HUDM6122 Homework_06

Chenguang Pan

2023-03-27

0.1 Github Address

All my latest homework can be found on Github: https://github.com/cgpan/hudm6122_homeworks . Thanks for checking if interested.

0.2 Ex 6.1

Apply k-means to the crime rate data after standardising each variable by its standard deviation. Compare the results with those given in the text found by standardising by a variable's range.

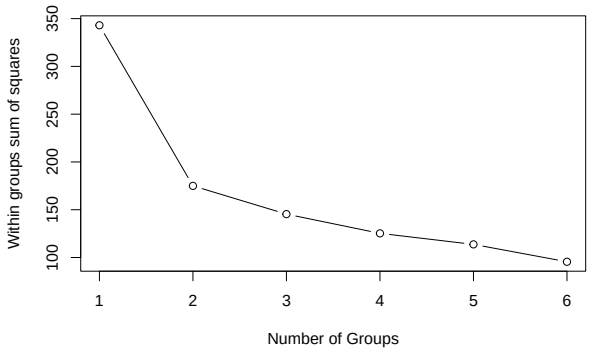
MY SOLUTION:

I began this assignment at Mar 27th, at that time there was no `crime` dataset available. Therfore, for this homework, I first extracted the data information from the `MVA` package and created the dataset in a csv format for reading convenience. For space-saving concern, the `R` syntax to create dataset was in a separated file called "HW06_test.R", which can be found it in my Github repo.

As discussed on page.179 of the textbook, one need to remove the outlier (i.e., DC) first.

```
> # import the dataset
> crime <- read.csv("crime.csv",row.names = 1)
> dim(crime)
[1] 51  7
>
> # drop the outlier DC and check the dimension of the datset
> df <- crime[-which(row.names(crime) == "DC"),]
> dim(df)
[1] 50  7
>
> # standardized the variable by its SD
> df_s <- sweep(df,2,apply(df, 2, sd), FUN = "/")
> sapply(df_s, var)
  Murder     Rape  Robbery  Assault Burglary    Theft  Vehicle
       1        1        1        1        1        1        1
>
> # similar to textbook, I use the within group variance to
> # investigate the appropriate number of cluster
> n <- nrow(df_s)
> # make an empty vector to load the wss for each # of cluster
> wss <- rep(0,6)
> # write the only one cluster(ie, the sum square of the total data)
> # for the one-group condition, I use the variance times the sample size minus 1
```

```
> # to get the withingroup sum of squares
> wss[1] <- (n-1) * sum(sapply(df_s, var))
> # for the following conditions, can use the kmeans to get the wss directly
> for (i in 2:6) {
+   wss[i] <- sum(kmeans(df_s,
+                        centers = i)$withinss)
+ }
> plot(1:6,wss, type = "b",
+      xlab = "Number of Groups",
+      ylab = "Within groups sum of squares")
```

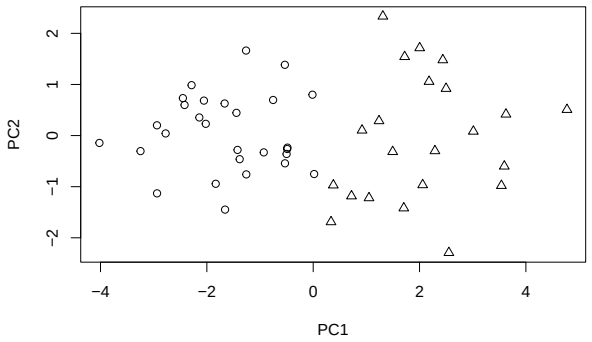


From the plot of within-group sum of squares for one- to six-cluster solutions is similar to the textbook's result. The elbow occurs at the two-group solution. Thus, I run the k-means method for this solutions.

```
> kmeans(df_s, centers=2)
K-means clustering with 2 clusters of sizes 22, 28

Cluster means:
    Murder     Rape   Robbery  Assault Burglary    Theft  Vehicle
1 2.711724 3.132781 2.1001609 2.836238 3.680055 4.479685 2.642471
2 1.371839 1.712574 0.6770169 1.308089 2.203667 3.411210 1.177732

Clustering vector:
ME NH VT MA RI CT NY NJ PA OH IN IL MI WI MN IA MO ND SD NE KS DE MD VA WV NC
 2  2  2  1  2  2  1  1  2  2  2  1  1  2  2  2  1  2  2  2  2  2  1  2  2  2
SC GA FL KY TN AL MS AR LA OK TX MT ID WY CO NM AZ UT NV WA OR CA AK HI
 1  1  1  2  1  2  2  2  1  1  1  2  2  2  1  1  1  2  1  1  1  1  1  2

Within cluster sum of squares by cluster:
[1] 102.3351  72.6241
 (between_SS / total_SS =  49.0 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Next, to visualize the result, I plot the two-group solution in the space of the first two principal components of the correlation matrix of the data.

