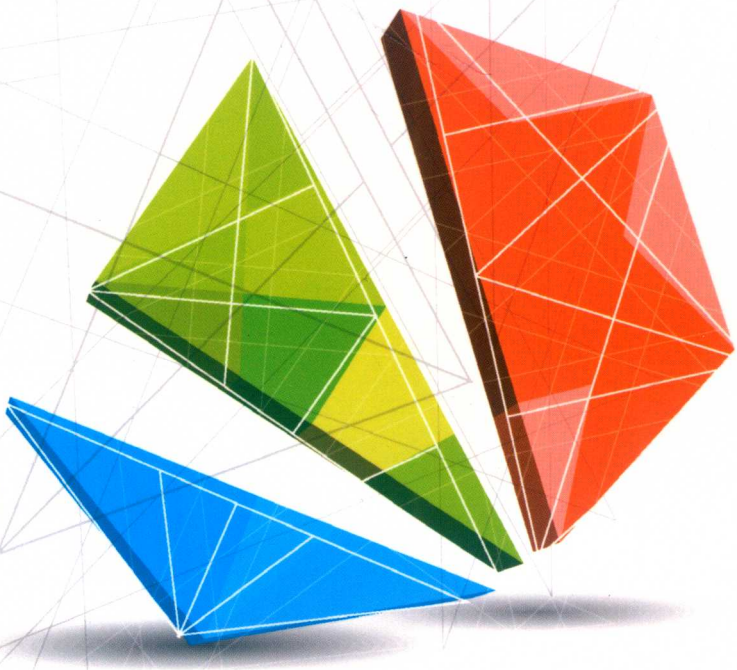
高等学校大数据技术与应用规划教材

R语言数据分析与挖掘

R YUYAN SHUJU FENXI YU WAJUE

杜 宾 钱亮宏 黄 勃 高永彬 编著



中国铁道出版社有限公司
CHINA RAILWAY PUBLISHING HOUSE CO., LTD.

高等学校大数据技术与应用规划教材

R语言数据分析与挖掘

杜　宾　钱亮宏　黄　勃　高永彬　编著

中国铁道出版社有限公司
CHINA RAILWAY PUBLISHING HOUSE CO., LTD.

内 容 简 介

本书从R语言的使用出发，在重点介绍R语言编程基础、操作、可视化、统计、高性能计算和机器学习的同时，注重实践能力的培养和数据分析与挖掘素质的全面提高。

本书分为统计分析基础和机器学习实践两部分，共12章，内容包括R语言概述、数据访问、数据操作、数据可视化、概率与分布、基本统计分析、回归分析、方差分析、大数据高性能计算、机器学习流程、有监督学习模型、无监督学习模型。本书的重点是让学生了解R语言数据分析与挖掘的基本技能和操作方法，并与数据分析与挖掘的典型方法、算法和应用场景结合。

本书内容丰富、体系新颖、结构合理、文字精练，适合作为普通高等院校信息类、管理类和数学统计类专业的R语言数据分析与挖掘课程的教材，也可作为数据科学行业相关从业人员的自学用书。

图书在版编目(CIP)数据

R语言数据分析与挖掘/杜宾等编著. —北京：中国铁道出版社有限公司，2019.7

高等学校大数据技术与应用规划教材

ISBN 978-7-113-25753-8

Ⅰ.①R… Ⅱ.①杜… Ⅲ.①程序语言-程序设计-高等学校-教材②数据处理-高等学校-教材③数据采集-高等学校-教材 Ⅳ.①TP312②TP274

中国版本图书馆CIP数据核字(2019)第081989号

书　　名：**R语言数据分析与挖掘**

作　　者：杜　宾　钱亮宏　黄　勃　高永彬

策　　划：曹莉群　　读者热线：(010)63550836

责任编辑：包　宁

封面设计：穆　丽

责任校对：张玉华

责任印制：郭向伟

出版发行：中国铁道出版社有限公司(100054，北京市西城区右安门西街8号)

网　　址：http://www.tdpress.com/51eds/

印　　刷：北京铭成印刷有限公司

版　　次：2019年7月第1版　2019年7月第1次印刷

开　　本：787 mm×1 092 mm　1/16　印张：22.75　字数：531千

书　　号：ISBN 978-7-113-25753-8

定　　价：59.80元

前 言

PREFACE

随着信息技术的普及和应用，各行各业产生了大量的数据，人们持续不断地探索处理这些数据的方法，以期最大限度地从中挖掘有用信息。面对如潮水般不断增加的数据，人们不再满足于数据的查询和统计分析，而是期望从数据中提取信息或者知识为决策服务。数据挖掘技术突破数据分析技术的种种局限，结合统计学、数据库、机器学习等技术解决从数据中发现新的信息并辅助决策这一难题，是正在飞速发展的前沿学科。近年来，随着教育部“新工科”建设的不断推进，大数据技术受到广泛的关注，数据挖掘作为大数据技术的重要实现手段，能够挖掘数据的关联规则、实现数据的分类、聚类、异常检测和时间序列分析等，解决商务管理、生产控制、市场分析、工程设计和科学探索等各行各业中的数据分析与信息挖掘问题。

