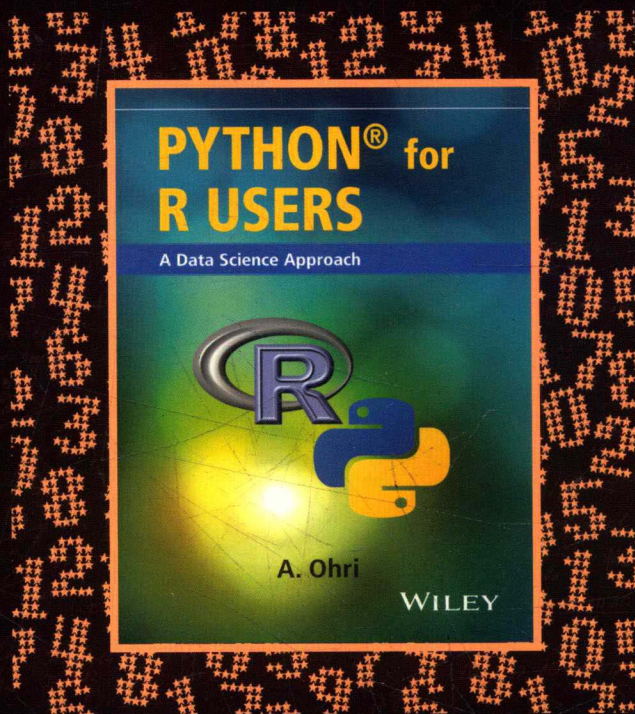


R用户Python学习指南

数据科学方法

[印度] A. 奥利 (A. Ohri) 著
潘文捷 李洪成 译



PYTHON FOR R USERS
A DATA SCIENCE APPROACH



机械工业出版社
China Machine Press

数据科学与工程技术丛书

PYTHON FOR R USERS

A DATA SCIENCE APPROACH

R用户Python学习指南

数据科学方法

[印度] A. 奥利 (A. Ohri) 著

潘文捷 李洪成 译



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

R 用户 Python 学习指南: 数据科学方法 / (印) A. 奥利 (A. Ohri) 著; 潘文捷, 李洪成译.
—北京: 机械工业出版社, 2018.10

(数据科学与工程丛书)

书名原文: Python for R Users: A Data Science Approach

ISBN 978-7-111-61195-0

I. R… II. ① A… ② 潘… ③ 李… III. 软件工具-程序设计-指南 IV. TP311.561-62

中国版本图书馆 CIP 数据核字 (2018) 第 241844 号

本书版权登记号: 图字 01-2018-2514

Copyright © 2018 John Wiley & Sons, Inc.

All rights reserved. This translation published under license. Authorized translation from the English language edition, entitled Python for R Users: A Data Science Approach, ISBN 9781119126768, by A. Ohri, Published by John Wiley & Sons. No part of this book may be reproduced in any form without the written permission of the original copyrights holder.

本书中文简体字版由约翰·威立父子公司授权机械工业出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

本书封底贴有 Wiley 防伪标签, 无标签者不得销售。

R 用户 Python 学习指南: 数据科学方法

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 毕彦青

责任校对: 李秋荣

印刷: 北京市荣盛彩色印刷有限公司

版次: 2019 年 1 月第 1 版第 1 次印刷

开本: 185mm×260mm 1/16

印张: 15.5

书号: ISBN 978-7-111-61195-0

定价: 69.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

译者序

R 语言是一款统计分析语言和软件，自 20 世纪 90 年代初作为开源软件公开发行以来，吸引了越来越多的用户。借助于丰富的添加包（add-on package），R 语言的统计、分析和数据可视化功能越来越强大。而 Python 是一款通用编程语言，也开发于 90 年代初。Python 语言以语法简洁而著称，并且有大量基于 Python 的模块（module）。随着大数据分析和机器学习的发展，这两款编程语言都得到了大量的应用。

作为数据分析从业人员，仅仅掌握一门分析语言有时候是不够的。在做传统的统计分析时，可根据具体情况来选择分析工具，既可以选择商业软件 SAS 和 SPSS，也可以选择开源软件 R。而在进行深度学习时，当前大多数模型都是基于开源软件 Python 或 R 的。因此，以前进行传统统计分析的人员，在应用深度学习模型时，必须掌握新的工具软件。

