

# R Frequently Asked Questions

刘思喆 (sunbjt<at>gmail.com)

October 10, 2007

本文档内容来源多样，既有来自于 R (非) 官方文档，也有来自于互联网的非正式捐赠文档；还有若干来自于 COS 论坛的讨论问题。

文档的最初目的是为新手提供一个快速认识 R 软件的平台，所列问题不可能详尽，如果想系统学习 R 软件，建议大家精读一本关于 R 的原版书籍。

---

本文档采用 The GNU Operating System 颁布的 GNU Free Documentation License。因此，你可以在保证文档的完整性前提下自由拷贝、传播这份文档。

你也可以摘录、转载这份文档中的部分内容，但是必须注明来源以及保证所有包含摘录内容的文档也都是自由文档，也就是可以免费得到的。详情请阅读 GNU Free Documentation License。

---

## §A 前言

### R 是做什么的？

R 是一个有着统计分析功能及强大作图功能的软件系统，是由奥克兰大学统计学系的 Ross Ihaka 和 Robert Gentleman 共同创立。由于 R 受 Becker, Chambers & Wilks 创立的 S 和 Sussman 的 Scheme 两种语言的影响，所以 R 看起来和 S 语言非常相似。

### 在哪里可以下载 R 的安装程序

在 R 的官方网址上，选择网站镜像 <http://cran.r-project.org/mirrors.html>，比如 UC Berkeley 下载软件副本。R 拥有在 Linux, MacOS X, Windows 平台下的各个版本，如果是 Windows 用户，进入镜像网站，选择 Windows (95 and later)，进入 base，下载 R-x.x.x-win32.exe。

### 为什么 R 叫做 R

部分是因为两位 R 的作者 (Robert Gentleman 和 Ross Ihaka) 的名字，部分是受到了贝尔实验室语言 S 的影响。

### CRAN 是什么意思？

CRAN 是 Comprehensive R Archive Network 的简写，是拥有同一资料，包括 R 的发布版本，包，文档和源代码的网站集合。

## 我是新手，我如何开始学习 R

R 的官方网页拥有大量英文学习资源，还可以参考 <http://cran.r-project.org/other-docs.html> 上的中文翻译文档。[统计之都 bbs](#) 是一个不错的学习 R 的中文网站。

## 初学者阅读 R 自行安装的 R-intro 就可以了么？

R-intro 确实是官方文档中最基础的，但它不是从 R 软件应用角度讲的，故并不适合于 R 初学者。可以参考 R-intro 中数据类型、结构部分，作为基础学习。

## 能列举一些 R 的经典书目么？

Modern Applied Statistics with S (Venables and Ripley)  
The New S Language: A Programming Environment for Data Analysis and Graphics  
— (Richard A. Becker, John M. Chambers, Allan R. Wilks)  
A Handbook of Statistical Analysis Using R (Brian S. Everitt, Torsten Hothorn)  
Data Analysis and Graphics using R (Maindonald and Braun)  
Introductory Statistics with R (Dalgaard)

## R 需要编程么？

不！大多数时候不需要，因为 R 有很多函数和包，而且每天都在增加，你用的一般方法和函数都可以在 R 自带包中找到。

## 能否简单举一个 R 的例子？

```
1 x <- rnorm(100, mean = 5, sd = 0.1)
  mean(x)
3 sd(x)
  summary(x)
```

当然你还可以使用 `demo()` 函数，比如 R 漂亮的图形演示：

```
demo(graphics)
```

## R 需要注册费用么？

不需要！R 是一款在 [GNU General Public License \(GPL\)](#) 下发布的开源软件，只是很少一部分包不能用于商业用途。不知道为什么有些费时、费力且价格不菲的商业统计软件，居然还在生存？

## 如何在发行出版物里引用 R

如果你是  $\text{\LaTeX}$  用户，可以在 R 中使用命令 `citation()` 得到可供  $\text{\BibTeX}$  使用的内容；或者是某一个包的引用

```
1 citation(package = 'package')
```

## §B 基础知识

### 如何获得帮助？

R 的帮助系统非常强大，可以直接使用 “?topic” 或 `help(topic)` 来获取 topic 的帮助；使用 `help.search("topic")` 来搜索帮助系统。

如果你只知道函数的部分名称，那么可以使用 `apropos("tab")` 来搜索得到载入内存所有包含 tab 字段的函数。

如果还没有得到需要的资料，还有 R Site Search: <http://finzi.psych.upenn.edu/search.html>，等价于在 R 平台上使用 `RSiteSearch()` 函数

### R 可使用的最大内存是多少？

Windows 平台下可以使用

```
1 rgui -- max - mem - size
```

来设定内存大小。设定值为 32Mb 到 3Gb 的任意数值。但 Windows 平台可用最大有效内存为 2Gb，也就是说，实际上 R 的内存区间为 32Mb 至 2Gb。