R 语言是一种通用的统计计算和数据可视化开源软件环境和编程语言，具有高度可扩展性。R 语言同时支持 Linux、Windows 和 Mac 操作系统。R 语言的前身为贝尔实验室研发的 S 语言。1992 年由新西兰奥克兰大学的 Ross Ihaka 和 Robert Gentleman 创建，并以他们的名字首字母作为项目名称。2007 年 Revolution Analytics 公司成立，对 R 语言做商用支持，2015 年 1 月被 Microsoft 收购。

1997 年，R 语言正式开源，吸引了世界范围内各行业的代码贡献者，实现各种各样的数据分析方法。截至 2018 年 11 月，CRAN（the Comprehensive R Archive Network）官方收录了 13 328 个算法库，常用的包括：

- 数据加载：RODBC、RMySQL、RSQLite、XLConnect、xlsx、foreign；
- 数据处理：dplyr、tidyr、stringr、lubridate；
- 数据可视化：ggplot2、ggvis、rgl、htmlwidgets、googleVis；
- 数据建模：car、mgcv、nlme、randomForest、multcomp、glmnet、survival、caret、mlr；
- 数据报告：shiny、xtable；
- 空间数据：sp、maptools、maps、ggmap；
- 时间序列和金融数据：zoo、xts、quantmod；
- 高性能计算：Rcpp、data. table、parallel；
- 网页数据：XML、jsonlite、httr。

截至本书出版，共有 283 所高校获批“数据科学与大数据技术”专业，其中 985 及 211 高校占比达 13%。目前，国内数据人才缺口更是达到百万

级。由于其开源性、易用性和强大的数据分析能力，R 语言已成为世界范围内应用最广泛的数据科学工具和语言之一。目前，R 语言数据分析与挖掘逐渐成为高校信息类、管理类和数学统计类专业的必修课程内容，同时，作为面向各专业的通识课也广受欢迎。

本书作为立足于应用型本科数据科学与大数据教学的 R 语言核心课教材，具有如下特色：

(1)内容安排合理且全面，从 R 语言的基本编程、数据处理、数据可视化、统计分析到高性能计算和机器学习，循序渐进，深入浅出。

(2)难度适中，适合作为本科中高年级的核心课教材，零基础要求，对编程及数学知识不作为必要基础。

(3)理论与案例相结合，理论与实践相结合，包含了泰坦尼克号乘客生存分析、航班准点数据处理、鸢尾花数据建模等实践案例。

本书全面介绍了 R 语言的基本编程、数据处理、数据可视化、统计分析到高性能计算和机器学习，主要内容分为以下两部分：

第一部分：统计分析基础。第 1 章为 R 语言概述，包括 R 语言的相关背景、基本概念和基本操作等。第 2 章为数据访问，包括基本数据类型、数据的输入和输出等。第 3 章为数据操作，包括数据的缺失值处理、转换、合并和取子集等。第 4 章为数据可视化，包括各种图形元素的绘制和各种图表的绘制。第 5 章为概率与分布，包括常用概率和中心极限定理。第 6 章为基本统计分析，包括描述性统计分析、相关性和常用检验等。第 7 章为回归分析，包括 OLS 回归和回归诊断等。第 8 章为方差分析，包括 ANOVA 模型、单因素和多元方差分析等。

第二部分：机器学习实践。第 9 章为大数据高性能计算，包括大数据的选择、聚合、引用、筛选、连接和变形等。第 10 章为机器学习流程，包括数据探索、划分、填充、特征选择、建模调优和测试评估等。第 11 章主要介绍常用的有监督学习模型，包括线性、朴素贝叶斯、k 近邻、决策树、随机森林、神经网络、支持向量机等。第 12 章主要介绍常用的无监督学习模型，包括 k 均值聚类、DBSCAN 聚类、AGNES 层次聚类和关联分析模型等。

本书由杜宾、钱亮宏、黄勃和高永彬编著。具体分工如下：杜宾编写第 1 章到第 8 章，黄勃编写第 9 章，钱亮宏编写第 10 章和第 11 章，高永彬编写第 12 章。全书由方志军、范磊和许华根主审。感谢孙冉、沈烨和周恒对本书的贡献。

由于编者水平有限，加之时间仓促，书中难免存在疏漏和不足之处，敬请老师和同学批评指正。

编　者

2018 年 11 月

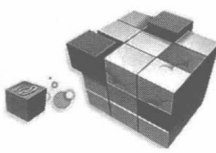
目 录

CONTENTS

第一部分 统计分析基础

第一部分　统计分析基础

第1章



概　　述

随着计算机和互联网的广泛应用，人类可获取并存储的数据量随时间呈指数级增长，当今世界已经进入大数据(big data)时代。企业、公司拥有TB级的客户交易数据，政府机关、学术团体以及各研究机构拥有各类研究课题的大量文本和各式各样的实验、调查数据。从这些海量数据中挖掘信息、探索规律已经形成社会经济结构的一种产业；同时，如何以容易让人理解和沟通的方式呈现隐藏在数据中的信息和知识日益成为有趣且前景可观的职业或工作。

