Ch 9. Clustering

9.1 클러스터링 학습 문제

9.2 성능 척도

9.3 거리 계산법

9.4.1 k 평균 클러스터링 알고리즘

21. 04. 07 김혜수

9.1.0 지도 학습 VS. 비지도 학습

▪ 지도 학습 (Supervised learning)

→ 정답이 있는 문제 !

Y = f (X) 에 대하여 입력 변수 X와 출력 변수 Y의 관계에 대하여 모델링

- 회귀 (regression) : X에 대해 **연속형** 변수 Y 를 예측
- 분류 (classification) : X에 대해 **이산형** 변수 Y (class) 를 예측

$$y = f(X)$$

회귀 모형

$$y = f(X)$$

분류 모형

9.1.0 지도 학습 VS. 비지도 학습

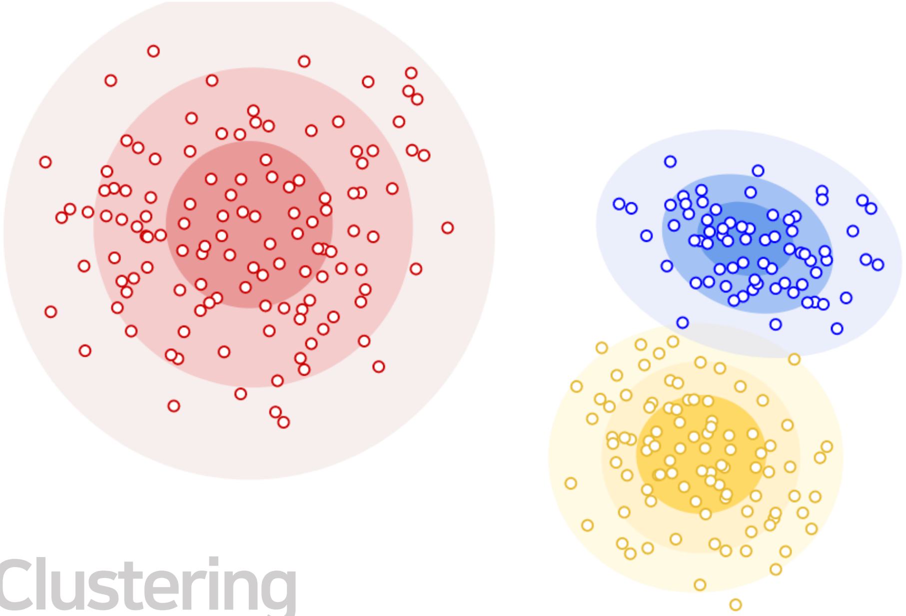
- **비지도 학습 (Unsupervised learning)** → **정답이 없는 문제 !**

  출력 변수 Y가 존재하지 않고, 입력 변수 X 간의 관계에 대해 모델링

  - **클러스터링** : 비슷한 데이터들을 군집화
  - PCA : 독립변수들의 차원을 축소화

※ 주의

분류 (지도 학습) ≠ 클러스터링 (비지도 학습)



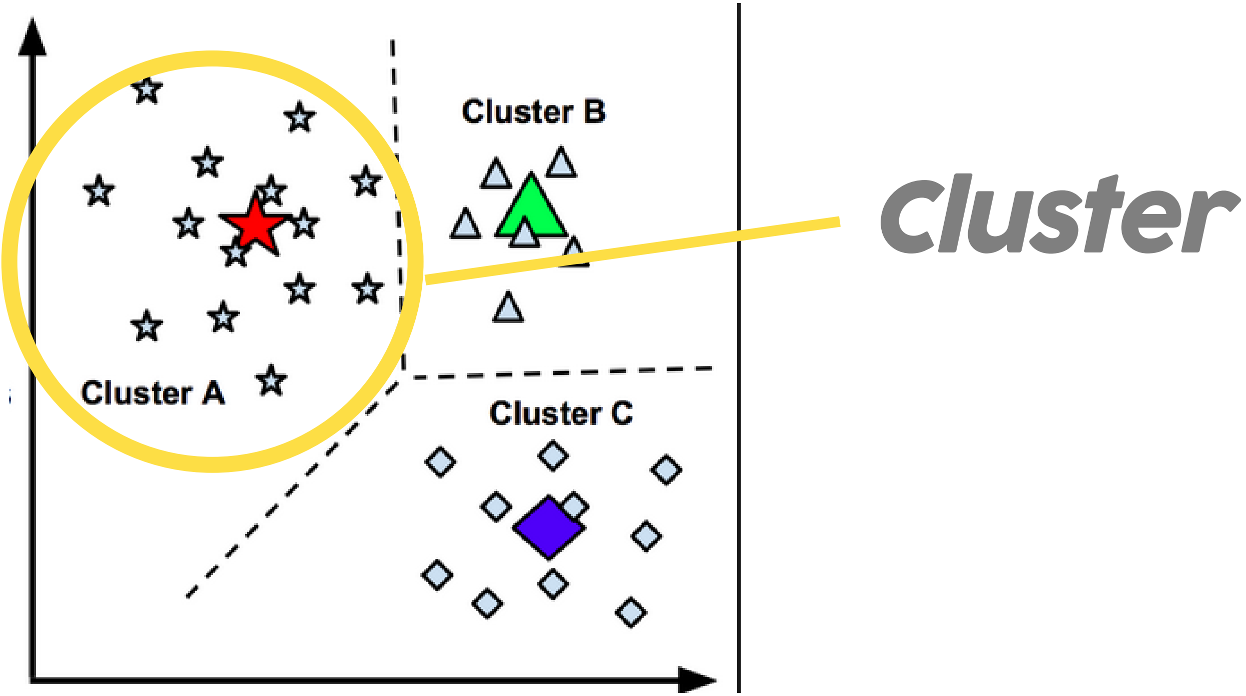
Clustering

9.1.1 클러스터링이란?

