

HZ BOOKS
华章科技



著名计算机科学家兼统计学家撰写，Amazon五星级畅销书，R语言领域公认经典著作。

从纯编程角度系统讲解R语言的数据结构、编程结构、语法、TCP/IP网络编程、并行计算、代码调试、程序性能优化、编程技巧以及R语言与其他语言的接口。

The Art of R Programming

A Tour of Statistical Software Design

R语言编程艺术

(美) Norman Matloff 著

陈堰平 邱怡轩 潘岚锋 等译



机械工业出版社
China Machine Press

013043440

TP312
941

The Art of R Programming
A Tour of Statistical Software Design

R语言编程艺术

(美) Norman Matloff 著

陈堰平 邱怡轩 潘岚锋 熊熹 译
林宇 严紫丹 程豪 审校



机械工业出版社
China Machine Press

TP312
941

0000000000

图书在版编目 (CIP) 数据

R 语言编程艺术 / (美) 麦特洛夫 (Matloff, N.) 著; 陈堰平, 等译. —北京: 机械工业出版社, 2013.6
(华章程序员书库)

书名原文: The Art of R Programming: A Tour of Statistical Software Design

ISBN 978-7-111-42314-0

I. R… II. ①麦… ②陈… III. 程序语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2013) 第 088547 号

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问 北京市展达律师事务所

本书版权登记号: 图字: 01-2012-2102

本书是 R 语言领域公认的经典著作, 由著名计算机科学家兼统计学家撰写, Amazon 五星级畅销书。它是一本面向 R 语言开发者的纯编程类书籍, 不需要读者具备统计学基础, 从编程角度而非统计学角度系统讲解了 R 语言的数据结构、编程结构、语法、TCP/IP 网络编程、并行计算、代码调试、程序性能优化、编程技巧以及 R 语言与其他语言的接口等所有与 R 编程相关的知识, 几乎面面俱到。本书的实用性也非常强, 44 个精选的扩展案例, 充分展示了 R 语言在数据处理和统计分析方面的强大能力。

全书一共 16 章: 第 1 章介绍了学习 R 语言需要掌握的预备知识以及它的一些重要数据结构; 第 2~6 章详细讲解了 R 语言的主要数据结构, 包括向量、矩阵、数组、列表、数据框和因子; 第 7~13 章全面讲解了 R 语言的语法, 包括编程结构、面向对象特性、数学运算与模拟、输入与输出、字符串处理、绘图, 以及 R 语言的调试方法。第 14~16 章讲解了 R 语言编程的高级内容, 如执行速度和性能的提升、R 语言与 C/C++ 或 Python 的混合编程, 以及 R 语言的并行计算等。

Copyright © 2011 by Norman Matloff, Title of English-language original: The Art of R Programming: A Tour of Statistical Software Design, ISBN 978-1-59327-384-2, published by No Starch Press.

Simplified Chinese-language edition copyright©2013 by Beijing Huazhang Graphics & Information Co., China Machine Press.

No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or any information storage and retrieval system, without permission, in writing, from the publisher.

All rights reserved.

本书中文简体字版由 No Starch Press 授权机械工业出版社在全球独家出版发行。未经出版者书面许可, 不得以任何方式抄袭、复制或节录本书中的任何部分。

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: 吴 怡

三河市杨庄长鸣印刷装订厂印刷

2013 年 5 月第 1 版第 1 次印刷

186mm×240mm·20 印张

标准书号: ISBN 978-7-111-42314-0

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) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzsj@hzbook.com

译者序

R是一种用于统计计算与做图的开源软件，同时也是一种编程语言，它广泛应用于企业和学术界的数据分析领域，正在成为最通用的语言之一。由于近几年数据挖掘、大数据等概念的走红，R也越来越多地被人关注。截至本文完成之日，CRAN (<http://cran.r-project.org/>) 上共有4383个包，涉及统计、化学、经济、生物、医学、心理、社会学等各个学科。不同类型的公司，比如Google、辉瑞、默克、美国银行、洲际酒店集团和壳牌公司都在使用它，同时以S语言环境为基础的R语言由于其鲜明的特色，一出现就受到了统计专业人士的青睐，成为国外大学里相当标准的统计软件。

