

R动态图

周世祥

2020年5月20日

除了上述R内置的绘图功能之外，R还可以通过安装和加载可视化功能包极大地扩展可视化功能，比较常用的可视化功能包有ggplot2、rCharts、plotly()、map(用于绘制地图)等。

ggplot2

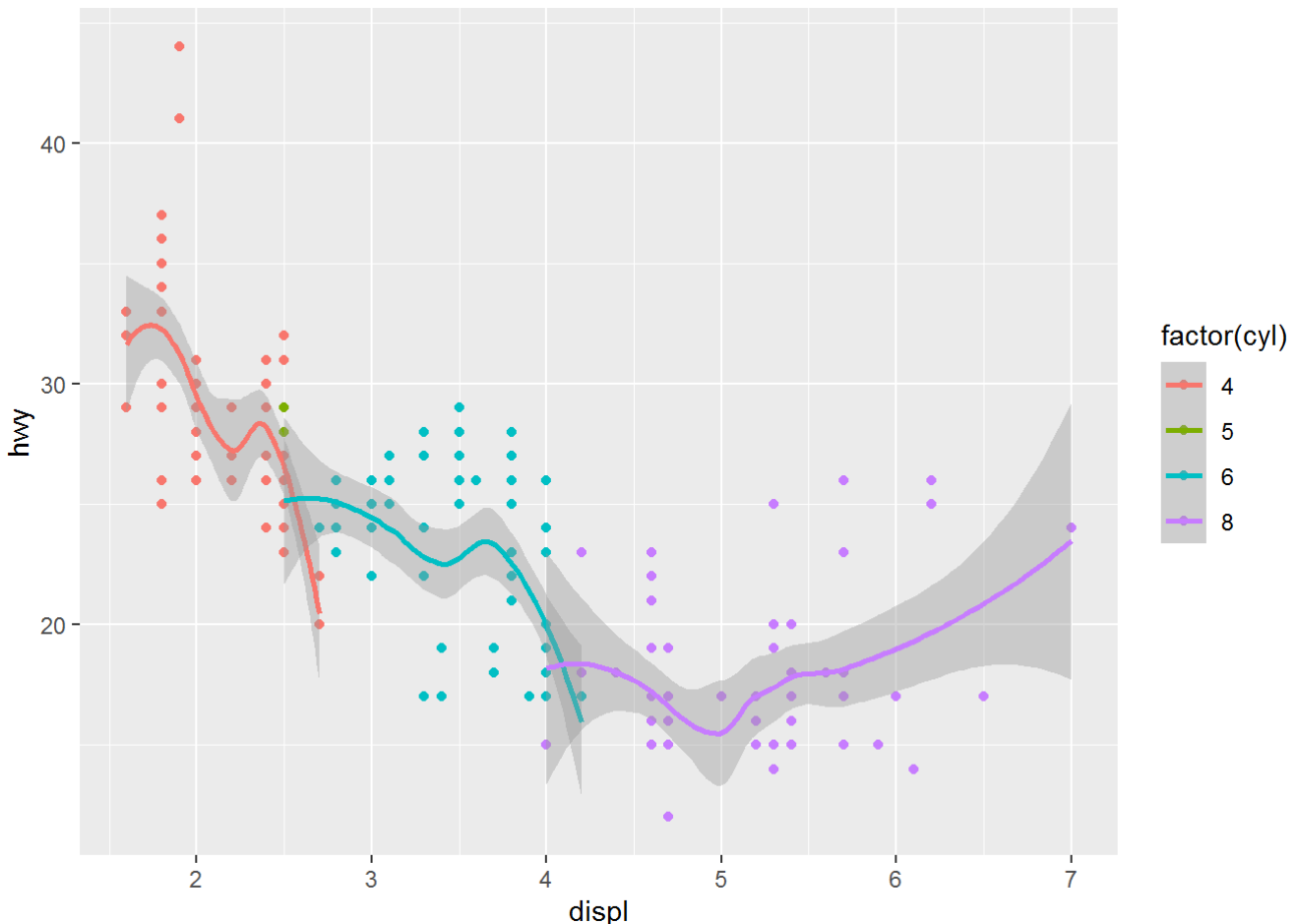
ggplot2是用于绘图的R语言扩展包，其理念根植于《Grammar of Graphics》一书。它将绘图视为一种映射，即从数学空间映射到图形元素空间。例如将不同的数值映射到不同的色彩或透明度。该绘图包的特点在于并不去定义具体的图形(如直方图，散点图)，而是定义各种底层组件(如线条、方块)来合成复杂的图形，这使它能以非常简洁的函数构建各类图形，而且默认条件下的绘图品质就能达到出版要求。

```
if(!require("ggplot2")){install.packages("ggplot2");library("ggplot2")}
```

```
## Loading required package: ggplot2
```

```
p <- ggplot(data=mpg, aes(x=displ, y=hwy, colour=factor(cyl)))  
p + geom_point() + geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



用汽车测试数据集mpg。三个变量分别是发动机容量displ,高速公路上每加仑行驶里数hwy,气缸数目cyl. 绘制带有平滑曲线的散点图。