R 和 Python 本身是不同的语言，但它们能够实现的功能具有一定的重合性。本书就是为 R 用户介绍如何应用 Python 的。在介绍 Python 的功能时，也介绍了对应的 R 的实现方法。通过这种对比学习，R 语言的读者可以较快掌握另外一种功能强大的语言——Python。本书适合已经了解 R 的用户学习 Python 时阅读，也适合掌握 Python 的用户来学习 R。

本书是从实践案例的角度撰写的，旨在帮助读者在追求卓越数据科学的过程中驾驭多种开源语言。没有一种软件或语言能够一直解决各种数据问题，所以在掌握多种语言和工具之后，就可以根据具体问题和数据来灵活地选用相应的工具。

本书的翻译由李洪成和潘文捷共同完成，另外，姜越和胡超也在校订中提供了有益的建议。

译者

2018 年 10 月

前言

我从2003年起开始了销售汽车的职业生涯。这是我在获得工商管理硕士学位2年和工程学硕士学位4年后的第一份工作。另外，我有2年的时间先后以一名军官学员（1年内退学）和物理学家（1年后退学）的身份进入军校。很久以后，我在诺克斯维尔攻读博士学位（数理统计），1年后退学。在工程学硕士、工商管理硕士甚至是研究生学院，我的统计理论都做得不好。我只对统计软件感兴趣，幸运的是，在使用软件方面我做得还不错。因此，在2004年，我不再销售汽车，而是开始为通用电气基于印度的离岸公司编写统计软件。

我使用名为Base SAS的软件以及SAS语言。由SAS软件公司提供的软件和编程语言的帮助系统非常好，因此整天摆弄数据和代码就能够赚钱，玩得很开心。经过几年的工作变动后，当我开始创建自己的公司时，我遇到了开源软件。我真的很喜欢作为一种语言和一家公司的SAS，但是在公司初创阶段，我无法承担它的费用，而2007年时，SAS大学版并没有出现。因为需要钱来支付我的小孩Kush的尿布，而数据分析是上帝给我的唯一礼物，因此我转向了R。

R、Open Office 和 Ubuntu Linux 是我进入开源统计计算领域便开始使用的三个工具，并坚持至今。2007年，我创立了自己的公司——商业分析写作和咨询公司 Decisionstats.com。2009年，我获得助学金进入田纳西大学，那年冬季我又在硅谷实习了几个星期，除了统计系外，我还在计算机科学系学习了图形设计和遗传算法等多门课程，其后因健康原因退学。跨领域培训帮助我以各种方式思考并能找到简单的解决方案，我将永远感谢田纳西大学统计与计算机科学系的友善的朋友们。

一旦掌握了解答Linux各种疑难的技巧和R的面向对象编程思想，我就可以为数据分析项目提供咨询服务了。那时候我们称之为商务分析，但今天我们称之为数据科学。

由于经常忘事，包括想不起来把代码保存在哪，因此我开始写博客，把有用的事情记录下来，并且这也可能对其他人有用。若干年后我才明白，在现实生活中真正对事业有帮助的，并不是我所知道的东西，而是我所认识的人。因此，我开始采访从事数据分析和应用R的人，并且我的博客访问人数也开始增加。我的博客哲学仍然是——博客文章应该是有用的，它应该是独一无二的，也应该是有兴趣的。2016年，我在 Decisionstats.com 积累了1 000 000次访问次数——这对我来说也是一次令人惊讶的转变。我非常感谢同意接受我

采访的 100 多人。

当然，2007 年和 2008 年只是刚开始准备分析博客。几年后，我有足够的材料来编写一本书，并且也赢得了出版商的信任。2012 年，我出版了第一本书，2014 年又出版了第二本书。2016 年，我的第一本书的中文版出版了。让我惊讶的是，我的第二本书的评论出现在杂志《Journal of Statistical Software》上。

