HZ BOOKS
华章IT

[PACKT]
PUBLISHING

全面、系统讲解使用Scala在Spark平台上实现机器学习算法的实用技术、方法和最佳实践

提供大量有针对性的编程实例，助你充分利用Scala、Spark和Hadoop提升数据分析和处理的工程实战能力

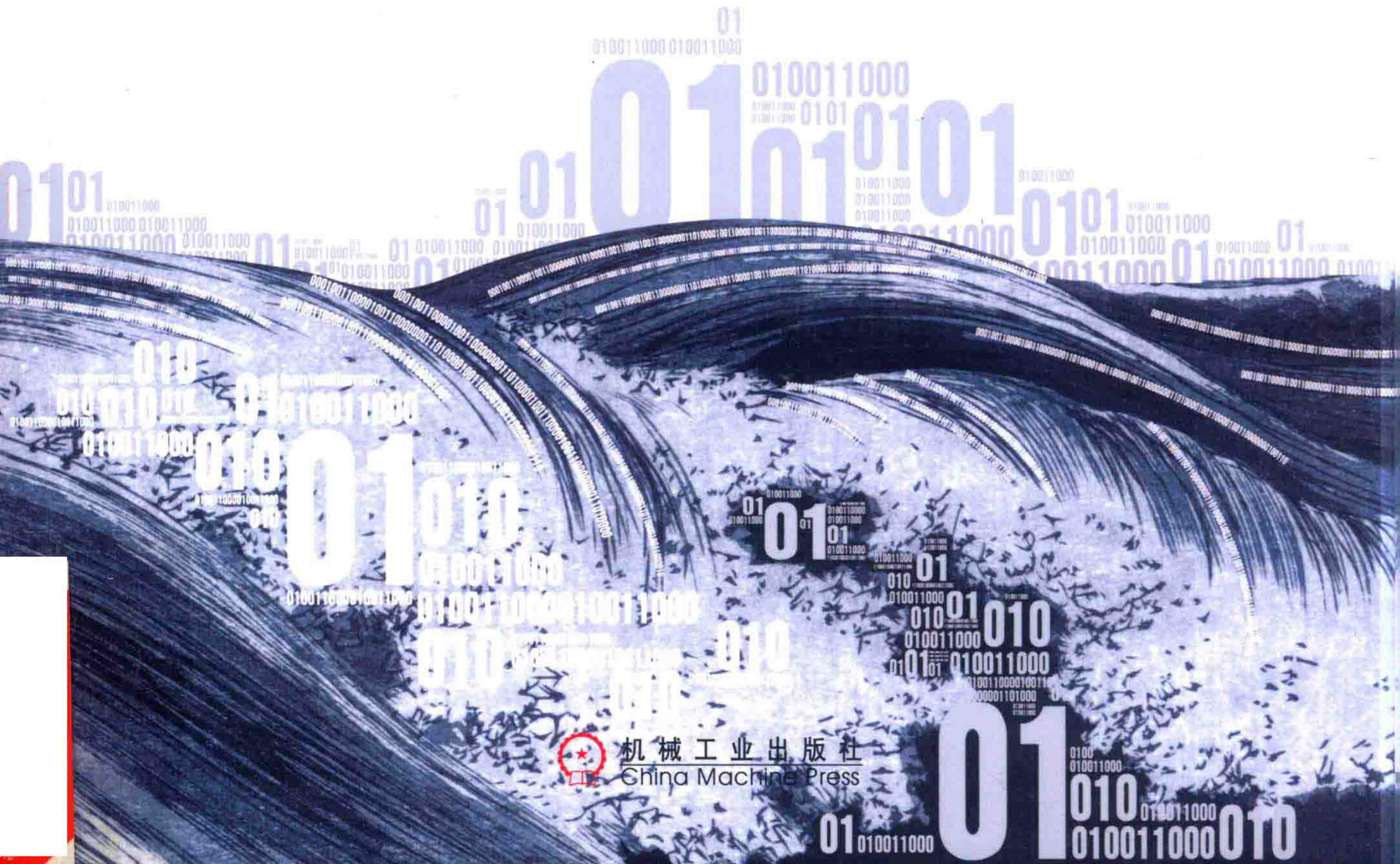
大数据技术丛书

Mastering Scala Machine Learning

Scala机器学习

[美] 亚历克斯·科兹洛夫(Alex Kozlov) 著

罗棻 刘波 译



机械工业出版社
China Machine Press

大数据技术丛书

Mastering Scala Machine Learning

Scala机器学习

[美] 亚历克斯·科兹洛夫(Alex Kozlov) 著

罗葵 刘波 译

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

Scala 机器学习 /（美）亚历克斯·科兹洛夫（Alex Kozlov）著；罗棻，刘波译 . —北京：机械工业出版社，2017.7
（大数据技术丛书）
书名原文：Mastering Scala Machine Learning