```
# library("ggplot2")
# #install.packages("quantmod");
# library("quantmod")
# getSymbols('~SSEC', src='yahoo', from = '1997-01-01')
# close <- CI(SSEC)
# time <- index(close)
# value <- as.vector(close)
# p <- ggplot(data.frame(time, value), aes(time, value))
# p + geom_line()
#首次运行没有问题，安装rCharts后有问题了
```

上证指数时间序列图。

rCharts包

交互包，直接在R中生成基于D3的web界面。

以鸢尾花数据集为例，通过name函数对列名进行重新赋值，去掉单词间的点，用rPlot函数绘制散点图，利用颜色进行分组。

这个包的安装很费周折！

终极解决办法：本地安装

```
devtools::install_local("idmap2.zip")
```

上述方法我没有试，下下来的zip文件有6兆多，如果仅仅单独安装此包，担心有其他依赖包没有安装。

在CRAN中的包可以用install.packages()安装。但rCharts包只能从github上安装。

install.packages()和install_github()函数所需的参数是不同的。install.packages()只需包名即可，而install_github()需要提供Github的仓库名。这意味着当你想从Github上安装一个R包的时候你必须准确的记得它的仓库名称。

麻烦的是，我们往往很难记住Github的用户名。开发者在给R包命名的时候一般会考虑包的名称，以使用户方便记忆。然而，他们在给自己取一个Github用户名的时候可能就没那么小心了。例如，ggfortify是Github上的一个非常好的包。但它是由谁开发的呢？用户名是什么呢？答案是sinhrks4。我们似乎很难记住它。

githubinstall包提供了一种类似于install.packages()的方式，只需包名即可安装R包。

```
#install.packages('githubinstall') #已发布至CRAN
library(githubinstall)
githubinstall('AnomalyDetection')
# 这个方法我没试。
```

网上还有一种安装非CRAN上包的方法：

```
install_url("...tar.gz")
install.packages("D:/mvpart_1.6-2.tar.gz", repos = NULL, type = "source") #从本地安装
```

在网上搜索，发现可以通过本地文件来安装（install_github无法下载 R 包的变通方法），于是先将URL地址文件下载到本地，由于下载下来的是.tar.gz的压缩包，因此现将压缩包解压到D:Files-3.5.1，文件结构如下所示 然后执行下面命令进行安装 安装过程中如有如下提示，则先将缺少的RCurl包和RJSONIO包安装好后再执行上面的命令进行安装 install.packages("D:Files-3.5.1",repos = NULL, type = "source") 安装完成后，执行下面命令不报错，则安装成功。

安装好之后我们就可以开始愉快的安装rCharts包了

```
devtools::install_github("ramnathv/rCharts")
```

PS:ramnathv代表这个包的作者，rCharts代表这个包的名称

devtools::install_github表示运用devtools下的install_github函数

#R4.0版本试了好多次后，安装有提示了：

