

	<b>단위별 학습내용 (Week13)</b>
<b>wk13-1</b>	<b>군집분석과 유사성척도</b>
<b>wk13-2</b>	<b>계층적 군집분석</b>
<b>wk13-3</b>	<b>비계층적 군집분석</b>

## Wk13-1 : 군집분석과 유사성척도

# 1. 군집분석

13-1 군집분석과 유사성척도

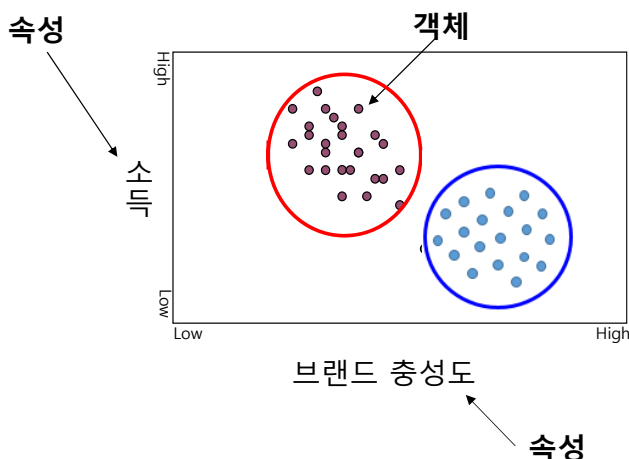
• 군집분석은 비지도학습(Unsupervised Learning) : 속성변수들의 특징으로 그룹화

모형화	특징	내용	적용기법
예측	• 타겟변수 값이 주어지는 경우 (supervised learning)	주어진 데이터를 기반으로 모델을 만든 후, y값을 예측 (y=continuous value)	• 다중회귀분석 • 주성분 회귀분석 • 부분최소자승법 • 신경망
분류	• 변수간의 관계	학습표본을 기반으로 분류 규칙을 생성. 분류규칙의 성능을 검증하기 위해 실제 범주와 추정된 범주를 비교 (y=0/1 혹은 다범주)	• K-인접기법 • 로지스틱 회귀모형 • 의사결정나무 • 선형판별분석 • 서포트벡터머신
군집	• 타겟변수 값이 없는 경우 (unsupervised learning)	주어진 데이터 (X변수들)의 속성으로 군집화	• 계층형 군집 분석 • k-means
연관규칙	• 개체간의 관계	연관성 있는 변수관계 도출 (동시 발생 빈도 분석)	• 연관규칙 분석

# 1. 군집분석

13-1 군집분석과 유사성척도

• 군집분석(cluster analysis)이란, 유사한 속성을 가진 객체들을 군집(cluster)으로 나누는(묶어주는) 데이터마이닝 기법

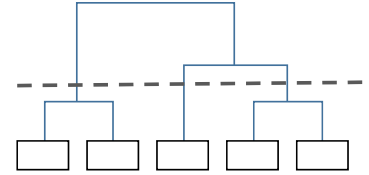


- 예제 : 고객들의 구매패턴을 반영하는 속성들에 관한 데이터가 수집된다고 할 때  
=> 군집분석을 통해 유사한 구매패턴을 보이는 고객들을 군집화하고 판매전략을 도출

- 군집분석의 방법은 (1) 계층적 방법과 (2) 비계층적 방법으로 구분

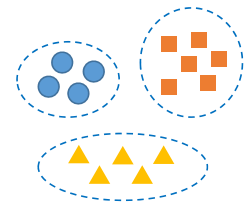
(1) 계층적 군집  
(Hierarchical  
Clustering)

- 사전에 군집 수  $k$ 를 정하지 않고  
단계적으로 군집 트리를 제공



(2) 비계층적 군집  
(Non-hierarchical  
Clustering)

- 사전에 군집 수  $k$ 를 정한 후  
각 객체를  $k$ 개 중 하나의 군집에 배정



## 군집분석 - 유사성척도의 계산 -

- 객체 간의 유사성 정도를 정량적으로 나타내기 위해서 척도가 필요
  - 거리 (distance) 척도
    - 거리가 가까울수록 유사성이 크다. 거리가 멀수록 유사성이 적어짐
  - 상관계수척도
    - 객체간 상관계수가 클수록 두 객체의 유사성이 커짐