在出版了两本有关 R 的书籍后，我通过咨询和培训指导了许多创业公司，让咨询客户参与到现在世界的问题中，并在社交媒体上成了一个众人皆知的名字，但我仍然觉得自己需要了解更多。

数据变得越来越多。知道如何使用一台机器在序列化代码中进行小型数据分析是不够的，也许是时候在多台机器上编写大数据分析的并行代码了。而且，统计人员和计算机科学之间的分歧让我着迷，因为我将数据视为数据，它们就是需要解决的问题。正如 Eric S. Raymond 在关于黑客的态度中所写的：“世界充满了有趣的问题。”

还有一种替代 R 的诱惑和智慧的吸引力，那就是 Python，它自身带有“电池”（据称）。

一旦我的科学好奇心受到激发，便开始学习 Python。我发现 Python 与 R 相比非常好，但也非常糟糕。它的社区有不同的规则和行为（开源开发者的激情世界总是动荡不安）。但语言本身是非常不同的。我不在乎语言，我喜欢科学。但是如果像我这样至少知道如何在 R 中编写代码的人，发现在 Python 中重做同样的事情非常困难，我想也许其他人也面临着这个问题。对于大数据和某些特定用例，Python 在速度方面表现更好。速度很重要，不管摩尔定律有多强，两者共同使用可以让你更容易地编写代码。R 似乎也变成了一种语言，我所做的只是导入一个添加包并运行带有调整参数的函数。随着 R 在科学领域取代 SAS 语言成为主流，SAS 仍然是企业统计语言，而 Python 则成为像我这样的匿名红帽黑客冒险钻研和探索的东西。

随着互联网逐渐扩展到物联网，我觉得新兴的数据科学家至少应该在分析中知道两种语言，以便他们的职业生涯能够相对平稳。当企业考虑使用哪种软件来为模型建模，以及在生产环境中部署哪些软件时，这也提供了一个开放的选择。

致谢

感谢许多在 Python 和 R 社区工作的人员，是他们使本书成为可能。特别感谢 Predictive Analytics Conference 的 Eric Siegel 博士和 JMP 的 John Sall。感谢 2012~2016 年的所有学生。

如果没有 MadhurBatra 的指导 and 后勤支持，这本书不会完成。在技术方面，感谢他的实习生 Yashika 和 ChandanRoutray（IIT Kharagpur）以及协助他的 DecisionStats 团队的投入和辛勤工作。他的核心研究人员 F.Xavier 为案例研究提供了宝贵的帮助。

范围

本书将 Python 作为数据科学实践者的平台，包括有抱负的新兴数据科学家。这本书的目标读者是那些知道各种专业知识的 R 程序员，但即使是那些不懂任何编程技巧的人也会发现它的价值。它的目标读者不是研究社区和研究部门的成员。本书的重点在于简单的教程和可实践的分析，而不是理论。我还试图将它与 R 代码结合起来，为学习者提供对比的方法。

第 1 章

这一章向读者介绍了 Python 和 R 语言的基础知识，并将 Python 与 R 进行比较。同时，列出了在这两种语言中使用的函数和添加包，还列出了一些我认为数据科学家应该意识的管理模式。

第 2 章

介绍人们在 Python 中获取各种量级数据的方法和速度，包括网络抓取、数据库、noSQL 数据和电子表格数据。

第 3 章

介绍了 Python 中验证数据质量的方法。

第 4 章

介绍基本的数据探索和数据汇总，并按照标准分组汇总数据。

第 5 章

基于统计分析创建模型，包括 OLS 回归，这些统计分析对于在工业中建立倾向模型 (propensity model) 非常有用。

第 6 章

介绍检查原始数据和汇总数据的可视化方法。

第 7 章

介绍模型构建中常用的数据挖掘方法。这里主要关注有监督和无监督的方法，并进一步强调回归和聚类技术。时间序列预测部分有助于用户进行时间序列预测。文本挖掘部分介绍了文本挖掘方法和自然语言处理。Web 分析着眼于使用 Python 来分析 Web 数据。高级数

