实用统计软件课程介绍

温灿红

中国科学技术大学管理学院

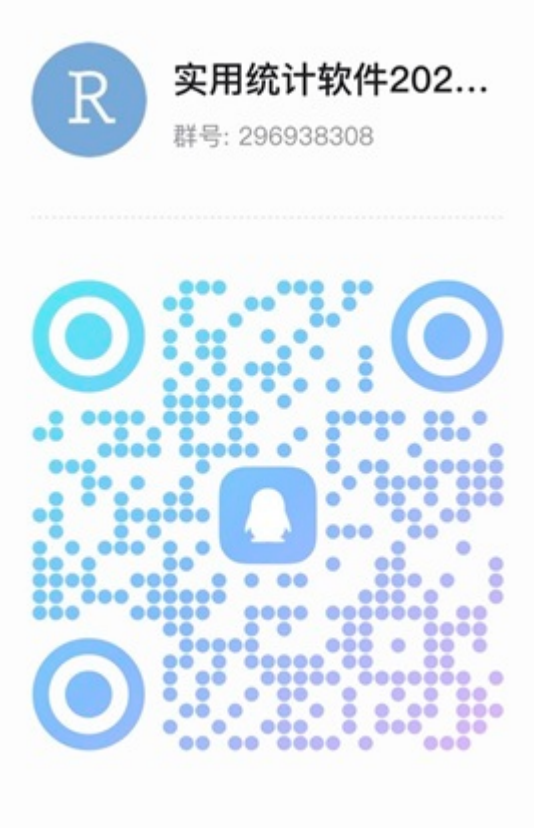
课程概括

课程信息

- 教师信息：温灿红（wench@ustc.edu.cn），办公室：管理科研楼1001
- 助教信息：井怡、罗梓丹
- 上课地点：5501
- 上课时间：周二(3,4,5)
- 实验平台：教学云平台(https://yun.ustc.edu.cn/#/home)

课程信息

- 课程QQ群：



课程描述

- 本课程围绕着 **R** 和 **Python** 这两个常用的统计软件展开：
  - 关于**R**的部分：主要讲述了基本语法和数据结构，各种类型数据的导入、整理和基本操作，画图实现，编程和搭建**R**包等.
  - 关于**Python**的部分：主要介绍了基本语法和numPy、matplotlib、pandas模块。
- 比例构成：R占比3/4，Python占比1/4。

课程目标

- 理解 **R** 和 **Python** 中常用的数据结构和数据类型，并清楚其优缺点和适用范围。
- 基于 **Rmarkdown** 或 **Jupyter** 撰写具有可重复性实验结果的研究报告。
- 掌握如何用 **R** 或者 **Python** 处理数据，包括但不限于导入数据、数据预处理、画图等。
- 打包**R**包或者搭建**Python**模块。

教材

- 教材

1. 《R语言核心技术手册（第二版）》 Joseph Adler著，刘思喆, 李舰, 陈钢，邓一硕译(2014)：R基本语法和编程

2. 《Python数据科学实践》常象宇，曾智亿，李春艳，程茜 著(2020)

参考书

- 参考书

1. 北京大学李东风老师的《R语言教程》
2. 《ggplot2：数据分析与图形艺术》 Hadley Wickham著，黄俊文译 (2016)：画图中ggplot部分
3. 《Rcpp:R和C++的无缝整合》 Dirk Eddelbuettel著，寇强，张哗译 (2015)：Rcpp部分
4. 《R包开发》Hadley Wickham (2016)： R包开发部分
5. 《Python编程：从入门到实践（第二版）》Eric Matthes著，袁国忠译(2020)：Python基本语法
6. 《利用Python进行数据分析（第二版）》 Wes McKinney著，徐敬一译(2018)：numPy、matplotlib、pandas模块

其他参考资料

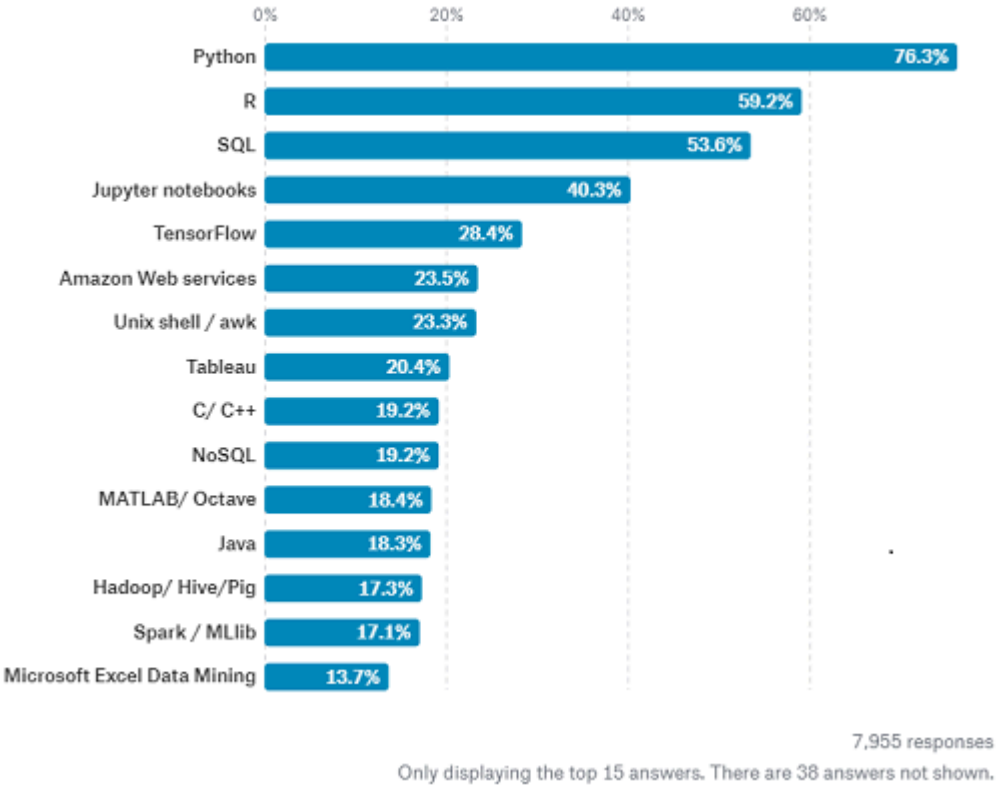
1. **R** 官方指南，初学者手册"An Introduction to R", PDF版本可见以下链接：https://cran.r-project.org/
2. Quick-R
3. **Rmarkdown** 参考资料，备忘单
4. A byte of Python (英文版), Python简明教程（中文版）

成绩评定办法

- 30% 平时成绩
  - 8次左右的作业（包括4次作业和4次实验作业），必须使用**Rmarkdown** 或**Jupyter**笔记本完成作业。
  - 每次作业需要提交两个文档，包括源代码文档和输出成HTML的文档，命名格式如下：姓名_Hw1.rmd 或 姓名_Hw1.md，以及姓名_Hw1.html。
  - 迟交一天(24小时内）打折20%，迟交两天(48小时内）打折50%，不接受晚交2天的作业和项目（任何时候理由都不接受）。
- 20% 项目（数据的初步分析）
  - 分组：自由组合，不超过2人一组，可以一个人。
  - 如果是多人组队，请明确说明每人负责的部分和内容。
- 50% 期末成绩
  - 半开卷，只能带一张A4纸，考试期间不能与人交谈。

初步认识 R 和 Python

为什么学习 R 和 Python



基于2020年在Kaggle上的用户使用的语言工具调查。

为什么学习 R 和 Python

通过分析用户的职业发现

- **R**：数据分析师、商业分析师、运筹研究员、预测模型工程师、统计学家等
- **Python**：计算机科学家、数据科学家、工程师、机器学习工程师、程序员、软件开发工程师等

