统计计算和绘图

基于 R 语言

黄湘云

理学院

中国矿业大学 (北京)

2016 年 6 月 7 日

目录

安装 R 包

从指定网站安装 R 包

```
>install.packages("ggplot2",repos = "https://mirrors.tuna.tsinghua.edu.cn/CRAN/")
>install.packages("pim",repos = "http://www.r-forge.r-project.org/",
+                  lib = "D:/library/")
```

一行命令安装 cran 上所有的 Packages

```
>install.packages(setdiff(available.packages()[,1],
+                         .packages(all.available = TRUE)))
```

安装 Bioconductor 上的 R 包

```
>source("https://bioconductor.org/biocLite.R")
>library(BiocInstaller)
>biocLite("ggtree")
```

安装 Github 上的 R 包

```
>library(devtools)
>install_github("ropensci/aRxiv")
```

安装注意几项

依赖问题 主要针对离线安装和源码编译安装

平台问题 有些 R 包只能装在类 unix 操作系统上

软件问题 有些 R 包要求相应 R 软件版本

特定库依赖 如 gputools 包依赖 cuda 库

```
>Sys.info()
```

```
      sysname       release       version      nodename
    "Windows"       "7 x64"  "build 9200"    "LENOVO-PC"
      machine         login          user effective_user
     "x86-64" "Xiangyun Huang" "Xiangyun Huang" "Xiangyun Huang"
```

R 软件信息

```
>sessionInfo()

R version 3.1.3 (2015-03-09)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 8 x64 (build 9200)

locale:
[1] LC_COLLATE=Chinese (Simplified)_China.936
[2] LC_CTYPE=Chinese (Simplified)_China.936
[3] LC_MONETARY=Chinese (Simplified)_China.936
[4] LC_NUMERIC=C
[5] LC_TIME=Chinese (Simplified)_China.936

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] knitr_1.13

loaded via a namespace (and not attached):
[1] evaluate_0.9    formatR_1.3     highr_0.5.1     magrittr_1.5
[5] RevoUtils_7.4.0 stringi_1.0-1   stringr_1.0.0   tools_3.1.3
```

R 软件系统自带的数据集

```
>
>
>
```

```
>
>
>
>
```

```
  carat       cut color clarity depth table price    x    y    z
1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
```

谈谈对象

```
>class(diamonds) #Object Classes

[1] "tbl_df"     "tbl"        "data.frame"

>str(diamonds) #Compactly Display the Structure of an Arbitrary R Object

Classes 'tbl_df', 'tbl' and 'data.frame':^^I53940 obs. of  10 variables:
 $ carat   : num  0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 0.23 ...
 $ cut     : Ord.factor w/ 5 levels "Fair"<"Good"<..: 5 4 2 4 2 3 3 3 1 3 ...
 $ color   : Ord.factor w/ 7 levels "D"<"E"<"F"<"G"<..: 2 2 2 6 7 7 6 5 2 5 ...
 $ clarity: Ord.factor w/ 8 levels "I1"<"SI2"<"SI1"<..: 2 3 5 4 2 6 7 3 4 5 ...
 $ depth   : num  61.5 59.8 56.9 62.4 63.3 62.8 62.3 61.9 65.1 59.4 ...
 $ table   : num  55 61 65 58 58 57 57 55 61 61 ...
 $ price   : int  326 326 327 334 335 336 336 337 337 338 ...
 $ x       : num  3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ...
 $ y       : num  3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 ...
 $ z       : num  2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39 ...

>#  mode(diamonds) #The (Storage) Mode of an Object
>#  typeof(diamonds) #The Type of an Object
>#  attributes(diamonds) #Object Attribute Lists
```

pryr 包撬开 R 的底层

- pryr: Useful tools to pry back the covers of R and understand the language at a deeper level.

```
>c(is.list(diamonds),is.matrix(diamonds))

[1]  TRUE FALSE

>c(is.vector(diamonds),is.numeric(diamonds))

[1] FALSE FALSE

>c(is.array(diamonds),is.data.frame(diamonds))

[1] FALSE  TRUE
```

撬开 tabulate 函数

```
>tabulate

function (bin, nbins = max(1L, bin, na.rm = TRUE))
{
    if (!is.numeric(bin) && !is.factor(bin))
        stop("'bin' must be numeric or a factor")
    if (typeof(bin) != "integer")
        bin <- as.integer(bin)
    if (nbins > .Machine$integer.max)
        stop("attempt to make a table with >= 2^31 elements")
    nbins <- as.integer(nbins)
    if (is.na(nbins))
        stop("invalid value of 'nbins'")
    .Internal(tabulate(bin, nbins))
}
<bytecode: 0x00000000184fddc0>
<environment: namespace:base>
```

撬开 tabulate 函数

```
library(pryr)
show_c_source(.Internal(tabulate(bin,nbins)))
SEXP attribute_hidden do_tabulate(SEXP call, SEXP op, SEXP args, SEXP rho)
{
    checkArity(op, args);
    SEXP in = CAR(args), nbin = CADR(args);
    if (TYPEOF(in) != INTSXP)  error("invalid input");
    R_xlen_t n = XLENGTH(in);
    /* FIXME: could in principle be a long vector */
    int nb = asInteger(nbin);
    if (nb == NA_INTEGER || nb < 0)
      error(_("invalid '%s' argument"), "nbin");
    SEXP ans = allocVector(INTSXP, nb);
    int *x = INTEGER(in), *y = INTEGER(ans);
    if (nb) memset(y, 0, nb * sizeof(int));
    for(R_xlen_t i = 0 ; i < n ; i++)
      if (x[i] != NA_INTEGER && x[i] > 0 && x[i] <= nb) y[x[i] - 1]++;
    return ans;
}
```

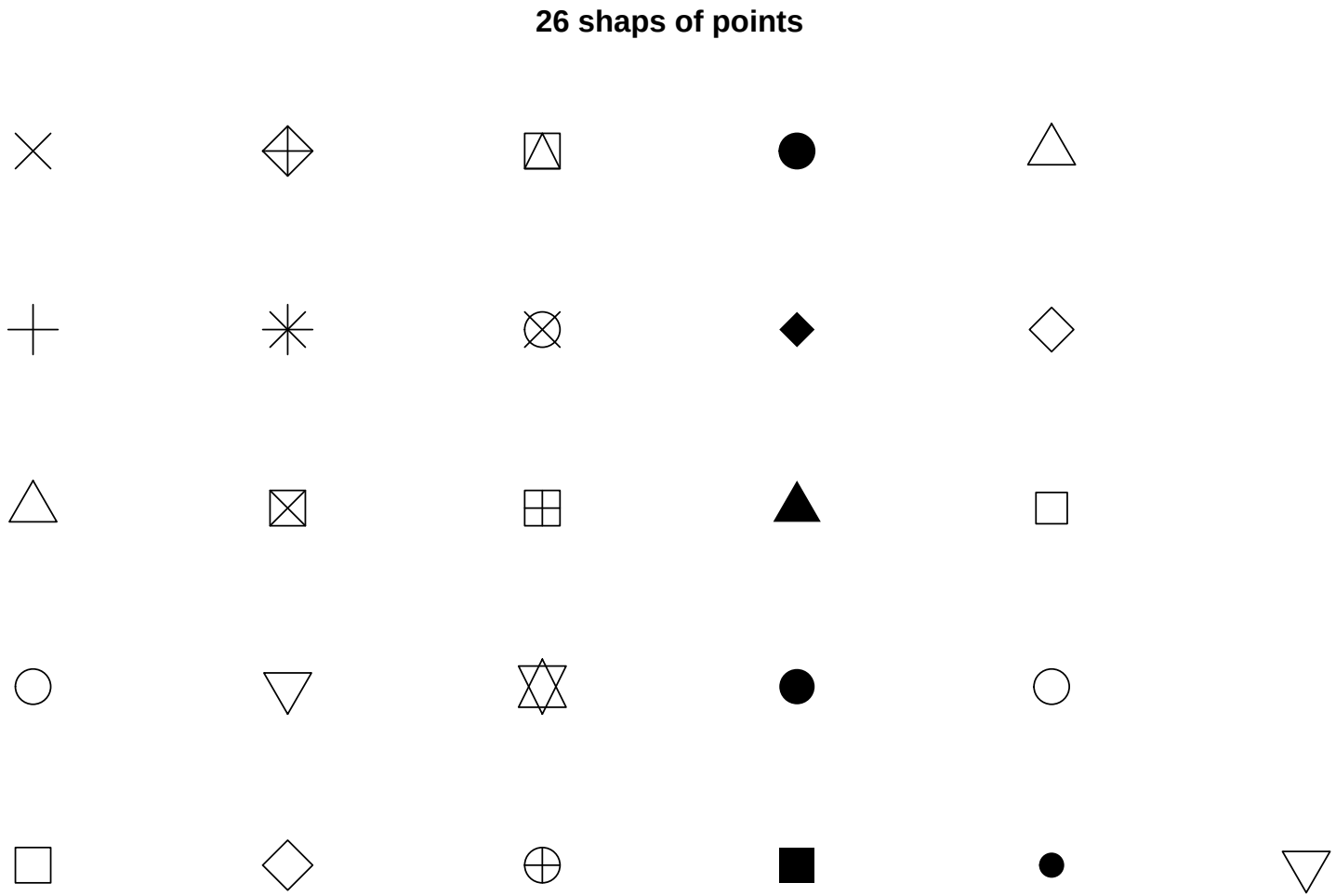
撬开 plot

```
>methods(plot)

 [1] plot.acf*           plot.data.frame*    plot.decomposed.ts*
 [4] plot.default        plot.dendrogram*    plot.density*
 [7] plot.ecdf           plot.factor*        plot.formula*
[10] plot.function       plot.ggplot*        plot.gtable*
[13] plot.hclust*        plot.histogram*     plot.HoltWinters*
[16] plot.isoreg*        plot.lm*            plot.medpolish*
[19] plot.mlm*           plot.ppr*           plot.prcomp*
[22] plot.princomp*      plot.profile.nls*   plot.spec*
[25] plot.stepfun        plot.stl*           plot.table*
[28] plot.ts             plot.tskernel*      plot.TukeyHSD*

   Non-visible functions are asterisked
```

```
                  visible                         from
plot.acf            FALSE registered S3method for plot
plot.data.frame     FALSE registered S3method for plot
plot.decomposed.ts  FALSE registered S3method for plot
plot.default         TRUE             package:graphics
plot.dendrogram     FALSE registered S3method for plot
plot.density        FALSE registered S3method for plot
plot.ecdf            TRUE                 package:stats
plot.factor         FALSE registered S3method for plot
plot.formula        FALSE registered S3method for plot
plot.function        TRUE             package:graphics
plot.ggplot         FALSE registered S3method for plot
plot.gtable         FALSE registered S3method for plot
plot.hclust         FALSE registered S3method for plot
plot.histogram      FALSE registered S3method for plot
plot.HoltWinters    FALSE registered S3method for plot
plot.isoreg         FALSE registered S3method for plot
plot.lm             FALSE registered S3method for plot
plot.medpolish      FALSE registered S3method for plot
plot.mlm            FALSE registered S3method for plot
plot.ppr            FALSE registered S3method for plot
plot.prcomp         FALSE registered S3method for plot
plot.princomp       FALSE registered S3method for plot
plot.profile.nls    FALSE registered S3method for plot
plot.spec           FALSE registered S3method for plot
plot.stepfun         TRUE                 package:stats
plot.stl            FALSE registered S3method for plot
plot.table          FALSE registered S3method for plot
plot.ts              TRUE                 package:stats
plot.tskernel       FALSE registered S3method for plot
plot.TukeyHSD       FALSE registered S3method for plot
```

