

# Lab R1:A Brief Introduction to R

**Haocheng Hu and Zhaopeng Qu**

**Nanjing University**

*2/26/2025*

- 1 Introduction to R
- 2 How to learn R
- 3 Using IDE: RStudio
- 4 Directory Management

# Section 1

## Introduction to R

# What is R ?

- Not only a statistical programming language, but a computing environment for statistical computing and graphics.

# What is R ?

- Not only a statistical programming language, but a computing environment for statistical computing and graphics.
- The **R** language has its *roots* in the *S* language developed by *AT&T*, which also developed the *C* language. It is not the only domain-specific language available for statistical analysis: there are [many others][[boc-comp](#)] like **SAS**, **SPSS** or **Stata**, or even applications of statistics with mathematical software and scripting languages like **Java** or **Python**.

# Why R: A Free but Powerful tool

- **Free and Open source**

# Why R: A Free but Powerful tool

- **Free and Open source**
- **Powerful program and brilliant visualization**

# Why R: A Free but Powerful tool

- **Free and Open source**
- **Powerful program and brilliant visualization**
- **Popularity**



# Why R: A Free but Powerful tool




















- **Free and Open source**
- **Powerful program and brilliant visualization**
- **Popularity**
  - Used across wide variety of disciplines both academics and business.

# Why R: A Free but Powerful tool

- **Free and Open source**
- **Powerful program and brilliant visualization**
- **Popularity**
  - Used across wide variety of disciplines both academics and business.
  - tremendous online resources: books, blogs, forums, videos and online courses.

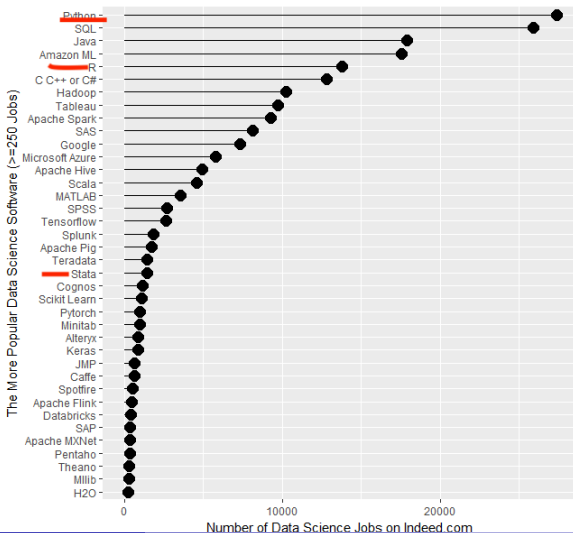
# Why R: Top 5 Programming Languages in 2019

- Top 10 Programming Languages in 2019 by IEEE

Rank	Language	Type	Score
1	Python	  	100.0
2	Java	  	96.3
3	C	  	94.4
4	C++	  	87.5
5	R		81.5
6	JavaScript		79.4
7	C#	   	74.5
8	Matlab		70.6

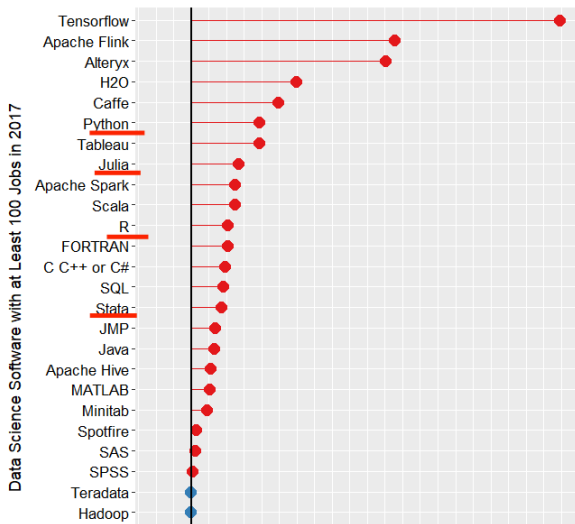
# Why R: Top 5 Software in 2019

- The Popularity of Data Science Software in 2019 by Jobs



# Why R: Top 11 increase during 2017-2019

- The Change during 2017-2019 by Jobs



# Wrap up

- The number-crunching language R rounds out the top 5.

# Wrap up

- The number-crunching language R rounds out the top 5.
- Despite being a much more specialized language than the others, it's maintained its popularity in recent years due to the world being awash in an ever-growing pile of big data.

