基于R语言的聚类分析

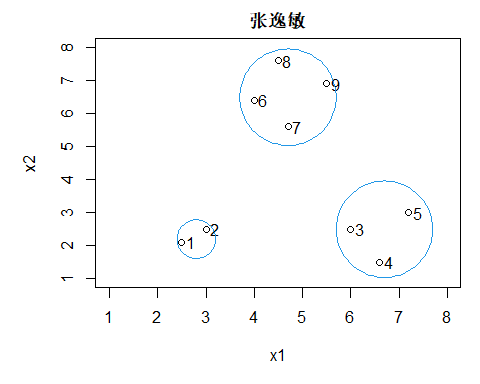
统计2001张逸敏

2023-03-26 21:57:33

## 聚类统计量

以下面一组数据为例，画出散点图，并根据图像粗略地画出圆圈进行分类

par(mar = c(5, 5, 2, 1))  
library(plotrix)  
x1 = c(2.5, 3.0, 6.0, 6.6, 7.2, 4.0, 4.7, 4.5, 5.5)  
x2 = c(2.1, 2.5, 2.5, 1.5, 3.0, 6.4, 5.6, 7.6, 6.9)  
plot(x1, x2, xlim = c(1, 8), ylim = c(1, 8),main="张逸敏")  
text(x1, x2, labels = c(1:9), adj = -0.5)  
draw.circle(x = 4.7, y = 6.5, r = 1, lty = 1, border = 4)  
draw.circle(x = 2.8, y = 2.2, r = 0.4, lty = 1, border = 4)  
draw.circle(x = 6.7, y = 2.5, r = 1, lty = 1, border = 4)



计算数据矩阵

X = cbind(x1, x2); #形成数据矩阵  
X

## x1 x2  
## [1,] 2.5 2.1  
## [2,] 3.0 2.5  
## [3,] 6.0 2.5  
## [4,] 6.6 1.5  
## [5,] 7.2 3.0  
## [6,] 4.0 6.4  
## [7,] 4.7 5.6  
## [8,] 4.5 7.6  
## [9,] 5.5 6.9

options(digits = 3) #设定有效位数  
dist(X) #默认为euclidean距离

## 1 2 3 4 5 6 7 8  
## 2 0.64   
## 3 3.52 3.00   
## 4 4.14 3.74 1.17   
## 5 4.79 4.23 1.30 1.62   
## 6 4.55 4.03 4.38 5.55 4.67   
## 7 4.13 3.54 3.36 4.52 3.61 1.06   
## 8 5.85 5.32 5.32 6.45 5.33 1.30 2.01   
## 9 5.66 5.06 4.43 5.51 4.25 1.58 1.53 1.22

dist(X, diag = TRUE) #添加主对角线距离

## 1 2 3 4 5 6 7 8 9  
## 1 0.00   
## 2 0.64 0.00   
## 3 3.52 3.00 0.00   
## 4 4.14 3.74 1.17 0.00   
## 5 4.79 4.23 1.30 1.62 0.00   
## 6 4.55 4.03 4.38 5.55 4.67 0.00   
## 7 4.13 3.54 3.36 4.52 3.61 1.06 0.00   
## 8 5.85 5.32 5.32 6.45 5.33 1.30 2.01 0.00   
## 9 5.66 5.06 4.43 5.51 4.25 1.58 1.53 1.22 0.00

dist(X, upper = TRUE) #添加上三角距离

## 1 2 3 4 5 6 7 8 9  
## 1 0.64 3.52 4.14 4.79 4.55 4.13 5.85 5.66  
## 2 0.64 3.00 3.74 4.23 4.03 3.54 5.32 5.06  
## 3 3.52 3.00 1.17 1.30 4.38 3.36 5.32 4.43  
## 4 4.14 3.74 1.17 1.62 5.55 4.52 6.45 5.51  
## 5 4.79 4.23 1.30 1.62 4.67 3.61 5.33 4.25  
## 6 4.55 4.03 4.38 5.55 4.67 1.06 1.30 1.58  
## 7 4.13 3.54 3.36 4.52 3.61 1.06 2.01 1.53  
## 8 5.85 5.32 5.32 6.45 5.33 1.30 2.01 1.22  
## 9 5.66 5.06 4.43 5.51 4.25 1.58 1.53 1.22

