多元统计分析作业3

辛柏嬴 2020111753

2023-04-21

### 在例6.3.3和例6.4.2中，不进行标准化处理，用同样的方法进行聚类分析，并比较结果。

|  |
| --- |
| library(readxl)  library(tidyverse)  # ex 6.5  setwd("D:/Sufe/Multivariate-Stat-Analysis/Hw&Proj/hw3")    ## Clustering for e.g.6.3.3  dat1 <- read\_xlsx("examp6.3.3.xlsx") %>% select(-region) #load data    dist <- dist(dat1, method="euclidean", diag = TRUE) #compute distance  dist\_std <- dist(scale(dat1), method = "euclidean", diag=TRUE) #compute std distance    ### WARD.Distance  hc\_ward <- hclust(dist, "ward.D") #ward clustering  hc\_ward\_std <- hclust(dist\_std,"ward.D") #ward clustering on std distance      par(mfrow=c(2,1)) #show cluster fig  plot(hc\_ward, hang =-1) #plot cluster  rect.hclust(hc\_ward, k=3))#plot frame to show cluster  plot(hc\_ward\_std, hang =-1)#plot cluster std  rect.hclust(hc\_ward\_std, k=3) #plot frame to show cluster std    cutree(hc\_ward,k=3) #show cluster result  cutree(hc\_ward\_std,k=3)#show cluster result std    ### Longest Distance  long\_dist <- hclust(dist, "complete") #longest dist clustering  long\_dist\_std <- hclust(dist\_std,"complete") #longest dist clustering on std distance      par(mfrow=c(2,1)) #show cluster fig  plot(long\_dist, hang =-1) #plot cluster  rect.hclust(long\_dist, k=3)#plot frame to show cluster  plot(long\_dist\_std, hang =-1)#plot cluster std  rect.hclust(long\_dist\_std, k=3) #plot frame to show cluster std    cutree(long\_dist,k=3) #show cluster result  cutree(long\_dist\_std,k=3)    ### Centroid  center\_dist <- hclust(dist, "centroid") #centroid dist clustering  center\_dist\_std <- hclust(dist\_std,"centroid") #centroid dist clustering on std distance      par(mfrow=c(2,1)) #show cluster fig  plot(center\_dist, hang =-1) #plot cluster  rect.hclust(center\_dist, k=3)#plot frame to show cluster  plot(center\_dist\_std, hang =-1) #plot cluster std  rect.hclust(center\_dist\_std, k=3) #plot frame to show cluster std    cutree(center\_dist,k=3) #show cluster result  cutree(center\_dist\_std,k=3) #show cluster result std    ## Clustering for e.g.6.4.2  kmean <- kmeans(dat1,5) #kmeans for original data  sort(kmean$cluster) #show result in order  kmean\_std <- kmeans(scale(dat1),5) #kmeans for std data  sort(kmean\_std$cluster) #show result in order      **Ward聚类结果对比：**  > 非标准化数据  [1] 1 1 2 2 2 2 2 2 1 3 1 3 1 2 2 2 3 3 1 3 3 3 3 2 3 1 2 2 2 2 2  > 标准化数据  [1] 1 2 2 3 3 3 3 3 1 2 1 3 3 3 2 3 2 2 1 3 3 2 2 3 2 2 3 3 3 3 2 |
|  |
| **最长距离法聚类结果对比：**  > 非标准化数据  [1] 1 1 2 2 2 2 2 2 1 3 1 2 3 2 2 2 2 2 1 3 3 3 2 2 3 3 2 2 2 2 2  > 标准化数据  [1] 1 1 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 |
| **重心法聚类结果对比：**  > 非标准化数据  [1] 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2  > 标准化数据  [1] 1 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 |
|  |
| **K-means聚类结果对比：**  > 非标准化数据  [1] 1 1 1 2 2 2 2 2 2 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 5 5 5 5 5 5 5  > 标准化数据  [1] 1 1 1 1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 5 |
| **综合上述输出结果，标准化数据之间的聚类结果较为接近，而非标准化与标准化数据的聚类结果之间相差较大。这说明标准化与否会在很多时候显著影响最终的聚类效果，尤其在数据的差异性较大的情况下，更应该考虑对数据进行标准化处理后再进行聚类，以得到更为有效的聚类结果。** |

