

Thực hành xác suất thống kê

Bài 2: LẬP TRÌNH CƠ BẢN TRONG R

1. Các phép toán và các hàm cơ bản trong R

1.1 Các phép toán trong R

- Cộng (+), trừ (-), nhân (*), chia (/)

1. Các phép toán và các hàm cơ bản trong R

1.2 Các phép toán logic trong R

$x == 3$	x bằng 3
$x != 3$	x khác 3
$x < y$	x nhỏ hơn y
$x > y$	x lớn hơn y
$x <= y$	x nhỏ hơn hay bằng y
$x >= y$	x lớn hơn hay bằng y
$z <= 5$	z nhỏ hơn hay bằng 5
$z >= 5$	z lớn hơn hay bằng 5
$\text{is.na}(x)$	có phải x là biến trống không
$A \& B$	A và B (so sánh chân trị (AND) trên mỗi thành phần tương ứng của 2 vectơ A và B)
$A \&\& B$	A và B (so sánh chân trị (AND) trên thành phần đầu tiên kể từ trái qua phải của 2 vectơ A và B)
$A B$	A và B (so sánh chân trị (OR) trên mỗi thành phần tương ứng của 2 vectơ A và B)
$A B$	A và B (so sánh chân trị (AND) trên thành phần đầu tiên kể từ trái qua phải của 2 vectơ A và B)
$!A$	phủ định A (NOT A)
$\text{xor}(x, y)$	lấy XOR trên mỗi thành phần tương ứng của 2 vectơ x và y

1. Các phép toán và các hàm cơ bản trong R

1.3 Các hàm toán học thường dùng trong R

`log(x)` : logarit cơ số e

`log10(x)`, `log(x,n)`: logarit cơ số n

`exp(x)`: e^x

`sqrt(x)`: căn bậc 2 của x

`factorial(x)`: $x!$

`choose(n,k)`: tổ hợp n chập k

`floor(x)`: giá trị nguyên $< x$ (sàn của x)

`ceiling(x)`: giá trị nguyên $> x$ (trần của x)

`trunc(x)`: làm tròn tới giá trị nguyên gần nhất giữa x và 0.

`round(x, digits=n)`: làm tròn x đến n chữ số

`signif(x, digits=n)`: hiển thị x dưới dạng dấu chấm thập phân, n tổng chữ số hiển thị

`sin(x)`, `cos(x)`, `tan(x)`

`abs(x)`: $|x|$

`x %/% y`: lấy phần nguyên của phép chia x/y

`x %% y`: lấy phần dư của phép chia x/y

2. Lệnh điều kiện: if

- Cấu trúc

```
if(biểu thức đk){  
    biểu thức 1 }
```

```
else{  
    biểu thức 2 }
```

3. Vòng lặp: for, while, repeat

- Cấu trúc
 - **for**(biến chạy **in** biểu thức đk) { biểu thức lệnh }
 - **repeat** biểu thức lệnh
 - **while** biểu thức đk biểu thức lệnh
- Ngắt vòng lặp: **break**, **next**.

4. Script và hàm

4.1 Script

- Tập hợp các đoạn lệnh
- Tạo script: File → New script
- Mở 1 script đã có:
 - File → Open script
 - **source**("tên script.r")

4. Script và hàm

4.2 Hàm

- Cấu trúc

tên hàm <- **function**(tham số 1, tham số 2, ...){
 các biểu thức lệnh }

- Ví dụ: Viết hàm đếm số phần tử lớn hơn 3 trong véc-tơ x.

```
# Viet ham dem so phan tu lon hon 3 trong vec-to x
Dem <- function(x){
  dem = 0
  n <- length(x)
  for(i in 1:n)
    if(x[i] > 3) dem = dem + 1
  return(dem)
}
```

5. Bài tập

- Bài 1. Tạo một véc-tơ X chứa n phần tử (n : tự cho). Viết hàm tính tổng tích lũy đến vị trí thứ i của X .

```
# Bai 1.
# Nhap gia tri cho n
n <- 40
# co the dung lenh sample thay cho runif
x <- round(runif(n,0,10),2)
# Ham tinh tong tích luy den vi tri thu i (1 <= i <= n)
cum.sum <- function(x,i){
  s <- 0
  for(k in 1:i) s <- s + x[k]
  return(s)
}
#Kiem tra lai voi vecto x vua tao
tongtl <- cum.sum(x,3);tongtl

#Xoa cac bien truoc khi chay bai sau
rm(list=ls(all=TRUE))
```

5. Bài tập

- Bài 2. Thể tích hình cầu với bán kính r được tính bởi công thức

$V = \frac{4}{3}\pi r^3$. Hãy viết hàm xây dựng 1 dataframe để tính thể tích hình cầu với bán kính tương ứng là 3,4,5,...,20. Cột Radius lưu bán kính và cột Volume lưu thể tích.

```
#Bai 2.
```

```
#Viet ham xuat ra dataframe chua the tich hinh cau tuong ung voi  
#ban kinh tu 3,4,5,...,n
```

```
cal.vol <- function(){  
  r <- 3:20  
  v <- round(4*pi*r^3/3,3)  
  d <- data.frame(Radius = r, Volume = v)  
  return(d)  
}
```

```
ans <- cal.vol();ans
```

5. Bài tập

- Bài 3. Từ dữ liệu trong file data01.csv, dùng lệnh if và vòng lặp for để tạo biến Index theo yêu cầu sau

Điều kiện	Index
Age <= 60	0
60 < Age <= 70	1
70 < Age <= 80	2
80 < Age	3

#Bai 3.

