O'REILLY®

TURING 图灵程序设计丛书

全彩印刷

R数据科学

R for Data Science

摒弃其他R语言工具书从头到尾讲统计的陋习
从实用的R包出发，带你重新认识R和数据科学

[新西兰] 哈德利·威克姆 [美] 加勒特·格罗勒芒德 著
陈光欣 译

中国工信出版集团 人民邮电出版社 POSTS & TELECOM PRESS

TURING 图灵程序设计丛书

R数据科学

R for Data Science

[新西兰] 哈德利·威克姆 [美] 加勒特·格罗勒芒德 著
陈光欣 译

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

O'Reilly Media, Inc.授权人民邮电出版社出版

人民邮电出版社
北 京

图书在版编目（CIP）数据

R数据科学 / (新西兰) 哈德利·威克姆 (Hadley Wickham) , (美) 加勒特·格罗勒芒德 (Garrett Grolemund) 著 ; 陈光欣译. -- 北京 : 人民邮电出版社, 2018.8
（图灵程序设计丛书）
ISBN 978-7-115-48639-4

Ⅰ. ①R… Ⅱ. ①哈… ②加… ③陈… Ⅲ. ①程序语言－程序设计 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2018)第124271号

内 容 提 要

本书的目标是教会读者使用最重要的数据科学工具，从而为实施数据科学奠定坚实的基础。读完本书后，你将掌握R语言的精华，并能够熟练使用多种工具来解决各种数据科学难题。每一章都按照这样的顺序组织内容：先给出一些引人入胜的示例，以便你可以整体了解这一章的内容，然后再深入细节。本书的每一节都配有习题，以帮助你实践所学到的知识。

本书适合R数据科学家阅读。

◆ 著 [新西兰] 哈德利·威克姆
[美] 加勒特·格罗勒芒德
译 陈光欣
责任编辑 朱 巍
责任印制 周昇亮

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 http://www.ptpress.com.cn
河北画中画印刷科技有限公司印刷

◆ 开本：800×1000 1/16
印张：23
字数：544千字 2018年8月第1版
印数：1－5 000册 2018年8月河北第1次印刷
著作权合同登记号 图字：01-2018-3442号

定价：139.00元

读者服务热线：(010)51095186转600 印装质量热线：(010)81055316
反盗版热线：(010)81055315
广告经营许可证：京东工商广登字 20170147 号

站在巨人的肩上
Standing on Shoulders of Giants

iTuring.cn

版权声明

O'Reilly Media, Inc.介绍

O'Reilly Media 通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自 1978 年开始，O'Reilly 一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly 的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly 为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了 *Make* 杂志，从而成为 DIY 革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly 的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly 现在还将先锋专家的知识传递给普通的计算机用户。无论是通过图书出版、在线服务或者面授课程，每一项 O'Reilly 的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

业界评论

“O'Reilly Radar 博客有口皆碑。”

——*Wired*

“O'Reilly 凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——*Business 2.0*

“O'Reilly Conference 是聚集关键思想领袖的绝对典范。”

——*CRN*

“一本 O'Reilly 的书就代表一个有用、有前途、需要学习的主题。”

——*Irish Times*

“Tim 是位特立独行的商人，他不光放眼于最长远、最广阔的视野，并且切实地按照 Yogi Berra 的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去，Tim 似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

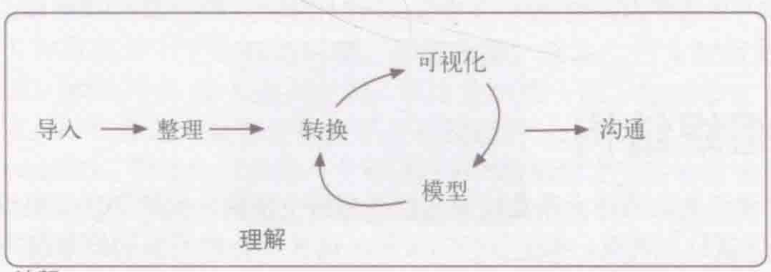
——*Linux Journal*

前言

数据科学是一门激动人心的学科，它可以将原始数据转化为认识、见解和知识。本书的目标是帮助你学习使用 R 语言中最重要的数据科学工具。读完本书后，你将掌握 R 语言的精华，并能够熟练使用多种工具来解决各种数据科学难题。