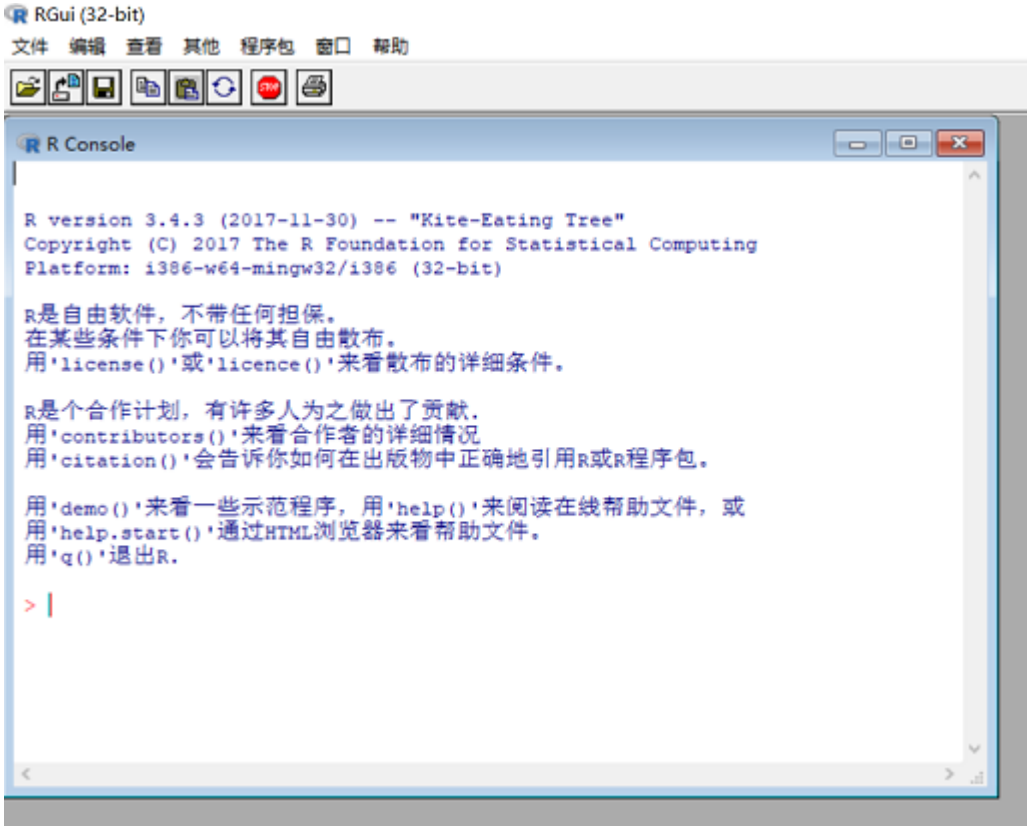
R 语言介绍

什么是R

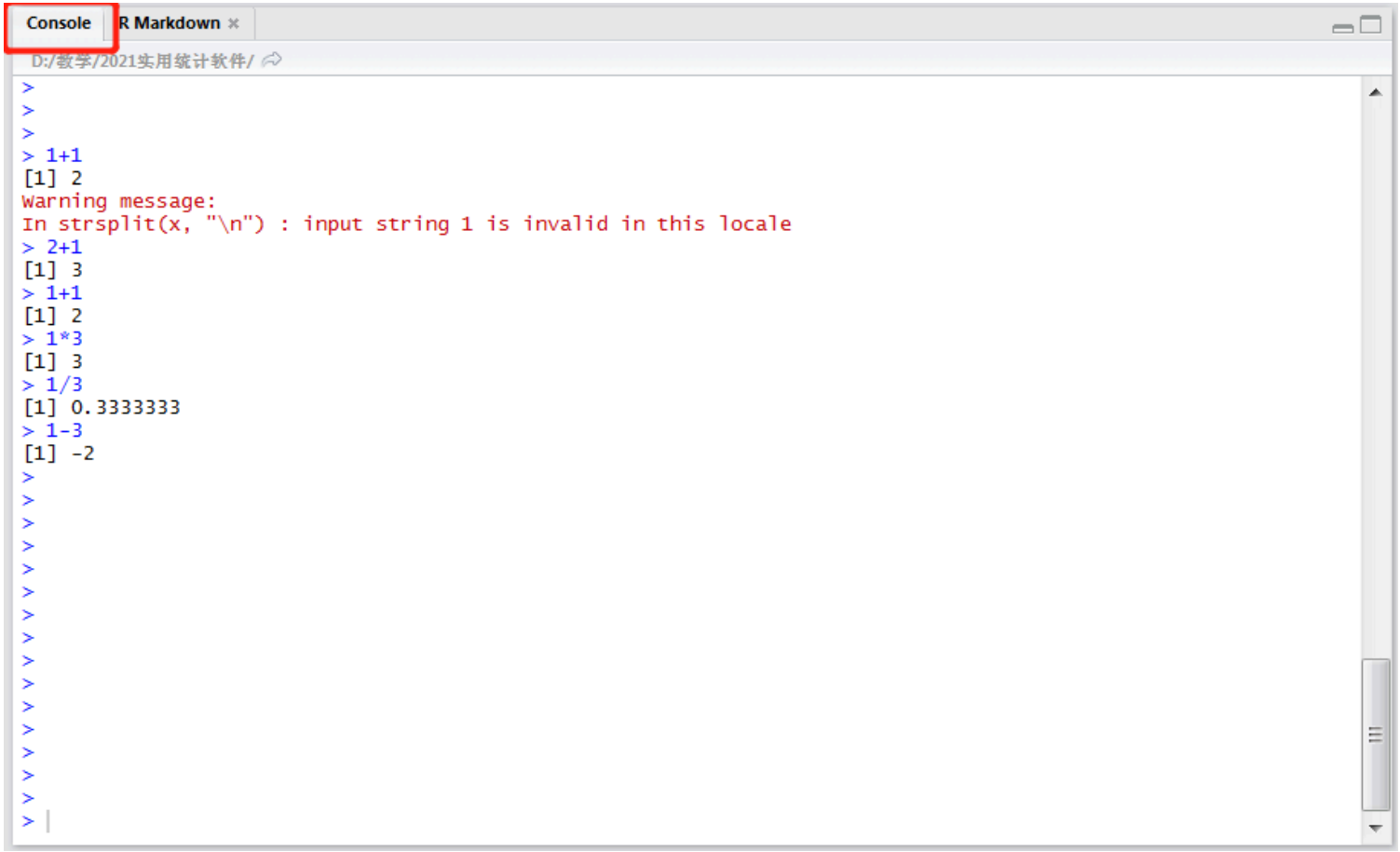
- 自由软件，免费、开放源代码，支持各个主要计算机系统；
- 完整的程序设计语言，基于函数和对象，可以自定义函数，调入C、C++、Fortran编译的代码；
- 具有完善的数据类型，如向量、矩阵、因子、数据集、一般对象等，支持缺失值，代码像伪代码一样简洁、可读；
- 强调交互式数据分析，支持复杂算法描述，图形功能强；
- 实现了经典的、现代的统计方法，如参数和非参数假设检验、线性回归、广义线性回归、非线性回归、可加模型、树回归、混合模型、方差分析、判别、聚类、时间序列分析等。
- 统计科研工作者广泛使用R进行计算和发表算法。R有上万软件包(截止2021年8月有一万七千多个)。

R的下载与安装

- 下载链接：https://cran.r-project.org/
- 下载后按照提示安装即可



命令行界面


Console
R Markdown
D:/教学/2021实用统计软件/
> 1+1
[1] 2
Warning message:
In strsplit(x, "\n") : input string 1 is invalid in this locale
> 2+1
[1] 3
> 1+1
[1] 2
> 1*3
[1] 3
> 1/3
[1] 0.3333333
> 1-3
[1] -2

运行

- 我们在">"后输入命令代码，输入完成后按"Enter"键即可运行，R会自动输出相应的结果。

```
1+2
```

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

- 如果输入太多，可断行输入(但要保证代码不是完整的)，如：

```
    1+2+3+
    4+5
```

```
## [1] 15
```

- 可在一行内输入多行命令，如：

```
1+2; 3+4
```

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

```
## [1] 7
```

赋值（一）

- 如果要对数据进行多种分析，一般我们会给数据起一个名称，并把数据复制给这个名称，这就是所有的R对象(object)。
- 标准的赋值符号是 <- ，也允许 = 和 -> 来赋值，但推荐用 <- 。

```
x <- c(70, 85, 92, 71, 55)  # 将5个数据赋值给对象x
y <- x                      # 将x赋值给对象y
z = 1
x -> z
z
```

```
## [1] 70 85 92 71 55
```

赋值（二）

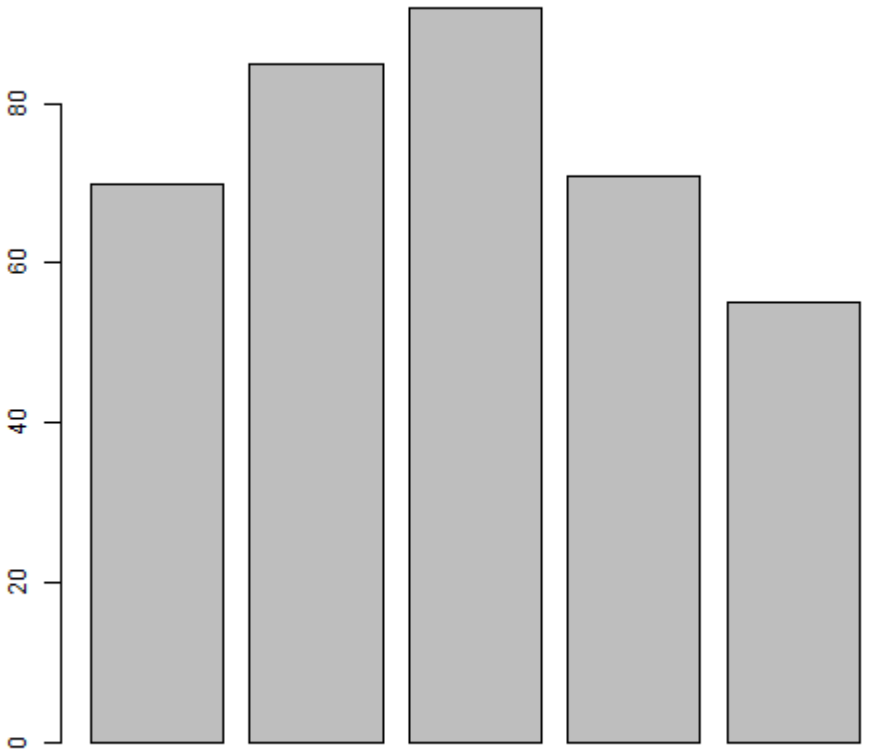
- 赋值后可对对象做进行分析、操作。如：

```
    sum(x)                  # 计算对象x的总和
```

```
## [1] 373
```

赋值（三）

```
barplot(x)                    # 绘制对象x的条形图
```



输出至命令行界面

- 用print()函数显示某个函数或表达式的结果，如

```
print("Hello World!")
```

```
## [1] "Hello World!"
```

- 用cat()函数显示多项内容，包括数值和文本，文本包在两个单撇号或两个双撇号中，如

```
cat("8的平方是", 8^2,  "\n") # '\n'是换行符
```

```
## 8的平方是 64
```

```
cat("1 + 1", 1+1, sep="=")
```

```
## 1 + 1=2
```

输出至文件

- cat() 默认显示在命令行窗口， 为了写入指定文件中， 在 cat() 调用中用 file = 选项。

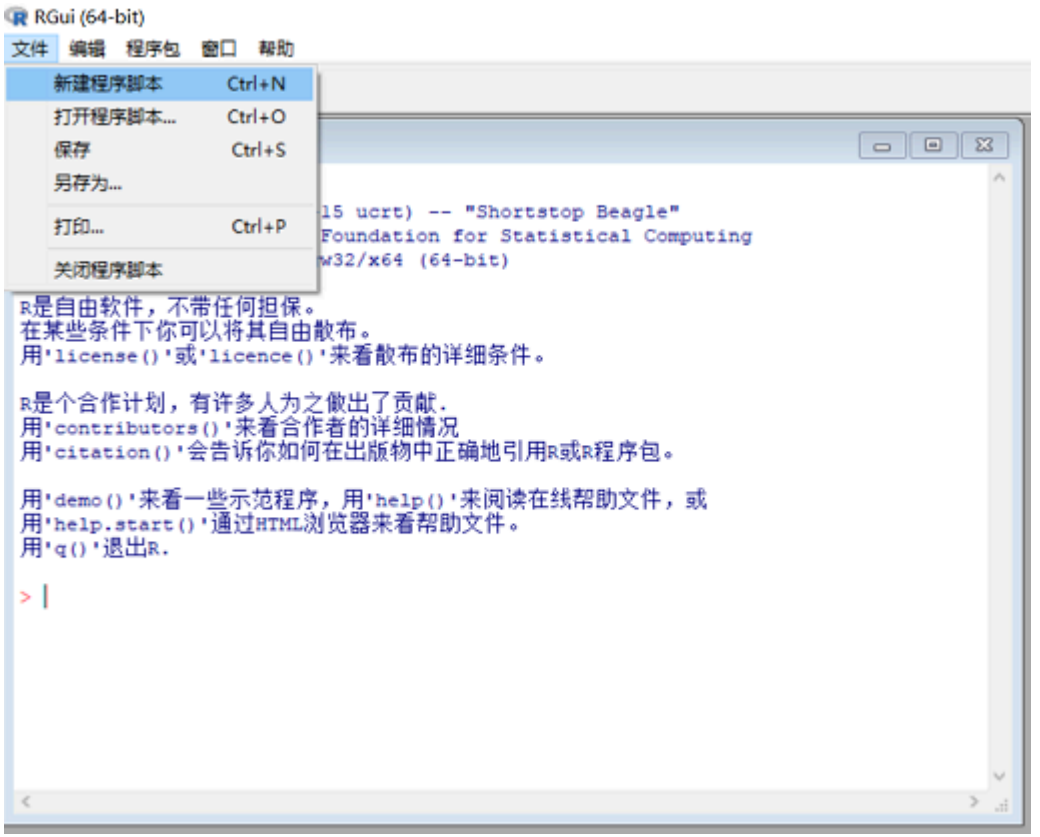
```
cat("=== test file ===\n", file="test.txt")
cat("1 + 1 =", 1 + 1, "\n", file="test.txt", append=TRUE) # 不覆盖原文件
```

- sink() 可以用来把命令行窗口显示的运行结果转向保存到指定的文本文件中，如果希望保存到文件的同时也在命令行窗口显示， 使用 split = TRUE 选项。

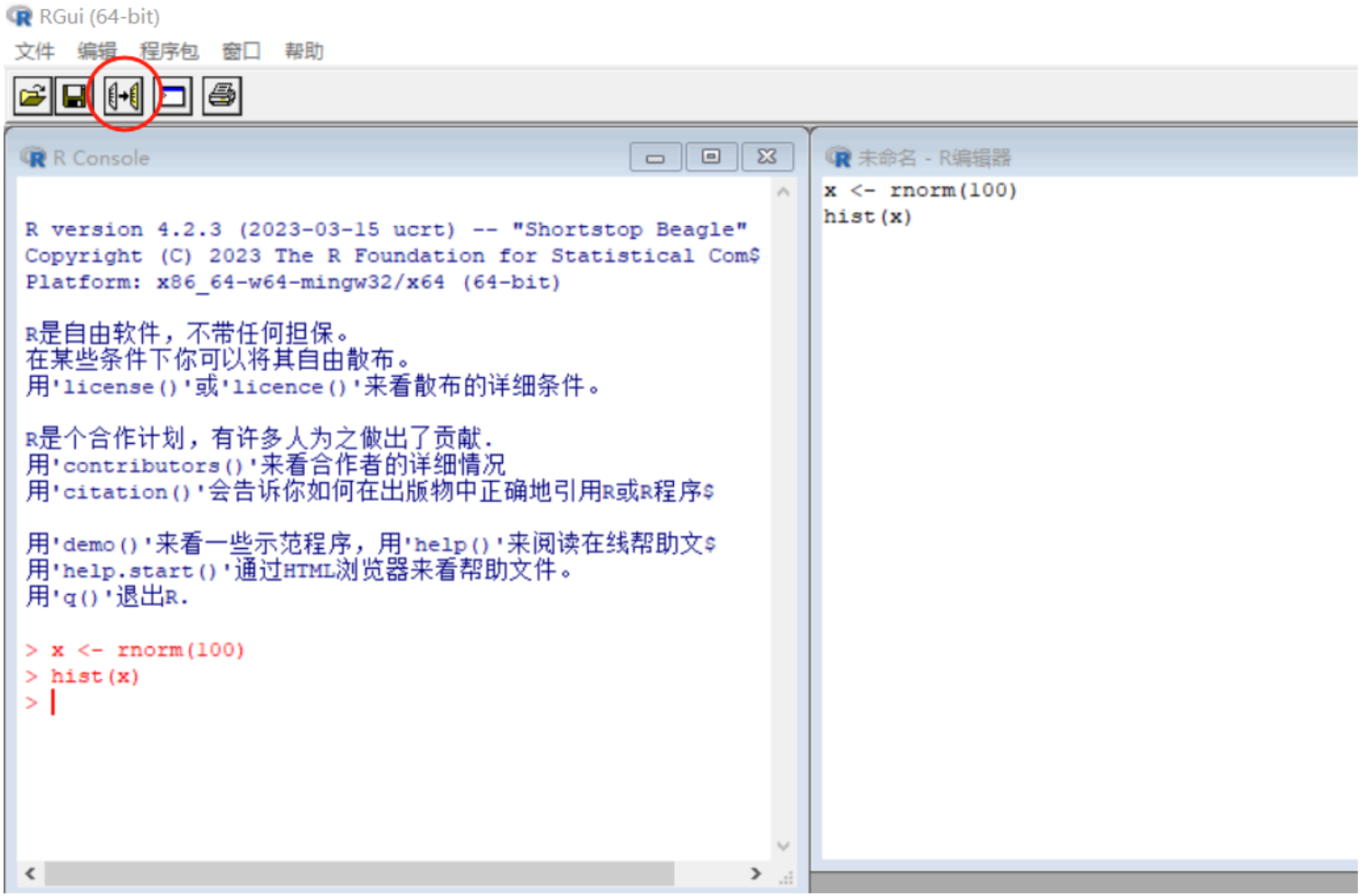
```
sink("test2.txt", split=TRUE)    # 开始输出
sink()                           # 停止输出
```