dist(X, diag = TRUE, upper = TRUE) #全矩阵距离

## 1 2 3 4 5 6 7 8 9  
## 1 0.00 0.64 3.52 4.14 4.79 4.55 4.13 5.85 5.66  
## 2 0.64 0.00 3.00 3.74 4.23 4.03 3.54 5.32 5.06  
## 3 3.52 3.00 0.00 1.17 1.30 4.38 3.36 5.32 4.43  
## 4 4.14 3.74 1.17 0.00 1.62 5.55 4.52 6.45 5.51  
## 5 4.79 4.23 1.30 1.62 0.00 4.67 3.61 5.33 4.25  
## 6 4.55 4.03 4.38 5.55 4.67 0.00 1.06 1.30 1.58  
## 7 4.13 3.54 3.36 4.52 3.61 1.06 0.00 2.01 1.53  
## 8 5.85 5.32 5.32 6.45 5.33 1.30 2.01 0.00 1.22  
## 9 5.66 5.06 4.43 5.51 4.25 1.58 1.53 1.22 0.00

dist(X, method = "manhattan") #manhattan距离

## 1 2 3 4 5 6 7 8  
## 2 0.9   
## 3 3.9 3.0   
## 4 4.7 4.6 1.6   
## 5 5.6 4.7 1.7 2.1   
## 6 5.8 4.9 5.9 7.5 6.6   
## 7 5.7 4.8 4.4 6.0 5.1 1.5   
## 8 7.5 6.6 6.6 8.2 7.3 1.7 2.2   
## 9 7.8 6.9 4.9 6.5 5.6 2.0 2.1 1.7

dist(X, method = "minkowski", p = 1) #明可夫斯基距离中p=1也是manhattan距离

## 1 2 3 4 5 6 7 8  
## 2 0.9   
## 3 3.9 3.0   
## 4 4.7 4.6 1.6   
## 5 5.6 4.7 1.7 2.1   
## 6 5.8 4.9 5.9 7.5 6.6   
## 7 5.7 4.8 4.4 6.0 5.1 1.5   
## 8 7.5 6.6 6.6 8.2 7.3 1.7 2.2   
## 9 7.8 6.9 4.9 6.5 5.6 2.0 2.1 1.7

dist(X, method = "minkowski", p = 2) #euclidean距离

## 1 2 3 4 5 6 7 8  
## 2 0.64   
## 3 3.52 3.00   
## 4 4.14 3.74 1.17   
## 5 4.79 4.23 1.30 1.62   
## 6 4.55 4.03 4.38 5.55 4.67   
## 7 4.13 3.54 3.36 4.52 3.61 1.06   
## 8 5.85 5.32 5.32 6.45 5.33 1.30 2.01   
## 9 5.66 5.06 4.43 5.51 4.25 1.58 1.53 1.22

D = dist(X);D #求距离阵

## 1 2 3 4 5 6 7 8  
## 2 0.64   
## 3 3.52 3.00   
## 4 4.14 3.74 1.17   
## 5 4.79 4.23 1.30 1.62   
## 6 4.55 4.03 4.38 5.55 4.67   
## 7 4.13 3.54 3.36 4.52 3.61 1.06   
## 8 5.85 5.32 5.32 6.45 5.33 1.30 2.01   
## 9 5.66 5.06 4.43 5.51 4.25 1.58 1.53 1.22

min(D) #求最小距离

## [1] 0.64

## 系统聚类法

### 最长距离法

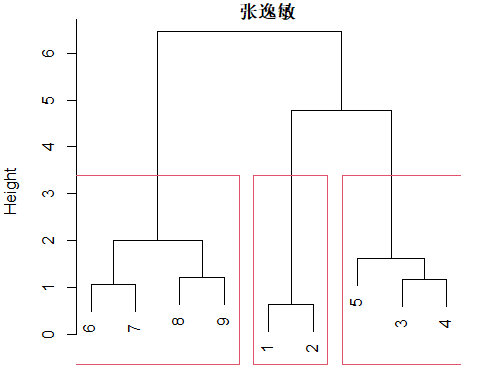
hc <- hclust(D);hc #默认最长距离法 D是欧氏距离矩阵

##   
## Call:  
## hclust(d = D)  
##   
## Cluster method : complete   
## Distance : euclidean   
## Number of objects: 9

cbind(hc$merge, hc$height) #分类过程

## [,1] [,2] [,3]  
## [1,] -1 -2 0.64  
## [2,] -6 -7 1.06  
## [3,] -3 -4 1.17  
## [4,] -8 -9 1.22  
## [5,] -5 3 1.62  
## [6,] 2 4 2.01  
## [7,] 1 5 4.79  
## [8,] 6 7 6.45