- **Clustering (군집화)**

  데이터셋에서, 유사한 성격을 가진 개체를 묶어 그룹으로 구성하는 것

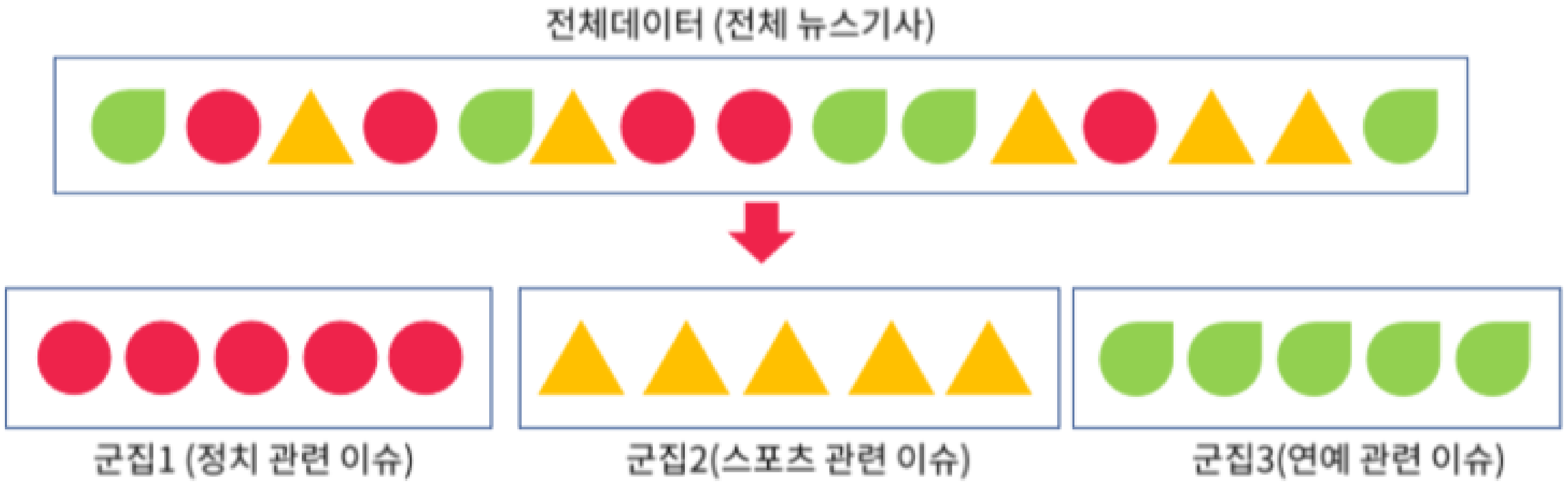
  - 클러스터 : 비슷한 특성을 가진 데이터들의 집합



9.1.2 클러스터링의 활용

▪ 클러스터링의 활용 목적

- 데이터 내의 분포 구조를 이해하기 위해서 활용
- 다른 학습 문제 해결 전 사전 프로세스로서 활용



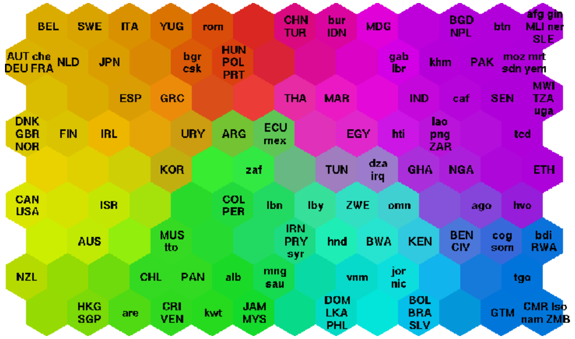
9.1.2 클러스터링의 활용

▪ 예시)

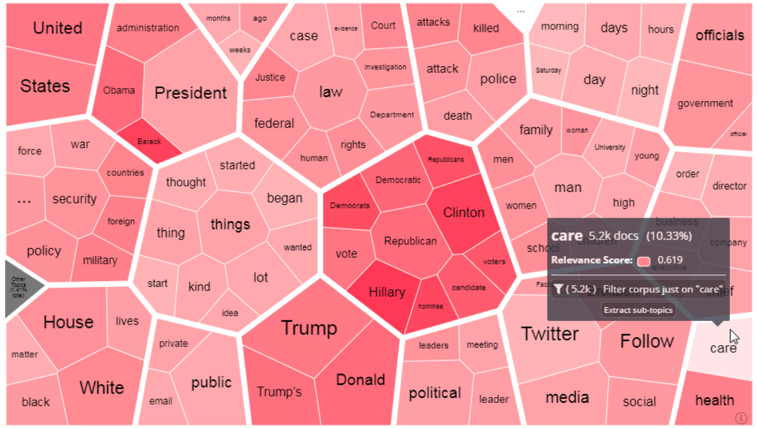
- 고객 세분화

[고객세분화]

- Self-Organizing Map (SOM)