# Setup

- Default your operating system environment



# Setup

- Default your operating system environment
  - Windows: Win7 or Win10(preferred)

# Setup

- Default your operating system environment
  - Windows: Win7 or Win10(preferred)
  - Mac: macOS 12 Monterey

# Setup

- Default your operating system environment
  - Windows: Win7 or Win10(preferred)
  - Mac: macOS 12 Monterey
- Installing

# Setup

- Default your operating system environment
  - Windows: Win7 or Win10(preferred)
  - Mac: macOS 12 Monterey
- Installing
  - R

# Setup

- Default your operating system environment
  - Windows: Win7 or Win10(preferred)
  - Mac: macOS 12 Monterey
- Installing
  - R
  - RStudio

# Setup

- Default your operating system environment
  - Windows: Win7 or Win10(preferred)
  - Mac: macOS 12 Monterey
- Installing
  - R
  - RStudio
  - Latex(not required)

# Setup

- Default your operating system environment
  - Windows: Win7 or Win10(preferred)
  - Mac: macOS 12 Monterey
- Installing
  - R
  - RStudio
  - Latex(not required)
  - Pandoc(not required)

## Installing R: 注意事项 (for Windows)

- 如果没有按照注意事项安装，最好先卸载，重新安装最新版本。



## Installing R: 注意事项 (for Windows)

- 如果没有按照注意事项安装，最好先卸载，重新安装最新版本。
- 安装路径里不要有中文

## Installing R: 注意事项 (for Windows)

- 如果没有按照注意事项安装，最好先卸载，重新安装最新版本。
- **安装路径里不要有中文**
  - 因为可能会涉及语言编码问题，后面更新包容易出错。

## Installing R: 注意事项 (for Windows)

- 如果没有按照注意事项安装，最好先卸载，重新安装最新版本。
- 安装路径里不要有中文
  - 因为可能会涉及语言编码问题，后面更新包容易出错。
- 选择安装语言时，最好选择 **English**，而不是中文（如果是系统默认的，后面最好也调整成英文）

## Installing R: 注意事项 (for Windows)

- 如果没有按照注意事项安装，最好先卸载，重新安装最新版本。
- 安装路径里不要有中文
  - 因为可能会涉及语言编码问题，后面更新包容易出错。
- 选择安装语言时，最好选择 **English**，而不是中文（如果是系统默认的，后面最好也调整成英文）
- 安装目录中去掉 R 的版本号（应付新旧版本 R 的兼容问题）

## Installing R: 注意事项 (for Windows)

- 如果没有按照注意事项安装，最好先卸载，重新安装最新版本。
- 安装路径里不要有中文
  - 因为可能会涉及语言编码问题，后面更新包容易出错。
- 选择安装语言时，最好选择 **English**，而不是中文（如果是系统默认的，后面最好也调整成英文）
- 安装目录中去掉 R 的版本号（应付新旧版本 R 的兼容问题）
  - 比如不要安装在 `C:/Program Files/R/R-3.6.1/`，而是把它改成 `C:/Program Files/R/`，这样更新就会自动覆盖之前的版本。

## Installing R: 注意事项 (for Windows)

- 如果没有按照注意事项安装，最好先卸载，重新安装最新版本。
- 安装路径里不要有中文
  - 因为可能会涉及语言编码问题，后面更新包容易出错。
- 选择安装语言时，最好选择 **English**，而不是中文（如果是系统默认的，后面最好也调整成英文）
- 安装目录中去掉 R 的版本号（应付新旧版本 R 的兼容问题）
  - 比如不要安装在 `C:/Program Files/R/R-3.6.1/`，而是把它改成 `C:/Program Files/R/`，这样更新就会自动覆盖之前的版本。
- 另外建议最好换个目录安装，不要在 `C:/Program Files/` 或者干脆不在 C 盘，比如 `D:/Application/R/`

## Installing R: 注意事项 (for Windows)

- 如果没有按照注意事项安装，最好先卸载，重新安装最新版本。
- 安装路径里不要有中文
  - 因为可能会涉及语言编码问题，后面更新包容易出错。
- 选择安装语言时，最好选择 **English**，而不是中文（如果是系统默认的，后面最好也调整成英文）
- 安装目录中去掉 R 的版本号（应付新旧版本 R 的兼容问题）
  - 比如不要安装在 `C:/Program Files/R/R-3.6.1/`，而是把它改成 `C:/Program Files/R/`，这样更新就会自动覆盖之前的版本。
- 另外建议最好换个目录安装，不要在 `C:/Program Files/` 或者干脆不在 C 盘，比如 `D:/Application/R/`
  - 这样不会因为在使用 R 需要读写安装目录时遭遇 Windows 特有的 *system administration* 权限问题。