### R 支持中文么？

支持，但不好！在 R 中，大部分包的作者都是以英文为母语的，不会对中文字符考虑太多，故建议使用全英文环境。

### 如何清除变量？

清除单个变量使用 `rm()` 函数，清除内存中所有的变量：

```
1 rm(list = ls(all = TRUE))
```

### 如何更改小数点后显示数字位数？

`options(digits = )`，digits 后面的参数为 1 至 22 的数字，默认为 7。options 函数还可以改变很多全局选项，如更改提示符 (prompt)，是否显示错误信息 (show.error.messages) 等。

### 如何调用系统内的程序？

使用 `system()` 函数或用 `shell.exec()` 调用相应程序来打开文件：

```
1 # go to the cran
  system(paste('C:/Program Files/Internet Explorer/iexplore.exe',
3 'cran.r-project.org'), wait = FALSE)
  # invoke the notepad
5 system("notepad")
  shell.exec("C:/WINDOWS/clock")
```

### Windows 下升级 R，但不想重装 packages？

在其他目录下安装 R，再将旧版本保留的 library 目录下的文件拷贝至新版本 library 目录下，然后 `update.packages()`；或卸载 R，把 R 装到旧的目录下，然后 `update.packages()`。

## R 的工作目录在哪里？

getwd() 获得 R 的工作目录 (Working Directory)，使用 setwd() 设置工作目录位置。

## 我怎样保存自己的工作？

使用 save.image() 函数。它将在 R 的起始目录保存记忆区 (working space) 至.RData 文件；或者使用 save(..., file = ) 保存需要保存的 R 对象。

## 如何获得已经安装包的列表？

可以使用

```
.packages(all.available = TRUE)
```

命令获得本地安装的包列表。

## R 如何安装包？

通过选择下载镜像，R 可以自动安装未安装在本地的包，当然也可以从镜像网站下载可用的包，直接本地安装<sup>1</sup>。

## library() 的逆向操作是什么？

当加载包后，需要分离 R 同包时，可以使用

```
1 detach("package:pkg")
```

## Library 和 Package 有什么区别？

这两个概念的确容易混淆，因为 R 中加载 Package 的命令是 Library! Library 是一个目录，可能包含一个或多个 Package；而 Package 是包含函数、数据、手册的一个集合，属于某个 Library，即 (Windows 下) 的 “\*.zip” 文件。

## 如何使用 R 内置的数据集？

R 在 datasets 包中共提供了 100 个可以使用的数据集，这些数据集都可以通过 data() 函数加载入内存。

```
1 dim(data()$results)
  data()$results[,4]
```

## data frame 是什么？

data.frame (数据框) 可以理解是一个松散的数据集。它可以是由不同类型的列 (数字、因子、字符等) 组成的类矩阵 (matrix-like)。

## 如何得到函数的代码？

通常情况你只需要写出你需要查看的函数名，比如：

```
dist
```

---

<sup>1</sup>R 有 Unix、Mac、Windows 三个版本，注意包也分别对应三个版本

但有时候这个函数可能是一个类函数 (Generic Function)，上面的方法就需要稍稍改进一下：先使用 `methods()` 函数来查看这个类函数的列表，找到具体需要的函数<sup>2</sup>，写出来，回车 — 问题解决。

```
1 summary # It is a generic function
2 methods(summary) # list of the S3 methods
3 summary.lm # maybe you want to know the linear models's summary
```

如果要究根问底，可以去下载源代码压缩包 (\*.tar.gz，比如 R-2.5.1.tar.gz)

### 我想查看一个矩阵的前（后）几行，怎么办？

可以使用 `head()` 或 `tail()` 函数。

```
1 head(CO2)
```

这两个函数是类函数，它们可以应用于向量、矩阵、数据框、表格或函数。如果只想随机看看对象中的一些内容，还可以使用 `car` 包中的 `some` 函数。

### R 里面可以使用科学计数法么？

可以。

```
1 1e10 == 10000000000
```

## §C 输入与输出

### R 可以读取其他统计软件录入的数据么？

可以，使用 `foreign` 包，它可以读取 Minitab, S, SAS, SPSS, Stata, Systat, dBase 保存的数据

### R 可以读 Excel 的数据么？

可以，但不推荐直接读取 Excel 文件，或许只有微软知道 Excel 里面有什么东西。通常有三种方法读取 Excel：

1. 将 Excel 另存为 csv(Comma Separated Values) 文件，使用 `read.csv()` 函数读取（推荐）；
2. 加载 RODBC 包，使用 `odbcConnectExcel()` 函数读取 xls 文件，

```
1 library(RODBC)
2 z <- odbcConnectExcel("rexceltest.xls")
3 dd <- sqlFetch(z, "Sheet1")
4 close(z)
```

详细请参考 R Data Import/Export;

3. `xlsReadWrite` 包中的 `read.xls` 函数。

---

<sup>2</sup>标注星号的函数可以使用 `getAnywhere()` 函数获得代码

### 可以将 R 中显示的结果输出到文件么？

可以。使用 `sink()` 函数。

```
1 data(CO2)
2 sink("CO2.txt")
3 CO2
4 sink() # go to your work directory , you will get CO2.txt
```

### R 可以从内存直接读写数据么？

可以。拷贝需要读取的内容，使用

```
1 data <- read.table("clipboard")
2 write.table("clipboard")
```

