处理文件 C:\Anaconda3\Scripts\activity.bat，参数如下：

```
C:\Anaconda3\Scripts\activate.bat C:\Anaconda3
```

提示：

注意，Anaconda3 文件夹的确切位置可能有所不同。例如，Windows 10 将 Anaconda 默认安装在 C:\Users\<username>\AppData\Local\Continuum\anaconda3，而不是 C:\anaconda3。

这为访问 Anaconda 及其库设置了必要的路径。要启动 Anaconda 提示符，请在 Windows Run 文本框中输入 Anaconda Prompt。要从 Anaconda 提示符下启动 Jupyter Notebook，请输入以下命令：

```
(base) C:\Users\Wei-Meng Lee\Python Machine Learning>jupyter notebook
```

结果如下：

```
[I 21:30:48.048 NotebookApp] JupyterLab beta preview extension
loaded from C:\Anaconda3\lib\site-packages\jupyterlab
[I 21:30:48.048 NotebookApp] JupyterLab application directory is
C:\Anaconda3\share\jupyter\lab
[I 21:30:49.315 NotebookApp] Serving notebooks from local directory:
```

```
    C:\Users\Wei-Meng Lee\Python Machine Learning
    [I 21:30:49.315 NotebookApp] 0 active kernels
    [I 21:30:49.322 NotebookApp] The Jupyter Notebook is running at:
    [I 21:30:49.323 NotebookApp]
    http://localhost:8888/?token=482bfe023bd77731dc132b5340f335b9e45
0ce5e1c4
    d7b2f
    [I 21:30:49.324 NotebookApp] Use Control-C to stop this server and
shut
    down all kernels (twice to skip confirmation).
    [C 21:30:49.336 NotebookApp]

       Copy/paste this URL into your browser when you connect for the
first time,
       to login with a token:
          http://localhost:8888/?token=482bfe023bd77731dc132b5340f3
35b9e45
    0ce5e1c4d7b2f
    [I 21:30:49.470 NotebookApp] Accepting one-time-token-authenticated
    connection from ::1
```

