R讲义：R基本语法

刘念夏

2025-09-19

Table of contents

# 初识RStudio介面

当我们从互联网下载安装好R和RStudio之后，就可以开始使用R进行数据分析了。R是一种开源的统计编程语言，而RStudio是一个强大的集成开发环境(IDE)。一般我们会使用RStudio来编写和运行R代码。

RStudio提供了一个用户友好的界面来编写和运行R代码，适合初学者和专业人士使用。RStudio的界面主要分为四个窗格(如图1-1)： 1.代码语法/脚本编写窗格(左上角)：用于编写和编辑R脚本。 2.控制台窗格(左下角)：用于直接输入和执行R命令。 3.环境设置/历史纪录窗格(右上角)：显示当前工作空间中的对象和变量，以及历史命令。 4.档案目录/图形输出/套件安装/线上协助窗格(右下角)：用于浏览和管理文件系统中的文件，查看R函数和包的文档，显示绘图结果，以及安装、加载和管理R套件包。

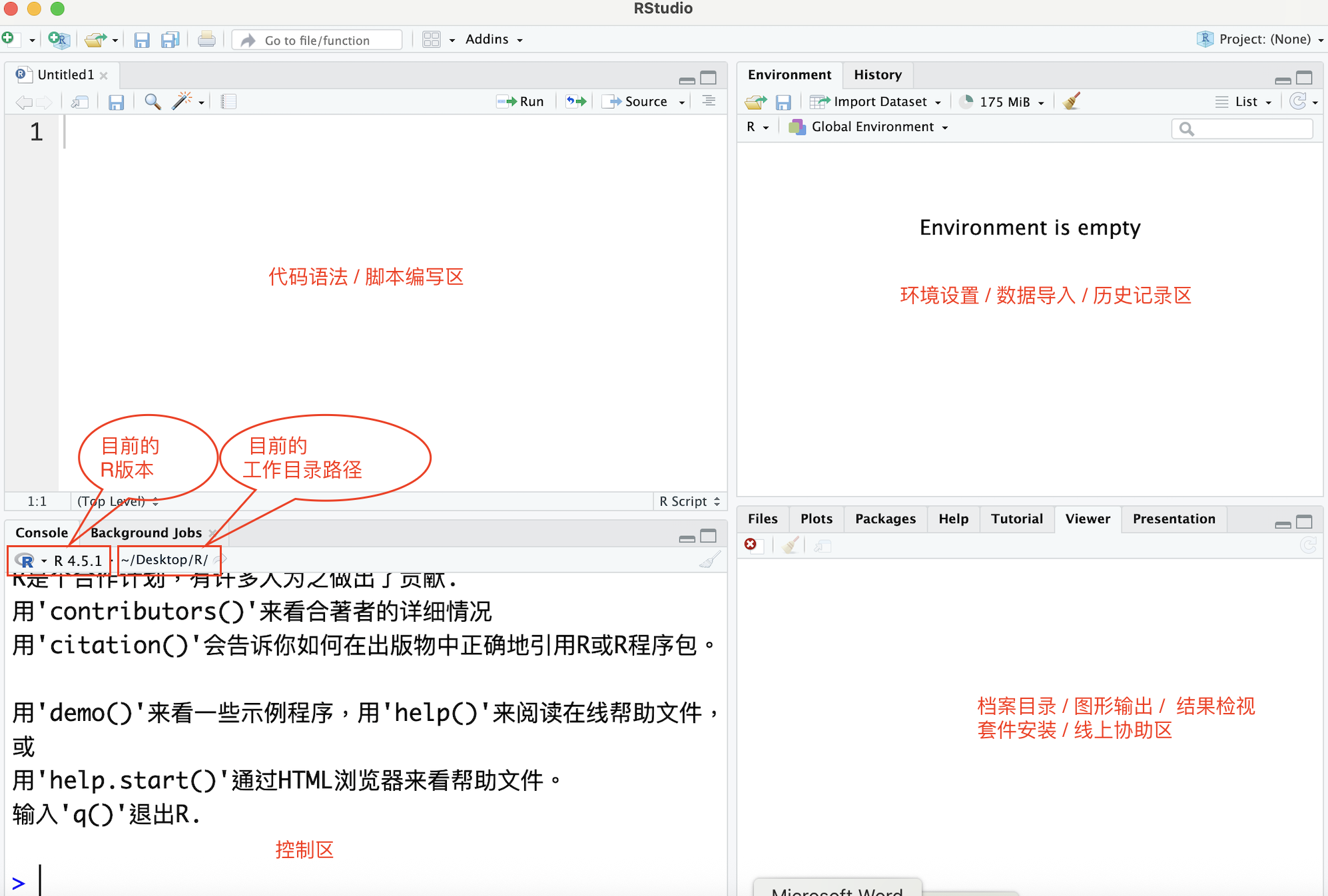


图1-1

R的版本信息可以通过RStudio左下角的控制台窗口(Console)查看。RStudio会自动检测并显示当前安装的R版本。R的版本信息包括版本号、发布日期等。通过控制台窗口(Console)输入 “version”， 即可查看当前安装的R版本信息。输出结果显示，我们目前使用的R是4.5.1版(2025-06-13版)。由于R的版本每隔一段时间就会更新一次，因此建议用户定期检查并更新R的版本，以获得最新的功能和修复。

version #查看R的版本信息

\_   
platform aarch64-apple-darwin20   
arch aarch64   
os darwin20   
system aarch64, darwin20   
status   
major 4   
minor 5.1   
year 2025   
month 06   
day 13   
svn rev 88306   
language R   
version.string R version 4.5.1 (2025-06-13)  
nickname Great Square Root

# 当作计算机

R 语言不仅是一种统计编程语言，也可以当作计算机来使用。R可以执行各种数学运算和逻辑运算，支持基本的四则运算、指数运算、平方根等。我们可以在RStudio的控制台窗口(Console)中直接输入数学表达式，按回车键后，R会自动计算并返回结果。

例如：

在控制台输入：

3 + 8 #按回车后，得到"11"

[1] 11

3 - 8 #按回车后，得到"-5"

[1] -5

3+8+34+98+34+45+56+45-34-42 #按回车后，得到"247"

[1] 247

3 \* 8 #按回车后，得到"24"

