

Bài tập thực hành
**XÁC SUẤT THỐNG
KÊ**

Bộ môn Toán - ĐẠI HỌC THĂNG LONG

Ngày 3 tháng 9 năm 2009

Mục lục

	Trang
Bài 1. Dữ liệu và một số thao tác tiền xử lý dữ liệu	1
1.1 Các phép toán	1
1.1.1 Các toán tử	1
1.1.2 Tính toán đơn giản trong R	1
1.1.3 Ma trận	2
1.2 Dữ liệu	4
1.2.1 Dữ liệu là một vector	4
1.2.2 Dữ liệu là một data frame	6
1.3 Tạo dữ liệu	11
1.3.1 Tạo dãy số bằng seq	11
1.3.2 Tạo dãy số lặp lại bằng rep	12
1.3.3 Tạo biến thứ bậc bằng gl	12
1.4 Nhập dữ liệu từ các file dữ liệu	13
1.4.1 Nhập dữ liệu từ một text file	13
1.4.2 Nhập dữ liệu từ Excel	13
1.4.3 Nhập dữ liệu từ SPSS	14
1.5 Chọn mẫu ngẫu nhiên	14
1.6 Bài tập	15

Bài 1

ĐỮ LIỆU VÀ MỘT SỐ THAO TÁC TIỀN XỬ LÝ ĐỮ LIỆU

1.1 Các phép toán

Các toán tử

Sau đây là danh sách ba loại toán tử trong R

Toán tử số học		Toán tử so sánh		Toán tử logic	
+	cộng	<	nhỏ hơn	!x	phủ định
-	trừ	>	lớn hơn	x&y	và
*	nhân	<=	nhỏ hơn hoặc bằng	x y	hoặc
/	chia	>=	lớn hơn hoặc bằng		
%%	modulo	!=	khác	xor(x,y)	tuyển loại
%/%	thương số				

Ngoài ra \$, [, [[, :, ~, ?[, <-, ... cũng là những toán tử trong R.

Tính toán đơn giản trong R

Một số hàm toán học dùng để tính toán đơn giản trên R:

abs, sqrt, floor, ceiling, round
exp, log, cos, sin, tan
cumsum, cumprod, cummax, cummin
sum, prod, min, max, range

Cộng và trừ > 20+568-100 [1] 488	Nhân và chia > 12*15/8 [1] 22.5
Lũy thừa > 15^2 [1] 225	Logarit > log(100,base=10) 2
Lũy thừa cơ số e > exp(3) [1] 20.08554	Logarit cơ số e > log(5) [1] 1.609438
Số pi (π) > pi [1] 3.141593	Hàm số lượng giác > cos(pi) [1] -1
Trị tuyệt đối > abs(-8) [1] 8	Căn bậc hai > sqrt(4) [1] 2
Vector > x<- c(1,4,7,2,9) > sum(x) [1] 23 > x^2 [1] 1 16 49 4 81	 > prod(x) [1] 504 > x/2 [1] 0.5 2.0 3.5 1.0 4.5
Số tổ hợp chập k của n > choose(n,k)	Số hoán vị của n phần tử > prod(1:n)

Ma trận

```
matrix(data = NA, nrow = 1, ncol = 1, byrow = FALSE)
```

Trong đó

data: vector dữ liệu

nrow: số dòng

ncol: số cột

byrow: nếu FALSE (TRUE) các phần tử trong data được xếp theo các cột

Cho ma trận A sau đây

$$A = \begin{pmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{pmatrix}$$

1.1. Các phép toán

Ma trận A được tạo trong R như sau

```
> y=c(1,2,3,4,5,6,7,8,9)
```

```
> A=matrix(y, nrow=3)
```

```
> A
```

