

Multi-omics data analysis and visualisation, #1

HUST Bioinformatics course series

Wei-Hua Chen (CC BY-NC 4.0)

28 November, 2019

section 1: TOC

Course contents

- ① 开发平台相关软件安装
- ② R 基础知识
- ③ 数据处理
- ④ 做图
- ⑤ 常用生信分析任务与解决方法

注：不一定按上面的顺序来

Class rules

- ① 每次随机点名
- ② 课堂随机提问
- ③ 每人 2 次无理由缺课机会，占平时成绩的一半
- ④ 考试、平时成绩各占总成绩的 50%

注：严格按以上进行，如多次表现不好，可能会不及格

section 2: why choose R?

R 语言简史

1993 到 2000 这段时间 R 只在小范围内流传。2000 年之后开始大爆发，用户数量直线上升。除去 R 本身的优秀之外，这种爆发与多个因素有关，比如自由软件的兴起，Linux 的成熟等等；经济危机也促进大家采用免费的自由软件替代统计领域的传统强者如 SPSS、SAS 和 Matlab 等（注：均为收费软件）。

首先，越来越多的学术文章使用 R 作为分析工具。根据来自著名学术搜索引擎 Google Scholar（谷歌学术）的数据，R 的流行趋势有以下两个特点：
1) 在学术领域的市场份额逐年增加，且增势迅猛，2) R 是为数不多市场份额增加的统计软件之一。

接下来我们就用 R 把这个趋势画出来！如下面代码所示，所需代码包括 4 个部分：装入所需要的包，读取数据，处理数据和作图。运行这段代码，既专业又美观的图片就生成了！

R 的流行性调查

代码

```
library("ggplot2"); library("reshape2");

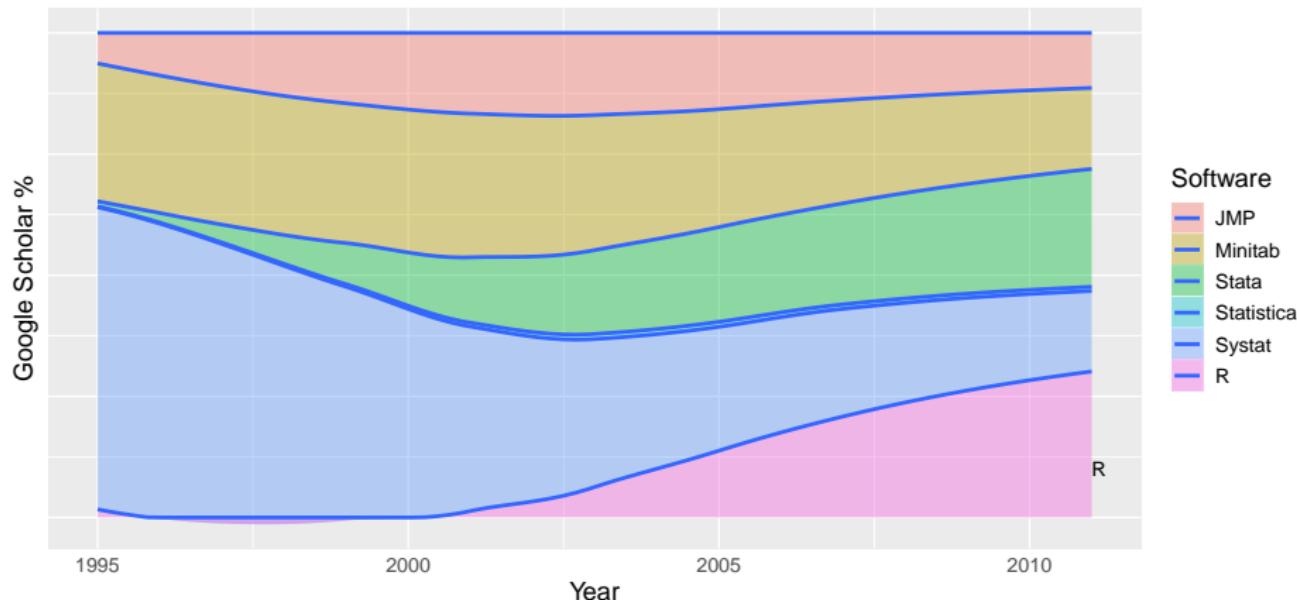
dat <- read.csv(file = "data/talk01/chaper01_preface_scholarly_impact_2012.4.9.csv");

cols.subset <- c("Year", "JMP", "Minitab", "Stata", "Statistica", "Systat", "R");
Subset <- dat[, cols.subset];
ScholarLong <- melt(Subset, id.vars = "Year");
names(ScholarLong) <- c("Year", "Software", "Hits");

plot1 <-
  ggplot(ScholarLong, aes(Year, Hits, group=Software)) + # 准备
    geom_smooth(aes(fill=Software), position="fill", method="loess") + # 画图
    ggtitle("Market share") + # 设置图标题
    scale_x_continuous("Year") + # 改变 X 轴标题
    scale_y_continuous("Google Scholar %", labels = NULL) +
    theme(axis.ticks = element_blank(), text = element_text(size=14)) +
    guides(fill=guide_legend(title = "Software", reverse = F)) +
    geom_text(data = data.frame(Year = 2011, Software = "R", Hits = 0.10),
              aes(label = Software), hjust = 0, vjust = 0.5);
```

