데이터과학을 위한 R프로그래밍

12주차. 군집분석

이혜선 교수
포항공과대학교 산업경영공학과

목차

12주차. 군집분석

12주차

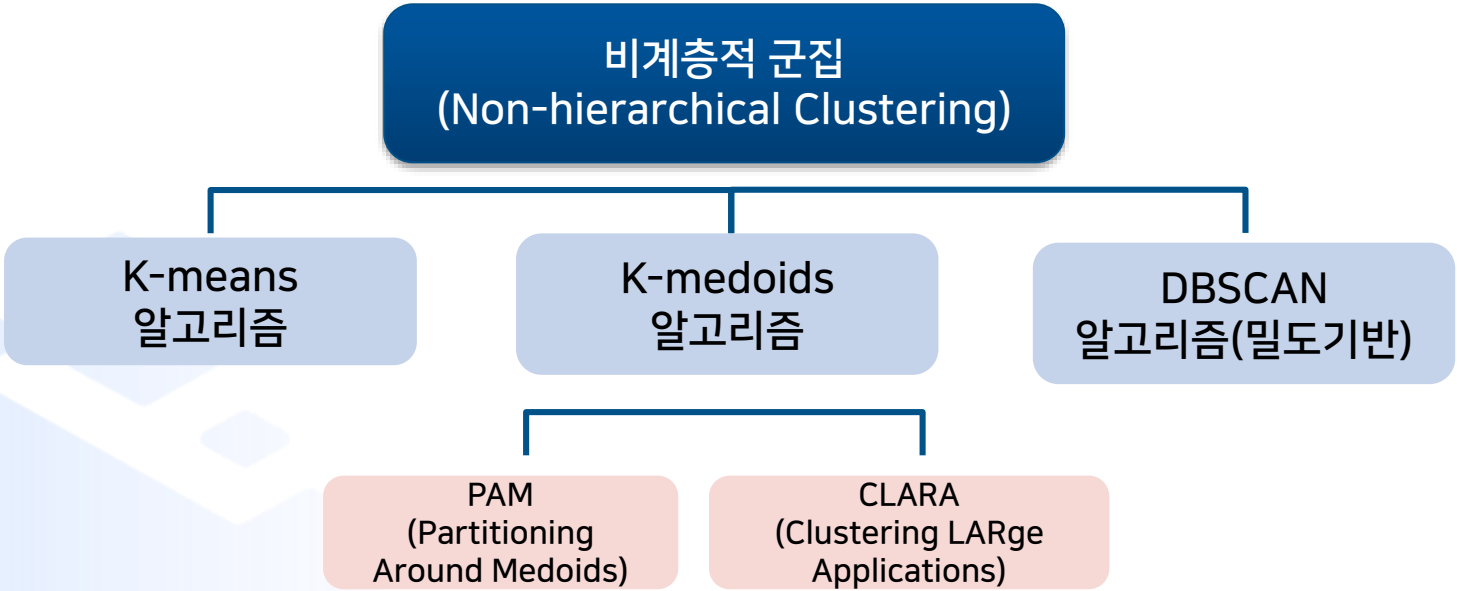
3차시

비계층적 군집분석

POSTECH

비계층적 군집분석

☑ 사전에 군집 수 k를 정한 후 각 객체를 k개 중 하나의 군집에 배정



k-means 군집분석

- k-means 군집분석은 비계층적 군집분석 중 가장 널리 사용되는 기법
  - k개 군집의 중심좌표를 고려하여 각 객체를 가장 가까운 군집에 배정을 반복

단계 0
(초기 객체 선정)
k개 객체 좌표를 초기 군집 중심좌표로 선정

단계 1
(객체 군집 배정)
각 객체와 k개 중심좌표와의 거리 산출 후, 가장 가까운 군집에 객체 배정

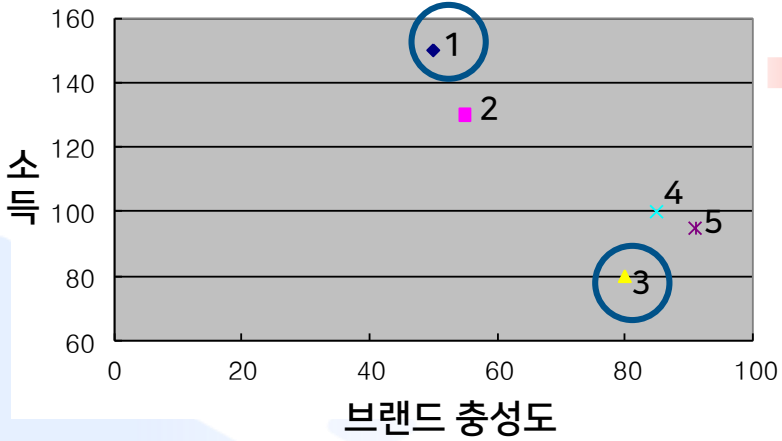
단계 2
(군집 중심좌표 산출)
새로운 군집의 중심좌표 산출

단계 3
(수렴 조건 점검)
새로 산출된 중심 좌표값과 이전 좌표값을 비교
수렴 조건 내에 들면 종료, 그렇지 않으면 단계 1 반복

POSTECH

k-means 군집분석 예제

☑ k-means 알고리즘을 적용(군집 수 k=2라 가정)