	[,1]	[,2]	[,3]
[,1]	1	4	7
[,1]	2	5	8
[,1]	3	6	9

Nhưng

```
> A=matrix(y, nrow=3, byrow=T)
```

```
> A
```

	[,1]	[,2]	[,3]
[,1]	1	2	3
[,2]	4	5	6
[,3]	7	8	9

```
> B= t(A)
```

Ma trận chuyển vị

```
> B
```

	[,1]	[,2]	[,3]
[,1]	1	2	3
[,2]	4	5	6
[,3]	7	8	9

```
> A+B
```

Cộng hai ma trận

	[,1]	[,2]	[,3]
[1,]	2	6	10
[2,]	6	10	14
[3,]	10	14	18

```
> A-B
```

Trừ hai ma trận

	[,1]	[,2]	[,3]
[1,]	0	2	4
[2,]	-2	0	2
[3,]	-4	-2	0

```
> A*2
```

Nhân ma trận với một số

	[,1]	[,2]	[,3]
[1,]	2	8	14
[2,]	4	10	16
[3,]	6	12	18

```
> A%*%B
```

Nhân hai ma trận

```

      [,1] [,2] [,3]
[1,]    66    78    90
[2,]    78    93   108
[3,]    90   108   126
> det(A)                                # Định thức
[1] 0
> x=c(1,0,2,3,5,4,-1,-5,2)
> F=matrix(x, nrow=3)
> solve(F)                              # Nghịch đảo của ma trận vuông khả
                                         nghịch

      [,1] [,2] [,3]
[1,]     3 -1.0 -1.0
[2,]    -1  0.4  0.5
[3,]    -1  0.2  0.5

```

1.2 Dữ liệu

Dữ liệu là một vector

Nhập dữ liệu (bằng c() hoặc scan())

Giả sử sau đây là số lỗi chính tả trong 10 trang sách đầu tiên của một cuốn sách

2 3 0 3 1 0 0 0 2 1

Dữ liệu trên được nhập vào R như sau

```

> SoLoi=c(2,3,0,3,1,0,0,0,2,1)
> SoLoi
[1] 2 3 0 3 1 0 0 0 2 1
Hoặc
> SoLoi=scan()
1: 2 3 0 3 1 0 0 0 2 1
11:
Read 10 items
> SoLoi
[1] 2 3 0 3 1 0 0 0 2 1
> is.vector(SoLoi)
[1] TRUE

```

Biên tập dữ liệu

Thay đổi giá trị của vector

```
> SoLoi1=SoLoi           # tạo một bản copy từ SoLoi
> SoLoi1[1]=1             # sửa số lỗi trong trang đầu tiên
                             thành 1
> SoLoi1[c(2,4,6)]=c(1,2,3) # sửa số lỗi trong trang 2,4,6
```

Thêm giá trị vào vector

```
> SoLoi1[11:15]=c(1,2,0,3,1,1)
```

Ghép nhiều vector

Giả sử SoLoi2 là vector chứa dữ liệu về số lỗi của 9 trang sách tiếp theo

```
> SoLoi3=c(SoLoi1,SoLoi2) # số lỗi trong 24 trang sách
                             đầu tiên
> z= c(x1, x2,... , xn)    # ghép các vector x1, x2,
                             ... ,xn theo thứ tự đó
```

Truy cập dữ liệu

Cho x là một vector

x có bao nhiêu phần tử	> length(x)
phần tử thứ i	> x[i]
mọi phần tử trừ phần tử thứ i	> x[-i]
k phần tử đầu	> x[1:k]
k phần tử cuối	> x[length(x)-k+1:length(x)]
phần tử ở vị trí i, j, k	> x[c(i,j,k)]
những phần tử lớn hơn α	> x[x> α]
vị trí của chúng	> which(x== α)
những phần tử lớn hơn α và nhỏ hơn β	> x[x> α & x< β]
vị trí của phần tử lớn nhất	> which(x==max(x))
vị trí đầu tiên của phần tử lớn nhất	> which.max(x)

Loại đi số liệu trống không

Giả sử x là một vector có chứa số liệu trống không (NA)

```
> x=c(1,2,3,4,5,NA)
```

Để loại bỏ số liệu trống không ra khỏi x ta làm như sau

```
> x1=na.omit(x)
```

```
> x1
```

```
[1] 1 2 3 4 5
```

Mã hóa số liệu

Giả sử ThuNhap là vector chỉ thu nhập một năm của 6 hộ gia đình. Thu nhập sẽ được chia làm 3 mức: thấp (dưới 35), trung bình (35 đến 50), cao (trên 50).

```
> ThuNhap=c(40,28,36,39,52,33)
```

```
> Muc=ThuNhap
```

```
> Muc[ThuNhap<35]="Thap"
```

```
> Muc[ThuNhap>=35&ThuNhap<=50]="TB"
```

```
> Muc[ThuNhap>50]="Cao"
```

```
> data.frame(ThuNhap,Muc)
```