[1] 24

3 / 8 #按回车后，得到"0.375"

[1] 0.375

3 ^ 8 #(3的8次方)按回车后，得到"6561"

[1] 6561

sqrt(3) #sqrt(square root, sqrt平方根)，

[1] 1.732051

#(3的平方根)按回车后，得到"1.732051"

控制台窗口(Console)的命令列，一次只能输入一行代码，如果使用者一次要执行比较多行的代码(脚本)就会比较不方便；这时我们一般会改在 “代码语法/脚本编写窗格”(左上角)编写R代码脚本并执行代码。执行代码可以通过点击 “Run” 按钮，或使用快捷键 Ctrl + Enter (Windows) 或 Cmd + Enter (Mac) 来完成。执行的结果会在左下角的 “控制台窗格” (Console)(左下角)或右下角的 “图形输出窗口”(Plots)或 “检视窗口(Viewer)”中显示。

# R的基础套件包

R语言安奘后之后会自带一组基础套件包(packages)，提供了R的核心功能和基本数据分析/绘图处理能力，以下是R默认安装的主要基础套件包：

* base: R的核心包，包含基本的函数和数据结构。
* datasets: 内置的示例数据集（如iris、mtcars）
* grDevices: 图形设备支持，用于绘图和图形输出(如生成PNG、PDF等)。
* graphics: 基础绘图系统，提供绘图函数（如plot、hist等）。
* methods: 面向对象编程（如S4类、方法分派等）。
* stats: 统计分析函数（如线性回归lm、t检验t.test等）。
* utils: 实用工具（如数据导入导出、文件操作、套件包管理等）。

这些基础套件包是R核心功能的一部分，无需额外安装即可直接使用。

# 安装与加载R的套件包

## 安装套件包

除了基础套件包外，R的功能可以通过安装和加载扩展套件包来进一步增强。 R拥有一个庞大的社区和生态系统，提供了数以万计的扩展套件包，这些包可以帮助用户更方便地进行数据分析、可视化等任务。

要安装套件包，首先要先确定电脑有连上互联网，然后在左上角的 “代码” 窗格，使用 “install.packages(”package\_name”)“命令，输入要安装的套件名称。例如，如果要安装”ggplot2” 套件包(R著名的绘图套件包)，我们可以在RStudio左上角的 “代码”窗格，输入以下代码：

#install.packages("ggplot2") #安装ggplot2套件包

然后用鼠标全选该行代码，并用鼠标点选”Run”按钮(如图1-2)，或使用快捷键 Ctrl + Enter (Windows) 或 Cmd + Enter (Mac) 来执行该行代码。R会自动从CRAN（Comprehensive R Archive Network）下载并安装该套件包。