据科学着眼于更新案例的方法和技术，包括启用云计算的大数据分析、社交网络分析、物联网等。

第 8 章

列出了本书中我们学习和尝试实现的内容，以及对 R 和 Python 未来发展的看法，包括统计计算的发展，并使数据科学成为未来可靠的立足点。

目的

本书是从实际实践案例的角度撰写的，旨在帮助人们在追求卓越数据科学的过程中驾驭多种开源语言。我认为，没有一种软件或语言能够一直解决各种数据问题。优化的学习方法比学习统计软件的思想方法更有用。过去的思维习惯必须调整以面对未来知识增长的速度。

计划

我将继续使用截图作为导读工具，并将利用我在数据科学咨询方面的经验，突出实际的数据分析问题。这是因为选择正确的工具和技术甚至添加包都不是那么耗时，但是数据和业务问题的多样性可能会花费数据科学家的时间，从而影响他的判断和解决方案的质量。

目标读者

这是一本为即将成为数据科学家和已经成为数据科学家（如使用 SPSS、R 或 Julia 等其他语言的数据科学家）的人编写的书。我致力于解决数据中的实际问题。因此，这本书的理论内容很少。

后记

我专注于实际解决方案。因此，我将继续假设用户希望以最低的成本和最高的准确性、稳健性和易用性进行数据挖掘或分析。一个真正的科学家（无论是如何成为的）总是对数据和选项持开放态度。我发现，信息不对称和名称混乱使得读者对 R 与 Python 相对其他语言的真正优势产生了混淆。本书中的说明和教程不保证可以解决任何领域的问题，读者需自行承担风险。

关于本书格式需要特别说明的是，我大多是在 Google Docs 上撰写的，但本书却使用了排版软件 LaTeX 的 GUI 版本 LyX，我承认对此并不擅长。我希望本书能够被商业用户、技

术用户、热衷于了解更多关于 R 和 Python 的首席技术官、使用开源分析以及希望成为数据科学家的学生阅读。R 以出色的图形而闻名，但不适合通过其原有的方式直接使用开源版本的较大数据集。Python 在大数据集和灵活性方面很出色，但它一直希望像 R 一样拥有大量可用的良好统计库。

企业首席技术官可以通过混合云和混合开源软件使用开源软件和硬件，从而极大地降低成本。

Python 之禅

Tim Peters (来源: <https://www.python.org/dev/peps/pep-0020/>)

- ❑ Beautiful is better than ugly.
优美胜于丑陋 (Python 以编写优美的代码为目标)。
- ❑ Explicit is better than implicit.
明了胜于晦涩 (优美的代码应当是明了的, 命名规范, 风格相似)。
- ❑ Simple is better than complex.
简洁胜于复杂 (优美的代码应当是简洁的, 不要有复杂的内部实现)。
- ❑ Complex is better than complicated.
复杂胜于凌乱 (如果复杂不可避免, 那么代码间也不能有难懂的关系, 要保持接口简洁)。
- ❑ Flat is better than nested.
扁平胜于嵌套 (优美的代码应当是扁平的, 不能有太多的嵌套)。
- ❑ Sparse is better than dense.
间隔胜于紧凑 (优美的代码有适当的间隔, 不要奢望一行代码解决问题)。
- ❑ Readability counts.
可读性很重要 (优美的代码是可读的)。
- ❑ Special cases aren't special enough to break the rules.
即便假借特例的实用性之名, 也不可违背这些规则 (这些规则至高无上)。
- ❑ Errors should never pass silently. Unless explicitly silenced.
不要包容错误, 除非你确定需要这样做 (精准地捕获异常, 不写 `except: pass` 风格的代码)。
- ❑ In the face of ambiguity, refuse the temptation to guess.
- ❑ There should be one—and preferably only one—obvious way to do it.
- ❑ Although that way may not be obvious at first unless you're Dutch.

当存在多种可能时，不要尝试去猜测，而是尽量找一种，最好是唯一一种明显的解决方案（如果不确定，就用穷举法）。虽然这并不容易，因为你不是 Python 之父。

- ❑ Now is better than never. Although never is often better than right now.