你将学到什么

数据科学是一个极其广阔的领域，仅靠一本书是不可能登堂入室的。本书的目标是教会你使用最重要的数据科学工具。在一个典型的数据科学项目中，需要的工具模型大体如下图所示。



首先，你必须将数据**导入** R。这实际上就是读取保存在文件、数据库或 Web API 中的数据，再加载到 R 的数据框中。如果不能将数据导入 R，那么数据科学就根本无从谈起。

导入数据后，就应该对数据进行**整理**。数据整理就是将数据保存为一致的形式，以满足其所在数据集在语义上的要求。简而言之，如果数据是整洁的，那么每列都是一个变量，每行都是一个观测。整洁的数据非常重要，因为一致的数据结构可以让你将工作重点放在与数据有关的问题上，而不用再费尽心思地将数据转换为各种形式以适应不同的函数。

一旦拥有了整洁的数据，通常下一步就是对数据进行**转换**。数据转换包括选取出感兴趣的观测（如居住在某个城市里的所有人，或者去年的所有数据）、使用现有变量创建新变量（如根据距离和时间计算出速度），以及计算一些摘要统计量（如计数或均值）。数据整理

和数据转换统称为数据处理。

一旦使用需要的变量完成了数据整理，那么生成知识的方式主要有两种：可视化与建模。这两种方式各有利弊，相辅相成。因此，所有实际的数据分析过程都要在这两种方式间多次重复。

可视化本质上是人类活动。良好的可视化会让你发现意料之外的现象，或对数据提出新的问题。你还可以从良好的可视化中意识到自己提出了错误的问题，或者需要收集不同的数据。可视化能够带给你惊喜，但不要期望过高，因为毕竟还是需要人来对其进行解释。

模型是弥补可视化缺点的一种工具。如果已经将问题定义得足够清晰，那么你就可以使用一个模型来回答问题。因为模型本质上是一种数学工具或计算工具，所以它们的扩展性一般非常好。即使扩展性出现问题，购买更多计算机也比雇用更多聪明的人便宜！但是每个模型都有前提假设，而且模型本身不会对自己的前提假设提出疑问，这就意味着模型本质上不能给你带来惊喜。

数据科学的最后一个步骤就是**沟通**。对于任何数据分析项目来说，沟通绝对是一个极其重要的环节。如果不能与他人交流分析结果，那么不管模型和可视化让你对数据理解得多么透彻，这都是没有任何实际意义的。

围绕在这些技能之外的是**编程**。编程是贯穿数据科学项目各个环节的一项技能。数据科学家不一定是编程专家，但掌握更多的编程技能总是有好处的，因为这样你就能够对日常任务进行自动处理，并且非常轻松地解决新的问题。

你将在所有的数据科学项目中用到这些工具，但对于多数项目来说，这些工具还不够。这大致符合 80/20 定律：你可以使用从本书中学到的工具来解决每个项目中 80% 的问题，但你还需要其他工具来解决其余 20% 的问题。我们将在本书中为你提供资源，让你能够学到更多的技能。

本书的组织结构

前面对数据科学工具的描述大致是按照我们在分析中使用它们的顺序来组织的（尽管你肯定会多次使用它们）。然而，根据我们的经验，这并不是学习它们的最佳方式，具体原因如下。

- 从数据导入和数据整理开始学习并不是最佳方式，因为对于导入和整理数据来说，80% 的时间是乏味和无聊的，其余 20% 的时间则是诡异和令人沮丧的。在学习一项新技术时，这绝对是一个糟糕的开始！相反，我们将从数据可视化和数据转换开始，此时的数据已经导入并且是整理完毕的。这样一来，当导入和整理自己的数据时，你就会始终保持高昂的斗志，因为你知道这种痛苦终有回报。
- 有些主题最好结合其他工具来解释。例如，如果你已经了解可视化、数据整理和编程，那么我们认为你会更容易理解模型是如何工作的。
- 编程工具本身不一定很有趣，但它们确实可以帮助你解决更多非常困难的问题。在本书的中间部分，我们会介绍一些编程工具，它们可以与数据科学工具结合起来以解决非常有趣的建模问题。

