2019 학년도 1학기
DATA MINING
HW9

과목명	데이터마이닝
담당교수명	송종우 교수님
제출일	2019.05.22
학번	182STG27
이름	임지연

Ⅰ. Description

10장에서는 Unsupervised Learning 방법에 대해서 공부할 것이다. 대표적인 방법인 PCA, Clustering 과 같은 방법들의 비교를 해볼 것이다. 각각의 모형에서 사용되는 방법의 원리에 대해서 알아본 후 Lab에 설명되어 있는 코드를 실행해 보며 함수를 공부한 후, 결과를 이해한 후 연습문제를 풀어볼 것이다.

Ⅱ. Implementation

Question 1.

Lab : PCA, Clustering, NCI60 Data Example의 코드를 실행해보고 감상문을 써라

PCA 방법을 수행하기 위하여 USArrets 자료를 이용하여 prcomp() 함수 사용법에 대해 알 수 있었다. Scale = TRUE 라는 옵션을 지정해 주면 자동으로 표준화 해준 후 주성분분석을 진행한다.

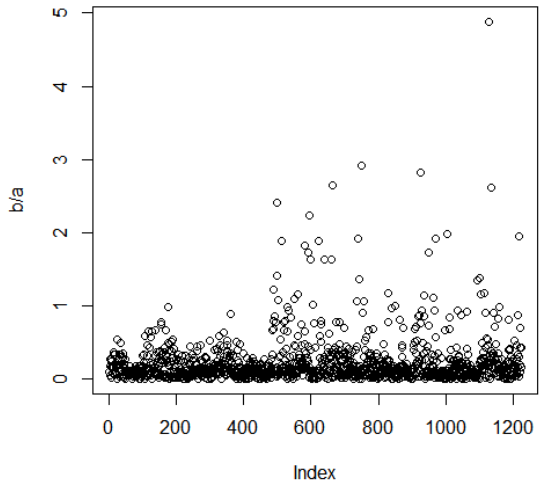
Clustering을 수행하기 위하여 kemans() 함수에 대해 공부할 수 있었다. Nstart 옵션을 이용해 초기에 점을 할당하는 횟수를 지정해 줄 수 있다. 마지막으로 NCL60 데이터에 대해 직접 함수를 적용해보며 공부할 수 있었다.

Question 2.

9.7 Excercises - Example 4, 5, 8 풀어라

[Example 7]

In the chapter, we mentioned the use of correlation-based distance and Euclidean distance as dissimilarity measures for hierarchical clustering. It turns out that these two measures are almost equivalent: if each observation has been centered to have mean zero and standard deviation one, and if we let rij denote the correlation between the ith and jth observations, then the quantity 1−rij is proportional to the squared Euclidean distance between the ith and jth observations. On the USArrests data, show that this proportionality holds. Hint: The Euclidean distance can be calculated using the dist() function, and correlations can be calculated using the cor() function.



Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.000	0.069	0.134	0.234	0.263	4.888

▶ correlation-based distance와 Euclidean distance의 비 값을 살펴본 후 그래프를 그려봤을 때 ,Median값은 약 0.134로 USArrests 데이터에서 비례관계가 성립하고 있다는 것을 알 수 있다.

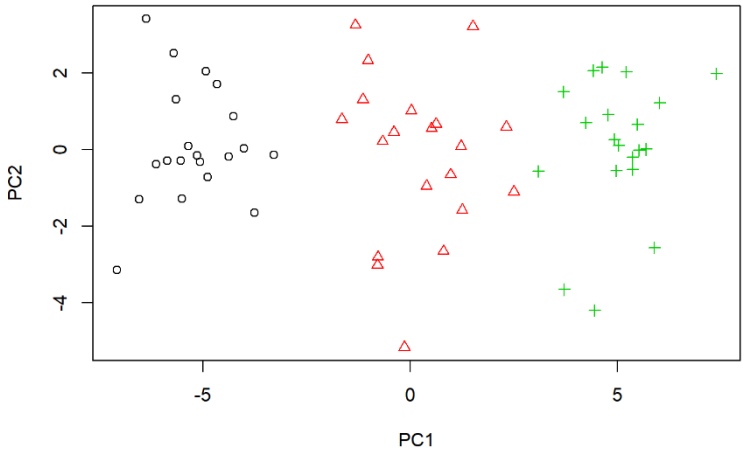
[Example 10]

In this problem, you will generate simulated data, and then perform PCA and K-means clustering on the data.

(a) Generate a simulated data set with 20 observations in each of three classes (i.e. 60 observations total), and 50 variables.

Hint: There are a number of functions in R that you can use to generate data. One example is the rnorm() function; runif() is another option. Be sure to add a mean shift to the observations in each class so that there are three distinct classes.