Market share, result

Market share



注：这里移除了市场占有率较大的 SAS 和 SPSS

R 的招聘趋势

其次，统计分析相关工作的招聘信息中要求申请者会用 R 的也越来越多了。根据美国招聘搜索引擎 indeed.com 的数据，自 2005 年（此搜索引擎提供的最早数据）起，需要用到 R 的招聘信息占总体招聘的比例逐年上升，目前仅排在 SAS 和 Matlab 之后，处于第 3 位。而且，除了 Stata 之外，R 是唯一一个占比上升。

同样的，我们用 R 把这个趋势画出来！

R job trends

代码

```

library("ggplot2"); ## 主作图包

##2. -- 读取数据 --
dat <- read.table(file ="data/talk01/chaper01_preface_indeed_com_stats_2015.txt",
                  header = T, as.is = T);
##3. 处理数据
dat$date <- as.Date(dat$date); ## 把第一列改为日期

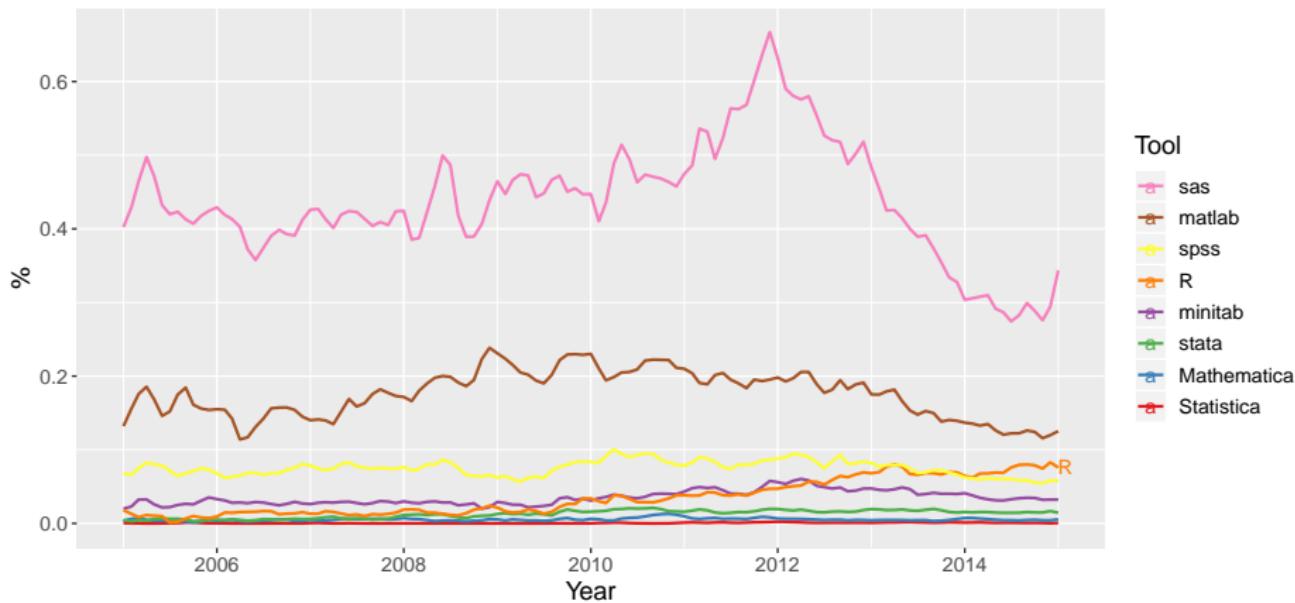
# 根据 job 对 software 进行调整
dat <- transform(dat, software = reorder(software, job));

plot2 <-
  ggplot( dat, aes( date, job, group = software, colour = software) ) +
  geom_line( size = 0.8 ) +
  ggtitle("Job trends (data from indeed.com)") + # 设置图标题
  xlab("Year") + ylab("%") +
  # 改变字体大小; 要放在 theme_grey() 后面
  theme( text = element_text(size=14) ) +
  guides(colour=guide_legend( title = "Tool", reverse = TRUE )) +
  scale_colour_brewer(palette="Set1") + # 改变默认颜色
  geom_text(data = dat[dat$date == "2015-01-01" & dat$software %in% c("R"), ],
            aes(label = software), hjust = 0, vjust = 0.5);

