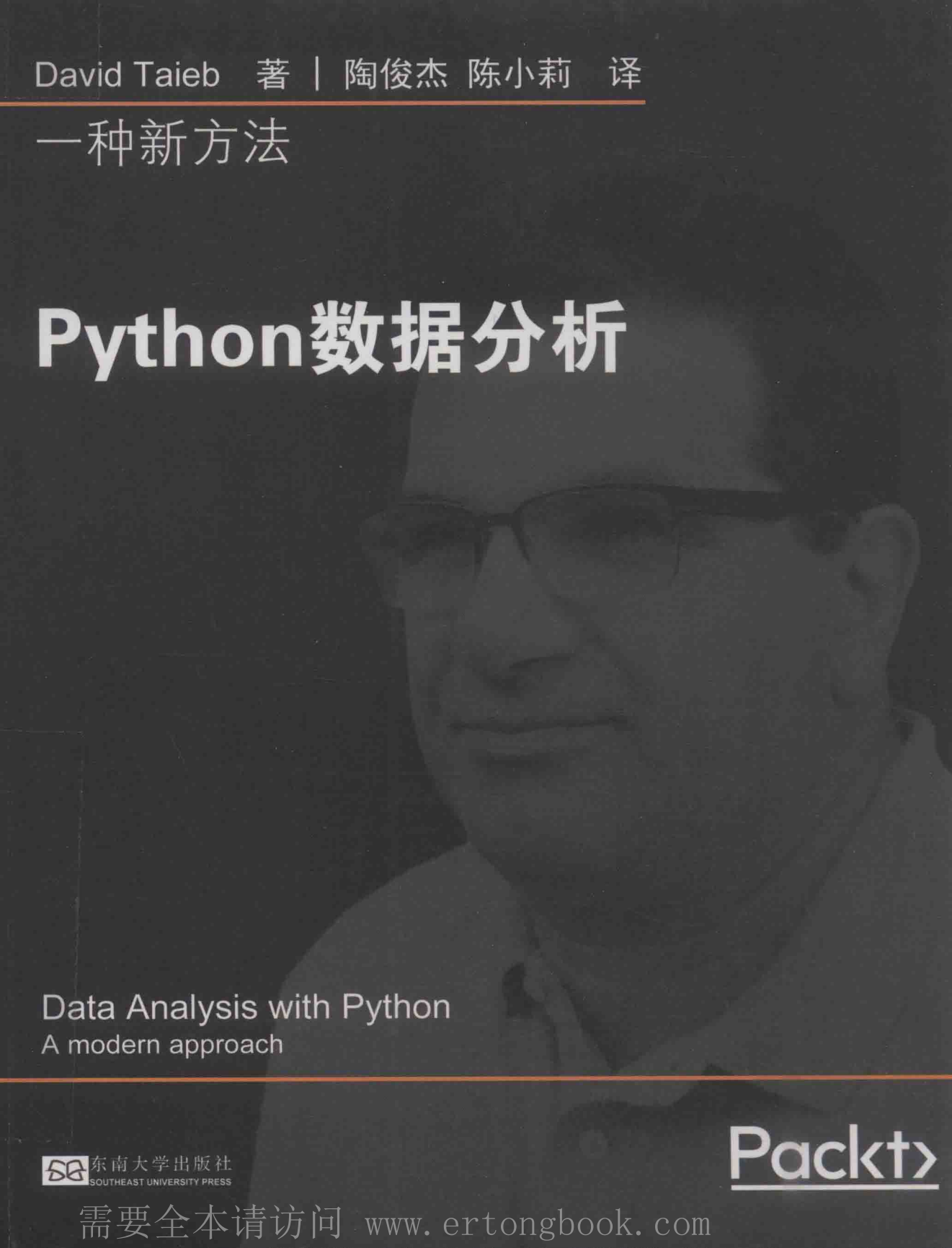
David Taieb 著 | 陶俊杰 陈小莉 译

一种新方法

Python数据分析



Data Analysis with Python
A modern approach

东南大学出版社
SOUTHEAST UNIVERSITY PRESS

Packt

需要全本请访问 www.ertongbook.com

Python 数据分析

David Taieb 著
陶俊杰 陈小莉 译



东南大学出版社
SOUTHEAST UNIVERSITY PRESS
·南京·

图书在版编目(CIP)数据

Python数据分析 /(美)大卫·塔伊布(David Taieb)著;陶俊杰,陈小莉译. —南京:东南大学出版社,2020.6

书名原文:Data Analysis with Python

ISBN 978-7-5641-8884-9

Ⅰ.①P… Ⅱ.①大… ②陶… ③陈… 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062176号

图字:10-2019-189号

© 2018 by PACKT Publishing Ltd.