编写代码脚本

- 提示符后输入容易出错，还不容易修改，也不利于保存代码。
- 在脚本文件中编写，脚本文件以".R"后缀保存。



编写代码脚本


RGui (64-bit)
文件 编辑 程序包 窗口 帮助
R Console
R version 4.2.3 (2023-03-15 ucrt) -- "Shortstop Beagle"
Copyright (C) 2023 The R Foundation for Statistical Com$
Platform: x86_64-w64-mingw32/x64 (64-bit)
R是自由软件，不带任何担保。
在某些条件下你可以将其自由散布。
用'license()'或'licence()'来看散布的详细条件。
R是个合作计划，有许多人为之做出了贡献.
用'contributors()'来看合作者的详细情况
用'citation()'会告诉你如何在出版物中正确地引用R或R程序$
用'demo()'来看一些示范程序，用'help()'来阅读在线帮助文$
用'help.start()'通过HTML浏览器来看帮助文件。
用'q()'退出R.
> x <- rnorm(100)
> hist(x)
>
未命名 - R编辑器
x <- rnorm(100)
hist(x)

运行脚本文件

用source()函数可以运行保存在一个.R文件中的代码脚本。 比如，如下内容保存在文件example.R里：

```
sqrt(pi)
```

```
## [1] 1.772454
```

```
source("example.R")
source("example.R", echo=TRUE)
```

```
##
## > sqrt(pi)
## [1] 1.772454
```

```
source("example.R", print.eval=TRUE)
```

```
## [1] 1.772454
```

工作目录

- 我们一般会把代码脚本文件和数据文件放在同一个目录下。
- 有时候需要调用别的程序，这时候可以通过改变工作目录来实现。
    - getwd: 获取当前工作目录
    - setwd: 改变当前工作目录

```
getwd()
```

```
## [1] "E:/2024春 实用统计软件/R部分"
```

工作空间

- 在命令行中定义的对象， 会保存在**R**的工作空间，可通过`ls()`查看当前空间的对象

```
ls()
```

```
## [1] "x" "y" "z"
```

- 如果想要把某个变量剔除掉，运行`rm()`

```
rm("r")
```

```
## Warning in rm("r"): 找不到对象'r'
```

- 在退出R时，会提问是否保存工作空间，如果选择保存，则再次启动R时能够看到以前定义的各个对象的值。
- 也可通过`rm(list = ls())`清空当前所有的变量。

```
rm(list = ls())
ls()
```

```
## character(0)
```

R扩展软件包

- **R**官网上发布有两万多个扩展软件包，提供各种各样的功能和数据，还有更多的在Github等平台上。
- 一个完整的**R**包包括代码（不仅仅是**R**代码，还有C，C++等源代码），文档说明，数据集等。
- 查看默认安装的**R**包

```
getOption("defaultPackages")
```

```
## [1] "datasets"  "utils"     "grDevices" "graphics"  "stats"     "methods"
```

- 目前加载在环境中的**R**包

```
(.packages())
```

```
## [1] "stats"     "graphics"  "grDevices" "utils"     "datasets"  "methods"
## [7] "base"
```

R扩展软件包的安装

- 从CRAN官网安装**R**包：

```
install.packages("BeSS")
```

- 从其他途径安装装**R**包
  - Github：

```
if(!require(devtools)) install.packages('devtools')
devtools::install_github("hailongba/csrrr") #
```

  - Bioconductor:

```
devtools::install_bioc("snpStats")
```

R扩展软件包的加载

```
library(BeSS)
data("SAheart")
```

```
require(BeSS)
data("SAheart")
```

$\Rightarrow$ 那么library和require的区别是什么?

帮助

- 如果你知道函数名字，那么直接在**R**中寻求帮助

```
help("mean")
```

```
## 打开httpd帮助服务器… 好了
```

```
?mean
example(mean)
```

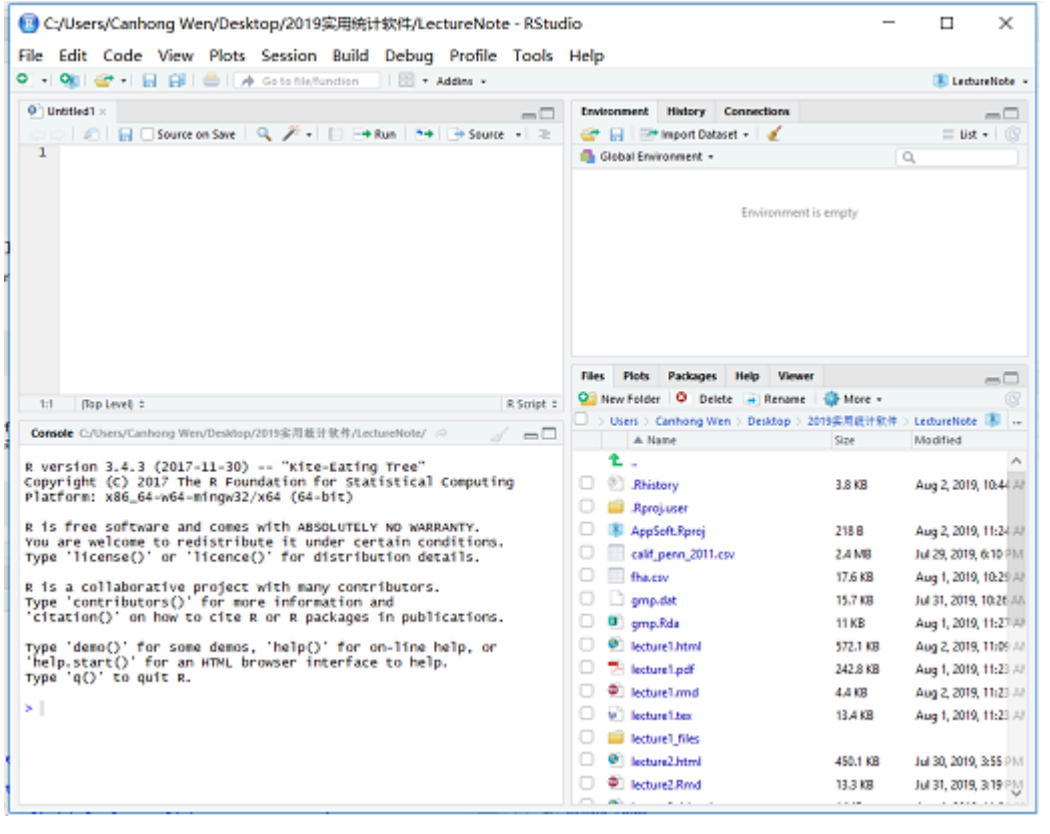
```
##
## mean> x <- c(0:10, 50)
##
## mean> xm <- mean(x)
##
## mean> c(xm, mean(x, trim = 0.10))
## [1] 8.75 5.50
```

帮助

- 如果你不知道函数名字，只知道要实现什么功能，那么毫无疑问选Google
  - 在你描述的问题后面加上"in R"
  - 在你写的程序出错了，但是不知道怎么解决的时候特别管用（如果你显示的不是英文，那么要在R中运行`Sys.setenv(LANGUAGE = "en")`）
- 如果Google都找不到，那么试试看 stackoverflow

什么是RStudio

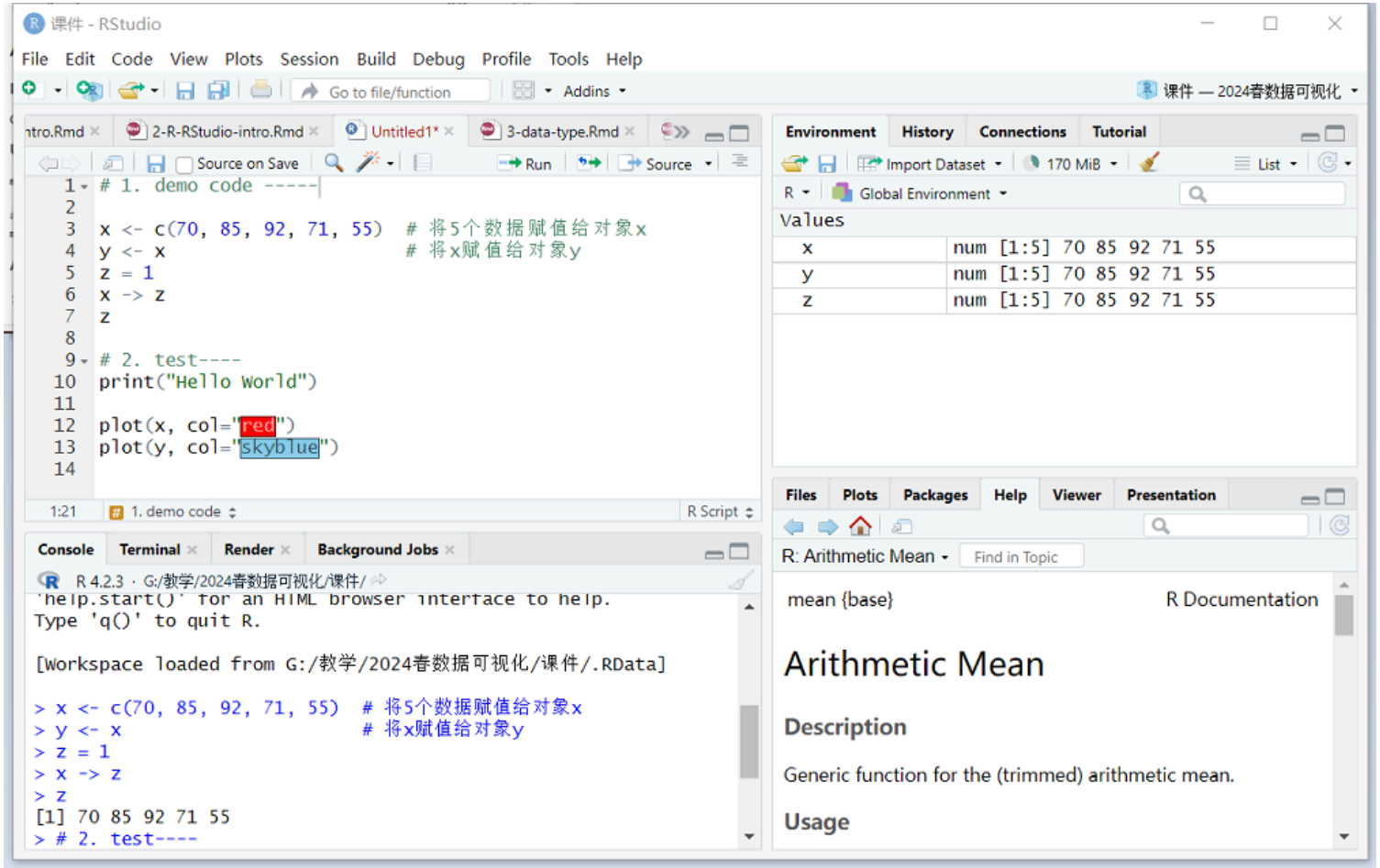
- **RStudio**是一个非常好用的关于**R**语言的集成开发环境。
- 一个强大的**R**语言编程和调试的编辑器，同时是一个强大的HTML、PDF、动态文档和幻灯片演示生成器。