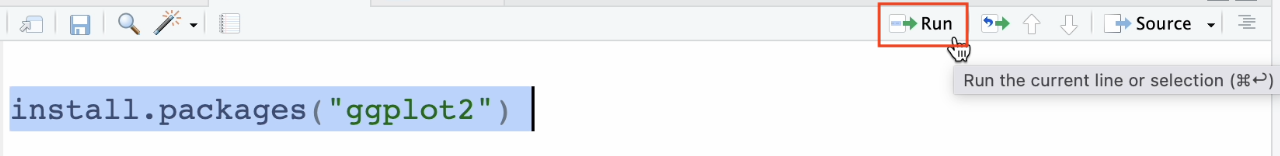


图1-2

## 加载套件包

安装完成套件包后，我们还要进一步的 “加载” (library)该套件包才能运作。以加载 “ggplot2” 套件包为例，我们可以在RStudio左上角的 “代码”窗格，输入以下代码：

library(ggplot2) #加载ggplot2套件包

一样用鼠标全选该行代码，并用鼠标点选 “Run” 按钮，就可以在RStudio中使用该套件包提供的函数和功能了。

## 检查与安装套件包

另外，我们可以使用 if (!require(“package\_name”)) install.packages(“package\_name”) 的方式来检查套件包是否已安装，如果已安装过会自动跳过不安装，如果未安装过则会自动安装。例如：我们想检查 “haven”这个套件包是否已安装， 可以在”代码”窗格，输入以下代码：

if (!require("haven")) install.packages("haven") #检查并安装haven套件包

载入需要的程序包：haven

如此就可以确保 “haven”套件包已安装并加载到R环境中。

## 使用 “Packages” 标签页来安装(Install)套件包

除了用写代码的方式安装套件包外，RStudio还提供了一个图形化界面来安装和管理套件包。我们可以在右下角的窗格中，点击 “Packages” 标签页，然后在搜索框中输入要安装的套件包名称，点击 “Install” 按钮即可。例如：如果要安装 “dplyr” 套件包，我们可以在 “Packages” 标签页中的Install窗格，搜索 “dplyr”，然后点击 “Install” 按钮，RStudio便会自动下载并安装该套件包(如图1-3)。

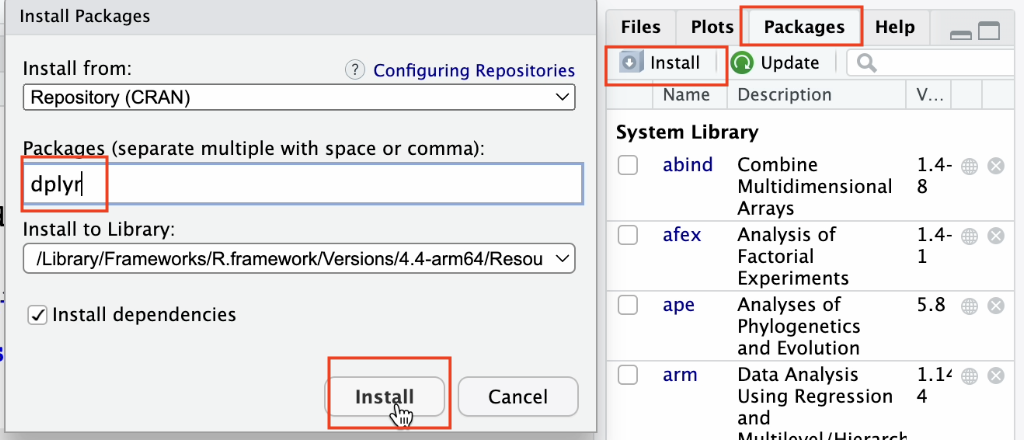


图1-3

## 使用 “Packages” 标签页来加载(Library)套件包

当然，我们也可以使用菜单点选的方式来加载套件包。 在RStudio右下角的窗格中，点击 “Packages” 标签页，然后勾选要加载的套件包名称即可。 例如：如果要加载 “dplyr” 套件包，我们可以在 “Packages” 标签页中的搜寻方框，输入 “dplyr”，然后按一下回车键，如果电脑之前已装过 “dplyr” 套件包，此时 “dplyr” 套件包 就会出现，我们只要在套件包前面的核取方块中，勾选 “dplyr”，就可以加载该套件包了(如图1-4)。RStudio1便会自动加载该套件包，并在控制台窗口(Console)中显示加载成功的信息。