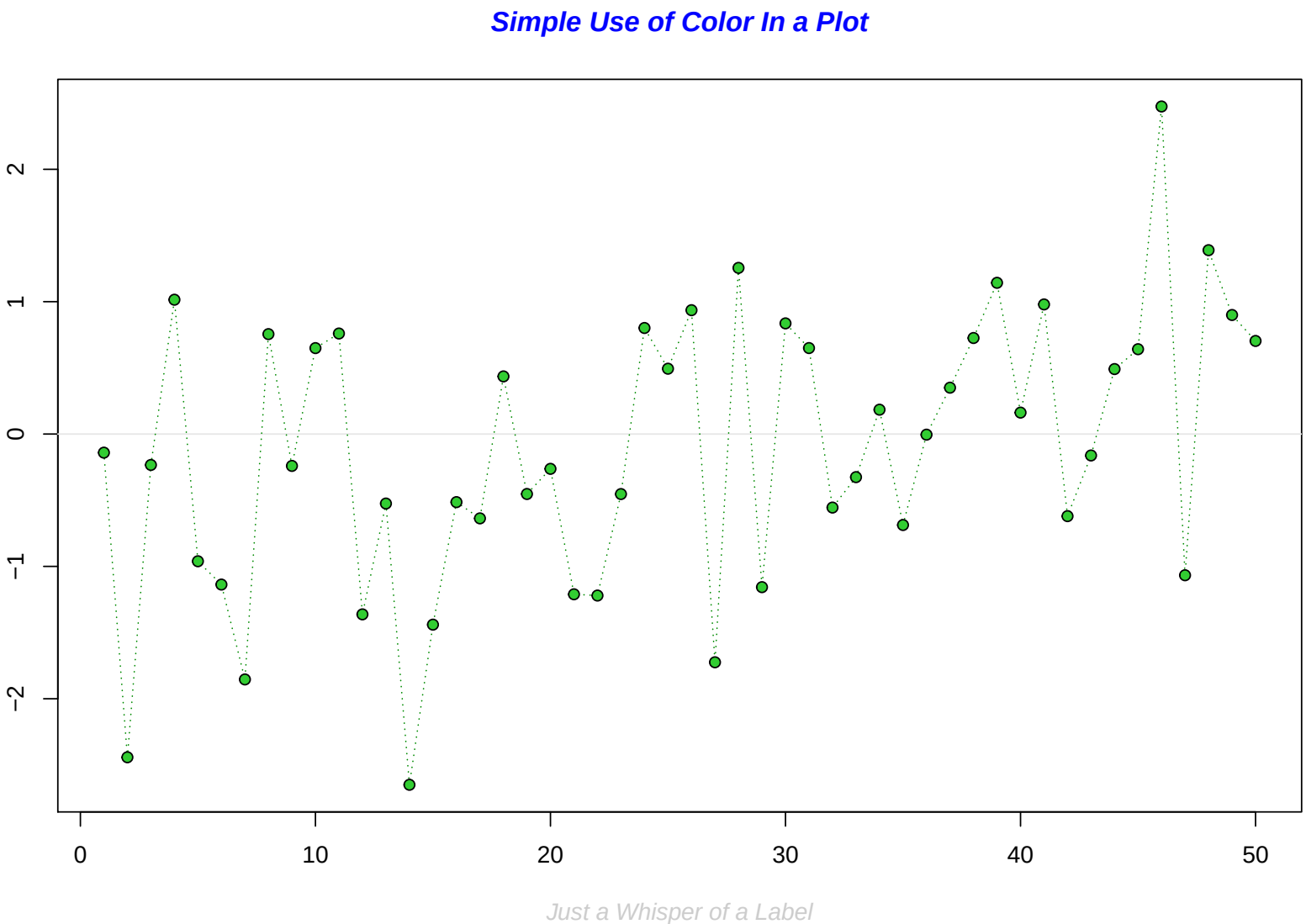
目录

plot 函数–pch
点

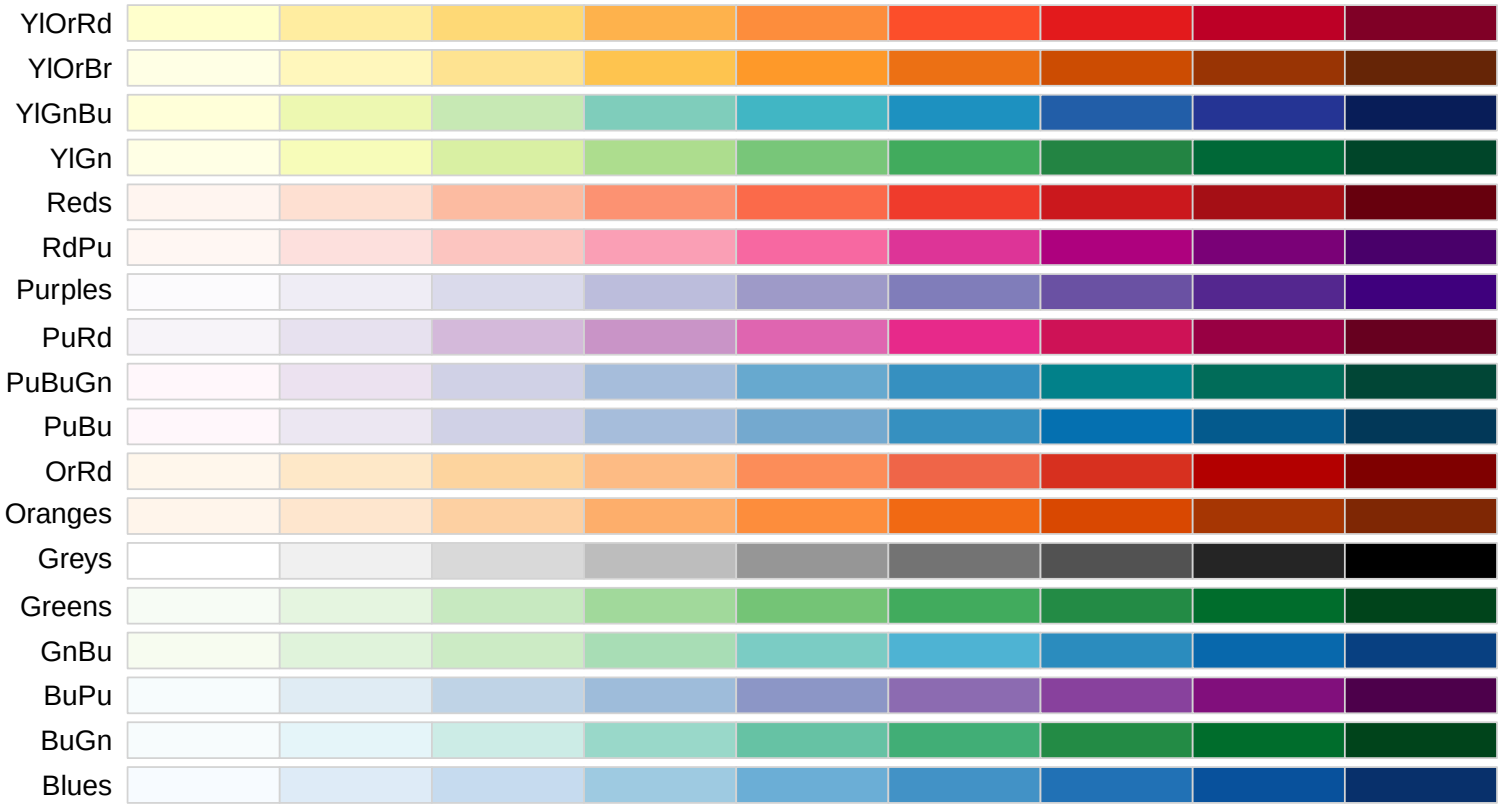


plot 函数–lty–lwd

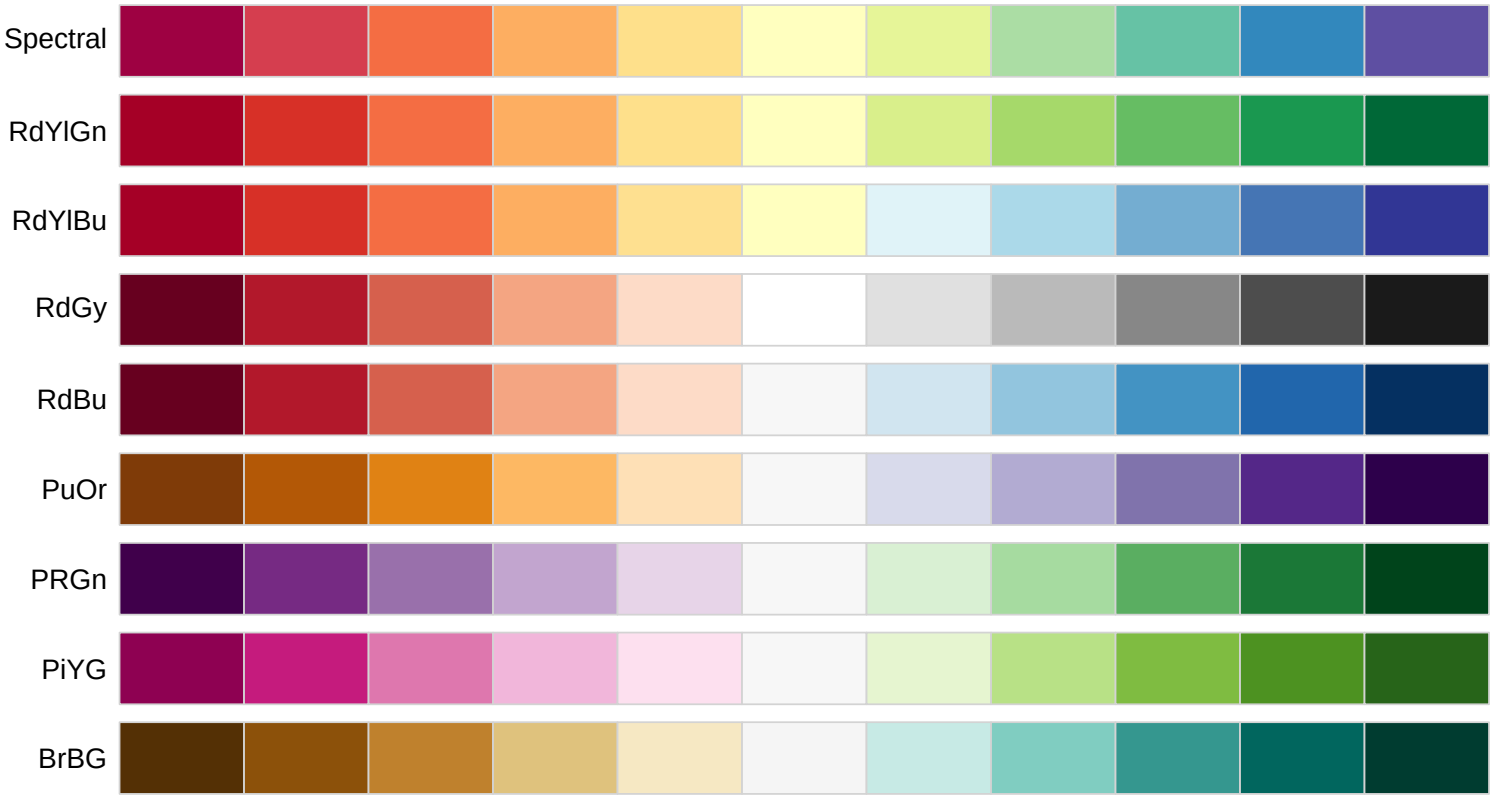
线



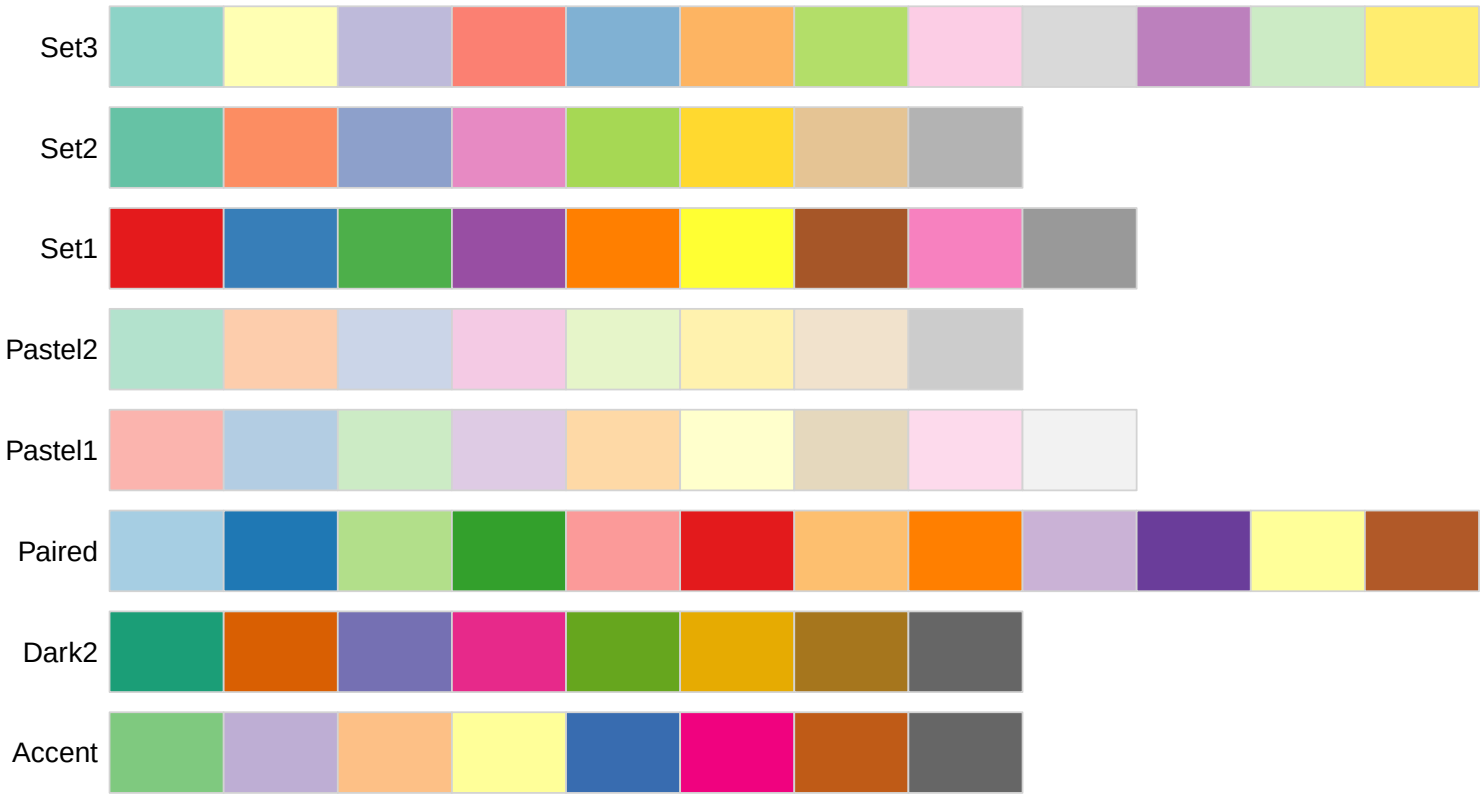
调色板–连续型


YlOrRd
YlOrBr
YlGnBu
YlGn
Reds
RdPu
Purples
PuRd
PuBuGn
PuBu
OrRd
Oranges
Greys
Greens
GnBu
BuPu
BuGn
Blues

调色板–极端型

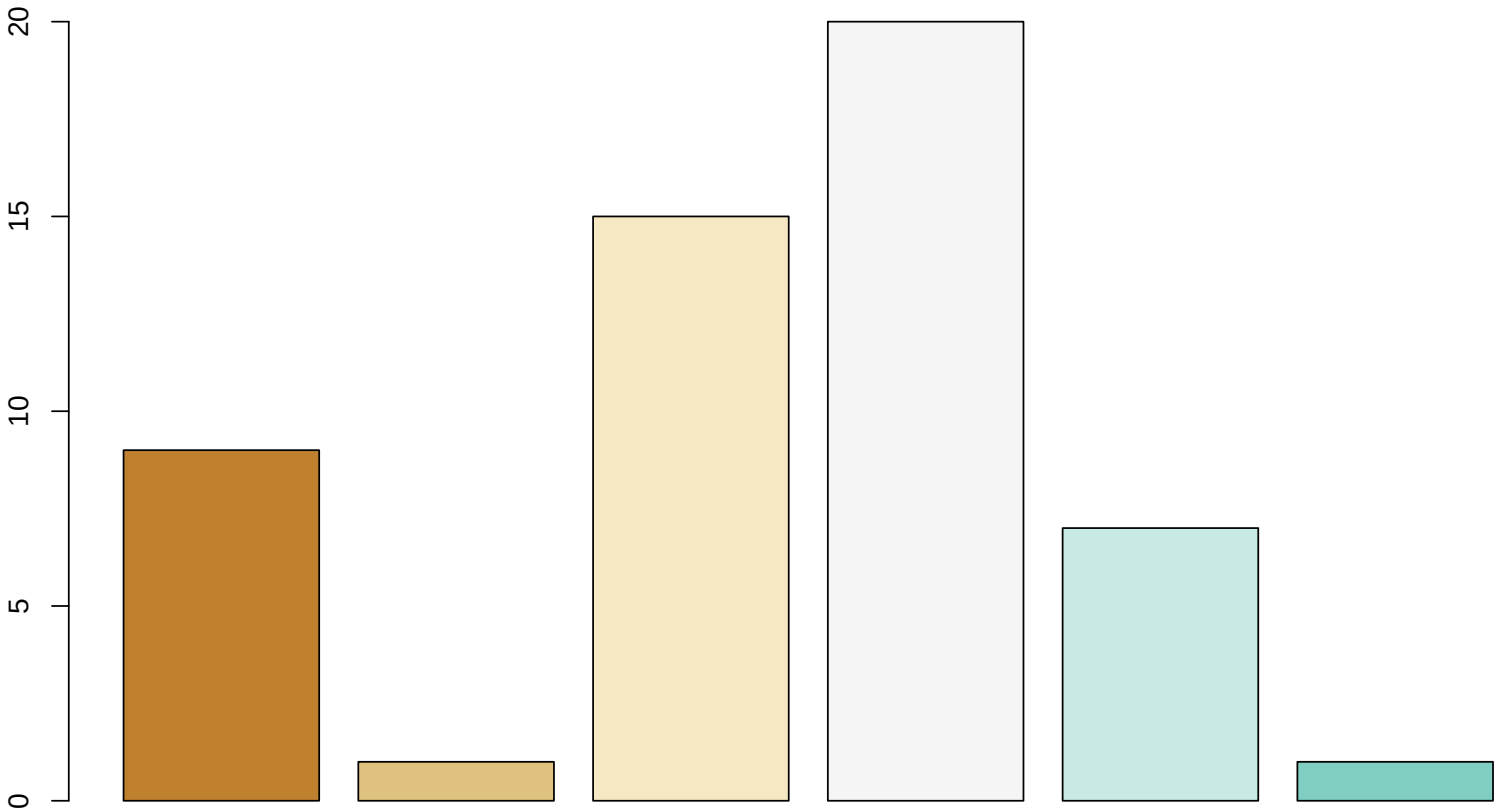


调色板–离散型