par(mar = c(1, 4, 1, 1))  
plot(hc,main="张逸敏") #聚类图  
rect.hclust(hc, 3) #加3分类框



cutree(hc, 1:3) #显示分类结果

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

从输出的分类过程可以看出，首先样品1和样品2形成新的类1，类间距离0.64；样品6和7形成新的类2，类间距离1.06；样品3和样品4形成新的类3，类间距离1.17；样品8和9形成新的类4，类间距离1.22；样品5和新的类3合成一类，形成新的类5，类间距离1.62；新的类2和新的类4合成一类，形成新的类6，类间距离2.01；新的类1和新的类5合成一类，形成新的类7，类间距离4.79；新的类6和新的类7合成一类，类间距离6.45，聚类结束。从输出的聚类图和分类框也可以清晰地看出聚类过程。

### 最短距离法

hc <- hclust(D, "single");hc #最短距离法

##   
## Call:  
## hclust(d = D, method = "single")  
##   
## Cluster method : single   
## Distance : euclidean   
## Number of objects: 9

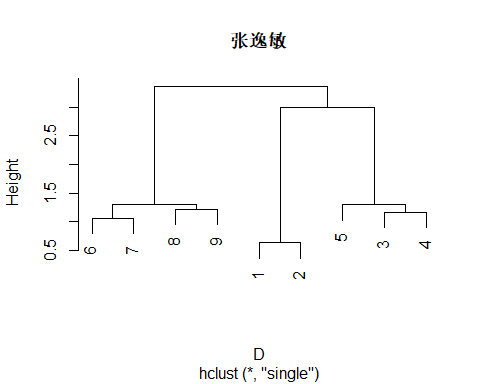
names(hc)

## [1] "merge" "height" "order" "labels" "method"   
## [6] "call" "dist.method"

cbind(hc$merge, hc$height) #分类过程

## [,1] [,2] [,3]  
## [1,] -1 -2 0.64  
## [2,] -6 -7 1.06  
## [3,] -3 -4 1.17  
## [4,] -8 -9 1.22  
## [5,] 2 4 1.30  
## [6,] -5 3 1.30  
## [7,] 1 6 3.00  
## [8,] 5 7 3.36

plot(hc,main="张逸敏") #聚类图

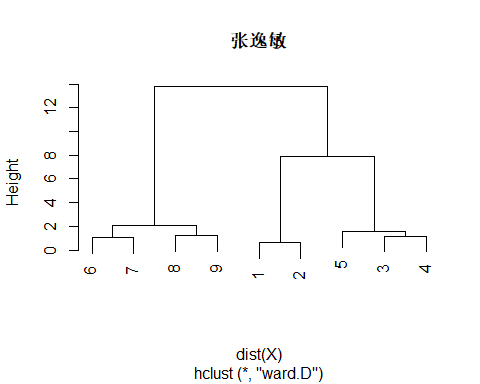


### 离差平方和法(Ward法)

hc <- hclust(dist(X), "ward.D") #ward距离法  
cbind(hc$merge, hc$height) #分类过程

## [,1] [,2] [,3]  
## [1,] -1 -2 0.64  
## [2,] -6 -7 1.06  
## [3,] -3 -4 1.17  
## [4,] -8 -9 1.22  
## [5,] -5 3 1.55  
## [6,] 2 4 2.07  
## [7,] 1 5 7.89  
## [8,] 6 7 13.81

plot(hc,main="张逸敏") #聚类图



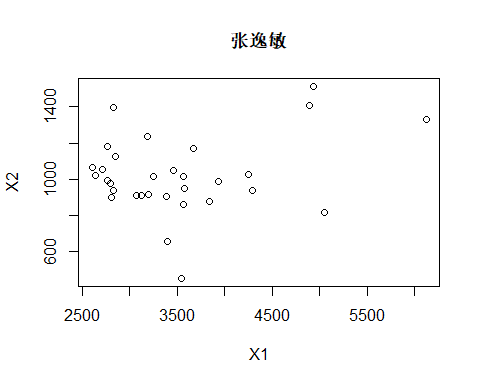
### 系统聚类法案例实战

首先自定义函数，实现距离类型、方法类型、聚类过程、聚类图像的一体化

H.clust <- function(X, d = "euc", m = "comp", proc = F, plot = T)  
{  
 D = dist(X, d)  
 hc <- hclust(D, m)  
 if (proc) { cat("\n cluster procdure: \n");  
 print(cbind(hc$merge, hc$height)) }  
 PROC = cbind(merge = hc$merge, height = hc$height)  
 if (proc) print(PROC)  
 if (plot) {  
 plot(hc, ylab = d,main="张逸敏")  
 }  
 #plot(hc,hang=hang,xlab="",ylab="",main="")  
 #hc1=as.dendrogram(hc)  
 #plot(hc1,xlab="G",ylab="D",horiz=TRUE)  
 #list(D=D,hc=hc,proc=proc)  
 return(hc)  
}