数据分析科学(如统计学、计量心理学、计量经济学、机器学习等)的发展一直与数据的爆炸式增长保持同步。在个人计算机普及之前，学术研究人员已经开发出很多新颖的统计方法，并将其研究成果以论文的形式发表在专业期刊上；同时，这些方法可能需要很多年才能够被程序员编写并整合到广泛用于数据分析的统计软件中。

首先，个人计算机将计算变得廉价且便捷，促使现代数据分析的方式发生变化。与过去一次性设置好完整的数据分析过程不同，现在这个过程已经变得高度交互化，每一阶段的输出都可以充当下一阶段的输入。一个典型的数据分析过程如图1.1所示。在任何时候，子循环都可能要进行数据变换、变量增加或减少、缺失值增补，甚至重新执行整个分析过程。当数据分析师认为分析过程已经深入地理解数据，并且可以回答相关问题时，分析过程可以结束。

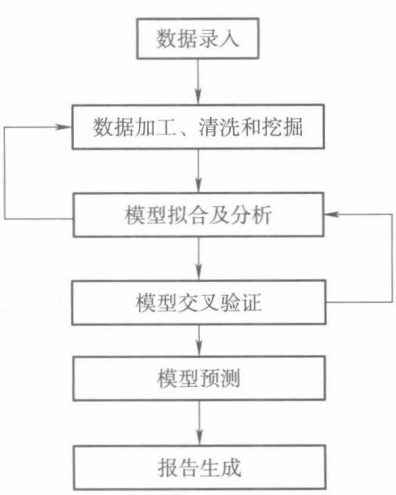


图1.1　典型的数据分析过程

身处“互联网+”时代，新需求层出不穷，同时新的解决方法不断涌现。统计研究者经常在专业性网站上发表新方法和改进的方法，并分享相应的实现代码，全球共同造就了当今的R语言。

个人计算机的出现对数据分析的方式产生影响之二是统计软件的数据处理方式。当数据分析需要在大型机

上完成的时候，机时非常宝贵难求。分析师们设定可能用到的所有参数和选项，再让计算机执行计算。程序运行完毕后，输出的结果可能长达几十甚至几百页。之后，分析师会仔细筛查整个输出，去芜存菁。许多受欢迎的统计软件正是在这个时期开发出来的。直到现在，统计软件依然在一定程度上沿袭这种处理方式。

个人计算机的出现对数据分析的方式产生影响之三是理解和呈现分析结果的变化。人类擅长通过视觉获取有用信息，现代数据分析也日益依赖通过呈现图形来揭示含义和表达结果。当代的数据分析人士需要从广泛的数据源（数据库管理系统、文件、电子表格及统计软件）获取数据，将数据片段融合到一起，对数据做清理和标注，用最新的方法进行分析，以有吸引力的图形化方式展示结果，最后将结果整合成令人感兴趣的报告并向利益相关者和公众发布。

R 语言作为统计学的一门语言应运而生，直到大数据的爆发，突变成为一门数据分析的强大工具，功能全面的数据加工、处理、分析、挖掘的集成软件。

1.1 为什么使用 R 语言

R 是一个具有强大统计分析与绘图功能的软件系统。最先由 Ross Ihaka 和 Robert Gentleman 共同开发，现在由 R 开发核心小组（R Development Core Team）维护，是完全自愿、负责且具有奉献精神的全球性研究型社区，将世界优秀的统计应用软件集成支持分享。与起源于贝尔实验室的 S 语言类似，R 可看作 S 语言的一种实现形式，是一套开源的数据分析解决方案。对比其他流行的统计和绘图软件，如 Microsoft Excel、SAS、SPSS、Stata 以及 Minitab，为什么选择 R？显而易见，互联网时代的 R 具有很多优良的品质和特性。

（1）R 语言开源（open source）意味着 R 是“免费的午餐”。比较而言，多数商业统计软件价格不菲，投入常常成千上万，例如 SAS 统计软件。

（2）R 是一个全面的统计研究平台，提供各式各样的数据分析技术，几乎任何类型的数据分析工作皆可在 R 中完成。

（3）R 囊括其他软件中尚不可用的、先进的统计计算例程。事实上，新方法的更新速度是以周来计算的。

（4）R 的绘图及可视化功能十分强大。如果希望复杂数据可视化，那么 R 拥有强大且全面的一系列可用功能。

R 是一个可进行交互式数据分析和探索的强大平台，其核心设计理念就是支持图 1.1 概述的分析方法。例如，任意一个分析步骤的结果均可被轻松保存、操作，并作为进一步分析的输入。

