ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH

TOÁN RỜI RẠC 1

Bài tập lớn đề tài MÃ ĐỀ

Thống kê & phân tích dữ liệu bằng R

GVHD: NNN
Mã nhóm: N
Nhóm SV: Nguyen Van A- 22102134
Tran Van B - 88471334

TP. HỒ CHÍ MINH, THÁNG 1/2015

Mục lục

Nhật ký làm việc nhóm

Nhật ký làm việc nhóm

Bài báo cáo này trình bày về thống kê và phân tích dữ liệu chiều cao của ca sĩ ở New York Choral Society năm 1979, được chia thành 4 cột lần lượt theo giọng nữ cao, nữ trầm, nam cao và nam trầm.

1 Giới thiệu bài toán

Ta cần phải phân tích dữ liệu để cung cấp các thông tin xác thực, trực quan, mô tả cụ thể, dễ hiểu vấn đề đang phân tích để phục vụ nghiên cứu khoa học. Đặc biệt trong các vấn đề kinh tế-xã hội và khi nghiên cứu số lớn chúng ta cần phải quan tâm đến các công cụ kỹ thuật về phân tích số liệu và biểu đồ.

Phân tích số liệu và biểu đồ thường được tiến hành bằng các phần mềm thông dụng như SAS, SPSS, Stata, Statistica, và S-Plus. Đây là những phần mềm được các công ti phần mềm phát triển và giới thiệu trên thị trường khoảng ba thập niên qua, và đã được các trường đại học, các trung tâm nghiên cứu và công ti kĩ nghệ trên toàn thếgiới sửdụng cho giảng dạy và nghiên cứu. Nhưng vì chi phí để sửdụng các phần mềm này tương đối đắt tiền (có khi lên đến hàng trăm ngàn đô-la mỗi năm). Do đó, các nhà nghiên cứu thống kê trên thế giới đã hợp tác với nhau để phát triển một phần mềm mới, với chủ trương mã nguồn mở, sao cho tất cảcác thành viên trong ngành thống kê học và toán học trên thếgiới có thể sử dụng một cách thống nhất và hoàn toàn miễn phí.

Năm 1996, trong một bài báo quan trọng vềtính toán thống kê, hai nhà thống kê học Ross Ihaka và Robert Gentleman [lúc đó] thuộc Trường đại học Auckland, New Zealand phát hoạmột ngôn ngữmới cho phân tích thống kê mà họ đặt tên là R. Nói một cách ngắn gọn, R là một phần mềm sử dụng cho phân tích thống kê và vẽ biểu đồ. Thật ra, về bản chất, R là ngôn ngữ máy tính đa năng, có thể sửdụng cho nhiều mục tiêu khác nhau, từ tính toán đơn giản, toán học giải trí (recreational mathematics), tính toán ma trận (matrix), đến các phân tích thống kê phức tạp. Vì là một ngôn ngữ, cho nên người ta có thể sử dụng R để phát triển thành các phần mềm chuyên môn cho một vấn đề tính toán cá biệt.

Sơ lược về đề tài : Phân tích đề tài chiều cao của các ca sĩ trong các hội hợp xướng New York vào năm 1979. Với giọng hát từ cao nhất đến thấp nhất với thứ tự Soprano, Alto, Tenor, Bass. Trong đó hai cột đầu tiên là giọng nữ còn hai cột sau là giọng nam. Các dữ liệu ban đầu bao gồm hai bộ phận cho từng phần. Bộ dữ liệu này chỉ báo cáo 1 Soprano, 1 Alto, 1 Tenor, 1 Bass. Với số trường hợp ban đầu là 39.

2 Cơ sở lý thuyết

2.1 Thống kê mô tả

Nói đến thống kê mô tả là nói đến việc mô tả dữ liệu bằng các phép tính và chỉ số thống kê thông thường mà chúng ta đã làm quen qua từ thuở trung học như số trung bình (mean), số trung vị (median), số lớn nhất (max), số nhỏ nhất (min), phương sai (variance), độ lệch chuẩn (standard deviation)...