一直以来，国内外关于R语言的著作都是以统计学专业的视角来介绍R语言的，对R语言本身的特性讲解得并不详尽，而软件自带的官方文档又显得过于技术，不那么亲民。另一方面，很多接触R的朋友都来自非计算机专业，没有接受过编程训练，他们使用R的时候，编写出来的代码通常只能算是一条条命令的集合，面对更复杂的问题，常常束手无策。记得在某届R语言大会上，有位SAS阵营的朋友说，他看到演讲者所展示的代码里只有函数调用，没有编程的东西，所以他觉得R不能算一种编程语言。其实，他错了，此时你手里这本书，覆盖了其他大部分R语言图书没有涉及的编程主题。这本书就如同R语言的九阳神功秘籍，当神功练成，任督二脉一旦打通，再学习针对某一领域应用的函数或包就如庖丁解牛一般。顺便提一下，据微博上的小道消息，前面提到的那位朋友最近也开始学R了。

本书的特点表现在以下几个方面：

第一，对读者的统计学知识和编程水平要求并不高。与很多R语言书籍不同，这本书并不需要很深的统计学功底，它从纯语言的角度入手来讲解R。对于有一定编程经验却没什么统计学背景的人来说，读这本书会比较顺畅，读者就可以重点关注R语言的特性在数据分析方面的应用。在有的地方，作者也会提醒那些有其他语言编程经验的人应该注意R语言有什么不同之处。而对于没有编程经验又想使用R做数据分析的人来说，这本书也是学习编程的绝佳教材。

第二，专注于R语言编程。作者没有把这本书定位为菜谱式的手册，也不像有些R语言图书那样介绍完统计学某方面应用之后简单地把R语言代码摆出来。翻开这本书的目录，你几乎看不到统计学的术语。本书系统介绍了R语言的各种数据结构和编程结构、面向对象编程方法、socket网络编程、并行计算、代码调试、程序性能提升以及R语言与其他语言的接口等主题。书中也提到了不少编程的小技巧，这都是作者多年编程经验的总结。

第三，丰富的案例分析。作者Matloff教授是位计算机科学家，同时也是位统计学家，有多年的教学经验，也做过统计学方法论的顾问。除了正文中的例子之外，本书还有44个扩展案例，很多案例源自作者亲身参与过的咨询项目。虽然本书没有讲解任何统计模型，但是扩展案例都是和数据分析相关的，比如对鲍鱼数据的重新编码（第2章）、寻找异常值（第3章）、文本词汇索引（第4章）、学习中文方言的辅助工具（第5章）等。通过学习这些案例，读者不仅能学到R语言的每种概念如何运作，也会学到如何把这些概念组合到一起成为有用的程序。比如第10章介绍了socket网络编程之后，就用一个扩展案例讲解如何用socket实现并行计算，这为第16章详细讲解并行计算做好了铺垫。在很多案例里，作者讨论了好几种设计方案，并比较了这几种方案的不同之处，以回答“为什么这样做”，这对于缺少编程经验的人来说，是非常好的安排。

本书第1章简要介绍了R语言的几种数据结构和编程基础，其余章节可分为三大部分。

第一部分（第2~6章）详细介绍R的几种主要的数据结构：向量、矩阵、列表、数据框和因子等。对很多人来说，R复杂多变的数据结构真的是一只拦路虎。而本书从最简单的向量开始，一步一步引导读者认识并掌握各种数据结构。

第二部分（第7~13章）涉及编程方面：编程结构和面向对象特性、输入/输出、字符串处理以及绘图。值得一提的是第13章，这章主要讲解的是R语言的调试。很多朋友在实际工作中有这样的经历，你可能用了一个小时就写好代码，却用了一天的时间来调试。可是到目前为止还没有在其他图书上看到与R语言调试相关的内容，甚至也很少见到关于其他编程语言调试的图书。本书刚好填补了这方面的空白。如果读者仔细读完第13章，并实践其中的调试技巧，一定能事半功倍，也就能少熬点儿夜，有延长寿命的功效。本书的作者同时也著有《调试的艺术》（The Art of Debugging），相信他在R语言调试方面的功力也是相当深厚的。

第三部分（第14~16章）介绍的是更高级的内容，比如执行速度和性能的提升（第14章）、R语言与C/C++或Python混合编程（第15章）以及R语言并行计算（第16章），虽然最后一部分属于编程的高级内容，但如果读者从前往后一直学下来，随着能力的提高，也是可以读懂的。