### 怎样将因子 (factor) 转换为数字

这个问题时有发生，假设 `f` 是一个这样的因子对象，我们可以使用

```
1 as.numeric(as.character(f))
2 # or
3 as.numeric(levels(f))[as.integer(f)]
```

### R 可以使用电子表格输入数据么？

可以使用 `edit()` 和 `fix()` 函数。

```
1 data <- data.frame()
2 edit(x) ; fix(x)
```

### 为什么当我使用 `source()` 时，不能显示输出结果？

对需要显示输出的对象使用 `print()`，或者使用 `source(file, echo = TRUE)`。如果 R 代码里面包含 `sink()` 之类的函数，必须使用 `source(file, echo = TRUE)` 才能得到正确的输出结果，否则 `sink` 的对象将为空。

### R 可以输出可供 $\text{T}_{\text{E}}\text{X}$ 使用的文本么？

可以，参考 `Hmisc` 包中的 `latex()` 函数和 `xtable` 包中的 `xtable()` 函数。

```
1 a <- matrix(1:6, nr=1) # require(xtable)
2 colnames(a) <- paste("col", 1:6)
3 xtable(a)
```

### 找不到文件，但我知道它在哪！

在 R 里面使用必须使用双反斜杠或单斜杠表示文件路径，比如：

```
1 d:\\R-2.4.1\\library\\xgobi\\scripts\\xgobi.bat
2 d:/R-2.4.1/library/xgobi/scripts/xgobi.bat
```

当然还可以使用 `file.choose()` 函数打开一个 Windows 标准文件选择对话框，手动选择文件。当然还有可以使用 `choose.dir()` 打开 Windows 标准目录选择对话框 ☺。

### R 可以直接从数据库读取数据么？

可以，并且还可以通过 SQL 语句对数据库进行操作。R 对于基于 SQL 语言的关系型数据库有良好的支持，这些数据库既有商业数据库 Oracle、Microsoft SQL Server、IBM DB2 等，也包含在 **GNU General Public License (GPL)** 下发布的 MySQL 等开源数据库。

RMySQL<sup>3</sup> 包中提供了到 MySQL 数据库的接口；RODBC 包提供了更为广泛数据库接口的解决方案 — 所有支持 ODBC 接口的数据库，并且，使用这种方式，相同的 R 代码可以应用于不同类型的数据库。

经测试，Windows 平台上的 Microsoft SQL Server、Access、Oracle、MySQL、PostgreSQL，和 Linux 平台上的 MySQL、Oracle、PostgreSQL、SQLite 都有良好的应用（详细参考 R-data）。

## §D 数据创建和操作

### 如何删掉缺失值？

在 R 中使用 NA (not available) 表示缺失值，既然 not available，故不能使用

```
x == NA
```

来判断缺失值。可以用函数 `is.na()` 来判断是否有缺失值，使用

```
1 x[!is.na(x)]
```

删除缺失值。

### 如何将字符串转变为命令执行？

```
1 x <- 1:10
  a <- "print(x)"
3 class(a)
  eval(parse(text = a))
```

### 如何向一个向量追加元素？

```
2 x <- 1:5
  (foo <- c(x[1], 0, x[2:5])) # expected result
  append(x, 0, after = 1)
```

### 如何移除某行 (列) 数据

可以使用函数 `subset(select = )`；或者使用下标：

```
1 x <- data.frame(matrix(1:30, nrow = 5, byrow = T))
  dim(x)
```

---

<sup>3</sup>需要包 DBI 的支持

```

3   print(x)
   new.x1 <- x[-c(1,4),]    #row
5   new.x2 <- x[, -c(2,3)]  #col
   new.x1 ; new.x2

```

### 我的数据框有相同的行，如何去掉这些行？

参考 unique 函数。unique 函数可以去掉向量、数据框或类似数列的数据中重复的元素。

```

x <- c(9:20, 1:5, 3:7, 0:8)
2   (xu <- x[!duplicated(x)])
   unique(x)    # is more efficient

```

这里 duplicated 函数返回了元素是否重复的逻辑值。

### 如何对数列 (array) 进行维度变换？

使用函数 aperm

```

1   x <- array(1:24, 2:4)
   xt <- aperm(x, c(2,1,3))
3   dim(x) ; dim(xt)

```

### 如何删除 list 中的元素？

R 中使用 NULL 表示无效的对象。

```

1   lst <- list("a"=list("b"=1,"c"=2),"b"=list("d"=3,"e"=4))
   lst[["a"]][["b"]] <- NULL # or lst$a$b <- NULL
3   lst

```

### 如何对矩阵按行 (列) 作计算？

使用函数 apply()

```

1   vec=1:20
   mat=matrix(vec,ncol=4)
3   vec
   cumsum(vec)
5   mat
   apply(mat,2,cumsum)
7   apply(mat,1,cumsum)

```

### 如何注掉大段的 R 脚本

如果你使用支持正则表达式的文本编辑器的话，可以考虑用正则表达式 (Regular Expression)；或者将大段的代码写入一个 \*.R 文件，如果需要注掉的话，在 source(\*.R) 前加入 # 即可；还可以使用



```

1   if (FALSE){
    something passby
3   }