	ThuNhap	Muc
1	40	TB
2	28	Thap
3	36	TB
4	39	TB
5	52	Cao
6	33	Thap

Biến đổi thành yếu tố

```
> Muc=factor(Muc)
```

```
> Muc
```

```
[1] TB Thap TB TB Cao Thap
```

```
Levels: cao TB thap
```

Dữ liệu là một data frame

Nhập dữ liệu

Giả sử ta có một tập dữ liệu về chiều cao, cân nặng, giới tính của 5 người như sau

ChieuCao	CanNang	GioiTinh
170	65	Nam
165	50	Nam
150	45	Nu
158	60	Nu
166	53	Nam

Dữ liệu trên có thể lưu trong một data frame bằng cách

```
> ChieuCao=c(170,165,150,158,166)
> CanNang= c(65,50,45,60,53)
> GioiTinh=c("Nam","Nam","Nu","Nu","Nam")
> DuLieu= data.frame(ChieuCao,CanNang,GioiTinh)
> DuLieu
  ChieuCao CanNang GioiTinh
1      170      65      Nam
2      165      50      Nam
3      150      45       Nu
4      158      60       Nu
5      166      53      Nam
> is.data.frame(DuLieu)
[1] TRUE
> DuLieu= data.frame(c1=ChieuCao,c2=CanNang,c3=GioiTinh)
> row.names(DuLieu)=c("Vu","Hieu","Hoa","Thuy","Nam")
> colnames(DuLieu)=c("cc","cn","gt")
      cc  cn  gt
Vu  170  65  Nam
Hieu 165  50  Nam
Hoa  150  45   Nu
Thuy 158  60   Nu
Nam  166  53  Nam
```

Ta có thể nhập dữ liệu nói trên bằng cách sử dụng lệnh `edit(data.frame())`

```
> DuLieu=edit(data.frame())
```

Một bảng gồm nhiều cột và nhiều dòng giống như một bảng của Excel sẽ xuất hiện để ta nhập dữ liệu. Dữ liệu ứng với từng biến sẽ được nhập theo một cột, ta có thể đặt tên tùy ý hoặc theo mặc định. Ta có thể lưu data frame DuLieu trên dưới dạng R để dùng sau này như sau

```
> setwd("D:/ThucHanhR")           # chọn địa chỉ để lưu giữ liệu
> save(DuLieu,file="DuLieu.rda")  # lưu data frame DuLieu vào một file
                                   có tên "DuLieu.rda"
```

Khi cần sử dụng ta dùng lệnh load

```
> setwd("D:/ThucHanhR")           # truy nhập vào địa chỉ lưu dữ liệu
> load("DuLieu.rda")              # nhập dữ liệu từ file "diemthi.rda"
```

Truy cập dữ liệu

Giả sử ta đã có một data frame

```
> DuLieu
  ChieuCao CanNang GioiTinh
1      170      65      Nam
2      165      50      Nam
3      150      45       Nu
4      158      60       Nu
5      166      53      Nam
```

```
> names(DuLieu)                   # tên các cột
> dim(DuLieu)                     # số dòng, số cột
```

```
> ChieuCao
```

Error: object "ChieuCao" not found

Để truy cập vào các cột dữ liệu của data frame ta cần dùng lệnh attach

```
> attach(DuLieu)
```

```
> ChieuCao
```

```
[1] 170 165 150 158 166
```

Ta cũng có thể sử dụng toán tử \$ như sau

```
> DuLieu$ChieuCao
```

```
[1] 170 165 150 158 166
```

Mỗi data frame là một bảng gồm các dòng và các cột, để truy cập vào một giá trị ở dòng hay cột nào đó ta tiến hành như sau

```
> DuLieu[, 'ChieuCao']           # cột ChieuCao
[1] 170 165 150 158 166

> DuLieu[, 1]                    # cột thứ nhất
> DuLieu[, c(1, 2)]              # cột thứ nhất và cột thứ hai
> DuLieu[1,]                     # dòng thứ nhất
  ChieuCao CanNang GioiTinh
1      170      65      Nam

> DuLieu[1, 3]                   # giới tính của người đầu tiên
[1] nam
```