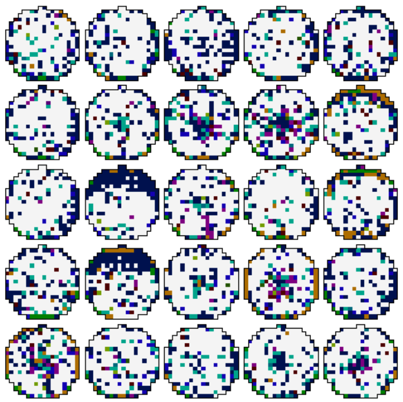


- 문서 데이터 토픽 클러스터링



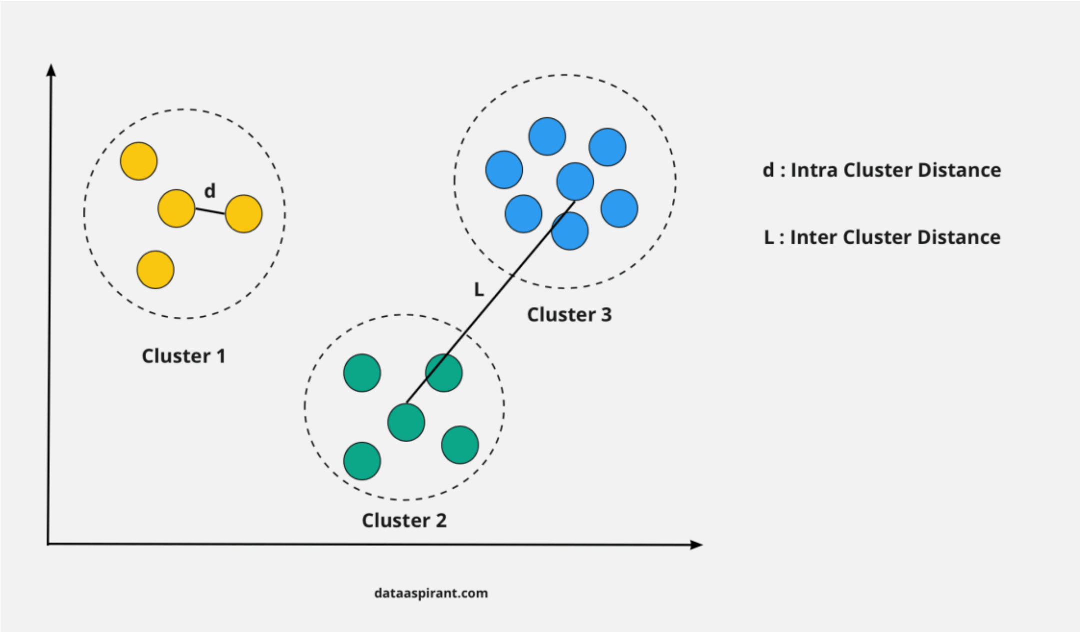
- In-depth Clustering

Sample lot exhibiting spatial patterning



9.2.1 좋은 클러스터링 결과

클러스터링은 목적은, **다른 개체는 다른 그룹**으로, **비슷한 개체는 한 그룹**으로!



클러스터 내 유사도 ↑ (군집 내 분산 **최소화**)

클러스터 간 유사도 ↓ (군집 간 분산 **최대화**)

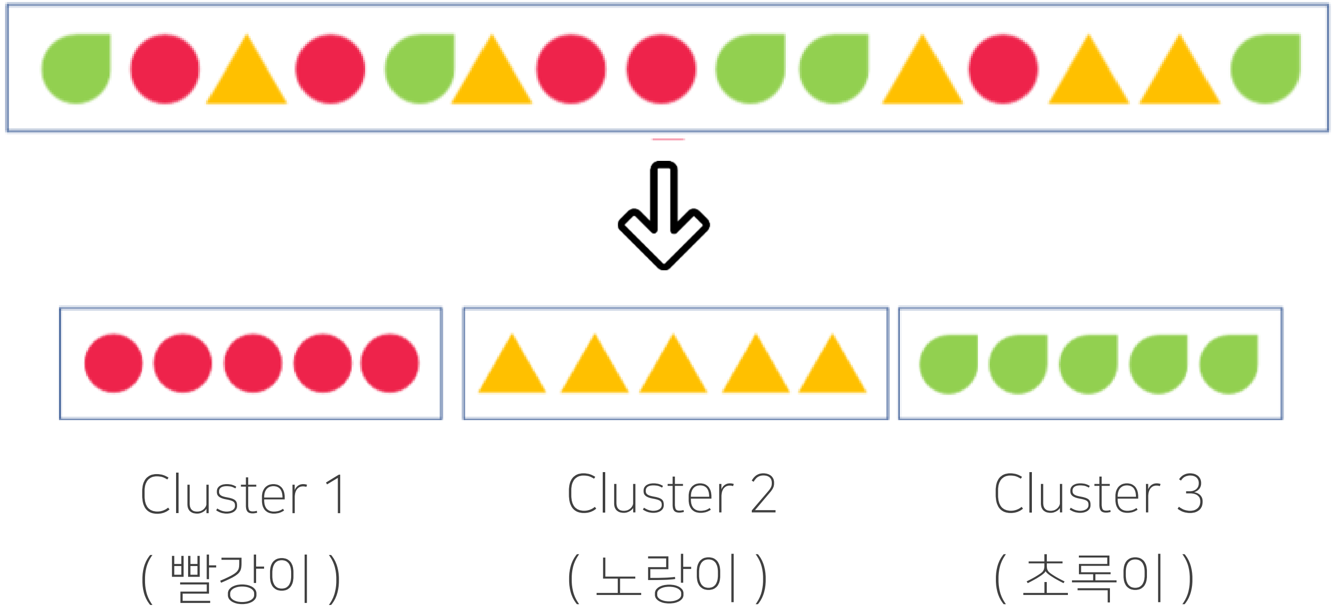
▪ 외부 지표 (External Index)

클러스터링 결과와 어떠한 **참고 모델**을 비교하는 것
known answer

$$D = \{ x_1, x_2, \dots, x_m \}$$

$$C^* = \{ C_1^*, C_2^*, \dots, C_k^* \}$$
$$C = \{ C_1, C_2, \dots, C_k \}$$

$$\lambda = \{ \lambda_1, \lambda_2, \dots, \lambda_k \}$$
$$\lambda^* = \{ \lambda_1^*, \lambda_2^*, \dots, \lambda_k^* \}$$



9.2.2 클러스터링 유효성 지표 Clustering Validity Index

▪ 외부 지표 (External Index)

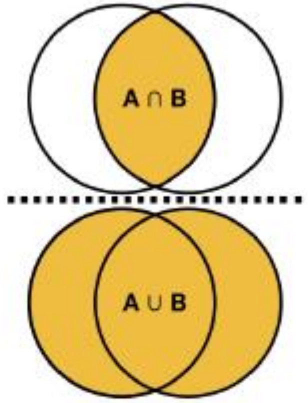
클러스터링 결과와 어떠한 참고 모델을 비교하는 것 → ***unrealistic***

known answer

TP
$$a = |SS|, \quad SS = \{(\boldsymbol{x}_i, \boldsymbol{x}_j) \mid \lambda_i = \lambda_j, \lambda_i^* = \lambda_j^*, i < j)\},$$
FP
$$b = |SD|, \quad SD = \{(\boldsymbol{x}_i, \boldsymbol{x}_j) \mid \lambda_i = \lambda_j, \lambda_i^* \neq \lambda_j^*, i < j)\},$$
FN
$$c = |DS|, \quad DS = \{(\boldsymbol{x}_i, \boldsymbol{x}_j) \mid \lambda_i \neq \lambda_j, \lambda_i^* = \lambda_j^*, i < j)\},$$
TN
$$d = |DD|, \quad DD = \{(\boldsymbol{x}_i, \boldsymbol{x}_j) \mid \lambda_i \neq \lambda_j, \lambda_i^* \neq \lambda_j^*, i < j)\},$$

- Jaccard 계수 Jaccard Coefficient, JC

$$\mathrm{JC} = \frac{a}{a+b+c}$$



위 성능 척도의 결괏값은 [0, 1] 구간에 있고 **클수록 좋음!**

9.2.2 클러스터링 유효성 지표 Clustering Validity Index

▪ 내부 지표 (Internal Index)

참고 모델을 사용하지 않고, 클러스터의 **compactness**(군집 내 분산 최소화)와 **separation**(군집 간 분산 최대화)에 집중