做也许好过不做，但不假思索就动手还不如不做（动手之前要细思量）。

- ❑ If the implementation is hard to explain, it's a bad idea.

- ❑ If the implementation is easy to explain, it may be a good idea.

如果无法向别人描述你的方案，那肯定不是一个好方案；反之亦然（方案测评标准）。

- ❑ Namespaces are one honking great idea -- let's do more of those!

命名空间是一种绝妙的理念，我们应当多加利用（倡导与号召）。

目 录

译者序

前言

第 1 章 Python、R 和 R 数据科学

简介..... 1

1.1 什么是 Python..... 1

1.2 什么是 R..... 2

1.3 什么是数据科学..... 2

1.4 数据科学家的未来..... 3

1.5 什么是大数据..... 4

1.6 商务分析与数据科学..... 5

1.7 数据科学家可用的工具..... 6

1.8 用于数据科学的 Python 添加包..... 7

1.9 Python 和 R 之间的异同..... 8

1.9.1 为什么 R 用户应该了解
有关 Python 的更多信息..... 8

1.9.2 为什么 Python 用户应该
了解有关 R 的更多信息..... 9

1.10 教程..... 9

1.11 同时使用 R 和 Python..... 9

1.12 其他软件和 Python..... 13

1.13 将 SAS 与 Jupyter 一起使用..... 13

1.14 如何将 Python 和 R 用于大数据
分析..... 13

1.15 什么是云计算..... 15

1.16 如何在云上使用 Python 和 R..... 15

1.17 Python 和 R 商业版本及其他

替代版本..... 16

1.17.1 数据科学家的常用 Linux
命令..... 17

1.17.2 学习 Git..... 18

1.18 数据驱动的决策..... 32

1.18.1 企业管理战略框架：非 MBA
和 MBA 的数据驱动决策
指南..... 32

1.18.2 商务分析的其他框架..... 37

参考文献..... 40

第 2 章 数据输入..... 41

2.1 pandas 中的数据输入..... 41

2.2 网页抓取数据输入..... 44

2.3 来自 RDBMS 的数据输入..... 48

2.3.1 Windows 教程..... 50

2.3.2 安装..... 50

2.3.3 配置 ODBC..... 52

第 3 章 数据检查和数据质量..... 62

3.1 数据格式..... 62

3.1.1 将字符串转换为日期和
时间..... 63

3.1.2 将数据框转换为 NumPy
数组并返回 Python..... 66

3.2 数据质量..... 68

3.3 数据检查	71	参考文献	138
3.4 数据选择	74	第 6 章 数据可视化	139
3.4.1 随机选择数据	76	6.1 数据可视化的概念	139
3.4.2 条件选择	76	6.1.1 数据可视化的历史	139
3.5 R 中的数据检查	79	6.1.2 Anscombe 案例研究	140
3.5.1 来自 R ggplot2 添加包		6.1.3 加载添加包	141
中的 diamond 数据集	83	6.1.4 获得均值和标准差	142
3.5.2 修改 R 中的日期格式和		6.1.5 结论 1	144
字符串	86	6.1.6 数据可视化	144
3.5.3 管理 R 中的字符串	88	6.1.7 结论 2	146
参考文献	88	6.2 Tufte 关于数据可视化的工作	147
第 4 章 探索性数据分析	89	6.3 Stephen Few 关于仪表盘的设计	147
4.1 根据分析分组	89	6.4 基本绘图	149
4.2 数值数据	89	6.5 高级绘图	156
4.3 分类数据	91	6.6 交互式绘图	160
第 5 章 统计建模	102	6.7 空间分析	160
5.1 回归的概念	102	6.8 R 中的数据可视化	160
5.1.1 OLS	103	6.8.1 关于 RStudio IDE 共享 R	
5.1.2 R^2 方	103	代码的注意事项	166
5.1.3 p 值	104	6.8.2 关于共享 Jupyter 笔记的	
5.1.4 异常值	104	注意事项	167
5.1.5 多重共线性和异方差性	104	6.8.3 特别说明: 关于 Python	
5.2 相关关系不是因果关系	105	的完整教程	168
5.2.1 关于数据科学家统计的		参考文献	180
说明	106	第 7 章 机器学习变得更容易	181
5.2.2 中心趋势的度量	107	7.1 删除最终决策树模型中不需要	
5.2.3 分散度的度量	107	的列	186
5.2.4 概率分布	110	7.2 时间序列	207
5.3 R 和 Python 中的线性回归	114	7.3 关联分析	211
5.4 R 和 Python 中的 Logistic 回归	132	7.4 清洗语料库并制作词袋	221
5.4.1 其他概念	136	7.4.1 聚类分析	224
5.4.2 ROC 曲线和 AUC	137	7.4.2 Python 中的聚类分析	224
5.4.3 偏差与方差	137	第 8 章 结论和总结	233