(b) Perform PCA on the 60 observations and plot the first two principal component score vectors. Use a different color to indicate the observations in each of the three classes. If the three classes appear separated in this plot, then continue on to part (c). If not, then return to part (a) and modify the simulation so that there is greater separation between the three classes. Do not continue to part (c) until the three classes show at least some separation in the first two principal component score vectors.



▶ 데이터를 그래프로 그려봤을 때, 세 그룹이 PCA1, PCA2 로부터 잘 나누어져 있는 것을 알 수 있다.

(c) Perform K-means clustering of the observations with K = 3. How well do the clusters that you obtained in K-means clustering compare to the true class labels?

Hint: You can use the table() function in R to compare the true class labels to the class labels obtained by clustering. Be careful how you interpret the results: K-means clustering will arbitrarily number the clusters, so you cannot simply check whether the true class labels and clustering labels are the same.

		True class		
		1	2	3
Clustering label	1	0	18	0
	2	0	2	20
	3	20	0	0

▶2 개의 관측치가 오분류된 것으로 보인다.

(d) Perform K-means clustering with K = 2. Describe your results.

		True class		
		1	2	3
Predict class	1	0	16	20
	2	20	4	0

▶가운데 class중 4개의 관측치가 오분류된 것으로 보인다. 하지만 이외의 class는 잘 분류한다.

(e) Now perform K-means clustering with K = 4, and describe your results.

		True class		
		1	2	3
Predict class	1	0	2	20
	2	0	13	0
	3	20	0	0
	4	0	5	0

▶4개의 class로 구분했을 때, 한가지 클래스가 두가지의 클래스로 나뉜다.

(f) Now perform K-means clustering with K = 3 on the first two principal component score vectors, rather than on the raw data. That is, perform K-means clustering on the 60 × 2 matrix of which the first column is the first principal component score vector, and the second column is the second principal component score vector. Comment on the results.

		True class		
		1	2	3
Predict class	1	0	0	20
	2	0	20	0
	3	20	0	0

▶ 첫번째, 두번째 주성분을 이용하여 k-means clustering방법을 수행해 본 결과, 매우 정확하게 잘 분류해냈다. 따라서 많은 정보를 잘 포함하고 있다는 것을 알 수 있다.

(g) Using the scale() function, perform K-means clustering with K = 3 on the data after scaling each variable to have standard deviation one. How do these results compare to those obtained in (b)? Explain.

		True class		
		1	2	3
Predict class	1	0	2	20
	2	0	18	0
	3	20	0	0

▶표준화해 본 결과 b에 비하여 성능이 개선되지 못했다. 따라서, 이 데이터에서 scaling 은 부적절하다는 것을 알 수 있다.

Question 3.

A quick tour of mclust 를 실행해보고 감상문을 써라.

Mclust 패키지를 이용해볼 수 있었다. mclustCL() 함수는 ICL 값에 따른 최적의 분산구조와 클래스 개수를 구할 수 있다. 또한 mclustBICupdate()를 통해서 최적의 BIC 값도 찾을 수 있었다. 그 외에도 다양한 옵션이 있다는 것을 공부할 수 있었다.

Ⅲ. Discussion

Unsupervised learning 방법에 대해 살펴봤다. 대표적으로 PCA, Clustering을 이용하여 각각의 장단점을 알 수 있었다. 또한 변수의 표준화가 clustering을 하는 데 미치는 영향에 대해서도 공부할 수 있었다. Lab을 통해 전반적인 함수 사용법에 대해 공부할 수 있었고, 문제를 풀며 복습할 수 있었다. 항상 scaling 을 하는 것이 더 성능을 좋게 하는 것이 아니며 적당한 방법을 사용해야 한다는 것을 알았다.

Ⅳ. Appendix – R code

```
## 7번 문제
```{r }
library(ISLR)
set.seed(1)
a = dist(scale(USArrests))^2 # 유클리드 거리의 제곱
b = as.dist(1 - cor(t(scale(USArrests)))) #
summary(b/a)
plot(b/a)
```
10번 문제
```{r }
#a
set.seed(12)
X <- rbind(matrix(rnorm(20*50, mean = 0), nrow = 20),
matrix(rnorm(20*50, mean=0.7), nrow = 20),
matrix(rnorm(20*50, mean=1.4), nrow = 20))
#b
X.pca = prcomp(X)$x
plot(X.pca[,1:2], col=c(rep(1,20), rep(2,20), rep(3,20)), pch =c(rep(1,20), rep(2,20), rep(3,20)))
#c    #k=3
res = kmeans(X, centers = 3)
true_class = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
#d    #k=2
res = kmeans(X, centers = 2)
true = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
#e    #k=4
res = kmeans(X, centers = 4)
true = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
#f   #1,2번째PCA에 대해 clustering
res = kmeans(X.pca[,1:2], centers = 3)
true = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
#g
res = kmeans(scale(X), centers = 3)
true = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
# b가 good
```
```