```
> devtools::install_github("ramnathv/rCharts")
Downloading GitHub repo ramnathv/rCharts@master
Installing 4 packages: plyr, RCurl, RJSONIO, whisker
trying URL 'http://mirrors.ustc.edu.cn/CRAN/bin/windows/contrib/4.0/plyr_1.8.6.zip'
Content type 'application/zip' length 1494803 bytes (1.4 MB)
downloaded 1.4 MB

trying URL 'http://mirrors.ustc.edu.cn/CRAN/bin/windows/contrib/4.0/RCurl_1.98-1.2.zip'
Content type 'application/zip' length 3043515 bytes (2.9 MB)
downloaded 2.9 MB
trying URL 'http://mirrors.ustc.edu.cn/CRAN/bin/windows/contrib/4.0/RJSONIO_1.3-1.4.zip'
Content type 'application/zip' length 1345303 bytes (1.3 MB)
downloaded 1.3 MB

trying URL 'http://mirrors.ustc.edu.cn/CRAN/bin/windows/contrib/4.0/whisker_0.4.zip'
Content type 'application/zip' length 83128 bytes (81 KB)
downloaded 81 KB

package 'plyr' successfully unpacked and MD5 sums checked
package 'RCurl' successfully unpacked and MD5 sums checked
package 'RJSONIO' successfully unpacked and MD5 sums checked
package 'whisker' successfully unpacked and MD5 sums checked
```

The downloaded binary packages are in

```
  C:\Users\Administrator\AppData\Local\Temp\RtmpyaFePV\downloaded_packages
✓ checking for file 'C:\Users\Administrator\AppData\Local\Temp\RtmpyaFePV\remotes27a46a4e718
\ramnathv-rCharts-479a4f9/DESCRIPTION' (2s)
- preparing 'rCharts': (2s)
✓ checking DESCRIPTION meta-information
- checking for LF line-endings in source and make files and shell scripts (352ms)
- checking for empty or unneeded directories
- building 'rCharts_0.4.5.tar.gz'
```

* installing *source* package 'rCharts' ...

** using staged installation

** R

** demo

** inst

** byte-compile and prepare package for lazy loading

** help

*** installing help indices

converting help for package 'rCharts'

finding HTML links ... 好了

create_gist	html
get_assets	html
get_token	html
post_gist	html
rPlot	html
renderChart	html
showOutput	html
toChain	html
toJSONArray	html

** building package indices

** testing if installed package can be loaded from temporary location

*** arch - i386

*** arch - x64

```
** testing if installed package can be loaded from final location
*** arch - i386
*** arch - x64
** testing if installed package keeps a record of temporary installation path
* DONE (rCharts)
Making 'packages.html' ... 好了
```

```

#家里电脑R3.6.3安装好多次，升级了Rstudio到最新版，试了好几次：
devtools::install_github("ramnathv/rCharts")
Downloading GitHub repo ramnathv/rCharts@master
Error in utils::download.file(url, path, method = method, quiet = quiet, :
  download from 'https://api.github.com/repos/ramnathv/rCharts/tarball/master' failed
# 按网上的说法，设置了一下浏览器的安全选项，重启Rstudio
devtools::install_github("ramnathv/rCharts")
Downloading GitHub repo ramnathv/rCharts@master
These packages have more recent versions available.
It is recommended to update all of them.
Which would you like to update?