从多个数据源获取并将数据转化为可用的形式，可能是一个富有挑战性的课题。R 可以轻松地从各种类型的数据源导入数据，包括文件、数据库管理系统、统计软件，乃至专门的数据仓库，同样可以将数据输出并写入到这些系统中。R 也可以直接从网页、社交媒体网站和各种类型的在线数据服务中获取数据。

R 可以使用一种简单且直接的方式编写新的统计方法，易于扩展，并为快速编程实现新方法提供一套自然语言。R 的功能可以被整合到其他语言编写的应用程序，包括 C ++、

Java、Python、PHP、Pentaho、SAS 和 SPSS，继续使用熟悉语言的同时还可以在应用程序中加入 R 的功能。

R 可运行于多种平台之上，包括 Windows、UNIX 和 Mac OS X。R 语言拥有各式各样的 GUI(Graphical User Interface，图形用户界面)工具，通过菜单和对话框提供相应的功能。

图 1.2 所示为 R 绘图功能的一个示例，使用一行代码就可以绘制这张图。后续章节将会进一步讨论这类图形。重要的是，R 能够以一种简单而直接的方式创建信息丰富、高度定制化的图形。使用其他统计语言创建类似的图形不仅费时费力，而且可能根本无法实现。

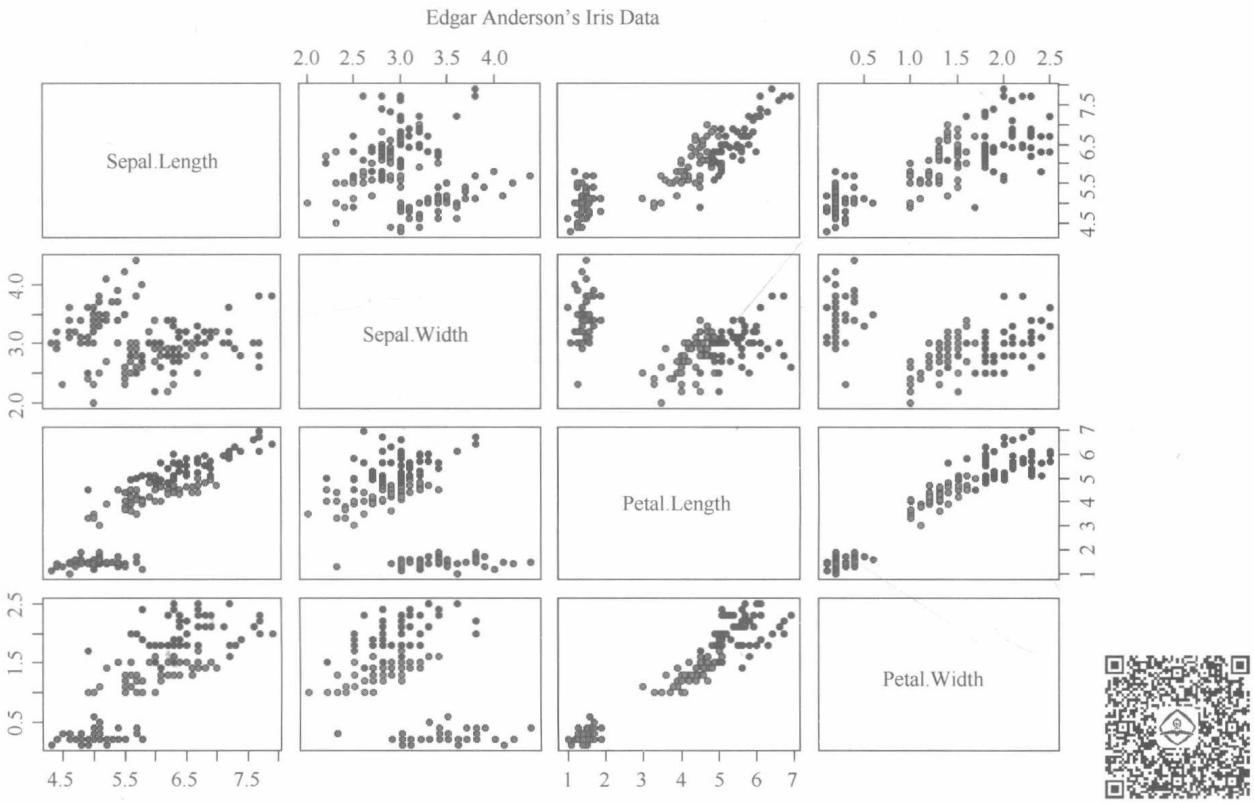


图 1.2 散点图矩阵

R 的学习曲线较为陡峭。因为它的功能非常丰富，所以文档和帮助文件也相当多。另外，由于许多功能由独立贡献者编写，这些文档比较零散而且大多是英文版本。事实上，掌握 R 的所有功能是一项挑战，正确的学习方式是各取所需。下面从 R 的安装开始学习。