我们尽量在每一章中使用同一种模式：先给出一些引人入胜的示例，以便你大体了解这一章的内容，然后再深入细节。本书的每一节都配有习题，以帮助你实践所学到的知识。虽然跳过这些习题是个非常有诱惑力的想法，但使用真实问题进行练习绝对是最好的学习方式。

本书未包含的内容

本书并未涉及一些重要主题。我们深信，重要的是将注意力坚定地集中在最基本的内容上，这样你就可以尽快入门并开展实际工作。这也就是说，本书不会涵盖每一个重要主题。

大数据

本书主要讨论那些小规模的、能够驻留在内存中的数据集。这是开始学习数据科学的正确方式，因为只有处理过小数据集，你才能处理大数据集。你从本书中学到的工具可以轻松地处理几百兆字节的数据，处理 1~2 GB 的数据也不会有什么大问题。如果你的日常工作是处理更大的数据（如 10~100 GB），那么你应该更多地学习一下 data.table（https://github.com/Rdatatable/data.table）。本书不会介绍 data.table，因为它的界面太过简洁，几乎没有语言提示，这使得学习起来很困难。但是如果你需要处理大数据，为了获得性能上的回报，多付出一些努力来学习它还是值得的。

如果你的数据比这还大，那么就需要仔细思考一下了，这个大数据问题是否其实就是一个小数据问题。虽然整体数据非常大，但回答特定问题所需要的数据通常较小。你可以找出一个子集、子样本或者摘要数据，该数据既适合在内存中处理，又可以回答你感兴趣的问题。此时的挑战就是如何找到合适的小数据，这通常需要多次迭代。

另外一种可能是，你的大数据问题实际上就是大量的小数据问题。每一个问题都可以在内存中处理，但你有数百万个这样的问题。举例来说，假设你想为数据集中的每个人都拟合一个模型。如果只有 10 人或 100 人，那这是小菜一碟，但如果有 100 万人，情况就完全不同了。好在每个问题都是独立于其他问题的（这种情况有时称为高度并行，embarrassingly parallel），因此你只需要一个可以将不同数据集发送到不同计算机上进行处理的系统（如 Hadoop 或 Spark）即可。如果已经知道如何使用本书中介绍的工具来解决独立子集的问题，那么你就可以学习一下新的工具（比如 sparklyr、rhipe 和 ddr）来解决整个数据集的问题。

Python、Julia以及类似的语言

在本书中，你不会学到有关 Python、Julia 以及其他用于数据科学的编程语言的任何内容。这并不是因为我们认为这些工具不好，它们都很不错！实际上，多数数据科学团队都会使用多种语言，至少会同时使用 R 和 Python。

但是，我们认为最好每次只学习并精通一种工具。如果你潜心研究一种工具，那么会比同时泛泛地学习多个工具掌握得更快。这并不是说你只应该精通一种工具，而是说每次专注于一件事情时，通常你会进步得更快。在整个职业生涯中，你都应该努力学习新事物，但是一定要在充分理解原有知识后，再去学习感兴趣的新知识。

我们认为 R 是你开始数据科学旅程的一个非常好的起点，因为它从根本上说就是一种用来支持数据科学的环境。R 不仅仅是一门编程语言，它还是进行数据科学工作的一种交互式环境。为了支持交互性，R 比多数同类语言灵活得多。虽然会导致一些缺点，但这种灵活性的一大好处是，可以非常容易地为数据科学过程中的某些环节量身定制语法。这些微型语言有助于你从数据科学家的角度来思考问题，还可以在你的大脑和计算机之间建立流畅的交流方式。

非矩形数据

本书仅关注矩形数据。矩形数据是值的集合，集合中的每个值都与一个变量和一个观测相关。很多数据集天然地不符合这种规范，比如图像、声音、树结构和文本。但是矩形数据框架在科技与工业领域是非常普遍的。我们认为它是开始数据科学旅途的一个非常好的起点。

假设验证

数据分析可以分为两类：假设生成和假设验证（有时称为验证性分析）。无须掩饰，本书的重点就在于假设生成，或者说是数据探索。我们将对数据进行深入研究，并结合专业知识生成多种有趣的假设来帮助你对数据的行为方式作出解释。你可以对这些假设进行非正式的评估，凭借自己的怀疑精神从多个方面向数据发起挑战。