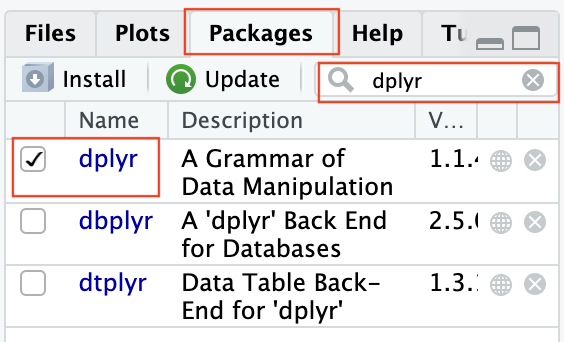


图1-4

## 查看套件包的使用方式

R的套件包通常会提供详细的文档和使用说明，用户可以通过查看套件包的帮助文档来了解如何使用该套件包中的函数和功能。 如果我们想查看某个套件包的具体使用方式，可以在RStudio左下角的控制台窗格(Console)中，或左上角的代码窗格输入 “help(package = ‘package\_name’)”，例如： help(package = ‘dplyr’)，就可以线上查看 “dplyr” 套件包的使用方式和文档。

help(package = 'dplyr') #查看dplyr套件包的使用方式

# 环境设置

当我们开始进行数据分析时，我们需要先进行一些环境设置，包括设置工作目录、查看工作目录、设置系统中文编码、设置图形中文呈现等。

## 设置工作目录

在R中，工作目录是指R当前操作的文件夹路径，所有经过R代码产生的文件与图表，都会被系统放在该文件夹。此处，我们建议使用 “菜单” 点选的方式，来设置工作目录。例如，如果我们要将工作目录设置为 “Desktop/R量化研究数据分析”，可以先点选RStudio最上面菜单的 “Session”标签，继之选取 “Set Working Directory”，而后再点选 “Choose Directory”，最后再将鼠标移到你所要设置的工作目录文件夹，最后再点选 “Open”按钮即可(如图1-5)。

