#给数据分组#

setwd("C:/Users/lenovo/Desktop/实验室/新work/Python\_DBAD")

rm(list=ls())

dat=read.csv("data1\_1.csv")

x=dat[,1]

y=dat[,3]

z=dat[,2]

m=length(x)

n=table(x)

breaks<-seq(0,11,by=1)

fenduan<-cut(dat[,1],breaks=breaks)

fenduanshu<-table(fenduan)

x1<-dat[1:fenduanshu[1],]

x2<-dat[(1+sum(fenduanshu[0:1])):sum(fenduanshu[0:2]),]

x3<-dat[(1+sum(fenduanshu[0:2])):sum(fenduanshu[0:3]),]

x4<-dat[(1+sum(fenduanshu[0:3])):sum(fenduanshu[0:4]),]

x5<-dat[(1+sum(fenduanshu[0:4])):sum(fenduanshu[0:5]),]

x6<-dat[(1+sum(fenduanshu[0:5])):sum(fenduanshu[0:6]),]

x7<-dat[(1+sum(fenduanshu[0:6])):sum(fenduanshu[0:7]),]

x9<-dat[(1+sum(fenduanshu[0:8])):sum(fenduanshu[0:9]),]

x10<-dat[(1+sum(fenduanshu[0:9])):sum(fenduanshu[0:10]),]

x11<-dat[(1+sum(fenduanshu[0:10])):sum(fenduanshu[0:11]),]

#这里是求加速度的所有参数#

canshus<-function(x){

X<-x

breaks<-seq(0,1500\*217,by=1500)#每个人都有217组数据#

fenduan<-cut(X[,2],breaks=breaks)

fenduanshu<-table(fenduan)

n=length(fenduanshu)-1

for(j in 1:n){

M=X[(1+sum(fenduanshu[0:j])):sum(fenduanshu[0:j+1]),3]

library(mclust)

canshu=Mclust(M)

mean<-canshu$parameters$mean

var<-canshu$parameters$var$sigmasq

alpha<-canshu$parameters$pro;

cat('zu',j,'mean',mean,'var',var,'alpha',alpha,'\n')

}

}

canshus(x1)

canshus(x2)

canshus(x3)

canshus(x4)

canshus(x5)

canshus(x6)

canshus(x7)

canshus(x9)

canshus(x10)

canshus(x11)

#这里是求一个人第j组的加速度的所有参数#

canshu<-function(x,j){

#其中x为一个人的数据，这个人第j组数据#

X<-x

breaks<-seq(0,1500\*217,by=1500)#每个人都有217组数据#

fenduan<-cut(X[,2],breaks=breaks)

fenduanshu<-table(fenduan)

n=length(fenduanshu)-1

M=X[(1+sum(fenduanshu[0:j])):sum(fenduanshu[0:j+1]),3]

library(mclust)

canshu=Mclust(M)

mean<-canshu$parameters$mean

var<-canshu$parameters$var$sigmasq

alpha<-canshu$parameters$pro;

if(length(mean)>length(var))

var=rep(var,length(mean))

d=matrix(c(mean,var,alpha),ncol=3)

return(d)

}

#返回值第一列为miu，第二列为var，第三列为权重#

JD<-function(P,Q){

n=nrow(P)

m=nrow(Q)

P<-P

Q<-Q

p<-function(x){

h=0

for(i in 1:n){

as=1/sqrt(2\*pi\*P[i,2])

h=h+P[i,3]\*as\*exp(-(x-P[i,1])^2/(2\*P[i,2]))

return(h)

}

}

q<-function(x){

h=0

for(i in 1:m){

as=1/sqrt(2\*pi\*Q[i,2])

h=h+Q[i,3]\*as\*exp(-(x-Q[i,1])^2/(2\*Q[i,2]))

return(h)

}

}

jd<-function(x){

(p(x)-q(x))\*(log(p(x)/q(x)))

}

#这个地方我用的是重要函数进行蒙特卡罗积分#

x <- rnorm(100000,P[1,1],sqrt(P[1,2]))

fg <- jd(x) / dnorm(x,P[1,1],sqrt(P[1,2]))

answer<- mean(fg)

return(answer)

}

JD(canshu(x1,3),canshu(x2,3))

JD(canshu(x1,3),canshu(x3,3))

JD(canshu(x1,3),canshu(x4,3))

JD(canshu(x1,3),canshu(x5,3))

JD(canshu(x1,3),canshu(x6,3))

JD(canshu(x1,3),canshu(x7,3))

JD(canshu(x1,3),canshu(x9,3))

JD(canshu(x1,3),canshu(x10,3))

JD(canshu(x1,3),canshu(x11,3))

JD(canshu(x3,3),canshu(x2,3))

JD(canshu(x4,3),canshu(x2,3))

JD(canshu(x5,3),canshu(x2,3))

JD(canshu(x6,3),canshu(x2,3))

JD(canshu(x7,3),canshu(x2,3))

JD(canshu(x9,3),canshu(x2,3))

JD(canshu(x10,3),canshu(x2,3))

JD(canshu(x11,3),canshu(x2,3))

JD(canshu(x3,3),canshu(x4,3))

JD(canshu(x3,3),canshu(x5,3))