```

### 如何对数据框 (data frame) 的某列作数学变换？

使用 transform() 函数对其操作，具体参考?transform

### 如何求解两组平行向量的极值？

pmax() 和 pmin() ，如：

```

1   x <- 1:10      ;   y <- rev(x)
    pmax(x,y)      ;   pmin(x,y)

```

### 如何对不规则数组进行统计分析？

参考 tapply() ：

```

    n <- 17; fac <- factor(rep(1:3, len = n), levels = 1:5)
2   table(fac)
    tapply(1:n, fac, sum)
4   tapply(1:n, fac, mean)
    ## or reverse a list
6   to <- list(a = 1, b = 1, c = 2, d = 1)
    tapply(to, unlist(to), names)

```

### 判断数据框的列是否为数字？

sapply(dataframe, is.numeric)

### 一组数中随机抽取数据？

函数 sample()

```

1   n <- 1000
    x <- sample(c(-1,1), n, replace=T)
3   plot(cumsum(x), type="l",
    main="Cumulated_sums_of_Bernoulli_variables")

```

还可以参考已知分布的随机抽取数据的函数。

### 如何根据共有的列将两个数据框合并？

我们经常会遇到两个数据框拥有相同的时间或观测值，但这些列却不尽相同。处理的办法就是使用 merge(x, y ,by.x = ,by.y = ,all = ) 函数。

### 如何将标准化数据？

参考 scale 函数。

```

1 x <- c(rnorm(100), 2*rnorm(30))
2 m <- scale(x, scale = F) # only centering
  n <- scale(x, center = F) # only scaling

```

## §E 数学运算

### 如何做出曲线积分？

R 语言使用 `integrate` 函数来得到积分结果，如

```

1 integrate(dnorm, -1.96, 1.96)
  integrate(dnorm, -Inf, Inf)
3 ## a slowly-convergent integral
  integrand <- function(x) {1/((x+1)*sqrt(x))}
5 integrate(integrand, lower = 0, upper = Inf)

```

### 如何得到一个列向量？

矩阵转置可以使用函数 `t()`， $x < -1:10$ ，R 中默认 `x` 为 “integer” 类型数据，这时可以用 `t(t(x))` 得到列向量：

```

1 x <- 1:10 ; class(x)
  t(x) ; class(t(x))
3 t(t(x)) ; class(t(t(x)))

```

### 如何生成对角矩阵？

对一个向量使用 `diag()` 函数，得到对角线元素为向量的对角矩阵；对整数 `Z` 使用此函数得到 `Z` 维的单位矩阵。

### 求矩阵的特征值和特征向量的函数是什么？

参考：`eigen` 函数

### 如何构造上（下）三角矩阵？

参考函数 `lower.tri()` 和 `upper.tri()`。

```

1 Rmat <- matrix(1:16, 4, 4)
  Rmat[lower.tri(Rmat)] <- 0
3 Rmat

```

### 求立方根如何运算？

$x^{1/3}$ 。在 R 里面 `sqrt()` 函数可以计算开平方，故新手容易推测开立方也有函数。事实上 R 里面使用 `^` 来作幂函数运算。`^` 不但是运算符，还可以看作是函数：

```

1 "^(x, 1/3)

```

### 如何求矩阵各行 (列) 的均值?

如果运算量不是很大, 当然可以使用 `apply()` 函数。`rowMeans()` 和 `colMeans` 函数可以更快地得到你要的结果。

```
1  m <- 1000 ; n <- 3000
   A <- matrix(1:m*n ,m ,n)
3  system.time(B1 <- matrix(apply(A,2,mean) , m, n ,by=T))
   system.time(B2 <- matrix(colMeans(A) , m, n, by=T))
```

### 如何计算组合数或得到所有组合?

`choose()` 用于计算组合数  $\binom{n}{k}$ , 函数 `combn()` 可以得到所有元素的组合。使用 `factorial()` 计算阶乘。

### 如何在 R 里面求 (偏) 导数?

使用函数 `D()`

```
1  f1 <- expression(sin(x)*x)
2  f2 <- expression(x^2*y + y^2)
   D(f,"x")
```

## §F 字符操作

### R 对大小写敏感么?

R 中有很多基于 Unix 的包, 故 R 对大小写是敏感的。可以使用 `tolower()`、`toupper()`、`casefold()` 这类的函数对字符进行转化。

### R 运行结果输出到文件中时, 文件名中可以用变量代替吗?

可以, 通过使用 `paste()` 函数。

```
1  for(var in letters[1:6]){
   x <- var
3  write.table(x ,paste("FOO_" , var , ".txt" , sep = ""))
   } # You will get "FOO_a.txt" ...
```

### 在 R 中如何使用正则表达式 (Regular Expressions )

在 R 中, 有三种类型的正则表达式: *extended* regular expressions, 使用函数 `grep(extended = TRUE)` (默认); *basic* regular expressions, 使用 `grep(extended = FALSE)`; *Perl-like* regular expressions, 使用 `grep(perl = TRUE)`。比如 “.” 用来匹配任意字符 (使用 “\.” 来匹配 “.”):

```
grep("J.", month.abb)
```