Step 0. 초기 객체 선정

임의의 두 객체 Obs1, Obs3 선정

Step 1. 객체 군집 배정

ID	1	3
1	0.0	76.2
2	20.6	<55.9
3	76.2	0.0
4	61.0	>20.6
5	68.6	>18.6

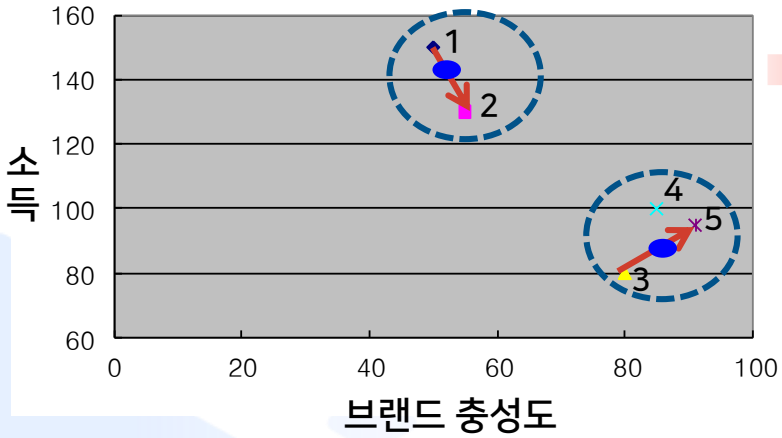
C1
Obs1, Obs2

C2
Obs3, Obs4, Obs5

POSTECH

k-means 군집분석 예제

☑ k-means 알고리즘을 적용(군집 수 k=2라 가정)



Step 2. 군집 중심좌표 산출

	C1	C2
객체	Obs1, Obs2	Obs3, Obs4, Obs5
중심좌표	$(\frac{50+55}{2}, \frac{150+130}{2})$ = (52.5, 140)	$(\frac{80+85+91}{3}, \frac{80+100+95}{3})$ = (85.33, 91.67)

Step 3. 수렴 조건 점검

ID	C1	C2
1	10.3	68.2
2	15.2	48.9
3	66.0	12.8
4	51.5	8.3
5	59.2	6.6

C1
Obs1, Obs2

C2
Obs3, Obs4, Obs5

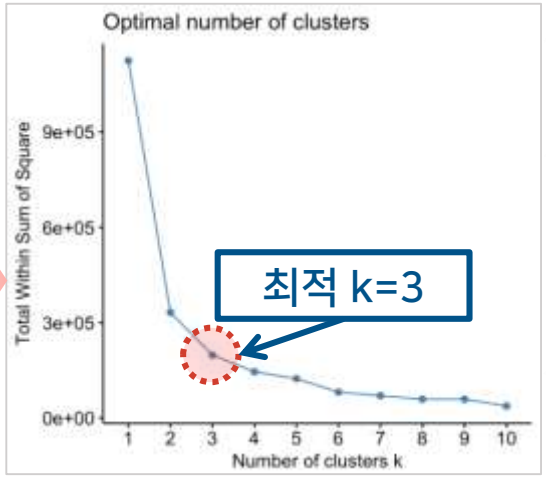
이전 군집결과와 변화 없으므로,
2개의 군집을 형성

POSTECH

k-means 군집분석

군집수 k 정하기

```
# Clustering
# Non-hierarchical Clustering
# k-means, PAM, DBSCAN

# set working directory
setwd("D:/tempstore/moocr/wk13")

# read csv file
wages1833<-read.csv(file="wages1833.csv", string
head(wages1833)

# preprocessing
# delete ID variable
dat1<-wages1833[ , -1]
# delete missing data
dat1<-na.omit(dat1)
head(dat1, n=5)

# to choose the optimal k, silhouette
install.packages("factoextra")
library(factoextra)
library(ggplot2)

fviz_nbclust(dat1, kmeans, method = "wss")
fviz_nbclust(dat1, kmeans, method = "gap_stat")
```



- 최적 군집수에 대한 시각화
- 최적값은 “silhouette”, “gap_stat”, “wss(그룹내제곱합)”으로 산출
- 그래프가 완만해지는 지점을 k의 값으로 추정

POSTECH

k-means 군집분석

k-means (k=3)