## Section 2

### How to learn R



# Tips

- 先从初步操作学起，熟悉界面，掌握基础命令。

# Tips

- 先从初步操作学起，熟悉界面，掌握基础命令。
- 参考书不是特别重要，有 1-2 本足以，主要用于初步学习上手以及后续的查询。

# Tips

- 先从初步操作学起，熟悉界面，掌握基础命令。
- 参考书不是特别重要，有 1-2 本足以，主要用于初步学习上手以及后续的查询。
- “干中学”：以完成作业或研究项目为目的有针对性的学习。

# Tips

- 先从初步操作学起，熟悉界面，掌握基础命令。
- 参考书不是特别重要，有 1-2 本足以，主要用于初步学习上手以及后续的查询。
- “干中学”：以完成作业或研究项目为目的有针对性的学习。
- 善于使用 R 或者 RStudio 的帮助文件 `help()`

# Tips

- 先从初步操作学起，熟悉界面，掌握基础命令。
- 参考书不是特别重要，有 1-2 本足以，主要用于初步学习上手以及后续的查询。
- “干中学”：以完成作业或研究项目为目的有针对性的学习。
- 善于使用 R 或者 RStudio 的帮助文件 `help()`
- 善于使用搜索引擎和关键字进行搜索。

# Tips

- 先从初步操作学起，熟悉界面，掌握基础命令。
- 参考书不是特别重要，有 1-2 本足以，主要用于初步学习上手以及后续的查询。
- “干中学”：以完成作业或研究项目为目的有针对性的学习。
- 善于使用 R 或者 RStudio 的帮助文件 `help()`
- 善于使用搜索引擎和关键字进行搜索。
  - 最好是 Google, 其次是 Bing, Baidu 慎用，其他国内搜索引擎不要用

# R 的基本界面

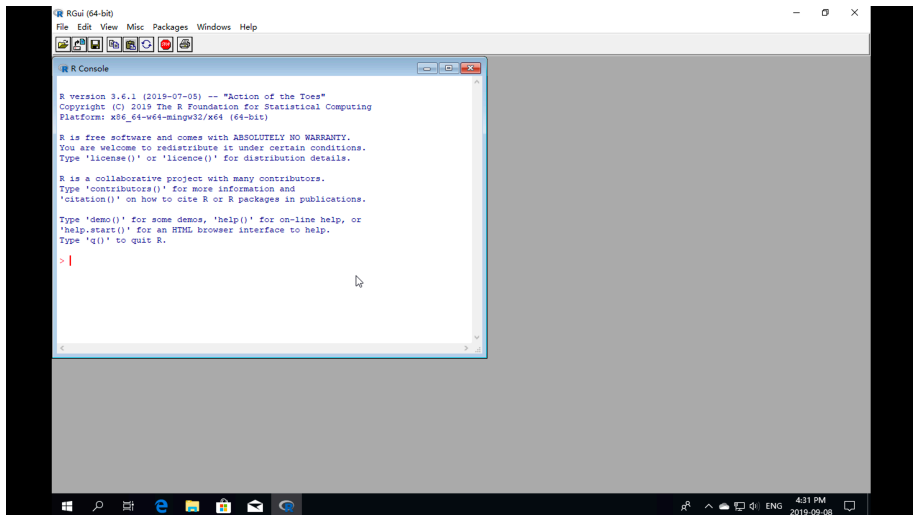


Figure 4: R Interface

# R 的简单操作

- 两种方式与 R 对话



# R 的简单操作

- 两种方式与 R 对话
  - 写命令在命令窗口 (Console)

# R 的简单操作

- 两种方式与 R 对话
  - 写命令在命令窗口 (Console)
  - 写在 R 脚本中 (R Script)

# R 的简单操作

```
1 + 2
```

```
## [1] 3
```

```
1 / (2 + 3) == .2
```

```
## [1] TRUE
```

```
1:5
```

```
## [1] 1 2 3 4 5
```

```
as.matrix(1:3)
```

```
##      [,1]
```

```
## [1,]    1
```

```
## [2,]    2
```

```
## [3,]    3
```