详细可以参考 `help("regex")`。

如何返回字符个数？

参考 `nchar` 。

```
1 | nchar(month.name[9])
```

## §G 日期与时间

日期可以做算术运算么？

可以。一般我们需要使用 `as.Date()`，`as.POSIXct()` 函数将读取的日期（字符串）转化为 “Date” 类型数据，“Date” 类型数据可以进行算术运算。

```
1 | d1 <- c("06/29/07") ; d2 <- c("07/02/07")
   | D1 <- as.Date(d1, "%m/%d/%y")
   | D2 <- as.Date(d2, "%m/%d/%y")
3 | D1 + 2 ; D1 - D2
   | difftime(D1,D2,units = "days")
5 |
```

如何将日期表示为 “星期日, 22 七月 2007” 这种格式？

使用 `format()` 函数。

```
1 | format((Sys.Date()), format="%A, %d %B %Y")
```

具体 `format` 参数可以参考 `help(strptime)` 的 `details` 部分。

## §H 绘图

如何在同一画面画出多张图？

三种办法：修改绘图参数，如 `par(mfrow = c(2,2))` 或 `par(mfcol = c(2,2))`；更为强大功能的 `layout` 函数，它可以设置图形绘制顺序和图形大小；`split.screen()` 函数

如何设置图形边缘大小

修改绘图参数 `par(mar = c(bottom, left, top, right))`，`bottom`，`left`，`top`，`right` 四个参数分别是距离 `bottom`，`left`，`top`，`right` 的长度，默认距离是 `c(5, 4, 4, 2) + 0.1`。

如何做密度曲线？

常用的办法是：做出 `x` 的一个序列，然后做出 `dfunction(x)`，比如：

```
1 | x=seq(-3, 3, .05)
   | plot(x, dnorm(x), type="l")
```

`dfunction(x)` 中的 `function` 是指分布族，可以参考 `R-intro` 中的 [Probability distributions](#) 章节，或 `help.search("distribution")`。

### 如何加图例？

绘制图形后，使用 legend 函数，help("legend")

```
1 with(iris , plot(Sepal.Length , Sepal.Width ,
2 pch=as.numeric(Species) , cex=1.2))
3 legend(6.1 , 4.4 , c("setosa" , "versicolor" , "virginica") ,
4 cex=1.5 , pch=1:3)
```

### 怎么做饼图？

参考 pie() 函数。饼图展示数据的能力较差，因为我们的眼睛对长度单位比较敏感，而对关联区域和角度感觉较差。建议使用条形图 (bar chart) 和点图 (dot chart)。

### 如何做茎叶图？

参考 stem 函数。

```
stem(faithful$eruptions)
```

### R 如何做双坐标图？

在 R 中可以通过绘图参数 par(new = TRUE) 使得绘制第二个绘图 (high-level plot) 时保留第一个绘图区域，这样两张绘图会重叠在一起，看起来就是双坐标图。下面的例子是在同一张图上绘制 GDP 和失业率 (UR)：

```
1 year <- 1995:2005
2 x1 <- data.frame(year , GDP = sort(rnorm(11,1000,100)))
3 x2 <- data.frame(year , UR = rnorm(11,5,1))
4 par(mar = c(5,4,4,6)+0.1)
5 plot(x1, axes = FALSE, type="l")
6 axis(1, at = year, label = year); axis(2)
7 par(new = T, mar = c(10,4,10,6) + 0.1)
8 plot(x2, axes = FALSE, xlab = "" , ylab = "" , col = "red", type= "b")
9 mtext("UR(%)", 4, 3, col="red")
10 axis(4, col ="red", col.axis = "red")
```

但不推荐使用双坐标图来进行数据描述，这样很容易造成误解。并且在 R 中做出并排图形作对比很容易，没有必要绘制双坐标图。

### 如何在已有图形上加一条水平线

使用低水平绘图命令 abline()，它可以作出水平线 (y 值 h=)、垂线 (x 值 v=) 和斜线 (截距 a=, 斜率 b=)。

## 常用的绘图装置都有哪些？

参考?Devices。

Names	Description
windows	The graphics driver for Windows (on screen, to printer and to Windows metafile).
postscript	Writes PostScript graphics commands to a file
pdf	Write PDF graphics commands to a file
pictex	Writes LaTeX/PicTeX graphics commands to a file
png	PNG bitmap device
jpeg	JPEG bitmap device
bmp	BMP bitmap device
xfig	Device for XFIG graphics file format
bitmap	bitmap pseudo-device via GhostScript (if available).

