군집분석(Cluster Analysis)

http://contents.kocw.net/KOCW/document/2015/chungbuk/najonghwa1/13.pdf
http://contents.kocw.net/KOCW/document/2015/chungbuk/najonghwa1/14.pdf

데이터 간의 유사도를 정의하고, 그 유사도에 가까운 것부터 순서대로 합쳐 가는 방법으로 그룹(군집)을 형성한 후 각 그룹의 성격을 파악하거나 그룹 간의 비교분석을 통해서 데이터 전체의 구조에 대한 이해를 돕고자 하는 탐색적인 분석 방법

유사도: 거리(distance)를 이용하는데 거리의 종류는 다양하지만, 가장 일반적으로 사용하는 것이 유클리디안(Euclidean)거리로 측정한 거리정보를 이용해서 분석대상을 몇 개의 집단으로 분류

군집 분석의 목적: 데이터 셋 전체를 대상으로 서로 유사항 개체 들을 몇 개의 군집으로 세분화하여 대상 집단을 정확하게 이해하고, 효율적으로 활용하기 위함.

군집 분석으로 그룹화된 군집은 변수의 특성이 그룹 내적으로는 동일하고, 외적으로는 이질적인 특성을 갖는다.

군집 분석의 용도는 고객의 충성도에 따라서 몇 개의 그룹으로 분류하고, 그룹별로 맞춤형 마케팅 및 프로모션 전략을 수립하는 데 활용된다.

군집 분석에서 중요한 사항:

- 군집화를 위해서 거리 측정에 사용되는 변인은 비율척도나 등간척도여야 하며, 인구통계적 변인, 구매패턴 변인, 생활패턴 변인 등이 이용된다.
- 군집 분석에 사용되는 입력 자료는 변수의 측정단위와 관계없이 그 차이에 따라 일정하게 거리를 측정하기 때문에 변수를 표준화하여 사용하는 것이 필요하다.
- 군집화 방법에 따라 계층적 군집 분석과 비계층적 군집분석으로 분류된다.

군집 분석에 이용되는 변인

- 인구 통계적 변인: 거주지, 성별, 나이, 교육수준, 직업, 소득수준 등
- 구매패턴 변인: 구매상품, 1회 평균 거래액, 구매횟수, 구매주기 등
- 생활패턴 변인: 생활습관, 가치관, 성격, 취미 등

군집 분석의 특징

- 전체적인 데이터 구조를 파악하는데 이용된다.

- 관측대상 간 유사성을 기초로 비슷한 것끼리 그룹화(clustering)한다.
- 유사성은 유클리디안 거리를 이용한다.
- 분석 결과에 대한 가설검정이 없다.
- 반응변수(y변수)가 존재하지 않는 데이터마이닝 기법이다.
- 규칙(Rule)을 기반으로 계층적인 트리구조를 생성한다.
- 활용분야: 구매패턴에 따른 고객 분류, 충성도에 따른 고객 분류 등

군집 분석의 절차

1) 분석대상의 데이터에서 군집 분석에 사용할 변수 추출
2) 계층적 군집 분석을 이용한 대략적인 군집의 수 결정
3) 계층적 군집 분석에 대한 타당성 검증(ANOVA분석)
4) 비계층적 군집 분석을 이용한 군집 분류
5) 분류된 군집의 특성 파악 및 업무 적용

군집의 종류:

1. 계층적 군집(Hierarchical Clustering): 트리 구조처럼 분리하는 방법

1) 병합(agglomeration)방법
단일(최단)연결법(Single Linkage Method)
완전(최장)연결법(Complete Linkage Method)
평균연결법(Average Linkage Method)
중심연결법(Centroid Linkage Method)
Ward 연결법(Ward Linkage Method)

2) 분할(Division)방법
다이아나 방법(DIANA Method)

2. 분할적 군집: 특정 점을 기준으로 가까운 것끼리 묶는 방법

1) 프로토타입(Prototype Based)
K-중심군집(K-Medoids Clustering)
퍼지군집(Fuzzy Clustering)