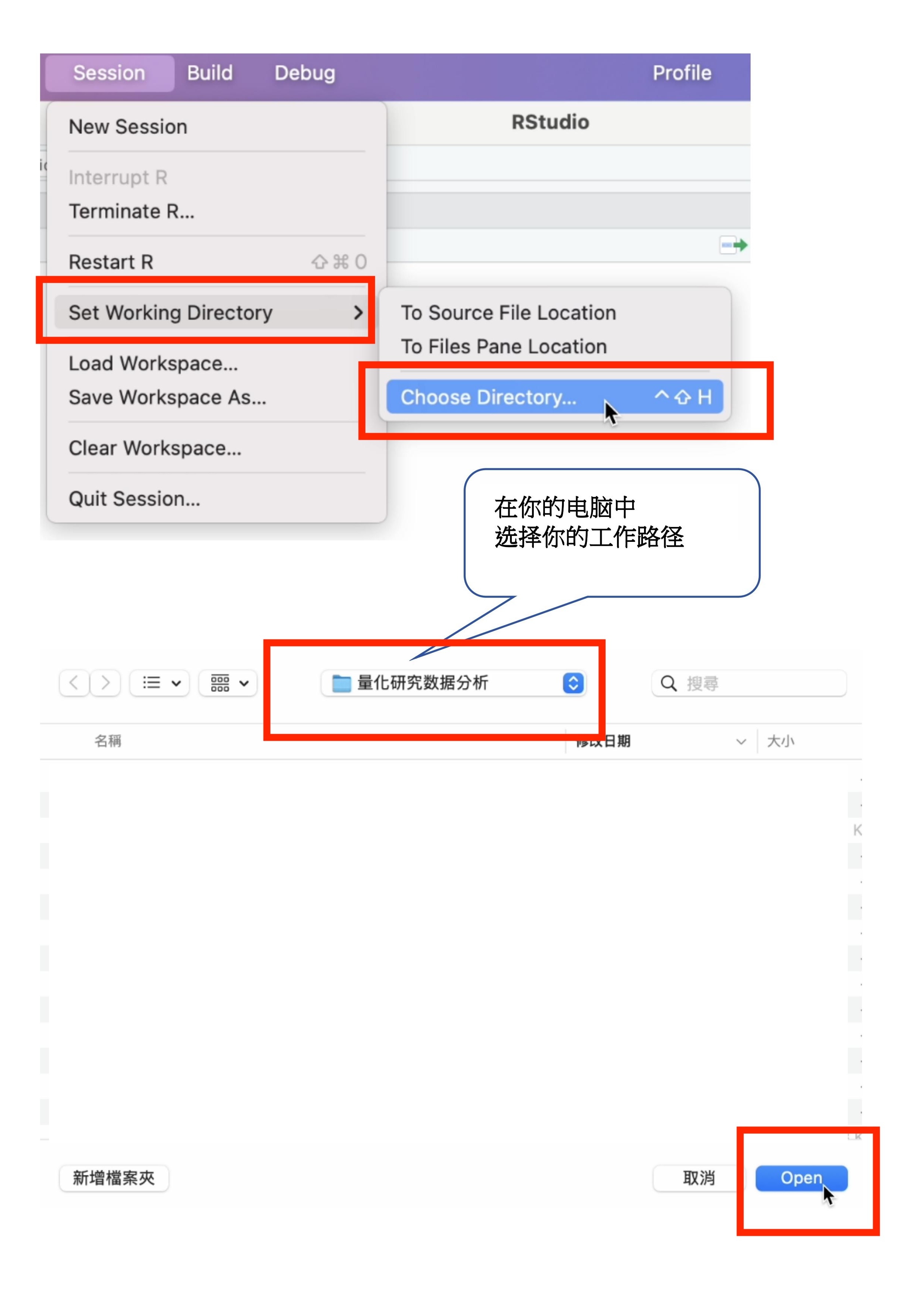


图1-5

点选 “Open”按钮后，RStudio便会将工作目录设置为你所选择的文件夹路径，并在左下角的控制台窗格(Console)中显示具体的工作目录路径。我们再将该路径复制下来，复制到左上角的代码窗格(Source)，然后用鼠标全选该行代码，并用鼠标点选 “Run” 按钮，即完成工作目录的设置。

setwd("~/Desktop/量化研究数据分析/R Ch1 R基本语法") #设置工作目录

## 查看工作目录

在设置好工作目录之后，可以使用 “getwd( )” 函数来查看当前的工作目录路径。

getwd( ) #查看当前工作目录

[1] "/Users/simonfair/Desktop/量化研究数据分析/R Ch1 R基本语法"

## 设置系统中文编码

R默认的文字编码可能不支持中文字符，因此我们需要设置系统的中文文字编码，以确保R能够正确处理中文字符。可以使用 “Sys.setlocale( )” 函数来设置系统的中文文字编码。设置方法如下：

#设置系统中文编码(简体中文)  
Sys.setlocale(category = "LC\_ALL", locale = "zh\_CN.UTF-8")

[1] "zh\_CN.UTF-8/zh\_CN.UTF-8/zh\_CN.UTF-8/C/zh\_CN.UTF-8/zh\_CN.UTF-8"

## 设置图形中文呈现

此外，R的图形输出默认也可能不支持中文字符，因此我们需要设置图形的中文呈现，以确保图形中的中文文字能够正确显示。可以使用 “showtext” 套件包来实现中文字体的支持。首先，我们需要安装(install)并加载(library) “showtext” 套件包，然后使用 “showtext\_auto( )” 函数来启用中文字体支持。

#检查并安装showtext套件包  
if (!require("showtext")) install.packages("showtext")

载入需要的程序包：showtext

载入需要的程序包：sysfonts

载入需要的程序包：showtextdb

library(showtext) #加载"showtext"套件包  
#使绘图物件中的中文文字能正确呈现  
showtext\_auto(enable = TRUE)

# 加载R内置数据集

在R中，我们可以使用内置数据集来进行数据分析和练习。R自带了一些常用的内置数据集，例如： “mtcars”数据集。 “mtcars” 数据集包含了1974年美国汽车的各种性能指标。我们可以使用 “data(mtcars)” 函数来加载该数据集。

data(mtcars) #加载内置数据集mtcars

# 查看R数据集内容的基本语法

R是一种功能强大的编程语言，广泛用于统计分析和数据科学。R的基本语法包括变量赋值、数据类型、控制结构、函数定义等。以下是一些查看R数据档基本语法的介绍。

## 查看数据集的外观

使用 “View( )” 函数， 可以查看数据集的外观(如图1-6)。

View(mtcars)