Simplified Chinese Edition, jointly published by PACKT Publishing Ltd and Southeast University Press, 2020. Authorized translation of the English edition, 9781789950069, 2018 PACKT Publishing Ltd, the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由PACKT Publishing Ltd出版2018。

简体中文版由东南大学出版社出版2020。此简体中文版的出版和销售得到出版权和销售权的所有者——PACKT Publishing Ltd的许可。

版权所有,未得书面许可,本书的任何部分和全部不得以任何形式重制。

Python数据分析

Python Shuju Fenxi

出版发行:东南大学出版社

地　　址:南京四牌楼2号　　邮编:210096

出 版 人:江建中

网　　址:http://www.seupress.com

电子邮件:press@seupress.com

印　　刷:常州市武进第三印刷有限公司

开　　本:787毫米×980毫米　16开本

印　　张:27.25

字　　数:534千字

版　　次:2020年6月第1版

印　　次:2020年6月第1次印刷

书　　号:ISBN 978-7-5641-8884-9

定　　价:94.00元

本社图书若有印装质量问题,请直接与营销部联系。电话(传真):025-83791830

前　言

“现如今无论在什么行业，开发者都是最重要、最有价值的支柱。”
——斯蒂芬·奥格雷迪(Stephen O'Grady)，《新国王的缔造者》(*The New Kingma-kers*)作者

首先，让我感谢你并向你祝贺，我的读者，因为你决定投入一些宝贵的时间来阅读这本书。在接下来的章节里，我将从开发人员的角度带你踏上焕然一新的数据科学旅程，一起探索本书的主题，即数据科学是一项团队运动，如果想要获得成功，开发人员必须在不久的将来扮演更重要的角色，与数据科学家更好地协作。但是在此之前，为了让数据科学对任意背景和行业的人更具包容性，我们首先需要通过让数据变得简单易行来使其民主化——这就是本书的主旨。

我为什么要写这本书？

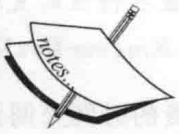
就像我将在第 1 章“编程和数据科学——一个新的工具集”中更详细地自我介绍的那样，我首先是一名具有 20 多年开发经验的开发人员，具有构建不同功能软件组件的经验，包括前端、后端、中间件等。回忆起那段时间，我一直觉得正确的算法在我的脑海中始终占据第一位，而数据总是别人的问题。我很少需要去分析它，也很少从中获得真知灼见。顶多是我设计了正确的数据结构来加载它，这样可以使我的算法更高效地运行，代码变得更优雅且可重用。

但是，随着人工智能和数据科学革命的蓬勃发展，我觉得像我这样的开发人员显然应该参与其中，于是在 2011 年，我抓住机会成为了 IBM Watson 核心平台 UI 和工具(IBM Watson Core Platform UI&Tooling)的首席架构师。当然，我也不能假装一跃成为机器学习或 NLP 方面的专家，实力的确大相径庭，而在实践中学习也不可能代替正式学术背景。

不过，我想在这本书中表达的一个重要观点是：只要掌握了正确的工具和方法，一个拥有足够数学基础的人(我只是说理解高中水平的微积分概念足矣)也可以很快地成为该领域的最佳实践者。成功的关键因素之一是尽可能地简化构建数据管道的不同步骤，

从获取、加载和清理数据，到可视化和探索数据，一直到构建和部署机器学习模型。

为了持续推进“使数据变得简单，让数据科学家之外的社区也能访问数据”这一理念，2015 年我开始在 IBM Watson 数据平台团队中担当领导角色，任务是壮大数据开发者社区，重点关注数据科学教育和活动。在这段时间里，作为主要的开发人员倡导者，我开始公开谈论开发人员和数据科学家在解决复杂数据问题时更好地协作的必要性。



注意：当我参加一些科技研讨会议时，有时会与数据科学家发生矛盾，他们会因为我说数据科学家不是好的软件开发人员而生气。我想在此澄清一个事实，包括你——数据科学家读者在内，情况远非如此。

大多数数据科学家都是优秀的软件开发人员，对计算机科学概念有全面的了解。然而，他们的主要目标是解决复杂的数据问题，这些问题需要快速的迭代实验来尝试新事物，而不是编写优雅的、可重用的组件。

