Homework 9:

Use the following clustering methods to analyze the U.S. crime data set (Sec.22.8) with the 𝐿2−𝑛𝑜𝑟𝑚 on standardized variables (use only the crime variables𝑋3~𝑋9).

1. K-means clustering algorithm

2. Ward algorithm

Make a complete conclusion from these analysis.

Hint: 1. Set the number of cluster𝐾=4.

|  |  |
| --- | --- |
| Variable | Description |
|  | murder (murd)謀殺 |
|  | rape強姦 |
|  | robbery (robb)搶劫 |
|  | assault (assa)襲擊 |
|  | burglary (burg)入室竊盜 |
|  | larcery (larc)竊盜 |
|  | autothieft (auto)汽車竊盜 |

Sol.

　　執行兩種分析方法前，首先標準化變數，接著可以透過Euclidean norm找到距離矩陣D，由距離矩陣可得知其值越小，代表組內距離、變異越小，這是我們所期望的，以下簡略列出該資料標準化後的距離矩陣：

1. 　K-means clustering algorithm

　　進行K-means分群方法時，預期將資料及分為四大類，接著重複分發25次，將最接近平均(中心)的目標分在一群中，最後可以得到四群類別裡分別有14, 9, 13, 14組資料，群內的組內平方和為65%，其中各地區資料南部地區以第一群分布最多，其他地區較為分散。從圖一與圖二中也可以看出類似的趨勢。

　　根據前次PCA分析方法分析同筆資料時得知，若變數僅保留犯罪類型時，表一顯示第一組主成分可以解釋58.24%的變異；前兩組主成分可以解釋78.69%的變異。另外，可重複驗證僅前兩組特徵值位於1之上，因此該組資料可以取前兩組主成分進行分析即可。本篇介紹之三維以上的資料，將以二維主

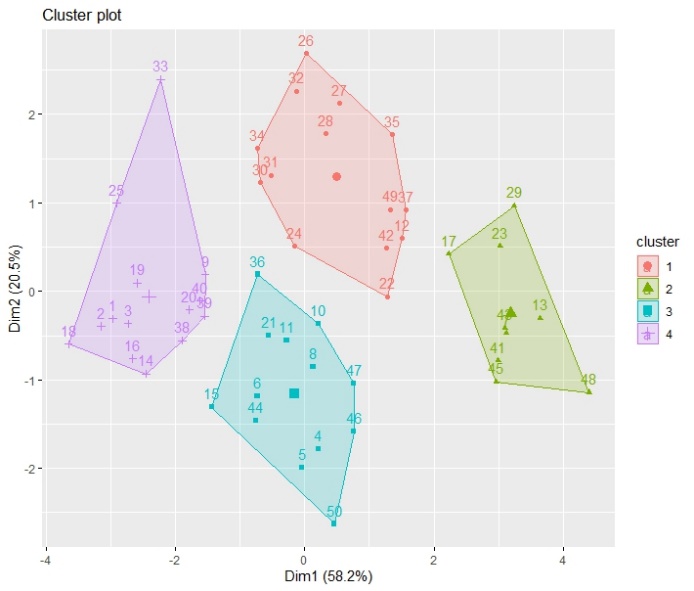
成分為視覺化基礎。

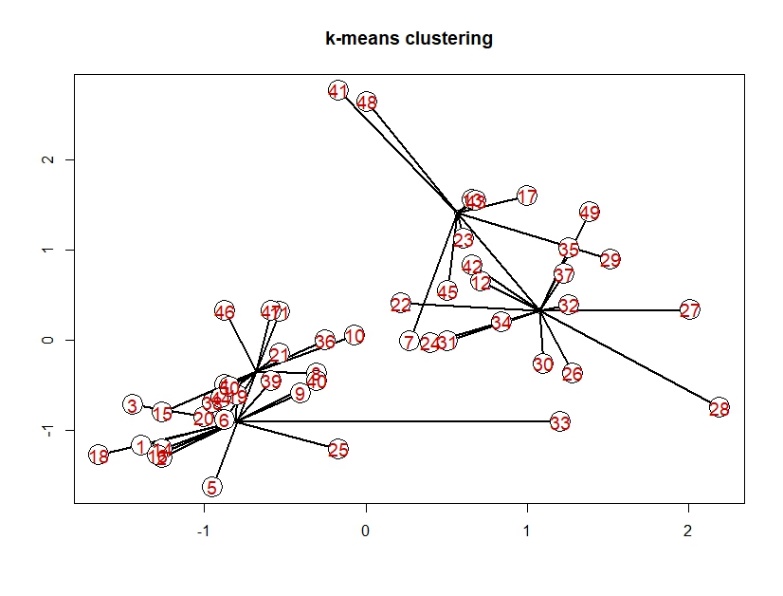
|  |  |  |
| --- | --- | --- |
|  | PC1 | PC2 |
| 謀殺 | 0.557 | **-0.771** |
| 強姦 | **0.851** | -0.139 |
| 搶劫 | **0.782** | 0.055 |
| 襲擊 | **0.784** | -0.546 |
| 入室竊盜 | **0.881** | 0.308 |
| 竊盜 | 0.728 | 0.480 |
| 汽車竊盜 | 0.714 | 0.438 |