Trong đó ta làm quen các định nghĩa chưa biết :

- Phương sai của một biến ngẫu nhiên là một độ đo sự phân tán thống kê của biến đó, nó hàm ý các giá trị của biến đó thường ở cách giá trị kỳ vọng bao xa.

- Độ lệch chuẩn, hay độ lệch tiêu chuẩn, là một đại lượng thống kê mô tả dùng để đo mức độ phân tán của một tập dữ liệu đã được lập thành bảng tần số. Có thể tính ra độ lệch chuẩn bằng cách lấy căn bậc hai của phương sai.

- số trung vị (tiếng Anh: median) là một số tách giữa nửa lớn hơn và nửa bé hơn của một mẫu, một quần thể, hay một phân bố xác suất. Nó là giá trị giữa trong một phân bố, mà số số nằm trên hay dưới con số đó là bằng nhau. Điều đó có nghĩa rằng 1/2 quần thể sẽ có các giá trị nhỏ hơn hay bằng số trung vị, và một nửa quần thể sẽ có giá trị bằng hoặc lớn hơn số trung vị.

2.2 Công cụ R

Như đã nói ở trên, R là một công cụ miễn phí dùng để phân tích dữ liệu. Chúng ta có thể sử dụng R để thực hiện các phép toán từ đơn giản đến phức tạp. Những bài toán tiêu biểu: các phép kiểm định thống kê, tính toán trên ma trận, hồi quy tuyến tính, gom cụm dữ liệu, bài toán phân lớp... Và vì R là một ngôn ngữ nên chúng ta có thể viết ứng dụng trên R để giải quyết các vấn đề cụ thể.

- Các hàm của R để tính toán thống kê mô tả:

```
> option (width=100)
# chuyển directory
> setwd ("C:/works/stats")

# đọc dữ liệu vào R
> igfdata <- read.table ("igf.txt", hearder = TRUE, na.string = ".")
> attach (igfdata)

# xem xét các cột số trong dữ liệu
> names (igfdata)
hoặc
> igfdata

# tính trung bình
> mean (age)

# phương sai và độ lệch chuẩn
> var (age)
> sd (age)
```