ISBN 978-7-111-57215-2

I. S…　II. ①亚…　②罗…　③刘…　III. JAVA 语言—程序设计　IV. TP312.8

中国版本图书馆 CIP 数据核字（2017）第 146773 号

本书版权登记号：图字：01-2016-8654

Scala 机器学习

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：缪　杰　　责任校对：殷　虹
印　　刷：三河市宏图印务有限公司　　版　　次：2017 年 7 月第 1 版第 1 次印刷
开　　本：186mm × 240mm　1/16　　印　　张：13.5
书　　号：ISBN 978-7-111-57215-2　　定　　价：59.00 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

华章IT
HZBOOKS | Information Technology

The Translator's Words 译 者 序

大数据是当前热门的话题，其特点为数据量巨大，增长速度快，拥有各种类型。分布式机器学习是一种高效处理大数据的方法，其目的是从大数据中找到有价值的信息。目前各大互联网公司都投入巨资研究分布式机器学习。

在实现分布式机器学习算法时，函数式编程有天生的优势。这是因为函数式编程不会共享状态，也不会造成资源竞争。Scala 是一种优秀的函数式编程语言，同时它也是基于 Java 虚拟机的面向对象的编程语言。使用 Scala 编程非常方便快捷。

Spark 是 2009 年出现的一种基于内存的分布式计算框架，它的处理速度比经典的分布式计算框架 Hadoop 快得多。Spark 的核心部分是由 Scala 实现的。Spark 对于处理迭代运算非常有效，而分布式机器学习算法经常需要迭代运算，因此 Spark 能很好地与机器学习结合在一起。

本书共 10 章，介绍了如何使用 Scala 在 Spark 平台上实现机器学习算法，其中 Scala 的版本为 2.11.7，Spark 采用基于 Hadoop 2.6 的版本，这些都是比较新的版本。本书从数据分析师怎么开始数据分析入手，介绍了数据驱动过程和 Spark 的体系结构；通过操作 Spark MLlib 库，介绍了机器学习的基本原理及 MLlib 所支持的几个算法；接着介绍了 Scala 如何表示和使用非结构化数据，以及与图相关的话题；再接着介绍了 Scala 与 R 和 Python 的集成；最后介绍了一些特别适合 Scala 编程的 NLP 常用算法及现有的 Scala 监控解决方案。总之，本书非常适合从事分布式机器学习的数据工作者，使用书中提供的大量针对性编程例子，可提高工程实战能力。

本书的第 1～3 章和第 7 章由重庆工商大学计算机科学与信息工程学院刘波博士翻译；第 4～6 章和第 8～10 章由重庆工商大学计算机科学与信息工程学院罗棻翻译。同时，刘波博士负责全书的技术审校工作。

翻译本书的过程也是译者不断学习的过程。为了保证专业词汇翻译的准确性，我们在翻译过程中查阅了大量相关资料。但由于时间和能力有限，书中内容难免出现差错。若有问题，读者可通过电子邮件（liubo7971@163.com; luofcn@163.com）与我们联系，欢迎一

起探讨，共同进步。并且，我们也会将最终的勘误信息公布在 http://www.cnblogs.com/ml-cv/ 上。

本书的顺利出版还要特别感谢机械工业出版社华章公司的编辑在翻译过程中给予的帮助！

本书的翻译也得到如下项目资助：（1）国家自然科学基金一般项目，非同步脉冲神经膜系统研究，项目号：61502063；（2）重庆市检测控制集成系统工程实验室新技术新产品开放课题，基于图像内容的目标检测算法及应用研究，项目号：KFJJ2016042。