但我并不是纸上谈兵而已，我还想实践出真知，于是开启了一个叫 PixieDust 的开源项目，作为我解决这个重要问题的微薄贡献。随着 PixieDust 项目工作的顺利开展，具体的应用程序示例让功能变得更加明朗，更容易理解，一些开发人员和数据科学家都表示很感兴趣。

在我有机会写一本关于这类主题的书之前，我为此次冒险犹豫了很长时间，这主要有两个原因：

- 我在博客、文章和教程中记录了大量关于我作为一个使用 Jupyter Notebook（Jupyter 笔记本）的数据科学从业者的经验。作为一名各种会议上的演讲者和研讨会主持人，我也具有充分的经验。一个很好的例子是，我在 2017 年伦敦 ODSC 上发表的主题演讲，题目是“数据科学的未来：更少的权力游戏，更多的联盟”（*Less Game of Thrones, More Alliances*，https://odsc.com/training/portfolio/future-data-science-less-game-thrones-alliances）。然而，我从来没有写过一本书，也不知道这将是一个多大的负担，即使我那些写过书的朋友们警告了我许多次。

- 我希望这本书是具有包容性的，能够满足开发人员、数据科学家和业务线用户的共同需求，我一直在努力寻找合适的内容和描述方式来实现这一目标。

最后，开始这场冒险的决定来得很容易。在 PixieDust 项目工作了两年之后，我觉得我们已经取得了很大的进步，我们的创新非常有趣，在开源社区的口碑也不错。写一本书正好可以作为补充，更好地帮助开发人员参与数据科学方面的宣传工作。

顺便提一下，对于那些正在考虑写书并且有类似担忧的读者来说，我只能对第一本

需要全本请访问 www.ertongbook.com

书给出一个很大的建议是："加油，去写吧。"当然，这是一个很郑重的承诺，需要做出大量的牺牲，但前提是你有一个好故事和充实的内容可以讲，这份努力值得付出。

本书目标读者

这本书将服务于初露头角的数据科学家和开发人员，让他们有兴趣发展自己的技能，或者任何希望成为一名专业数据科学家的人。根据 PixieDust 创建者的介绍，这本书也将成为已经取得成就的数据科学家们的一个不错的办公伙伴。

无论个人对数据科学的兴趣如何，清晰、易于阅读的文字和真实的生活场景都适合那些对该领域感兴趣的人，因为他们可以使用运行在 Jupyter Notebook 中的 Python 代码。

要生成一个功能正常的 PixieDust 仪表盘，只需要少量的 HTML 和 CSS。流畅的数据解释和可视化也是必要的，因为这本书针对的是数据专业人员，包括业务分析师和一般数据分析师。后面章节中还有很多内容进行介绍。

本书内容介绍

这本书包含两个篇幅大致相等的逻辑部分。在上半部分，我阐述了本书的主题，即需要消除数据科学与工程之间的壁垒，包括我所建议的 Jupyter＋PixieDust 解决方案的深入细节；在下半部分，我致力于将我们在上半部分所学到的知识应用于 4 个行业案例。

第 1 章　编程和数据科学——一个新的工具集

我尝试根据自己的经验提供一个数据科学定义，构建一个对 Twitter 帖子进行情感分析的数据管道。我要辩护的核心观点是：数据科学是一项团队运动，而数据科学和工程团队之间往往存在隔阂，这会造成不必要的摩擦、效率低下，并最终导致团队未能充分发挥其潜力。我认为数据科学将长期存在，并最终成为今天所说的计算机科学的一个组成部分（我希望有一天会出现新的术语，例如更好地捕捉这种二元性的"计算机数据科学"）。

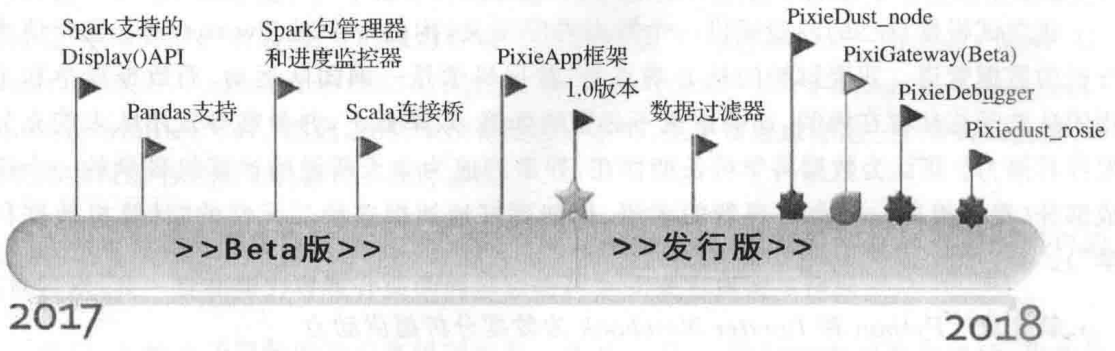
第 2 章　Python 和 Jupyter Notebook 为数据分析提供动力