```

R job trends, plot

Job trends (data from indeed.com)



Programming language trends 2016 vs 2015

| Rank | Change | Language | Share | Trend |
|------|--------|-------------|--------|--------|
| 1 | | JAVA | 23.7 % | -0.1 % |
| 2 | | Python | 14.0 % | +2.6 % |
| 3 | | PHP | 9.7 % | -1.0 % |
| 4 | | C# | 8.4 % | -0.4 % |
| 5 | ▲ ▲ | Javascript | 7.9 % | +0.7 % |
| 6 | ▼ | C++ | 6.9 % | -0.9 % |
| 7 | ▼ | C | 6.8 % | -0.8 % |
| 8 | | Objective-C | 4.6 % | -0.5 % |
| 9 | ▲ | R | 3.3 % | +0.5 % |
| 10 | ▼ | Swift | 3.1 % | +0.3 % |

Figure 1: Worldwide Google Trends, Dec 2016 vs Dec 2015

Github pulls, 2017



Figure 2: Github pulls, 2017

Popularity by year

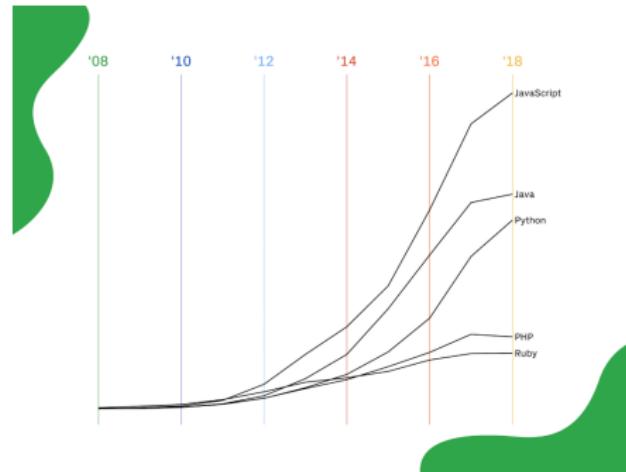


Figure 3: popularity by year from 2008 to 2018

Popularity by year, cont.

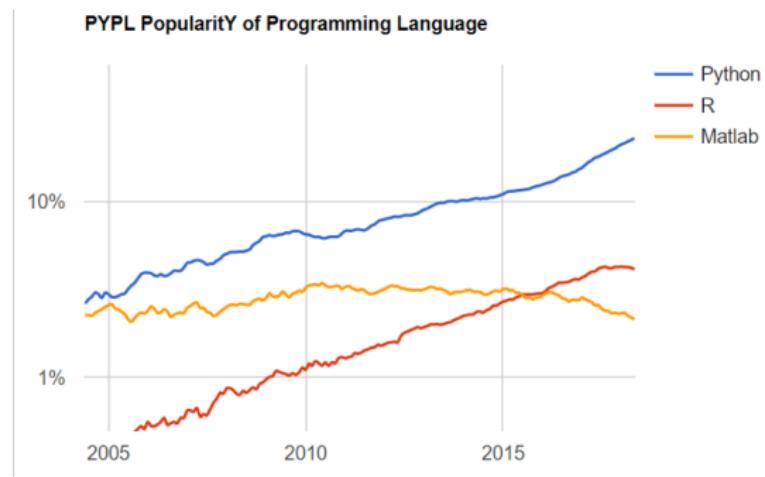


Figure 4: yet another statistics

Data from: <http://blog.revolutionanalytics.com/2018/06/pypl-programming-language-trends.html> Accessed on: Sep 24, 2018

Programming languages for bioinformatics

Perl 或 Python

- 强大的文本处理能力（包括序列）
- 不错的运行速度（尤其是 Python）
- 强大的生信和统计学扩展包（尤其是 Python）
- 方便的并行计算

R

- 强大的格式数据处理能力（二维表格, dplyr）
- 无以伦比的统计学专业性
- 专业而好看的数据可视化软件（ggplot2）
- 专业的生信扩展包（Bioconductor）
- 超级好用的整合开发环境 IDE（RStudio）