2) 밀도기반(Density Based)
중심 밀도 군집(Center Density Clustering)
격자 기반 군집(Grid Based Clustering)
커널 기반 군집(Kernel Based Clustering)

3) 분포기반(Distribution Based)
혼합분포 군집(Mixture Distribution Clustering)

4) 그래프 기반(Graph Based)
코호넨 군집(Kohonen Clustering)

1.1 유클리디안 거리

유클리디안 거리 계산식

실습 (유클리디안 거리 계산법)

1단계: matrix 객체 생성

```
x <- matrix(1:9, nrow = 3, by = T)
x
```

2단계: 유클리디안 거리 생성

```
dist <- dist(x, method = "euclidean")
dist
```

dist()함수
형식: dist(x, method = "euclidean")
https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/dist
다른 거리 계산
https://cran.r-project.org/web/packages/philentropy/vignettes/Distances.html

실습 (1행과 2행 변량의 유클리디안 거리 구하기)

```
s <- sum((x[1, ] - x[2, ]) ^ 2)
sqrt(s)
```

실습 (1행과 3행 변량의 유클리디안 거리 구하기)

```
s <- sum((x[1, ] - x[3, ]) ^ 2)
sqrt(s)
```

1.2 계층적 군집 분석

계층적 군집 분석(Hierarchical Clustering): 개별대상 간의 거리에 의하여 가장 가까운 대상부터 결합하여 나무 모양의 계층구조를 상향식(Bottom-up)으로 만들어가면서 군집을 형성

군집 대상 간의 거리를 산정하는 기준에 따라
단일(최단)연결법(Single Linkage Method): 최소거리 기준
완전(최장)연결법(Complete Linkage Method): 최대거리 기준
평균연결법(Average Linkage Method): 평균 거리 기준
중심연결법(Centroid Linkage Method): 중심값의 거리 기준
Ward 연결법(Ward Linkage Method): 유클리디안 제공 거리 기준

장점: 군집이 형성되는 과정을 파악
단점: 자료의 크기가 큰 경우 분석이 어렵다

실습 (유클리디안 거리를 이용한 군집화)

1단계: 군집 분석(Clustering)을 위한 패키지 설치

```
install.packages("cluster")
library(cluster)
```

cluster 패키지 설치

2단계: 데이터 셋 생성

```
x <- matrix(1:9, nrow = 3, by = T)
```

3단계: matrix 객체 대상 유클리디안 거리 생성

```
dist <- dist(x, method = "euclidean")
```

4단계: 유클리디안 거리 matrix를 이용한 군집화

```
hc <- hclust(dist)
```

hclust()함수
https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/hclust

5단계: 클러스터 시각화

```
plot(hc)
```

덴드로그램(dendrogram)

실습 (신입사원의 면접시험 결과를 군집 분석)

1단계: 데이터 셋 가져오기

```
interview <- read.csv("C:/Rwork/interview.csv", header = TRUE)
names(interview)
head(interview)
```

2단계: 유클리디안 거리 계산

```
interview_df <- interview[c(2:7)]
idist <- dist(interview_df)
head(idist)
```

3단계: 계층적 군집 분석

```
hc <- hclust(idist)
hc
```

4단계: 군집 분석 시각화

```
plot(hc, hang = -1)
```

plot()함수의 "hang = -1" 속성을 이용하여 덴드로그램에서 음수값 제거

5단계: 군집 단위 테두리 생성

```
rect.hclust(hc, k = 3, border ="red")
```

rect.hclust()함수
https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/rect.hclust

실습 (군집별 특징 보기)

1단계: 군집별 서브 셋 만들기

```
g1 <- subset(interview, no == 108 | no == 110 | no == 107 |
                    no == 112 | no == 115)
g2 <- subset(interview, no == 102 | no == 101 | no == 104 |
                    no == 106 | no == 113)
g3 <- subset(interview, no == 105 | no == 114 | no == 109 |
                    no == 103 | no == 111)
```

2단계: 각 서브 셋의 요약통계량 보기

```
summary(g1)
summary(g2)
summary(g3)
```

군집분석으로 그룹화된 군집은 다변량적 특성이 그룹 내적으로는 동일하고, 외적으로는 이질적인 특성을 갖는다.