JD(canshu(x3,3),canshu(x6,3))

JD(canshu(x3,3),canshu(x7,3))

JD(canshu(x3,3),canshu(x9,3))

JD(canshu(x3,3),canshu(x10,3))

JD(canshu(x3,3),canshu(x11,3))

JD(canshu(x5,3),canshu(x4,3))

JD(canshu(x6,3),canshu(x4,3))

JD(canshu(x7,3),canshu(x4,3))

JD(canshu(x9,3),canshu(x4,3))

JD(canshu(x10,3),canshu(x4,3))

JD(canshu(x11,3),canshu(x4,3))

JD(canshu(x5,3),canshu(x6,3))

JD(canshu(x5,3),canshu(x7,3))

JD(canshu(x5,3),canshu(x9,3))

JD(canshu(x5,3),canshu(x10,3))

JD(canshu(x5,3),canshu(x11,3))

JD(canshu(x7,3),canshu(x6,3))

JD(canshu(x9,3),canshu(x6,3))

JD(canshu(x10,3),canshu(x6,3))

JD(canshu(x11,3),canshu(x6,3))

JD(canshu(x9,3),canshu(x7,3))

JD(canshu(x10,3),canshu(x7,3))

JD(canshu(x11,3),canshu(x7,3))

JD(canshu(x9,3),canshu(x10,3))

JD(canshu(x9,3),canshu(x11,3))

JD(canshu(x10,3),canshu(x11,3))

我选用第三个窗口数据产生的结果：

> JD(canshu(x1,3),canshu(x2,3))

[1] 0.04410861

> JD(canshu(x1,3),canshu(x3,3))

[1] 4.359635

> JD(canshu(x1,3),canshu(x4,3))

[1] 0.3726263

> JD(canshu(x1,3),canshu(x5,3))

[1] 0.07608305

> JD(canshu(x1,3),canshu(x6,3))

[1] 0.7842962

> JD(canshu(x1,3),canshu(x7,3))

[1] 0.5524851

> JD(canshu(x1,3),canshu(x9,3))

[1] 0.7730884

> JD(canshu(x1,3),canshu(x10,3))

[1] 1.590133

> JD(canshu(x1,3),canshu(x11,3))

[1] Inf

> JD(canshu(x3,3),canshu(x2,3))

[1] 1.073344

> JD(canshu(x4,3),canshu(x2,3))

[1] 0.353382

> JD(canshu(x5,3),canshu(x2,3))

[1] 0.07137275

> JD(canshu(x6,3),canshu(x2,3))

[1] 0.6576485

> JD(canshu(x7,3),canshu(x2,3))

[1] 0.6990901

> JD(canshu(x9,3),canshu(x2,3))

[1] 1.197401

> JD(canshu(x10,3),canshu(x2,3))

[1] 1.495439

> JD(canshu(x11,3),canshu(x2,3))

[1] 0.2987155

> JD(canshu(x3,3),canshu(x4,3))

[1] 1.067565

> JD(canshu(x3,3),canshu(x5,3))

[1] 2.245876

> JD(canshu(x3,3),canshu(x6,3))

[1] 0.7871747

> JD(canshu(x3,3),canshu(x7,3))

[1] 0.9149557

> JD(canshu(x3,3),canshu(x9,3))

[1] 1.493691

> JD(canshu(x3,3),canshu(x10,3))

[1] 1.148388

> JD(canshu(x3,3),canshu(x11,3))

[1] 60.72768

> JD(canshu(x5,3),canshu(x4,3))

[1] 0.6725218

> JD(canshu(x6,3),canshu(x4,3))

[1] 0.7629402

> JD(canshu(x7,3),canshu(x4,3))

[1] 0.17167

> JD(canshu(x9,3),canshu(x4,3))

[1] 0.4844043

> JD(canshu(x10,3),canshu(x4,3))

[1] 0.4156897

> JD(canshu(x11,3),canshu(x4,3))

[1] 0.1824567

> JD(canshu(x5,3),canshu(x6,3))

[1] 1.433218

> JD(canshu(x5,3),canshu(x7,3))

[1] 1.117576

> JD(canshu(x5,3),canshu(x9,3))

[1] 1.396359

> JD(canshu(x5,3),canshu(x10,3))

[1] 2.546446

> JD(canshu(x5,3),canshu(x11,3))

[1] Inf

> JD(canshu(x7,3),canshu(x6,3))

[1] 0.6970732

> JD(canshu(x9,3),canshu(x6,3))

[1] 1.219982

> JD(canshu(x10,3),canshu(x6,3))

[1] 1.778723

> JD(canshu(x11,3),canshu(x6,3))

[1] 0.2763933

> JD(canshu(x9,3),canshu(x7,3))

[1] 0.1155373

> JD(canshu(x10,3),canshu(x7,3))

[1] 0.2929987

> JD(canshu(x11,3),canshu(x7,3))

[1] 0.2734229

> JD(canshu(x9,3),canshu(x10,3))

[1] 0.4723046

> JD(canshu(x9,3),canshu(x11,3))

[1] Inf

> JD(canshu(x10,3),canshu(x11,3))

[1] Inf

>

目前问题：JD值现在算的时候还是会有inf出现，我个人认为是由于r语言的积分算的不是很好。