我用过的 programming languages

- - C
- - Perl
- - R
- - PHP
- - Java
- - MySQL
- - HTML
- - Javascript

Evolview ver3.0

cited 95 times in 2018, 75 so far
(as of July 19)

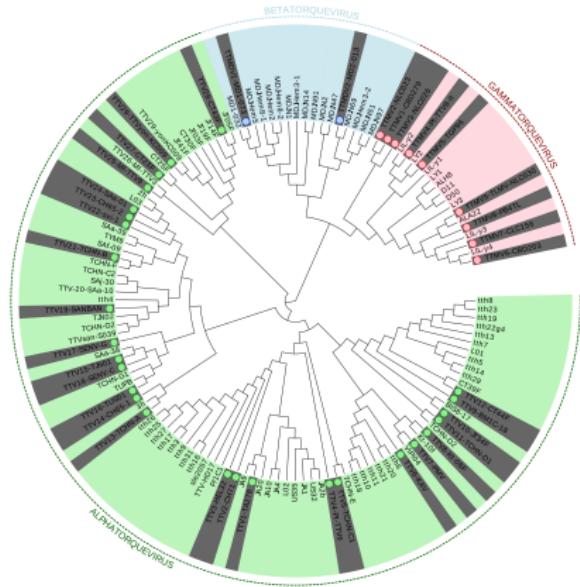


Figure 5: .Evolview showcase 3

网站链接、参考文献和扩展阅读

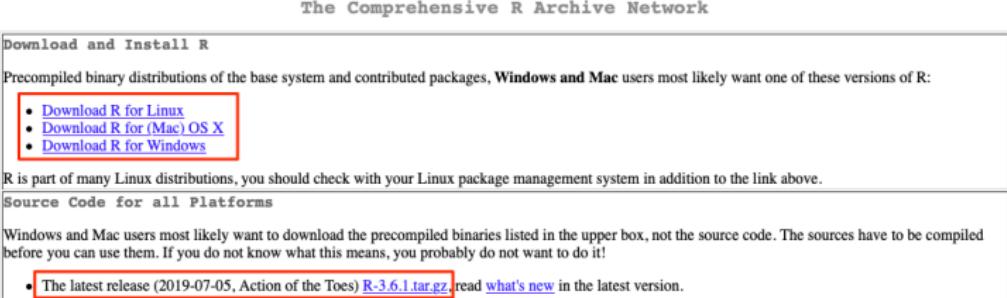
综上所述，R 已经是最流行的免费统计分析软件，排名仅在几个传统的分析软件之后，而且大有赶超它们的趋势。学好 R，不仅有助于在学术研究领域的发展，对找工作也有不少的帮助。

- R 的官方网站: <http://www.r-project.org>
- R 档案综合网络，即 CRAN(Comprehensive R Archive Network):
<http://cran.r-project.org/>
- ggplot2: <http://ggplot2.org/>
- RStudio: <http://www.rstudio.com/>
- 如何从 Google Scholar 抓取引用数据:
<http://librestats.com/2012/04/12/statistical-software-popularity-on-google-scholar/>
- indeed 招聘趋势: www.indeed.com/jobtrends
- R for data science: <https://r4ds.had.co.nz> (必读!!)

Section 3: setting up working environment

Install R

Go to <https://mirrors.tuna.tsinghua.edu.cn/CRAN/> (清华镜像),
 R supports mainstream operating systems including Linux, Windows and MacOS, please download the corresponding installation files according to the operating system.



The screenshot shows the "Download and Install R" section of the CRAN website. It lists precompiled binary distributions for Windows and Mac users, with three specific links highlighted in a red box:

- Download R for Linux
- Download R for (Mac) OS X
- Download R for Windows

Below this, a note states: "R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above." Further down, it says: "Source Code for all Platforms" and provides a link to the latest release (R-3.6.1.tar.gz) with a note: "The latest release (2019-07-05, Action of the Toes) [R-3.6.1.tar.gz](#), read [what's new](#) in the latest version."