Để lấy ra thông tin về những người có giới tính nữ

```
> DuLieu[DuLieu$GioiTinh=="Nu",]
  ChieuCao CanNang GioiTinh
3       150      45      Nu
4       158      60      Nu
```

hoặc

```
> subset(DuLieu, GioiTinh=="Nu")
```

Để lấy thông tin về những người có cân nặng trên 50

```
> subset(DuLieu, CanNang>50)
  ChieuCao CanNang GioiTinh
1       170      65      Nam
4       158      60      Nu
5       166      53      Nam
```

Để kết thúc làm việc với dataframe ta dùng lệnh

```
> detach(DuLieu)
```

Thêm vào dataframe một số dòng, cột

Chẳn hạn ta có thêm thông tin về quê quán và tuổi của 5 người nói trên. Để ghép thêm 2 biến này vào dataframe DuLieu ta làm như sau

```
> QueQuan=c("HaNoi", "QuangNinh", "NgheAn", "NamDinh", "LangSon")
> Tuoi=c(24,27,30,35,28)
> DuLieu=data.frame(DuLieu, QueQuan, Tuoi)
```

Khi đó ta có một dataframe mới như sau

```
> DuLieu
  ChieuCao CanNang GioiTinh  QueQuan  Tuoi
1       170      65      Nam    HaNoi   24
2       165      50      Nam QuangNinh  27
3       150      45      Nu    NgheAn   30
4       158      60      Nu    NamDinh  35
5       166      53      Nam    LangSon  28
```

Hoặc

```
> DuLieu[,4:5]=c(QueQuan, Tuoi)
> DuLieu
```

	ChieuCao	CanNang	GioiTinh	V4	V5
1	170	65	Nam	HaNoi	24
2	165	50	Nam	QuangNinh	27
3	150	45	Nu	NgheAn	30
4	158	60	Nu	NamDinh	35
5	166	53	Nam	LangSon	28

Để đổi tên hai cột cuối

```
> names(DuLieu)[,4:5]=c("QueQuan", "Tuoi")
```

```
> DuLieu
```

	ChieuCao	CanNang	GioiTinh	QueQuan	Tuoi
1	170	65	Nam	HaNoi	24
2	165	50	Nam	QuangNinh	27
3	150	45	Nu	NgheAn	30
4	158	60	Nu	NamDinh	35
5	166	53	Nam	LangSon	28

Để thêm một số dòng vào dataframe ta làm tương tự như đối với cột

```
> a=c(150,48,"Nu","CaoBang",29)
```

```
> DuLieu[6,]=a
```

```
> DuLieu
```

	ChieuCao	CanNang	GioiTinh	QueQuan	Tuoi
1	170	65	Nam	HaNoi	24
2	165	50	Nam	QuangNinh	27
3	150	45	Nu	NgheAn	30
4	158	60	Nu	NamDinh	35
5	166	53	Nam	LangSon	28
6	150	48	Nu	CaoBang	29

Thay đổi dữ liệu trong dataframe

Bằng cách truy nhập vào dữ liệu tại dòng thứ i, cột thứ j của dataframe ta có thể sửa đổi thông tin của một dòng, một cột hay một ô nào đó.

```
> DuLieu[6,5]=39
```

```
> DuLieu
```

	ChieuCao	CanNang	GioiTinh	QueQuan	Tuoi
1	170	65	Nam	HaNoi	24
2	165	50	Nam	QuangNinh	27
3	150	45	Nu	NgheAn	30
4	158	60	Nu	NamDinh	35
5	166	53	Nam	LangSon	28
6	150	48	Nu	CaoBang	39

Loại bỏ số liệu trống không

Tương tự trường hợp vector, dùng lệnh `na.omit()` để loại đi số liệu trống không (NA).

```
> na.omit(DuLieu)
```

Chú ý: Đối với data frame được nhập sau lệnh `edit(data.frame())`, mọi thao tác sửa đổi, thêm bớt dữ liệu có thể gọi bảng dữ liệu ra và sửa trực tiếp trên đó.