假设验证与假设生成是互补的。假设验证比较困难，原因如下。

- 你需要一个精确的数学模型来生成可证伪的预测，这通常需要深厚的统计学修养。
- 为了验证假设，每个观测只能使用一次。一旦使用观测的次数超过了一次，那么就回到了探索性分析。这意味着，若要进行假设验证，你需要“预先注册”（事先拟定好）自己的分析计划，而且看到数据后也不能改变计划。在本书的第四部分中，我们将讨论一些相关的策略，你可以使用它们让假设验证变得更容易。

经常有人认为建模是用来进行假设验证的工具，而可视化是用来进行假设生成的工具。这种简单的二分法是错误的：模型经常用于数据探索；只需稍作处理，可视化也可以用来进行假设验证。核心区别在于你使用每个观测的频率：如果只用一次，那么就是假设验证；如果多于一次，那么就是数据探索。

准备工作

为了最有效地利用本书，我们对你的知识结构做了一些假设。你应该具有一定的数学基础，如果有一些编程经验也会有所帮助。如果从来没有编写过程序，那么你应该学习一下 Garrett 所著的《R 语言入门与实践》[1]，它可以作为本书的有益补充。

为了运行本书中的代码，你需要 4 个工具：R、RStudio、一个称为 tidyverse 的 R 包集合，以及另外几个 R 包。包是可重用 R 代码的基本单位，它们包括可重用的函数、描述函数使用方法的文档以及示例数据。

注 1：有关本书的详细信息，请参见图灵社区：http://www.ituring.com.cn/book/1540。

——编者注

R

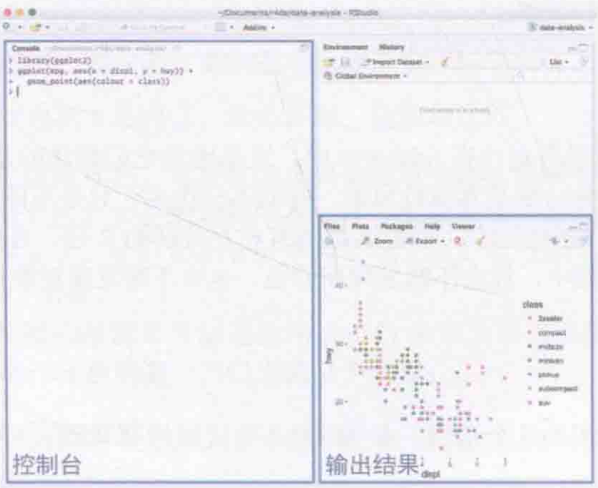
可以在 CRAN（comprehensive R archive network）下载 R。CRAN 由分布在世界各地的很多镜像服务器组成，用于分发 R 和 R 包。不要尝试选择离你近的服务器，而应该使用云镜像：https://cloud.r-project.org，它会自动找出离你最近的服务器。

R 的主版本一年发布一次，每年也会发布两三个次版本，因此你应该定期更新。更新 R 有一点麻烦，特别是更新主版本会要求你重新安装所有的 R 包，但是如果不更新的话，麻烦会更多。

RStudio

RStudio 是用于 R 编程的一种集成开发环境（integrated development environment，IDE）。你可以从 http://www.rstudio.com/download 下载并安装。RStudio 每年会更新多次。当有新版本时，RStudio 会进行通知。应该定期更新，这样你就可以使用其最新、最强大的功能。为了运行本书中的代码，请确认安装了 RStudio 1.0.0。

启动 RStudio 后，你会看到界面有以下两个关键区域。



现在，你只要知道可以在控制台窗格中输入 R 代码，然后按回车键运行就够了。在学习本书的过程中，你会学到 RStudio 的更多使用方法。

tidyverse

你还需要安装一些 R 包。R **包**是函数、数据和文档的集合，是对 R 基础功能的扩展。只有学会如何使用 R 包，才能真正掌握 R 语言的精华。你在本书中学习的大多数 R 包都是 tidyverse 的一部分。tidyverse 中的 R 包有着同样的数据处理与编程理念，它们的设计从根本上就是为了协同工作。