- **클러스터 C 내** 개체 간의 평균 거리

$$\mathrm{avg}(C) = \frac{2}{|C|(|C|-1)} \sum\nolimits_{1 \leqslant i < j \leqslant |C|} \mathrm{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

- **클러스터 C 내** 개체 간의 최대 거리

$$\mathrm{diam}(C) = \max\nolimits_{1 \leqslant i < j \leqslant |C|} \mathrm{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

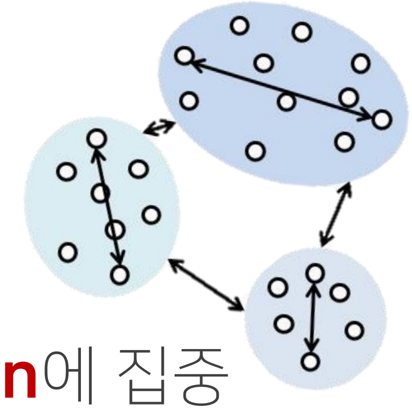
- **클러스터 C_i와 C_j 간** 개체의 최단 거리

$$d_{\min}(C_i, C_j) = \min\nolimits_{\boldsymbol{x}_i \in C_i, \boldsymbol{x}_j \in C_j} \mathrm{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

- **클러스터 C_i와 C_j** 의 중심점 간의 거리

$$d_{\mathrm{cen}}(C_i, C_j) = \mathrm{dist}(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j)$$

9.2.2 클러스터링 유효성 지표 Clustering Validity Index



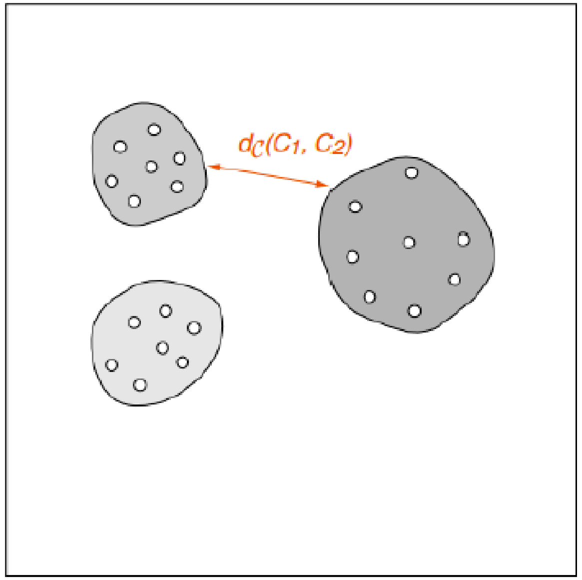
▪ 내부 지표 (Internal Index)

군집 내 분산 최소화

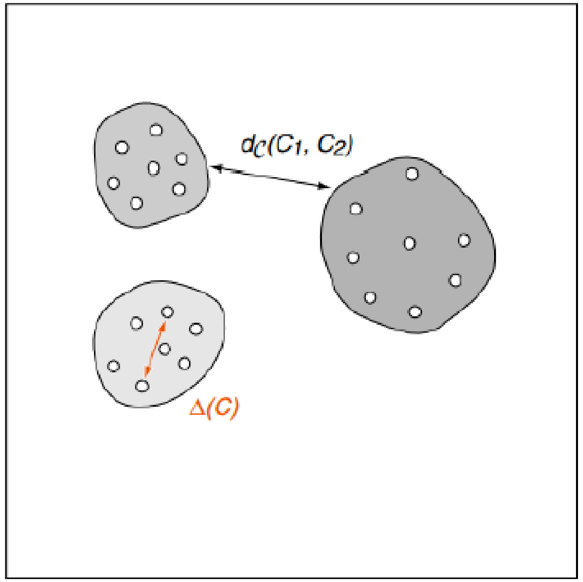
참고 모델을 사용하지 않고, 클러스터의 **compactness**와 **separation**에 집중

군집 간 분산 최대화

Distance between two clusters



Diameter of a cluster



- Dunn 지수 Dunn Index, DI

군집 간 거리의 최솟값 ⇧

$$DI(C) = \frac{min_{i \neq j}\{\, d_c\left(C_i, C_j\right) \}}{max_{1 \leq l \leq k}\{\, diam(\, C_l\,)\, \}}$$

군집 내 거리의 최댓값 ⇩

Dunn 지수는 클수록 좋음!

9.3.1 계산척도

- **계산 척도 (distance measure)**

함수 $dist\ (\cdot,\cdot)$ 가 계산척도라면 아래 기본 성질 만족해야 함

- 비음수성 : $dist\left(x_i, x_j\right) \geq 0$
- 동일성 : $x_i = x_j$ 일 때만, $dist\left(x_i, x_j\right) = 0$
- 대칭성 : $dist\left(x_i, x_j\right) = dist\left(x_j, x_i\right)$
- 삼각부등식 성질 : $dist\left(x_i, x_j\right) \leq dist\left(x_i, x_k\right) + dist\left(x_k\ x_j\right)$

9.3.2 거리

- **민코프스키 거리 (Minkowski distance)** → **순서형 속성에 사용 가능**

맨하탄 거리와 유클리디안 거리를 하나의 공식으로 일반화한 거리 계산 척도

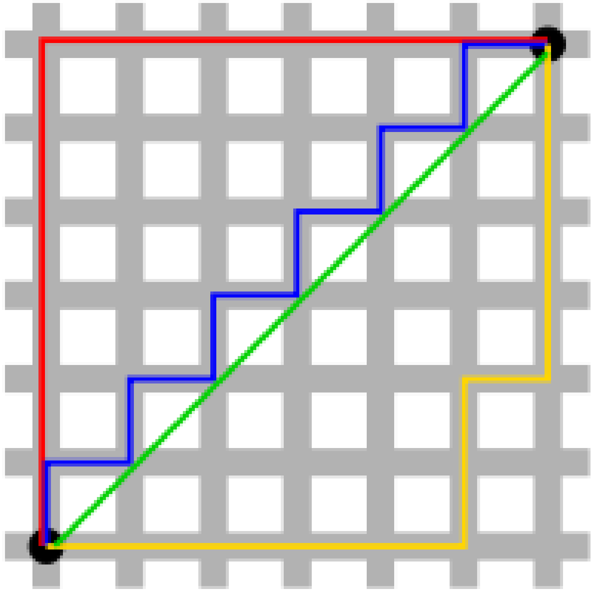
$$(\Sigma_{i=1}^{n} \mid x_i - y_i|^p)^{1/p}$$