Preface 前　言

这是一本关于机器学习的书，它以 Scala 为重点，介绍了函数式编程方法以及如何在 Spark 上处理大数据。九个月前，当我受邀写作本书时，我的第一反应是：Scala、大数据、机器学习，每一个主题我都曾彻底调研过，也参加了很多的讨论，结合任何两个话题来写都具有挑战性，更不用说在一本书中结合这三个主题。这个挑战激发了我的兴趣，于是就有了这本书。并不是每一章的内容都像我所希望的那样圆满，但技术每天都在快速发展。我有一份具体的工作，写作只是表达我想法的一种方式。

下面先介绍机器学习。机器学习经历了翻天覆地的变换；它是由人工智能和统计学发展起来的，于 20 世纪 90 年代兴起。后来在 2010 年或稍晚些时候诞生了数据科学。数据科学家有许多定义，但 Josh Wills 的定义可能最通俗，我有幸在 Cloudera 工作时和他共事过。这个定义在图 1 中有具体的描述。虽然细节内容可能会有争议，但数据科学确实是几个学科的交叉，数据科学家不一定是任何一个领域的专家。据 Jeff Hammerbacher（Cloudera 的创始人，Facebook 的早期员工）介绍，第一位数据科学家工作于 Facebook。Facebook 需要跨学科的技能，以便从当时大量的社交数据中提取有价值的信息。虽然我自称是一个大数据科学家，但我已经关注这个交叉领域很久了，以至于有太多知识出现混淆。写这本书就是想使用机器学习的术语来保持对这些领域的关注度。

图 1　数据科学家的一种可能定义

最近，在机器学习领域出现了另一个被广泛讨论的话题，即数据量击败模型的复杂度。在本书中可以看到一些 Spark MLlib 实现的例子，特别是 NLP 的 word2vec。机器学习模型可以更快地迁移到新环境，也经常击败需要数小时才能构建的更复杂的模型。因此，机器学习和大数据能够很好地结合在一起。

最后也很重要的一点是微服务的出现。作者在本书中花了大量的篇幅介绍机器和应用程序通信，所以会很自然地提及 Scala 与 Akka actor 模型。

对于大多数程序员而言，函数式编程更多是关于编程风格的变化，而不是编程语言本身。虽然 Java 8 开始有来自函数式编程的 lambda 表达式和流，但是人们仍然可以在没有这些机制的情况下编写函数式代码，甚至可以用 Scala 编写 Java 风格的代码。使得 Scala 在大数据世界中名声鹊起的两个重要思想是惰性求值和不可变性，其中惰性求值可大大简化多线程或分布式领域中的数据处理。Scala 有一个可变集合库和一个不可变集合库。虽然从用户的角度来看它们的区别很小，但从编译器的角度来看，不变性大大增加了灵活性，并且惰性求值能更好地与大数据相结合，因为 REPL 将大多数信息推迟到管道的后期处理，从而增加了交互性。

大数据一直备受关注，其主要原因是机器产生的数据量大大超越了人类在没有使用计算机以前的数量。Facebook、Google、Twitter 等社交网络公司已经证明专门用于处理大数据的工具（如 Hadoop、MapReduce 和 Spark）可以从这些数据块中提取丰富的信息。

本书后面将介绍关于 Hadoop 的内容。最初它能在廉价硬件上处理大量的信息，因为当时传统的关系数据库不能处理这样的信息（或能处理，但是代价过高）。大数据这个话题太大了，而 Spark 才是本书的重点，它是 Hadoop MapReduce 的另一个实现，Spark 提高了磁盘上持久化保存数据的效率。通常认为使用 Spark 有点贵，因为它消耗更多的内存，要求硬件必须更可靠，但它也更具交互性。此外，Spark 使用 Scala 工作（也可以使用 Java 和 Python 等），但 Scala 是主要的 API 语言。因此 Spark 用 Scala 在数据管道的表达方面有一定的协同性。

本书主要内容

第 1 章介绍数据分析师如何开始数据分析。除了允许用户使用新工具查看更大的数据集以外，该章并没有什么新东西。这些数据集可能分布在多台计算机上，但查看它们就像在本地机器上一样简单。当然，不会阻止用户在单个机器上顺序执行程序。但即使如此，作者写作的这个笔记本电脑也有四个核，可同时运行 1377 个线程。Spark 和 Scala（并行集合）允许用户透明地使用整个设备，有时并没有显式指定需要并行运行。现代服务器可对 OS 服务使用多达 128 个超线程。该章将展示如何使用新工具来进行数据分析，并用它来研究以前的数据集。