1: All
2: CRAN packages only
3: None
4: RCurl (1.98-1.1 -> 1.98-1.2) [CRAN]
5: Rcpp (1.0.4 -> 1.0.4.6) [CRAN]

Enter one or more numbers, or an empty line to skip updates:
1
Installing 3 packages: RCurl, RJSONIO, Rcpp
试开URL' https://mirrors.ustc.edu.cn/CRAN/bin/windows/contrib/3.6/RCurl_1.98-1.2.zip'
length 2987986 bytes (2.8 MB)
downloaded 2.8 MB

试开URL' https://mirrors.ustc.edu.cn/CRAN/bin/windows/contrib/3.6/RJSONIO_1.3-1.4.zip'
length 1532891 bytes (1.5 MB)
downloaded 1.5 MB

试开URL' https://mirrors.ustc.edu.cn/CRAN/bin/windows/contrib/3.6/Rcpp_1.0.4.6.zip'
length 3031047 bytes (2.9 MB)
package 'RCurl' successfully unpacked and MD5 sums checked
package 'RJSONIO' successfully unpacked and MD5 sums checked
package 'Rcpp' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\Administrator\AppData\Local\Temp\RtmpoxH9BN\downloaded_packages
√ checking for file 'C:\Users\Administrator\AppData\Local\Temp\RtmpoxH9BN\remotes2a8c1aa03c81\ramnathv-rCharts-479a4f9\DESCRIPTION' (1.2s)
- preparing 'rCharts': (2.3s)
√ checking DESCRIPTION meta-information ...
- checking for LF line-endings in source and make files and shell scripts
- checking for empty or unneeded directories
- building 'rCharts_0.4.5.tar.gz'

* installing *source* package 'rCharts' ...
** using staged installation
** R
** demo
** inst
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
converting help for package 'rCharts'
  finding HTML links ... 好了
  create_gist                                html
  get_assets                                html

```

```

get_token          html
post_gist          html
rPlot              html
renderChart        html
showOutput         html
toChain            html
toJSONArray        html
** building package indices
** testing if installed package can be loaded from temporary location
*** arch - i386
*** arch - x64
** testing if installed package can be loaded from final location
*** arch - i386
*** arch - x64
** testing if installed package keeps a record of temporary installation path
* DONE (rCharts)

```

```

# library("devtools")
# install_github('rCharts', 'ramnathv')
library("rCharts")
library("ggplot2")
# names(iris) = gsub("\\.", "", names(iris))
# p1 <- rPlot(SepalLength ~ SepalWidth | Species, data = iris, color = 'Species', type = 'point')
# p1
# 这段代码有问题: Error in eval(modelRHS.vars[[i]], data, env) : 找不到对象'Species'

```

rCharts支持多个JavaScript图表库，每个都有自己的长处。NVD3是一个旨在建立可复用的图表和组件的d3.js项目，rCharts包提供了nPlot函数来实现NVD3功能。以眼睛和头发的颜色数据为例，说明nPlot绘图的基本原理。按眼睛的颜色进行分组，对头发颜色人数绘制柱状图，并将类型设置为柱状图组合方式，实现分组和叠加效果。

```

# hair_eye_male <- subset(as.data.frame(HairEyeColor), Sex == "Male")
# hair_eye_male[,1] <- paste0("Hair", hair_eye_male[,1])
# hair_eye_male[,2] <- paste0("Eye", hair_eye_male[,2])
# n1 <- nPlot(Freq ~ Hair, group = "Eye", data = hair_eye_male,
#             type = "multiBarChart")
# n1
# 单独运行没有问题

```

rCharts包提供了HPlot函数来实现Highcharts。MASS包中的学生调查数据集survey为例，绘制学生身高和每分钟的脉搏跳动次数的气泡图，以年龄变量作为调整气泡大小的变量。

```

# a <- hPlot(Pulse ~ Height, data = MASS::survey, type = "bubble",
#            title = "Zoom demo", subtitle = "bubble chart",
#            size = "Age", group = "Exer")
# a$colors('rgba(223, 83, 83, .5)', 'rgba(119, 152, 191, .5)',
#           'rgba(60, 179, 113, .5)')
# a$chart(zoomType = "xy")
# a$exporting(enabled = T)
# a
# 单独运行可以

```

plotly功能包

plotly建立在开源的JavaScript图表库plotly.js之上，是一个基于浏览器的交互式图表功能库。官网上提供了Python，R，MATLAB，Excel等接口的使用方法。主要利用plot_ly函数绘制交互表。以鸢尾花数据集绘制散点图。