### 下表中列出各个国家和地区男子比赛的数据，分别用类平均法、离差平方和法和kmeans法进行聚类，在聚类之前先对数据进行标准化处理。

# ex.6.6

dat2 <- scale(read\_excel("exec6.6.xlsx")%>%select(-nation)) # load data

dist <- dist(dat2, diag=TRUE) # compute distance

# Average Linkage Method

hc\_avg <- hclust(dist, method = "average")#clustering

par(mfrow=c(1,1))

plot(hc\_avg, hang=-1) #plot cluster result

rect.hclust(hc\_avg, k=4) #frame out clusters

cutree(hc\_avg, k=4) #show results

# Ward Method

hc\_wrd <- hclust(dist, method = "ward.D")#clustering

plot(hc\_wrd, hang=-1) #plot cluster result

rect.hclust(hc\_wrd, k=4) #frame out clusters

cutree(hc\_wrd, k=4) #show results

# kmeans

kmeans <- kmeans(dat2,4)

kmeans$cluster

|  |
| --- |
| **类平均法：**    分类结果：  1 1 1 1 2 1 2 1 1 1 1 3 2 1 1 2 1 1 1 1 1 1 2 1 1 2 1 1 1 1 1 1  2 1 2 2 1 1 1 1 2 2 1 1 1 2 1 1 1 1 2 1 1 1 4 |
| **离差平方和法：**    分类结果：  1 2 1 2 3 2 3 2 1 1 1 4 3 2 2 3 2 2 2 2 2 1 3 2 1 3 1 1 2 2 2 1  3 1 3 3 1 2 2 1 3 3 2 1 2 3 2 2 2 3 3 1 2 2 4 |
| **k-means法：**  分类结果：  2 4 2 4 3 4 3 4 2 2 2 1 3 4 2 3 4 4 4 4 4 2 3 2 2 3 2 2 4 2 4 2  3 2 3 3 2 4 2 2 3 3 4 2 2 3 2 4 4 3 3 2 4 4 1 |

### 对例6.3.7进行PCA

# ex.7.5

dat3 <- as.matrix(read\_excel("examp6.3.7.xlsx")%>%select(-1)) #load data

eign <- eigen(dat3) #calculate eign value and eign vector

eign.value <- eign$values

eign.vector <- eign$vectors

contribution <- eign.value/sum(eign.value) #calculate contribution rate

accumulative.contribution <- cumsum(contribution)#cal accu.contribution rate

par(mfrow=c(2,1)) #plot contribution of each PC

plot(contribution,type='o', main='Contribution of each PC',

xlab = "Principal Components",ylab="Percentage")

plot(accumulative.contribution, type='o',

main="Accumulative Contribution of PC",

xlab = "Principal Components",

ylab="Percentage")

par(mfrow=c(1,1))

print(round(eign.vector,3))

|  |
| --- |
| **对该矩阵求特征值特征向量有（均四舍五入到小数点后三位）：**  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]  [1,] -0.398 -0.280 -0.101 0.107 -0.408 -0.152 0.636 -0.384  [2,] -0.389 -0.331 0.113 -0.068 0.341 -0.072 0.278 0.723  [3,] -0.376 -0.345 0.015 0.047 0.541 0.392 -0.242 -0.482  [4,] -0.388 -0.297 -0.145 -0.124 -0.459 -0.251 -0.662 0.112  [5,] -0.351 0.394 -0.213 0.114 -0.296 0.720 0.026 0.237  [6,] -0.312 0.401 -0.073 0.713 0.219 -0.410 -0.112 -0.007  [7,] -0.286 0.436 -0.421 -0.630 0.257 -0.258 0.080 -0.125  [8,] -0.310 0.314 0.853 -0.221 -0.110 -0.041 -0.033 -0.117  **其每一列为一个特征向量。对应的特征值为：**  4.673 1.771 0.481 0.421 0.233 0.187 0.137 0.096  **由PCA的原理可知对于这一系列从大到小排列的特征值，第个特征值对应的特征向量即为该组数据的第个主成分。**  **同时可以求解其贡献率与累计贡献率与下所示：**    **故大约可以选取三至四个主成分为最佳，对原始变量进行后续分析。** |