```
> crime_pca <- prcomp(df_s)
> plot(crime_pca$x[,1:2],
+      pch = kmeans(df_s, centers = 2)$cluster)
```



The result is similar to the plot found in the textbook. It suggests that the cluster analysis here is dividing into two parts a homogeneous set of data.

0.3 Ex 6.2

Calculate the first five principal components scores for the Romano- British pottery data, and then construct the scatterplot matrix of the scores, displaying the contours of the estimated bivariate density for each panel of the plot and a boxplot of each score in the appropriate place on the diagonal. Label the points in the scatterplot matrix with their kiln numbers.

MY SOLUTION:

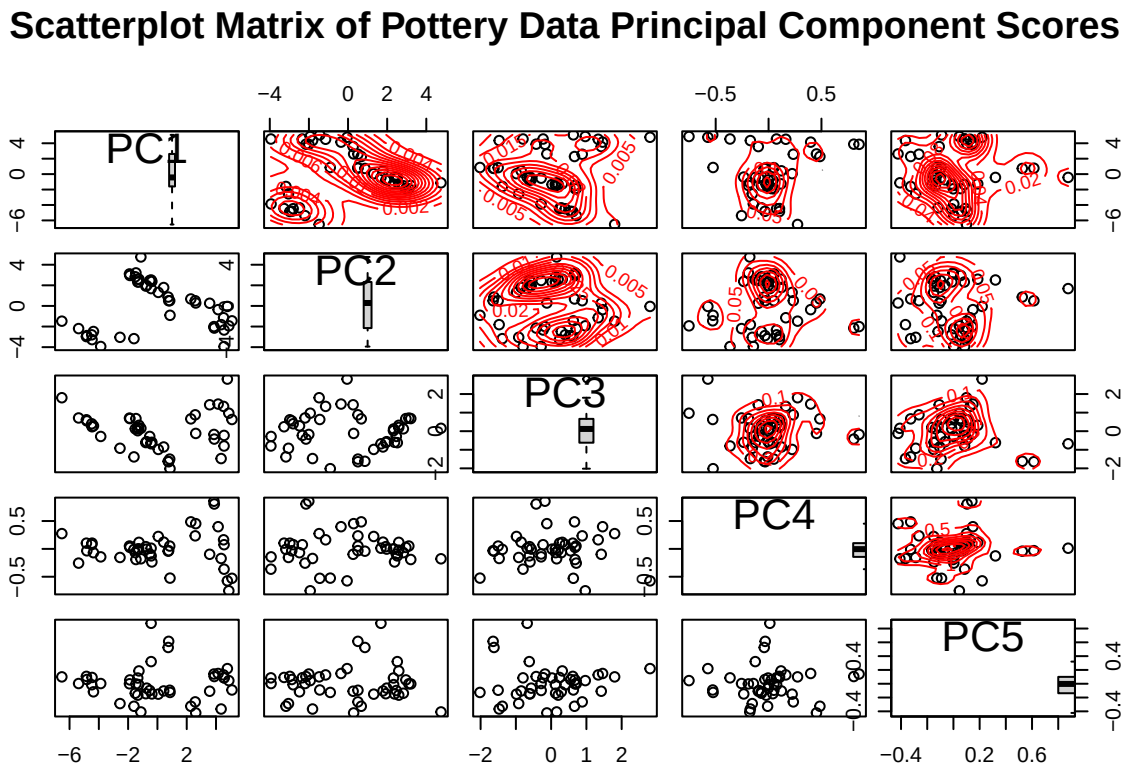
Since the chemical elements are different scales, it is necessary to standarize them first.

```
> # import the data
> data("pottery", package = "HSAUR2")
> dim(pottery)
[1] 45 10
> # run PCA for five-components solution;
> # since the chemical elements are at different scales, standardization is necessary.
>
> pca <- prcomp(pottery[,-10],center = TRUE)
> scores <- pca$x[,1:5]
```

Next, I use the scores of the first five principal components scores to construct the scatterplot matrix of the scores.

```
> # create a scatterplot matrix with density contours
> pairs(scores,
+       main = "Scatterplot Matrix of Pottery Data Principal Component Scores",
+       upper.panel = function(x, y) {
+         points(x, y)
+         den <- MASS::kde2d(x,y)
+         contour(den, add = TRUE,col="red",lwd=1)
+       },
```

```
+       diag.panel = function(x) {
+         boxplot(x, horizontal = F, add = TRUE)
+       },
+       lower.panel = function(x, y) {
+         points(x, y)})
> # Add kiln number labels to the scatterplot matrix
> text(scores, labels = pottery$kiln, pos = 1, cex = 0.1)
```



I do not know why I can't label the point with the `kiln` categorical variable.

0.4 Ex 6.3

Return to the air pollution data given in Chapter 1 and use finite mixtures to cluster the data on the basis of the six climate and ecology variables (i.e., excluding the sulphur dioxide concentration). Investigate how sulphur dioxide concentration varies in the clusters you find both graphically and by formal significance testing.

MY SOLUTION:

```
> # import the data
> data("USairpollution", package = "HSAUR2")
> dim(USairpollution)
[1] 41  7
> colnames(USairpollution)
[1] "SO2"     "temp"    "manu"    "popul"   "wind"    "precip"  "predays"
>
```

```
> # before running the finite mixture approach, standardize the data first
> air_s <- scale(USairpollution[,-1])
>
> # run the finite mixtures to the cluster the data
> # install.packages("mclust")
> library(mclust)
> mc <- Mclust(air_s)
> plot(mc$BIC)
```