第 2 章介绍在 Scala/Spark 之前一直存在的数据驱动过程，也会介绍完全数据驱动的企

业，这类企业通过多台数据生成机器的反馈来优化业务。大数据需要新的技术和架构来适应新的决策过程。该章借鉴了一些学术资料来阐述数据驱动型业务的通用架构。在这种架构下，大多数工人的任务是监控和调整数据管道。

第 3 章重点介绍 Spark 的体系结构，它是前面提及的 Hadoop MapReduce 的替代者（或补充）。该章还将特别介绍 MLlib 所支持的几个算法。虽然这是一个崭新的话题，但许多算法都对应着各种实现。该章将给出一些例子，比如怎样运行 org.apache.spark.mllib 包中标准的机器学习算法。最后介绍 Spark 的运行模式及性能调整。

第 4 章介绍机器学习的原理，虽然 Spark MLlib 的内容可能会不断变化，但这些原理是不会变的。监督学习和无监督学习是经典的机器学习算法，对大多数数据而言，它们对数据按行进行操作。该章是每一本机器学习书的经典部分，但作者增加了一些知识点，使其围绕 Scala/Spark 来介绍监督学习和无监督学习。

第 5 章引入回归和分类，这是机器学习算法的另一个经典内容。虽然分类算法可以用来做回归，回归算法也可以用于分类，但它们仍然是两种不同的算法。该章通过具体的算法展示回归和分类的实际例子。

第 6 章介绍社交数据的新特性。使用非结构化数据需要新的技术和格式，该章将详细介绍显示、存储以及改进这类数据的方法。Scala 在这里成为了一个大赢家，因为它天生具备处理数据管道中复杂数据结构的能力。

第 7 章介绍图，传统按行存储的数据库系统很难处理这类信息。最近图数据库也再次流行起来。该章将介绍两个不同的库：一个是 Assembla 的 Scala 图，它对图的表示和理解都非常方便；另一个是 Spark 的图类，并在其基础上实现了几个图算法。

第 8 章介绍与 Scala 相关的内容，但许多人因为太谨慎了而不愿意放弃他们以前所使用的库。该章将演示如何透明地引用以 R 和 Python 编写的遗留代码，这是作者经常听到的要求。简单地说，这里有两种运行机制可以满足这类需求：一种是使用 Unix 的管道；另一种是在 JVM 中启动 R 或 Python。

第 9 章介绍自然语言处理，即如何处理人机交互，以及计算机如何理解人类的这种非标准沟通方式。该章将重点介绍 Scala 为自然语言处理、主题关联以及处理大量文本信息（Spark）所提供的几个工具。

第 10 章介绍通常如何开发数据管道，人们怎样使用和调试这些管道。监控不仅对数据管道的终端用户非常重要，而且对寻求优化运行方案或进一步做设计的开发人员来说也非常重要。该章介绍用于监控系统和分布式机器集群的标准工具，以及如何设计一个钩子服务，以便在不附加调试器的情况下查看其功能。该章也讨论了新出现的统计模型监控。

本书所需的工具

本书所使用的工具都是开源软件。首先是 Java，可以从 Oracle 的 Java 下载页面下载

它。读者必须接受安装许可，并为你的平台选择合适的映像。不要使用 OpenJDK，它与 Hadoop/Spark 的兼容性不好。

其次是 Scala。如果读者使用 Mac，建议安装 Homebrew：

```
$ ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/
install/master/install)"
```

读者还需要使用多个开源包。为了安装 Scala，请运行 brew install scala。在 Linux 平台上安装需要从 http://www.scala-lang.org/download/site 下载合适的 Debian 或 RPM 软件包。本书使用的版本是 2.11.7。

Spark 发行版可以从 http://spark.apache.org/downloads.html 上下载。本书使用预构建的 Hadoop 2.6（或更高版本）的映像。因为 Hadoop 是以 Java 编写的，只需要解压，然后运行 bin 子目录中的脚本。