Chẳng hạn ta đã tạo ra một data frame như sau

```
> BaiTap = edit(data.frame())
```

Để thay đổi, thêm bớt số liệu của BaiTap ta dùng lệnh

```
> BaiTap1 = edit(data.frame(BaiTap))
```

Khi đó data frame BaiTap sẽ xuất hiện trở lại, ta tiến hành những thao tác biên tập số liệu theo ý muốn. Dữ liệu sau khi biên tập được gán vào một data frame mới có tên BaiTap1.

1.3 Tạo dữ liệu

Tạo dãy số bằng seq

```
seq(from, to, by), seq(length = , from = , by = ) hoặc  
seq(length = , from = , to = )
```

Trong đó

from, to giá trị bắt đầu, kết thúc của dãy số

by khoảng cách giữa các số

length số phần tử của dãy số

```
> seq(from=1,to=5,by=1) # tạo một vector số tự nhiên từ 1  
đến 5
```

```
[1] 1 2 3 4 5
```

```
> 1:5
```

```
[1] 1 2 3 4 5
```

```
> seq(2,5,0.5) # tạo một vector số từ 2 đến 5 khoảng  
cách 0.5
```

```
[1] 2 2.5 3 3.5 4 4.5 5
```

```
> seq(length=8,from=4,by=3)           # tạo một vector gồm 8 số từ 4 với
                                         khoảng cách 3

[1] 4 7 10 13 16 19 22 25
> seq(length=5,from=4,to=6)           # tạo một vector gồm 8 số cách đều
                                         từ 4 đến 6

[1] 4.0 4.5 5.0 5.5 6.0
```

Tạo dãy số lặp lại bằng rep

`rep(x, times)`

Trong đó

x: một vector các giá trị
times: số lần lặp lại

```
> rep(8,5)                             # tạo số 8 lặp 5 lần
[1] 8 8 8 8 8
> rep(1:4,3)                           # dãy số 1,2,3,4 lặp lại 3 lần
[1] 1 2 3 4 1 2 3 4 1 2 3 4
> rep(1:4,each=3)                      # dãy số 1,2,3,4, mỗi số lặp lại 3
                                         lần
[1] 1 1 1 2 2 2 3 3 3 4 4 4
> rep(c(1,3,5),c(2,3,4))              # 1 lặp 2 lần, 3 lặp 3 lần, 5 lặp 4
                                         lần
[1] 1 1 3 3 3 5 5 5 5
```

Tạo biến thứ bậc bằng gl

`gl(n ,k ,length = ,labels = ,ordered =)`

Trong đó

n: một số nguyên dương cho biết số bậc
k: số lần lặp lại của mỗi bậc
length: chiều dài của kết quả
labels: một vector các nhãn gán cho tên của các bậc
ordered: T/F các bậc có/không xếp thứ tự các bậc

```
> gl(2,5)                             # tạo biến gồm bậc 1, 2 mỗi bậc lặp
                                         lại 5 lần
[1] [1] 1 1 1 1 1 2 2 2 2 2
```

1.4. Nhập dữ liệu từ các file dữ liệu

```
Levels: 1 2
> gl(2,5,length=12)                                # tạo biến gồm bậc 1, 2 mỗi bậc lặp
                                                    lại 5 lần, chiều dài 12

[1] [1] 1 1 1 1 1 2 2 2 2 2 1 1
Levels: 1 2
> gl(3,2,length=10,labels=c("a","b","c"))
[1] a a b b c c a a b b                                # tạo biến gồm bậc a, b, c mỗi bậc
                                                    lặp lại 2 lần, chiều dài 10

Levels: a b c
```

1.4 Nhập dữ liệu từ các file dữ liệu

Nhập dữ liệu từ một text file

Giả sử tại địa chỉ D:/ThucHanhR có một file dữ liệu tên là DiemThi.txt. Để nhập dữ liệu vào R ta dùng lệnh read.table

```
> setwd("D:/ThucHanhR")                            #truy nhập vào địa chỉ lưu giữ liệu
> DiemThi=read.table("DiemThi.txt",header=T)
                                                    # nhập dữ liệu từ file DiemThi.txt
                                                    vào đối tượng tên là DiemThi, dòng
                                                    đầu tiên là tên các cột dữ liệu
```

Nhập dữ liệu từ Excel

Giả sử tại địa chỉ D:/ThucHanhR có một file dữ liệu tên là DiemThi.xls. Để nhập dữ liệu vào R, trước hết ta lưu file dưới dạng đuôi .csv