Figure 6: 选择合适的安装包

新版本的 Mac OS X 还需要安装

XQuartz(<http://xquartz.macosforge.org/landing/>)。某些还需要用到 Xcode，可以从 App Store 免费安装。

Install R on Linux

目前大多 Linux 发行版都带有 R，因此可直接使用。从 CRAN 下载文件进行安装稍嫌复杂，要求用户对 Linux 系统有一定的了解，而且需要有管理员权限。建议初级用户在 Linux 高手指导下安装。点击上图中的“Download R for Linux”后，发行版为 Redhat（红帽）或 Suse 的用户要先阅读网站上提供的 `readme` 或 `readme.html` 文件，然后其中的指示进行安装。这里就不再累述了。

r-base-core_3.1.3-1lucid_amd64.deb 或 r-base-core_3.1.2-1lucid0_i386.deb
----- ----- ----- ----- -----
1 2 3 4 5 1 2 3 4 5

Figure 7: R 安装包文件名

R studio

RStudio 可以从 <http://www.rstudio.com/products/rstudio/download/> 下载，支持等主流的操作系统。

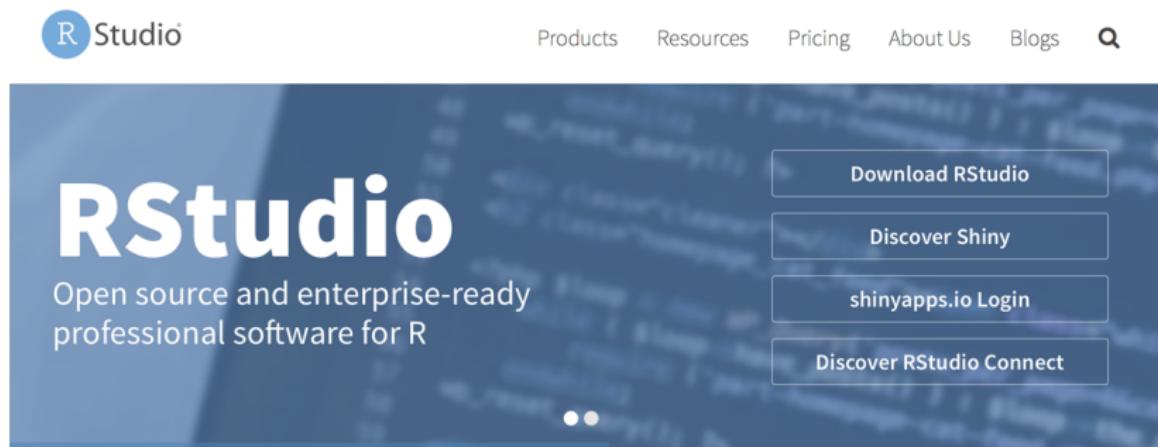


Figure 8: RStudio website main page

R studio versions

RStudio 有商业和免费版本；也有 server 版

| | RStudio Desktop Open Source License | RStudio Desktop Commercial License | RStudio Server Open Source License | RStudio Server Pro Commercial License | RStudio Server Pro + RStudio Connect Commercial License |
|------------------------|--|---|--|--|---|
| Integrated Tools for R | FREE DOWNLOAD Learn More | \$995 per year BUY Learn More | FREE DOWNLOAD Learn More | \$9,995 per year DOWNLOAD Learn More | \$29,995 per year TALK Learn More |
| Priority Support | ● | ● | ● | ● | ● |
| Access via Web Browser | | | ● | ● | ● |
| Enterprise Security | | | | ● | ● |
| Project Sharing | | | | ● | ● |

R studio, cont.

RStudio 运行时的界面如下图所示，除了顶部的菜单栏工具栏之外，主界面还包括 4 个子窗口：

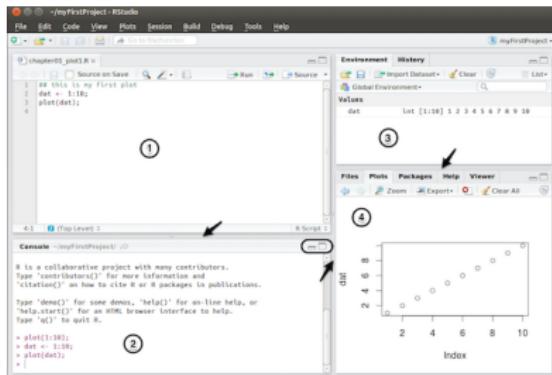


Figure 9: R studio 单机版主界面

R studio, cont.

1. 代码编辑器

- 具有代码编辑、语法高亮、代码和变量提示、代码错误检查等功能
- 选中并向 R 控制台（窗口 2）发送并运行代码。用快捷键 Ctrl+Enter (MacOS 下是 Cmd+Enter) 进行代码发送。没有代码选中时，发送光标所在行的代码
- 可同时打开编辑多个文件
- 除 R 代码外，还支持 C++、R MarkDown、HTML 等其它文件的编辑
- 也可用于显示数据

2. R console

- 可在此直接输入各种命令并查看运行结果。支持代码提示

3. 变量列表及代码运行的历史记录

R studio, cont.