RStudio的下载与安装

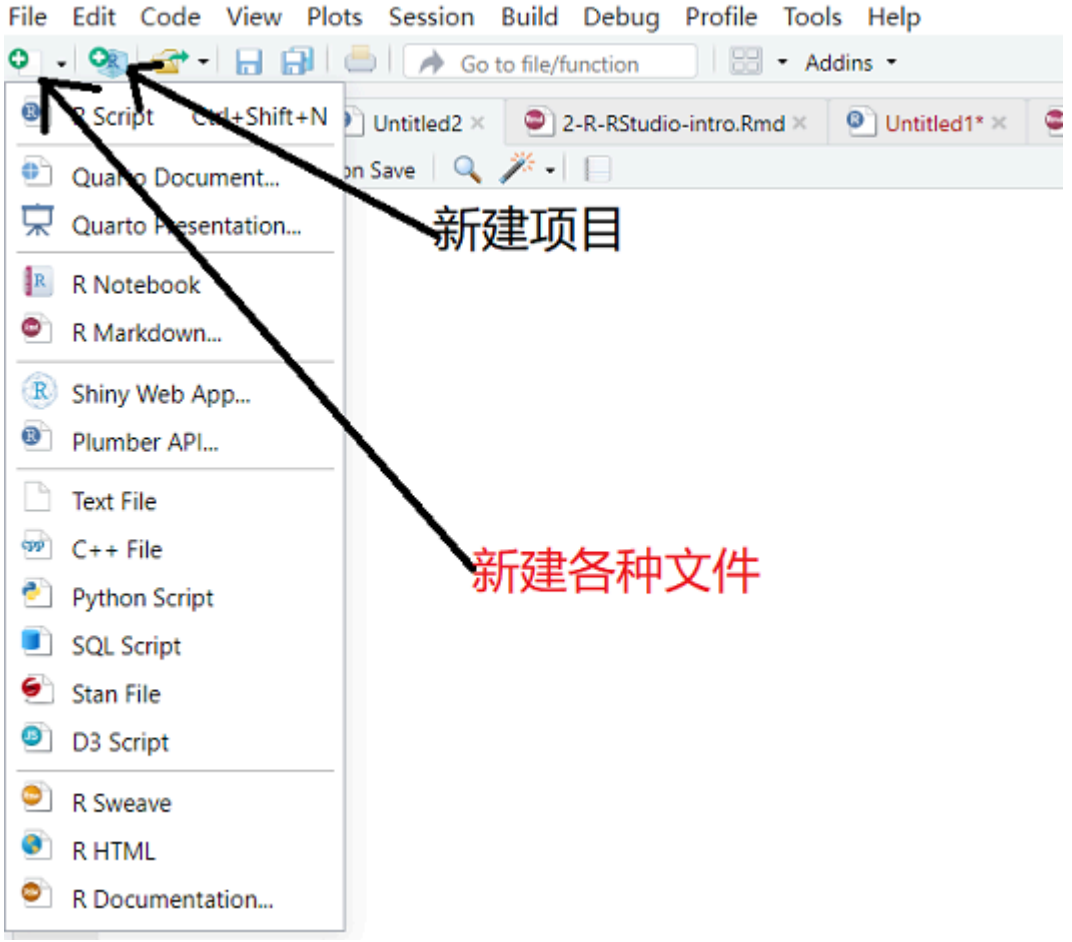
- 可在终端机或者云端运行。
- 下载地址：http://www.rstudio.com/download
- 根据提示安装即可