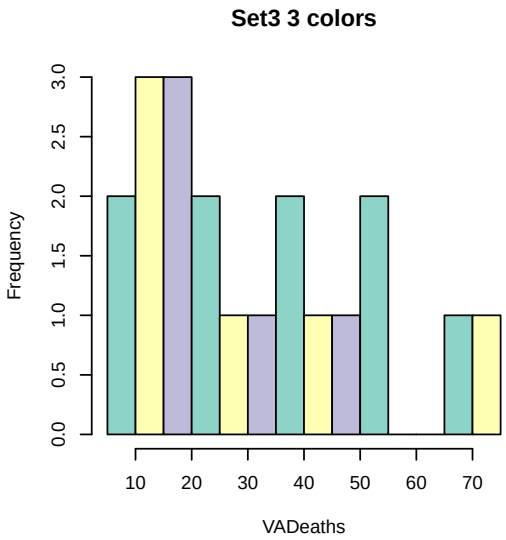


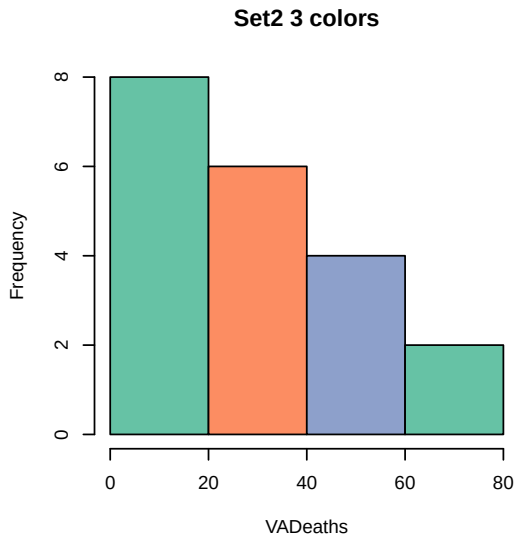
举个栗子

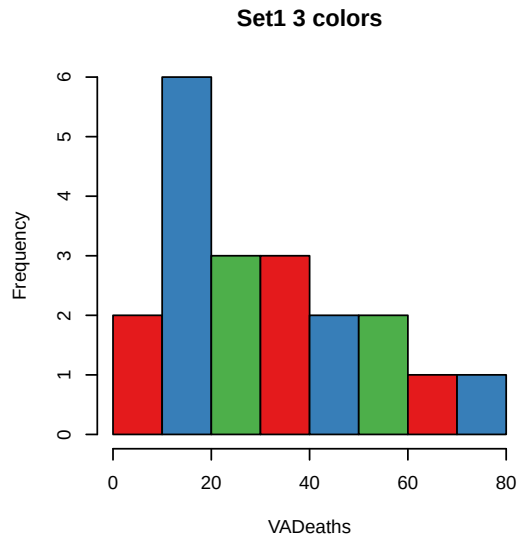
```
>barplot(sample(seq(20),6,replace = T),col=brewer.pal(11, "BrBG")[3:8])
```

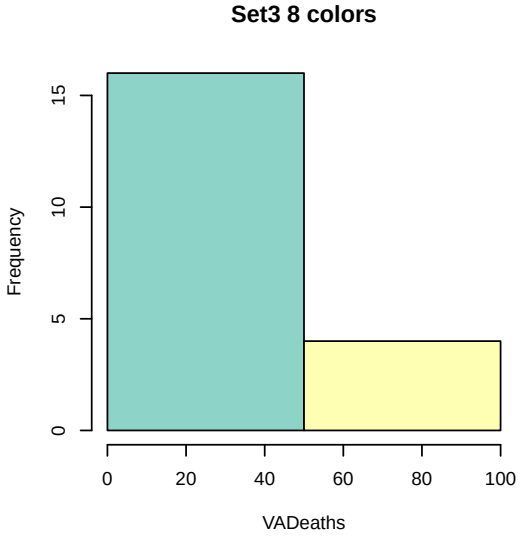


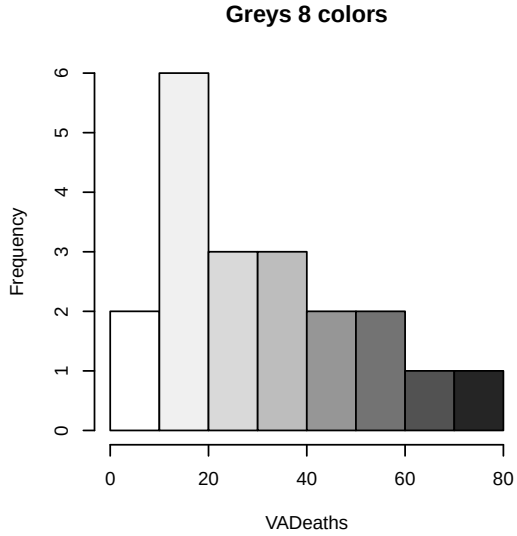
再举个栗子

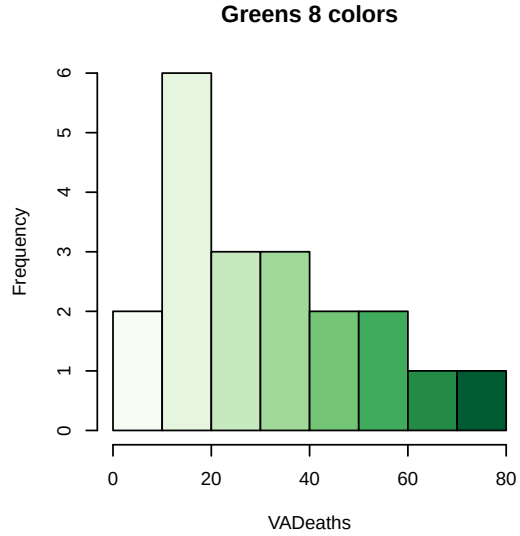

Set3 3 colors
Frequency
0.0
0.5
1.0
1.5
2.0
2.5
3.0
10
20
30
40
50
60
70
VADeaths