- $p = 2$ 일 때, 민코프스키 거리는 유클리디안 거리

$$\mathrm{dist}_{\mathrm{ed}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = ||\boldsymbol{x}_i - \boldsymbol{x}_j||_2 = \sqrt{\sum_{u=1}^{n} |x_{iu} - x_{ju}|^2}$$

- $p = 1$ 일 때, 민코프스키 거리는 맨해튼 거리

$$\mathrm{dist}_{\mathrm{man}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = ||\boldsymbol{x}_i - \boldsymbol{x}_j||_1 = \sum_{u=1}^{n} |x_{iu} - x_{ju}|$$



참고) Distance Metric Types

9.3.2 거리

- **VDM (Value Difference Metric)** → **무순서형 속성에 사용 가능**

$$\mathrm{VDM}_p(a,b) = \sum_{i=1}^{k} \left| \frac{m_{u,a,i}}{m_{u,a}} - \frac{m_{u,b,i}}{m_{u,b}} \right|^p$$

- **혼합 속성 : 민코프스키 거리 + VDM**

순서형은 민코프스키 거리로, 무순서형은 VDM으로 계산

$$\mathrm{MinkovDM}_p(\boldsymbol{x}_i, \boldsymbol{x}_j) = \left(\sum_{u=1}^{n_c} |x_{iu} - x_{ju}|^p + \sum_{u=n_c+1}^{n} \mathrm{VDM}_p(x_{iu}, x_{ju}) \right)^{\frac{1}{p}}$$

거리로 유사도를 측정 : 거리 ⇧ 유사도 ⇩

9.4.0 프로토타입 클러스터링 Prototype-based clustering

- **프로토타입 기반 클러스터링 (Prototype-based clustering)**

  각 클러스터가 하나의 프로토타입으로 표현될 수 있다고 가정

  (즉, 미리 정해 놓은 각 클러스터의 프로토타입에 각 객체가 얼마나 유사한지에 따라서 클러스터링 함)

- **프로토타입 유형**

  - 연속형 데이터 : 평균 Mean, 중앙값 Median
  - 이산형 데이터 : 최빈값 Mode, 중간점 Medoid

9.4.1 K 평균 클러스터링 알고리즘 K-Means Clustering

- **K 평균 클러스터링 알고리즘 (K – Means Clustering)**

  클러스터 내의 각 객체들이 클러스터 중심과의 거리의 분산을 최소화하는 알고리즘

$$E = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|_2^2$$

클러스터 C_i 평균 벡터 (Centroid)

- **그리디 전략 (Greedy Algorithms) 사용**

  Centroid 와 Membership 의 최적해를 모두 찾아야 하므로, 반복적인 최적화를 통해 최적해의 근사치를 찾음

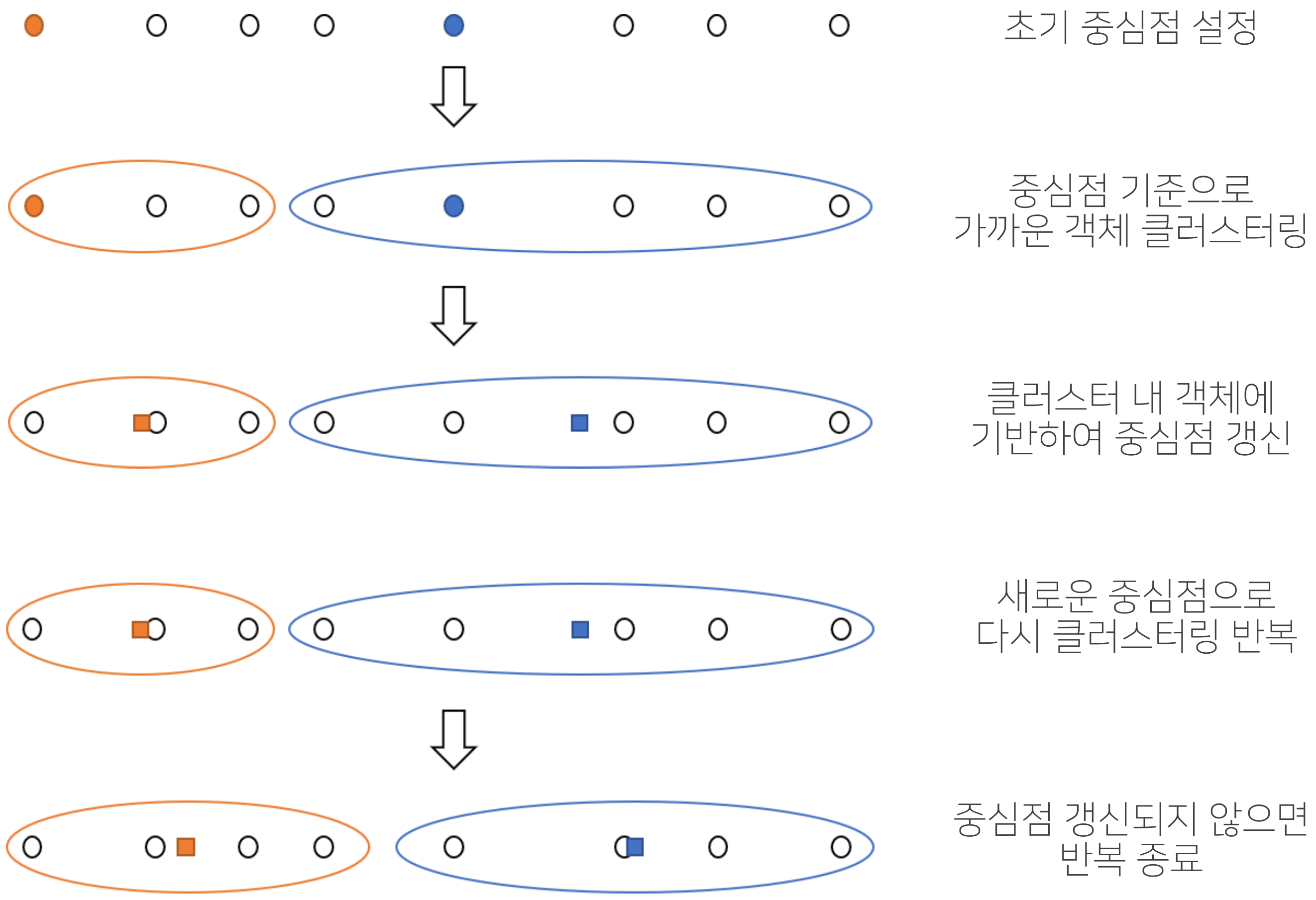
▪ K 평균 클러스터링 알고리즘 (K – Means Clustering)