本人从2007年开始接触R语言，那时候市面上几乎没有R语言方面的书籍。当时我关于R语言的所有信息几乎都是来自统计之都（<http://cos.name>）和谢益辉的博客（<http://yihui.name>）。2008年冬天，统计之都成功举办了“第一届中国R语言会议”，来自各地的R语言用户们齐聚一堂，交流心得。从那以后，每年的R语言会议都会在北京和上海举办。这几年来，统计之都的队伍也逐渐壮大，比如本书的其他三位主要译者：邱怡轩、潘岚锋和熊熹，当年他们参加R语言会议的时候还是人大统计学院大一、大二的学生，后来也成为R语言社区的领军人物。去年我们接到本书的翻译任务时，他们三人分别收到了美国普度大学、爱荷华州立大学以及明尼苏达大学的录取通知，现在已经在美国留学深造。希望有越来越多的人加入统计之都的大家庭，和大家一起成长，为中国统计事业的发展尽自己的一份力。

在翻译过程中，几位译者力求忠实于原文，但纠正了原书的几处错误，同时也兼顾中文表达的流畅，不过译文中可能仍有不当之处，欢迎读者予以指正。

除了本人以及前面提到的三位译者之外，统计之都的三位老朋友林宇、严紫丹和程豪也参与了本书部分章节的校审和初稿翻译，在此表示感谢。全书译文最后由本人统稿，如有错误之处，均由本人承担。

也感谢机械工业出版社的吴怡编辑，她给予了我们细心的帮助。

统计之都的图书出版栏目（网址是<http://cos.name/books/>）有本书的页面，读者可以在这里下载本书的数据和代码，也可以留言提问。

陈堰平
2013年3月于新华通讯社第三工作区

译者简介

本书三位主要译者都是统计之都（<http://cos.name>）的管理员、中国 R 语言会议理事会成员。

陈堰平毕业于中国人民大学统计学院，现任国家金融信息中心数据中心研发部负责人，从事指数编制、指数化投资、金融衍生品方面的工作，对金融数据分析有多年的研究，博客网址为 <http://yanping.me>。邱怡轩是普度大学统计系在读博士研究生，开发过 `rarpark`、`R2SWF`、`Layer` 等 R 语言程序包，博客网址为 <http://yixuan.cos.name/>。潘岚峰是爱荷华州立大学统计系在读博士研究生，开发过 R 语言程序包 `bignmf`。

前言

R是一种用于数据处理和统计分析的脚本语言，它受到由AT&T实验室开发的统计语言S的启发，且基本上兼容于S语言。S语言的名称代表统计学（statistics），用来纪念AT&T开发的另一门以一个字母命名的编程语言，这就是著名的C语言。后来一家小公司买下了S，给它添加了图形用户界面并命名为S-Plus。

由于R是免费的，而且有更多的人贡献自己的代码，R语言变得比S和S-Plus更受欢迎。R有时亦称为GNU S，以反映它的开源属性。（GNU项目是开源软件的一个重要集合。）

为什么在统计工作中用R

粤语有个词“又便又靚”，意思是“物美价廉”，R语言就是这样一种工具，为什么还要用别的呢？

R语言有许多优点：

- 它是广受关注的统计语言S在公众领域的实现，而且R/S已经是专业统计学家的实际标准语言。
- 在绝大多数情况下，它的功能不亚于甚至优于商业软件，比如它有大量的函数、良好的可编程性、强大的绘图功能，等等。
- 在Windows、Mac、Linux等操作系统上都有相应的版本。
- 除了提供统计操作以外，R还是门通用编程语言，所以你可以用它做自动分析、创建新的函数来拓展语言的现有功能。
- 它结合了面向对象语言和函数式编程语言的特性。
- 系统在两次会话之间可以保存数据集，所以不需要每次重新加载数据集。R还可以保存历史命令。
- 因为R是开源软件，所以很容易从用户社区获得帮助。另外，用户们贡献了大量的新函数，其中很多用户都是杰出的统计学家。

我必须事先提醒你，最好直接在终端窗口输入命令并提交给R，而不是在GUI里用鼠标点击菜单，并且大多数R用户都不用GUI。这并不是说R不能图形化操作。相反，它有很多工具可以生成实用、美观的图形，不过这些工具是用在系统输出方面，比如画图，而不是用在输入方面。

如果你离不开GUI，则可以选用一种免费的GUI，它们是为R开发的，比如下面几种开源的或免费的工具：

- RStudio, <http://www.rstudio.org/>
- StatET, <http://www.walware.de/goto/statet/>
- ESS (Emacs Speaks Statistics), <http://ess.r-project.org/>
- R Commander: John Fox, “The R Commander: A Basic-Statistics Graphical Interface to R,” *Journal of Statistical Software* 14, no. 9 (2005):1–42.
- JGR (Java GUI for R), <http://cran.r-project.org/web/packages/JGR/index.html>

前三种软件，RStudio、StatET和 ESS属于集成开发环境（Integrated Development Environments, IDE），更多地是为编程设计的。StatET和ESS则为R程序员分别提供了针对著名的Eclipse 和Emacs环境的IDE。