我将深入阐述流行的数据科学工具，比如 Python 及其专门用于数据科学的开源库生态系统，当然还有 Jupyter Notebook。我会解释为什么我认为 Jupyter Notebook 将成为未来几年的大赢家。我会从简单的 `display()` 方法开始介绍 PixieDust 开源库的功能，该方法允许用户构建引人注目的图表，从而可视化地浏览交互式用户界面中的数据。

用户通过这个 API 可以对不同的绘图引擎进行选择，如 Matplotlib、Bokeh、Seaborn 和 Mapbox。display()功能是 PixieDust MVP(Minimum Viable Product，最小可行产品)阶段的唯一特性，但随着与许多数据科学从业者之间交流的不断增多，我很快就向 PixieDust 工具箱添加了新特性：

- **sampleData()**：一个简单的 API，用于轻松地将数据加载到 Pandas 和 Apache Spark DataFrames。
- **wrangle_data()**：一个简单的 API，用于清理和整理数据集。此功能包括使用正则表达式将列分解为新列，从非结构化文本中提取内容。wrangle_data() API 还可以根据预定义的模式提出建议。
- **包管理器(PackageManager)**：允许用户在 Python Notebook 中安装第三方 Apache Spark 包。
- **Scala 连接桥 (Scala Bridge)**：使用户能够在 Python Notebook 中运行 Scala 代码。在 Python 中定义的变量可以在 Scala 中被访问，反之亦然。
- **Spark 作业进度监控器(Spark Job Progress Monitor)**：允许你使用实时进度条跟踪 Spark 作业的状态，该进度条直接显示在正在执行的代码的输出单元格中。
- **PixieApp**：提供了一个以 HTML/CSS 为中心的编程模型，允许开发人员构建复杂的仪表盘来操作笔记本中内置的分析功能。PixieApp 可以直接在 Jupyter Notebook 中运行，也可以使用 PixieGateway 微服务作为分析 Web 应用程序来部署。PixieGateway 是 PixieDust 的开源配套项目。

下图总结了 PixieDust 的开发历程，包括最近添加的 PixieGateway 和 PixieDebugger，PixieDebugger 是 Jupyter Notebook 的第一个可视化 Python 调试器。



PixieDust 开发历程

本章要传达的一个关键信息是，PixieDust 首先是一个开源项目，依赖开发人员社区的贡献而得以生存和发展。与无数开源项目的情况一样，随着时间的推移，我们可以期

待 PixieDust 被添加更多突破性的特性。

第 3 章　使用 Python 库加速数据分析

我将会引导读者深入了解 PixieApp 编程模型，并用一个分析 GitHub 数据的示例应用程序演示其中的每个概念。首先，我会对 PixieApp 的结构进行概述，包括它的生命周期和路由概念下的执行流。然后，我会详细介绍开发人员如何使用常规 HTML 和 CSS 代码段构建仪表盘的用户界面，与分析无缝交互，并利用 PixieDust 的 `display()` API 添加复杂的图表。

PixieApp 编程模型是消除数据科学和工程之间壁垒的工具策略的基石，因为它简化了数据分析的操作过程，从而增加了数据科学家和开发人员之间的协作，缩短了应用程序的发布时间。

第 4 章　用 PixieApp 工具发布数据分析结果

我将探讨 PixieGateway 微服务，它使开发人员能够将 PixieApp 发布为数据分析 Web 应用程序。首先，我将演示如何在本地和云计算平台上，通过 Kubernetes 容器快速部署 PixieGateway 微服务实例。然后，我将介绍 PixieGateway 管理控制台功能，包括各种配置文件以及如何实时监视已部署的 PixieApp 实例和相关联的后端 Python 内核。我还提供了 PixieGateway 的图表共享功能，允许用户将用 PixieDust `display()` API 创建的图表转换为团队中任何人都可以访问的网页。