Set2 3 colors
Frequency
0
2
4
6
8
0
20
40
60
80
VADeaths

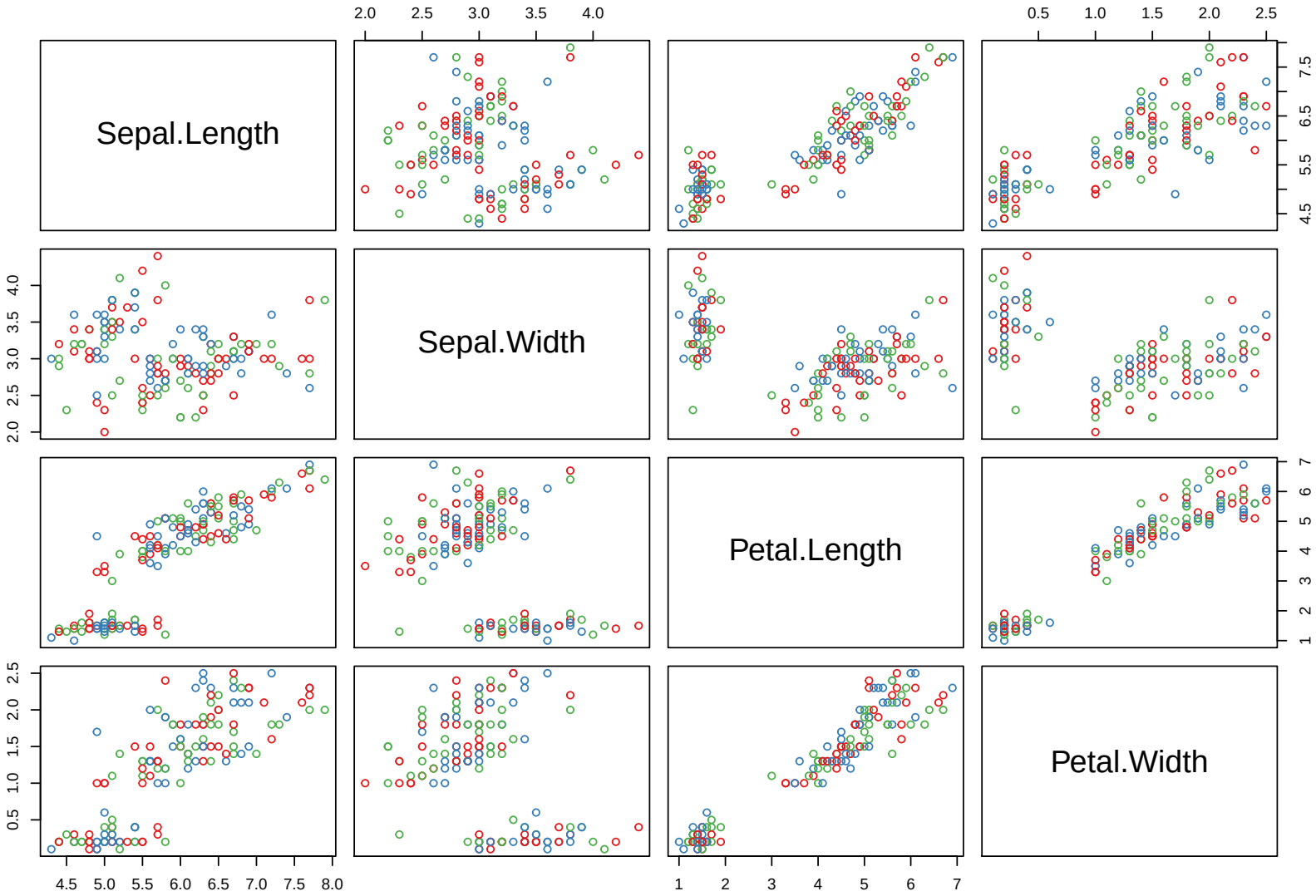

Set1 3 colors
Frequency
0
1
2
3
4
5
6
0
20
40
60
80
VADeaths


Set3 8 colors
Frequency
0
5
10
15
0
20
40
60
80
100
VADeaths

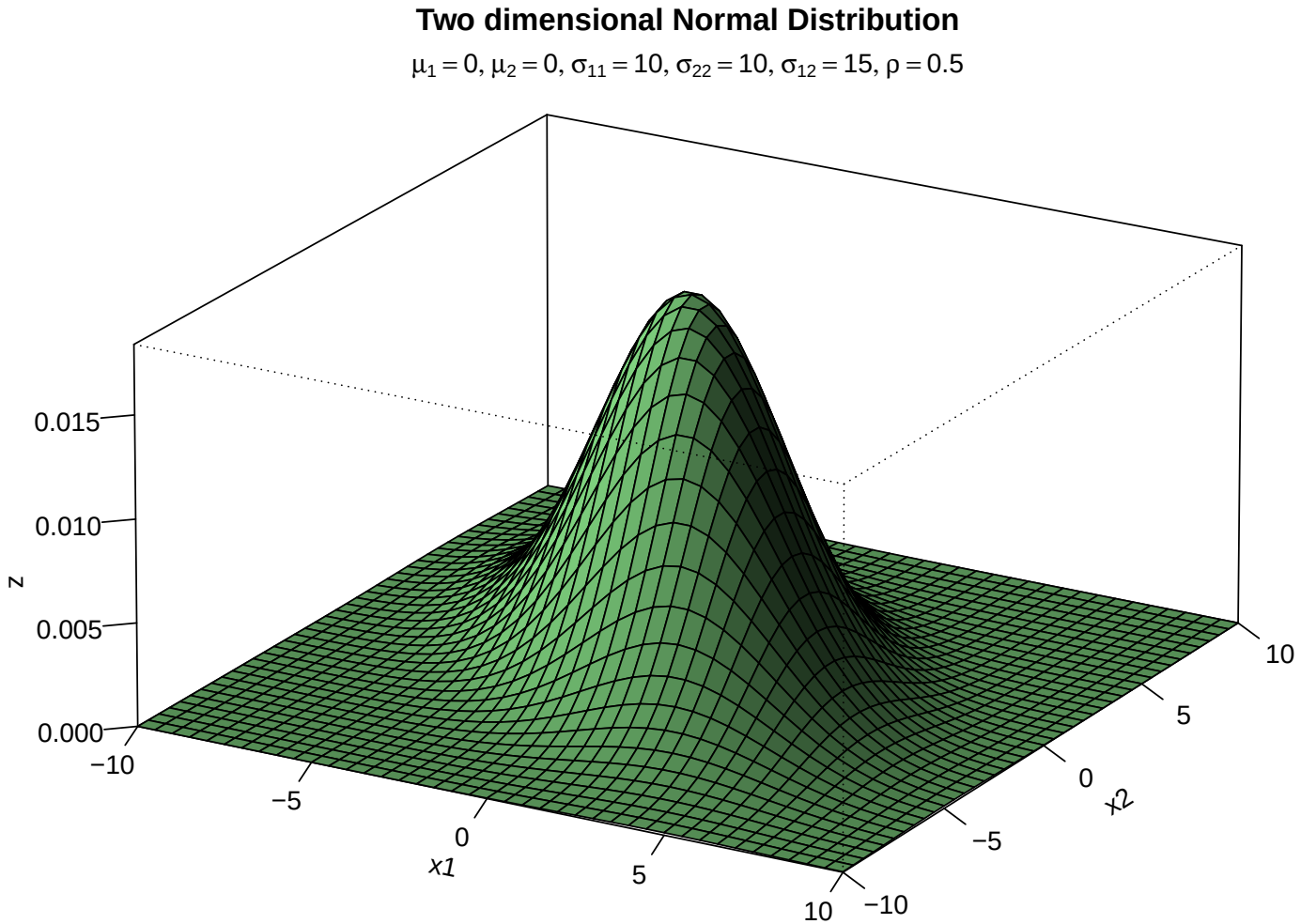

Greys 8 colors
Frequency
0
1
2
3
4
5
6
0
20
40
60
80
VADeaths


Greens 8 colors
Frequency
0
1
2
3
4
5
6
0
20
40
60
80
VADeaths

继续举个栗子

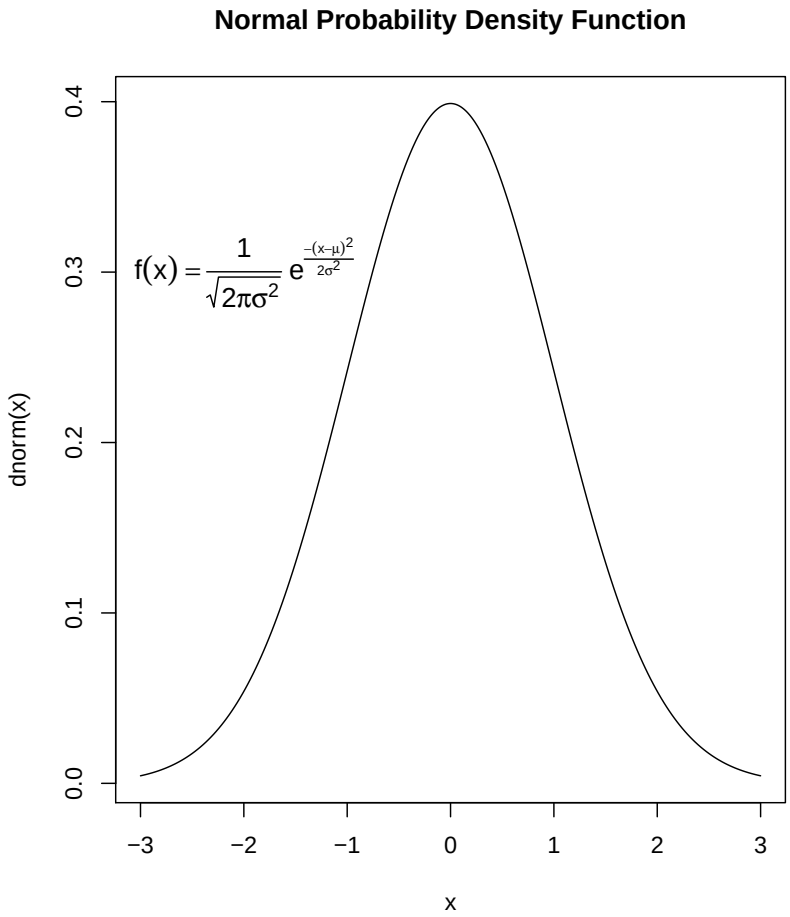


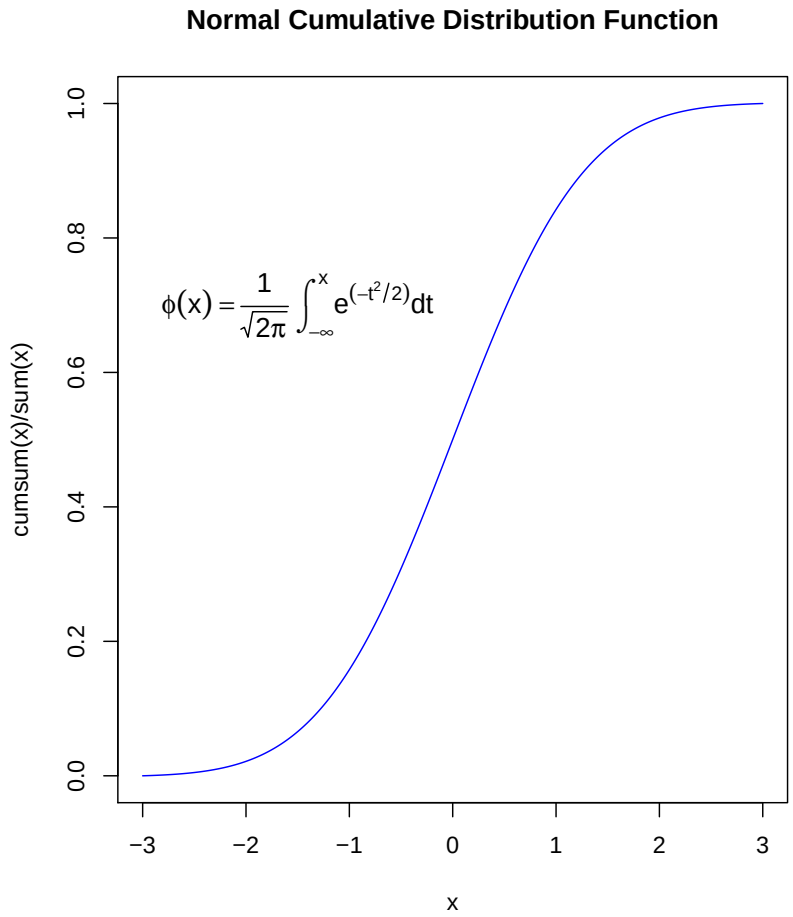
三维图–persp



$$f(\mathbf{x}) = \frac{1}{2\,\pi\sqrt{\sigma_{11}\,\sigma_{22}\,(1-\rho^2)}} \cdot \exp\left\{-\frac{1}{2(1-\rho^2)}\left[\frac{(x_1-\mu_1)^2}{\sigma_{11}} - 2\,\rho\,\frac{x_1-\mu_1}{\sqrt{\sigma_{11}}}\frac{x_2-\mu_2}{\sqrt{\sigma_{22}}} + \frac{(x_2-\mu_2)^2}{\sigma_{22}}\right]\right\}$$