# R 的简单操作

- 赋值 (Assignment): 在 R 中一般使用 `<-` 作为赋值符号, 而不是 `=`。

```
v <- 2  
w <- 3  
v
```

```
## [1] 2
```

```
w/v
```

```
## [1] 1.5
```

# R 的简单操作

- 赋值向量

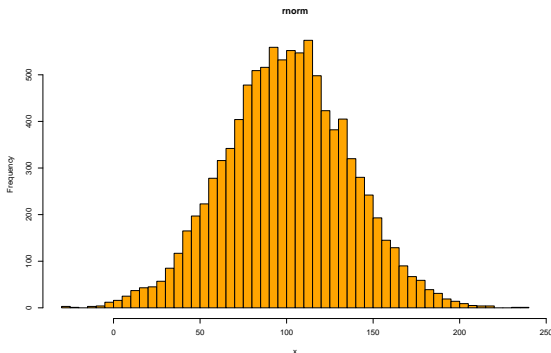
```
cnumber <- c(2,3)  
cnumber
```

```
## [1] 2 3
```

# R 的简单操作

- 赋值函数

```
x <- rnorm(10000,mean=100,sd=36)
hist(x,breaks=51,col="orange",main="rnorm")
```



# R 的简单操作

- 对象命名规则

```
var.name2 <- cnumber  
var.name2
```

```
## [1] 2 3
```

# R 的简单操作

- 对象命名规则
  - 可包含大小写字母  $a_z$ 、 $A_Z$ 、下划线  $_$  和小数点  $.$ ，不能包含其他符号。

```
var.name2 <- cnumber  
var.name2
```

```
## [1] 2 3
```



# R 的简单操作

- 对象命名规则

- 可包含大小写字母  $a_z$ 、 $A_Z$ 、下划线  $_$  和小数点  $.$ ，不能包含其他符号。
- 大小写敏感。

```
var.name2 <- cnumber  
var.name2
```

```
## [1] 2 3
```

# R 的简单操作

- 对象命名规则

- 可包含大小写字母  $a_z$ 、 $A_Z$ 、下划线  $_$  和小数点  $.$ ，不能包含其他符号。
- 大小写敏感。
- 只能以字母或点开头，不能以数字或下划线开头，当以点开头时，第一个点之后不能紧接着为数字。

```
var.name2 <- cnumber  
var.name2
```

```
## [1] 2 3
```

# R 的简单操作

- 对象命名规则

- 可包含大小写字母  $a_z$ 、 $A_Z$ 、下划线  $_$  和小数点  $.$ ，不能包含其他符号。
- 大小写敏感。
- 只能以字母或点开头，不能以数字或下划线开头，当以点开头时，第一个点之后不能紧接着为数字。
- 最好不要与系统默认的命令或函数名称相同。

```
var.name2 <- cnumber  
var.name2
```

```
## [1] 2 3
```

# R 的简单操作: 输入一组数据

**Table 1.1** The ages and weights of 10 infants

Age (mo.)	Weight (kg.)
01	4.4
03	5.3
05	7.2
02	5.2
11	8.5

Age (mo.)	Weight (kg.)
09	7.3
03	6.0
09	10.4
12	10.2
03	6.1

Note: These are fictional data.

Figure 5: R Interface

## R 的简单操作:

- 使用 Rscript 文件 (.R) 输入一组数据并求统计量

```
age <- c(1,3,5,2,11,9,3,9,12,3)
weight <- c(4.4,5.3,7.2,5.2,8.5,7.3,6.0,10.4,10.2,6.1)
mean(age)
sd(age)
cor(age,weight)
plot(age,weight)
```

## R 的简单操作:

- 由于 R 本身是开源免费软件，负责开发和维护的团队人数和资源都有限，所以 R 本身的界面和操作都显得过于原始。

## R 的简单操作:

- 由于 R 本身是开源免费软件，负责开发和维护的团队人数和资源都有限，所以 R 本身的界面和操作都显得过于原始。
- 在这种情况下，以 R 作为平台，进行二次开发的应用软件层出不穷，这些软件通常被叫做集成开发环境 (Integrated Development Environment, IDE)。它集成了代码编写功能、分析功能、编译功能、调试功能等一体化的开发软件服务套件。