1.2 R 的安装

将需要安装 R 的计算机连接互联网，R 便可以在 CRAN(Comprehensive R Archive Network，http://cran.r.project.org)的任一镜像站点上免费下载。操作系统 Windows、Mac OS X 和 Linux 均有对应的安装程序，只需根据平台的安装向导进行选择即可。通过安装包等可选模块(类似从 CRAN 下载)增强 R 的系统功能。

1.3 RStudio 集成环境

RStudio 是一套功能强大的开发环境，提供友好且简约的用户交互界面，并能够兼容各种操作系统，充分发挥 R 语言的各项功能，主界面如图 1.3 所示。本书所有 R 操作和代码均在 R 3.5 版本下运行并实现，如果不强调说明，默认集成开发平台为 RStudio。

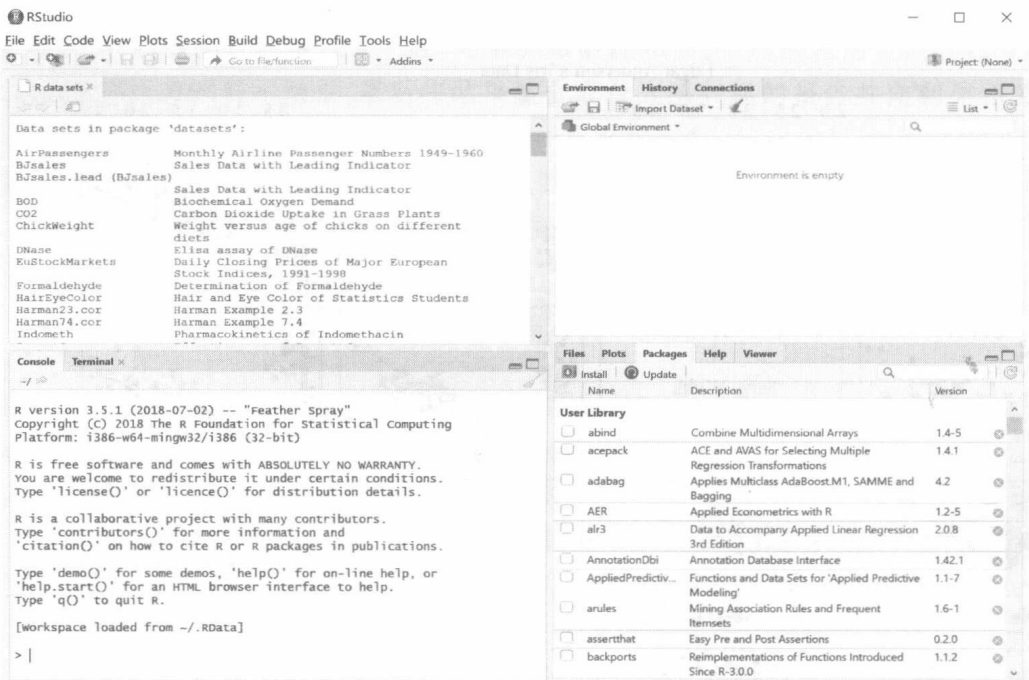


图 1.3 RStudio 主界面

运行 RStudio，在该环境中，最上方是菜单栏；左边区域是控制台(console)，可以输入各种 R 语言命令；右上角区域有 3 个标签，其中，Environment 标签列出当前环境中的所有变量，History 标签列出执行过的历史命令，Connections 标签列出当前连接的外部数据源；右下角区域有 5 个标签，Files 标签列出当前工作目录中的所有文件，Plots 标签显示最近一次所绘制的图，Packages 标签列出当前环境载入的所有包，Help 标签显示最近一次查看的帮助信息，Viewer 标签显示本地网页文件。

1.4 R 的基础操作

R 是一种区分大小写的解释型语言。可以在命令提示符(>)后每次输入并执行一条命令，或者一次性执行写在脚本中的一组命令。R 中有多种数据类型，包括向量、矩阵、数据框(与数据集类似)以及列表(各种对象的集合)。

R 中的多数功能是由程序内置函数、用户自编函数和对对象的创建和操作所提供的。一个对象可以是任何能被赋值的东西。对于 R 来说，对象可以是任何东西(如数据、函数、图形、分析结果等)。每个对象都有一个类属性，类属性可以告诉 R 如何对其进行处理。

一次交互式会话期间的所有数据对象都被保存在内存中。一些基本函数是默认直接可用的,而其他高级函数则包含于按需加载的程序包中。

R 语句由函数和赋值构成。R 使用 <-,而不是传统的 = 作为赋值符号。例如:

```
x <- rnorm(7)
```

该语句创建了一个名为 x 的向量对象,它包含 7 个来自标准正态分布的随机偏差。

R 允许使用 = 为对象赋值,但是这样写的 R 程序并不多,因为它不是标准语法。一些情况下,用等号赋值会出现问题。还可以反转赋值方向,如 rnorm(7) -> x 与上面的语句等价。注释由符号#开头。在#之后出现的同一行所有文本都会被 R 解释器忽略。