4. 其它窗口

- 当前工作目录下的文件列表
- 作图结果
- 可用和已安装的扩展包；在这里可以直接安装新的和升级已有的扩展包
- 帮助

注意，子窗口之间可以通过快捷键 $\text{Ctrl} +$ 子窗口编号进行切换。如 $\text{Ctrl} + 1$ 可以切换到代码编辑子窗口， $\text{Ctrl} + 2$ 则切换到 R 控制台。

其它特点

- 创建、管理 projects

R studio 特点详解

代码提示/自动完成

子窗口 1 和 2 都提供有代码提示功能，即：用户输入 3 个字母时，RStudio 会列出所有前 3 个字母相同的变量或函数名供用户选择；用户可通过键盘的上下键选择，然后用 Enter（回车）选定，非常方便。变量或函数名前面的小图标表示了它们的类型；如果当前高亮的是函数，RStudio 还会显示其部分帮助内容。

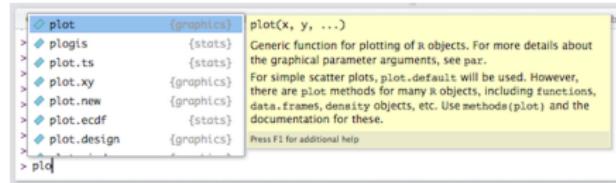


Figure 10: R studio code autocomplete

R studio 特点详解, cont.

查看变量内容

子窗口 3 内会列出所有当前使用的变量、变量的类型以及大小，如下图：

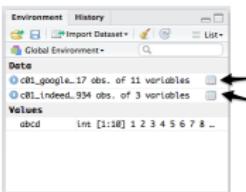


Figure 11: loaded variables

有些简单变量，如数组，RStudio 会直接显示其部分值；对于复杂一些的变量，比如 data.frame（类似于二维表格），则可以点击变量名前边的小三角标识展开其内容。当变量的最右侧出现小网格状图标时（如上图箭头所指位置），点击它们后可以在子窗口 2 内察看。

R studio 特点详解, cont.

导出作图并选择导出格式

RStudio 的第 4 子窗口里集中了许多有用的功能，组织在不同的‘Tab’（标签）内。比如作图（plots），不仅可以察看画图的结果，还可以导出当前图像至硬盘，或拷贝至剪贴板；如下图所示。支持导出格式有 png 和 pdf。

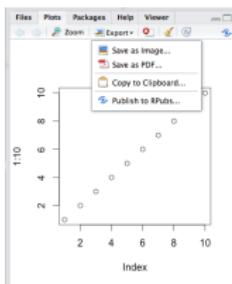


Figure 12: export active plot to various graphical formats

R studio 特点详解, cont.

查看已安装的包

通过第 4 子窗口的“包”(Packages)标签内的工具，用户可以很方便的查看已安装的包：

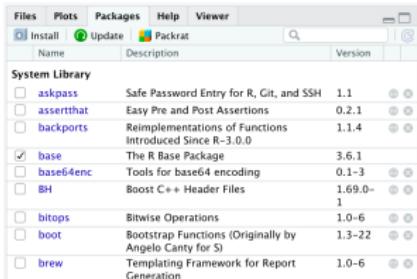


Figure 13: Check installed packages

install new package(s)

同样通过第 4 子窗口的“包”（Packages）标签内的工具，安装新的包：

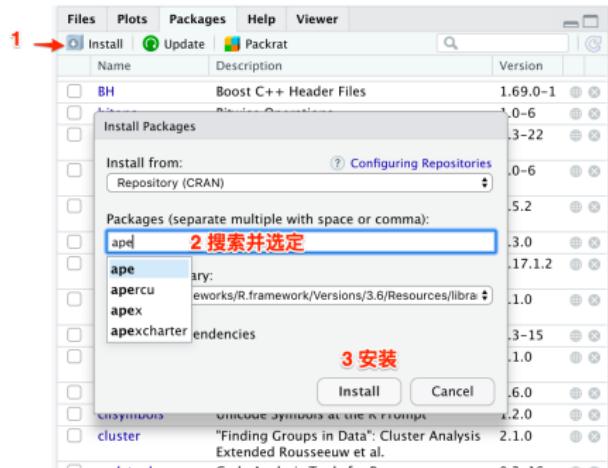


Figure 14: install new package

Packages needed for this study

我们在这里一次性的安装所有本书所需要的包。不过，为方便读者直接从后面章节阅读，在每一次使用新包时，我们会再次进行提示安装方法。

```
install.packages( c("ggplot2", "reshape2") );
```

也可以单独安装：

```
install.packages( "ggplot2" ); # 安装作图用的 ggplot2  
install.packages( "reshape2" ); # 安装数据处理用的 reshape2
```

第一次运行命令 `install.packages()` 时，系统会提示选择镜像网站；请选择地理位置上距你最近的镜像（比如中国）。

install packages, cont.