## Section 3

### Using IDE: RStudio



# Intro RStudio

- The most popular IDE for R

# Intro RStudio

- The most popular IDE for R
  - 如果把 R 认为是发动机的话，IDE 可以认为是操作面板，两者共同构成数据分析的“战车”。

# Intro RStudio

- The most popular IDE for R
  - 如果把 R 认为是发动机的话，IDE 可以认为是操作面板，两者共同构成数据分析的“战车”。
  - 所以发动机可以都是 R，但可以使用不同品牌或公司的操作面板 (IDE)。

# Intro RStudio

- The most popular IDE for R
  - 如果把 R 认为是发动机的话，IDE 可以认为是操作面板，两者共同构成数据分析的“战车”。
  - 所以发动机可以都是 R，但可以使用不同品牌或公司的操作面板 (IDE)。
  - 所有操作面板中，RStudio 是最容易上手的，使用者最多的。

# Intro RStudio

- The most popular IDE for R
  - 如果把 R 认为是发动机的话，IDE 可以认为是操作面板，两者共同构成数据分析的“战车”。
  - 所以发动机可以都是 R，但可以使用不同品牌或公司的操作面板 (IDE)。
  - 所有操作面板中，RStudio 是最容易上手的，使用者最多的。
- Also Free(for basic version)

# Intro RStudio

- The most popular IDE for R
  - 如果把 R 认为是发动机的话，IDE 可以认为是操作面板，两者共同构成数据分析的“战车”。
  - 所以发动机可以都是 R，但可以使用不同品牌或公司的操作面板 (IDE)。
  - 所有操作面板中，RStudio 是最容易上手的，使用者最多的。
- Also Free(for basic version)
- Combine with Markdown and Latex to make scientific writings or presentation easier

# Intro RStudio

- The most popular IDE for R
  - 如果把 R 认为是发动机的话，IDE 可以认为是操作面板，两者共同构成数据分析的“战车”。
  - 所以发动机可以都是 R，但可以使用不同品牌或公司的操作面板 (IDE)。
  - 所有操作面板中，RStudio 是最容易上手的，使用者最多的。
- Also Free(for basic version)
- Combine with Markdown and Latex to make scientific writings or presentation easier
- Download it from here: [RStudio](#)

# Installing RStudio

- 注意事项 (for Windows)



# Installing RStudio

- 注意事项 (for Windows)
  - 官网下载

# Installing RStudio

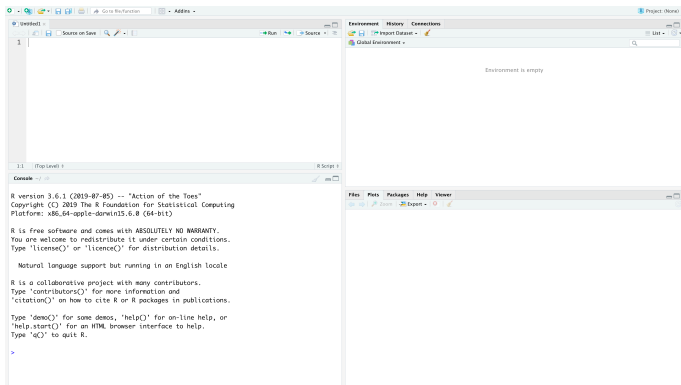
- **注意事项 (for Windows)**
  - **官网下载**
  - **安装路径里不要有中文**

# Installing RStudio

- 注意事项 (for Windows)
  - 官网下载
  - 安装路径里不要有中文
  - 目录最好不是 `program files`, 不然后面可能会由于“管理员权限”影响使用。