你可以用一行代码完整地安装 tidyverse：

```
install.packages("tidyverse")
```

在计算机上启动 RStudio 并在控制台中输入这行代码，然后按回车键来运行。R 会从 CRAN 下载这个包并将其安装在你的计算机上。如果安装有问题，请先确认你连接了互联网，再确认 https://cloud.r-project.org 没有被你的防火墙或代理服务器阻拦。

如果没有使用 `library()` 函数加载 R 包，那么你就不能使用其中的函数、对象和帮助文件。一旦 R 包安装完成，你就可以使用 `library()` 函数进行加载：

```
library(tidyverse)
#> Loading tidyverse: ggplot2
#> Loading tidyverse: tibble
#> Loading tidyverse: readr
#> Loading tidyverse: purrr
#> Loading tidyverse: dplyr
#> Conflicts with tidy packages ----------------------------------
#> filter(): dplyr, stats
#> lag():    dplyr, stats
```

以上结果表明，tidyverse 正在加载 R 包 ggplot2、tibble、readr、purrr 和 dplyr。这些包被视为 tidyverse 的**核心**，因为几乎在所有的分析中都会用到它们。

tidyverse 中的包修改得相当频繁。你可以通过运行 `tidyverse_update()` 函数来检查是否有更新，并选择是否进行更新。

其他包

还有很多优秀的 R 包没有包含在 tidyverse 中，这是因为它们解决的是其他领域中的问题，或者它们遵循的是另外一套基本设计原则。它们不分优劣，只是不同而已。换句话说，与 tidyverse 互补的不是 messyverse，而是其他所有相互关联的 R 包。在使用 R 完成越来越多的数据科学项目的过程中，你会不断发现新的包，也会不断更新对数据的认识。

运行R代码

前面展示了运行 R 代码的几个示例。本书以如下方式展示 R 代码：

```
1 + 2
#> [1] 3
```

如果在本地控制台中运行同样的代码，将得到如下结果：

```
> 1 + 2
[1] 3
```

这里有两处主要区别。在控制台中，需要在 `>`（**提示符**）后面输入代码，但本书不显示提示符。书中的输出结果被 `#>` 注释掉了，但是控制台中的输出结果则直接显示在代码后面。这样一来，如果你阅读的是本书电子版，你就可以轻松地将代码从书中复制到控制台。

全书使用一致的规则来表示代码。

- 函数与代码的字体相同，并且其后有圆括号，如 `sum()` 或 `mean()`。
- 其他 R 对象（比如数据或函数参数）也使用代码字体，但其后没有圆括号，如 `flights` 或 `x`。

- 如果想要明确指出对象来自于哪个 R 包，那么我们会在包的名称后面加两个冒号，如 `dplyr::mutate()` 或 `nycflights13::flights`；R 代码也支持这种形式。

获取帮助及更多学习资源

本书并非知识孤岛，单单利用一种资源是无法精通 R 语言的。当开始将本书介绍的技术应用于自己的数据时，你很快就会发现本书并未解答所有问题。本节将介绍几个获取帮助的小技巧，以帮助你持续地学习。

如果遇到问题，首先应该求助于 Google。通常来说，在查询内容时加上一个“R”，就足以得到与 R 相关的结果。如果查不到有用的结果，这意味着目前还没有特定的 R 解决方案。Google 特别适合查询错误消息。如果收到一条错误消息，但根本不知道其含义，那就用 Google 搜索一下吧！很可能有人遇到过这种错误，而答案就在网上。（如果错误消息不是英文，可以运行 `Sys.setenv(LANGUAGE = "en")`，接着重新运行代码；使用英文错误消息进行查询更可能获得帮助。）

如果 Google 没有奏效，那么可以试试 Stack Overflow。先花点时间搜索一下现成的答案；使用 `[R]` 可以将搜索范围限定在与 R 相关的问题和答案中。如果没有发现任何有用的内容，那么就准备一个最简单的可重现实例，即 reprex。良好的 reprex 让你更容易从他人那里获得帮助，而且在准备 reprex 时，你往往自己就能发现问题所在。

reprex 的准备工作应该包括 3 项内容：所需 R 包、数据和代码。

- 应该在脚本开头就加载 R 包，这样就会很容易知道 reprex 都需要哪些 R 包。应该趁机检查自己是否使用了每个 R 包的最新版本；你可能会发现，当安装了某个 R 包的最新版本后，问题就解决了。对于 tidyverse 中的 R 包来说，检查版本的最简单方式是运行 `tidyverse_update()` 函数。
- 在 reprex 中包含数据的最简单方法是使用 `dput()` 函数生成重建数据的 R 代码。例如，要想在 R 中重建 `mtcars` 数据集，可以遵循以下步骤：