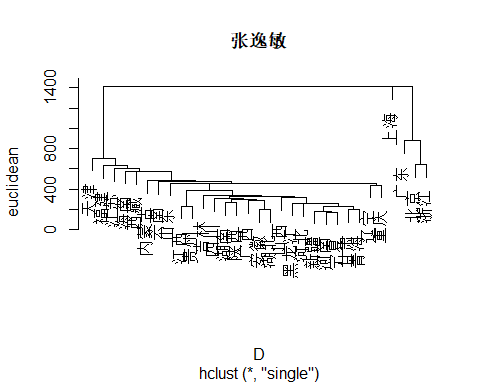
读入数据并画散点图。由于有8个指标，二维图像无法画出，这里只取前两个指标画图

d7.2 = read.table('7.2.txt', header = T)  
d7.2 <- as.matrix(d7.2)  
plot(d7.2,main="张逸敏") #用指标X1和X2画图



最短距离法

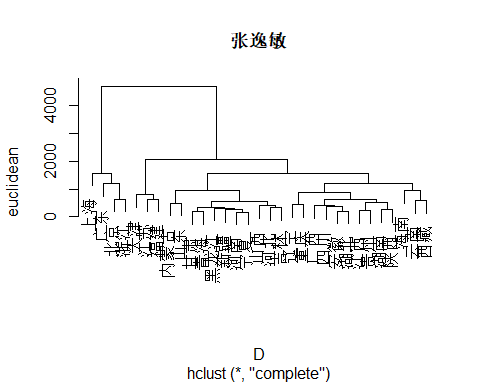
H.clust(d7.2, 'euclidean', 'single', plot = T)



##   
## Call:  
## hclust(d = D, method = m)  
##   
## Cluster method : single   
## Distance : euclidean   
## Number of objects: 31

最长距离法

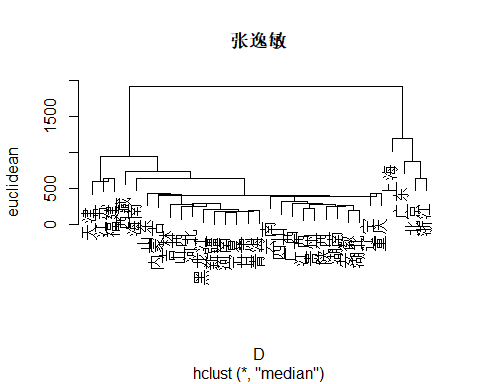
H.clust(d7.2, 'euclidean', 'complete', plot = T)



##   
## Call:  
## hclust(d = D, method = m)  
##   
## Cluster method : complete   
## Distance : euclidean   
## Number of objects: 31

中间距离法

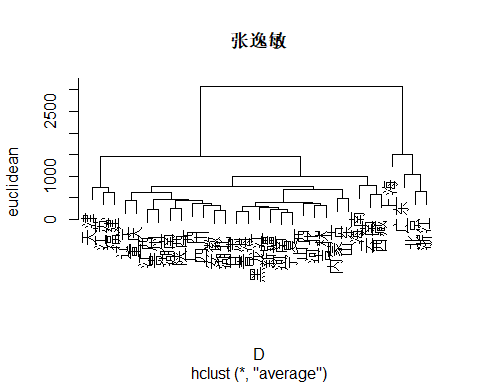
H.clust(d7.2, 'euclidean', 'median', plot = T)



##   
## Call:  
## hclust(d = D, method = m)  
##   
## Cluster method : median   
## Distance : euclidean   
## Number of objects: 31

类平均法

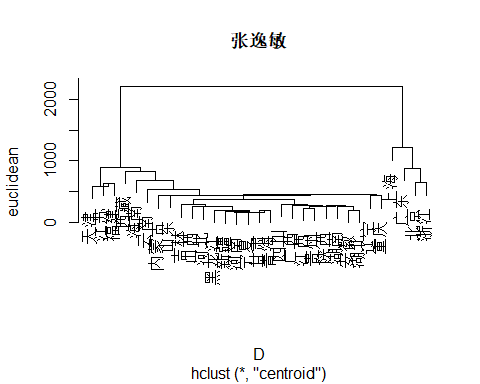
H.clust(d7.2, 'euclidean', 'average', plot = T)