```
# if(!require("plotly")){install.packages("plotly");library("plotly")}#安装方式1
# if(!require("devtools")){install.packages("devtools");library("devtools")}#安装方式2
# devtools::install_github("ropensci/plotly")#安装方式2
library("plotly")
```

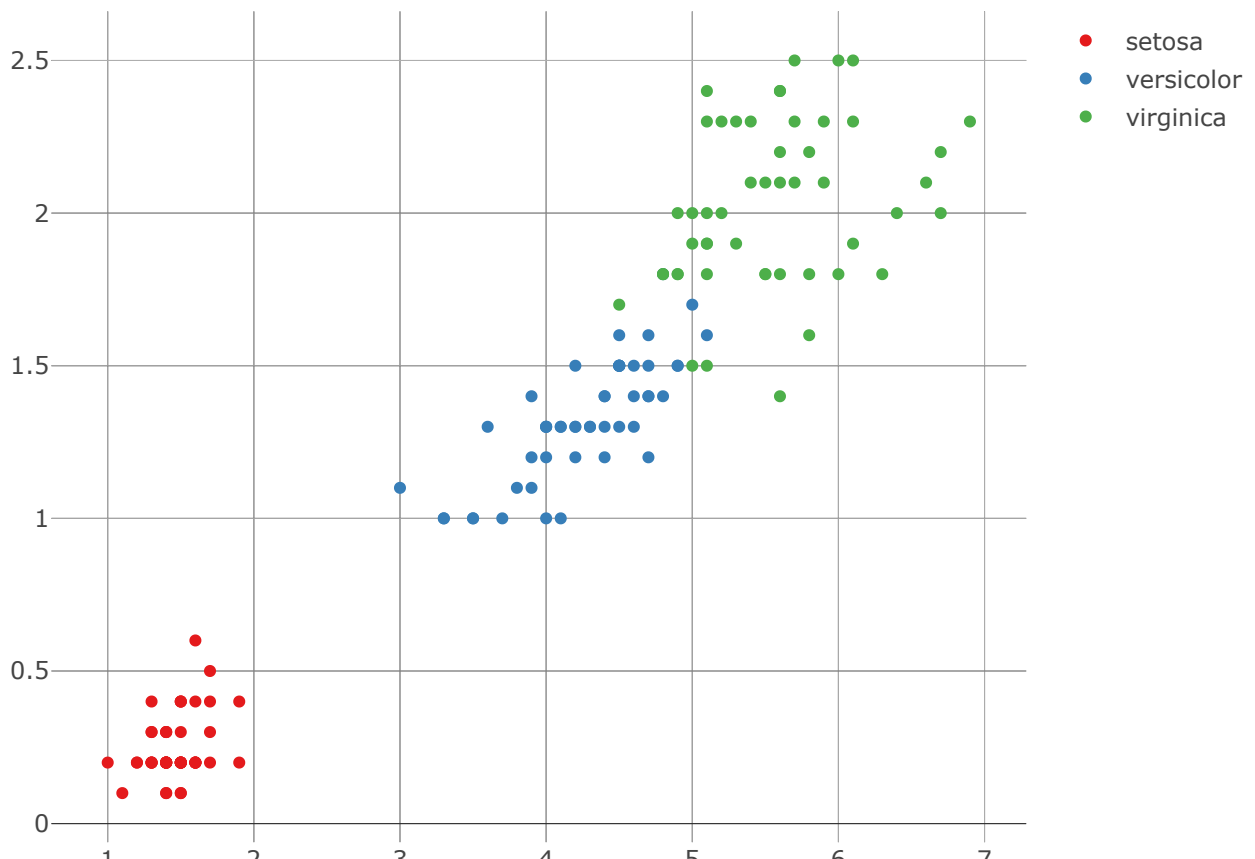
```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##   last_plot
```

```
## The following object is masked from 'package:stats':
##
##   filter
```

```
## The following object is masked from 'package:graphics':
##
##   layout
```