3 Kết quả phân tích dữ liệu

3.1 Tập dữ liệu

```
install.packages("readxl")     #Package đọc file .xlsx
library(readxl)                #Tải thư viện readxl

#----- Tập các file Route -----

setwd("E:/BTL_RR/data/RouteCell40") #Đến thư mục RouteCell40
RC40.names <- list.files(pattern = ".xlsx") #Lọc các file có đuôi .xlsx
rc40 <- lapply(RC40.names,read_excel) #Đọc các file excel
R.names <- {} #Tạo vector trống
for (i in 1:length(RC40.names)){
if ((as.integer(rc40[[i]][2,1]) <= 117) & (as.integer(rc40[[i]][2,1]) >= 57)){
```

```
R.names<- c(R.names, RC40.names[i])
}
} #Lọc lấy các file có RouteId từ 57 đến 117, lấy tên các file thỏa yêu cầu
RC40 <- lapply(R.names,read_excel) #Đọc các file thỏa yêu cầu đề
for (i in 1:length(RC40))
(RC40[[i]]<- na.omit(RC40[[i]])) #Xóa các hàng có giá trị NA
setwd("E:/BTL_RR/data/RouteGPS") #Đến thư mục RouteGPS
RGPS <- lapply(R.names,read_excel) #Đọc các file excel thỏa yêu cầu đề bài
for (i in 1:length(RGPS))
(RGPS[[i]]<- na.omit(RGPS[[i]])) #Xóa các hàng có giá trị NA
setwd("E:/BTL_RR/data/RouteCell60") #Đến thư mục RouteCell60
RC60 <- lapply(R.names,read_excel) #Đọc các file excel thỏa yêu cầu đề bài
for (i in 1:length(RC60))
(RC60[[i]]<- na.omit(RC60[[i]])) #Xóa các hàng có giá trị NA
#----- Chuyển Polyline sang Cell -----
#Cell60
install.packages("stringr") #Cài package stringr
library("stringr") #Tải thư viện stringr
for (i in 1:length(RC60)){
(RC60[[i]]$Polyline<-sapply(str_extract_all(RC60[[i]]$Polyline,"\\d+\\.*\\d*")
,as.numeric)) #Tách chuỗi dài trong Polyline thành các số riêng biệt
for (k in 1:nrow(RC60[[i]])){
for (j in 1:length(RC60[[i]]$Polyline[[k]])){
if (j%%2 != 0)
(RC60[[i]]$Polyline[[k]][j]<- floor((((RC60[[i]]$Polyline[[k]][j]-106.309795))
*10^5)/60)) #Vị trí lẻ được xử lí thành Lng
else (RC60[[i]]$Polyline[[k]][j]<- floor((((11.175186-RC60[[i]]$Polyline[[k]][j]))
*10^5)/60)) #Vị trí chẵn được xử lí thành Lat
}
}
}
for (i in 1:length(RC60)){
for (k in 1:nrow(RC60[[i]])){
RC60[[i]]$Polyline[[k]]<-c(RC60[[i]]$Polyline[[k]],RC60[[i]]$Lng[k],RC60[[i]]$
Lat[k]) #Lấy thêm các giá trị Lat Lng của các bến vào trong cột Polyline
}
}
for (i in 1:length(RC60)){
for (k in 1:nrow(RC60[[i]])){
RC60[[i]]$Polyline[[k]]<-RC60[[i]]$Polyline[[k]][RC60[[i]]$Polyline[[k]]>0]
}
} #Bỏ các điểm nhập liệu sai
for (i in 1:length(RC60)){
for (k in 1:nrow(RC60[[i]])){
a<-{}
for (j in 1:length(RC60[[i]]$Polyline[[k]])){
if (j%%2 !=0)
(a<-c(a,paste(RC60[[i]]$Polyline[[k]][j],RC60[[i]]$Polyline[[k]][j+1],sep = ",")))
}
```

```
RC60[[i]]$Polyline[[k]]<-a}} #Chuyển các tọa độ Lng Lat riêng lẻ thành tọa độ dạng Lng,Lat
#Cell40
for (i in 1:length(RC40)){
(RC40[[i]]$Polyline<-sapply(str_extract_all(RC40[[i]]$Polyline,"\\d+\\.*\\d*"),as.numeric))
for (k in 1:nrow(RC40[[i]])){
for (j in 1:length(RC40[[i]]$Polyline[[k]])){
if (j%%2 != 0)
(RC40[[i]]$Polyline[[k]][j]<- floor((((RC40[[i]]$Polyline[[k]][j]-106.309795))*10^5)/60))
else (RC40[[i]]$Polyline[[k]][j]<- floor((((11.175186-RC40[[i]]$Polyline[[k]][j]))*10^5)/60))
}
}
}
for (i in 1:length(RC40)){
for (k in 1:nrow(RC40[[i]])){
RC40[[i]]$Polyline[[k]]<-c(RC40[[i]]$Polyline[[k]],RC40[[i]]$Lng[k],RC40[[i]]$Lat[k])
}
}
for (i in 1:length(RC40)){
for (k in 1:nrow(RC40[[i]])){
RC40[[i]]$Polyline[[k]]<-RC40[[i]]$Polyline[[k]][RC40[[i]]$Polyline[[k]]>0]
}
}
for (i in 1:length(RC40)){
for (k in 1:nrow(RC40[[i]])){
a<-{}
for (j in 1:length(RC40[[i]]$Polyline[[k]])){
if (j%%2 !=0)
(a<-c(a,paste(RC40[[i]]$Polyline[[k]][j],RC40[[i]]$Polyline[[k]][j+1],sep = ",")))
}
RC40[[i]]$Polyline[[k]]<-a}} #Tương tự Cell40
#----- Đọc các file Journey -----

setwd("E:/BTL_RR/data/JourneyCell40") #Đến thư mục JourneyCell40
Jn40.names <- list.files(pattern = ".xlsx") #Chọn các file đuôi .xlsx
Jn.names <- {}
for (i in 1:length(Jn40.names)){
if((as.integer(gsub(".xlsx","",Jn40.names[i])) >=201) & (as.integer(gsub(".xlsx","",
Jn40.names[i])) <=670)) #Chọn các file có tên từ 201 đến 670
{Jn.names <- c(Jn.names, Jn40.names[i]) #Tạo vector chứa tên các file thỏa đề bài
}
}
Jn40 <- lapply(Jn.names, read_excel) #Đọc các file excel đó
for (i in 1:length(Jn40))
(Jn40[[i]]<- na.omit(Jn40[[i]])) #Loại bỏ các giá trị NA
setwd("E:/BTL_RR/data/JourneyGPS") #Đến thư mục JourneyGPS
JnGPS <- lapply(Jn.names, read_excel) #Đọc các file excel thỏa đề bài
for (i in 1:length(JnGPS))
(JnGPS[[i]]<- na.omit(JnGPS[[i]])) #Loại bỏ các giá trị NA
setwd("E:/BTL_RR/data/JourneyCell60") #Đến thư mục JourneyCell60
```

```
Jn60 <- lapply(Jn.names,read_excel) #Đọc các file excel thỏa đề bài
for (i in 1:length(Jn60))
(Jn60[[i]]<- na.omit(Jn60[[i]])) #Loại bỏ các giá trị NA

#----- Lưu các dataset thỏa đề bài -----

setwd("E:/BTL_RR/Data")
save(RC40, file="RC40.rda")
save(RGPS, file="RGPS.rda")
save(Jn40, file="Jn40.rda")
save(JnGPS, file="JnGPS.rda")
save(Jn60,file="Jn60.rda")
save(RC60,file="RC60.rda")
```