第 1 章

Python、R 和 R 数据科学简介

1.1 什么是 Python

Python 是一种编程语言，可以让你更快地工作，并且能够更有效地集成系统。Python 语言由 Guido van Rossum 开发。你可以在 Python 的历史博客 <http://python-history.blogspot.in/2009/01/introduction-and-overview.html> 上阅读 Guido 讲述的 Python 历史。

对于 Python 初学者甚至是经验丰富的人来说，这是值得一读的。以下是其中一段摘录：

许多 Python 的关键字（if、else、while、for 等）与 C 语言中的相同，Python 标识符与 C 语言具有相同的命名规则，大多数标准运算符与 C 语言具有相同的含义。当然，显然 Python 不是 C 语言，它和 C 语言一个主要的不同是使用缩进而不是大括号进行语句分组。例如，在 C 语言中按如下方式编写语句：

```
if (a < b) {  
    max = b;  
} else {  
    max = a;  
}
```

Python 只是完全省略了括号（以及为了维护良好标准而使用的句末分号）并使用以下结构：

```
if a < b:  
    max = b  
else:  
    max = a
```

Python 与 C 语言不同的另一个主要领域是动态类型的使用。在 C 中，必须始终明确声明变量并给定其特定类型，如 int 或 double。然后，此信息用于执行程序

的静态编译时检查以及分配存储变量值的内存位置。在 Python 中，变量只是引用对象的名称。

Python Package Index (PyPI) (<https://pypi.python.org/pypi>) 托管 Python 的第三方模块。目前有 91 625 个添加包。你可以通过 <https://pypi.python.org/pypi?%3Aaction=browse> 按主题浏览 Python 的添加包。

1.2 什么是 R

有关 R 的官方定义见主网站 <http://www.r-project.org/about.html>。

R 是用于数据处理、计算和图形显示的集成软件工具套件。它包括一个有效的数据处理和存储设施，一套其中包含用于计算数列（特别是矩阵）的算子，一个用于数据分析的大型、连贯、集成的中间工具集合，其中包含用于数据分析的图形工具，可在屏幕上或硬拷贝上显示，以及一种发展良好、简单有效的编程语言，包括条件、循环、用户定义的递归函数以及输入和输出设施。

“环境”一词旨在将其描述为一个完全规划和连贯的系统，而不是非常特殊和不灵活的工具的增量增长（这是其他数据分析软件的常见情况）。

Comprehensive R Archive Network (CRAN) (<https://cran.r-project.org/web/packages/>) 上托管了数千个添加包 R 包，GitHub 也是如此（请参阅 <https://github.com/search?utf8=%E2%9C%93&q=stars%3A%3E1+language%3AR>），还有 Bioconductor 作为添加包库。你可以在 <http://www.rdocumentation.org/> 上查看来自这些存储库的所有包（截至 2016 年共有 11 885 个包）。

R 既是统计学中的语言，也是计算机科学和分析软件，在分析商务数据和应用数据科学方面非常有用。特别是 R 的吸引力仍然存在：它是一个免费的开源软件，拥有大量的添加包可用于数据分析。

R 的缺点仍然是生产环境中的内存处理，缺乏对 R 开发人员的激励，以及它有时生成的是稍显学术导向而不是企业用户导向的粗略文档。

1.3 什么是数据科学

数据科学位于编程、统计和业务分析的交叉点。它是根据统计技术以系统和科学的方式分析数据的编辑工具的应用。Drew Conway 的著名图表将数据科学作为三者的交集，见 <http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram>。