R 和 Python 的包可分别从站点 http://cran.r-project.org/bin 和 http://python.org/ftp/python/$PYTHON_VERSION/Python-$PYTHON_VERSION.tar.xz 上获得。还有文档介绍具体如何配置它们。本书使用的 R 版本是 3.2.3，Python 的版本为 2.7.11。

本书面向的读者

想要提高实战技能的数据科学家，通过本书可以学习使用大数据的例子；想学习从大数据中有效地提取可靠信息的数据分析师；想超越现有的界限，成为数据科学家的统计师。

本书注重动手操作，除了少数几个例子有深入的介绍以外，本书不会深入地介绍数学证明。但本书会尽力提供代码示例和技巧，使读者可以尽快开始使用标准算法库。

下载示例代码

本书的代码包放在 GitHub 上，网址为 https://github.com/PacktPublishing/Mastering-Scala-Machine-Learning。

下载本书的彩色图片

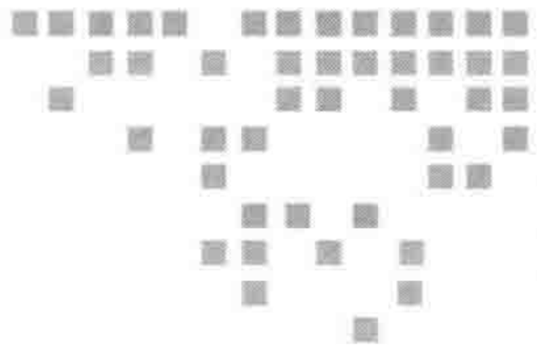
我们还为读者提供了一个 PDF 文件，其中包含本书中使用的截图 / 图的彩色图片。彩色图片将帮助读者更好地了解输出的变化。读者可以从 https://www.packtpub.com/sites/default/files/downloads/MasteringScalaMachineLearning_ColorImages.pdf 上下载此文件。

致谢

我曾有几次都想写一本书，当 Packt 在我 50 岁生日之前给我打电话时，我几乎立马就同意了。Scala？机器学习？大数据？这三者如何组合才能做到既容易理解，其主题又很有市场的推广性？为了把我的想法转换成文字，随之而来的是 8 个月的熬夜。实际上，这个过程让我发现我的身体每天需要至少三个小时的睡眠。总的来说，这个经历是完全值得的。我真心感激身边每个人的帮助，首先是我的家人，他们陪伴我度过许多不眠之夜，也容忍我对家庭暂时缺少关爱。

我想感谢我的妻子，因为我经常写作到深夜而让她承担了很多额外的家庭琐事。我知道这是非常不容易的。我还要感谢编辑，特别是 Samantha Gonsalves，他不仅时时叮咛我要按时完成任务，也给我非常多的好建议，还忍受着我的拖延。尤其要感谢在 E8 Security 产品发布的几个关键阶段顶替我的同事（我们一起做了 GA，在这段时间至少发行了好几个版本）。我们的很多想法都渗透到了 E8 产品中。我要特别感谢 Jeongho Park，Christophe Briguet，Mahendra Kutare，Srinivas Doddi 和 Ravi Devireddy。感谢 Cloudera 公司所有同事的反馈和讨论，特别是 Josh Patterson，Josh Wills，Omer Trajman，Eric Sammer，Don Brown，Phillip Zeyliger，Jonathan Hsieh 等。最后，我要感谢我的博士生导师 Walter A. Harrison，Jaswinder Pal Singh，John Hennessy 和 Daphne Koller，是他们将我带入技术和创新的世界。

目　录 *Contents*



第 1 章 Chapter 1

探索数据分析

在本书深入研究复杂的数据分析方法之前，先来关注一些基本的数据探索任务，这些任务几乎会占据数据科学家 80%～90% 的工作时间。据估计，每年仅仅是数据准备、清洗、转换和数据聚合就有 440 亿美元的产值（*Data Preparation in the Big Data Era* by Federico Castanedo; Best Practices for Data Integration, O' Reilly Media, 2015）。即便如此，人们最近才开始把更多的时间花费在如何科学地开发最佳实践，以及为整个数据准备过程建立文档、教学材料的良好习惯上，这是一件令人惊讶的事情（*Beautiful Data: The Stories Behind Elegant Data Solutions*, edited by Toby Segaran and Jeff Hammerbacher, O' Reilly Media, 2009；*Advanced Analytics with Spark: Patterns for Learning from Data at Scale* by Sandy Ryza et al., O' Reilly Media, 2015）。