입력: 샘플 세트 $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$
클러스터 k

과정:

D에서 랜덤으로 k개의 샘플을 선택해 초기 평균 벡터$_{\text{mean vector}}$ $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k\}$로 정한다
repeat
  $C_i = \varnothing \ (1 \leqslant i \leqslant k)$로 설정한다
  for $j = 1, 2, \ldots, m$ **do**
    샘플 $\boldsymbol{x}_j$와 각 평균 벡터 $\boldsymbol{\mu}_i \ (1 \leqslant i \leqslant k)$ 간의 거리를 계산한다: $d_{ji} = \|\boldsymbol{x}_j - \boldsymbol{\mu}_i\|_2$
    거리가 가장 가까운 평균 벡터를 기반으로 $\boldsymbol{x}_j$의 클러스터 레이블을 정한다: $\lambda_j = \operatorname{argmin}_{i \in \{1,2,\ldots,k\}} d_{ji}$
    샘플 $\boldsymbol{x}_j$를 상응하는 클러스터에 포함한다 $C_{\lambda_j} = C_{\lambda_j} \bigcup \{\boldsymbol{x}_j\}$
  end for
  for $i = 1, 2, \ldots, k$ **do**
    새로운 평균 벡터 $\boldsymbol{\mu}'_i = \frac{1}{|C_i|} \sum_{\boldsymbol{x} \in C_i} \boldsymbol{x}$를 계산한다
    if $\boldsymbol{\mu}'_i \neq \boldsymbol{\mu}_i$ **then**
      평균 벡터 $\boldsymbol{\mu}_i$를 $\boldsymbol{\mu}'_i$로 갱신한다
    else
      현재 평균 벡터를 변하지 않도록 보존한다
    end if
  end for
until 평균 벡터가 갱신되지 않을 때까지

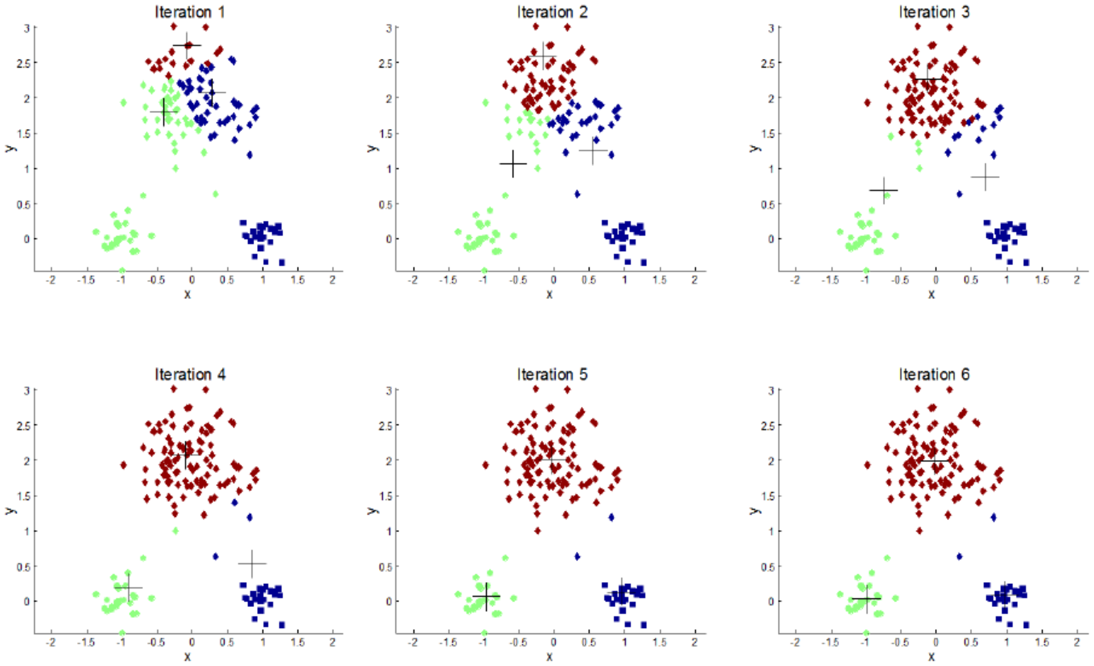
출력: 클러스터 분할 $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$



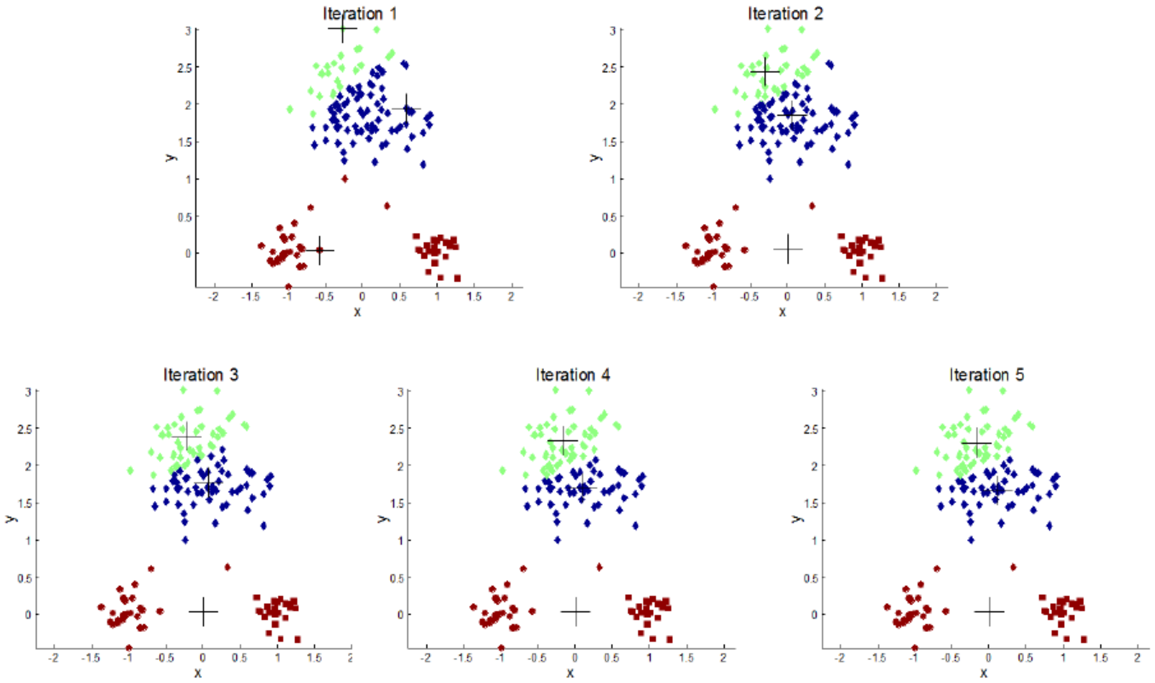
9.4.1 K 평균 클러스터링 알고리즘 K-Means Clustering