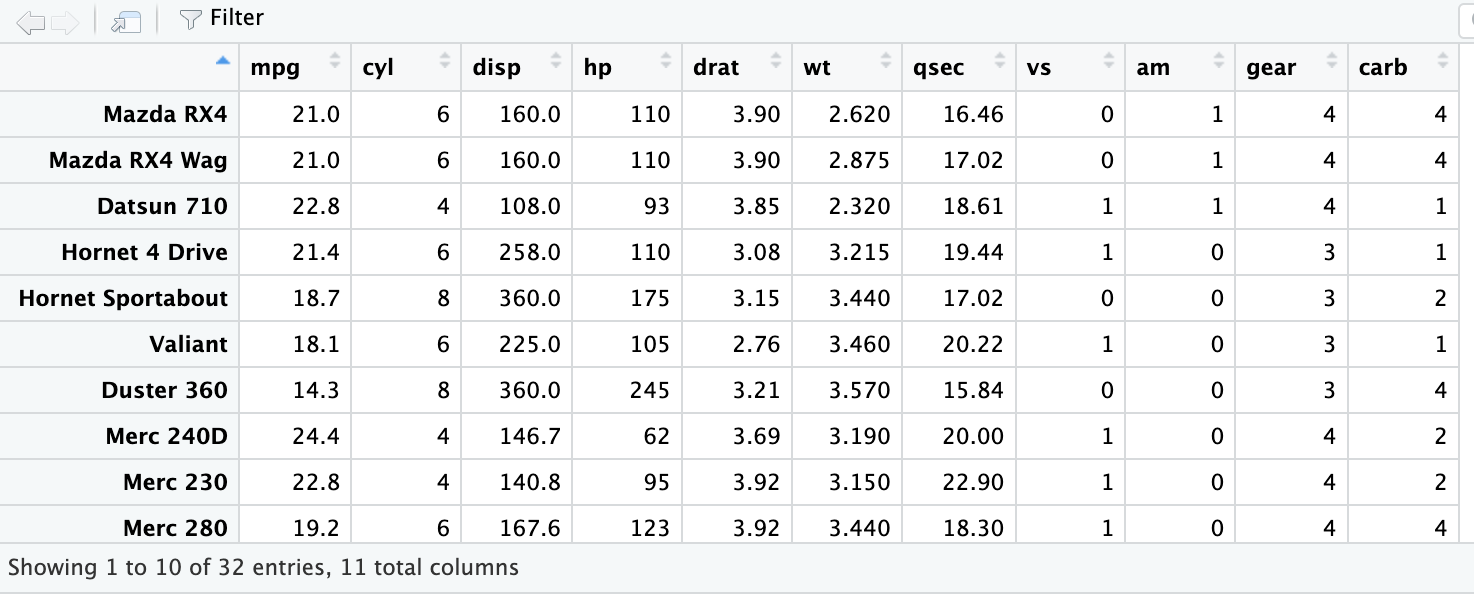


图1-6

## 查看数据集的维度

使用 “dim( )”函数， 可以查看数据集的维度，即数据集包含多少行和多少列。

dim(mtcars) #查看mtcars数据集的维度(32行数和11列数)

[1] 32 11

## 查看数据集的行数

使用 “nrow( )” 函数，可以查看数据集的行数。

nrow(mtcars) #查看mtcars数据集的行数(32行)

[1] 32

## 查看数据集的列数

使用 “ncol( )” 函数，可以查看数据集的列数。

ncol(mtcars) #查看mtcars数据集的列数(11列)

[1] 11

## 查看数据集的行名

使用 “rownames( )” 函数，可以查看数据集的行名。

rownames(mtcars) #查看mtcars数据集的行名

[1] "Mazda RX4" "Mazda RX4 Wag" "Datsun 710"   
 [4] "Hornet 4 Drive" "Hornet Sportabout" "Valiant"   
 [7] "Duster 360" "Merc 240D" "Merc 230"   
[10] "Merc 280" "Merc 280C" "Merc 450SE"   
[13] "Merc 450SL" "Merc 450SLC" "Cadillac Fleetwood"   
[16] "Lincoln Continental" "Chrysler Imperial" "Fiat 128"   
[19] "Honda Civic" "Toyota Corolla" "Toyota Corona"   
[22] "Dodge Challenger" "AMC Javelin" "Camaro Z28"   
[25] "Pontiac Firebird" "Fiat X1-9" "Porsche 914-2"   
[28] "Lotus Europa" "Ford Pantera L" "Ferrari Dino"   
[31] "Maserati Bora" "Volvo 142E"