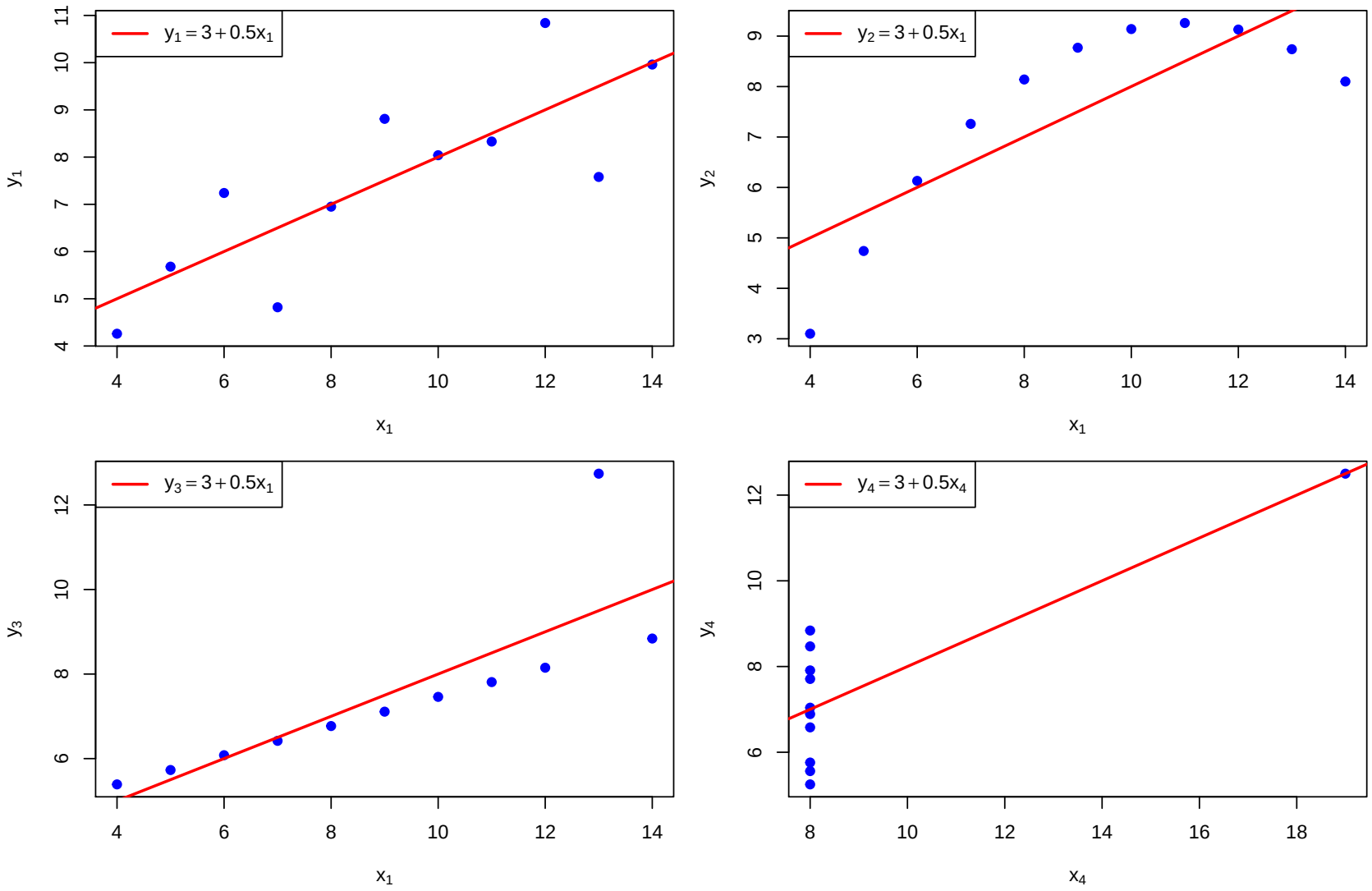
添加数学公式


Normal Probability Density Function
$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$
dnorm(x)
0.0
0.1
0.2
0.3
0.4
−3
−2
−1
0
1
2
3
x

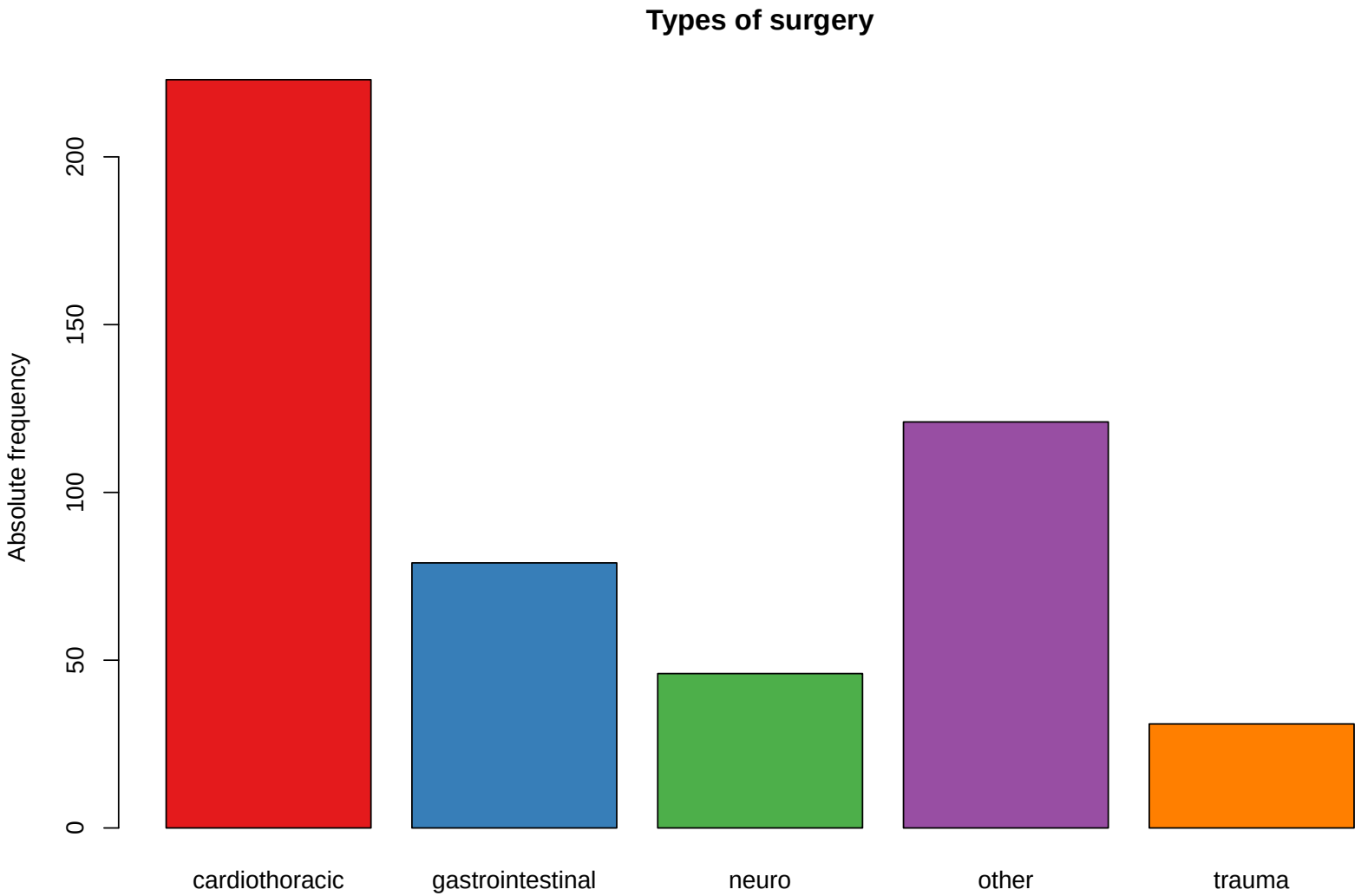

Normal Cumulative Distribution Function
$\phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{(-t^2/2)} dt$
cumsum(x)/sum(x)
0.0
0.2
0.4
0.6
0.8
1.0
−3
−2
−1
0
1
2
3
x

anscombe 数据

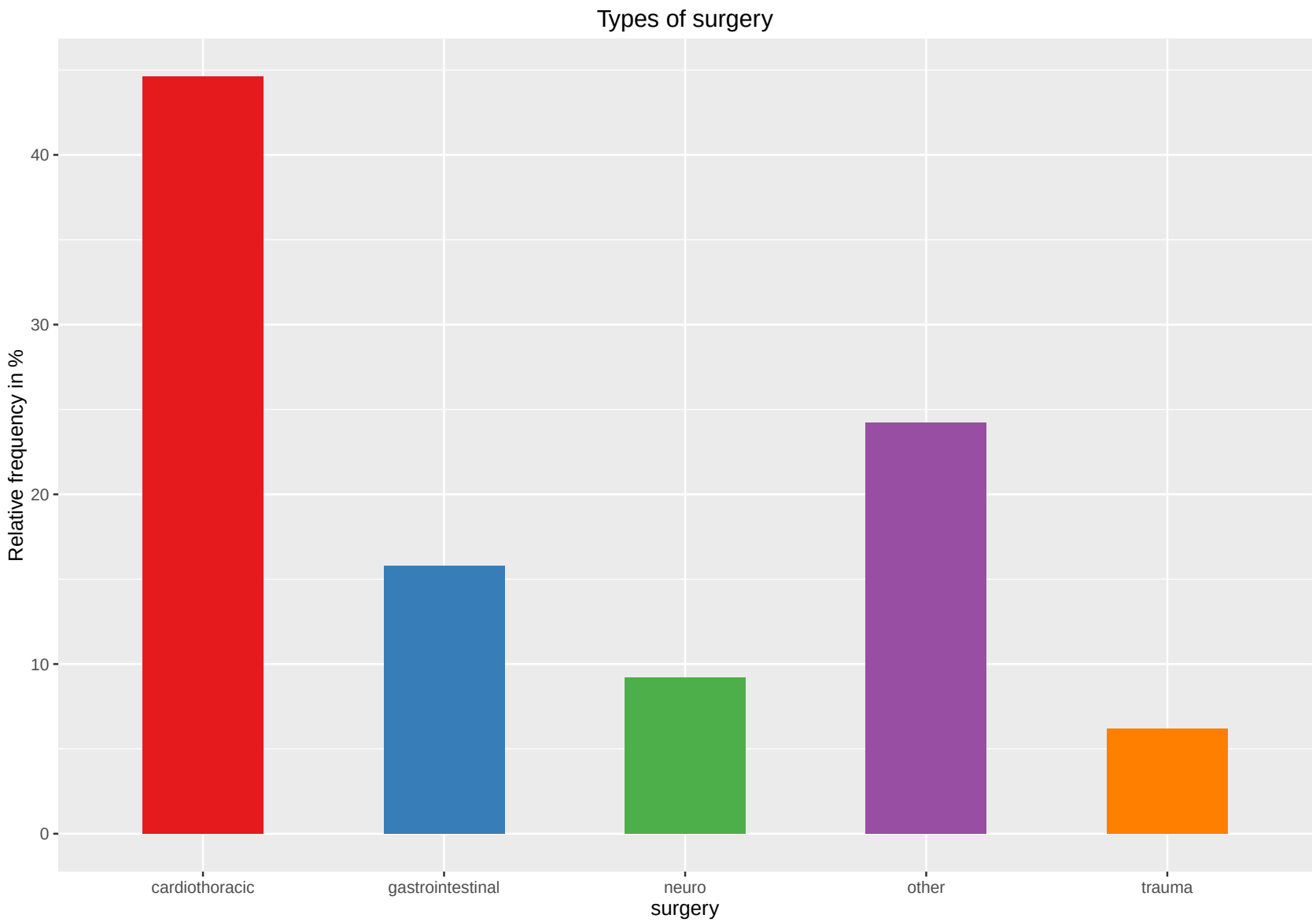
$\hat{\sigma}^2 = 1.531$,Multiple $R^2 = 0.667$,Adjusted $R^2 = 0.629$



barplot


Types of surgery
Absolute frequency
0
50
100
150
200
cardiothoracic
gastrointestinal
neuro
other
trauma

barplot


Types of surgery
Relative frequency in %
0
10
20
30
40
cardiothoracic
gastrointestinal
neuro
other
trauma
surgery