▪ K 평균 클러스터링의 한계

- 초기 중심점의 영향이 큼



Desirable centroid selection

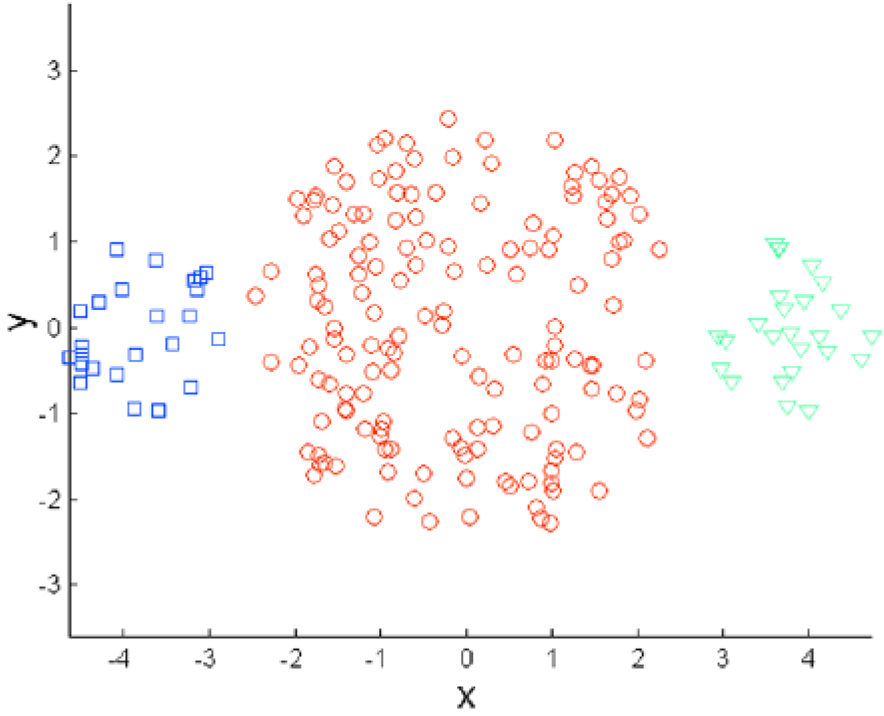


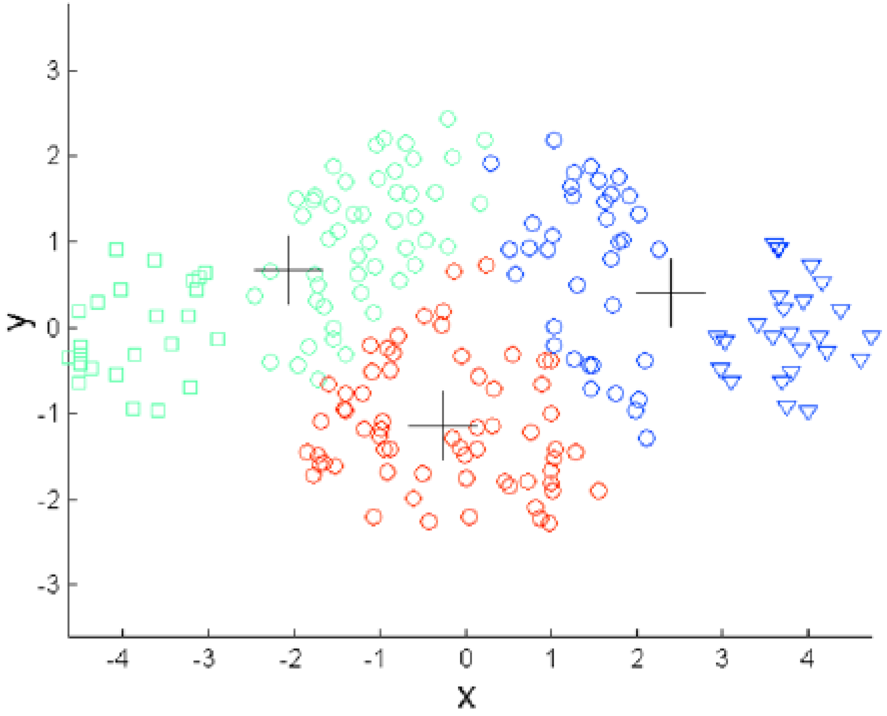
Undesirable centroid selection

9.4.1 K 평균 클러스터링 알고리즘 K-Means Clustering

▪ K 평균 클러스터링의 한계