```
attach(iris)
p<- plot_ly(iris, x = Petal.Length, y = Petal.Width,
            color = Species, colors = "Set1", type = "scatter",mode="markers")
p
```



画地图 要在R 语言中绘制地图，除了要安装map包，还要注意以下三个步骤：（1）的确需要绘制地图；（2）具备绘制地图的地理信息，如经纬度、边界等；（3）利用（2）的数据在 R 中画出来。这个包中预存了世界地图和美国地图的相关数据，所以，几行代码便可以画出世界地图。

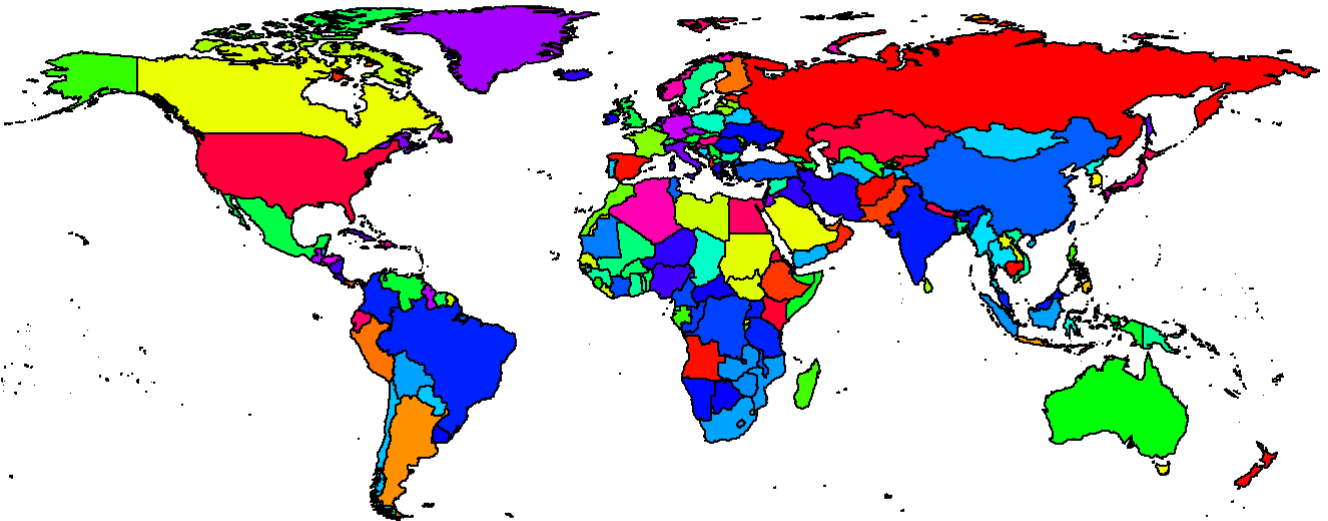
```
if(!require("maps")){install.packages("maps");
library("maps")}
```