表一：特徵值及其解釋變異的 表二、前二組主成分與原始變數

百分比() 關係()

|  |  |
| --- | --- |
| Eigenvalue | Cumulated percentages |
|  | **0.5824** |
|  | **0.7869** |
|  | 0.8771 |
|  | 0.9257 |
|  | 0.9612 |
|  | 0.9811 |
| 0.132 | **1** |

圖一、分群散佈圖(K-means)　　　　　　圖二、分群中心距離圖



　　接著使用 fivz\_cluster() 函數在散佈圖上視覺化各個分群，如圖一，其中兩軸分別是前兩個主成分，可以看到各群之間存在一定差距（距離），各群內每筆資料又多為集中，表示透過k-means可成功將資料分為的4個互不干擾的群體。除此之外，由表二可以看出PC1以強姦、搶劫、襲擊、入室竊盜為主要解釋變數；PC2以謀殺為主要解釋變數，再對照圖一便可以得到第一群的犯罪類型以謀殺占比最高，第一群中又以南部的22~37號資料點為多數，因此可以間接說明南部地區的犯罪類型以謀殺占多數。另外，由圖二可以比圖一更清楚看到每筆資料與到四群中心（平均）的距離。

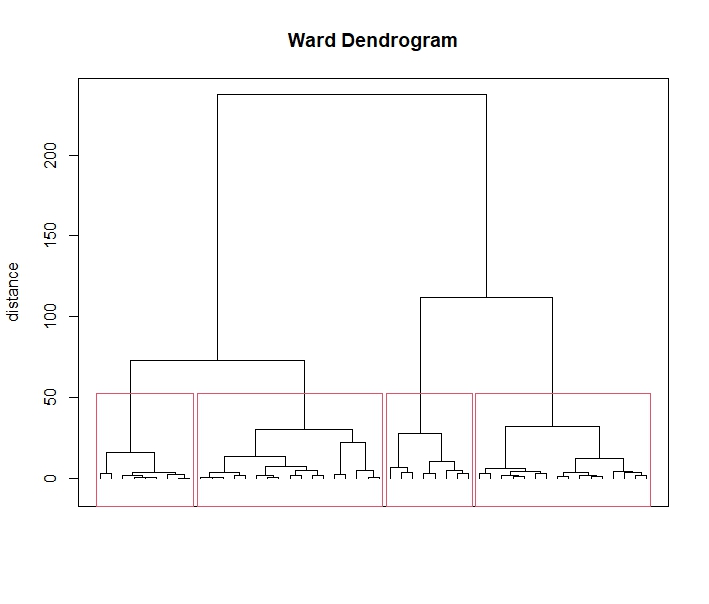
　　根據表三的數據資料，可看到各群內的平均數與標準誤，透過平均數與標準誤可以得到各群之間最受影響的變數分別為何者、影響程度如何等等。像是我們能明顯看出第一群內最受到謀殺與襲擊的影響，第二群則是根據強姦、搶劫、入室竊盜而有所變動，第三群中雖然有些犯罪類型有正想影響、有些是反向影響，但數值幾乎都不大，可以說第三群對所有犯罪類型呈現平均表現，第四群則以入室竊盜有最大影響。