RStudio界面介绍



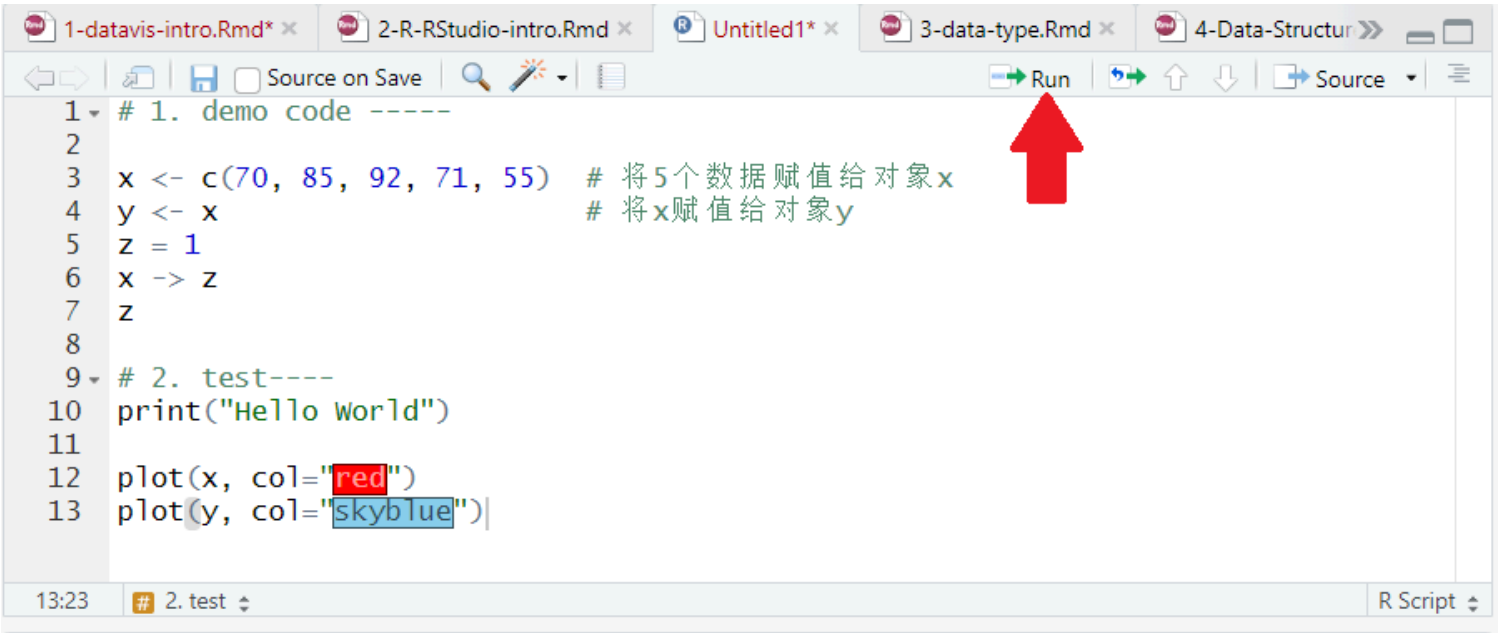
RStudio界面介绍

- **工具栏**：包含常见操作的快捷方式，如新建脚本、保存脚本、运行代码和管理项目。通过提供对基本功能的快速访问，提高工作流效率。



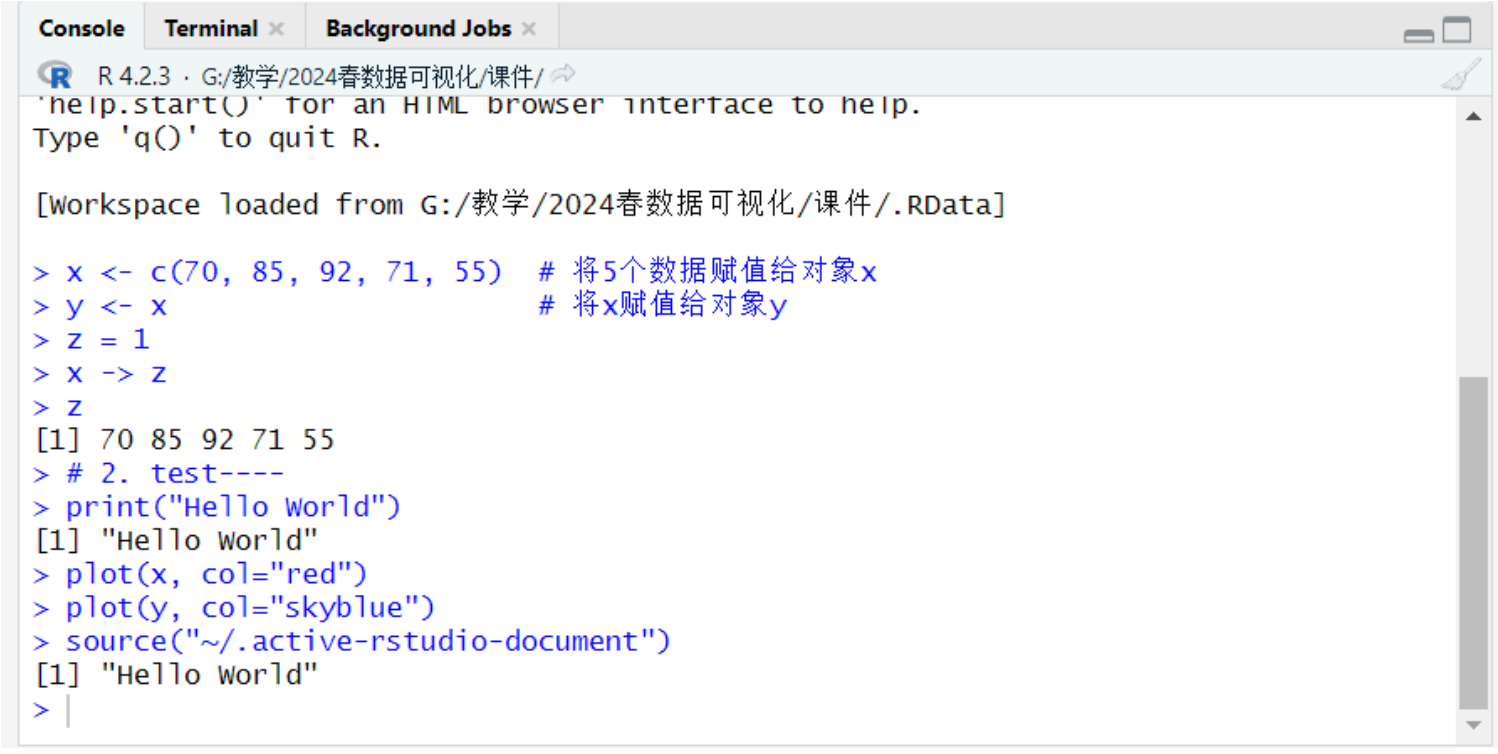
RStudio界面介绍

- **编辑器窗格**：用于编写和编辑R脚本和代码的主要区域。包括.R, .Rmd等文档。具有语法高亮显示、代码补全和错误检查等功能。



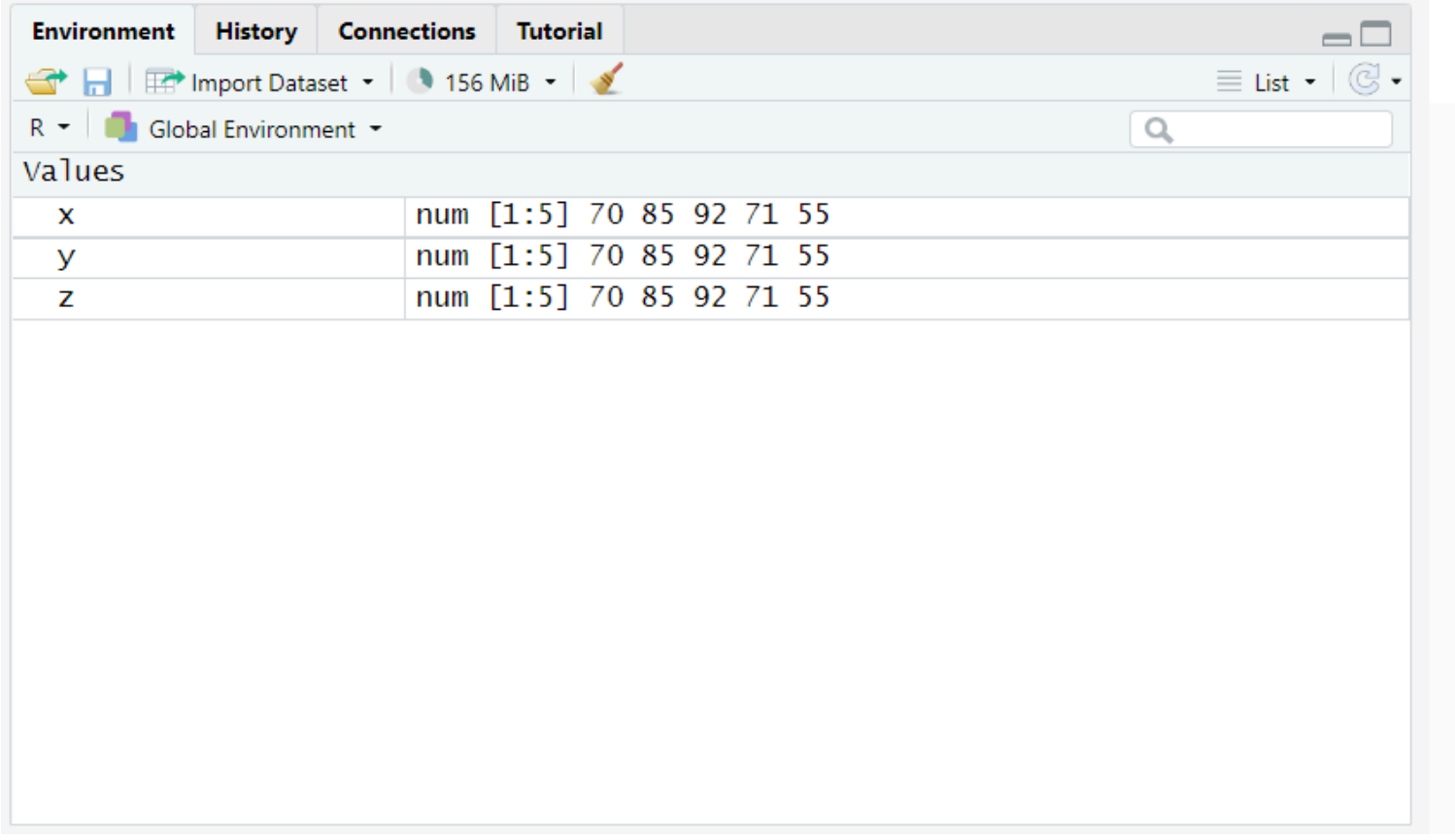
RStudio界面介绍

- **控制台窗格**：交互式控制台，用于执行R代码并显示结果。可用于测试代码片段和交互式运行命令。



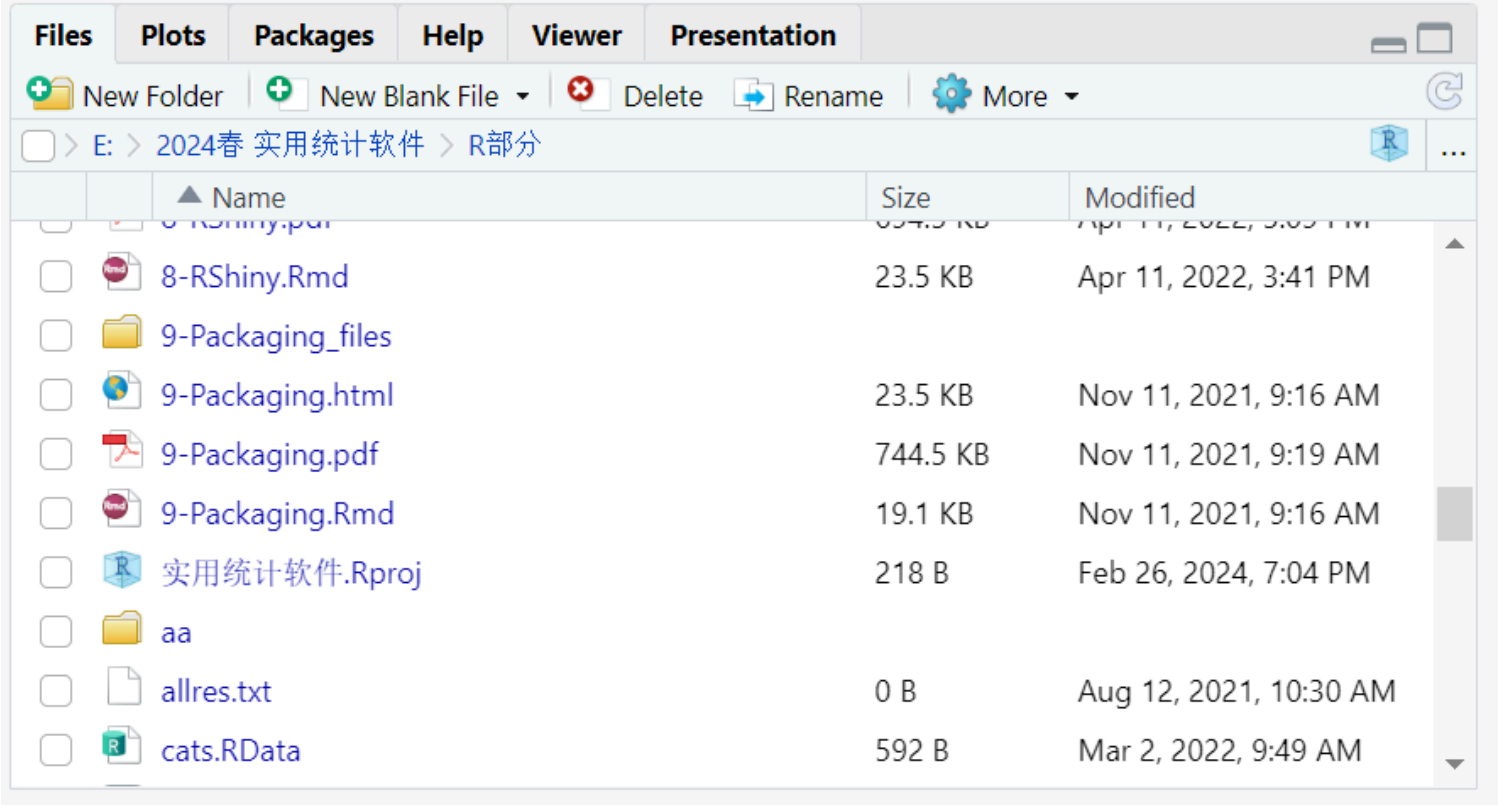
RStudio界面介绍

- **环境窗格**：显示当前R会话中的对象、变量和数据框等信息。可用于管理和浏览工作空间。



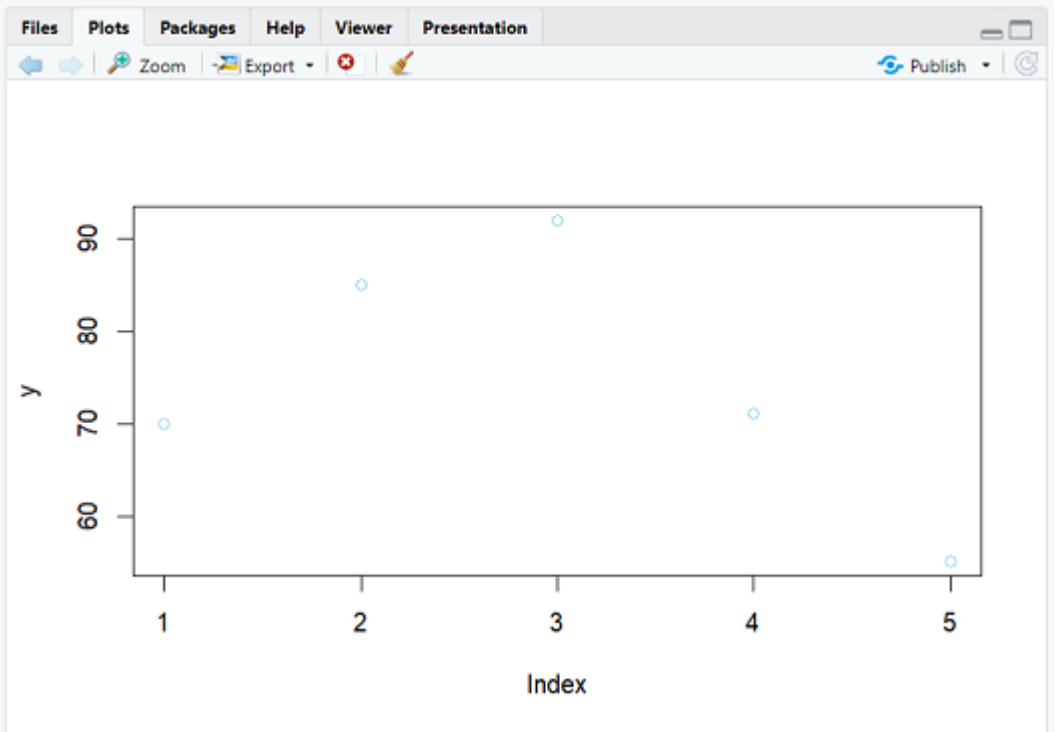
RStudio界面介绍

- **文件窗格**：提供文件浏览器，用于浏览目录和访问文件。支持导入数据文件和组织项目资源。



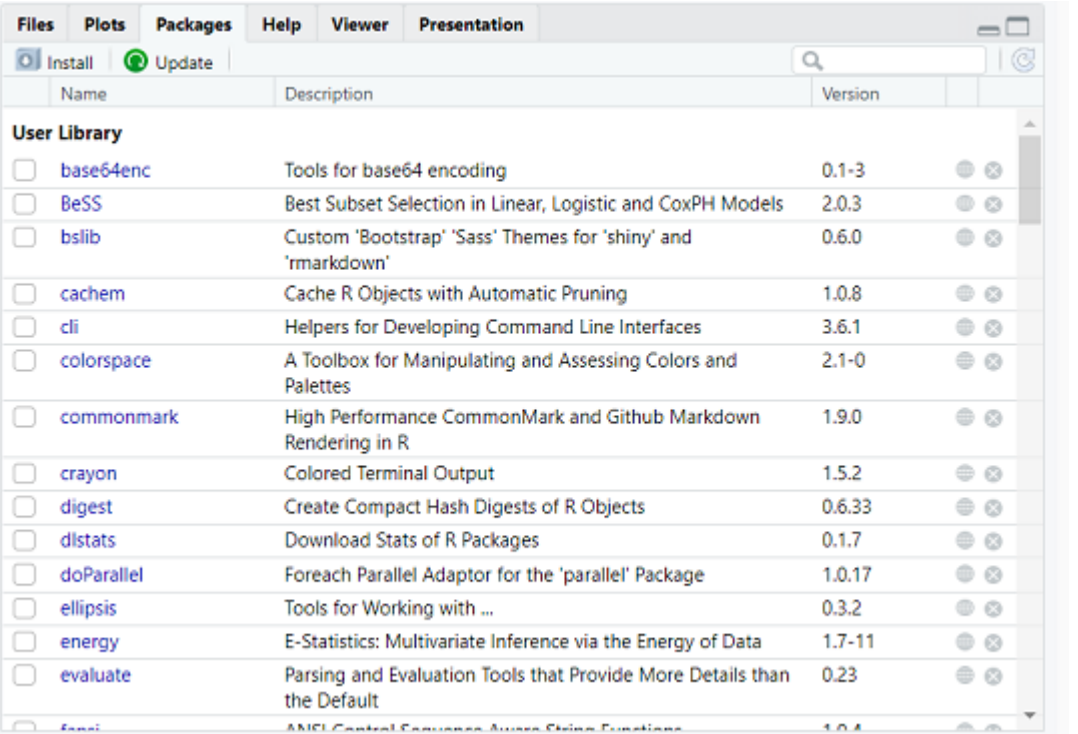
RStudio界面介绍

- **图形窗格**：显示由R代码生成的图形输出，如图表和可视化结果。可进行交互式探索和定制绘图。



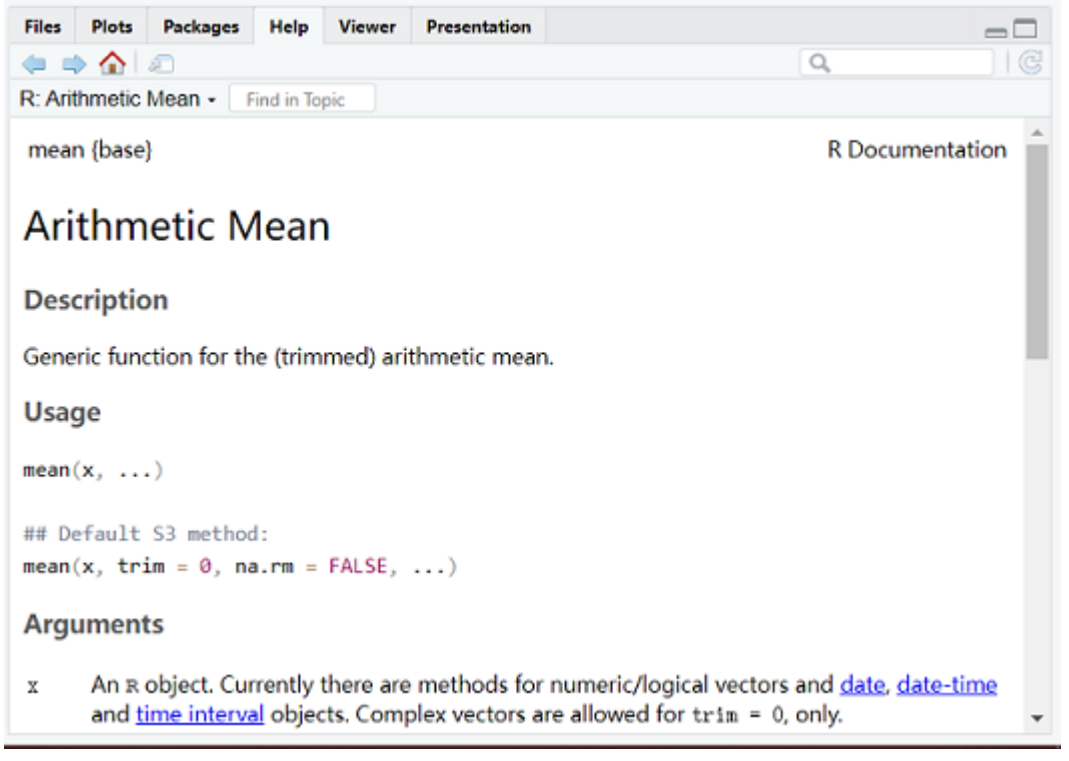
RStudio界面介绍

- **包窗格**：管理安装在系统上的R包。支持安装、加载和更新包以获得额外功能。



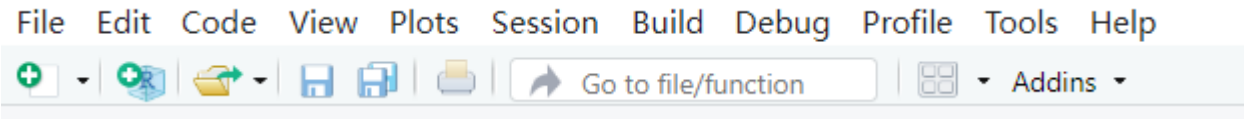
RStudio界面介绍

- **帮助窗格**：提供R函数、包和命令的文档和帮助资源。可访问手册、指南和在线资源以获取帮助。



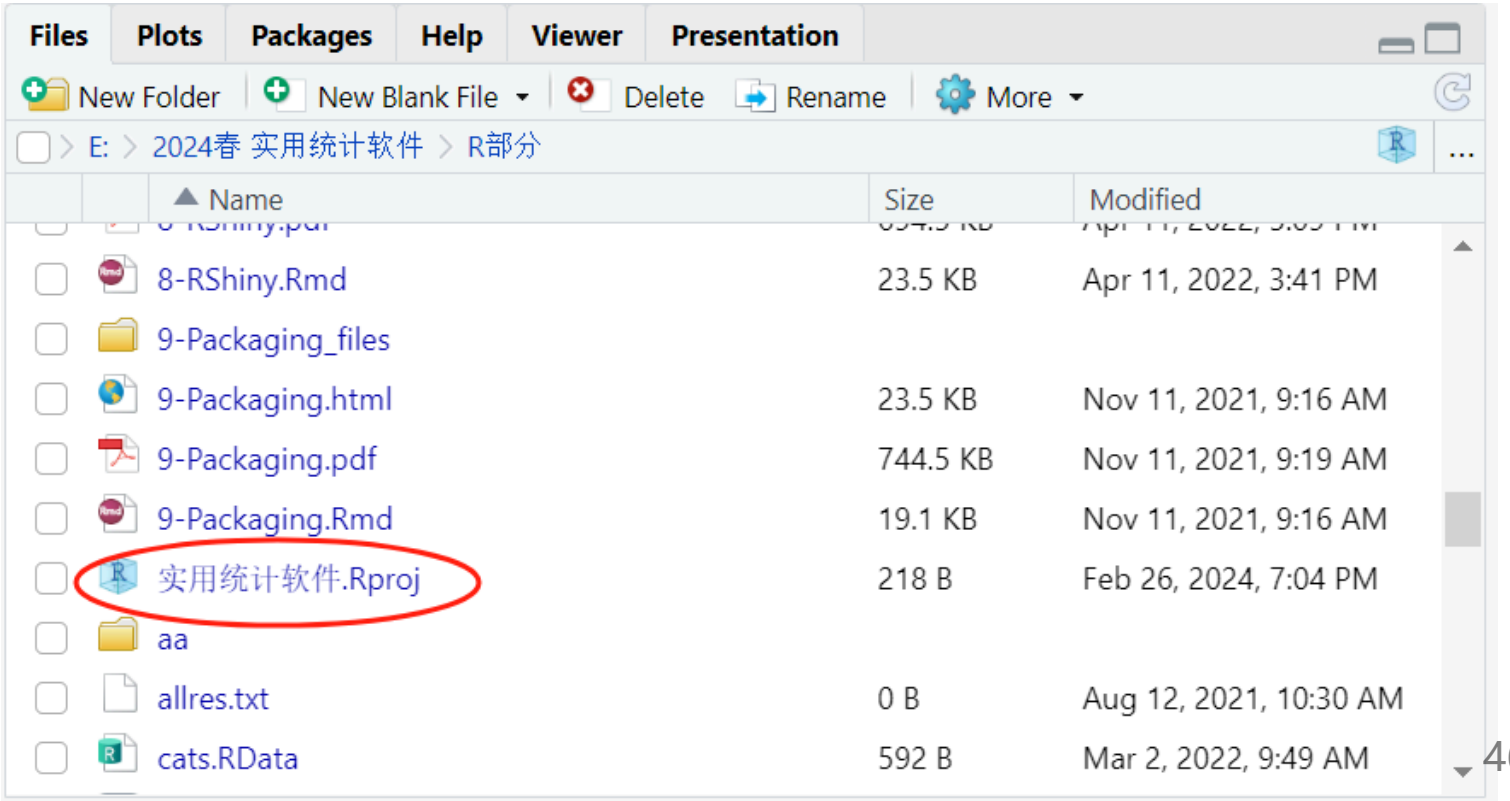
用RStudio管理项目