在商业软件中，另一种IDE出自Revolution Analytics公司，一家提供R语言服务的公司（<http://www.revolutionanalytics.com/>）。

因为R是一种编程语言而不是各种不相关联的命令汇总，你可以把几个命令组合起来使用，每条命令用前一条命令的输出作为输入。（Linux用户可能会认出：这类似于用管道将shell命令串联起来。）这种组合R函数的能力带来了巨大的灵活性，如果使用恰当，功能会非常强大。

下面是个简单的例子，请看这条命令：

```
nrow(subset(x03, z == 1))
```

首先，subset()函数针对数据框x03提取出变量z（取值为1）的所有记录，得到一个新的数据框，再把这个新数据框代入nrow()函数。这个函数计算数据框的行数。这行命令的最终效果是给出原数据框中z=1的记录个数。

之前提到过面向对象编程和函数式编程这两个术语。这两个主题会激起计算机科学家的兴趣，尽管它们对大多数读者来说可能有点陌生，但是它们跟任何使用R做统计编程的人都有关系。下面概述这两个主题。

面向对象编程

面向对象的优点可以用例子来解释，例如回归模型。当你用SAS、SPSS等其他统计软件做回归分析时，你会在屏幕上看到一大堆的输出结果。与之相反，如果在R里调用回归函数lm()，函数会返回一个包含所有结果的对象，对象里含有回归系数的估计、估计值的标准差、残差等。接下来你可以用编程的方式挑选对象里需要的部分并提取出来。

你会看到通过R的方式可使编程变得更容易，部分因为它提供了访问数据的一致性。这种一致性源于R是多态的，即一个函数可以应用于不同类型的输入，函数在运行过程中会选择适当的方式来处理。这样的函数称为泛型函数。（如果你是C++程序员，肯定见过类似的概念虚函数。）

例如plot()函数，如果你把它应用到一列数上，会得到一幅简单的图。但是如果把它应用到某个回归分析的输出结果中，会得到关于回归分析多个方面的一整套图形。当然，你只能在R生成的对象上使用plot()函数。这样也好，这意味着用户需要记的命令更少了！

函数式编程

避免显式迭代是R语言的一个常见话题，这对于函数式编程语言来说是很典型的问题。你可以利用R的函数特性把迭代行为表达成隐式的，而不是用循环语句。这可以让代码执行起来更有效率，当R运行在大数据集上时运行时间会相差很大。

正如你看到的，R语言函数式编程的属性有许多优点：

- 更清晰，更紧凑的代码。
- 有潜力达到更快的执行速度。
- 减少了调试的工作量，因为代码更简单。
- 容易转化为并行编程。

本书的读者对象

许多人以特定的方式使用R——直方图、回归分析或者其他涉及统计运算的任务。不过本书针对的是那些希望用R开发软件的读者。本书的目标读者从专业软件开发人员，到只在大学修过编程课、为了完成特定任务而写R代码的人。（统计学知识一般不是必需的。）

以下几类人可能会从本书受益：

- 受雇于某个需要定期制作统计报告的机构，比如医院、政府机关等，为此需要开发专用程序。
- 开发统计学方法论的学术研究人员，所研究的方法论要么是全新的，要么就是结合了现有的方法并将其整合到一起，这些需要编程来实现，让学术界里更多的人能够使用。
- 在市场营销、诉讼支持、新闻、出版等领域工作，需要通过编码来制作复杂图形以实现数据可视化的专家。
- 有软件开发经验，参与的项目涉及统计分析的专业程序员。
- 学习统计计算课程的学生。