  (1) 在 R 中运行 `dput(mtcars)`；

  (2) 复制输出结果；

  (3) 在可重现脚本中输入 `mtcars <-`，然后粘贴输出结果。

  应该努力找出依然能够反映问题的数据最小子集。
- 花一点时间确保别人可以轻松理解你的代码：
  - 确保使用了空格，并且变量名简明扼要；
  - 用注释来说明你的问题所在；
  - 尽最大努力去除所有与问题不相关的内容。

  代码越短，越容易理解，问题也就越容易解决。

启动一个新的 R 会话，将你的脚本复制并粘贴进去，检查 reprex 是否已经准备完毕。

你还应该花些时间来防患于未然。每天花一点时间学习 R，长远来看你将获得丰厚的回

报。可以在 RStudio 博客上关注 Hadley、Garrett 和其他 RStudio 开发人员的动态。我们会在博客上发布有关新 R 包、新 IDE 功能和面授课程的一些公告。你还可以在 Twitter 上关注 Hadley（@hadleywickham）和 Garrett（@statgarrett），也可以关注 @rstudiotips 来了解 RStudio 的新功能。

为了更好地掌握 R 社区的最新动态，我们建议你关注 R-bloggers 这个网站，该网站汇集了世界各地 500 多个关于 R 的博客。如果你是活跃的 Twitter 用户，可以关注 #rstats 这个主题标签。Twitter 是 Hadley 跟踪 R 社区最新发展的一个关键工具。

致谢

本书不仅是 Hadley 和 Garrett 的作品，还是我们与 R 社区很多用户（面对面和线上）对话的结果。我们要特别感谢一些人，因为他们花费了很多时间来回答我们的问题，并帮助我们更加深刻地理解了数据科学。

- 感谢 Jenny Bryan 和 Lionel Henry 就列表和列表列的使用与我们进行了多次有益的讨论。
- 感谢 Jenny Bryan 允许我们改编其文章“R basics, workspace and working directory, RStudio projects”，进而形成了本书关于工作流的 3 章内容。
- 感谢 Genevera Allen 与我们讨论模型、建模、统计学习前景以及假设生成和假设验证的区别。
- 感谢谢益辉为 R 包 bookdown 所做的工作，同时还要感谢他不断满足我们的功能需求。
- 感谢 Bill Behrman 仔细通读了全书，并在其斯坦福数据科学课堂中试用了本书。
- 感谢使用 #rstats 主题标签的所有 Twitter 用户，他们审阅了本书全部章节的草稿，并提供了大量有用的反馈。
- 感谢 Tal Galili 扩展了其 R 包 dendextend 以支持与聚类相关的一节，虽然最终稿中并未包含这项内容。

本书是以开源方式写成的，很多人提出了修改意见并帮助改正了各种小问题。特别感谢所有通过 GitHub 为本书做出贡献的人们（按字母顺序排列）：adi pradhan、Ahmed ElGabbas、Ajay Deonarine、@Alex、Andrew Landgraf、@batpigandme、@behrman、Ben Marwick、Bill Behrman、Brandon Greenwell、Brett Klamer、Christian G. Warden、Christian Mongeau、Colin Gillespie、Cooper Morris、Curtis Alexander、Daniel Gromer、David Clark、Derwin McGeary、Devin Pastoor、Dylan Cashman、Earl Brown、Eric Watt、Etienne B. Racine、Flemming Villalona、Gregory Jefferis、@harrismcgehee、Hengni Cai、Ian Lyttle、Ian Sealy、Jakub Nowosad、Jennifer (Jenny) Bryan、@jennybc、Jeroen Janssens、Jim Hester、@jjchern、Joanne Jang、John Sears、Jon Calder、Jonathan Page、@jonathanflint、Julia Stewart Lowndes、Julian During、Justinas Petuchovas、Kara Woo、@kdpsingh、Kenny Darrell、Kirill Sevastyanenko、@koalabearski、@KyleHumphrey、Lawrence Wu、Matthew Sedaghatfar、Mine Cetinkaya-Rundel、@MJMarshall、Mustafa Ascha、@nate-d-olson、Nelson Areal、Nick Clark、@nickelas、@nwaff、@OaCantona、Patrick Kennedy、Peter Hurford、Rademeyer Vermaak、Radu Grosu、@rlzijdeman、Robert Schuessler、@robinlovelace、@robinsones、S'busiso Mkhondwane、@seamus-mckinsey、@seanpwilliams、Shannon Ellis、@shoili、@sibusiso16、@spirgel、Steve Mortimer、@svenski、Terence Teo、Thomas Klebel、TJ Mahr、Tom Prior、Will Beasley，以及谢益辉。