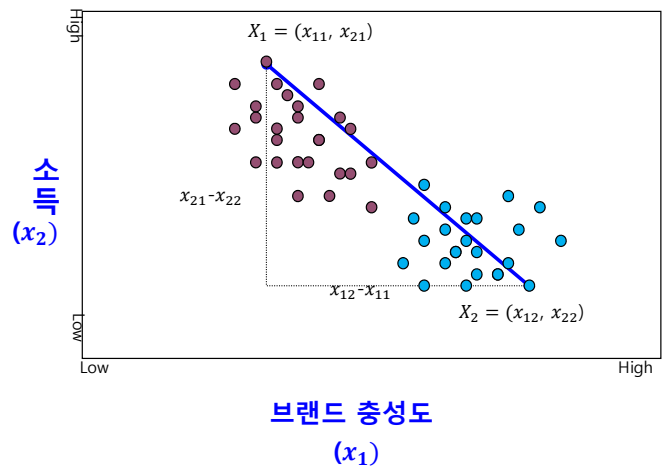
- 객체  $i$ 의  $p$ 차원 공간에서의 좌표는 다음과 같은 열벡터로 표현
  - $p$ 개의 속성을 가진 객체  $i$ 에 대하여,  $j$ 번째 속성은  $X_{ji}$  으로 표현

$$x_i = (X_{1i}, X_{2i}, \dots, X_{pi})^T \quad i = 1, \dots, n$$

- 유클리디안 거리

$$d_{ij} = d(x_i, x_j) = \sqrt{\sum_{k=1}^p |X_{ki} - X_{kj}|^2}$$

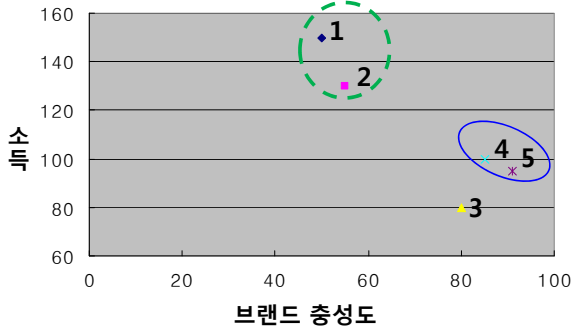
$$Distance = \sqrt{(x_{12} - x_{11})^2 + (x_{21} - x_{22})^2}$$



### 3. 유클리디안 거리

13-1 군집분석과 유사성척도

#### • 유클리디안 거리(Euclidean distance)



$$d_{12} = d_{21} = \sqrt{(150 - 130)^2 + (50 - 55)^2} = 20.6$$

데이터

ID	소득	브랜드 충성도
1	150	50
2	130	55
3	80	80
4	100	85
5	95	91

ID	1	2	3	4	5
1	0.0				
2	20.6	0.0			
3	76.2	55.9	0.0		
4	61.0	42.4	20.6	0.0	
5	68.6	50.2	18.6	7.8	0.0

- (4와 5번)이 가장 가까움
- (1과 2번)이 두번째로 가까움

### 3. 유클리디안 거리

13-1 군집분석과 유사성척도

#### • 거리(distance)계산 함수 : dist(데이터, method= , )

```
# lec13_1_clus.R
# Clustering
# Distance measure

# similarity measures - distance
m1 <- matrix(
  c(150, 50, 130, 55, 80, 80, 100, 85, 95, 91),
  nrow = 5,
  ncol = 2,
  byrow = TRUE)
# m1 is a matrix
m1
is.data.frame(m1)
# m1 is defined as dataframe
m1<-as.data.frame(m1)
```

데이터 생성 ( m1, 5x2 행렬)

```
> m1
      [,1] [,2]
[1,]  150   50
[2,]  130   55
[3,]   80   80
[4,]  100   85
[5,]   95   91
```

m1을 data frame으로 저장

```
# 1. Euclidean distance
p1 <- dist(m1)
p1
```

```
> D1 <- dist(m1)
> D1
      1      2      3      4
2 20.61553
3 76.15773 55.90170
4 61.03278 42.42641 20.61553
5 68.60029 50.20956 18.60108 7.81025
```

## • 거리계산 옵션

help("dist")

### Distance Matrix Computation

#### Description

This function computes and returns the distance matrix computed by using the specified distance measure to compute the distances between the rows of a data matrix.

#### Usage

```
dist(x, method = "euclidean", diag = FALSE, upper = FALSE, p =
as.dist(m, diag = FALSE, upper = FALSE)
## Default S3 method:
as.dist(m, diag = FALSE, upper = FALSE)

## S3 method for class 'dist'
print(x, diag = NULL, upper = NULL,
      digits = getOption("digits"), justify = "none",
      right = TRUE, ...)

## S3 method for class 'dist'
as.matrix(x, ...)
```

#### Arguments

**x** a numeric matrix, data frame or "dist" object.  
**method** the distance measure to be used. This must be one of "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". Any unambiguous substring can be given.