因此，本书不是R包中各种统计方法的纲要，其实本书更侧重于编程，覆盖了大部分R语言图书没有涉及的与编程相关的主题，我甚至是围绕编程主题展开论述的。下面是一些具体

的例子：

- “扩展案例”展示完整的、特定用途的函数，而不是针对某个数据集的独立代码片段。你可能发现其中有些函数对你平常用R语言进行工作很有帮助。通过学习这些案例，你不仅能学到R语言的基本功能如何运作，也会学到如何把它们组合成有用的程序。在很多案例里，我讨论了其他设计方案，以回答“为什么我们这样做？”。
- 内容上符合程序员的思维习惯。例如，在讨论数据框的时候，我不仅表明数据框是一种R的列表，也指出这一事实对编程的潜在影响。我在适当的时候加入R语言与其他语言的比较，为那些刚好了解其他语言的人提供参考。
- 在任何语言里，调试对于编程都非常关键，而在其他大多数R语言书籍中却鲜有涉及。本书用了一章的篇幅来介绍调试技巧，用“扩展案例”方法真刀真枪地展示如何调试实际工作中的程序。
- 如今，多核计算机甚至普及到普通家庭，图形处理单元（GPU）已经悄然在科学计算界引发了一场革命。越来越多的R应用涉及非常大量的计算，并行处理已经成为R程序员面临的主要课题。所以本书用一章的篇幅讨论这个主题，同样给出技术细节和扩展案例。
- 本书单独用一章介绍如何利用R内部行为以及其他工具的优势来加速R代码。
- 用一章来讨论R语言与其他语言（如C和Python）的接口，用扩展案例展示了应用方法，同时也介绍了调试的技巧。

我的学术背景

我绕了一个圈才走进R的世界。

我完成了抽象概率论领域的博士论文之后，在担任统计学教授的前几年，我教过学生、做过研究、也做过统计方法论的顾问。我是加州大学戴维斯分校统计系的创建者之一。

后来我去了这个学校的计算机科学系，在那里我度过了职业生涯的大部分时间。我研究并行编程、网络流量、数据挖掘、磁盘系统性能以及其他领域。我在计算机科学领域的教学和研究大多都涉及统计学。

所以我既有纯粹的计算机科学家的视角，也有作为统计学家和统计研究者的视角。我希望这样的综合视角能够使我更好地解读R语言，使本书更具实用价值。

致 谢

本书很大程度上得益于很多人的帮助和支持。

首先，也是最重要的，我必须感谢技术审稿人Hadley Wickham先生，他的成名作是ggplot2和plyr这两个包。我曾向No Starch出版社推荐过Hadley，因为除了这两个包之外，他开发的其他包在CRAN（R用户贡献的代码库）上也备受欢迎，可说是经验丰富。正如我期待的那样，Hadley的很多评论为本书增色不少，尤其是他对某些代码示例的评论，通常他都这样开头：“我在想，如果你这么写会怎么样……”。有时这些评论会导致原本只带有一两个版本代码的例子变得要用两三种甚至更多种不同方式来实现编程目的，这样可以比较不同方法的优点和缺点，我相信读者会因此受到启发。

非常感谢Jim Porzak，他是湾区R用户小组（Bay Area useR Group，BARUG，网址<http://www.bay-r.org/>）的联合创始人，在我写这本书时他曾多次鼓励我。说起BARUG，我必须感谢Jim和另一位联合创始人Mike Driscoll，感谢他们创建了这个充满活力而又富有启发性的论坛。在BARUG，介绍R语言精妙应用的演讲者们经常让我感觉写这本书是个很有价值的项目。BARUG也得益于Revolution Analytics公司的资助以及该公司员工David Smith和Joe Rickert付出的时间、精力，以及奇妙的想法。

Jay Emerson和Mike Kane，CRAN上备受赞誉的bigmemory包的作者，他们通读了第16章的早期文稿，并给出了极富价值的评论。

John Chambers（S语言的缔造者，而S语言是R语言的前身）和Martin Morgan提供了关于R内核的建议，这对我在第14章讨论R的性能问题有很大帮助。

7.8.4节涉及了一个在编程社区很有争议的主题——全局变量的使用。为了有一个更广阔的视角，我征求了几位专家的意见，特别是R核心小组的成员Thomas Lumley和加州大学戴维斯分校计算机科学学院的Sean Davis。当然，这并不意味着他们认可了我在这一节的观点，不过他们的评论非常有用。

在本项目的前期，我写了份非常粗糙的（也是非常不完整的）草稿以供公众评论，后来Ramon Diaz-Uriarte、Barbara F. La Scala、Jason Liao以及我的老朋友Mike Hannon给了我很有帮助的反馈。我的女儿Laura，一名工科学生，阅读了前面部分章节并给出了一些建议，使得本书得以完善。

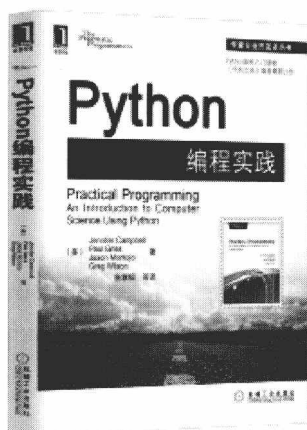
我自己的CRAN项目以及与R相关的研究（有些成为了本书的示例）得益于许多人的建议、反馈和（或）鼓励，特别是Mark Bravington、Stephen Eglen、Dirk Eddelbuett、Jay Emerson、Mike Kane、Gary King、Duncan Murdoch和Joe Rickert。