[표 16.1] 군집별 특징 요약
그룹별 요약통계량, 자격증 유무, 군집 특징

1.3 군집 수 자르기

계층형 군집 분석 결과에서 분석자가 원하는 군집 수만큼 잘라서 인위적으로 군집을 만들 수 있다.

실습 (iris 데이터 셋을 대상으로 군집 수 자르기)

1단계: 유클리디안 거리 계산

```
idist <- dist(iris[1:4])
hc <- hclust(idist)
plot(hc, hang = -1)
```

2단계: 군집 수 자르기

```
ghc <- cutree(hc, k = 3)
ghc
```

그룹수를 자르는 함수는 stats패키지에서 제공되는 cutree()함수 이용

형식: cutree(rPcmdwjr 군집 분석 결과, k=군집 수)
https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/cutree

3단계: iris데이터 셋에 ghc컬럼 추가

```
iris$ghc <- ghc
table(iris$ghc)
head(iris)
```

4단계: 요약통계량 구하기

```
g1 <- subset(iris, ghc == 1)
summary(g1[1:4])
g2 <- subset(iris, ghc == 2)
summary(g2[1:4])
g3 <- subset(iris, ghc == 3)
summary(g3[1:4])
```

1.4 비계층적 군집 분석

군집의 수가 정해진 상태에서 군집의 중심에서 가장 가까운 개체를 하나씩 포함해 나가는 방법.

대표적인 방법이 K-means clustering

K-means clustering 방법은 군집 수를 미리 알고 있는 경우 군집 대상의 분포에 따라 군집의 초기값을 설정해 주면, 초기값에서 가장 가까운 거리에 있는 대상을 하나씩 더해가는 방식.

계층적 군집 분석을 통해 대략적인 군집의 수를 파악하고 이를 초기 군집 수로 설정하여 비계층적 군집 분석을 수행하는 것이 효과적

장점: 대량의 자료를 빠르고 쉽게 분류 가능
단점: 군집의 수를 미리 알고 있어야 한다.

실습 (K-means 알고리즘에 군집 수를 적용하여 군집별로 시각화)

ggplot2패키지에서 제공하는 diamonds데이터 셋 대상
계층적 군집 분석으로 군집 수를 파악
K-means 알고리즘에 군집 수를 적용하여 군집별로 시각화

1단계: 군집 분석에 사용할 변수 추출

```
library(ggplot2)
data(diamonds)
t <- sample(1:nrow(diamonds), 1000)
test <- diamonds[t, ]
dim(test)
head(test)

mydia <- test[c("price", "carat", "depth", "table")]
head(mydia)
```

2단계: 계층적 군집 분석(탐색적 분석)

```
result <- hclust(dist(mydia), method = "average")
result
plot(result, hang = -1)
```

3단계: 비계층적 군집 분석

```
result2 <- kmeans(mydia, 3)
names(result2)
result2$cluster

mydia$cluster <- result2$cluster
head(mydia)
```

kmeans()함수
https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/kmeans

4단계: 변수 간의 상관계수 보기

```
cor(mydia[ , -5], method = "pearson")
plot(mydia[ , -5])
```

cor()함수
https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/cor

5단계: 상관계수를 색상으로 시각화

```
install.packages("mclust")
library(mclust)
install.packages("corrgram")
library(corrgram)
corrgram(mydia[ , -5], upper.panel = panel.conf)
corrgram(mydia[ , -5], lower.panel = panel.conf)
```

mclust 패키지 설치
corrgram 패키지 설치

corrgram()함수
https://www.rdocumentation.org/packages/corrgram/versions/1.13/topics/corrgram
주요 군집분석 실습(iris데이터 이용)