##   
## Call:  
## hclust(d = D, method = m)  
##   
## Cluster method : average   
## Distance : euclidean   
## Number of objects: 31

重心法

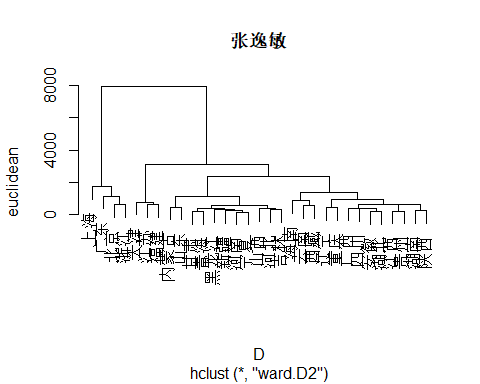
H.clust(d7.2, 'euclidean', 'centroid', plot = T)



##   
## Call:  
## hclust(d = D, method = m)  
##   
## Cluster method : centroid   
## Distance : euclidean   
## Number of objects: 31

ward法

H.clust(d7.2, 'euclidean', 'ward.D2', plot = T)



##   
## Call:  
## hclust(d = D, method = m)  
##   
## Cluster method : ward.D2   
## Distance : euclidean   
## Number of objects: 31

## kmeans聚类

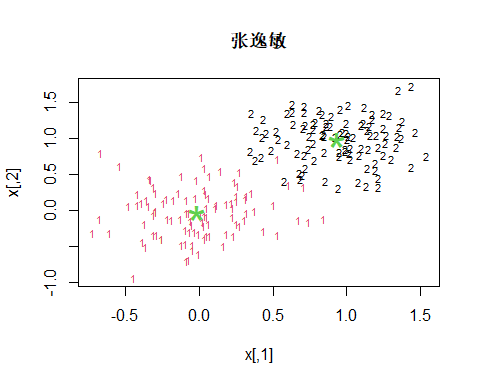
### kmeans和系统聚类对比

生成两类数据，一类是100个样本，每个样本有10个指标，指标服从均值0标准差0.3的正态分布，另一类是100个样本，每个样本10个指标，指标服从均值1标准差0.3的正态分布。 使用两种方法，系统聚类(欧氏距离、最长距离法)和kmeans聚类(k=2)，画出散点图，对比两方法区别 生成数据

x1 = matrix(rnorm(1000, mean = 0, sd = 0.3), ncol = 10)  
x2 = matrix(rnorm(1000, mean = 1, sd = 0.3), ncol = 10)  
x = rbind(x1, x2) # 按行连接

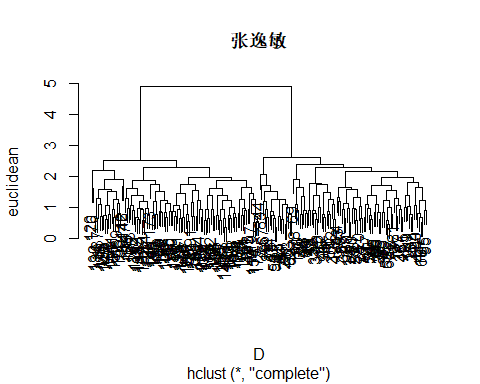
kmeans聚类结果：

cl = kmeans(x, 2) # kmeans聚类，类别数为2  
pch1 = rep('1', 100)  
pch2 = rep('2', 100)  
plot(x, col = cl$cluster, pch = c(pch1, pch2), cex = 0.7, main="张逸敏")  
points(cl$centers, col = 3, pch = '\*', cex = 3)



系统聚类结果：

hc <- H.clust(x, 'euclidean', 'complete') # 欧氏距离、最长距离法进行系统聚类



hc.cut = cutree(hc, k = 2)

画出系统聚类的散点图(取x1,x2画图)，可以发现，在这组数据下，系统聚类和kmeans聚类结果相同

library(dplyr)