PixieGateway 是一项具有开拓性的创新，有可能大大加快数据分析的速度——这正是当前行业所迫切需要的——以充分展示数据科学的前景。它是市场上已经存在的同类产品的开源替代方案，比如 R－Studio（`https://shiny.rstudio.com/deploy`）的 Shiny Server 和 Plotly（`https://dash.plot.ly`）的 Dash。

第 5 章　Python 和 PixieDust 最佳实践与高级概念

我将深入 PixieDust 工具箱，详细阐述 PixieApp 编程模型的高级概念：

- **CaptureOutput 修饰器**：默认情况下，PixieApp 路由要求开发人员提供一个 HTML 片段，并将其注入应用程序的用户界面中。当我们想调用一个不了解 PixieApp 体系结构的第三方 Python 库并直接生成输出到 Notebook 时就容易出问题。`@captureOutput` 通过自动重定向由第三方 Python 库生成的内容并将其封装到适当的 HTML 片段中来解决这个问题。

- **利用 Python 类继承提高模块性和代码重用**：将 PixieApp 代码分解为逻辑类，这些逻辑类可以使用 Python 类继承功能组合在一起。我还会演示如何使用 `pd_app` 自定义属性来调用外部 PixieApp。

• **PixieDust 对流式数据的支持**：演示如何用 PixieDust `display()` 和 PixieApp 处理流式数据。

• **使用 PixieApp 事件实现仪表盘数据挖掘**：提供一种机制，允许 PixieApp 组件发布和订阅用户与 UI 交互时生成的事件（例如图表和按钮）。

• **为 PixieDust display()API 构建自定义显示渲染器**：浏览扩展 PixieDust 菜单的简单渲染器的代码。通过渲染器展示一个呈现用户所选数据的自定义 HTML 表。

• **调试技术**：介绍 PixieDust 提供的各种调试技术，包括可视化 Python 调试器 PixieDebugger 和用于显示 Python 日志消息的 `%% PixiedustLog` 魔法函数。

• **运行 Node.js 代码的能力**：我们还会介绍 `pixiedust_node` 的扩展功能，它管理负责从 Python Notebook 中直接执行任意 Node.js 脚本的 Node.js 进程的生命周期。

感谢开源模型具备的透明开发过程以及不断增长的用户社区所提供的有价值反馈，随着时间的推移，我们能够对这些高级特性进行优先排序并实现。我想表达的关键一点是，遵循具有适当许可证的开源模型（PixieDust 使用 `https://www.apache.org/licenses/LICENSE-2.0` 提供的 Apache2.0 许可证）的确非常有效。它既可以帮助我们扩展用户社区，反过来又为我们提供必要的反馈，从而让我们可以优先考虑高价值的新特性，并且在某些情况下使用用户通过 GitHub pull 请求贡献的代码。

第 6 章　分析案例：人工智能与 TensorFlow 图像识别

我将介绍 4 个行业案例，首先介绍机器学习的概念，接着介绍机器学习的一个子领域——深度学习，以及使构建神经网络模型变得更容易的 TensorFlow 框架。然后，我将构建一个图像识别示例应用程序，包含相关联的 PixieApp，一共分为 4 个部分：

• *第 1 部分*：利用预训练 ImageNet 模型建立图像识别 TensorFlow 模型。使用 TensorFlow for poets 教程（`https://github.com/googlecodelabs/tensorflow-for-poets-2`），我将演示如何建立分析器来加载神经网络模型并为其评分。

• *第 2 部分*：创建一个 PixieApp，用于操作第 1 部分中创建的分析结果。这个 PixieApp 从用户提供的网页 URL 中抓取图像，根据 TensorFlow 模型对它们进行评分，然后以图形方式显示结果。

• *第 3 部分*：我将演示如何将 TensorBoard 图形可视化组件直接集成到 Notebook 中，提供调试神经网络模型的能力。

• *第 4 部分*：我将演示如何使用自定义训练数据重新训练模型并更新 PixieApp 以显示来自两个模型的结果。

我决定用 TensorFlow 的深度学习图像识别来开始一系列示例应用程序，因为这是一个日益流行的重要用例，演示我们如何在同一 Notebook 中构建模型并将其部署到应