# 4. 그 외 거리 척도

## • 민코프스키 거리(Minkowski distance)

- 유클리디안 거리의 일반화된 방법 (m=2 일 때는 유클리디안 거리와 동일)

$$d(x_i, x_j) = (\sum_{k=1}^p |X_{ki} - X_{kj}|^m)^{1/m}$$

## • 마할라노비스 거리(Mahalanobis distance)

- 변수 간의 상관 관계가 존재할 때 사용

$$d(x_i, x_j) = \sqrt{(x_i - x_j)^T S^{-1} (x_i - x_j)}$$

$$S = \begin{bmatrix} s_{11}^2 & s_{12} & \cdots & s_{1p} \\ s_{21} & s_{22}^2 & & s_{2p} \\ \vdots & & \ddots & \vdots \\ s_{p1} & s_{p2} & \cdots & s_p^2 \end{bmatrix}$$

$$s_{ab} = \frac{\sum_i (X_{ai} - \bar{X}_a)(X_{bi} - \bar{X}_b)}{n - 1}$$

## 4. 그 외 거리 척도

13-1 군집분석과 유사성척도

### • 민코프스키 거리(Minkowski distance)

: `dist(data(or matrix), method="minkowski", p=3)`

```
# 2. Minkowski distance
D2<- dist(m1, method="minkowski", p=3)
D2
```



민코프스키 거리

```
> D2<- dist(m1, method="minkowski", p=3)
> D2
```

	1	2	3	4
2	20.103629			
3	71.790544	52.002096		
4	55.164795	37.797631	20.103629	
5	61.735957	44.736068	16.757812	6.986368

유클리디안 거리

```
> D1 <- dist(m1)
> D1
```

	1	2	3	4
2	20.61553			
3	76.15773	55.90170		
4	61.03278	42.42641	20.61553	
5	68.60029	50.20956	18.60108	7.81025

민코프스키 계산식에서  $p=2$ 이면 유클리디안거리와 동일

## 5. 상관계수를 척도로 사용

13-1 군집분석과 유사성척도

### • 또 다른 유사성 척도로 객체 간의 상관계수를 사용

- 상관계수가 클수록 두 객체의 유사성이 크다고 추정

- 객체  $i$  와 객체  $j$  간의 표본상관계수는 다음과 같이 정의

$$sim(x_i, x_j) = r_{ij} = \frac{\sum_{k=1}^p (X_{ki} - m_i)(X_{kj} - m_j)}{\sqrt{\sum_{k=1}^p (X_{ki} - m_i)^2} \sqrt{\sum_{k=1}^p (X_{kj} - m_j)^2}}$$

- 이때  $m_i$ 는 객체  $i$ 의 평균값으로 다음과 같음

$$m_i = \frac{1}{p} \sum_{k=1}^p X_{ki}$$

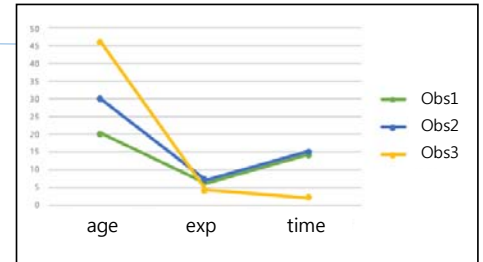
## • 상관계수측정(cor)

# 3. correlation coefficient

```
m2 <- matrix(
  c(20, 6, 14, 30, 7, 15, 46, 4, 2),
  nrow = 3, ncol = 3, byrow = TRUE,
  dimnames = list(
    c("obs1", "obs2", "obs3"),
    c("age", "exp", "time")))
m2
```

데이터 생성 ( 3x3 matrix)

```
> m2
      age exp time
obs1  20  6  14
obs2  30  7  15
obs3  46  4   2
```



# correlation between Obs1~Obs2

```
cor(m2[1,],m2[2,])
```

# correlation between Obs1~Obs3

```
cor(m2[1,],m2[3,])
```

상관계수 측정

```
> # correlation between Obs1~Obs2
```

```
> cor(m2[1,],m2[2,])
```

```
[1] 0.9673518
```

```
> # correlation between Obs1~Obs3
```

```
> cor(m2[1,],m2[3,])
```

```
[1] 0.7984081
```

객체1(obs1)과 객체2의 유사성이

객체1(obs1)과 객체3간의 유사성보다 크다 (0.9674 > 0.7984)