```
## Loading required package: maps
```

```
## Warning: package 'maps' was built under R version 4.0.0
```

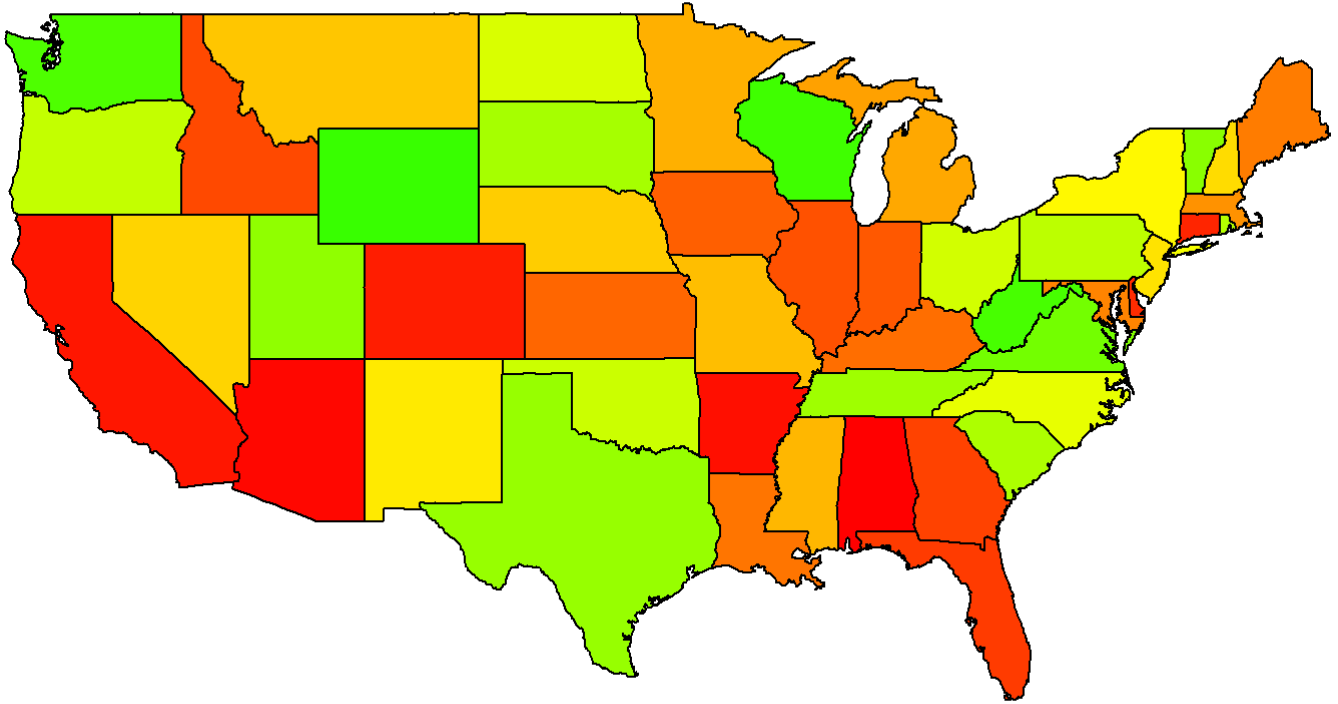
```
map("world", fill = TRUE, col = rainbow(200),
    ylim = c(-60, 90), mar = c(0, 0, 0, 0))
title("世界地图")
```

世界地图



```
library("maps")
map("state", fill = TRUE, col = rainbow(209),
    mar = c(0, 0, 2, 0))
title("美国地图")
```

美国地图



R提供了许多地图功能包，不同的包中存储着不同的常见地图的数据，比如maps包中存有世界地图、美国地图、美国各州郡地图、法国地图以及加拿大城市地图等，加载了这个包，就可以轻松愉快地绘制上述地图。mapdata包中存有中国地图的数据(比较旧了，连重庆从四川分出来都没有更新)，显然这种方法受到已有数据的限制；

先去专业机构下载相应的地理数据，然后读入R进行绘制。例如mapdata中的中国地图比较久远了，但我们可以先从国家基础地理信息中心下载中国各省市的地理数据，之后再绘制。请课外关注googleVis包和OpenStreetMap包，都提供了方便下载地理数据的途径。但是这种方法步骤繁杂，不利于上手。

结合了前两种方法的有点。既然R是自由的，显然我们可以直接去调取专业的地图企业或者网站的数据呢，这样就不会受包中数据集所限。只需要有一个途径去Google Map，Baidu Map等专业的地图供应商那取数据即可。自由的R只需要连接Google Map的API，一切就都有了，但是Google对数据请求限制为每天2000次(负荷考虑)；在这种情况下ggmap包(实验环境可能需要VPN，供课外学习参考)横空出世了，感谢两位作者David Kahle和Hadley Wickham的努力。

参考文献

王小峰等，《新媒体数据挖掘，基于R语言》，清华大学出版社，2017年10月。

谢佳标，《R语言游戏数据分析与挖掘》，机械工业出版社，2017年7月