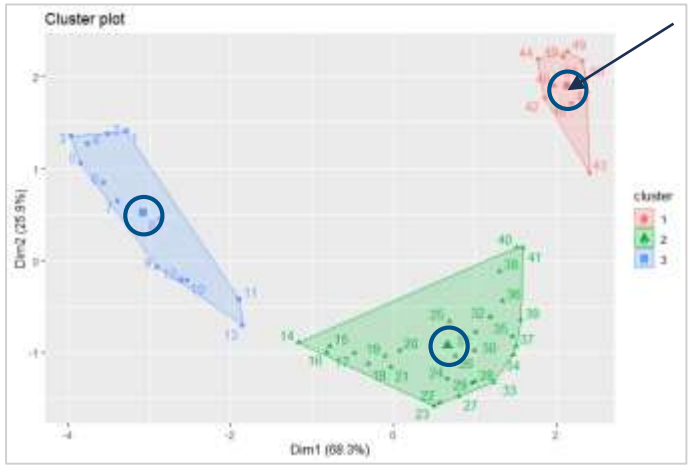
random set의 수(nstart)

k

```
# compute kmeans
set.seed(123)
km <- kmeans(dat1, 3, nstart = 25)
km

# visualize
fviz_cluster(km, data = dat1,
             ellipse.type="convex",
             repel = TRUE)
```

```
> km
K-means clustering with 3 clusters of sizes 10, 28, 13

Cluster means:
   age      mnum      mwage      fnum     fwage
1 55.5    6.9000 178.99000   0.00000    0.0000
2 36.5   43.2500 241.73214  31.21429  107.9643
3 16.0  187.2308  96.36154 225.23077   71.0000
```

중심좌표



- kmeans 결과 시각화
- Convex 모양으로 구역 표시
- repel을 통해 관측치 표기

POSTECH

k-medoids 군집분석

k-medoids 군집분석은 중앙값을 각 군집의 대표 객체로 사용

- K-medoids 군집분석은 객체들을 K개의 군집으로 구분하는데,
  객체와 속하는 군집의 대표 객체와의 거리 총합을 최소로 하는 방법

PAM 알고리즘

모든 객체에 대하여 대표 객체가 변했을 때 발생하는 거리 총합의 변화를 계산
- 데이터 수가 많아질수록 연산량이 크게 증가함

CLARA 알고리즘

적절한 수의 객체를 샘플링 한 후, PAM 알고리즘을 적용하여 대표 객체 선정
- 샘플링을 여러 번 한 후 가장 좋은 결과를 택함
- 편향된 샘플링은 잘못된 결과값을 도출할 수 있음

출처 전치혁, 데이터마이닝 기법과 응용, 2012, 한나래출판사 참조

POSTECH

PAM(Partitioning Around Medoids) 알고리즘

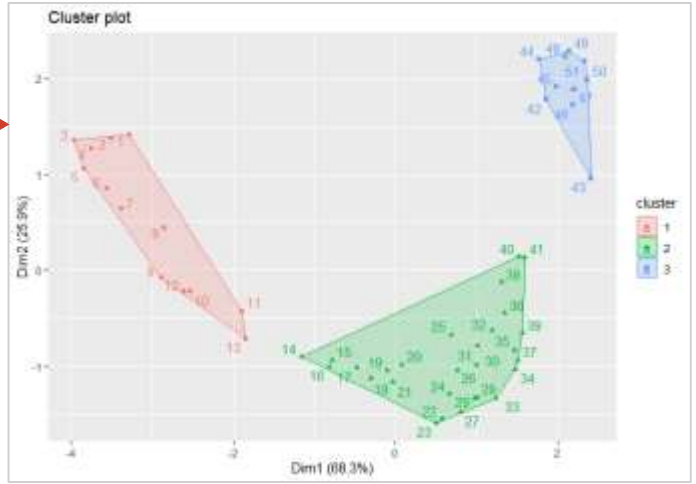
PAM (k=3)

```
# compute PAM
library("cluster")
pam_out <- pam(dat1, 3)
pam_out

# freq of each cluster
table(pam_out$clustering)

# visualize
fviz_cluster(pam_out, data = dat1,
             ellipse.type="convex",
             repel = TRUE)
```

```
> pam_out
Medoids:
   ID age mnum mwage fnum fwage
7   7  16  204  83.5  256    72
31 31  40   38 243.5   15   104
45 45  54   12 174.0    0     0
```

대표 객체

```
> # freq of each cluster
> table(pam_out$clustering)

 1  2  3
13 28 10
```



DBSCAN 알고리즘

(Density Based Spatial Clustering of Application with Noise)

밀도 기반 군집화 알고리즘

- 특정 공간 내에 데이터의 밀도 차이를 기반으로 군집화
- 복잡한 기하학적 분포도를 가진 데이터도 군집화를 잘 수행
- k-means와 다르게 군집 수를 지정할 필요가 없고, 자동으로 군집 수를 찾음