表三、標準化後變數四個群內的平均數與標準誤(K-means)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Mean C1 | SE C1 | Mean C2 | SE C2 | Mean C3 | SE C3 | Mean C4 | SE C4 |
| 謀殺 | **1.076** | 0.153 | 0.565 | 0.169 | **-0.681** | 0.093 | -0.808 | 0.197 |
| 強姦 | 0.326 | 0.157 | **1.410** | 0.303 | -0.346 | 0.159 | -0.912 | 0.087 |
| 搶劫 | -0.078 | 0.158 | **1.519** | 0.403 | -0.099 | 0.104 | -0.806 | 0.076 |
| 襲擊 | **0.783** | 0.165 | 1.120 | 0.174 | -0.595 | 0.112 | -0.950 | 0.124 |
| 入室竊盜 | -0.115 | 0.109 | **1.445** | 0.206 | 0.320 | 0.172 | **-1.111** | 0.092 |
| 竊盜 | -0.321 | 0.181 | 1.219 | 0.254 | 0.337 | 0.213 | -0.775 | 0.191 |
| 汽車竊盜 | -0.126 | 0.182 | 0.978 | 0.169 | **0.537** | 0.281 | -1.001 | 0.104 |

2. 　Ward algorithm

圖三、分群樹狀圖

由圖三的樹狀圖可以看到從k=2到最底部的50筆資料可以分群的概況，分為4群的情況為圖中紅色框線框起來的部分，基本上各有9, 17, 8, 16筆資料在各個分群內。

圖四、分群散佈圖(Ward’ s method)

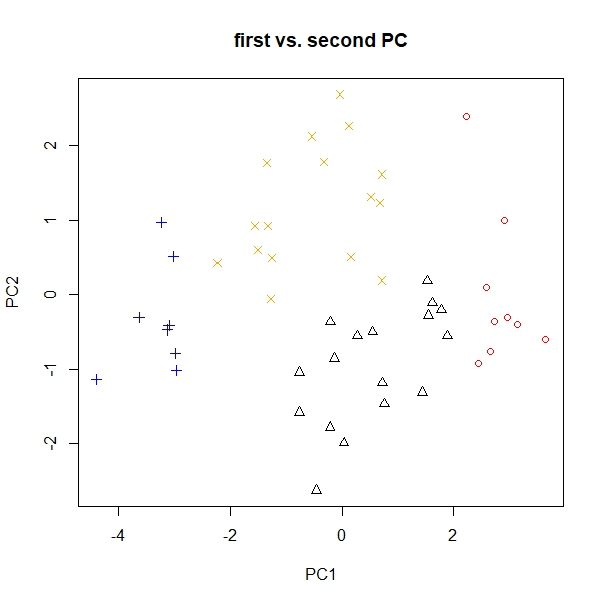
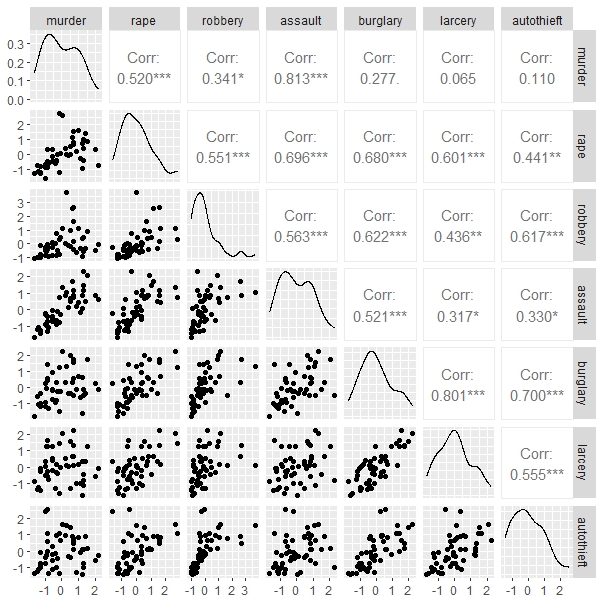


由圖四一樣能看出四群互不相交，表示群間有變異，可以更直接看到四群中分別有哪些資料在裡面，其中可以看到PC2較高的紫色十字圖例第四群中數字為22~37的南部地區占了多數，說明第四群以南部地區的犯罪類型為主，且其犯罪類型為PC2的主要解釋變數：謀殺。

根據表四也可再次驗證第四群多為謀殺及襲擊兩種犯罪類型。此外，也可清楚看到第三群與搶劫的相關，而在圖四中第三權在PC1表現最高，可以驗證PC1為一強姦、搶劫、襲擊、入室竊盜為主的主成分。