## 查看数据集的列名

使用 “colnames( )” 函数，可以查看数据集的列名(变量名称)。 或使用 “names( )” 函数， 也可以查看数据集的变量名称。

colnames(mtcars) #查看mtcars数据集的列名

[1] "mpg" "cyl" "disp" "hp" "drat" "wt" "qsec" "vs" "am" "gear"  
[11] "carb"

names(mtcars) #查看mtcars数据集的变量名称

[1] "mpg" "cyl" "disp" "hp" "drat" "wt" "qsec" "vs" "am" "gear"  
[11] "carb"

## 查看数据集的类型

使用 “class( )” 函数，可以查看数据集的类型。R中的数据集通常是一个数据框(data.frame)，它是一种二维表格结构，类似于电子表格或数据库表格。

class(mtcars) #查看mtcars数据集的类型

[1] "data.frame"

## 查看数据集的属性

使用 “attributes( )” 函数， 可以同时查看数据集的属性，包括行名(row.names)、列名(names)与数据集的类型(class)等。

attributes(mtcars) #查看mtcars数据集的属性

$names  
 [1] "mpg" "cyl" "disp" "hp" "drat" "wt" "qsec" "vs" "am" "gear"  
[11] "carb"  
  
$row.names  
 [1] "Mazda RX4" "Mazda RX4 Wag" "Datsun 710"   
 [4] "Hornet 4 Drive" "Hornet Sportabout" "Valiant"   
 [7] "Duster 360" "Merc 240D" "Merc 230"   
[10] "Merc 280" "Merc 280C" "Merc 450SE"   
[13] "Merc 450SL" "Merc 450SLC" "Cadillac Fleetwood"   
[16] "Lincoln Continental" "Chrysler Imperial" "Fiat 128"   
[19] "Honda Civic" "Toyota Corolla" "Toyota Corona"   
[22] "Dodge Challenger" "AMC Javelin" "Camaro Z28"   
[25] "Pontiac Firebird" "Fiat X1-9" "Porsche 914-2"   
[28] "Lotus Europa" "Ford Pantera L" "Ferrari Dino"   
[31] "Maserati Bora" "Volvo 142E"   
  
$class  
[1] "data.frame"

## 查看数据集的结构

使用 “str( )” 函数，可以查看数据集的结构，包括每个变量的类型和前几行数据。

str(mtcars) #查看mtcars数据集的结构

'data.frame': 32 obs. of 11 variables:  
 $ mpg : num 21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...  
 $ cyl : num 6 6 4 6 8 6 8 4 4 6 ...  
 $ disp: num 160 160 108 258 360 ...  
 $ hp : num 110 110 93 110 175 105 245 62 95 123 ...  
 $ drat: num 3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...  
 $ wt : num 2.62 2.88 2.32 3.21 3.44 ...  
 $ qsec: num 16.5 17 18.6 19.4 17 ...  
 $ vs : num 0 0 1 1 0 1 0 1 1 1 ...  
 $ am : num 1 1 1 0 0 0 0 0 0 0 ...  
 $ gear: num 4 4 4 3 3 3 3 4 4 4 ...  
 $ carb: num 4 4 1 1 2 1 4 2 2 4 ...

## 查看数据集的前几行

使用 “head( )” 函数，可以查看数据集的前几行数据，默认显示前6行。

head(mtcars) #查看mtcars数据集的前6行

mpg cyl disp hp drat wt qsec vs am gear carb  
Mazda RX4 21.0 6 160 110 3.90 2.620 16.46 0 1 4 4  
Mazda RX4 Wag 21.0 6 160 110 3.90 2.875 17.02 0 1 4 4  
Datsun 710 22.8 4 108 93 3.85 2.320 18.61 1 1 4 1  
Hornet 4 Drive 21.4 6 258 110 3.08 3.215 19.44 1 0 3 1  
Hornet Sportabout 18.7 8 360 175 3.15 3.440 17.02 0 0 3 2  
Valiant 18.1 6 225 105 2.76 3.460 20.22 1 0 3 1