- Vào Excel, chọn **File**, chọn **Save as**
- Chọn **Save as type "CSV (Comma delimited)"**

Khi đó ta có một file tên là DiemThi.csv tại địa chỉ D:/ThucHanhR, để nhập dữ liệu vào R ta làm như sau

```
> setwd("D:/ThucHanhR")                            # truy nhập vào địa chỉ lưu giữ liệu
> DiemThi=read.csv("DiemThi.csv",header=T)
                                                    # nhập dữ liệu từ file DiemThi.sav
                                                    vào đối tượng tên là DiemThi, dòng
                                                    đầu tiên là tên các cột dữ liệu
```

Nhập dữ liệu từ SPSS

Giả sử tại địa chỉ D:/ThucHanhR có một file dữ liệu tên là DiemThi.sav. Để nhập dữ liệu vào R, ta làm như sau

```
> library(foreign)           # sử dụng gói foreign
> setwd("D:/ThucHanhR")     # truy nhập vào địa chỉ lưu giữ liệu
> DiemThi=read.spss("DiemThi.sav",to.data.frame=T)
                                # nhập dữ liệu từ file DiemThi.csv
                                vào một data frame tên là DiemThi
```

Trong tất cả các trường hợp trên, sau khi nhập dữ liệu, ta có thể lưu dữ liệu trong đối tượng DiemThi dưới dạng R để xử lý sau này bằng lệnh save

```
> setwd("D:/ThucHanhR")     # chọn địa chỉ để lưu giữ liệu
> save(DiemThi,file="DiemThi.rda")
                                # lưu dữ liệu từ đối tượng DiemThi
                                vào một file có tên "DiemThi.rda"
```

Khi cần sử dụng ta dùng lệnh load

```
> setwd("D:/ThucHanhR")     # truy nhập vào địa chỉ lưu giữ liệu
> load("DiemThi.rda")       # nhập dữ liệu từ file "DiemThi.rda"
```

1.5 Chọn mẫu ngẫu nhiên

```
sample(x ,size ,replace = ,prob = )
```

Trong đó

x: một vector gồm những phần tử được chọn, hoặc một số nguyên dương
 size: cỡ mẫu
 replace: T/F lần mẫu có hoàn lại/ không hoàn lại
 prob: một vector cho biết xác suất được chọn của những phần tử trong x

Giả sử ta cần chọn ngẫu nhiên ra 10 phần tử từ tổng thể gồm 200 người được đánh số từ 1 tới 200. Để làm việc đó ta dùng lệnh sample

```
> sample(1:200, 10)
[1] 176 173 195 80 5 66 91 64 98 43
> sample(1:200, 10)
```


1.6. Bài tập

```
[1] 28 7 101 29 188 192 71 75 19 6
> sample(1:200, 10, replace=T) # chọn mẫu có hoàn lại
[1] 130 4 138 35 110 127 4 7 146 118
```

Giả sử ta có một hộp gồm 3 quả bóng: xanh (X), đỏ (D), vàng (V). Tiến hành chọn (có hoàn lại) ngẫu nhiên 5 quả bóng, kết quả có thể là

```
> sample(c("X", "D", "V"),
5, replace=T)
[1] "V" "X" "X" "V" "X"
```

Chọn ra 5 phần tử từ tổng thể gồm 3 phần tử mà xác suất được chọn của chúng lần lượt là 0.5, 0.3 0.2

```
> sample(3, 5, prob=c(0.5,0.3,0.2), replace=T)
[1] 2 2 2 1 2
> sample(4, 3, prob=c(0.4,0.3,0.2,0.1), replace=F)
[1] 2 1 3
```

1.6 Bài tập

BÀI TẬP

Bài tập 1 Giả sử bạn theo dõi số tiền trong hóa đơn điện thoại hàng tháng theo thứ tự từ tháng 1 đến tháng 12 trong năm vừa qua như sau (đơn vị nghìn đồng)

198, 185, 223, 221, 207, 203, 180, 195, 222, 177, 214, 216

1. Tính tổng số tiền bạn phải trả cho phí điện thoại trong năm đó.
2. Cho biết tháng nào có số tiền ít (nhiều) nhất, và số tiền là bao nhiêu?
3. Cho biết những tháng bạn phải trả hơn 200 nghìn tiền điện thoại. Có bao nhiêu tháng như thế.
4. Nhập thêm vào dãy trên tiền điện thoại của 3 tháng tiếp theo nhưng bạn quên mất số tiền tháng thứ 13, biết tháng thứ 14, 15 số tiền là 201, 185. Sau đó tính số tiền trung bình bạn phải trả mỗi tháng (dùng hàm mean).