实际上，Jupyter Notebook 启动一个 Web 服务器，监听端口 8888。然后启动 Web 浏览器，显示图 1.9 中的页面。

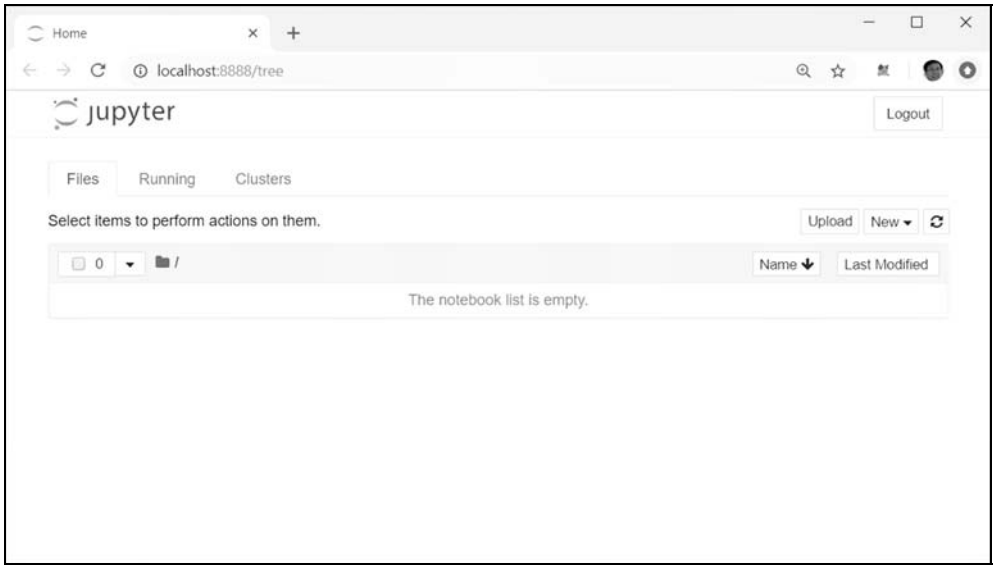

图 1.9　Jupyter Notebook 显示主页

3. 创建新的笔记本

要创建一个新的笔记本，找到屏幕右侧的 New 按钮并单击它。在下拉框中应该能够看到 Python 3(见图 1.10)。单击此选项。

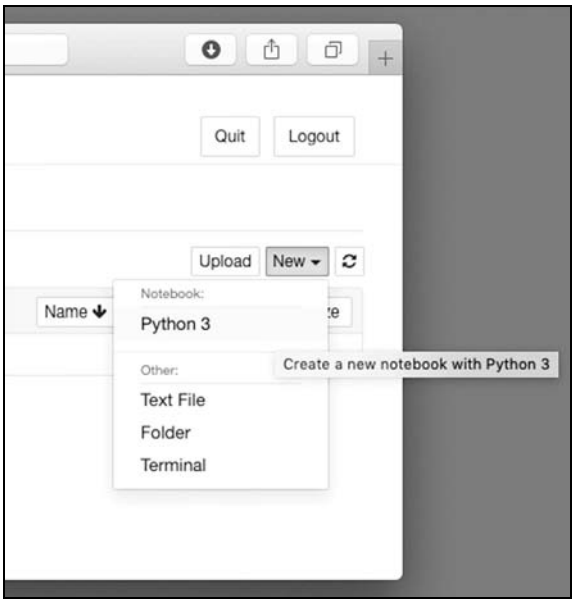

图 1.10　创建一个新的 Python 3 笔记本

现在会出现新笔记本(见图 1.11)。

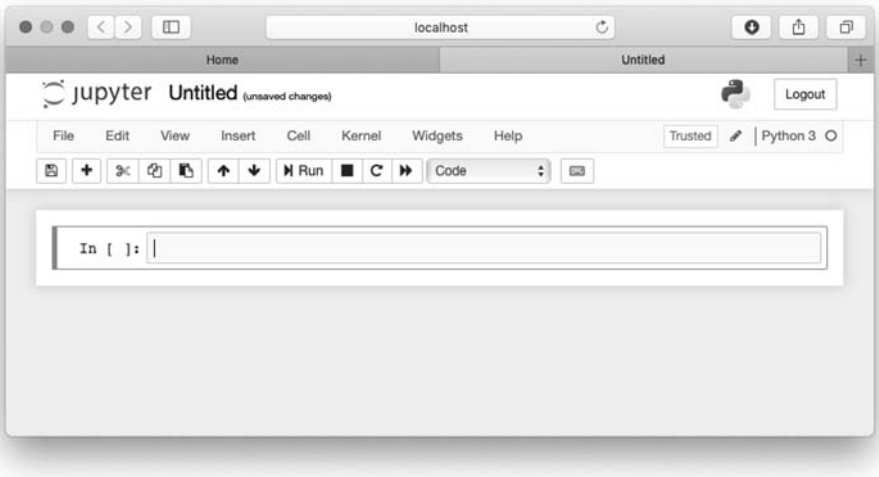

图 1.11　用 Jupyter Notebook 创建的 Python 3 笔记本

4. 给笔记本命名

默认情况下，笔记本命名为 Untitled。要给它一个合适的名称，单击 Untitled 并输入一个新名称。笔记本将保存在启动 Jupyter Notebook 的目录中。该笔记本用所提供的文件名以及.ipynb 扩展名来保存。

提示：

Jupyter Notebook 以前称为 iPython Notebook；因此扩展名是.ipynb。

5. 添加和删除单元格

笔记本包含一个或多个单元格。可在每个单元格中输入 Python 语句。使用 Jupyter Notebook，可将代码分成多个片段，并将它们放入单元格中，以便能够单独运行。

要向笔记本中添加更多单元格，请单击该按钮。还可以使用 Insert 菜单项并选择 Insert Cell Above 选项，在当前单元格之上添加新单元格，或者选择 Insert Cell Below 选项，在当前单元格之下添加新单元格。