## 可以打开多个图形装置么？

可以。当打开多个图形装置后，使用 `dev.list()` 察看图形装置的数目（除了装置一），使用 `dev.cur()` 察看当前使用的图形装置，`dev.set()` 改变激活指定的图形装置，`dev.off()` 关闭图形装置。

## 坐标 y 上的数字如何水平放置？

```
?par      #      see las
2 plot(0,0,xaxt="n", type="n", ylim=c(0,100), las=1 )
  mtext("35",side=2,at=35, line =1, las=1)
```

## 如何做雷达图？

R 里面使用 `stars` 函数来做雷达图。

```
1 stars(state.x77[, c(7, 4, 6, 2, 5, 3)], full = FALSE,
  key.loc = c(10, 2))
```

这里的 `full = FALSE` 参数表示只绘制雷达图的上半部分（反之，绘制整个雷达图）；`key.loc` 参数表示基准图例的位置。

## 为什么 R 不能显示 8 种以上的颜色？

当绘图参数 `col` 使用数字来代替颜色名时会有这种情形：

```
barplot(rnorm(15, 10 , 3) , col = 1:15)
```

在 R 中共有 657 种颜色名称可以使用，它们的名称可以通过

```
1 colors()
```

来得到，但事实上有些颜色名称代表的颜色重复，R 中颜色名称只能显示 502 种颜色。当然可以使用函数 `rgb()` 来指定任意色彩。

### 如何用不同的颜色来代表数据？

高级绘图函数一般都有 `col` 参数可以设置。对于像 `barplot()` 这类图形，可以使用“颜色组”(color sets) 来设置颜色，颜色组包括如下几类：

Names	Description
<code>rainbow()</code>	彩虹色 (  )
<code>heat.colors()</code>	红色至黄色 (  )
<code>terrain.colors()</code>	绿色、棕色至白色 (  )
<code>topo.colors()</code>	深蓝色至浅棕色 (  )
<code>cm.colors()</code>	浅蓝到白色，浅紫色 (  )
<code>gray()</code> 、 <code>grey()</code>	灰色 (  )

```
1 x <- 1:10 ; names(x) <- letters[1:10]
  barplot(x, col = rev(heat.colors(10)))
3 barplot(x, col = gray((1:10)/10));
```

### 怎样将 R 的颜色同 RGB 对应起来？

参考函数 `col2rgb()`

```
1 write.table(t(col2rgb(rainbow(7))/255), sep = ",")
```

### 如何调整所绘图形的大小？

Windows 平台下，正常情况打开绘图窗口，调整窗口大小，点击菜单直接保存，或使用 `savePlot()` 函数保存；当然也可以事先用

```
1 windows(width = , height = )
```

打开一个定义好大小的窗口，然后绘图；还可以使用 `pdf()`、`postscript()`、`png()`、`jpeg()`、`pictex()` 等“后台生成”函数，这些函数都有设置图形大小的参数；还可以使用

```
1 dev.copy(device , file="", height , width)
```

命令。

### 如何模拟布朗运动？

布朗运动可以用标准正态的随机模拟值的累积和来模拟：

```
1 # two dimensions
  n <- 100
3 x <- cumsum(rnorm(n))
  y <- cumsum(rnorm(n))
5 plot(x, y, type = 'l')
```

### 如何获得连接若干点的平滑曲线？

如果已知做出这些点的函数可以使用 `curve(expr, from, to, add = T)` 函数。反之，使用立方曲线差值函数 `spline(x, y, n=)`，如：

```
1 x <- 1:5
  y <- c(1,3,4, 2.5,2)
3 plot(x, y)
  sp <- spline(x, y, n = 50)
5 lines(sp)
```

### 如何绘制三维图？

参考 `persp()`，`contour()` 函数。

### 想把一个数值的矩阵映射为一个颜色方格的矩阵，什么函数？

参考 `image()` 和 `filled.contour()` 函数：

```
1 x <- y <- seq(-10, 10, length=50)
  f <- function(x,y){
3     r <- sqrt(x^2 + y^2)
      10*sin(r)/r
5     }
  z <- outer(x, y, f)
7 image(x, y, z)
  filled.contour(x, y, z)
```

### 我想为一个数据框的每一列都做 Q-Q 图？

```
1 table <- data.frame(x1 = rnorm(100), x2 = rnorm(100,1,1))
2 par(ask=TRUE) # wait for changing
  results = apply(table, 2, qqnorm)
4 par(ask=FALSE)
```

### 如何在 R 的绘图中加入数学公式？

参考?plotmath

```
1 x <- 1:10 ; plot(x, type = "n")
2 text(3,2,expression(paste("Temperature", degree, "C)in_2003")))
  text(4,4,expression(bar(x) == sum(frac(x[i], n), i=1, n)))
4 text(6,6,expression(hat(beta) == (X^t * X)^{.1} * X^t * y))
  text(8,8,expression(z[i] == sqrt(x[i]^2 + y[i]^2)))
```



如何在条形图上显示每个 bar 的数值？

```
1 x <- 1:10 ; names(x) <- letters[1:10]
  b <- barplot(x, col = rev(heat.colors(10)))
3 text(b, x, labels = x, pos = 3)
```

## §I 统计模型与拟合

如何模拟高斯（正态）分布数据？

使用 `rnorm(n, mean, sd)` 来产生  $n$  个来自于均值为 `mean`，标准差为 `sd` 的高斯（正态）分布的数据。在 R 里面通过分布前增加字母 ‘**d**’ 表示概率密度函数，‘**p**’ 表示累积分布函数，‘**Q**’ 表示分位数函数，‘**r**’ 表示产生该分布的随机数。这些分布具体可以参考 R-intro 中的 [Probability distributions](#) 章节，或 `help.search("distribution")`。