用程序中，这是一个消除数据科学与工程之间壁垒的有力陈述。

第 7 章　分析案例：自然语言处理、大数据与 Twitter 情感分析

我将实现 Twitter 数据分析规模的自然语言处理。在本章中，我将展示如何使用 IBM Watson 自然语言理解云服务(Natural Language Understanding Cloud Based Service)进行推文(tweet)情感分析。这一点非常重要，因为它提醒读者，重复利用成熟的托管服务而不是从零开始构建功能有时可能是一个更有吸引力的选择。

我首先介绍了 Apache Spark 并行计算框架，然后分 4 个部分进行了应用构建：

- **第 1 部分**：利用 Spark Structured Streaming 获取 Twitter 数据；
- **第 2 部分**：利用从文本中提取的情感和最相关的实体来丰富数据；
- **第 3 部分**：通过创建实时仪表盘 PixieApp 来操作分析；
- **第 4 部分**：这是一个可选部分，使用 Apache Kafka 和 IBM Streaming Designer 托管服务重新实现应用程序，以演示如何添加更大的可伸缩性。

我认为部分读者，特别是那些不熟悉 Apache Spark 的读者，会喜欢这一章，因为它比上一章更容易理解。关键的问题是，如何构建一个与连接到 Spark 集群(cluster)的 Jupyter Notebook 相适应的分析工具。

第 8 章　分析案例：预测——金融时间序列分析与预测

我要介绍的时间序列分析是数据科学中一个非常重要的领域，在行业中有许多实践应用。本章一开始我会介绍 NumPy 库，它是许多其他库的基础，比如 pandas 和 SciPy。然后，我会构建分析由历史股票数据组成的时间序列的示例应用程序，一共分为两部分：

- **第 1 部分**：对包括各种图表如自相关函数(ACF)和偏自相关函数(PACF)的时间序列进行统计分析；
- **第 2 部分**：使用 `statsmodels` Python 库建立一个基于 ARIMA 算法的预测模型。

时间序列分析是数据科学的一个重要领域，我认为它被低估了。我在写这一章的时候学到了很多知识。当然我也希望读者会喜欢它、阅读它，以激发兴趣，进而了解更多关于时间序列的伟大主题。如果你真愿意做的话，也希望你能在下一次学习时间序列分析的时候尝试一下 Jupyter 和 PixieDust。

第 9 章　分析案例：图形算法——美国国内航班数据分析

我利用图形模型实现这一章的行业案例。我选择了一个分析航班延误数据的示例应用程序，因为数据很容易获得，而且它非常适合使用图形算法(说实话，我之所以选择它，

其实是因为我已经使用 Apache Spark MLlib 编写了一个类似的应用程序，根据天气数据预测航班延误情况：https:// developer.ibm.com/clouddataservices/2016/08/04/predict-flight-delays- with-apache-spark-mllib-flightstats-and-weather-data)。

首先，我将介绍图形的概念和相关的图形算法，包括几个最流行的图形算法，如广度优先搜索和深度优先搜索。然后，我将介绍用于构建示例应用程序的 networkx Python 库。

示例应用程序由 4 部分组成：

- 第 1 部分：展示如何将美国国内航班数据加载到图中；
- 第 2 部分：创建 USFlightsAnalysis PixieApp，允许用户选择始发机场和目的地机场，然后根据所选的中心性显示两个机场之间最短路径的 Mapbox 地图；
- 第 3 部分：在 PixieApp 中添加数据探索功能，该功能包括从选定的始发机场起飞的每条航线的各种统计信息；
- 第 4 部分：使用第 8 章中学习的技术构建用于预测航班延误情况的 ARIMA 模型。

图论也是数据科学的另一个重要且不断发展的领域，本章对该系列知识进行了细致总结，我希望能够提供一组多样化且具有代表性的行业案例。对于在大数据中使用图形算法特别感兴趣的读者，我建议查看 Apache Spark Graphx(https://spark.apache.org/graphx)，它使用非常灵活的 API 实现了许多图形算法。

第 10 章　数据分析的未来与拓展技能的途径

我在本书的结尾部分给出了一个简短的总结，并解释了我对德鲁·康威的维恩图的看法。之后，我将介绍人工智能和数据科学的未来，以及企业如何为人工智能和数据科学革命做好准备。此外，我列出了一些有价值的参考资料，以供读者进一步学习。