图 1.12 显示了包含两个单元格的笔记本。

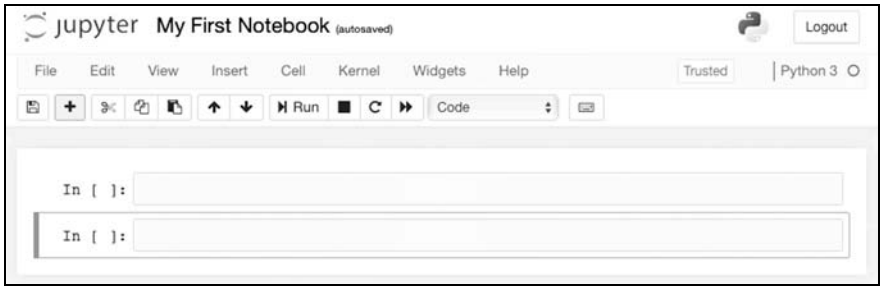

图 1.12　带两个单元格的笔记本

6. 运行一个单元格

Jupyter Notebook 中的每个单元都可独立运行。要执行(运行)单元格中的代码，请按 Ctrl+Enter 键，或单击鼠标悬停在单元格左侧时显示的箭头图标(见图 1.13)。

图 1.13　执行单元格中的代码

当单元格运行时，它们执行的顺序显示为一个运行编号。图 1.14 显示了按如下顺序执行的两个单元格。第一个单元格中的编号 1 表示首先执行该单元格，其次执行编号为 2 的第二个单元格。单元格的输出显示在单元格之后。如果回到第一个单元格并运行它，这个编号将变为 3。

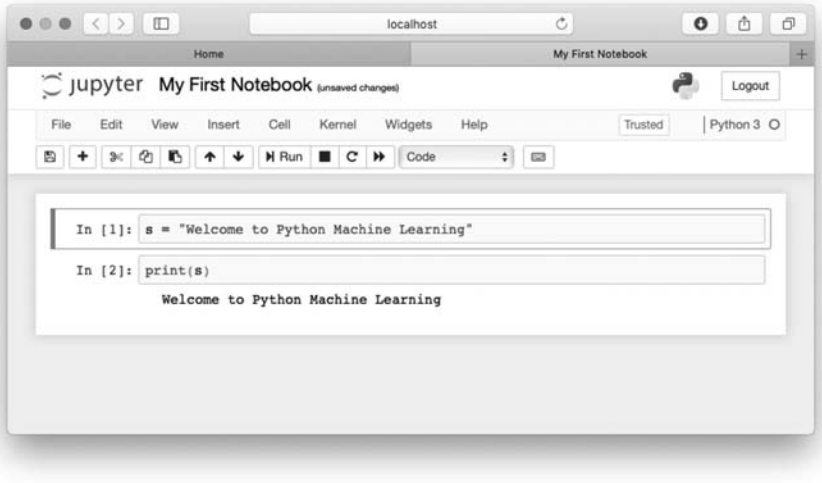

图 1.14　单元格旁边显示的数字指示了它的运行顺序

可以看到，以前在另一个单元格中执行的代码在执行当前单元格时在内存中保留了其值。但在执行不同顺序的单元格时需要小心。考虑图 1.15 中的示例。这里有三个单元格。在第一个单元格中，初始化字符串的值；在第二个单元格中打印其值；在第三个单元格中，将 s 的值改为另一个字符串。