很少有数据科学家会对数据分析的具体工具和技术看法一致，因为有多种方式可进行数据分析，从 UNIX 命令行到使用非常流行的开源包，或商业的 ETL 和可视化工具等。本章重点介绍在笔记本电脑上如何通过 Scala 进行函数式编程。后面的章节会讨论如何利用这些技术在分布式框架 Hadoop/Spark 下进行数据分析。

那函数式编程有什么用呢？ Spark 用 Scala 开发是有原因的。函数式编程的很多基本原则（比如惰性求值、不变性、无副作用、列表推导式和单子（monad）），在分布式环境下做数据处理都表现得很好，特别是在大数据集上做数据准备和转换等任务时更是如此。也可在 PC 或笔记本上使用这些技术。通过笔记本电脑连接到分布式存储 / 处理集群就可处理多达数十 TB 的超级数据集。可以一次只涉及一个主题或关注一个领域，但通常进行数据采样或过滤时，不必考虑分区是否合适。本书使用 Scala 作为基本工具，必要时也会采用其他工具。

从某种意义上讲，Scala 能实现其他语言所能实现的一切功能。Scala 从根本上讲是一

种高级语言，甚至可称其为脚本语言。Scala 有自己的容器，并且实现了一些基本的算法，这些功能已经通过大量的应用程序（比如 Java 或 C++）和时间的测试，程序员不必关心数据结构和算法实现的底层细节。本章也只会关注如何用 Scala/Spark 来实现高级任务。

本章会涉及如下主题：

- 安装 Scala
- 学习简单的数据挖掘技术
- 学习如何下采样（downsample）原始数据集来提高效率
- 探讨在 Scala 上实现基本的数据转换和聚合
- 熟悉大数据处理工具，比如 Spark 和 Spark Notebook
- 通过编程实现对数据集的简单可视化

1.1 Scala 入门

如果已经安装了 Scala，可以跳过本节。可以从 http://www.scala-lang.org/download/ 下载最新版本的 Scala，本书的 Scala 版本为 2.11.7，操作系统为 Mac OS X El Capitan 10.11.5。读者可以选择自己喜欢的版本，不过可能会遇到与其他包（如 Spark）的兼容性问题。开源软件的一个通病就是所采用的技术可能会滞后几个版本。