- 用RStudio进行研究和数据分析，每个研究问题应该单独建立一个文件夹（目录）。
- 确保分析和代码的可重现性和可维护性，简化项目管理。
- 在RStudio中， 用“File – New Project – Existing Directory”选中该问题的目录， 建立一个新的“项目” （project）。



用RStudio管理项目

- 所有的项目信息都保存在"*.Rproj"文件里，每次只要打开".Rproj"，会自动加载你上次退出时的所有环境。
- 把数据、程序等都放在同一文件夹里，便于你在RStudio右下角的"Files"里面查找打开。



Python 语言介绍

什么是Python

Python 是一种开源、扩充性很强的编程语言，在数据科学家中广受欢迎。尤其是最近几年的深度学习研究中，绝大部分的人都采用**Python** 作为分析工具。**Python** 可用作

- 存储和分析数据
- 网络爬虫和数据收集
- 数据可视化
- 自然语言处理和文本分析
- 深度学习
- 等等

Python 的库

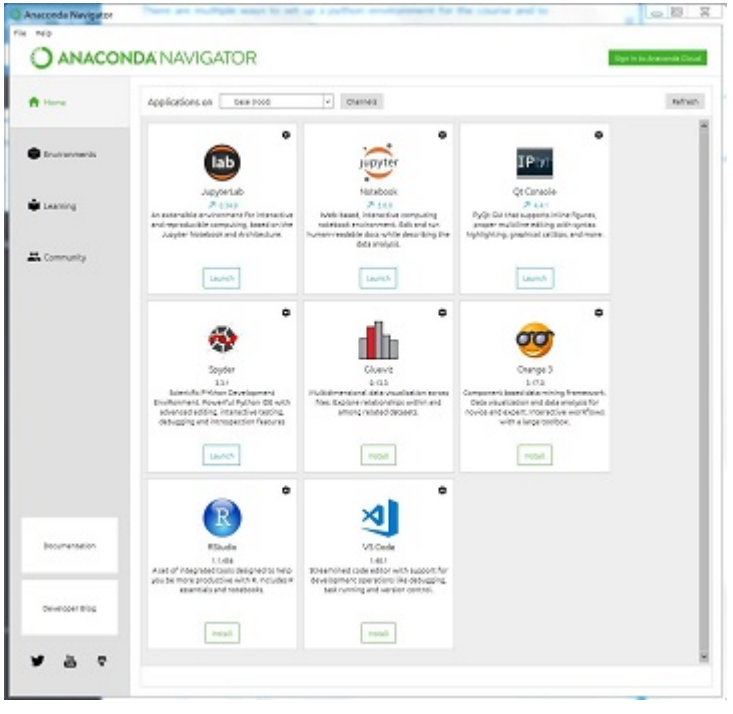
类似于**R**，**Python**也有着成千上万的库来扩展其基本功能，下面介绍几个重要的库：

- `numpy`: Numerical Python. **Python**科学计算的基础库，如线性代数运算、傅立叶变换、随机数的生成等
- `pandas`: panel data. 提供了是我们能够快速便捷处理结构化数据的大量数据结构和函数
- `matplotlib`: 回执数据图表的库，可支持交互式的数据绘图环境
- `scikit-learn`: 机器学习库，包括但不限于分类，回归，聚类，PCA等
- `statsmodels`: 统计建模库，包括但不限于回归，方差分析，时间序列，核密度估计等

Python的下载与安装

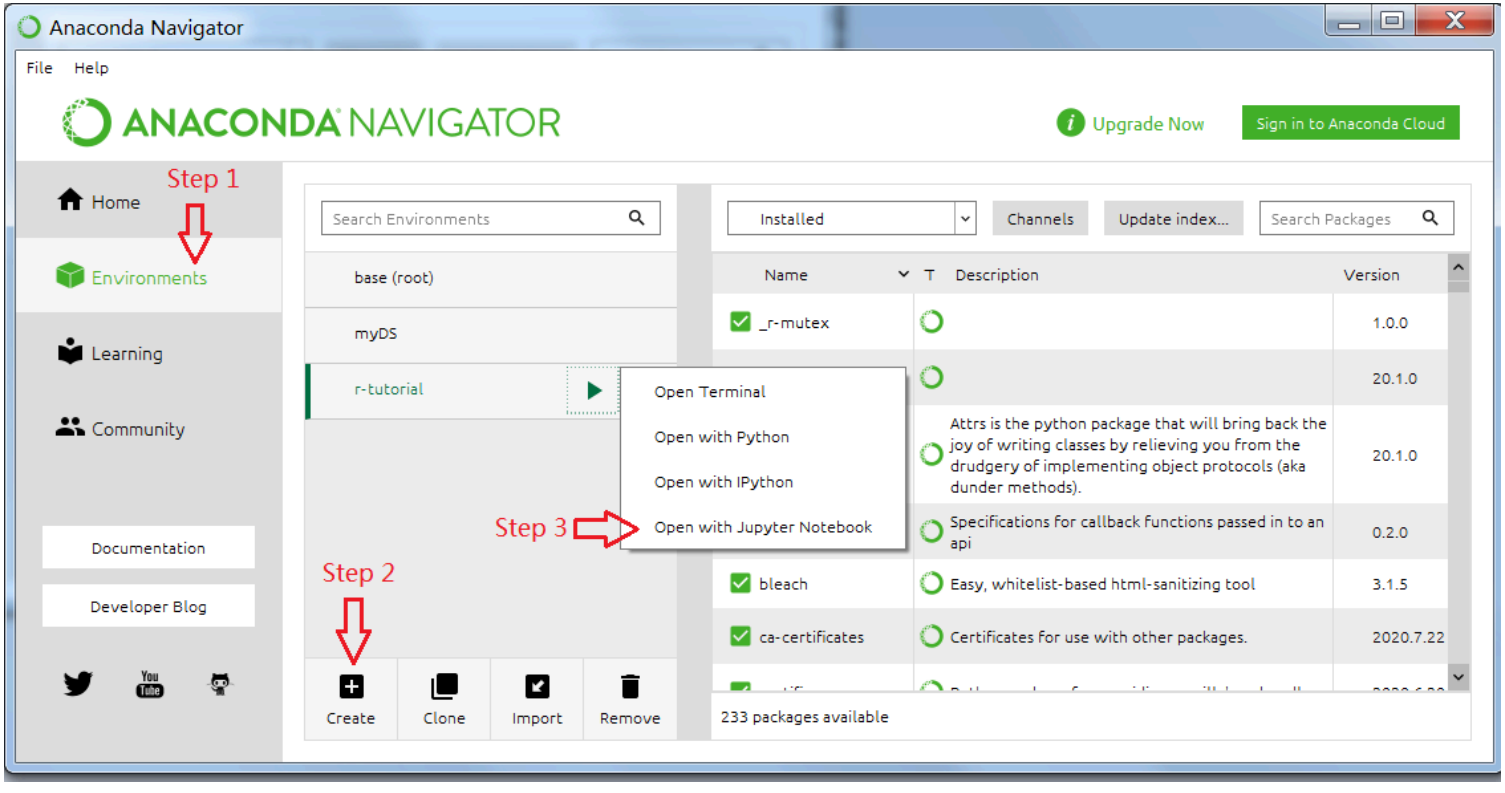
可以通过官网下载**Python**，但更推荐使用通过anaconda来安装。Anaconda是适合数据分析的Python开发环境，其中包括Python，R等常用的科学数据包。