目录

多元概率分布函数

The multivariate t distribution (MVT) is given by

$$T(\mathbf{a}, \mathbf{b}, \Sigma, \nu) = \frac{2^{1-\frac{\nu}{2}}}{\Gamma(\frac{\nu}{2})} \int_0^\infty s^{\nu-1} e^{-\frac{s^2}{2}} \Phi(\frac{s\mathbf{a}}{\sqrt{\nu}}, \frac{s\mathbf{b}}{\sqrt{\nu}}, \Sigma) ds$$

multivariate normal distribution function (MVN)

$$\Phi(\mathbf{a}, \mathbf{b}, \Sigma) = \frac{1}{\sqrt{|\Sigma|(2\pi)^m}} \int_{a_1}^{b_1} \int_{a_2}^{b_2} \cdots \int_{a_m}^{b_m} e^{-\frac{1}{2} x^\top \Sigma^{-1} x} dx$$

$x = (x_1, x_2, \ldots, x_m)^\top$, $-\infty \leq a_i \leq b_i \leq \infty$ for all i, and Σ is a positive semi-definite symmetric $m \times m$ matrix

计算多元正态分布的概率

```
>library(mvtnorm)
>library(matrixcalc)
>path<-"C:/Users/Xiangyun Huang/Desktop/optimization/"
>setwd(path)
>sigma <- read.csv(file="data/sigma1.csv",header=F, sep=",")
>mat<-matrix(0,nrow = nrow(sigma),ncol = ncol(sigma))
>sigma <- as.matrix(sigma)
>attributes(sigma)<-attributes(mat)
># str(sigma)
># is.symmetric.matrix(sigma)
># is.positive.definite(sigma)
>m = nrow(sigma)
>Fn = pmvnorm(lower=rep(-Inf, m), upper=rep(0, m),
+              mean=rep(0, m), sigma =sigma)
>Fn

[1] 6.745319e-29
attr(,"error")
[1] 2.903713e-31
attr(,"msg")
[1] "Normal Completion"
```

多元 t 分布分位数计算

```
>n <- c(26, 24, 20, 33, 32)
>V <- diag(1/n); df <- 130
>C <- matrix(c(1,1,1,0,0,-1,0,0,1,0,
+         0,-1,0,0,1,0,0,0,-1,-1,
+         0,0,-1,0,0),ncol=5)
>cv <- C %*% V %*% t(C) ## covariance matrix
>dv <- t(1/sqrt(diag(cv)))
>cr <- cv * (t(dv) %*% dv) ## correlation matrix
>delta <- rep(0,5)
>Tn<-qmvt(0.95, df = df, delta = delta, corr = cr,
+       abseps = 0.0001,maxpts = 100000, tail = "both")
>Tn

$quantile
[1] 2.561101

$f.quantile
[1] 2.299229e-07

attr(,"message")
[1] "Normal Completion"
```

优化分类

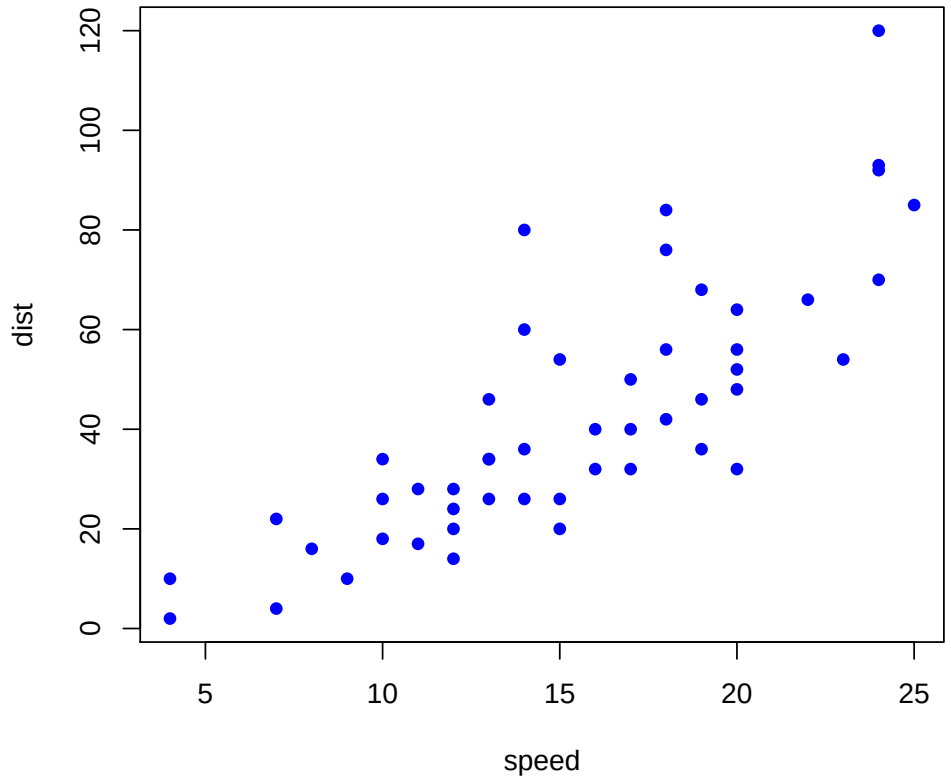
1. 无约束优化
   1. optimize(optimise) 一元函数极值 (黄金分割搜索算法)
   2. nlm 多元非线性函数极值 (Newton 型算法)
   3. optim 多元非线性函数极值 (Nelder-Mead 算法、拟牛顿法、共轭梯度算法、模拟退火算法)
   4. nlminb 无约束优化
   5. BB[4] 和 numDeriv 包高维非线性目标函数优化 (Barzilai-Borwein 算法)
2. 约束优化
   1. 箱式 (box) 约束：nlminb 和 BB 包
   2. 线性不等式约束：constrOptim 障碍罚函数方法
   3. 非线性约束：暂无

无约束优化

$$
\begin{aligned}
Q_1 &= \min_{\beta_0,\beta_1} \sum_{i=1}^{n} |\beta_0 + \beta_1 x_i - y_i| \\
Q_2 &= \min_{\beta_0,\beta_1} \sum_{i=1}^{n} (\beta_0 + \beta_1 x_i - y_i)^2 \\
Q_3 &= \min_{\beta_0,\beta_1} \max_{1\le i\le n} |\beta_0 + \beta_1 x_i - y_i|
\end{aligned}
\tag{1}
$$

举例子

```
>library(dplyr)
>tbl_df(cars)

Source: local data frame [50 x 2]

   speed  dist
   (dbl) (dbl)
1      4     2
2      4    10
3      7     4
4      7    22
5      8    16
6      9    10
7     10    18
8     10    26
9     10    34
10    11    17
..   ...   ...
```



回归系数

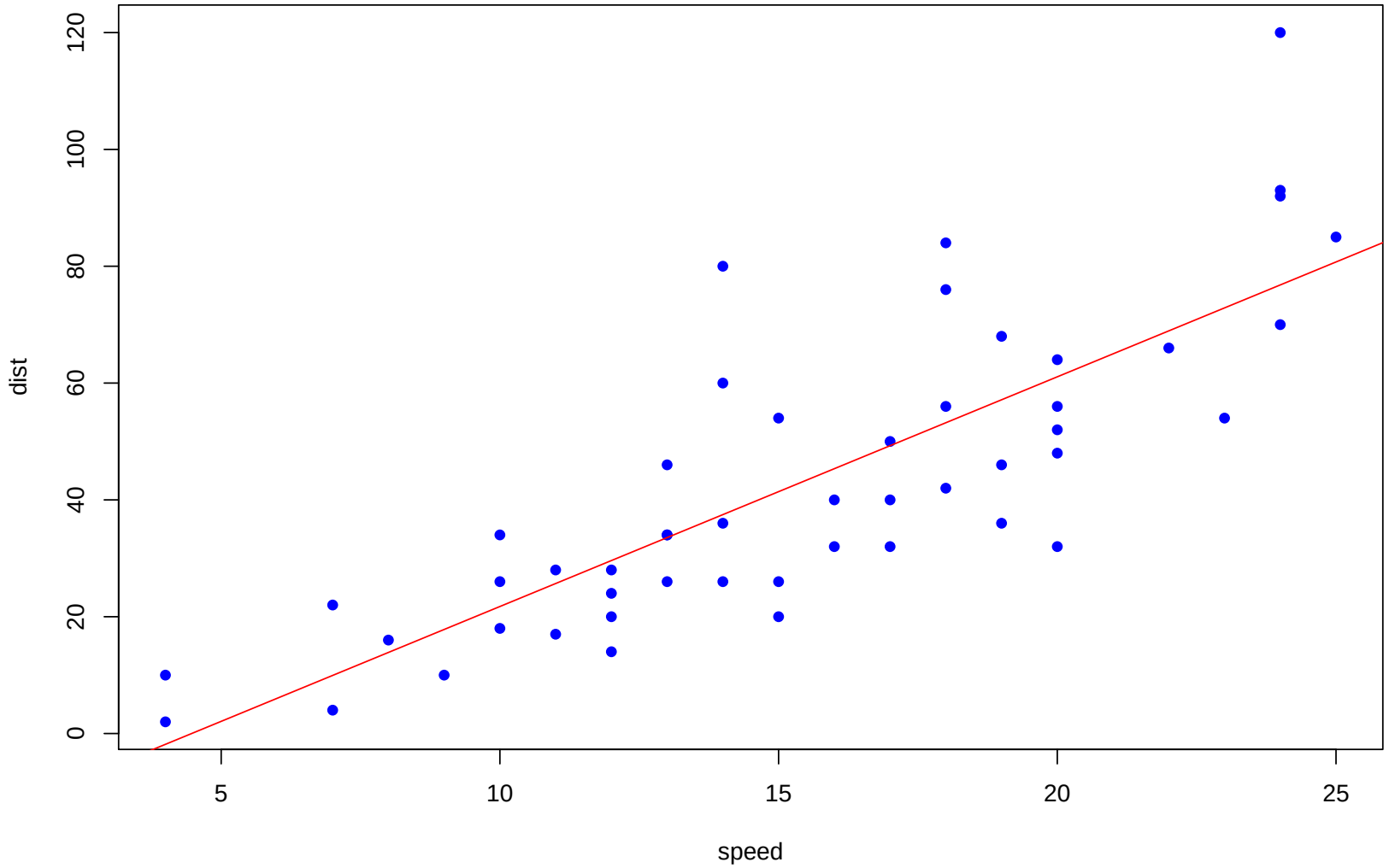
```
>## Linear Regression with L1 Regularization
>reg1fun<-function(beta,data){
+   sum(abs(data[,2]-beta[1]-beta[2]*data[,1]))
+ }
>optim(c(0,4),reg1fun,data=cars)$par

[1] -11.60001   3.40000

>## Linear Regression with L2 Regularization
>reg2fun<-function(beta,data){
+   sum((data[,2]-beta[1]-beta[2]*data[,1])^2)
+ }
>optim(c(0,4),reg2fun,data=cars)$par

[1] -17.570424   3.931992

>## Linear Regression with minimax
>reg3fun<-function(beta,data){
+   max(abs(data[,2]-beta[1]-beta[2]*data[,1]))
+ }
>optim(c(0,4),reg3fun,data=cars)$par

[1] -11.998563   3.999935
```