R核心小组成员Duncan Temple Lang和我在同一个机构——加州大学戴维斯分校（UCD）。尽管我们在不同的系，以前也没有太多接触，但是这本书也得益于他在这个校园。他帮助UCD创造了一种广泛认可R的文化氛围，这让我能够很容易地向系里证明我用大量的时间写这本书是有价值的。

这本书是我跟No Starch出版社合作的第二个项目。当我决定写这本书的时候，很自然地想到去找No Starch出版社，因为我喜欢他们产品的这种不拘形式的风格、高度实用性和可接受的价格。感谢Bill Pollock同意这个项目，感谢编辑人员Keith Fancher和Alison Law以及自由编辑Marilyn Smith。

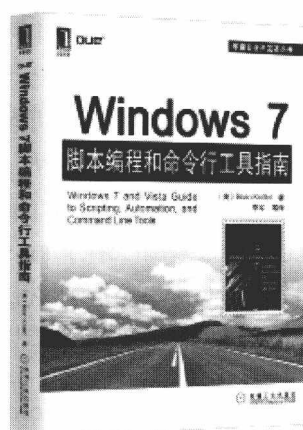
最后，但非常重要的是，我要感谢两位美丽、聪明、有趣的女人——我的妻子Gamis和前面提到的Laura，每次她们问我为什么如此埋头工作，我说“我正在写这本R书”，她们都会欣然接受。

推荐阅读



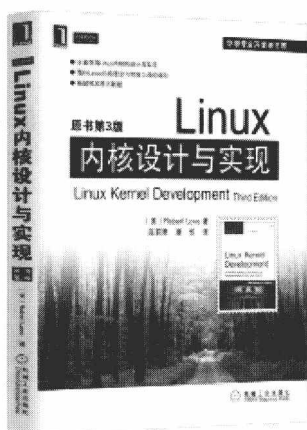
Python编程实践

作者: Jennifer Campbell ISBN: 978-7-111-36478-8 定价: 49.00元



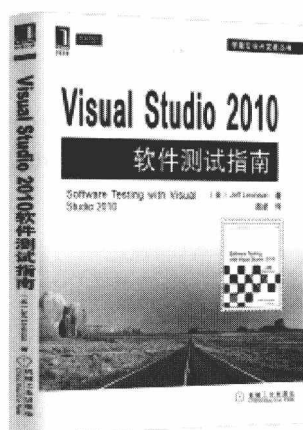
Windows 7脚本编程和命令行工具指南

作者: Brian Knittel ISBN: 978-7-111-35677-6 定价: 79.00元



Linux内核设计与实现 (原书第3版)

作者: Robert Love ISBN: 978-7-111-33829-1 定价: 69.00元



Visual Studio 2010软件测试指南

作者: Jeff Levinson ISBN: 978-7-111-35931-9 定价: 49.00元



北航

C1651837

目 录

译者序 前 言 致 谢

第1章 快速入门	1
1.1 怎样运行R	1
1.1.1 交互模式	1
1.1.2 批处理模式	2
1.2 第一个R会话	3
1.3 函数入门	5
1.3.1 变量的作用域	7
1.3.2 默认参数	8
1.4 R语言中一些重要的数据结构	8
1.4.1 向量, R语言中的战斗机	8
1.4.2 字符串	9
1.4.3 矩阵	9
1.4.4 列表	10
1.4.5 数据框	12
1.4.6 类	12
1.5 扩展案例: 考试成绩的回归分析	13
1.6 启动和关闭R	16
1.7 获取帮助	17
1.7.1 help()函数	18
1.7.2 example()函数	18
1.7.3 如果你不太清楚要查找什么	19
1.7.4 其他主题的帮助	20
1.7.5 批处理模式的帮助	21
1.7.6 互联网资源	21
第2章 向量	22
2.1 标量、向量、数组与矩阵	22

2.1.1 添加或删除向量元素	22
2.1.2 获取向量长度	23
2.1.3 作为向量的矩阵和数组	24
2.2 声明	24
2.3 循环补齐	25
2.4 常用的向量运算	26
2.4.1 向量运算和逻辑运算	26
2.4.2 向量索引	27
2.4.3 用:运算符创建向量	28
2.4.4 使用seq()创建向量	28
2.4.5 使用rep()重复向量常数	29
2.5 使用all()和any()	30
2.5.1 扩展案例: 寻找连续出现 1的游程	30
2.5.2 扩展案例: 预测离散值 时间序列	31
2.6 向量化运算符	34
2.6.1 向量输入, 向量输出	34
2.6.2 向量输入, 矩阵输出	36
2.7 NA与NULL值	37
2.7.1 NA的使用	37
2.7.2 NULL的使用	37
2.8 筛选	38
2.8.1 生成筛选索引	38
2.8.2 使用subset()函数筛选	40
2.8.3 选择函数which()	40
2.9 向量化的ifelse()函数	41
2.9.1 扩展案例: 度量相关性	42
2.9.2 扩展案例: 对鲍鱼数据集 重新编码	44