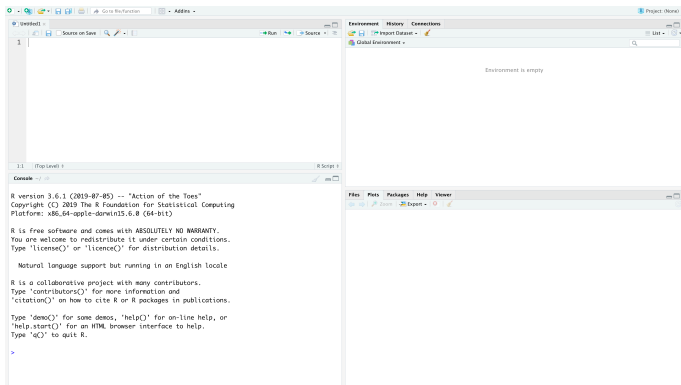
# RStudio 的基本界面

- 三种方式与 R 对话



# RStudio 的基本界面

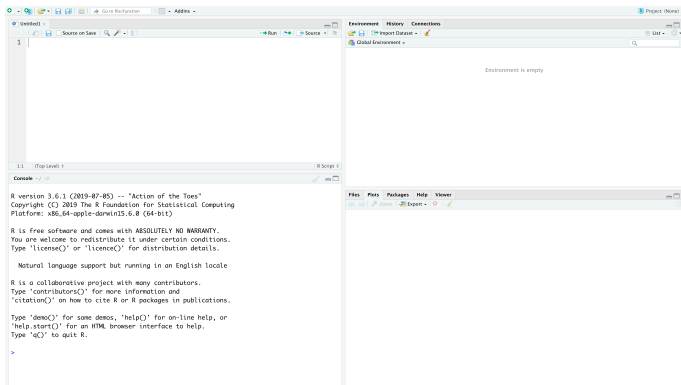
- 三种方式与 R 对话
  - 写命令在命令窗口 (Console): 简单命令



# RStudio 的基本界面

- 三种方式与 R 对话

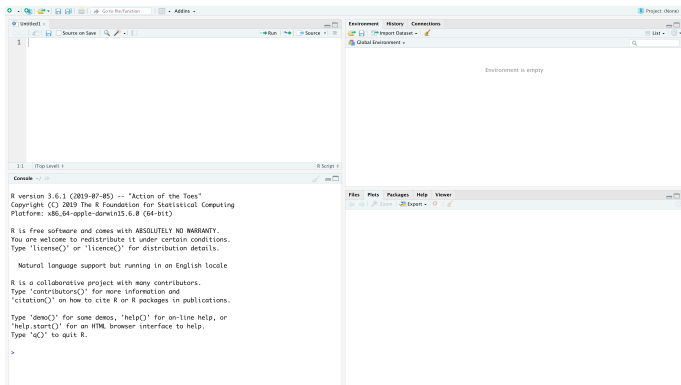
- 写命令在命令窗口 (Console): 简单命令
- 写在 R 脚本中 (R Script): 复杂的一系列的命令或函数等



# RStudio 的基本界面

## ● 三种方式与 R 对话

- 写命令在命令窗口 (Console): 简单命令
- 写在 R 脚本中 (R Script): 复杂的一系列的命令或函数等
- 写在 Rmarkdown 文档中, 生成分析代码 + 结果 + 文字的报告。



# RStudio 的基本界面

- Console



# RStudio 的基本界面

- Console
- R Script or Rmarkdown

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data
  - History

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data
  - History
  - Connection: data sources

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data
  - History
  - Connection: data sources
  - Git

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data
  - History
  - Connection: data sources
  - Git
- Others

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data
  - History
  - Connection: data sources
  - Git
- Others
  - Files



# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data
  - History
  - Connection: data sources
  - Git
- Others
  - Files
  - Plots

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data
  - History
  - Connection: data sources
  - Git
- Others
  - Files
  - Plots
  - Packages

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data
  - History
  - Connection: data sources
  - Git
- Others
  - Files
  - Plots
  - Packages
  - Help

# RStudio 的基本界面

- Console
- R Script or Rmarkdown
- Environment
  - Import Data
  - History
  - Connection: data sources
  - Git
- Others
  - Files
  - Plots
  - Packages
  - Help
  - Viewer

# Packages(包)

- 包是基于 R 的基本功能，用来完成某些高级功能的专属模块。可以通过全世界的镜像地址，在线下载和安装。

```
.packages(TRUE)
```

# Packages(包)

- Packages(包) 基本分成三类:

# Packages(包)

- Packages(包) 基本分成三类：
  - 基础包

# Packages(包)

- Packages(包) 基本分成三类：
  - 基础包
  - 附加包



# Packages(包)

- Packages(包) 基本分成三类：
  - 基础包
  - 附加包
  - 个人包

# Packages(包)

- Packages(包) 基本分成三类：
  - 基础包
  - 附加包
  - 个人包
- 因为开源，提交包的门槛相对较低，所以导致包的质量良莠不齐。因此在使用一个包之前，最好对这个包的使用情况有一定了解。

# Packages(包)

- 基本命令方式

# Packages(包)

- 基本命令方式
  - 安装：用 `install.packages("包的名字")`

# Packages(包)

- 基本命令方式
  - 安装：用 `install.packages("包的名字")`
  - 安装之后，要想使用还必须加载：用 `library(包的名字)`