附录　PixieApp 快速参考

这是一份开发人员快速参考指南，它提供了所有 PixieApp 属性的摘要，通过适当示例介绍各种注解(annotation)、自定义 HTML 属性和方法。

关于导言的内容已经足够多了，下面让我们从第 1 章“编程和数据科学——一个新的工具集”开始我们的旅程吧。

如何充分利用这本书

- 遵循这个示例所需的大多数软件都是开源的，因此可以免费下载。本书提供了完

需要全本请访问 www.ertongbook.com

整的说明，从安装 Anaconda 开始，包括 Jupyter Notebook 服务器。

• 在第 7 章中，示例应用程序需要使用 IBM Watson 云服务，包括 NLU 和 Streams Designer。这些服务附带了一个免费层级(free tier)计划，足以满足该示例的需要。

下载示例代码文件

你可以用你的账户从 http://www.packtpub.com 下载本书的示例代码文件。如果你是从其他途径购买的本书，可以访问 http://www.packtpub.com/support 并注册，我们将通过电子邮件把文件发送给你。

你可以通过以下步骤下载示例代码文件：

1. 登录或注册 http://www.packtpub.com。
2. 选择 **SUPPORT**(客户支持)标签。
3. 单击 **Code Downloads & Errata**(代码下载与勘误)按钮。
4. 在 **Search**(搜索框)中输入书名并根据屏幕提示操作。

下载文件后，请确保用以下软件的最新版来解压文件：

• Wondows：WinRAR / 7－Zip for Windows
• Mac：Zipeg / iZip / UnRarX for Mac
• Linux：7－Zip /PeaZip for Linux

本书的代码包也可以在 GitHub 上获得，网址是 https://github.com/PacktPublishing/Data- Analysis- with- Python。另外，我们在 https://github.com/PacktPublishing 上还有其他书的代码包和视频，请有需要的读者自行下载。

下载本书中的彩图

我们也为你提供了一份 PDF 文件，里面包含了书中的截屏和图表等彩图，下载地址为 http://www.packtpub.com/sites/ default/files/downloads/DataAnalysiswithPython_ColorImages.pdf。彩图能帮助你更好地理解输出的变化。

排版约定

本书使用了许多排版约定。

codeintext：表示文本中的代码、数据库表名、文件夹名、文件名、文件扩展名、路径名、伪 URL、用户输入和 Twitter 账号。例如："可以使用{% if ...%}...{% elif ...%}...{%else%}...{%endif%}标识符有条件地输出文本。"

代码块设置如下：

```
import pandas
data_url = "https://data.cityofnewyork.us/api/views/e98g- f8hy/rows. csv?
accessType= DOWNLOAD"
building_df = pandas.read_csv(data_url)
building_df
```

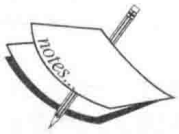
当我们希望提请你注意代码块的特定部分时，相关行或项以黑体设置：

```
import pandas
data_url = "https://data.cityofnewyork.us/api/views/e98g- f8hy/rows. csv?
accessType= DOWNLOAD"
building_df = pandas.read_csv(data_url)
building_df
```

任何命令行的输入或输出形式如下：

```
jupyter notebook - - generate- config
```

Bold：表示一个新的术语、一个重要的词或屏幕上看到的词，例如菜单或对话框中的词，也会出现在文本中。例如："下一步是创建一个获取用户值并返回结果的新路由。此路由将由 **Submit Query** 按钮调用。"



警告或重要注释这样显示。



提示和技巧这样显示。

联系我们

欢迎读者提供反馈信息。

一般反馈：给 feedback@ packtpub.com 发邮件，并在邮件主题中加入这本书的书名。如果你对本书内容存有疑问，不管是哪个方面的，都可以通过 questions@ packtpub.com 联系我们。

勘误：虽然我们已尽力确保本书内容正确，但出错仍旧在所难免。如果你在书中发现错误，希望能告知我们，我们将不胜感激。请访问 http://www.packtpub.com/submit- errata，选择本书，单击 **Errata Submission Form**（勘误提交表单）链接，并输入详细说明。

盗版：如果你发现我们的作品在互联网上被非法复制，不管是什么形式的，都请立即为我们提供相关网址或网站名称，我们将不胜感激。请把可疑盗版材料的链接发到 copyright@packtpub.com。

如果你有兴趣成为一名作者：如果你有一个擅长的主题，并且有兴趣写一本书或者为一本书撰写部分内容，请访问 http://authors.packtpub.com。

评论

欢迎留下评论。一旦你阅读并使用了这本书，为什么不在你购买它的网站上留下一个评论呢？潜在的读者看到你无偏见的意见，会据此做出购买决定，而我们可以感受到你对我们产品的想法，书籍作者同样可以看到你对他们的书的反馈。非常感谢！