线性回归 lm

Table: Regression Results

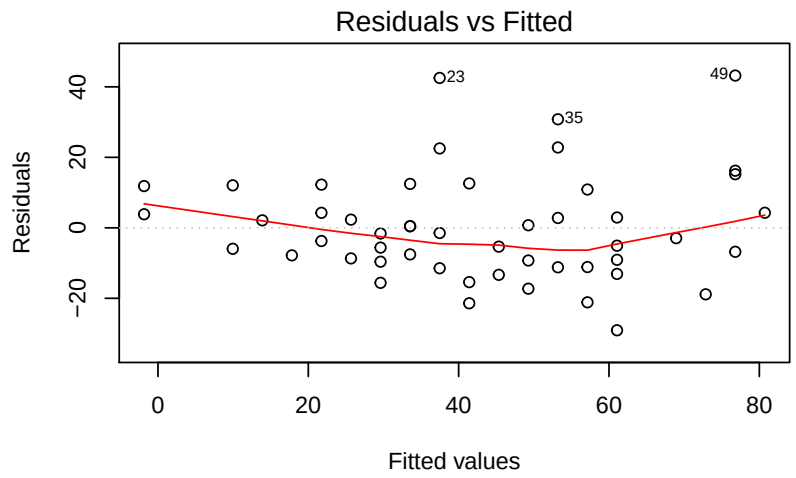
	Dependent variable:
	dist
speed	3.932***
	(0.416)
Constant	−17.579**
	(6.758)
Observations	50
R^2	0.651
Adjusted R^2	0.644
Residual Std. Error	15.380 (df = 48)
F Statistic	89.567*** (df = 1; 48)
Note:	*p<0.1; **p<0.05; ***p<0.01

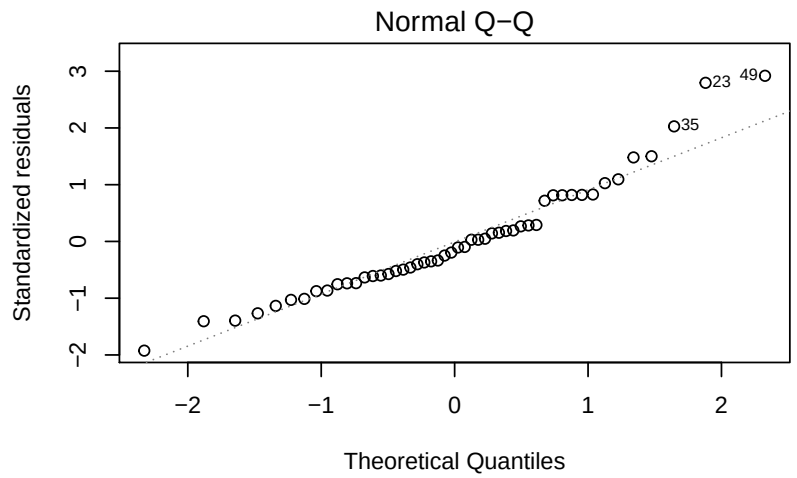
线性回归

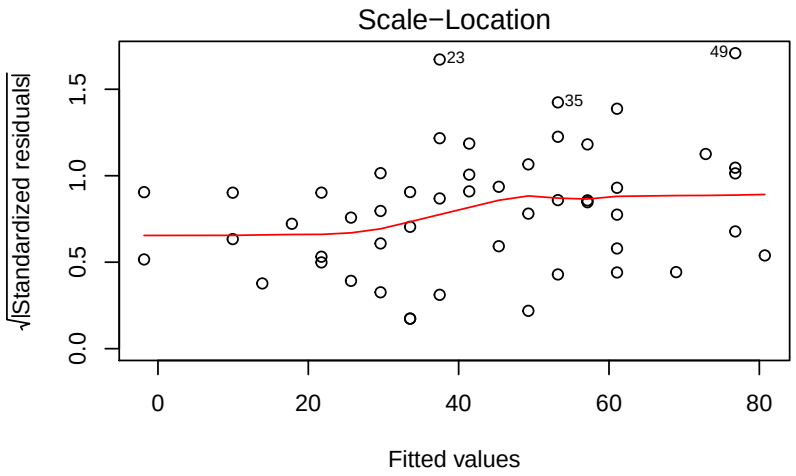


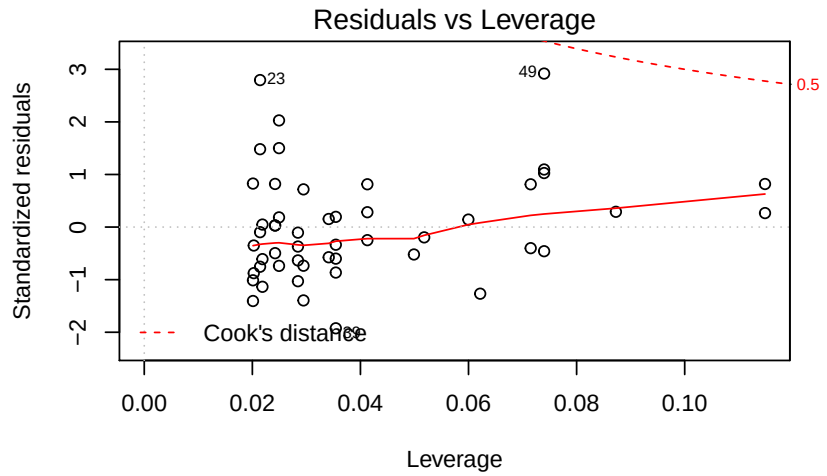
回归诊断

lm(dist ~ speed)









Γ 分布参数估计

The Gamma distribution with parameters shape $= a$ and scale $= s$ has density

$$f(x) = \frac{1}{s^a \Gamma(a)} x^{a-1} e^{-\frac{x}{s}}$$

$$E(x) = as, Var(x) = as^2$$

矩估计：用样本均值 $\bar{x}$ 替换总体均值 $E(x)$，样本方差 S^2 替换总体方差 $Var(x)$

$$\hat{a}_{ME} = \frac{\bar{x}^2}{S^2}, \hat{s}_{ME} = \frac{S^2}{\bar{x}}$$

最大似然估计

似然函数

$$L(x_1, x_2, \ldots, x_n; a, s) = (\frac{1}{s^a \Gamma(a)})^n (\prod_{i=1}^{n} x_i)^{a-1} e^{-\frac{\sum_{i=1}^{n} x_i}{s}}$$

对数似然函数

$$logL(x_1, x_2, \ldots, x_n; a, s) = -n(alogs + log\Gamma(a)) + (a-1)\sum_{i=1}^{n} logx_i - \frac{1}{s}\sum_{i=1}^{n} x_i$$

似然方程组

$$\begin{cases} \frac{\partial logL}{\partial a} & = -n[logs + (log\Gamma(a))'] + \sum_{i=1}^{n} logx_i = 0 \\ \frac{\partial logL}{\partial s} & = -\frac{na}{s} + \frac{\sum_{i=1}^{n} x_i}{s^2} = 0 \end{cases}$$

最大似然估计

将 $s = \frac{1}{na}\sum_{i=1}^{n} x_i$ 代入似然方程组，可得

$$loga - [log\Gamma(a)]' = \frac{1}{n}\sum_{i=1}^{n} logx_i - log(\frac{1}{n}\sum_{i=1}^{n} x_i) \tag{2}$$

其中

$$\Gamma(a) = \int_0^{+\infty} e^{-t}t^{a-1}\,dt$$

$$[\Gamma(a)]' = \int_0^{+\infty} e^{-t}t^{a-1}\,logt dt$$

似然方程组的求解问题，归结为求一元非线性方程 (2). 调用 Brent 算法[1] 求解

数值模拟结果

```
>set.seed(1234)
>x<-rgamma(100,shape = 3/2,scale = 2)
>## Moment Estimation
>c(mean(x)^2/var(x),var(x)/mean(x))  #shape and scale

[1] 1.224125 2.368657

>## Maximum Likelihood Estimation
>C<-log(mean(x))- mean(log(x))
>Gammafun<-function(a){
+   tempfun<-function(s){
+     exp(-s)*s^(a-1)*log(s)
+   }
+   return(tempfun)
+ }
>myfun<-function(alpha){
+   log(alpha)-C-integrate(Gammafun(alpha),lower = 0,upper = Inf)$value/gamma(alpha)
+ }
>(a<-uniroot(myfun,interval = c(1,3))$root) #shape

[1] 1.442917

>mean(x)/a #scale

[1] 2.009493
```

目录

非凸函数

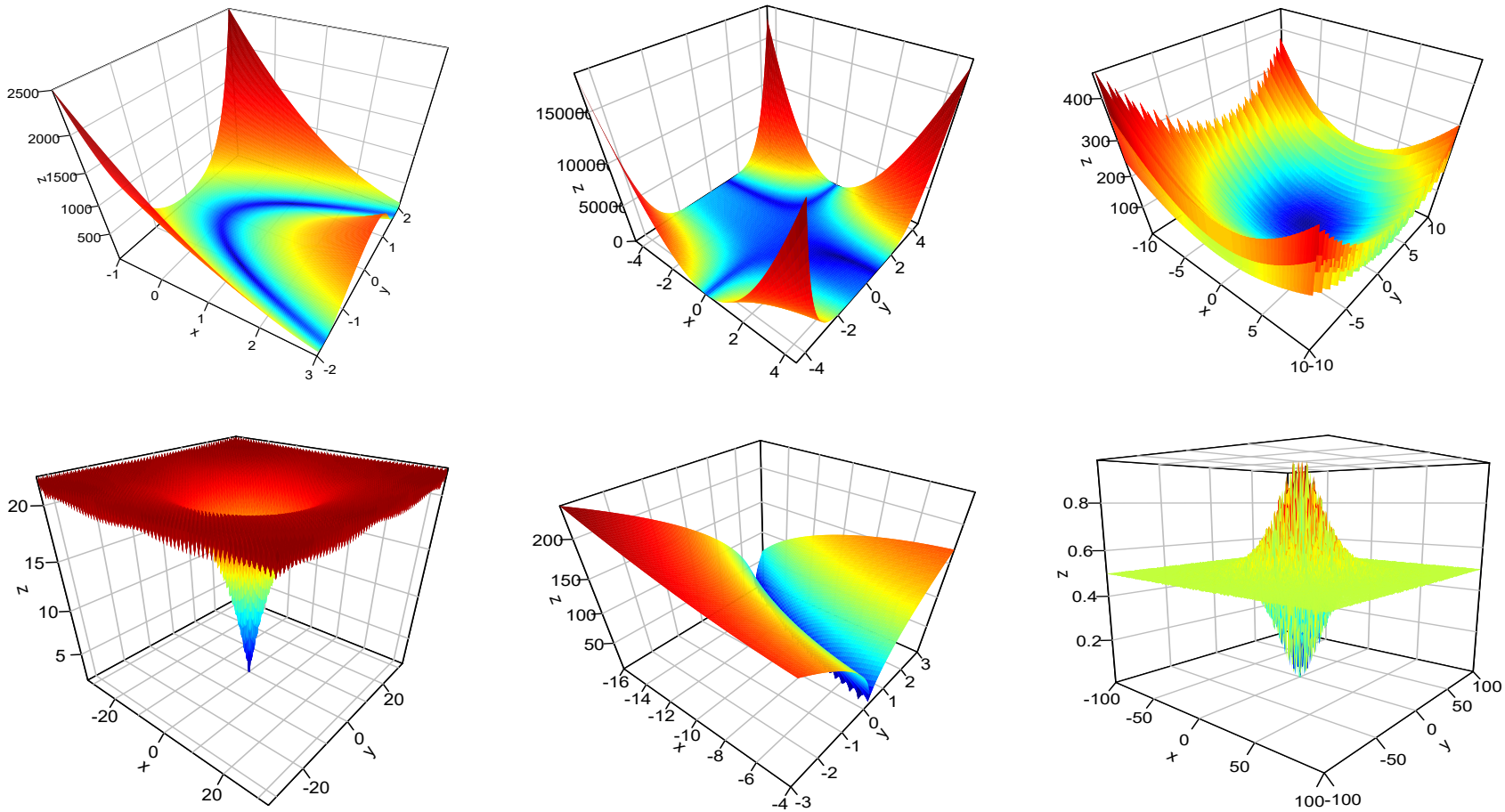


Figure: Rosenbrock Beale Levy Ackley Bukin Schaffer

介绍

蒙特卡罗 (Monte Carlo) 是随机模拟的别称. Monte Carlo 本是 Monaco(摩纳哥) 的著名赌城，第二次世界大战期间，N.Metropolis 在曼哈顿计划中取其博彩游戏和随机模拟算法之间的相似之处，同时也为了保密起见，首次借用其名作为随机模拟算法的名称.

定义.

如果存在自然数 τ，使得马氏链的转移概率矩阵 $\boldsymbol{P}$ 满足 $\boldsymbol{P}^{\tau} > 0$ (这里矩阵 $\boldsymbol{A} > 0$ 表示 $\boldsymbol{A}$ 的每个元素都严格大于 0)，则称该马氏链为本原的

时齐马氏链的遍历定理.