You can also choose your CRAN mirror manually (recommended when installing takes a long time):

```
chooseCRANmirror();
```

```
> chooseCRANmirror()  
Secure CRAN mirrors  
  
1: 0-Cloud [https] 2: Algeria [https] 3: Australia (Canberra) [https]  
4: Australia (Melbourne 1) [https] 5: Australia (Melbourne 2) [https] 6: Australia (Perth) [https]  
7: Austria [https] 8: Belgium (Ghent) [https] 9: Brazil (PR) [https]  
10: Brazil (RJ) [https] 11: Brazil (SP 1) [https] 12: Brazil (SP 2) [https]  
13: Bulgaria [https] 14: Chile 1 [https] 15: Chile 2 [https]  
16: China (Hong Kong) [https] 17: China (Guangzhou) [https] 18: China (Lanzhou) [https]  
19: China (Shanghai 1) [https] 20: China (Shanghai 2) [https] 21: Colombia (Cali) [https]
```

Figure 15: Choose CRAN mirror

You can also use `chooseBioCmirror()`; to choose mirror for BioConductor packages.

Packages needed for this study, cont.

实际上我们只需要安装一个包就可以了：

```
install.packages("tidyverse")
```

它是以下包的集合，都由 <https://www.tidyverse.org> 开发：



Figure 16: tidyverse: a mega package

R studio server

特点：

- 在服务器上安装，使用服务器的强大计算资源
- 通过网页登录，使用服务器帐号密码（方便，安全）
- 一直运行

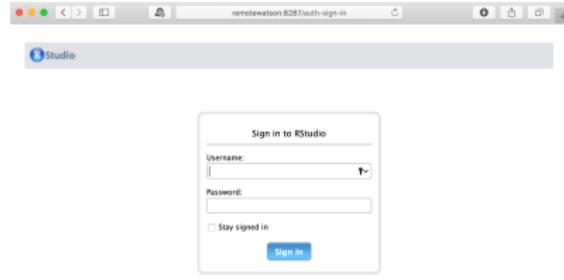


Figure 17: RStudio server web login form (w/ linux account)

R studio server, cont.

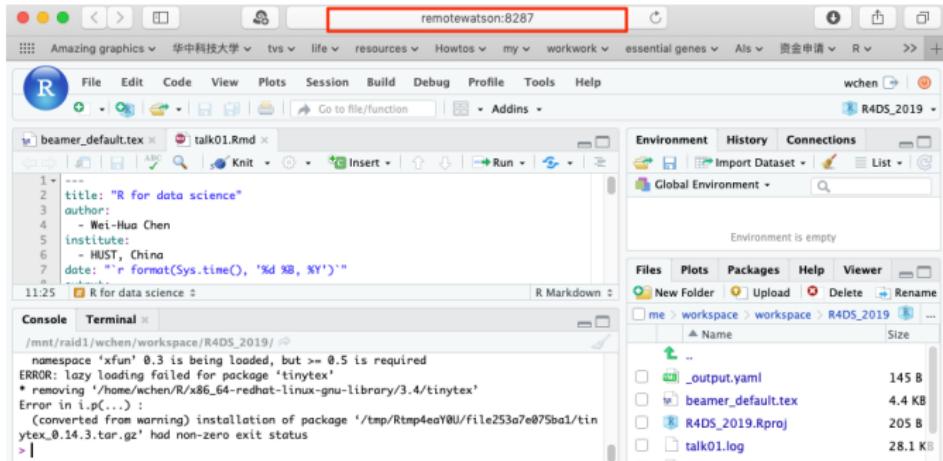


Figure 18: RStudio server 界面（通过浏览器）

R studio packages for data science

all part of (业界良心) tidyverse

- dplyr: 强大且方便的数据处理
- tydvr: 数据转换工具
- readr: 方便的文件 IO
- stringr: 文本处理
- Tibble: 代替 data.frame 的下一代数据存储格式
- purr: (暂时还未用到的包 ~~)