3.2 Kết quả phân tích

- Câu 2:

```
load("E:/BTL_RR/data/Jn40.rda") #Mở file Jn40 để phân tích
length(Jn40) #Mỗi file Journey là 1 xe buýt nên độ dài vector Jn40 là số các xe
Kết quả:
[1] 470 #Vậy có 470 xe buýt trong tập mẫu
```

- Câu 3:

```
load("E:/BTL_RR/data/RC60.rda") #Mở file RC60 để phân tích
length(RC60) #Mỗi file Route là 1 tuyến xe buýt nên số tuyến là độ dài vector RC60
Kết quả:
[1] 37 #Vậy có 37 tuyến xe buýt trong tập mẫu
```

- Câu 4:

```
#Xác định file Route cần xét thỏa yêu cầu đề bài
load("E:/BTL_RR/data/RC60.rda") #Mở file RC60 để phân tích
RouteId<- {}
for (i in 1:length(RC60))
(RouteId<- c(RouteId, as.integer(RC60[[i]][2,1]))) #Tạo vector với các giá trị RouteId
của các file
nearestId<- RouteId[which(abs(RouteId-67)==min(abs(RouteId-67)))] #Lấy các file có giá trị
gần 67 nhất
nearestId<-nearestId[which(nearestId==min(nearestId))] #Nếu có nhiều file thì sẽ lấy file có
RouteId nhỏ nhất
for (i in 1:length(RC60)){
if (as.integer(RC60[[i]][2,1])==nearestId)
(RC60in4<- RC60[[i]])
} #Chọn file có RouteId thỏa yêu cầu đề bài là 67
```