1.4.1 启动和退出

首先,通过一个简单的虚构示例直观地感受 RStudio 界面。假设正在研究生理发育问题,并收集 1 名小孩从出生到成长共 12 年内的年龄和体重数据,感兴趣的是体重的分布及体重与年龄的关系,如表 1.1 所示。

表 1.1 儿童的年龄与体重

年龄/岁	体重/斤	年龄/岁	体重/斤
1	15	7	50
2	24	8	55
3	30	9	60
4	35	10	65
5	40	11	70
6	45	12	75

代码 1.1 显示分析的过程。可以使用函数 c()以向量的形式输入年龄和体重数据,此函数可将其参数组合成一个向量或列表。然后用 mean()、sd()和 cor()函数分别获得体重的均值和标准差,以及年龄和体重的相关度。最后使用 plot()函数,从而用图形展示年龄和体重的关系,这样就可以用可视化的方式检查其中可能存在的趋势。函数 q()将结束会话并允许退出 R。

【代码 1.1】一个会话示例

```
age <- c(1,2,3,4,5,6,7,8,9,10,11,12)
weight <- c(15,24,30,35,40,45,50,55,60,65,70,75)
mean(weight)
## [1] 47
sd(weight)
## [1] 18.87278
cor(age,weight)
## [1] 0.9979766
plot(age,weight)
```

代码 1.1 中可以看到，这个小孩从 1 岁到 12 岁的平均体重是 47 斤，标准差为 18.87278，年龄和体重之间存在较强的线性关系（相关度 = 0.9979766）。这种关系也可以从图 1.4 所示的散点图看到。不出意料，随着年龄的增长，儿童的体重也趋于增加。散点图的信息量充足，但不够美观。

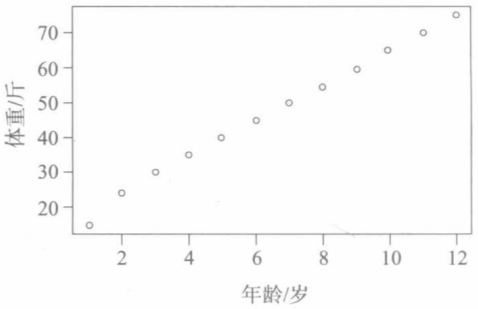


图 1.4　儿童的体重（斤）和年龄（岁）的散点图

如果要初步了解 R 能够绘制何种图形，在命令行中运行 demo( ) 函数即可。其他演示还有 demo(Hershey)、demo(persp) 和 demo(image)。若要浏览完整的演示列表，不加参数直接运行 demo( ) 函数即可。

1.4.2　获取帮助

R 提供大量的帮助功能，学会使用这些帮助文档可以在相当程度上提升编程能力。帮助系统提供当前已安装包中所有函数的细节、参考文献以及使用示例，可以通过表 1.2 中列出的函数查看帮助文档。

表 1.2　R 的帮助函数

函　数	功　能
help. start( )	打开帮助文档首页
help("foo") 或? foo	查看函数 foo 的本地帮助文档（引号可以省略）
help. search("foo") 或?? foo	以 foo 为关键词搜索网络帮助文档
example("foo")	函数 foo 的使用示例（引号可以省略）
RSiteSearch("foo")	以 foo 为关键词搜索在线文档和邮件列表存档
apropos("foo" , mode = "function")	列出名称中含有 foo 的所有可用函数
data( )	列出当前已加载包中所含的所有可用示例数据集
vignette( )	列出当前已安装包中所有可用的 vignette 文档
vignette("foo")	为主题 foo 显示指定的 vignette 文档

help. start( ) 函数打开一个浏览器窗口，可在其中查看入门和高级的帮助手册、常见问题集，以及参考材料。RSiteSearch( ) 函数可在在线帮助手册和 R. Help 邮件列表的讨论存档中搜索指定主题，并在浏览器中返回结果。由 vignette( ) 函数返回的 vignette 文档一般是 PDF 格式的实用介绍性文章。不过，并非所有的包都提供 vignette 文档，经常使用？或?? 可查看某些函数的功能（如选项或返回值）。

1.4.3　工作空间

工作空间（workspace）就是当前 R 的工作环境，存储所有用户定义的对象（向量、矩阵、函数、数据框、列表）。在一个 R 会话结束时，可以将当前工作空间保存到一个镜像，并在下次启动 R 时自动载入。各种命令可在 R 命令行中交互式地输入，使用上下方向键查看已输入命令的历史记录。这样就可以选择一个之前输入过的命令并适当修改，最后按【Enter】键

重新运行。

当前工作目录(working directory)是 R 用来读取文件和保存结果的默认目录。可以使用 getwd()函数查看当前的工作目录,或使用 setwd()函数设定当前的工作目录。如果需要读入一个不在当前工作目录下的文件,则需在调用语句中写明完整的路径,注意使用引号界定这些目录名和文件名。用于管理工作空间的部分函数如表 1.3 所示。