R studio packages for data visualisation

tidyverse

- ggplot2: 专业好用（但学习曲线很陡）的画图工具
 - <http://ggplot2.tidyverse.org>
 - gallery: <http://www.ggplot2-exts.org/gallery/>

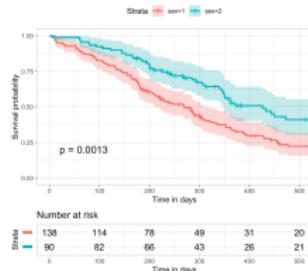


Figure 19: surminer <https://rpkgs.datanovia.com/survminer/index.html>

RStudio packages for data visualisation, cont.

ggvis (currently ver0.4): <http://ggvis.rstudio.com>

- from the **ggplot2** team
- create interactive graphics in RStudio and web browser
- top 50 ggplot2 visualisations

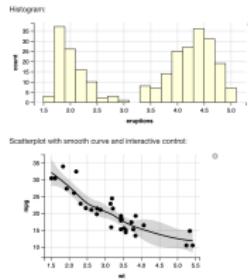


Figure 20: ggvis example plots

RStudio packages for data visualisation, cont.

Shiny: <http://shiny.rstudio.com/gallery/>

- build professional, interactive visualizations
- equipped with popular web widgets
- can be deployed as independent websites

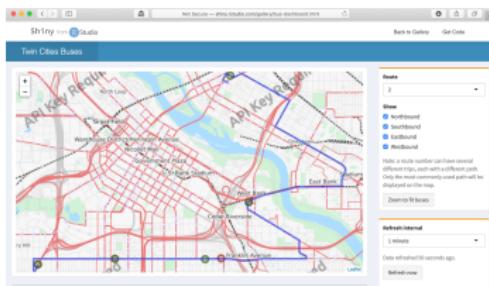


Figure 21: Shiny website example:

<http://shiny.rstudio.com/gallery/bus-dashboard.html>

RStudio packages for data visualisation, cont.

other packages

- rmarkdown : create professional documents
- knitr: convert rmarkdown to pdf, html and more ...

Section 4: other tools used in our group

Database tools

- MySQL
- phpmyadmin



Figure 22: phpmyadmin login page using web browser

Database tools, cont.

Figure 23: phpmyadmin database view

other tools

programming languages

- R
- Python
- Perl

code repository

- Github
- Github desktop
- Atom (general-purpose text editor and more)

web development

- AngularJS
- Vue.js
- Bootstrap
- plot.ly (interactive visualisation)

An example: cross-talking among tools

Here I use the following to show how we process data in our lab:

```
library(RMySQL);
library(dplyr);

mysql.dbname = "r4ds_test";
dbCon <- dbConnect(MySQL(), user="r4ds", password="r4ds",
                   dbname=mysql.dbname );

dat <- dbGetQuery(dbCon, "SELECT * FROM grades_stats");

## Warning in .local(conn, statement, ...): Decimal MySQL column 2 imported as
## numeric

## -- 任务: 为每个人计算: 平均成绩、上课总数、及格门数、不及格门数
stats <- dat %>% group_by(name) %>%
  summarise( avg_grade = mean(grade), count = n(),
             passed = sum( grade >= 60 ), failed = sum( grade < 60 ) ) %>%
  arrange(-avg_grade);
```

Show the data

| name | course | grade |
|-------------|----------------|-------|
| Zhi Liu | Microbiology | 100 |
| Zhi Liu | English | 50 |
| Zhi Liu | Chinese | 69 |
| Weihua Chen | Microbiology | 89 |
| Weihua Chen | English | 99 |
| Weihua Chen | Bioinformatics | 99 |
| Kang Ning | Bioinformatics | 100 |
| Kang Ning | Chinese | 20 |
| Kang Ning | Chemistry | 76 |

and show the results

| name | avg_grade | count | passed | failed |
|-------------|-----------|-------|--------|--------|
| Weihua Chen | 95.66667 | 3 | 3 | 0 |
| Zhi Liu | 73.00000 | 3 | 2 | 1 |
| Kang Ning | 65.33333 | 3 | 2 | 1 |

Concluding remarks

- 生信分析必备编程语言
- 强大、专业又好用
- RStudio 及其众多扩展包
- To be continued

About the course

- all codes are available at Github:

<https://github.com/evolgeniusteam/R-for-bioinformatics>

作业

- 安装 R、RStudio
- 熟悉 Github
- 自学 R 编程