本原马氏链的转移概率矩阵 $\boldsymbol{P}$ 有唯一的不变分布 $\boldsymbol{\pi}$，且对任意初始分布 $\boldsymbol{\nu}$，有

$$\lim_{n \to +\infty} \| \boldsymbol{\nu} \boldsymbol{P}^n - \boldsymbol{\pi} \| = 0$$

举例子

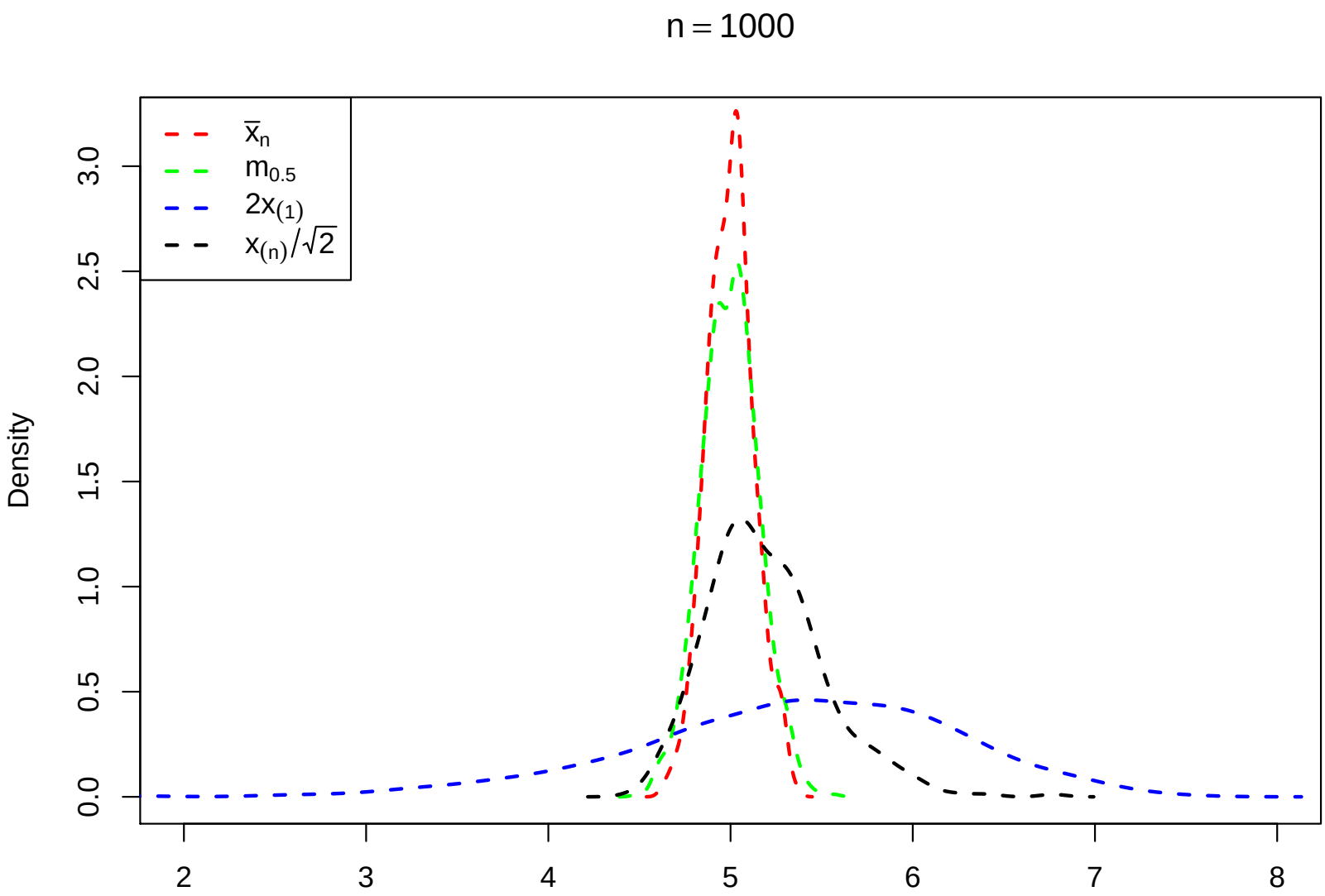
样本 $X_1, X_2, ..., X_n$ 抽自总体 $N(\theta, 1)$，现分别用样本均值 $\bar{X}_n$ 和中位数 $m_{0.5}$ 估计总体均值 θ，得到

$$\sqrt{n}(\bar{X}_n - \theta) \xrightarrow{L} N(0, 1)$$

和

$$\sqrt{n}(m_{0.5} - \theta) \xrightarrow{L} N(0, \frac{\pi}{2})$$

模拟结果


n = 1000
$\bar{x}_n$
$m_{0.5}$
$2x_{(1)}$
$x_{(n)}/\sqrt{2}$
Density
0.0
0.5
1.0
1.5
2.0
2.5
3.0
2
3
4
5
6
7
8

Integration

计算下面的积分

$$\int_3^6 \frac{1}{10\sqrt{2\pi}} e^{-\frac{(x-1)^2}{2\cdot 10^2}} dx$$

```
>runs <- 100000
>sims <- rnorm(runs,mean=1,sd=10)
>mc.integral <- sum(sims >= 3 & sims <= 6)/runs
>mc.integral

[1] 0.11149

>mc.fun<-function(x){
+   1/(10*sqrt(2*pi))*exp(-(x-1)^2/(2*10^2))
+ }
>integrate(mc.fun,3,6)

0.1122028 with absolute error < 1.2e-15
```

Approximating the Binomial Distribution

We flip a coin 10 times and we want to know the probability of getting more than 3 heads.

```
>runs <- 100000
>#one.trail simulates a single round of toss 10 coins
>#and returns true if the number of heads is > 3
>one.trial <- function(){
+   sum(sample(c(0,1),10,replace=T)) > 3
+ }
>#now we repeat that trial 'runs' times.
>mc.binom <- sum(replicate(runs,one.trial()))/runs
>mc.binom

[1] 0.82871

>pbinom(3,10,0.5,lower.tail=FALSE)

[1] 0.828125
```

Introduction

Definition

EM Algorithm a general approach to iterative computation of maximum-likelihood estimates when the observations can be viewed as incomplete data.

1. including missing value situations
2. applications to grouped, censored or truncated data
3. finite mixture models
4. variance component estimation
5. hyperparameter estimation
6. iteratively reweighted least squares
7. factor analysis

Why EM ?

有限混合模型 (高斯正态混合)

样本 $x_1, x_2, ..., x_n$ 取自密度函数为 $f(x)$ 的总体，$f(x)$ 由 m 个正态分布 $N(\mu_i, \sigma_i^2), i = 1, 2, ..., m$ 分别以比例 p_i 混合.

$$f(x) = \sum_{i=1}^{m} p_i f_i(x)$$

$$f_i(x) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}} \tag{3}$$

$$\sum_{i=1}^{m} p_i = 1$$

Goal 根据样本 $x_1, x_2, ..., x_n$ 估计总体的参数 $\mu_i, \sigma_i^2, p_i, i = 1, ..., m$

最大似然估计

似然函数

$$L(x_i;\mu_i,\sigma_i^2,p_i)=\prod_{j=1}^{n}f(x_j)$$

对数似然函数

$$logL(x_i;\mu_i,\sigma_i^2,p_i)=\sum_{j=1}^{n}log(\sum_{i=1}^{m}p_if_i(x_j))$$

对数似然方程组

$$\begin{aligned}
\frac{\partial logL(x_i;\mu_i,\sigma_i^2,p_i)}{\partial p_i}&=\sum_{j=1}^{n}\frac{f_i(x_j)}{\sum_{i=1}^{m}p_if_i(x_j)}=0\\
\frac{\partial logL(x_i;\mu_i,\sigma_i^2,p_i)}{\partial \mu_i}&=\sum_{j=1}^{n}\frac{p_if_i(x_j)\frac{x_j-\mu_i}{\sigma_i^2}}{\sum_{i=1}^{m}p_if_i(x_j)}=0\\
\frac{\partial logL(x_i;\mu_i,\sigma_i^2,p_i)}{\partial \sigma_i^2}&=\sum_{j=1}^{n}\frac{p_if_i(x_j)\frac{(x_j-\mu_i)^2-\sigma_i^2}{2\sigma_i^4}}{\sum_{i=1}^{m}p_if_i(x_j)}=0
\end{aligned}\tag{4}$$

非线性对数似然方程组的挑战

将 $\sum_{i=1}^{m} p_i = 1$ 代入 (4)，可得到共 $3m-1$ 个方程，$3m-1$ 个未知数. 显然这是一个非线性方程组，怎么解?

1. 精确解的数学理论 (存在性、唯一性等) 不清楚
2. 有些算法的原理在多维情形下不成立，如二分法
3. 有些算法能推广到多维，但如何推广值得研究，如牛顿迭代和割线法
4. 当方程组含有许多方程时，每步迭代的运算量大成为突出问题

对于多项式方程组，同伦算法[2](或称延拓法) 可以解，但是对于更一般类型，没有严格的理论基础!

Genetic Algorithms(遗传算法[3])

1. Genetic algorithms (GAs) are stochastic search algorithms inspired by the basic principles of biological evolution(生物进化) and natural selection(自然选择).
2. GAs simulate the evolution of living organisms, where the fittest individuals dominate over the weaker ones, by mimicking the biological mechanisms of evolution, such as selection(选择), crossover(杂交) and mutation(突变).
3. GAs have been successfully applied to solve optimization problems, both for continuous (whether differentiable or not) and discrete functions.

GA 包介绍

- 目标函数可离散或连续
- (不) 带限制条件
- 可以自定义目标函数
- 有可供选择的基因操作
- 可以自定义基因操作，并对其评估
- GA 算法可串行或并行

```
>library(foreach)
>library(iterators)
>library(GA)
>args("ga")

function (type = c("binary", "real-valued", "permutation"), fitness,
    ..., min, max, nBits, population = gaControl(type)$population,
    selection = gaControl(type)$selection, crossover = gaControl(type)$crossover,
    mutation = gaControl(type)$mutation, popSize = 50, pcrossover = 0.8,
    pmutation = 0.1, elitism = base::max(1, round(popSize * 0.05)),
    updatePop = FALSE, postFitness = NULL, maxiter = 100, run = maxiter,
    maxFitness = Inf, names = NULL, suggestions = NULL, optim = FALSE,
    optimArgs = list(method = "L-BFGS-B", poptim = 0.05, pressel = 0.5,
        control = list(fnscale = -1, maxit = 100)), keepBest = FALSE,
    parallel = FALSE, monitor = if (interactive()) gaMonitor else FALSE,
    seed = NULL)
NULL
```

其他 R 包

- gafit：Genetic Algorithm for Curve Fitting(已经不维护)
- galts：Genetic algorithms and C-steps based LTS (Least Trimmed Squares) estimation.
- mcga：Machine Coded Genetic Algorithms for Real-Valued Optimization Problems.
- rgenoud：R Version of GENetic Optimization Using Derivatives.
- genalg：R based genetic algorithm for binary and floating point chromosomes.
- DEoptim：Implements the differential evolution algorithm for global optimization of a real-valued function of a real-valued parameter vector.

MC 方法小结

step 1 产生随机数

step 2 从已知分布抽样

step 3 计算相应统计量

step 4 重复上述过程 10000 次

数值算法 VS 概率算法

如果数值算法能解，就不用概率算法，数值算法效率高，精度高！但是，现实问题很复杂，唱主角的往往是概率算法，不过，数值算法依然在局部搜索中发挥重要作用，在概率算法中结合数值算法是现在研究的主流！

工具

计算 R[1]

绘图 graphics

渲染 Cairo

排版 TEX

编辑 LYX[2] RStudio[3]

[1] https://www.r-project.org/
[2] http://www.lyx.org/
[3] https://www.rstudio.com/

部分参考文献 I

R. P. Brent.

Algorithms for Minimization without Derivatives.
Prentice-Hall, 1973.

T. Y. Li.

Numerical solution of multivariate polynomial systems by homotopy continuation methods.
Acta Numerica, 6(2):399–436, 1997.

L. Scrucca.

GA: A package for genetic algorithms in R.
Journal of Statistical Software, 53(4):1–37, 2013.

R. Varadhan and P. Gilbert.

BB: An R package for solving a large system of nonlinear equations and for optimizing a high-dimensional nonlinear objective function.
Journal of Statistical Software, 32(4):1–26, 2009.

Thanks for your attention!

Q & A