生成环境

本书的生成环境如下所示。

```
devtools::session_info(c("tidyverse"))
#> Session info -------------------------------------------------
#>  setting  value
#>  version  R version 3.3.1 (2016-06-21)
#>  system   x86_64, darwin13.4.0
#>  ui       X11
#>  language (EN)
#>  collate  en_US.UTF-8
#>  tz       America/Los_Angeles
#>  date     2016-10-10
#> Packages -----------------------------------------------------
#>  package      * version    date       source
#>  assertthat     0.1        2013-12-06 CRAN (R 3.3.0)
#>  BH             1.60.0-2   2016-05-07 CRAN (R 3.3.0)
#>  broom          0.4.1      2016-06-24 CRAN (R 3.3.0)
#>  colorspace     1.2-6      2015-03-11 CRAN (R 3.3.0)
#>  curl           2.1        2016-09-22 CRAN (R 3.3.0)
#>  DBI            0.5-1      2016-09-10 CRAN (R 3.3.0)
#>  dichromat      2.0-0      2013-01-24 CRAN (R 3.3.0)
#>  digest         0.6.10     2016-08-02 CRAN (R 3.3.0)
#>  dplyr        * 0.5.0      2016-06-24 CRAN (R 3.3.0)
#>  forcats        0.1.1      2016-09-16 CRAN (R 3.3.0)
#>  foreign        0.8-67     2016-09-13 CRAN (R 3.3.0)
#>  ggplot2      * 2.1.0.9001 2016-10-06 local
#>  gtable         0.2.0      2016-02-26 CRAN (R 3.3.0)
#>  haven          1.0.0      2016-09-30 local
#>  hms            0.2-1      2016-07-28 CRAN (R 3.3.1)
#>  httr           1.2.1      2016-07-03 cran (@1.2.1)
#>  jsonlite       1.1        2016-09-14 CRAN (R 3.3.0)
#>  labeling       0.3        2014-08-23 CRAN (R 3.3.0)
#>  lattice        0.20-34    2016-09-06 CRAN (R 3.3.0)
#>  lazyeval       0.2.0      2016-06-12 CRAN (R 3.3.0)
#>  lubridate      1.6.0      2016-09-13 CRAN (R 3.3.0)
#>  magrittr       1.5        2014-11-22 CRAN (R 3.3.0)
#>  MASS           7.3-45     2016-04-21 CRAN (R 3.3.1)
#>  mime           0.5        2016-07-07 cran (@0.5)
#>  mnormt         1.5-4      2016-03-09 CRAN (R 3.3.0)
#>  modelr         0.1.0      2016-08-31 CRAN (R 3.3.0)
#>  munsell        0.4.3      2016-02-13 CRAN (R 3.3.0)
#>  nlme           3.1-128    2016-05-10 CRAN (R 3.3.1)
#>  openssl        0.9.4      2016-05-25 cran (@0.9.4)
#>  plyr           1.8.4      2016-06-08 cran (@1.8.4)
#>  psych          1.6.9      2016-09-17 CRAN (R 3.3.0)
#>  purrr        * 0.2.2      2016-06-18 CRAN (R 3.3.0)
#>  R6             2.1.3      2016-08-19 CRAN (R 3.3.0)
#>  RColorBrewer   1.1-2      2014-12-07 CRAN (R 3.3.0)
#>  Rcpp           0.12.7     2016-09-05 CRAN (R 3.3.0)
#>  readr        * 1.0.0      2016-08-03 CRAN (R 3.3.0)
#>  readxl         0.1.1      2016-03-28 CRAN (R 3.3.0)
#>  reshape2       1.4.1      2014-12-06 CRAN (R 3.3.0)
#>  rvest          0.3.2      2016-06-17 CRAN (R 3.3.0)
```

```
#>  scales        0.4.0.9003 2016-10-06 local
#>  selectr       0.3-0      2016-08-30 CRAN (R 3.3.0)
#>  stringi       1.1.2      2016-10-01 CRAN (R 3.3.1)
#>  stringr       1.1.0      2016-08-19 cran (@1.1.0)
#>  tibble      * 1.2        2016-08-26 CRAN (R 3.3.0)
#>  tidyverse   * 1.0.0      2016-09-09 CRAN (R 3.3.0)
#>  xml2          1.0.0.9001 2016-09-30 local
```