表 1.3 R 工作空间的函数

函 数	功 能
getwd()	显示当前的工作目录
setwd("mydirectory")	修改当前的工作目录为 mydirectory
ls()	列出当前工作空间中的对象
rm(objectlist)	移除(删除)一个或多个对象
help(options)	显示可用选项的说明
options()	显示或设置当前选项
history(#)	显示最近使用过的#个命令(默认值为 25)
savehistory("myfile")	保存命令历史到文件 myfile 中(默认值为 .Rhistory)
loadhistory("myfile")	载入一个命令历史文件(默认值为 .Rhistory)
save.image("myfile")	保存工作空间到文件 myfile 中(默认值为 .RData)
save(objectlist,file = "myfile")	保存指定对象到一个文件中
load("myfile")	读取一个工作空间到当前会话中(默认值为 .RData)
q()	退出 R,将会询问是否保存工作空间

了解表 1.3 中命令是如何运作的,运行代码 1.2 可查看结果。

【代码 1.2】用于管理 R 工作空间的命令使用示例

```
setwd("myprojects/project1")
options()
options(digits = 3)
x <- runif(20)
summary(x)
hist(x)
q()
```

首先,当前工作目录被设置为 myprojects/project1,当前的选项设置情况将显示出来,而数字将被格式化,显示为具有小数点后三位有效数字的格式。然后,创建一个包含 20 个均匀分布随机变量的向量,生成此数据的摘要统计量和直方图。当 q()函数被运行的时候,程序将向用户询问是否保存工作空间。如果用户输入 y,命令的历史记录保存到文件 .Rhistory 中,工作空间(包含向量 x)保存到当前目录中的文件 .RData 中,会话结束,R 程序退出。

setwd()函数的路径中使用正斜杠。反斜杠\在 R 中作为一个转义符。即使在 Windows 平台上运行 R,在路径中也要使用正斜杠。同时,setwd()函数不会自动创建一个不存在的目录。如有必要,可以使用 dir.create()函数创建新目录,然后使用 setwd()函数将工作目录指向这个新目录。

在独立的目录中保存项目是一个好习惯。也许在启动一个会话时使用 setwd( )命令指定到某一个项目的路径,后接不加选项的 load("RData")命令。这样做可以让从上一次会话结束的地方重新开始,并保证各个项目之间的数据和设置互不干扰。这样做可以启动 R,载入保存的工作空间,并设置当前工作目录到这个文件夹中。

1.4.4 输入和输出

启动 R 后将默认开始一个交互式的会话,从键盘接收输入并从屏幕进行输出,也可以运行脚本(包含 R 语句的文件)中的命令集并直接将结果输出到多类目标。

1. 输入

source("filename")函数可在当前会话中执行一个脚本。如果文件名中不包含路径,R 将假设此脚本在当前工作目录中。例如,source("myscript. R")将执行包含在文件 myscript. R 中的 R 语句集合。依照惯例,脚本以 . R 作为扩展名。

2. 文本输出

sink("filename")函数将输出重定向到文件 filename 中。默认情况下,如果文件已经存在,则它的内容将被覆盖。使用参数 append = TRUE 可以将文本追加到文件后,而不是覆盖它。参数 split = TRUE 可将输出同时发送到屏幕和输出文件中。不加参数调用 sink( )函数将仅向屏幕返回输出结果。

3. 图形输出

虽然 sink( )函数可以重定向文本输出,但它对图形输出没有影响。要重定向图形输出,使用表 1.4 中列出的函数即可。最后,使用 dev. off( )函数将输出返回到终端。

表 1.4 图形输出的函数

函　数	输　出
bmp("filename. bmp")	BMP 文件
jpeg("filename. jpg")	JPEG 文件
pdf("filename. pdf")	PDF 文件
png("filename. png")	PNG 文件
postscript("filename. ps")	PostScript 文件
svg("filename. svg")	SVG 文件
win. metafile("filename. wmf")	Windows 图元文件

下面通过一个示例理解整个流程。假设有包含 R 代码的三个脚本 script1. R、script2. R 和 script3. R。执行语句:source("script1. R")。将会在当前会话中执行 script1. R 中的 R 代码,结果将出现在屏幕上。如果执行语句:

```
sink( "myoutput" ,append = TRUE,split = TRUE)
pdf( "mygraphs. pdf" )
source( "script2. R" )
```

文件 script2. R 中的 R 代码将执行,结果显示在屏幕上。除此之外,文本输出将被追加到文件 myoutput 中,图形输出将保存到文件 mygraphs. pdf 中。最后,如果执行语句:

```
sink()
dev.off()
source("script3.R")
```

文件 script3.R 中的 R 代码将执行,结果将显示在屏幕上。这一次,没有文本或图形输出保存到文件中。

R 对输入来源和输出走向的处理相当灵活,可控性很强。

1.5 包

R 提供大量下载即用的功能,最有价值的部分功能是通过可选模块的下载和安装实现的。目前有 5 500 多个称为包(Package)的模块可从 http://cran.r.project.org/web/packages 下载。这些包提供横跨各领域、数量惊人的新功能。例如:分析地理数据、处理蛋白质质谱,甚至是心理测验分析。