## 查看数据集的后几行

使用 “tail( )” 函数，可以查看数据集的后几行数据，默认显示后6行。

tail(mtcars) #查看mtcars数据集的后6行

mpg cyl disp hp drat wt qsec vs am gear carb  
Porsche 914-2 26.0 4 120.3 91 4.43 2.140 16.7 0 1 5 2  
Lotus Europa 30.4 4 95.1 113 3.77 1.513 16.9 1 1 5 2  
Ford Pantera L 15.8 8 351.0 264 4.22 3.170 14.5 0 1 5 4  
Ferrari Dino 19.7 6 145.0 175 3.62 2.770 15.5 0 1 5 6  
Maserati Bora 15.0 8 301.0 335 3.54 3.570 14.6 0 1 5 8  
Volvo 142E 21.4 4 121.0 109 4.11 2.780 18.6 1 1 4 2

## 保存与载入R数据资料档(\*.rda)

在R中，我们可以将数据集保存为R数据资料档格式（\*.rda），以便后续使用。可以使用 “save( )” 函数来保存数据集，使用 “load( )” 函数来载入数据集。

save(mtcars, file = "mtcars.rda") #将mtcars数据集保存为R数据档  
load("mtcars.rda") #载入mtcars.rda数据档

# 安装 “sj-family” 套件包

“sj-family” 套件包是一个集合了多个R套件包的集合，包含了: – “sjlabelled”(主要作为外部数据导入/汇出,变量/变量值的标签设置,变量属性的转换之用), – “sjmisc”(主要作为数据处理、变量转换、数据清理之用), – “sjstats”(主要作为统计分析、模型诊断之用), – “sjPlot”(主要作为数据可视化、模型结果展示之用), – “ggeffects”(主要作为回归模型预测结果的可视化之用)等多个套件包。 这些套件包提供了丰富的功能，用于数据分析、可视化和统计建模等任务，本书作者强烈建议使用者下载安装。以下为安装步骤：

1. 首先，确保已经安装了 “devtools” 套件包，如果没有安装，可以使用以下代码安装：

if (!require("devtools")) install.packages("devtools")

载入需要的程序包：devtools

载入需要的程序包：usethis

1. 然后，使用 “devtools” 套件包的 “install\_github( )” 函数来安装 “sj-family” 套件包。

devtools::install\_github("strengejacke/strengejacke")

Skipping install of 'strengejacke' from a github remote, the SHA1 (f2c47216) has not changed since last install.  
 Use `force = TRUE` to force installation

安装成功后，RSudio左下角的控制视窗(Console)会呈现这些套件包的最新安装版本信息。套件开发者会不定期更新优化这些套件包，因此建议使用者定期检查并更新套件包，以获得最新的功能和修复。

以下的各章中，我们将会使用到 “sj-family” 套件包中的各个子套件包来进行数据分析的工作。

# 清除R的记忆体中所有物件与资料

在R中，我们可以使用 “rm(list = ls( ))” 函数来清除当前工作环境中的所有对象和数据，作为下一次分析的准备工作。

rm(list = ls()) #清除记忆体中所有物件与资料

# 本章小结

本章所使用到的R套件包与函数摘录如下表。

| 套件包 | 函数 | 说明 |
| --- | --- | --- |
| 内建 | version( ) | 查看R的版本信息 |
|  | install.packages( ) | 安装套件包 |
|  | library( ) | 加载套件包 |
|  | if (!require( )) install.packages( ) | 检查并安装套件包 |
|  | help( ) | 查看套件包的使用方式 |
|  | setwd ( ) | 设置当前工作目录 |
|  | getwd ( ) | 显示当前工作目录 |
|  | Sys.setlocale ( ) | 设置系统中文编码 |
|  | data( ) | 加载R内建数据集 |
|  | View ( ) | 查看数据集的外观 |
|  | dim( ) | 查看数据集的维度 |
|  | nrow( ) | 查看数据集的行数 |
|  | ncol( ) | 查看数据集的列数 |
|  | rownames( ) | 查看数据集的行名 |
|  | colnames( ) | 查看数据集的列名 |
|  | names( ) | 查看数据集的变量名称 |
|  | class( ) | 查看数据集的类型 |
|  | attributes( ) | 查看数据集的属性 |
|  | str( ) | 查看数据集的结构 |
|  | head( ) | 查看数据集的前几行 |
|  | tail( ) | 查看数据集的后几行 |
|  | save( ) | 保存数据集为R数据资料档格式(\*.rda) |
|  | load( ) | 载入R数据资料档(\*.rda) |
|  | rm (list=ls( )) | 去除R当前工作环境中所有的数据与物件 |
| showtext | showtext\_auto ( ) | 使图形能正确呈现中文 |
| devtools | install\_github ( ) | 从GitHub安装套件包 |