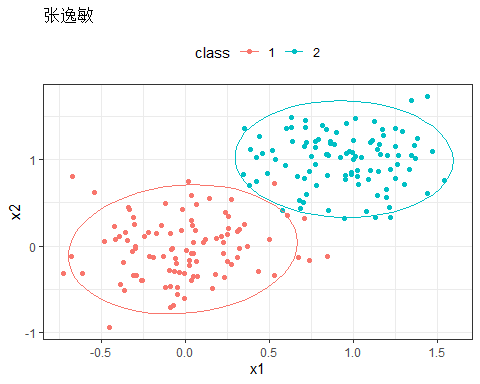
## Warning: 程辑包'dplyr'是用R版本4.2.3 来建造的

##   
## 载入程辑包：'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

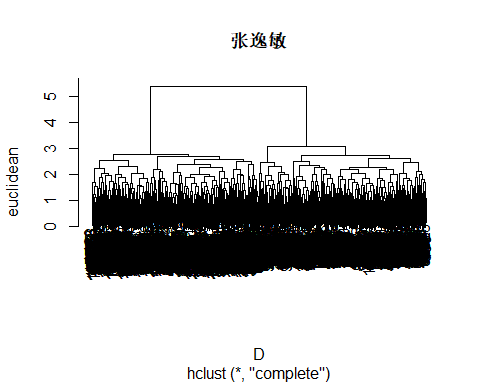
## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(ggplot2)  
tibble(x1 = x[, 1],  
 x2 = x[, 2],  
 class = as.factor(hc.cut)) %>%  
 ggplot(aes(x1, x2, col = class)) +  
 geom\_point() +  
 stat\_ellipse(level = 0.95) +  
 theme\_bw() +  
 theme(legend.position = "top") +  
 labs(title="张逸敏")



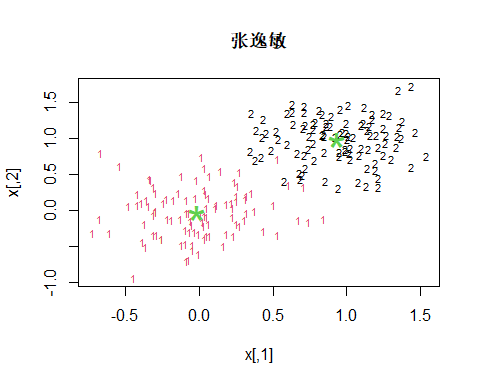
### kmeans处理大样本

y1 = matrix(rnorm(10000, mean = 0, sd = 0.3), ncol = 10)  
y2 = matrix(rnorm(10000, mean = 1, sd = 0.3), ncol = 10)  
y = rbind(y1, y2)  
H.clust(y, 'euclidean', 'complete')



##   
## Call:  
## hclust(d = D, method = m)  
##   
## Cluster method : complete   
## Distance : euclidean   
## Number of objects: 2000

cl\_y = kmeans(y, 2)  
pch1 = rep('1', 100)  
pch2 = rep('2', 100)  
plot(x, col = cl$cluster, pch = c(pch1, pch2), cex = 0.7, main="张逸敏")  
points(cl$centers, col = 3,pch='\*', cex=3)



## 案例实战

读入数据

Case6=read.table('7\_case.txt', header=T)  
Z=scale(Case6) # 标准化

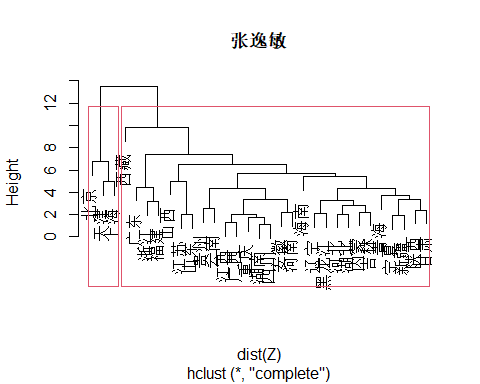
### 系统聚类

欧氏距离、最长距离法的系统聚类

hc=hclust(dist(Z))

分成两类

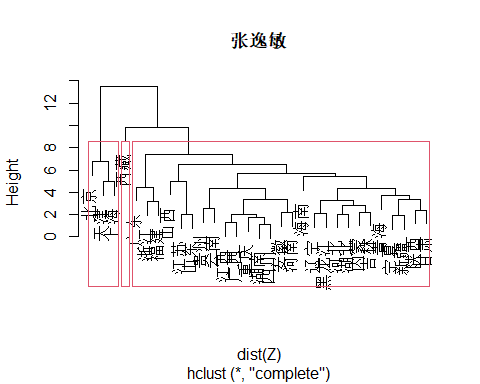
plot(hc, main="张逸敏"); rect.hclust(hc, 2); cutree(hc, 2)



## 北京 天津 河北 山西 内蒙 辽宁 吉林 黑龙江 上海 江苏 浙江   
## 1 1 2 2 2 2 2 2 1 2 2   
## 安徽 福建 江西 山东 河南 湖北 湖南 广东 广西 海南 重庆   
## 2 2 2 2 2 2 2 2 2 2 2   
## 四川 贵州 云南 西藏 陕西 甘肃 青海 宁夏 新疆   
## 2 2 2 2 2 2 2 2 2