如何做交叉列联表？

`table()` 函数。`table(x)` 为  $x$  的频数表；`table(x,y)` 为交叉列联表。

如何更新模型？

参考 `update()` 函数：

```
1 summary(f0 <- lm(Fertility ~ ., data = swiss))
  f1 <- update(f0, . ~ . - Examination)
3 summary(f1)
```

## R 中如何实现分位数回归 (Quantile Regression)

参考 `quantreg` 和 `quantregForest` 包

如何做聚类分析？

K 均值聚类 (`kmeans()`)：

```
1 x <- rbind(matrix(rnorm(100, sd = 0.3), ncol = 2),
  matrix(rnorm(100, mean = 1, sd = 0.3), ncol = 2))
3 cl <- kmeans(x, 2, 20)
  plot(x, col = cl$cluster, pch=3, lwd=1)
5 points(cl$centers, col = 1:2, pch = 7, lwd=3)
  segments(x[cl$cluster==1,][,1], x[cl$cluster==1,][,2],
7         cl$centers[1,1], cl$centers[1,2])
  segments(x[cl$cluster==2,][,1], x[cl$cluster==2,][,2],
9         cl$centers[2,1], cl$centers[2,2],
         col=2)
```

层次聚类 (`hclust()`)：

```
hc <- hclust(dist(USArrests), "ave")
```

```
2 | (dend1 <- as.dendrogram(hc)) # "print()" method
   | plot(dend1, nodePar=list(pch = 2:1, cex=.4*2:1, col = 2:3), horiz=T)
```

当然还有专做聚类的包: cluster

```
1 | library(cluster)
   | clusplot(x, pam(x, 2)$clustering)
```

### 如何做主成分分析?

stats 包中的 princomp 函数。

```
2 | (pc.cr <- princomp(USArrests, cor = TRUE))
   | plot(pc.cr, type = "lines" # or "barplot"
   |      ) # or screeplot
4 | loadings(pc.cr)
```

princomp() 中的参数 cor = TRUE 表示使用样本相关矩阵作主成分分析, 反之使用样本协方差矩阵。loadings() 返回因子荷载。screeplot() 绘制碎石图。

### 怎样做因子分析?

在 R 中, 使用factanal() 函数对矩阵进行极大似然因子分析。

```
example(factanal)
```

### 如何对样本数据进行正态检验?

比较常见的方法: shapiro.test(), ks.test()(Kolmogorov-Smirnov 检验), jarque.bera.test() (需要 tseries 包)

### R 如何做结构方程模型?

参考 sem 包。

### 多项式回归应该使用什么函数?

使用 I(), 例如:

```
1 | lm(y ~ x + I(x^2) + I(x^3))
```

### 如何求解没有常数项的线性回归模型?

只需在公式中引入 0 即可:

```
1 | result <- lm(smokes ~ 0 + male + female, data=smokerdata)
```

### D-W 检验在哪里?

```
1 | help.search("Durbin-Watson")
```

car 包中的 `durbin.watson` 函数, `lmtest` 包中的 `dwtest` 函数。

### 如何做 **Decision Tree**?

参考 `rpart` 包。

### **box-cox** 变换?

`MASS` 包中的

```
1 | boxcox()
```

函数。

### 检验异方差的 **Breusch-Pagan** 检验?

`lmtest` 包中的 `bptest()` 函数, 或者利用 `car` 包中的 `ncv.test()` 函数

### 计算 **OLS** 有没有简便方法?

有, 可以使用函数 `qr.solve()` ,

```
1 | qr.solve(X,y)
```

等价于  $(X'X)^{-1}X'y$

## §J 其他

### **R** 可以使用网页来显示结果么?

可以。包 `Rpad` 提供基于同 `R` 的网页接口, 假设已经安装了包 `Rpad` , 可以在本地查看 `Rpad` 的效果:

```
1 | library(Rpad)
   | Rpad() # enjoy it
```

### **R** 有类似于 **SPSS** 的界面么?

有! 安装包 `Rcmdr` , 加载包后, 使用命令

```
Commander()
```

调出可供使用的图形使用界面。由于这个图形使用界面需要若干基础包外的其他函数, 故还需要包 `car` 、 `effects` 、 `abind` 、 `lmtest` 、 `multcomp` 、 `relimp` 、 `RODBC` 、 `rgl` 的支持。

### 怎样来计算函数运行使用时间?

使用 `system.time()` 。 `proc.time()` 可以获得 `R` 进程存在的时间, `system.time()` 通过调用两次 `proc.time()` 来计算函数运行的时间。

### 如何释放 **R** 运行后占用的内存?

使用函数 `gc()` 。因为 `R` 是在内存中运算, 所以当 `R` 读入了体积比较大的数据后, 即使删除了相关对象, 内存空间仍不能释放。 `gc()` 函数虽然主要用来报告内存使用情况, 但是一个重要的用途便是释放内存。

### 用什么文本编辑器比较好?

比较常用的是 Tinn-R , **RWinEdt** <sup>4</sup>, **ESS**(Emacs Speaks Statistics) , 甚至任意一款编辑器, 如 UltraEdit<sup>5</sup>, 这些都支持 R 语法的高亮显示。如果是 Windows 桌面环境下的用户, 对这些不是很了解, 记事本也不失为一种选择。