#Doc file data01.csv

data01 <- read.csv("data01.csv",header=T)

names(data01)

attach(data01)

Tao bien va gan gia tri

n <- length(Age)

Index <- numeric(n)

for(i in 1:n){

 if (Age[i] <= 60) {Index[i] <- 0}

 else{

 if (Age[i] <= 70) {Index[i] <- 1}

 else{

 if (Age[i] <= 80) {Index[i] <- 2}

 else{

 Index[i] <- 3

 }

 }

 }

ans <- data.frame(Age,Index); ans

#Xoa cac bien truoc khi chay bai sau

detach(data01)

rm(list=ls(all=TRUE))

Điều kiện	Index
Age <= 60	0
60 < Age <= 70	1
70 < Age <= 80	2
80 < Age	3

5. Bài tập

- Bài 4. File data11.xls chứa số liệu về chiều cao của 1 loại cây trồng theo bảng tần số dạng khoảng. Thực hiện các bước sau
 - a. Đọc số liệu từ data11.xls và gán vào 1 dataframe.
 - b. Viết một hàm tính tham số là các biến trong dataframe vừa nhập, xuất ra các giá trị sau: chiều cao bé nhất, lớn nhất của cây, trung bình mẫu, phương sai mẫu hiệu chỉnh.

5. Bài tập

- Bài 4. File data11.xls chứa số liệu về chiều cao của 1 loại cây trồng theo bảng tần số dạng khoảng. Thực hiện các bước sau
 - a. Đọc số liệu từ data11.xls và gán vào 1 dataframe.

```
#Bai 4.  
# a/ Doc file data11.csv và gán vào 1 dataframe.  
data11 <- read.csv("data11.csv",header=T)  
names(data11)  
attach(data11)  
data11
```

5. Bài tập

- Bài 4. File data11.xls chứa số liệu về chiều cao của 1 loại cây trồng theo bảng tần số dạng khoảng. Thực hiện các bước sau
 - b. Viết một hàm tính tham số là các biến trong dataframe vừa nhập, xuất ra các giá trị sau: chiều cao bé nhất, lớn nhất của cây, trung bình mẫu, phương sai mẫu hiệu chỉnh.

5. Bài tập

X	$[a_1; b_1)$	$[a_2; b_2)$...	$[a_k; b_k)$
N	n_1	n_2	...	n_k



X	$\frac{a_1 + b_1}{2}$	$\frac{a_2 + b_2}{2}$...	$\frac{a_k + b_k}{2}$
N	n_1	n_2	...	n_k

```
# b/ Viet ham tinh cac tham so mau cho bang tan so dang khoang
cal.sample <- function(d){
#d: data.frame co dang bang tan so dang khoang gom can tren a,
#can duoi b, va tan so n
  t <- (a+b)/2
  x <- rep(t,n)
  m <- min(x)
  M <- max(x)
  x.bar <- mean(x)
  s2 <- var(x)
  dat <- data.frame(Min = m, Max = M, Mean = x.bar, Var = s2)
  rownames(dat) <- c("")
  return(dat)
}

ans <- cal.sample(data11)

#Xoa cac bien truoc khi chay bai sau
detach(data11)
rm(list=ls(all=TRUE))
```

5. Bài tập

- Bài 5. Cho véc-tơ X chứa n giá trị quan sát, phân vị thứ p được xác định như sau
 - Sắp xếp dữ liệu theo thứ tự tăng dần (từ nhỏ đến lớn).
 - Tính chỉ số i

$$i = \left(\frac{p}{100} \right) \cdot n$$

- Nếu i không phải là số nguyên, làm tròn i . Phân vị thứ p chính là giá trị nằm ở vị trí thứ i đã được làm tròn.
 - Nếu i nguyên, phân vị thứ p chính là giá trị trung bình của 2 giá trị nằm ở vị trí thứ i và thứ $i + 1$.
- Hãy viết hàm $\text{phanvi}(X,p)$ cho kết quả là phân vị thứ p từ véc-tơ X .

#Bai 5.

#Tim phan vi

```
phanvi <- function(x,p){  
  x <- sort(x)  
  i <- (p/100)*length(x)  
  if (i - round(i)!=0)  
    quantile <- x[ceiling(i)]  
  else  
    quantile <- (x[i] + x[i+1])/2  
  return(quantile)  
}
```

Tao vecto x

```
x <- round(runif(22,0,20),2)
```

x

Goi lai ham de tim phan vi thu p = 25

```
q <- phanvi(x,25)
```

q