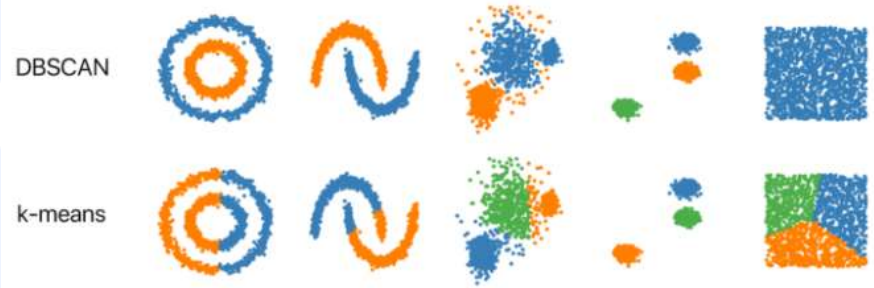


사진 출처 : https://scikit-learn.org/stable/modules/clustering.html

POSTECH

DBSCAN 알고리즘

(Density Based Spatial Clustering of Application with Noise)

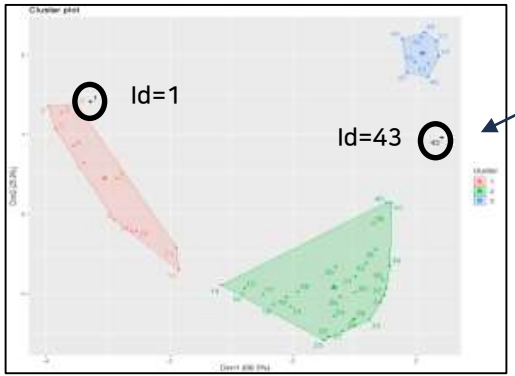
```
# 3. DBSCAN
install.packages("fpc")
library(fpc)
db<-dbscan(dat1, eps=70, MinPts=3)

# result of clustering
db
db$cluster

#visualization
fviz_cluster(db, data = dat1,
             ellipse.type="convex",
             repel = TRUE)
```

```
> db
dbscan Pts=51 MinPts=3 eps=70
       0  1  2 3
border 2  1  0 0
seed   0 11 28 9
total  2 12 28 9
```

각 클러스터에 border point 개수

각 클러스터에 seed 개수

각 클러스터에 포함된 데이터 전체 개수



Min points(M) : 군집을 이루기 위한 $N_{eps}(p)$ 의 최소 원소 개수

Eps : 군집을 이루기 위한 최소 반경 (Neighborhood of point p)

비계층적 군집분석

실루엣계수(Silhouette coefficient) : 군집의 성능평가척도

$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)}$$

a_i= i번째 데이터와 같은 군집 안에 있는 다른 데이터들과 평균 거리(dissimilarity)

b_i= i번째 데이터와 다른 군집과의 평균 거리 중 작은 거리

- 실루엣 계수는 한 군집 안의 데이터들이 다른 군집 비교해서 얼마나 비슷한 가를 나타냄
- 군집 안의 거리가 작을수록 좋고, 다른 군집과 거리는 클수록 좋음
- 실루엣 계수는 -1부터 1사이의 값을 가짐 => 클수록 좋음

비계층적 군집분석

실루엣계수(Silhouette coefficient)

- 평균 실루엣 계수가 클수록 군집화가 잘 되었다고 할수 있음

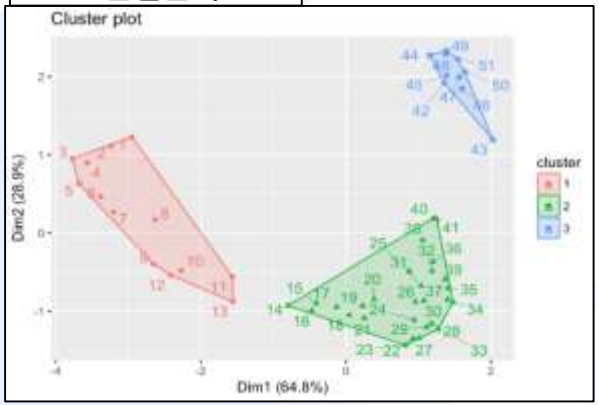
```
# calculate silhouette
library(cluster)
sil_pam<- silhouette(pam_out$clustering, dist(dat1))
mean(sil_pam)
sil_db<- silhouette(db$cluster, dist(dat1))
mean(sil_db)
```

```
> mean(sil_pam)
[1] 1.679085
```

```
> mean(sil_db)
[1] 1.518474
```

PAM의 실루엣 계수 값이 더큼

PAM 군집분석



DBSCAN