---

<sup>4</sup>下载、安装 WinEdt 后, 在 R 中安装 RWinEdt 包即可使用

<sup>5</sup>需要下载、修改 wordfile

## 索引

Symbols		
\\	7	Commander . . . . . 19
.....	11	contour . . . . . 16
.packages	4	cumsum . . . . . 8, 15
/	7	curve . . . . . 16
^	10, 18	
{}	9	
<b>A</b>		<b>D</b>
abline	13	D . . . . . 11
aperm	8	data . . . . . 4
apply	8, 11, 16	data.frame . . . . . 4, 9
as.Date	12	data.frame . . . . . 8
as.numeric	13	demo . . . . . 2
as.POSIXct	12	detach . . . . . 4
axes	13	dev.copy . . . . . 15
axis	13	dev.cur . . . . . 14
<b>B</b>		dev.list . . . . . 14
barplot	15	dev.off . . . . . 14
boxcox	19	dev.set . . . . . 14
bptest	19	Devices . . . . . 14
Breusch-Pagan	19	diag . . . . . 10
<b>C</b>		difftime . . . . . 12
car	5, 19	dim . . . . . 8
casefold	11	uplicated . . . . . 8
choose	11	durbin.watson . . . . . 19
citation	2	dwtest . . . . . 19
clipboard	6	
cluster	18	<b>E</b>
cm.colors	15	edit . . . . . 6
col	15	eigen . . . . . 10
col2rgb	15	ESS . . . . . 20
colMeans	11	eval . . . . . 7
colors	14	
combn	11	<b>F</b>
		factanal . . . . . 18
		factorial . . . . . 11
		FALSE . . . . . 9
		file.choose . . . . . 7
		filled.contour . . . . . 16

fix .....	6	Library .....	4
format .....	12	library .....	3, 4
<b>G</b>			
gc .....	19	lines .....	16
getAnywhere .....	5	list .....	3
getwd .....	4	lm .....	18
gray .....	15	lmtest .....	19
grep .....	11	loadings .....	18
grey .....	15	lower.tri .....	10
<b>H</b>			
hclust .....	17	ls .....	3
head .....	5	lstest .....	19
heat.colors .....	15	<b>M</b>	
help .....	3	mar .....	12
help.search .....	3	matrix .....	8, 11
<b>I</b>			
I .....	18	merge .....	9
if .....	9	methods .....	5
image .....	16	<b>N</b>	
integer .....	10	nchar .....	12
integrate .....	10	ncv.test .....	19
iris .....	13	NULL .....	8
is.na .....	7	<b>O</b>	
is.numeric .....	9	options .....	3
<b>J</b>			
jarque.bera.test .....	18	<b>P</b>	
jpeg .....	15	Package .....	4
<b>K</b>			
kmeans .....	17	par .....	12, 13, 16
ks.test .....	18	parse .....	7
<b>L</b>			
latex .....	6	paste .....	3, 11
layout .....	12	pch .....	13
legend .....	13	pdf .....	15
letters .....	15	persp .....	16
<b>M</b>			
Library .....	4	pictex .....	15
library .....	3, 4	pie .....	13
lines .....	16	plotmath .....	16
list .....	3	pmax .....	9
lm .....	18	pmin .....	9
lmtest .....	19	png .....	15
loadings .....	18	postscript .....	15
lower.tri .....	10		
ls .....	3		
lstest .....	19		

princomp .....	18	sink .....	6
print .....	6	some .....	5
proc.time .....	19	sort .....	13
prompt .....	3	source .....	6
<b>Q</b>			
qqnorm .....	16	spline .....	16
qr.solve .....	19	split.screen .....	12
Quantile Regression .....	17	sqrt .....	10
<b>R</b>			
rainbow .....	15	stars .....	14
Rcmdr .....	19	stats .....	18
read.table .....	6	stem .....	13
read.xls .....	5	subset .....	7
Regular Expressions .....	8, 11	system .....	3
rev .....	9	system.time .....	11, 19
rgb .....	14	<b>T</b>	
rm .....	3	t .....	10
RMySQL .....	7	table .....	17
rnorm .....	13, 15, 17	tail .....	5
RODBC .....	7	tapply .....	9
rowMeans .....	11	terrain.colors .....	15
Rpad .....	19	Tinn-R .....	20
rpart .....	19	tolower .....	11
RSiteSearch .....	3	topo.colors .....	15
RWinEdt .....	20	toupper .....	11
<b>S</b>			
sample .....	9	transform .....	9
save .....	4	tseries .....	18
save.image .....	4	<b>U</b>	
savePlot .....	15	unique .....	8
scale .....	9	update .....	17
screeplot .....	18	update.packages .....	3
sem .....	18	upper.tri .....	10
setwd .....	4	<b>W</b>	
shapiro.test .....	18	windows .....	15
shell.exec .....	3	with .....	13
show.error.messages .....	3	<b>X</b>	
		xlsReadWrite .....	5
		xtable .....	6