Bài tập 2 Cho hai tập dữ liệu dạng vectơ $x = c(1,3,5,7,9)$ và $y = c(1,2,8,6,4,5,7)$ thực hiện các thao tác sau

1. $x + 1$, $y*2$, $\text{length}(x)$, $\text{length}(y)$, $x + y$
2. $\text{sum}(x > 5)$ và $\text{sum}(x[x > 5])$

Bài tập 3 Chọn ngẫu nhiên 5 người từ danh sách gồm 40 người.

Bài tập 4 Tung một đồng xu 50 lần. Mô phỏng phép thử và đếm số mặt sấp.

Bài tập 5 Tung một con xúc sắc 100 lần. Mô phỏng phép thử và đếm số lần xuất hiện mặt 1 chấm.

Bài tập 6 Chọn ngẫu nhiên năm cây bài từ bộ bài tứ lơ khơ. Mô phỏng phép thử và kiểm tra xem có bộ đôi nào trong mẫu không? Hãy lập lại cho đến khi có được một đôi trong 5 cây bài.

Bài tập 7 Dưới đây là thông tin về 8 sinh viên mới ra trường của một khóa học

Thứ tự	Lương khởi điểm	Giới tính	Xếp loại tốt nghiệp	Tuổi
1	6	Nam	K	22
2	5	Nu	K	25
3	4.5	Nam	TB	23
4	3.8	Nu	K	22
5	8	Nu	G	22
6	12	Nam	G	23
7	4	Nam	TB	22
8	5	Nu	TB	24

1. Nhập dữ liệu vào một data frame gồm các cột TT, Luong, GioiTinh, Tot-Nghiep, Tuoi
2. Đưa ra dữ liệu về những sinh viên nữ.
3. Đưa ra danh sách những sinh viên có lương khởi điểm trên 6 triệu/tháng.
4. Cho biết những thông tin về người có lương cao nhất trong danh sách.
5. Thêm vào danh sách một sinh viên nam tốt nghiệp xếp loại giỏi, lương khởi điểm 7.5 triệu nhưng không có thông tin về tuổi.
6. Loại đi số liệu trống không trong data frame. Nhận xét.

Bài tập 8

1. Tạo dãy số từ 1 đến 100.
2. Tạo dãy số chẵn từ 0 đến 100.
3. Tạo dãy số trong đó 3 lặp 4 lần, 5 lặp 10 lần, 16 lặp 7 lần.
4. Tạo dãy số trong đó có các giá trị 1, 2, 3, 4 đều lặp lại 10 lần.

5. Tạo biến thứ bậc gồm 3 bậc, mỗi bậc lặp 4 lần.
6. Tạo biến thứ bậc gồm 3 bậc, số lần lặp lại tương ứng là 2, 5, 8 với ký hiệu a, b, c.

Bài tập 9

File dữ liệu ChieuCao.csv lưu dữ liệu về chiều cao và giới tính của 1000 người.

1. Chọn ngẫu nhiên 30 người từ danh sách và đưa ra thông tin về họ.
2. Lưu lại dữ liệu mẫu dưới dạng . rda.
3. Cho biết số nam giới và nữ giới trong mẫu.
4. Tính chiều cao trung bình của hai nhóm và so sánh.

Bài tập 10

xác định loại thang đo trong các trường hợp sau

1. Thời gian chờ thang máy của một người tại một khu chung cư.
2. Số khối nước một gia đình sử dụng trong một tháng.
3. Xếp hạng 5 chiếc máy trong nhà máy theo đánh giá: rất tốt, tốt, trung bình, kém.
4. Mã vùng điện thoại của các địa phương.
5. Tuổi của các nhân viên trong công ty.
6. Doanh thu (VN đồng) của một cửa hàng bán báo trong một tháng
7. Mã sinh viên trong một trường đại học.
8. Điểm thi một môn của sinh viên một lớp.
9. Chiều cao của một người.