我将数据科学家定义如下：

数据科学家只是一个能够在硬件（本地机器、数据库、云、服务器上）上高效快速地编写代码（使用 R、Python、Java、SQL、Hadoop (Pig、HQL、MR) 等语言）来处理数据（存储、查询、汇总、可视化）的人，同时了解足够的统计数据以从数据中获取洞察力，以便业务部门做出决策。

1.4 数据科学家的未来

权威的《哈佛商业评论》将数据科学家定义为 21 世纪最性感的工作（<https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century/>）。

对工资的调查显示，数据科学家的需求和薪水都在上涨，而且受过训练的专业人员也面临严重短缺（参见 <http://www.forbes.com/sites/gilpress/2015/10/09/the-hunt-for-unicorn-data-scientists-lifts-salaries-for-all-data-analytics-professionals/>）。事实上，这创造了一个新术语：独角兽数据科学家（unicorn data scientist）。独角兽数据科学家很难找到，因为他具备编程、统计和商业方面的所有技能。更新的图 1.1 中的数据科学维恩图可在 <http://www.anlytcs.com/2014/01/data-science-venn-diagram-v20.html> 上查看。

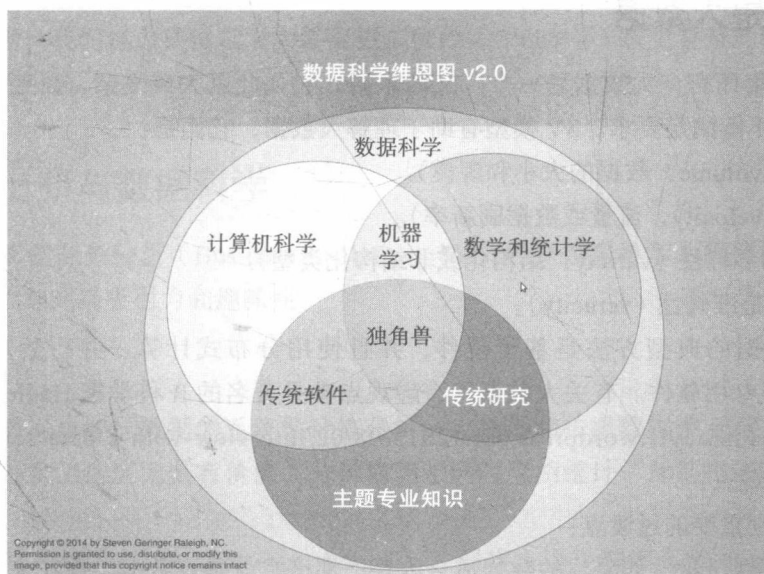


图 1.1 数据科学维恩图。来源：Copyright©2014 Steven Geringer Raleigh, NC

此外，独角兽是投资行业的术语，特别是风险投资行业，它代表一家初创公司，其估值超过 10 亿美元。这个词已经被 Cowboy Ventures 的 Aileen Lee 推广。可以在 <http://graphics.wsj.com/billion-dollar-club/> 和 <http://fortune.com/unicorns/> 上查看。

毫不奇怪，数据科学也为这些初创企业提供了关键优势。因此，我们可以同时满足不断增长的需求和数据科学家的短缺，创造更安全的工作环境。可以在 <http://yclist.com/> 的 Y Combinator 上看到一份初创企业名单，其中包括与数据科学相关的初创企业。你可以在 <http://www.burtchworks.com/2015/07/14/compensation-of-data-scientists-insights-from-the-past-year> 上查看有关数据科学家工资的调查。年度 Rexer Analytics 调查有助于衡量数据挖掘者的技能和使用情况。你可以在 <http://decisionstats.com/2013/12/25/karl-rexer-interview-on-the-state-of-analytics/> 上阅读访谈，或阅读 www.rexeranalytics.com 上的报告。因此，我们可以总结认为，具有合适技能的数据科学家在专业领域拥有美好的未来。