关于 Packt 出版社的更多信息，请访问 packtpub.com。

目　录

1　编程和数据科学——一个新的工具集 …… 1
什么是数据科学 …… 1
数据科学会长期存在吗？ …… 2
为什么数据科学正在兴起？ …… 3
数据科学与开发人员有什么关系？ …… 4
将这些概念付诸实践 …… 5
深入研究一个具体的示例 …… 6
数据管道蓝图 …… 7
数据科学家应该具备什么技能？ …… 9
IBM Watson Deep QA …… 10
回到 Twitter 带＃标签的情感分析项目 …… 13
从构建第一条企业级数据管道中汲取的经验教训 …… 15
数据科学策略 …… 16
Jupyter Notebook 是我们的战略核心 …… 18
为什么 Notebook 这么流行？ …… 19
本章小结 …… 21
2　Python 和 Jupyter Notebook 为数据分析提供动力 …… 23
为什么选择 Python？ …… 23
PixieDust 入门 …… 27
SampleData——一个用于加载数据的简单 API …… 31
用 pixiedust_rosie 整理数据 …… 36
Display——一个简单的交互式数据可视化 API …… 42
过滤 …… 50
用 PixieApp 消除开发人员和数据科学家之间的壁垒 …… 52
操作化数据科学分析的体系结构 …… 55
本章小结 …… 59

3 使用 Python 库加速数据分析 …… 60
PixieApp 深度剖析 …… 61
路由 …… 62
生成路由请求 …… 65
GitHub 项目跟踪示例程序 …… 67
在表格中显示搜索结果 …… 70
使用 pd_entity 属性调用 PixieDust display()API …… 78
使用 pd_script 调用任意 Python 代码 …… 85
用 pd_refresh 让应用程序更具响应性 …… 90
创建可重用的小部件 …… 93
本章小结 …… 94
4 用 PixieApp 工具发布数据分析结果 …… 95
Kubernetes 概述 …… 96
安装和配置 PixieGateway 服务器 …… 97
PixieGateway 服务器配置 …… 101
PixieGateway 体系结构 …… 105
发布应用程序 …… 108
PixieApp URL 中的编码状态 …… 112
通过将图表发布为网页来共享它们 …… 114
PixieGateway 管理控制台 …… 118
Python 控制台 …… 120
显示 PixieApp 的预热和运行代码 …… 121
本章小结 …… 121
5 Python 和 PixieDust 最佳实践与高级概念 …… 122
使用@captureOutput 装饰器集成第三方 Python 库的输出 …… 123
使用@captureOutput 创建词云图像 …… 123
增加模块化和代码重用 …… 126
使用 pd_widget 创建小部件 …… 128
PixieDust 中的流式数据支持 …… 130
向 PixieApp 添加流处理功能 …… 134
添加带有 PixieApp 事件的仪表盘下钻功能 …… 136

扩展 PixieDust 可视化 …… 141
调试 …… 148
使用 pdb 调试 Jupyter Notebook …… 148
使用 PixieDebugger 进行可视化调试 …… 152
使用 PixieDebugger 调试 PixieApp 路由 …… 154
使用 PixieDust 日志记录排除问题 …… 157
客户端调试 …… 159
在 Python 笔记本中运行 Node.js …… 161
本章小结 …… 165
6 分析案例:人工智能与 TensorFlow 图像识别 …… 167
什么是机器学习? …… 168
什么是深度学习? …… 169
开始使用 TensorFlow …… 172
用 DNNClassifier 进行简单分类 …… 176
图像识别示例应用程序 …… 187
第 1 部分——加载预训练的 MobileNet 模型 …… 188
第 2 部分——为我们的图像识别示例应用程序创建一个 PixieApp …… 196
第 3 部分——集成 TensorBoard 图形可视化 …… 200
第 4 部分——使用自定义训练数据重新训练模型 …… 206
本章小结 …… 217
7 分析案例:自然语言处理、大数据与 Twitter 情感分析 …… 218
ApacheSpark 入门 …… 218
Apache Spark 体系结构 …… 219
将 Notebook 配置为使用 Spark …… 220
“Twitter 情感分析”应用程序 …… 222
第 1 部分——以 Spark Structured Streaming 获取数据 …… 223
数据管道的体系结构图 …… 224
Twitter 的身份验证 …… 224
创建 Twitter 流 …… 226
创建 Spark Streaming DataFrame …… 229
创建和运行结构化查询 …… 232

需要全本请访问 www.ertongbook.com