排版约定

本书使用了下列排版约定。

- **黑体**
  表示新术语和重点强调的内容。
- 等宽字体（`constant width`）
  表示程序片段，以及正文中出现的变量、函数名、数据库、数据类型、环境变量、语句和关键字等。
- 加粗等宽字体（**`constant width bold`**）
  表示应该由用户输入的命令或其他文本。
- 等宽斜体（*`constant width italic`*）
  表示应该由用户输入的值或根据上下文确定的值替换的文本。

该图标表示提示或建议。

使用代码示例

本书源代码可以从图灵社区本书页面免费下载：http://www.ituring.com.cn/book/2113。

本书是要帮你完成工作的。一般来说，如果本书提供了示例代码，你可以把它用在你的程序或文档中。除非你使用了很大一部分代码，否则无须联系我们获得许可。比如，用本书的几个代码片段写一个程序就无须获得许可，销售或分发 O'Reilly 图书的示例光盘则需要获得许可；引用本书中的示例代码回答问题无须获得许可，将书中大量的代码放到你的产品文档中则需要获得许可。

我们很希望但并不强制要求你在引用本书内容时加上引用说明。引用说明一般包括书名、作者、出版社和 ISBN。比如："*R for Data Science* by Hadley Wickham and Garrett Grolemund (O'Reilly). Copyright 2017 Garrett Grolemund, Hadley Wickham, 978-1-491-91039-9."

如果你觉得自己对示例代码的用法超出了上述许可的范围，欢迎你通过 permissions@oreilly.com 与我们联系。

O'Reilly Safari

Safari（原来叫 Safari Books Online）是一个会员制的培训和参考咨询平台，面向企业、政府、教育从业者和个人。

会员可以访问几千种书籍、培训视频、学习路径、互动式教程和精选播放列表，提供这些资源的出版商超过 250 家，包括 O'Reilly Media、Harvard Business Review、Prentice Hall Professional、Addison-Wesley Professional、Microsoft Press、Sams、Que、Peachpit Press、Adobe、Focal Press、Cisco Press、John Wiley & Sons、Syngress、Morgan Kaufmann、IBM Redbooks、Packt、Adobe Press、FT Press、Apress、Manning、New Riders、McGraw-Hill、Jones & Bartlett、Course Technology，等等。

要获得更多信息，请访问 http://oreilly.com/safari。

联系我们

请把对本书的评价和问题发给出版社。

美国：

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472

中国：

北京市西城区西直门南大街 2 号成铭大厦 C 座 807 室（100035）
奥莱利技术咨询（北京）有限公司

O'Reilly 的每一本书都有专属网页，你在上面可以找到图书的相关信息，包括勘误表、示例代码以及其他信息。

对于本书的评论和技术性问题，请发送电子邮件到：

bookquestions@oreilly.com

要了解更多 O'Reilly 图书、培训课程、会议和新闻的信息，请访问以下网站：http://www.oreilly.com

我们在 Facebook 的地址如下：http://facebook.com/oreilly

请关注我们的 Twitter 动态：http://twitter.com/oreillymedia

我们的 YouTube 视频地址如下：http://www.youtube.com/oreillymedia