分成三类

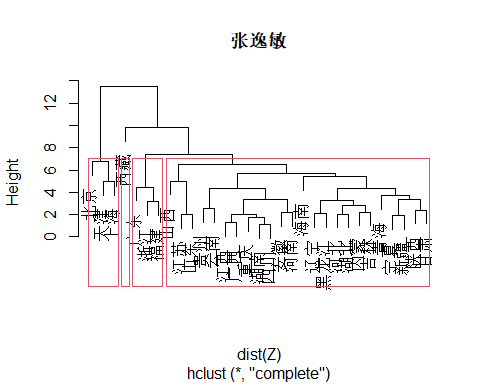
plot(hc, main="张逸敏"); rect.hclust(hc, 3); cutree(hc, 3)



## 北京 天津 河北 山西 内蒙 辽宁 吉林 黑龙江 上海 江苏 浙江   
## 1 1 2 2 2 2 2 2 1 2 2   
## 安徽 福建 江西 山东 河南 湖北 湖南 广东 广西 海南 重庆   
## 2 2 2 2 2 2 2 2 2 2 2   
## 四川 贵州 云南 西藏 陕西 甘肃 青海 宁夏 新疆   
## 2 2 2 3 2 2 2 2 2

分成四类

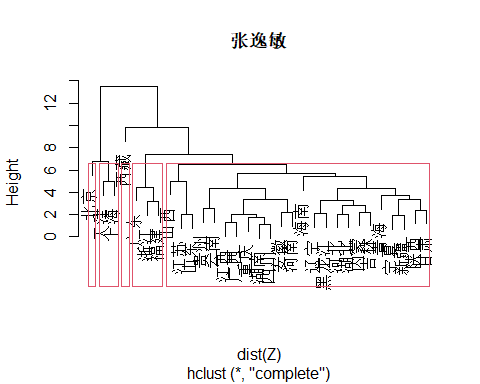
plot(hc, main="张逸敏"); rect.hclust(hc, 4); cutree(hc, 4)



## 北京 天津 河北 山西 内蒙 辽宁 吉林 黑龙江 上海 江苏 浙江   
## 1 1 2 2 2 2 2 2 1 2 3   
## 安徽 福建 江西 山东 河南 湖北 湖南 广东 广西 海南 重庆   
## 2 3 2 2 2 2 2 3 2 2 2   
## 四川 贵州 云南 西藏 陕西 甘肃 青海 宁夏 新疆   
## 2 2 2 4 2 2 2 2 2

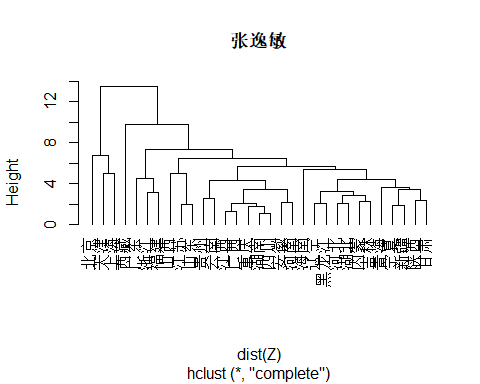
分成五类

plot(hc, main="张逸敏");rect.hclust(hc, 5); cutree(hc, 5)



## 北京 天津 河北 山西 内蒙 辽宁 吉林 黑龙江 上海 江苏 浙江   
## 1 2 3 3 3 3 3 3 2 3 4   
## 安徽 福建 江西 山东 河南 湖北 湖南 广东 广西 海南 重庆   
## 3 4 3 3 3 3 3 4 3 3 3   
## 四川 贵州 云南 西藏 陕西 甘肃 青海 宁夏 新疆   
## 3 3 3 5 3 3 3 3 3

plot(hc, main="张逸敏" ,hang=-1) # hang取负值时，代表类的竖线对齐到底边

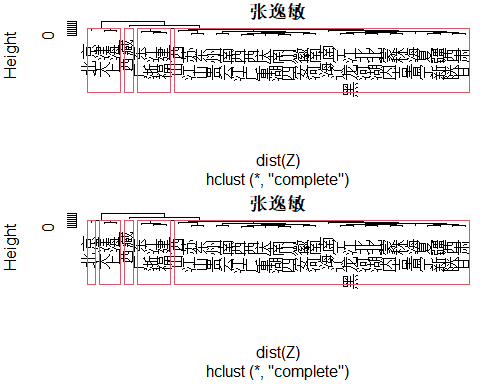


生成两行一列的聚类谱系图

opar<-par(mfrow=c(2, 1), mar=c(5.2, 4, 1, 0)) #生成两行一列 ;图像距离边界的距离  
plot(hc, main="张逸敏"); rect.hclust(hc, 4); cutree(hc, 4)