提示 大多数情况需要确保所下载的版本和推荐的版本完全一致。因为不同版本间的差异会导致隐蔽的错误，由此带来漫长的调试过程。

如果已经正确安装 Scala，输入 scala 之后就可以看到与下面类似的信息：

```
[akozlov@Alexanders-MacBook-Pro ~]$ scala
Welcome to Scala version 2.11.7 (Java HotSpot(TM) 64-Bit Server VM, Java
1.8.0_40).
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```

这是 Scala 的一个 REPL 环境（read-evaluate-print-loop，读取 – 求值 – 输出 – 循环）提示符。虽然 Scala 程序是可以编译的，但本章的内容会在 REPL 中运行，这是因为本章只专注于交互性，这种交互性有时可能会出现一些异常。:help 命令会给出一些 REPL 环境中的实用工具（留意开头的冒号）。

1.2 去除分类字段的重复值

请准备好数据集和电脑。为了方便起见，本书已经提供了一些关于点击流（clickstream）

数据的样本，它们是经过预处理过的，在 https://github.com/alexvk/ml-in-scala.git 上可以找到这些数据。chapter01/data/clickstream 文件夹中包含了时间戳、会话编号（session ID），以及在调用时的一些额外事件信息（比如 URL、类别信息等）。首先要对数据集的各个列做一些变换，以此得到数据的分布情况。

图 1-1 给出了在命令行中执行 gzcat chapter01/data/clickstream/clickstream_sample.tsv.gz | less-U 所得到的结果。列之间用 tab 键（^I）隔开。读者可能会注意到，许多值都空缺了，许多现实应用中的大数据集都是这样。数据的第一列是时间戳，文件包含了复杂的数据（比如数组（array）、结构（struct），以及映射（map）），这也是大数据集的另一个特征。

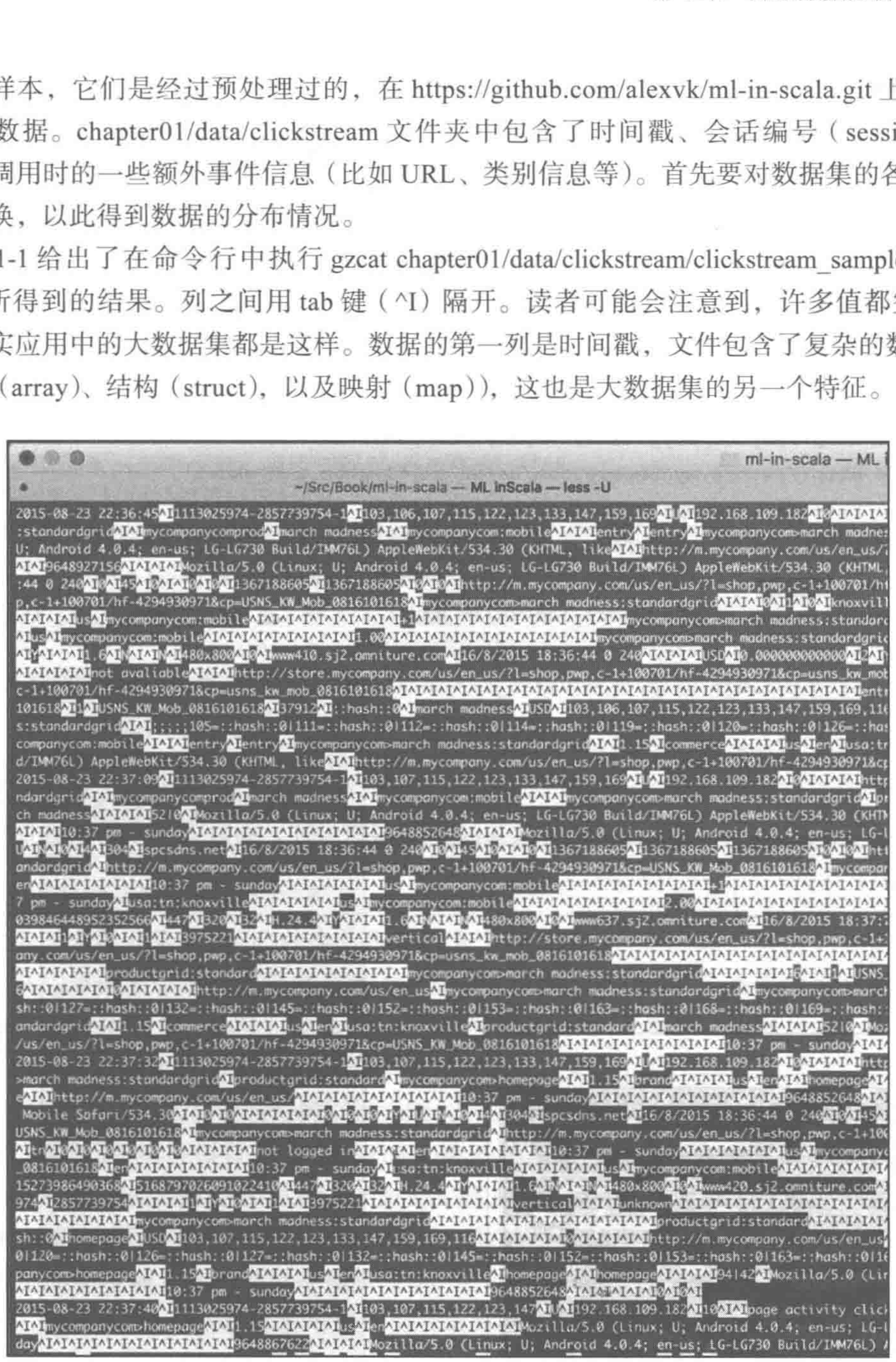


图 1-1　使用 Unix 的 less-U 命令后，clickstream 文件得到的输出

Unix 提供了一些工具来分析数据。less、cut、sort 和 uniq 大概是文本处理中最常用的命令行工具。awk、sed、perl 和 tr 可以做更复杂的转换和提取操作。

幸运的是，Scala 允许在 REPL 中透明地使用命令行工具来做转换：