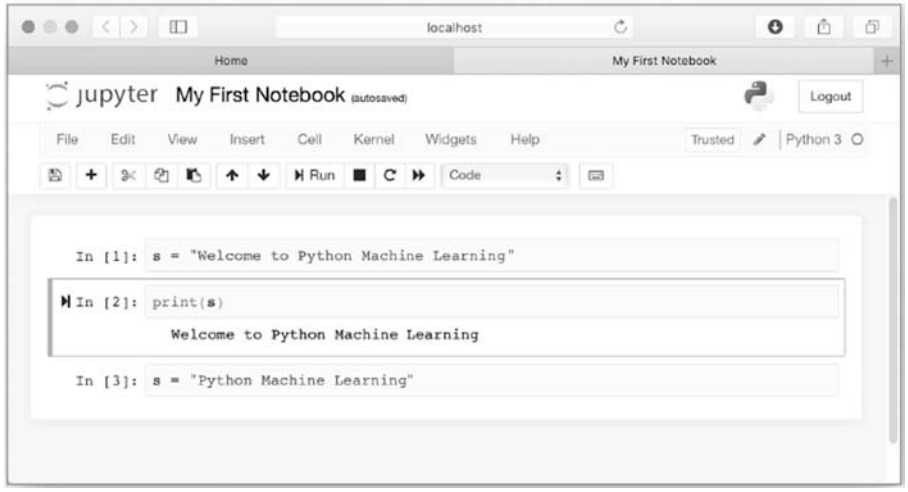

图 1.15　带有三个单元格的笔记本

通常，在测试代码的过程中，可能会在一个单元格中进行修改，然后回到前面的单元格重新测试代码，这是十分常见的。在本例中，假设返回并重新运行第二个单元格，现在将输出新值(见图 1.16)。你可能预期会看到字符串 Welcome to Python Machine Learning，但由于第二个单元格是在第三个单元格之后重新运行的，因此值为字符串 Python Machine Learning。

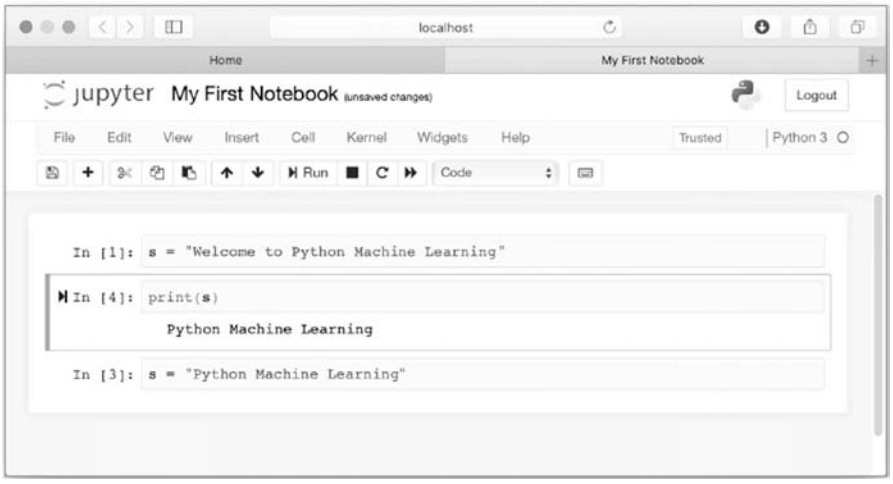

图 1.16　以非线性顺序执行单元格

为从第一个单元格中重启执行，需要重启内核，或选择 Cell | Run All。

7. 重启内核

因为可在笔记本上以任何顺序运行任何单元，一段时间后，事情可能变得有点混乱。此时可能希望重启执行，并重新开始。这就需要重启内核(见图 1.17)。

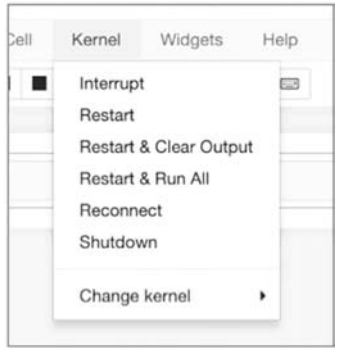

图 1.17　重启内核

提示：

当代码进入无限循环时，就需要重启内核。重启内核有两种常见的场景。

Restart & Clear Output：重启内核并清除所有输出。现在可按喜欢的任何顺序运行任何单元格。

Restart & Run All：重启内核并从第一个单元格运行到最后一个单元格。如果对代码感到满意，并希望对其进行完整的测试，就可以使用这个选项。

8. 导出笔记本

在 Jupyter Notebook 中完成测试后，现在可将笔记本中的代码导出到 Python 文件中。为此，选择 File | Download as | python(.py)，见图 1.18。