值得注意的是，数据科学家需要非常快速地更新技能，他们需要响应业务需求来构建数据科学解决方案。因此，过时的风险仍然能够鼓励数据科学家获得多种技能。Insight 在 <http://insightdatascience.com/> 上提供了一个有趣的数据科学研究奖学金计划，<https://github.com/okulbilisim/awesome-datascience> 上免费提供了数据科学的存储库。

言归正传，位于纽约的 Byte 学院提供基于 Python 的数据科学计划，网址为 <http://byteacademy.co/>。

1.5 什么是大数据

对于数据集而言，大数据是一个广义的术语，它如此巨大或复杂，以至于传统的数据处理应用程序不能满足要求。3V 模型有助于理解大数据，它们是：

- ❑ 数量 (volume, 数据的大小和规模)。
- ❑ 速度 (velocity, 流量或数据刷新率)。
- ❑ 数据的多样性 (variety, 结构化或非结构化类型)。

第四个 V 是准确性 (veracity)。

处理大数据的典型方法是基于硬件，并且使用分布式计算、并行处理、云计算和 Hadoop 堆栈等专用软件。有关大数据的有趣观点参见著名的 R 科学家 Hadley Wickham 博士在 <https://peadarcoyle.wordpress.com/2015/08/02/interview-with-a-data-scientist-hadley-wickham/> 上的观点：

有两个特别重要的过渡点：

- ❑ 从内存到磁盘。如果你的数据适合内存，那么它就是小数据。现在你可以获得 1TB 的内存，所以即使是小数据也很大！从内存迁移到磁盘是一个重要的过渡，因为访问速度是如此不同。你可以对内存数据进行非常朴素的计算，它会足够快。而对于使用磁盘数据，你需要计划（和索引）更多。
- ❑ 从一台计算机到多台计算机。当你的数据不再适合一台计算机上的一个磁盘时，会

出现下一个重要阈值。迁移到分布式环境会使计算变得更具挑战性，因为你进行计算所需的所有数据不在同一个地方。设计分布式算法要困难得多，并且你会极大地受到数据在计算机之间分离方式的限制。

Wes McKinney，数据科学的主要 Python 添加包 pandas 的作者，在 <http://wesmckinney.com/blog/the-problem-with-the-data-science-language-wars/> 上提出此问题：

任何允许你使用以 Python 或 R 等语言编写的用户定义代码做扩展的数据处理引擎必须解决至少这 3 个基本问题：

- ❑ 数据移动或访问：使用 Python 可以使用的形式访问运行时数据。不幸的是，这通常需要昂贵的序列化或反序列化，并可能主导系统运行时间。通过仔细创建共享字节级内存布局可以避免序列化成本，但这样做需要大量经验丰富且报酬丰厚的人员同意为更大的成果进行重大工程投资。
- ❑ 向量化计算：启用 Python 或 R 等解释语言来分摊开销并调用面向数组的快速编译代码（例如 NumPy 或 pandas 操作）。这些语言中的大多数添加包也希望使用数组/向量值而不是标量值。因此，如果你想使用自己喜欢的 Python 或 R 添加包，则需要此功能。
- ❑ IPC 开销：调用外部函数的低级机制。这可能涉及通过 UNIX socket 发送带有一些简短指令的简短消息。

对比小数据，我们将大数据定义为需要更多硬件（Cloud 等）或更复杂的编程或专用软件（Hadoop）的数据。

1.6 商务分析与数据科学

从统计计算到商务分析（Business Analytics, BA）再到数据科学的历史演变，在各种主张的争鸣声中都充满吸引力和趣味性。

定义分析

分析是对数据或统计数据系统性计算分析。它发现和传播数据中有意义的模式。在具有大量记录信息的地区尤其有价值，分析依赖于同时应用统计、计算机编程和运营研究来量化性能。

“信息阶梯”由教育学教授 Norman Longworth 创建，用于描述人类学习的各个阶段。根据这一阶梯，学习者通过以下进展来构建“数据”中的“智慧”：

数据 (data) → 信息 (information) → 知识 (knowledge) → 理解 (understanding) →
领悟 (insight) → 智慧 (wisdom)