# Packages(包)

- 基本命令方式

- 安装：用 `install.packages("包的名字")`
- 安装之后，要想使用还必须加载：用 `library(包的名字)`
- 有时还要卸载：`detach("package: 包的名字", unload=TRUE)`

# Packages(包)

- 基本命令方式

- 安装：用 `install.packages("包的名字")`
- 安装之后，要想使用还必须加载：用 `library(包的名字)`
- 有时还要卸载：`detach("package: 包的名字", unload=TRUE)`
- 彻底移除：`remove.packages("包的名字")`

# Packages(包)

- 基本命令方式
  - 安装：用 `install.packages("包的名字")`
  - 安装之后，要想使用还必须加载：用 `library(包的名字)`
  - 有时还要卸载：`detach("package: 包的名字", unload=TRUE)`
  - 彻底移除：`remove.packages("包的名字")`
- 在 RStudio 里，也可以采用窗口菜单点选的方式。



# Packages(包)

- 基本命令方式

- 安装：用 `install.packages("包的名字")`
- 安装之后，要想使用还必须加载：用 `library(包的名字)`
- 有时还要卸载：`detach("package: 包的名字", unload=TRUE)`
- 彻底移除：`remove.packages("包的名字")`

- 在 RStudio 里，也可以采用窗口菜单点选的方式。

- 安装计量经济学常用包 AER

```
install.packages("AER",  
  repos = "https://mirrors.nju.edu.cn/CRAN/")
```

# Packages(包)

- 选择合适的镜像

# Packages(包)

- 选择合适的镜像
- **Tools**→**Global Options**→**Packages**→

# Packages(包)

- 选择合适的镜像
- **Tools**→**Global Options**→**Packages**→
- 在 Primary CRAN repository 中选择国内的相关镜像。

# Packages(包)

- 查看 Packages 的内容

```
library(AER)
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
## Loading required package: lmtest
```

```
## Loading required package: zoo
```

```
##
```

```
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      as.Date, as.Date.numeric
```

# Packages(包)

- 查看 Packages 的内容
- 加载 Packages

```
library(AER)
```

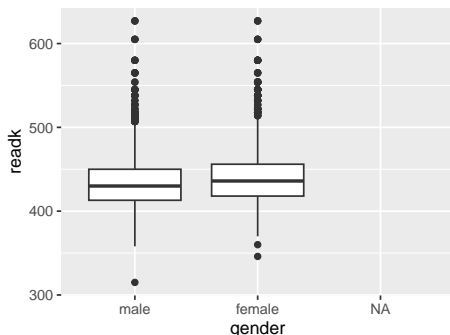
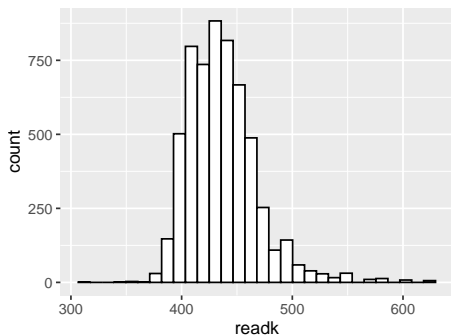
```
## Loading required package: car
## Loading required package: carData
## Loading required package: lmtest
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##      as.Date, as.Date.numeric
```

# Packages(包)

```
library(ggplot2)
p1<-ggplot(STAR, aes(readk)) +
  geom_histogram(bins=30, colour="black", fill="white")
p2<-ggplot(STAR, aes(x=gender, y=readk)) +
  geom_boxplot()
```

# Packages(包)

```
library(gridExtra)  
grid.arrange(p1,p2,ncol = 2, nrow = 1)
```





## Section 4

# Directory Management

# Workspace

- The workspace is your current R **working environment** and includes any user-defined objects (vectors, matrices, functions, data frames, and lists).

```
save.image("myfile")  
load("myfile")
```

# Workspace

- The workspace is your current R **working environment** and includes any user-defined objects (vectors, matrices, functions, data frames, and lists).
  - When you quit R, you can save an **image** of the current workspace that's automatically reloaded the next time R starts.

```
save.image("myfile")  
load("myfile")
```

## Working directory

- The current **working directory** is the directory from which R will read files and to which it will save results by default.

```
getwd() # 显示当前目录  
setwd("C:/Users/lifan/Desktop/RLab_1_5_2022") # 设定工作目录
```