表四、標準化後變數四個群內的平均數與標準誤(Ward’ s method)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Mean C1 | SE C1 | Mean C2 | SE C2 | Mean C3 | SE C3 | Mean C4 | SE C4 |
| 謀殺 | -0.90 | 0.300 | **-0.70** | 0.076 | 0.51 | 0.182 | **0.99** | 0.156 |
| 強姦 | -1.08 | 0.087 | -0.44 | 0.123 | 1.39 | 0.343 | 0.39 | 0.160 |
| 搶劫 | -0.91 | 0.035 | -0.23 | 0.109 | **1.57** | 0.453 | -0.03 | 0.159 |
| 襲擊 | -1.09 | 0.168 | -0.65 | 0.085 | 1.20 | 0.174 | **0.71** | 0.155 |
| 入室竊盜 | **-1.22** | 0.124 | -0.01 | 0.193 | 1.54 | 0.209 | -0.07 | 0.109 |
| 竊盜 | -1.06 | 0.194 | 0.20 | 0.191 | 1.35 | 0.247 | **-0.29** | 0.161 |
| 汽車竊盜 | -1.18 | 0.081 | 0.24 | 0.257 | 0.99 | 0.191 | -0.08 | 0.172 |

圖五、主成分分群散佈圖 圖六、相關係數矩陣

O : cluster 1 △ : cluster 2

+ : cluster 3 x : cluster 4

Conclusion:

表五、地區與分群對照(k-means/ward)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Cluster 1 | | Cluster 2 | | Cluster 3 | | Cluster 4 | |
| k-means | ward | k-means | ward | k-means | ward | k-means | ward |
| 東北部 | 0 | 3 | 1 | 5 | 4 | 1 | 4 | 0 |
| 中西部 | 1 | 4 | 2 | 5 | 4 | 1 | 5 | 2 |
| 南部 | **11** | 2 | 2 | 0 | 1 | 2 | 2 | **12** |
| 西部 | 2 | 0 | 4 | 7 | 4 | 4 | 3 | 2 |

　　透過圖一與圖四對表二PC1及PC2的比較，可以得到類似的結論：南部地區犯罪類型以謀殺占多數，表示在兩種分群方式下雖然略有不同，但關鍵性資訊仍在兩這間達成共識。最後用圖五與圖六整理該資料，圖六顯示竊盜一類普遍呈現高度相關，謀殺與扣除襲擊之外的犯罪類型普遍出現較低的關聯性，可以明確地切分出PC1與PC2。圖五則看出第一群與PC1類型(強姦、搶劫、襲擊、入室竊盜)相像，第二群與PC2(謀殺、襲擊)相反，第三群與PC1相反，第四群與PC2犯罪類型相似。

R code:

rm(list = ls(all = TRUE))

graphics.off()

# load data

data <- read.table("C:/Users/user/Desktop/多變量11101/uscrime.dat")

x <- data[,(3:9)]

# x <- cbind(x, rep("en"=x[1:9,], "mw"=x[10:21,], "sou"=x[22:37,], "wes"=x[38:50,]))

x1 <- scale(x) # standardize variable

# define variable names

colnames(x1) = c("murder", "rape", "robbery", "assault", "burglary", "larcery", "autothieft")

row.s = apply(x, 1, sum) # row sums

column.s = apply(x, 2, sum) # column sums

mat.s = sum(x) # matrix sum

D = matrix(0, nrow = 50, ncol = 50)

# distance for rows

for (i in 1:(dim(x)[1] - 1)) {

for (j in (i + 1):dim(x)[1]) {

for (z in 1:dim(x)[2]) {

D[i, j] = 1/(column.s[z]/mat.s) \* ((x[i, z]/row.s[i]) - (x[j, z]/row.s[j]))^2

}

}

}

D

# k-means clustering algorithm

# install.packages("ClusterR")

library(ClusterR)

library(cluster)

library(factoextra)

set.seed(1221)

k.means <- kmeans(x1, 4, nstart = 25)

#plot results of final k-means model

fviz\_cluster(k.means, data = x1)

#find means of each cluster

mc.km <- cbind(colMeans(subset(x1, k.means$cluster == "1")), colMeans(subset(x1, k.means$cluster == "2")),

colMeans(subset(x1, k.means$cluster == "3")), colMeans(subset(x1, k.means$cluster == "4")))

library(matrixStats)

sc.km <- cbind(colSds(subset(x1, k.means$cluster == "1")[, 1:ncol(x1)]),

colSds(subset(x1, k.means$cluster == "2")[, 1:ncol(x1)]),

colSds(subset(x1, k.means$cluster == "3")[, 1:ncol(x1)]),

colSds(subset(x1, k.means$cluster == "4")[, 1:ncol(x1)]))

tbl.km <- cbind(mc.km[, 1], sc.km[, 1]/sqrt(nrow(subset(x1, k.means$cluster == "1"))),

mc.km[, 2], sc.km[, 2]/sqrt(nrow(subset(x1, k.means$cluster == "2"))),

mc.km[, 3], sc.km[, 3]/sqrt(nrow(subset(x1, k.means$cluster == "3"))),

mc.km[, 4], sc.km[, 4]/sqrt(nrow(subset(x1, k.means$cluster == "4"))))