- 下载链接：https://www.anaconda.com/download 或者 https://mirror.tuna.tsinghua.edu.cn/help/anaconda/ (后者更快一些)
- 下载后按照提示安装即可



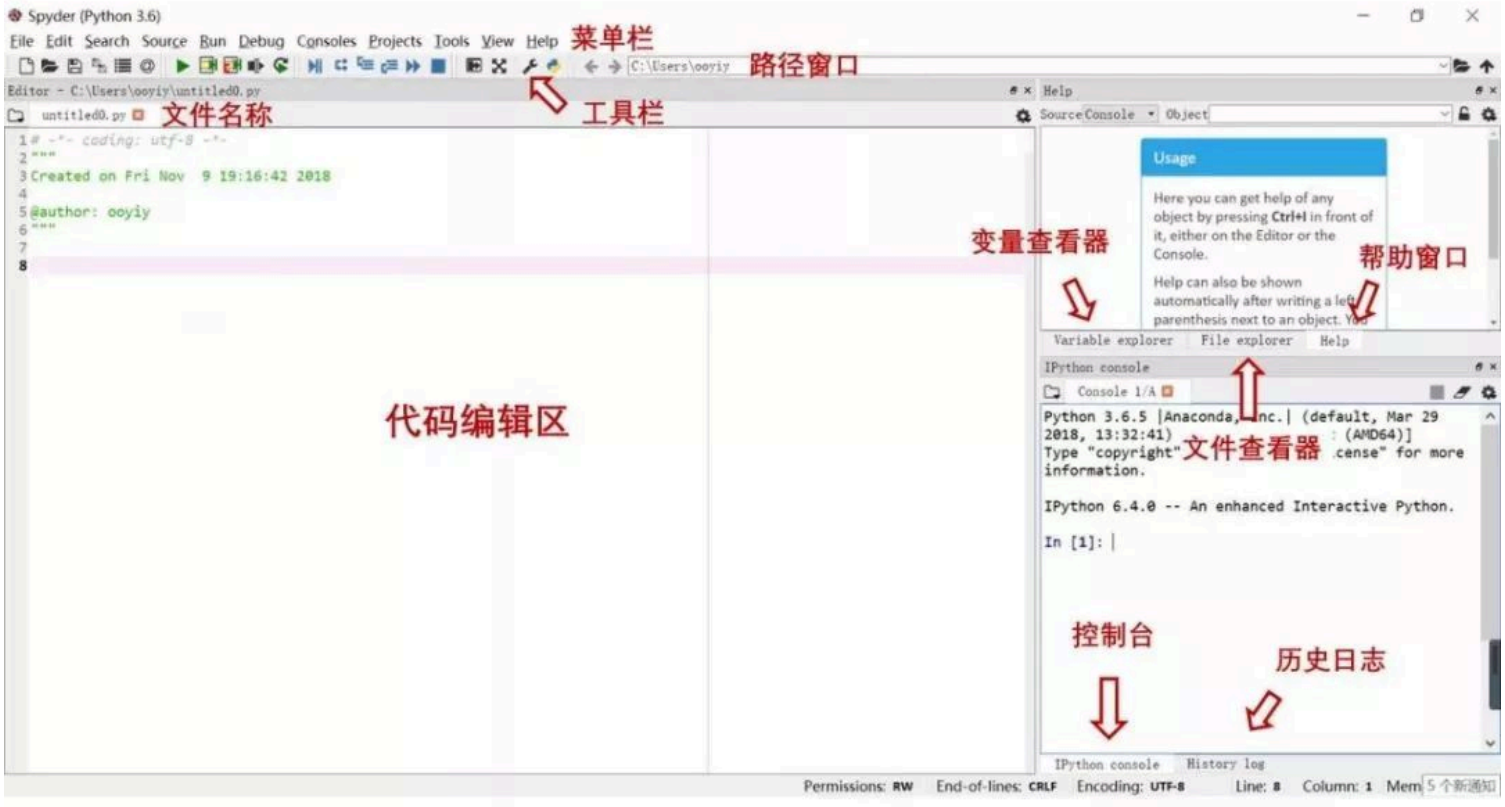
在anaconda中新建环境

类似于R的项目，搭建一个合适的环境，更方便接下来的数据分析、程序开发工作。



Spyder

类似于**RStudio**，是关于**Python**的集成开发环境（Scientific PYthon Development EnviRonment）



Jupter Notebook

- Jupter Notebook是一个在线编辑器、web应用程序，它可以在线编写代码，并创建和共享文档，支持实时编写代码、数学方程式、说明文本和可视化数据分析图表等。
- 除了支持**Python**语言，也支持**R**和**Julia**等语言，通过更换内核即可。

Jupter Notebook

- 代码单元格里支持以下几种格式：
  - 代码
  - **Markdown**
  - 原始说明文字
- 更多的可以查看在线参考指南

Markdown介绍

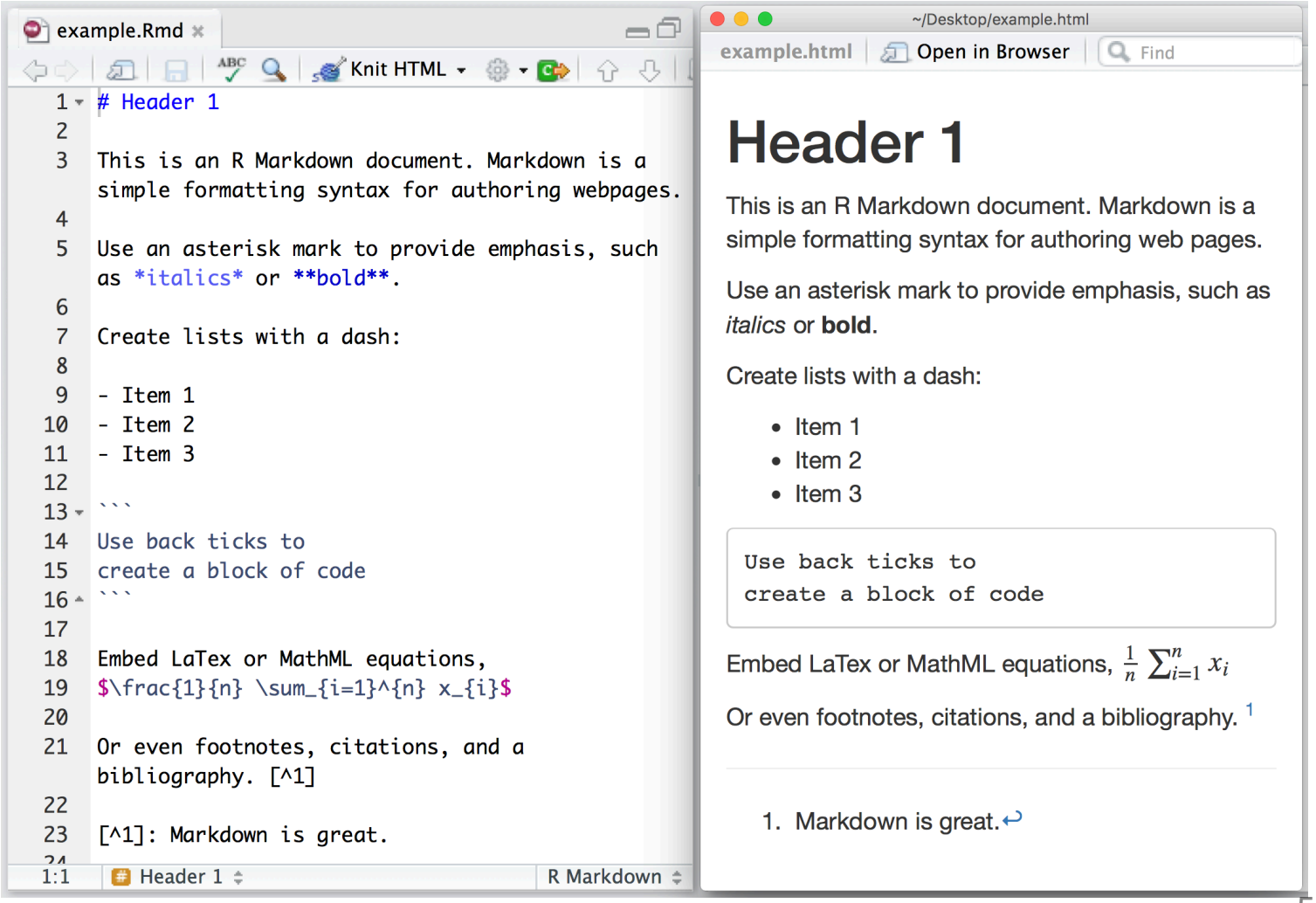
Markdown是什么?

- **Markdown** 是一种把撰写报告与计算程序有机地结合在一起，并且可以产生可重复性报告的文本文件格式，通常保存为.md扩展名。
- **Markdown** 的语法全由一些符号所组成， 这些符号经过精挑细选，其作用一目了然。 比如：
  - 在文字两旁加上一对星号，看起来就像*斜体*；加两对星号就是**强调**。
  - **Markdown** 的列表看起来就像我们平常在邮件中写一个列表的方法。
  - **Markdown** 的区块引用看起来就真的像是引用一段文字。
- 支持LaTeX数学公式，如x^2显示为 x^2。
- 代码支持**R**, **Python**, **C**等，代码包在两个反向单撇号内。如`sin(pi)`

R Markdown

- 借助于R的knitr和rmarkdown扩展包的帮助， 可以在Markdown格式的源文件中插入R代码， 使得R代码的结果能够自动插入到最后生成的研究报告中。
- R Markdown格式，简称为Rmd格式，相应的源文件扩展名为".Rmd"。
- 输出格式可以是HTML、docx、pdf、beamer等。
- 安装：`install.packages("rmarkdown")`

R Markdown编译界面



初识 R Markdown

- 动态文档：撰写报告、绘制图表等。
- 根据数据动态地改变图表，结果具有可重复性。

比如:

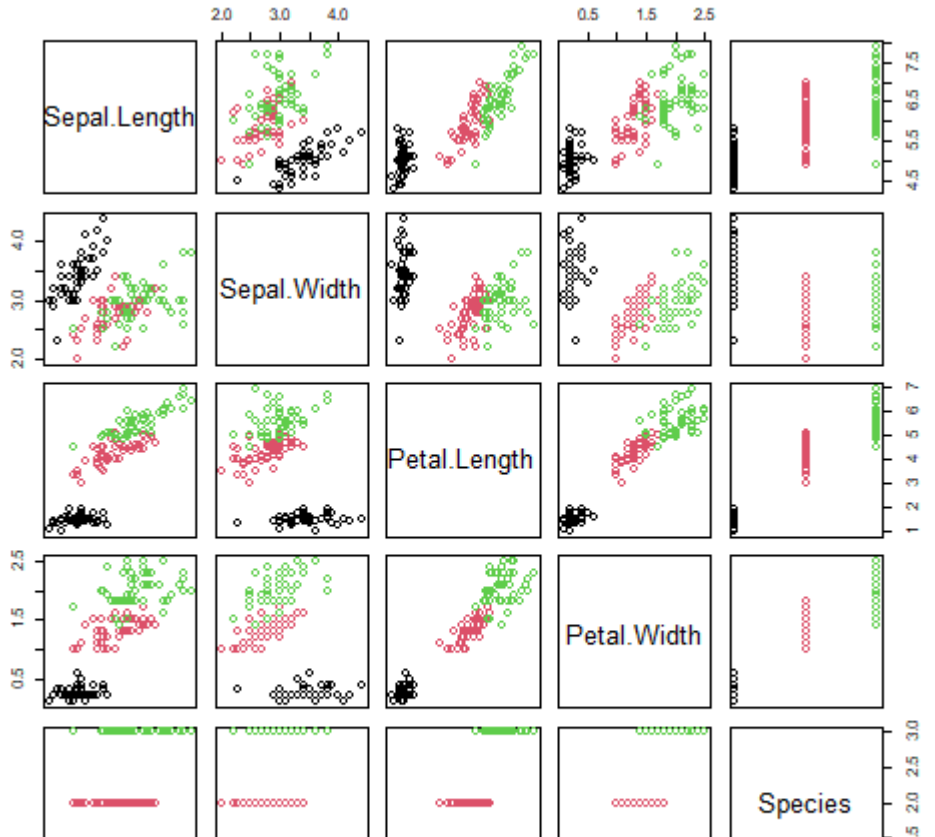
```
knitr::kable(head(iris), format = "html")
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa

初识 R Markdown

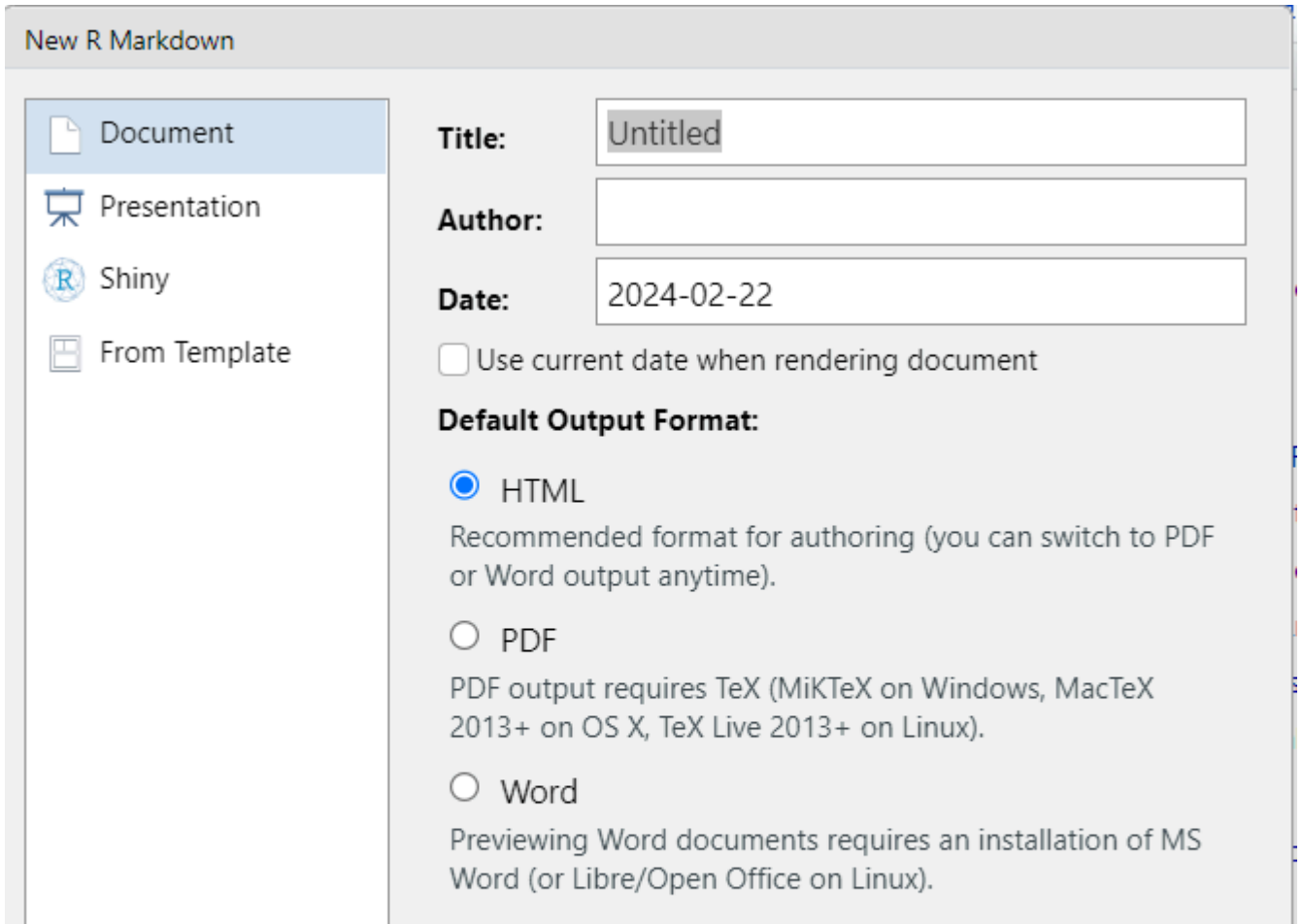
又比如：

```
plot(iris, col=iris$Species)
```



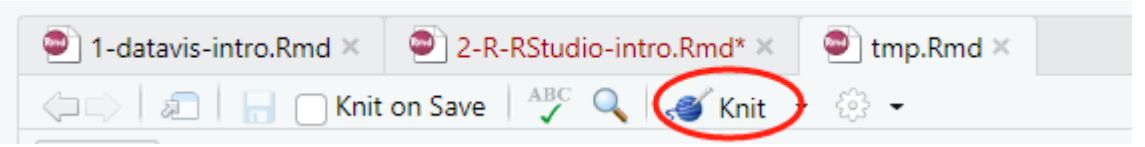
我的第一个.Rmd文档

- 在RStudio中， 用“File – New File – R Markdown...”弹出以下窗口，根据需要新建".Rmd"。



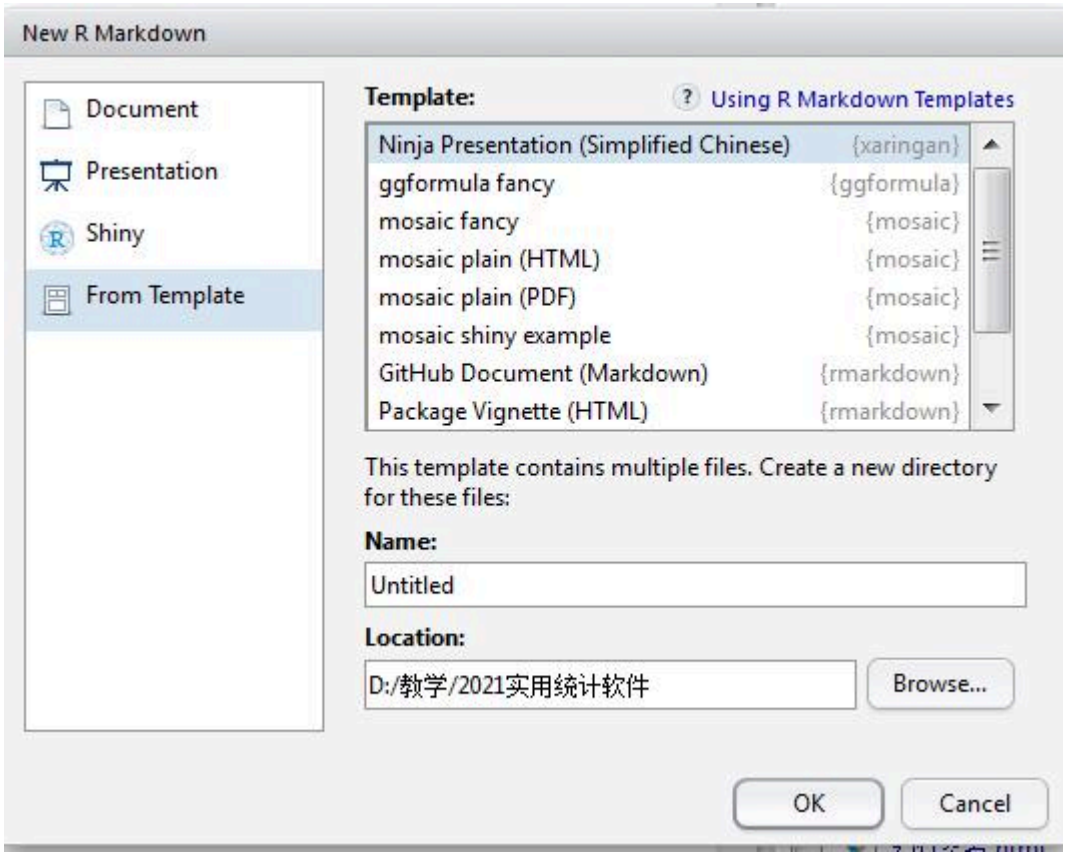
我的第一个.Rmd文档

- 编辑好即可点击"knit"进行编译产生html文档:



R Markdown中文排版

- R Markdown默认语言是英文，如果你想写中文的幻灯片，那么xaringan包是你的不二之选。
- 安装：install.packages("xaringan")



Markdown标题

- 以一个井号#开始的行是一级标题， 以两个井号##开始的行是二级标题。如：

一级标题

二级标题

三级标题

一级标题

二级标题

三级标题

三个或者以上的***输出为分割线

Markdown引用

> 鹅鹅鹅，曲上向天歌，白毛浮绿水，红掌拨清波

> 鹅鹅鹅，曲上向天歌，白毛浮绿水，红掌拨清波

输入为：

```
> 鹅 鹅 鹅
>
> 曲上向天歌
>
> 白毛浮绿水
>
> 红掌拨清波
```

输出为：

> 鹅 鹅 鹅
>
> 曲上向天歌
>
> 白毛浮绿水
>
> 红掌拨清波

Markdown列表——有序列表

输入为：

```
1. 有序列表1
    1. 有序列表1.1
    2. 有序列表1.2
2. 有序列表2

3. 有序列表3
```

输出为:

1. 有序列表1
    1. 有序列表1.1
    2. 有序列表1.2
2. 有序列表2

3. 有序列表3

Markdown列表——无序列表

输入为：

```
- 无序列表1
    - 无序列表1.1
    - 无序列表1.2
- 无序列表2

- 无序列表3
```

输出为:

- 无序列表1
    - 无序列表1.1
    - 无序列表1.2
- 无序列表2
- 无序列表3

Markdown文字格式

输入为:	输出为:
斜体	*斜体*
_斜体_	*斜体*
粗体	**粗体**
粗体	**粗体**
斜粗体	***斜粗体***
~~划掉~~	~~划掉~~

Markdown文字颜色

比如你想将一段文字的颜色改为红色，你可以定义一个 CSS 类，如：

```
.red {
  color: #FF0000;
}
```

我们把这段代码保存在一个 CSS 文件中，如 `extra.css`（假设它跟你的 R Markdown 文件在同一文件夹下），然后通过 `css` 选项将它引入：

```
output:
  xaringan::moon_reader:
    css: ["zh-CN.css", "extra.css"]
```

现在在 R Markdown 中你就可以用 `.red[]` 来标记一段文字为红色，如 `.red[我是红色的]`显示为我是红色的。

Markdown插入表格

输入为：

```
表头一|表头二|表头三
-|-|-
A|B|C
甲|乙|丙
```

输出为:

表头一	表头二	表头三
A	B	C
甲	乙	丙

输入为：

```
居中|右对齐|左对齐|默认
:-:|-:|:-|-
A|B|C|D
甲|乙|丙|丁
:-:|-|-:|:-:|-
```

输出为:

居中	右对齐	左对齐	默认
A	B	C	D
甲	乙	丙	丁

Markdown插入超链接和图片

- 自动超链接：http://www.ustc.edu.cn
- 标准超链接:
  - 输入：[中科大](http://www.ustc.edu.cn)
  - 输出： 中科大
- 带标题（鼠标）的标准超链接：
  - 输入： [中科大](http://www.ustc.edu.cn "南七技校")
  - 输出： 中科大
- 插图图片：

Markdown源代码

- 如想要执行代码并输出结果，则需要放入代码块内，如

输入为：

```
```{r}
sin(pi)
```
```

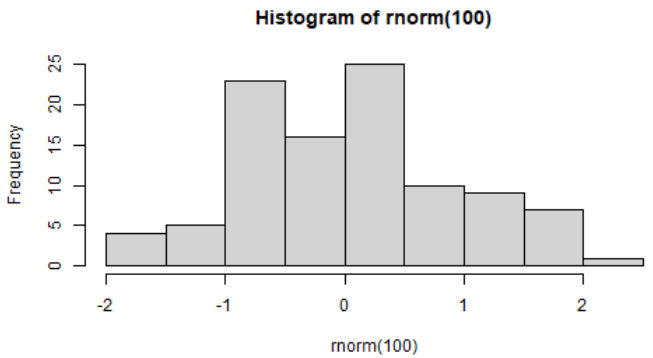
输出为

```
```r
sin(pi)
```

```
## [1] 1.224606e-16
```

输入为：

```
```{r}
hist(rnorm(100))
```
```


```

Markdown中的数学公式

- 支持插入LaTeX格式的数学公式， 编译成HTML或者docx格式后都可以正常显示数学公式。
- 数学公式公式分为行内公式和独立公式。
  - 行内公式和段落的文字混排， 写在两个美元符号\$中间，如$z= \dfrac{2x}{3y}$输出为 $z = \dfrac{2x}{3y}$。
  - 独立公式写在成对的美元符号中间，如 \$$z= \dfrac{2x}{3y}\$$输出为

$$z = \dfrac{2x}{3y}$$

- 有关LaTeX格式的数学公式，请查阅相关文献。

谢 谢