### 下表是美国犯罪率数据。对该数据进行PCA。

# ex.7.6

dat4 <- as.matrix(read\_excel("exec7.6.xlsx")%>%select(-state)) #load data

dat4

PCA = prcomp(dat4, center = TRUE, scale. = TRUE) #get std. data PCA

summary(PCA) #summary pca

screeplot(PCA,type="lines")

|  |
| --- |
| **通过计算可以得到各个主成分的贡献率与累计贡献率如下：**  Importance of components:  PC1 PC2 PC3 PC4 PC5 PC6 PC7  Standard deviation 2.0285 1.1130 0.8519 0.5625 0.50791 0.47121 0.35222  Proportion of Variance 0.5878 0.1770 0.1037 0.0452 0.03685 0.03172 0.01772  Cumulative Proportion 0.5878 0.7648 0.8685 0.9137 0.95056 0.98228 1.00000  **通过如下碎石图可以的大致从统计学上判断可以选取4个主成分为宜。**    **其对应的前四个主成分的旋转矩阵如下所示。**  PC1 PC2 PC3 PC4  x1 -0.3002792 -0.62917444 0.17824530 -0.23211411  x2 -0.4317594 -0.16943512 -0.24419758 0.06221567  x3 -0.3968755 0.04224698 0.49586087 -0.55798926  x4 -0.3966517 -0.34352815 -0.06950972 0.62980445  x5 -0.4401572 0.20334059 -0.20989509 -0.05755491  x6 -0.3573595 0.40231912 -0.53923144 -0.23488987  x7 -0.2951768 0.50242093 0.56838373 0.41923832  **进一步结合其各变量的实际含义，可以看出PC1约表示的是整体的犯罪水平，PC2约表示暴力犯罪水平，PC3、PC4的显示解释意义稍差。** |

### 下表是纽约股票交易数据。对该数据进行PCA。

# ex.7.7

dat5 <- as.matrix(read\_excel("exec7.7.xlsx")%>%select(-week)) #load data

dat5

PCA = prcomp(dat5, center = TRUE, scale. = TRUE) #get std. data PCA

summary(PCA) #summary pca

screeplot(PCA,type="lines")

round(PCA$rotation,3)

|  |
| --- |
| **通过计算可得各主成分的相应贡献率与累计贡献率：**  Importance of components:  PC1 PC2 PC3 PC4 PC5  Standard deviation 1.6901 0.8995 0.7349 0.67182 0.5857  Proportion of Variance 0.5713 0.1618 0.1080 0.09027 0.0686  Cumulative Proportion 0.5713 0.7331 0.8411 0.93140 1.0000  **相应可以做出碎石图：**    **由碎石图可见，在统计意义上，约2~3个主成分即可较好的对数据进行归纳概括。**  **具体分析前三个矩阵的旋转矩阵：**  PC1 PC2 PC3 PC4 PC5  x1 0.464 -0.241 0.613 -0.381 0.453  x2 0.457 -0.509 -0.178 -0.211 -0.675  x3 0.470 -0.261 -0.337 0.664 0.396  x4 0.422 0.525 -0.539 -0.473 0.179  x5 0.421 0.582 0.434 0.381 -0.387  **PC1约表示股票整体的回报水平，PC2表示不同的股票类型，其中是一种股票，是一种股票，这也与其现实含义相对应：1、2、3为化工类股票，4、5为石油类股票。后续主成分的解释性较差，因此也可以主要选取前两类主成分进行后续分析。** |
|  |
|  |