2.10 测试向量相等	46	5.1.1 访问数据框	85
2.11 向量元素的名称	47	5.1.2 扩展案例：考试成绩的 回归分析（续）	86
2.12 关于c()的更多内容	48	5.2 其他矩阵式操作	87
第3章 矩阵和数组	49	5.2.1 提取子数据框	87
3.1 创建矩阵	49	5.2.2 缺失值的处理	88
3.2 一般矩阵运算	50	5.2.3 使用rbind()和cbind()等 函数	89
3.2.1 线性代数运算	50	5.2.4 使用apply()	90
3.2.2 矩阵索引	51	5.2.5 扩展案例：工资研究	90
3.2.3 扩展案例：图像操作	52	5.3 合并数据框	92
3.2.4 矩阵元素筛选	55	5.4 应用于数据框的函数	95
3.2.5 扩展案例：生成协方差矩阵	57	5.4.1 在数据框上应用lapply() 和sapply()函数	95
3.3 对矩阵的行和列调用函数	58	5.4.2 扩展案例：应用Logistic模型	95
3.3.1 使用apply()函数	58	5.4.3 扩展案例：学习中文方言的 辅助工具	96
3.3.2 扩展案例：寻找异常值	60	第6章 因子和表	102
3.4 增加或删除矩阵的行或列	61	6.1 因子与水平	102
3.4.1 改变矩阵的大小	61	6.2 因子的常用函数	103
3.4.2 扩展案例：找到图中距离 最近的一对端点	63	6.2.1 tapply函数	103
3.5 向量与矩阵的差异	65	6.2.2 split()函数	105
3.6 避免意外降维	66	6.2.3 by()函数	106
3.7 矩阵的行和列的命名问题	68	6.3 表的操作	107
3.8 高维数组	68	6.3.1 表中有关矩阵和类似 数组的操作	109
第4章 列表	71	6.3.2 扩展案例：提取子表	111
4.1 创建列表	71	6.3.3 扩展案例：在表中寻找频数 最大的单元格	113
4.2 列表的常规操作	72	6.4 其他与因子和表有关的函数	114
4.2.1 列表索引	72	6.4.1 aggregate()函数	115
4.2.2 增加或删除列表元素	73	6.4.2 cut()函数	115
4.2.3 获取列表长度	75	第7章 R语言编程结构	116
4.2.4 扩展案例：文本词汇索引	75	7.1 控制语句	116
4.3 访问列表元素和值	78	7.1.1 循环	116
4.4 在列表上使用apply系列函数	79	7.1.2 对非向量集合的循环	119
4.4.1 lapply()和sapply()的使用	79	7.1.3 if-else结构	120
4.4.2 扩展案例：文本词汇索引（续）	80		
4.4.3 扩展案例：鲍鱼数据	82		
4.5 递归型列表	83		
第5章 数据框	85		
5.1 创建数据框	85		