# Building your own system of directory

- Project name directory

# Building your own system of directory

- Project name directory
  - RawData

# Building your own system of directory

- Project name directory
  - RawData
  - WorkData

# Building your own system of directory

- Project name directory
  - RawData
  - WorkData
  - Figures



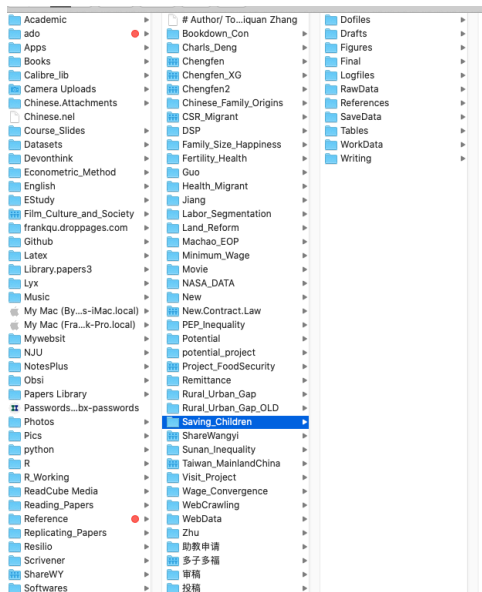
# Building your own system of directory

- Project name directory
  - RawData
  - WorkData
  - Figures
  - Tables

# Building your own system of directory

- Project name directory
  - RawData
  - WorkData
  - Figures
  - Tables
  - .....

## My imperfect example:



# Where are packages installed

- ① R 的安装目录可写吗？如果有写的权限，那么就把包装到 R 安装目录下，比如 `C:/Software/R/library/`

# Where are packages installed

- ① R 的安装目录可写吗？如果有写的权限，那么就把包装到 R 安装目录下，比如 `C:/Software/R/library/`
- ② 如果安装不可写，那么 R 会要求新建一个文件夹来安装包。

# User's directory

```
normalizePath('~')
```

```
## [1] "D:\\Documents"
```

```
list.files('~', all.files = TRUE) # how many files in the HOME
```

```
## [1] "."                ".."                ".Renviron"
## [4] "Adobe"            "Blackmagic Design" "ClassIn Files"
## [7] "Downloads"        "DyingLight"        "Klei"
## [10] "lab1.log"          "lab1.pdf"           "lab1.Rmd"
## [13] "lab1.tex"          "League of Legends" "MATLAB"
## [16] "My eBooks"         "My Games"           "myfile"
## [19] "OneNote 笔记本"   "others"              "prscrn Files"
## [22] "Python Scripts"   "Sound recordings"    "Tencent Files"
## [25] "Tencent OD Files" "TencentMeeting"     "Warm Lamp Gam
## [28] "WeChat Files"     "Zoom"
```

## .Renviron and .Rprofile

```
file.edit('~/.Renviron') # open the file and edit
```

- Restart **R**(Cmd+Shift+F10), then try to install a package

## .Renviron and .Rprofile

```
file.edit('~/.Renviron') # open the file and edit
```

- Restart **R**(Cmd+Shift+F10), then try to install a package
- 自定义包的安装路径的好处：当重装系统、更换电脑等情况下，不需要重新 reinstall 这些包。



## .Renviron and .Rprofile

- 而.Rprofile 文件则是一个 R 代码文件，在 R 启动时，如果这个文件存在，它会被首先执行。

```
file.edit('~/.Rprofile') # open the file and edit
```

# Windows 下的 Rconsole 文件

```
file.path(R.home('etc'),'Rconsole') # find the path
```

- 修改语言为英语

```
language = en # find the path
```

# MacOS 下的系统环境

```
Sys.setenv(LANGUAGE = 'en')  
Sys.unsetenv()
```

# Reference

- Jared P. Lander (2013), *R for Everyone: Advanced Analytics and Graphics*

# Reference

- Jared P. Lander (2013), *R for Everyone: Advanced Analytics and Graphics*
- Robert I. Kabacoff(2011), *R in Action: Data Analysis and Graphics With R*

# Reference

- Jared P. Lander (2013), *R for Everyone: Advanced Analytics and Graphics*
- Robert I. Kabacoff(2011), *R in Action: Data Analysis and Graphics With R*
- 谢益辉等 (2018), *R 语言忍者秘籍*