6단계: 비계층적 군집 시각화

```
plot(mydia$carat, mydia$price, col = mydia$cluster)
points(result2$centers[ , c("carat", "price")],
       col = c(3, 1, 2), pch = 8, cex = 5)
```

1) 계층적 군집법

계층적 군집 분석을 수행하는 과정은 주어진 데이터를 순차적으로 가까운 값들끼리 묶어 주는 병합 방법과 관측 값을 나누는 분할 방법이 있다. 주로 병합 방법을 사용

실습.

```
====================
### Hierarchical Clustering

# iris 데이터셋 로딩(50개)
idx <- sample(1:dim(iris)[1], 50)
idx
irisSample <- iris[idx, ]
head(irisSample)

# species 데이터 제거
irisSample$Species <- NULL
head(irisSample)

# 계층적 군집법 실행
hc_result <- hclust(dist(irisSample), method="ave")
hc_result

# 군집 결과 시각화
plot(hc_result, hang=-1, labels = iris$Species[idx])

rect.hclust(hc_result, k=4)

========================
```

2) K 평균 군집법(K-means clustering)

k개의 평균(mean)을 구하여 각 군집(클러스터)은 평균 값으로 대표된다.
N개의 데이터가 주어졌을 때, K개의 군집(클러스터)으로 분할하는 방법
각 데이터는 평균점과의 거리가 가장 가까운 군집(클러스터)에 속하게 된다.

실습.

```
====================
### k-means 실습

# iris 데이터셋 로딩
data(iris)
iris
iris2 <- iris

# species 데이터 제거
iris2$Species <- NULL
head(iris2)

# k-means clustering 실행
kmeans_result <- kmeans(iris2, 6)
kmeans_result
str(kmeans_result)


# 군집 결과 시각화
plot(iris2[c("Sepal.Length", "Sepal.Width")], col=kmeans_result$cluster)
points(kmeans_result$centers[, c("Sepal.Length", "Sepal.Width")], col=1:4, pch=8, cex=2)

plot(iris2[c("Petal.Length", "Petal.Width")], col=kmeans_result$cluster)
points(kmeans_result$centers[, c("Petal.Length", "Petal.Width")], col=1:4, pch=8, cex=2)


# 군집의 수 결정
kmeans_result <- kmeans(iris2, 7)
plot(iris2[c("Sepal.Length", "Sepal.Width")], col=kmeans_result$cluster)
points(kmeans_result$centers[, c("Sepal.Length", "Sepal.Width")], col=1:4, pch=8, cex=2)
```

==========

3) K-중심 군집(K-Medoids Clustering)

K-Means clustering 과 유사
K-Means clustering: 임의의 좌표를 중심점으로 잡음
K-Medoids clustering: 실제 점 하나를 중심점으로 잡아서 계산 수행
PAM(Partitioning Around Medoids)알고리즘이 대표적.

실습.

```
====================

### K-Medoids Clustering

#패키지 설치
install.packages("fpc")
library(fpc)

# 군집의 수 설정. 임의로 6개
pamk_result <- pamk(iris2, 5)
pamk_result

# 군집의 수 확인
pamk_result$nc
table(pamk_result$pamobject$clustering, iris$Species)

#한 윈도우에서 그림을 여러개 그림
layout(matrix(c(1,2),1,2))
plot(pamk_result$pamobject)

==========================
```

4) 밀도 기반 군집법(Density Based Clustering)

특정 기준에 의거하여 많이 모여 있는 것을 군집으로 설정하는 방법

실습.

==========

```
# 밀도기반 군집법

#패키지 로딩
library(fpc)

#species 컬럼 제거
iris2 <- iris[-5]
head(iris2)

#밀도기반 군집
db_result <- dbscan(iris2, eps=0.42, MinPts=5)
db_result

#시각화
plot(db_result, iris2)

#자세히 보기
plot(db_result, iris2[c(1,4)])

#군집 결과 보기
plotcluster(iris2, db_result$cluster)
```

==========

ch16 연습문제 1-2번