7.2 算术和逻辑运算符及数值	121	8.1.1 扩展例子: 计算概率	161
7.3 参数的默认值	122	8.1.2 累积和与累积乘积	162
7.4 返回值	123	8.1.3 最小值和最大值	162
7.4.1 决定是否显式调用return ()	124	8.1.4 微积分	163
7.4.2 返回复杂对象	124	8.2 统计分布函数	164
7.5 函数都是对象	124	8.3 排序	165
7.6 环境和变量作用域的问题	127	8.4 向量和矩阵的线性代数运算	166
7.6.1 顶层环境	127	8.4.1 扩展示例: 向量叉积	169
7.6.2 变量作用域的层次	128	8.4.2 扩展示例: 确定马尔科夫链 的平稳分布	170
7.6.3 关于ls()的进一步讨论	131	8.5 集合运算	171
7.6.4 函数(几乎)没有副作用	131	8.6 用R做模拟	173
7.6.5 扩展案例: 显示调用框的函数	132	8.6.1 内置的随机变量发生器	173
7.7 R语言中没有指针	134	8.6.2 重复运行时获得相同的 随机数流	175
7.8 向上级层次进行写操作	136	8.6.3 扩展案例: 组合的模拟	175
7.8.1 利用超赋值运算符对非 局部变量进行写操作	136	第9章 面向对象的编程	177
7.8.2 用assign()函数对非局部 变量进行写操作	137	9.1 S3类	177
7.8.3 扩展案例: 用R语言实现 离散事件仿真	138	9.1.1 S3泛型函数	177
7.8.4 什么时候使用全局变量	145	9.1.2 实例: 线性模型函数 lm()中的OOP	178
7.8.5 闭包	147	9.1.3 寻找泛型函数的实现方法	179
7.9 递归	148	9.1.4 编写S3类	181
7.9.1 Quicksort的具体实现	149	9.1.5 使用继承	182
7.9.2 拓展举例: 二叉查找树	150	9.1.6 扩展示例: 用于存储上 三角矩阵的类	183
7.10 置换函数	155	9.1.7 扩展示例: 多项式回归程序	187
7.10.1 什么是置换函数	155	9.2 S4类	191
7.10.2 扩展案例: 可记录元素 修改次数的向量类	156	9.2.1 编写S4类	191
7.11 写函数代码的工具	158	9.2.2 在S4类上实现泛型函数	193
7.11.1 文本编辑器和集成开发环境	158	9.3 S3类和S4类的对比	193
7.11.2 edit()函数	158	9.4 对象的管理	194
7.12 创建自己的二元运算符	159	9.4.1 用ls()函数列出所有对象	194
7.13 匿名函数	159	9.4.2 用rm()函数删除特定对象	194
第8章 数学运算与模拟	161	9.4.3 用save()函数保存对象集合	195
8.1 数学函数	161	9.4.4 查看对象内部结构	196

9.4.5 exists()函数	197	第12章 绘图	224
第10章 输入与输出	198	12.1 创建图形	224
10.1 连接键盘与显示器	198	12.1.1 基础图形系统的核心: plot()函数	224
10.1.1 使用scan()函数	198	12.1.2 添加线条: abline()函数	225
10.1.2 使用readline()函数	200	12.1.3 在保持现有图形的基础上 新增一个绘图窗口	226
10.1.3 输出到显示器	201	12.1.4 扩展案例: 在一张图中绘制 两条密度曲线	227
10.2 读写文件	202	12.1.5 扩展案例: 进一步考察 多项式回归	228
10.2.1 从文件中读取数据框或矩阵	202	12.1.6 添加点: points()函数	231
10.2.2 读取文本文件	203	12.1.7 添加图例: legend()函数	231
10.2.3 连接的介绍	203	12.1.8 添加文字: text()函数	232
10.2.4 扩展案例: 读取PUMS普查 数据	204	12.1.9 精确定位: locator()函数	232
10.2.5 通过URL在远程计算机 上访问文件	208	12.1.10 保存图形	233
10.2.6 写文件	209	12.2 定制图形	233
10.2.7 获取文件和目录信息	210	12.2.1 改变字符大小: cex选项	233
10.2.8 扩展案例: 多个文件内容的和	211	12.2.2 改变坐标轴的范围: xlim和ylim选项	234
10.3 访问互联网	211	12.2.3 添加多边形: polygon()函数	235
10.3.1 TCP/IP概述	212	12.2.4 平滑散点: lowess()和 loess()函数	236
10.3.2 R中的socket	212	12.2.5 绘制具有显式表达式的函数	237
10.3.3 扩展案例: 实现R的并行计算	213	12.2.6 扩展案例: 放大曲线的一部分	237
第11章 字符串操作	216	12.3 将图形保存到文件	240
11.1 字符串操作函数概述	216	12.3.1 R图形设备	240
11.1.1 grep()	216	12.3.2 保存已显示的图形	241
11.1.2 nchar()	216	12.3.3 关闭R图形设备	241
11.1.3 paste()	217	12.4 创建三维图形	241
11.1.4 sprintf()	217	第13章 调试	243
11.1.5 substr()	217	13.1 调试的基本原则	243
11.1.6 strsplit()	217	13.1.1 调试的本质: 确认原则	243
11.1.7 regexpr()	218	13.1.2 从小处着手	243
11.1.8 gregexpr()	218	13.1.3 模块化的、自顶向下 的调试风格	244
11.2 正则表达式	218	13.1.4 反漏洞	244
11.2.1 扩展案例: 检测文件名的后缀	219		
11.2.2 扩展案例: 生成文件名	220		
11.3 在调试工具edtdbg中使用字符 串工具	221		