## 北京 天津 河北 山西 内蒙 辽宁 吉林 黑龙江 上海 江苏 浙江   
## 1 1 2 2 2 2 2 2 1 2 3   
## 安徽 福建 江西 山东 河南 湖北 湖南 广东 广西 海南 重庆   
## 2 3 2 2 2 2 2 3 2 2 2   
## 四川 贵州 云南 西藏 陕西 甘肃 青海 宁夏 新疆   
## 2 2 2 4 2 2 2 2 2

plot(hc, main="张逸敏"); rect.hclust(hc, 5); cutree(hc, 5)



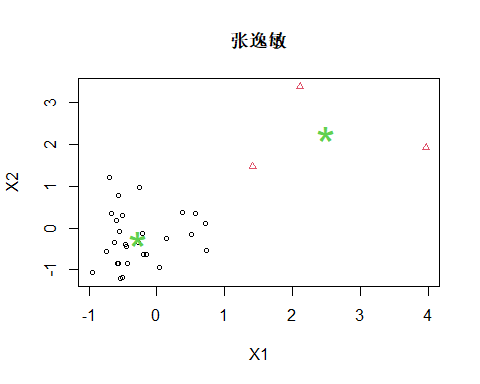
## 北京 天津 河北 山西 内蒙 辽宁 吉林 黑龙江 上海 江苏 浙江   
## 1 2 3 3 3 3 3 3 2 3 4   
## 安徽 福建 江西 山东 河南 湖北 湖南 广东 广西 海南 重庆   
## 3 4 3 3 3 3 3 4 3 3 3   
## 四川 贵州 云南 西藏 陕西 甘肃 青海 宁夏 新疆   
## 3 3 3 5 3 3 3 3 3

par(opar)

### kmeans聚类

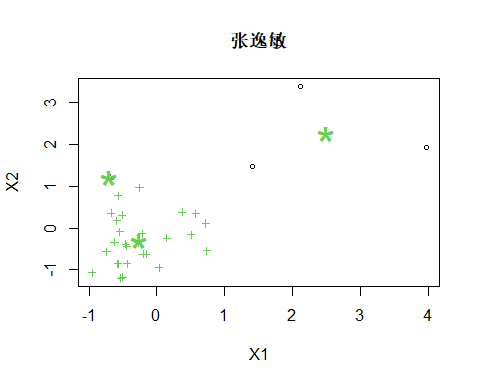
k=2

res1 = kmeans(Z, 2)  
plot(Z, col = res1$cluster, pch = as.data.frame(res1$cluster)[,1], cex = 0.7, main="张逸敏")  
points(res1$centers, col = 3, pch = '\*', cex = 3)



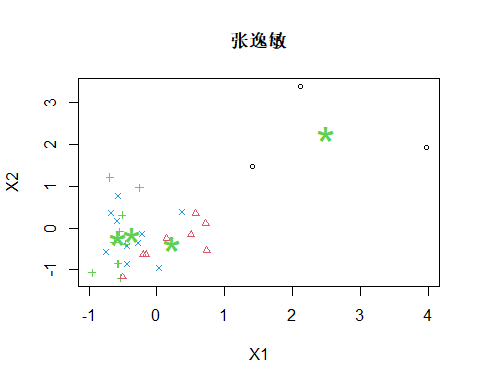
k=3

res2 = kmeans(Z, 3)  
plot(Z, col = res2$cluster, pch = as.data.frame(res2$cluster)[,1], cex = 0.7, main="张逸敏")  
points(res2$centers, col = 3, pch = '\*', cex = 3)



k=4

res3 = kmeans(Z, 4)  
plot(Z, col = res3$cluster, pch = as.data.frame(res3$cluster)[,1], cex = 0.7, main="张逸敏")  
points(res3$centers, col = 3, pch = '\*', cex = 3)



k=5

res4 = kmeans(Z, 5)  
plot(Z, col = res4$cluster, pch = as.data.frame(res4$cluster)[,1], cex = 0.7, main="张逸敏")  
points(res4$centers, col = 3, pch = '\*', cex = 3)