- Câu 4a:

```
a<-{}
for (i in 1:nrow(RC60in4)){
a<-c(a,RC60in4$Polyline[[i]]) #Tạo vector với các giá trị Lng,Lat của RouteId 67
}
length(unique(a)) #Đọ dài của vector các giá trị khác nhau của Lng,Lat chính là số Cell khác
nhau mà tuyến đó đi qua
Kết quả:
[1] 308 #Vậy có 308 Cell khác nhau mà tuyến này đi qua
```

- Câu 4b:

```
sum(as.numeric(RC60in4$Distance)) #Tổng các giá trị trong cột Distance là tổng quãng đường mà
tuyến đi qua
Kết quả:
[1] 36833 #Vậy quãng đường tuyến đó đi là 36833
```

- Câu 4c:

```
VectorViTriCacGiaTriQuaNhieuLan<- which(table(a)>1) #Tạo vector với các cell xuất hiện nhiều
hơn 1 lần
names(VectorViTriCacGiaTriQuaNhieuLan) #Liệt kê tên các vector đó
Kết quả:
    [1] "527,770" "530,768" "532,766" "533,753" "533,754" "533,765" "534,750"
    [8] "534,752" "534,764" "535,748" "535,749" "535,750" "535,751" "535,762"
    [15] "535,763" "536,746" "536,747" "536,750" "536,751" "536,752" "536,760"
    [22] "537,746" "537,752" "539,753" "540,751" "542,748" "543,749" "545,742"
    [29] "545,743" "546,742" "548,731" "548,738" "549,731" "549,733" "550,730"
    [36] "550,734" "550,736" "550,737" "551,729" "551,730" "551,735" "551,736"
    [43] "553,728" "554,728" "555,727" "556,726" "557,725" "558,725" "561,723"
    [50] "562,723" "567,720" "569,718" "571,717" "572,718" "575,718" "576,718"
    [57] "577,717" "577,718" "581,715" "583,715" "584,714" "585,715" "587,716"
    [64] "588,716" "590,716" "592,716" "594,715" "596,715" "597,716" "599,714"
    [71] "601,714" "602,714" "603,713" "604,713" "605,714" "606,713" "607,712"
    [78] "607,713" "608,712" "609,711" "610,711" "611,710" "612,709" "613,708"
    [85] "614,708" "616,708" "617,707" "619,707" "621,693" "621,694" "621,695"
    [92] "622,693" "622,695" "622,696" "622,710" "623,692" "623,699" "623,707"
    [99] "623,709" "623,711" "624,692" "624,700" "624,701" "624,702" "624,709"
    [106] "625,691" "625,692" "625,710" "627,687" "627,688" "628,685" "628,686"
    [113] "628,687" "630,680" "630,681" "630,682" "631,678" "631,679" "631,683"
    [120] "632,678" "634,675" "634,677" "638,672" "638,673" "638,695" "638,696"
    [127] "639,672" "639,673" "639,675" "639,694" "639,696" "639,697" "640,700"
    [134] "641,676" "641,677" "641,701" "642,678" "642,702" "642,703" "643,675"
    [141] "643,679" "643,691" "643,692" "643,703" "644,673" "644,676" "644,679"
    [148] "644,682" "644,691" "644,703" "645,689" "646,672" "646,674" "646,690"
    [155] "646,691" "647,685" "647,691" "647,692" "648,686" "648,691" "648,692"
    [162] "648,703" "649,688" "649,691" "650,688" "650,689" "650,692" "650,703"
    [169] "651,690" "651,703" "653,703"
#Đó là các cell tuyến đi qua nhiều lần
```

- Câu 4d:

```
mean(as.numeric(RC60in4$Distance)) #Giá trị trung bình của cột Distance là quãng đường trung
bình giữa các trạm trên tuyến
Kết quả:
[1] 454.7284 #Vậy khoảng cách trung bình ấy là 454.7284
```