# 蜘蛛圖

dev.new()

plot(x1,type="n", xlab="",ylab="", main="k-means clustering")

points(k.means$centers[1,1],k.means$centers[1,2], col = "black")

points(k.means$centers[2,1],k.means$centers[2,2], col = "black")

# Plot Lines

k.means$cluster

for (i in 1:50){

segments(x1[i,1], x1[i,2],

k.means$centers[k.means$cluster[i], 1],k.means$centers[k.means$cluster[i], 2],lwd=2)

}

segments(k.means$centers[1,1],k.means$centers[1,2],k.means$centers[2,1],k.means$centers[2,2],lwd=2)

points(x1, pch=21, cex=3, bg="white")

text(x1, as.character(1:50),col="red3",cex=1.2)

# Ward algorithm

d <- dist(x1, "euclidean", p = 2) # euclidean distance matrix(p=power of Minkowski distance)

dd <- d^2

w <- hclust(dd, method = "ward.D") # cluster analysis with ward algorithm

tree.wd = cutree(w, 4)

# plot results of ward algorithm

fviz\_cluster(list(data = x1, cluster = tree.wd))

t1 = subset(x1, tree.wd == 1)

t2 = subset(x1, tree.wd == 2)

t3 = subset(x1, tree.wd == 3)

t4 = subset(x1, tree.wd == 4)

# Plot 1: Dendrogram for the standardized data after Ward

plot(w, hang = -0.1, labels = FALSE, frame.plot = TRUE, ann = FALSE)

title(main = "Ward Dendrogram", ylab = "distance")

rect.hclust(w,k=4) # k=4的分類線

# means for 4 Clusters

mc.wd <- cbind(colMeans(subset(x1, tree.wd == "1")), colMeans(subset(x1, tree.wd == "2")),

colMeans(subset(x1, tree.wd == "3")), colMeans(subset(x1, tree.wd == "4")))

# standard deviations for 4 Clusters

sc.wd <- cbind(colSds(t1[, 1:ncol(x1)]), colSds(t2[, 1:ncol(x1)]),

colSds(t3[, 1:ncol(x1)]), colSds(t4[, 1:ncol(x1)]))

# means and standard deviations of the standardized variables for 4 Clusters

tbl.wd = cbind(mc.wd[, 1], sc.wd[, 1]/sqrt(nrow(t1)), mc.wd[, 2], sc.wd[, 2]/sqrt(nrow(t2)),

mc.wd[, 3], sc.wd[, 3]/sqrt(nrow(t3)), mc.wd[, 4], sc.wd[, 4]/sqrt(nrow(t4)))

# spectral decomposition

eig = eigen(cov(x1))

e = eig$values

v = eig$vectors[, 1:2]

dav = x1 %\*% v

# PC選取

cum = cumsum(e)/sum(e)

corr = cor(x1, -dav)[, 1:2]

plot(e, ylim = c(0, 6), xlab = "Index", ylab = "Lambda", main = "Eigenvalues",

cex.lab = 1.2, cex.axis = 1.2, cex.main = 1.8)

abline(h=1, col="blue")

# 自定義點的形狀&顏色

tree.wd[tree.wd == 1] = 1

tree.wd[tree.wd == 2] = 2

tree.wd[tree.wd == 3] = 3

tree.wd[tree.wd == 4] = 4

tr = tree.wd

tr[tr == 1] = "red"

tr[tr == 2] = "black"

tr[tr == 3] = "blue"

tr[tr == 4] = "orange"

# Scatterplot for the first two PCs displaying the 4 clusters

dev.new()

plot(dav[, 1], dav[, 2], pch = tree.wd, col = tr, xlab = "PC1", ylab = "PC2", main = "first vs. second PC")

# c.f.

table <- cbind(x1, tree.wd, k.means$cluster)

colnames(table) <- c("murder", "rape", "robbery", "assault", "burglary",

"larcery", "autothieft", "ward", "kmeans")

# correlation coefficient

r <- round(cor(x1), digits = 3)

library(ggplot2)

library(GGally)

# Scatterplot matrix for variables X1 to X7

ggpairs(data = as.data.frame(x1), alpha = 0.5)

# dev.new()

# ggpairs(data = x, mapping = aes(color = tr), alpha = 0.5)