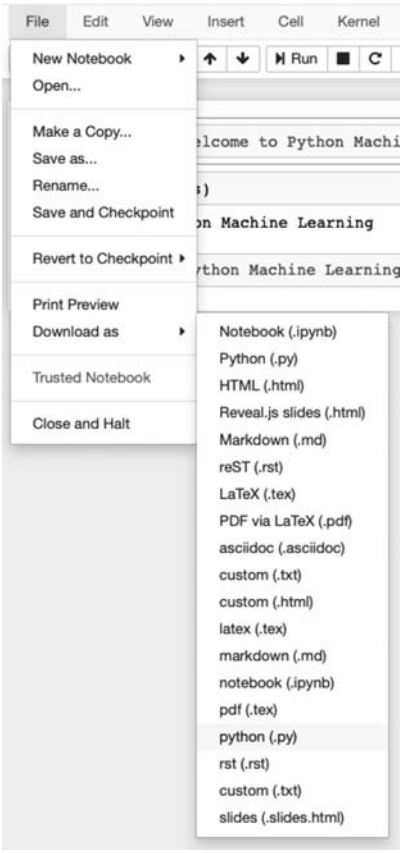

图 1.18　将笔记本导出到 Python 文件

在计算机中会下载与笔记本同名的文件，但现在扩展名为.py。

提示：

确保选择了 python(.py)选项，而不是 Python(.py)选项。后一个选项保存的文件带有.html 扩展名。

9. 获得帮助

很容易在 Jupyter Notebook 上得到帮助。要获得 Python 中函数的帮助，请将光标放在函数名上并按 Shift+Tab 键。这将显示一个名为“工具提示”的弹出窗口(见图 1.19)。

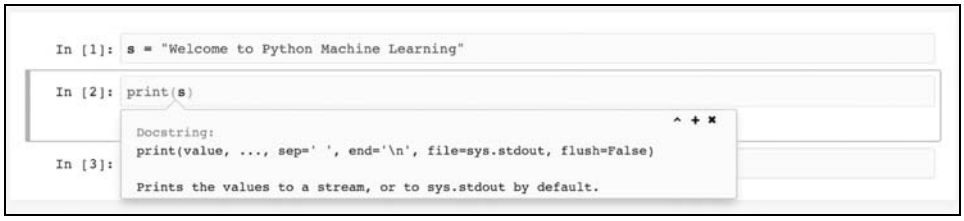

图 1.19　工具提示显示帮助信息

要展开工具提示(见图 1.20)，请单击工具提示右上角的按钮。还可在按下 Shift+Tab+Tab 键时获得工具提示的扩展版本。

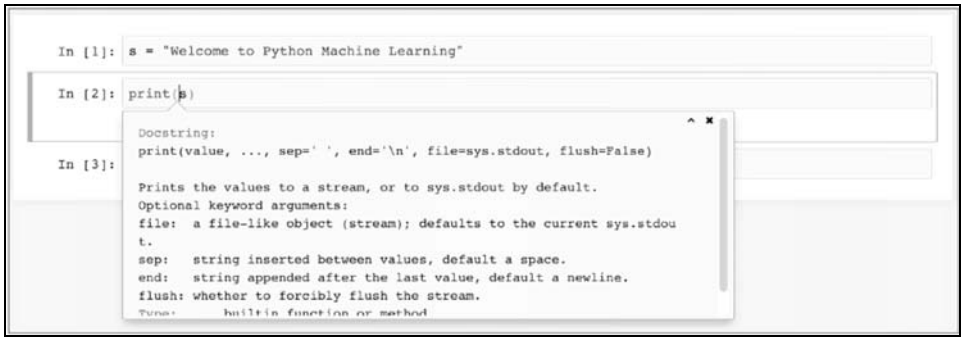

图 1.20　展开工具提示，以显示更多细节

1.3　本章小结

本章介绍了机器学习以及它可解决的问题类型。还了解了监督学习和非监督学习的主要区别。对于 Python 编程新手，强烈建议安装 Anaconda，它将提供完成本书示例需要的所有库和包。我知道读者渴望开始学习，所以让我们进入第 2 章！