1.5.1 什么是包

包是 R 函数、数据、预编译代码以一种定义完善的格式组成的集合。计算机上存储包的目录称为库(library)。.libPaths()函数能够显示库所在的位置,library()函数则可以显示库中有哪些包。

R 默认内置一系列标准包,包括 stats、graphics、grDevices、utils、datasets、methods 以及 base,覆盖许多基本的函数和数据集。其他包可通过下载进行安装,安装后包必须被载入到会话中才能使用。search()函数可以显示当前哪些包已加载并可使用。

1.5.2 包的安装

R 程序包的安装有三种方式:

(1)菜单方式:在接入互联网的条件下,按步骤“程序包 > 安装程序包… > 选择 CRAN 镜像服务器 > 选定程序包”进行实时安装。

(2)命令方式:在接入互联网的条件下,在命令提示符后输入 install.packages("ggplot2"),就可以完成程序包 ggplot2 的安装。

(3)本地安装:在无互联网条件下,先从 CRAN 社区下载需要的程序包及与之关联的程序包,再按第一种方式通过“程序包”菜单中的“用本机的 zip 文件安装程序包”选定本机上的程序包(zip 文件)进行安装。

如果需要,还可通过步骤“程序包 > 更新程序包…”对本机的程序包进行实时更新。

一个包仅需安装一次,但和其他软件类似,包经常被其作者更新。可以使用 update.packages() 函数更新已经安装的包。要查看已安装包的描述,可以使用 installed.packages()函数,将列出已安装的包,以及版本号、依赖关系等信息。

1.5.3 包的载入

包的安装指从某个 CRAN 镜像站点下载它并将其放入库中的过程。R 会话使用包,还需

要使用 library( ) 函数载入这个包。除 R 的标准程序包(如 base 包)外,新安装的程序包在使用前必须先载入,有两种载入方式:

(1)菜单方式:按步骤“程序包 > 载入程序包…”,再从已有的程序包中选定需要的一个加载。

(2)命令方式:在命令提示符后输入 library("ggplot2"),实现加载程序包 ggplot2。

1.5.4 包的使用

载入一个包之后,就可以使用包所包含的一系列函数和数据集。包中往往提供演示性的小型数据集和示例代码,能够让用户尝试这些新功能。帮助系统包含每个函数的一个描述,同时带有示例,每个数据集的信息也被包括其中。help(package = "package_name")命令可以输出某个包的简短描述以及包中的函数名称和数据集名称的列表。使用 help( ) 函数可以查看其中任意函数或数据集的更多细节,这些信息也能以 PDF 帮助手册的形式从 CRAN 下载。

有一些错误是初学者可能常犯的,当程序出错时可以从以下几方面检查:

(1)没有区分大小写。help( )、Help( ) 和 HELP( ) 是三个不同的函数(只有第一个是正确的)。

(2)忽略引号的作用。install. packages("gclus") 能够正常执行,然而 install. packages(gclus) 将会报错。

(3)函数调用时忘记使用括号。例如,要使用 help( ) 而非 help。即使函数无须参数,仍需加上( )。

(4)在 Windows 上,路径名中使用了反斜杠 \。R 将反斜杠视为一个转义字符。所以,setwd("c:\mydata") 会报错。正确的写法是 setwd("c:/mydata") 或 setwd("c:\\mydata")。

(5)使用尚未载入包中的函数或数据。例如,order. clusters( ) 函数包含在 gclus 包中,如果还没有载入 gclus 包就使用 order. clusters( ) 函数将会报错。

R 的出错信息可能不准确,如果严格遵守以上几点,就能避免许多不必要的错误,节省时间,少走弯路。

1.6 结果的重用性

R 有一个非常实用的特点是重用性,即输出结果可以轻松保存,并作为进一步分析的输入使用。例如,运用汽车数据 mtcars 执行一次简单线性回归,通过车身质量(wt)预测每加仑汽油行驶的英里数(mpg)。可以通过以下语句实现:

```
lm(mpg ~ wt,data = mtcars)
```

结果将显示在屏幕上,不会保存任何信息。下一步,执行回归,区别是在一个对象中保存结果:

```
lmfit <- lm(mpg ~ wt,data = mtcars)
```

以上赋值语句创建一个名为 lmfit 的列表对象,其中包含分析的大量信息,如包括预测值、残差、回归系数等。虽然屏幕上没有显示任何输出,但分析结果可在稍后被显示和继续使用。输入 summary(lmfit) 将显示分析结果的统计概要,plot(lmfit) 将生成回归诊断图形,而语句 cook <- cooks. distance(lmfit) 将计算和保存影响度量统计量,plot(cook) 对其绘图。要在新的车身