- 클러스터들의 사이즈가 다를 경우

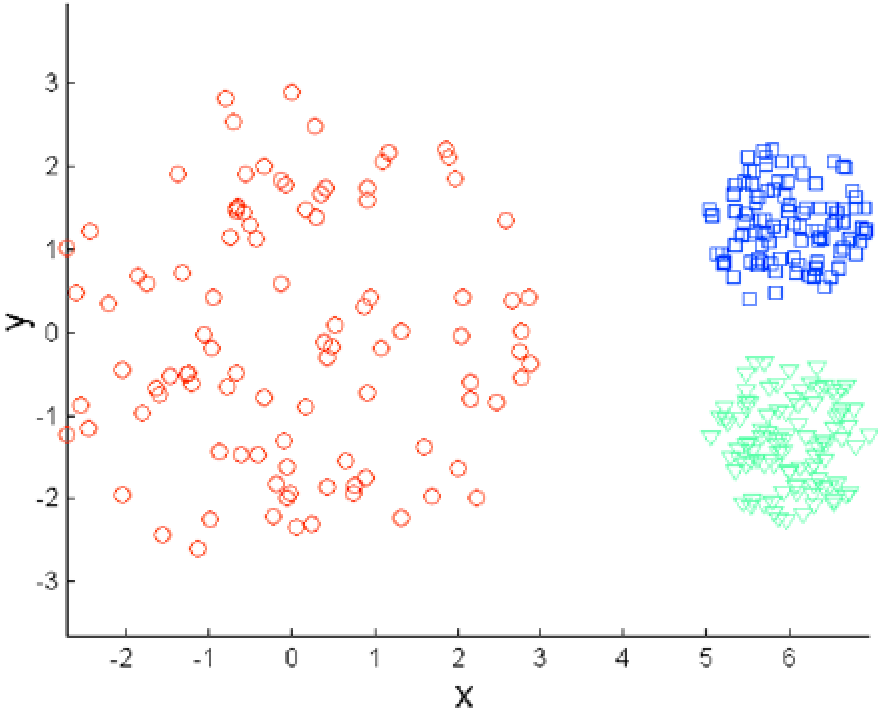


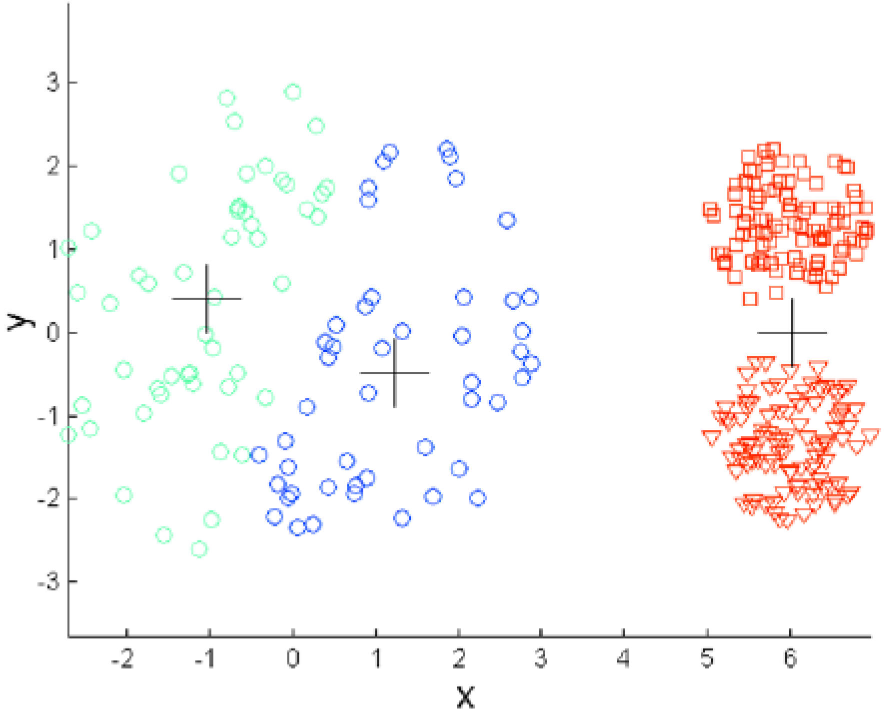


9.4.1 K 평균 클러스터링 알고리즘 K-Means Clustering

▪ K 평균 클러스터링의 한계

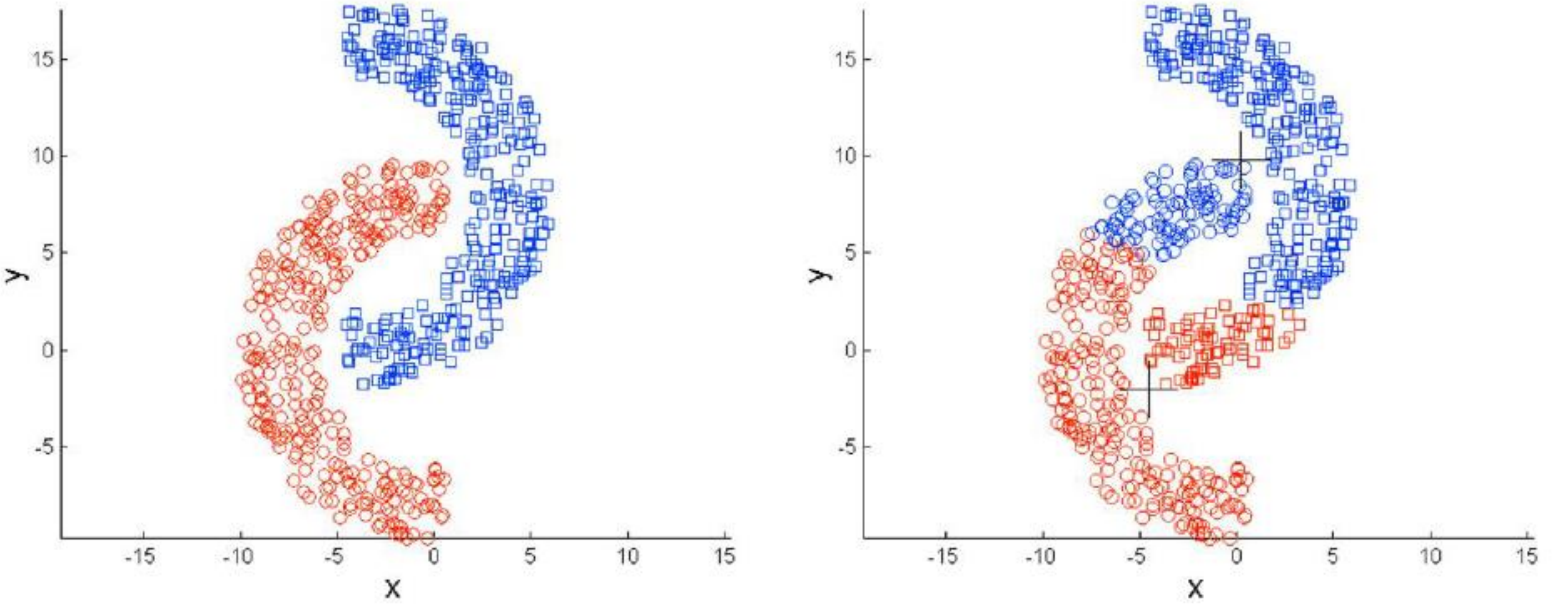
- 클러스터들의 밀도가 다를 경우





▪ K 평균 클러스터링의 한계

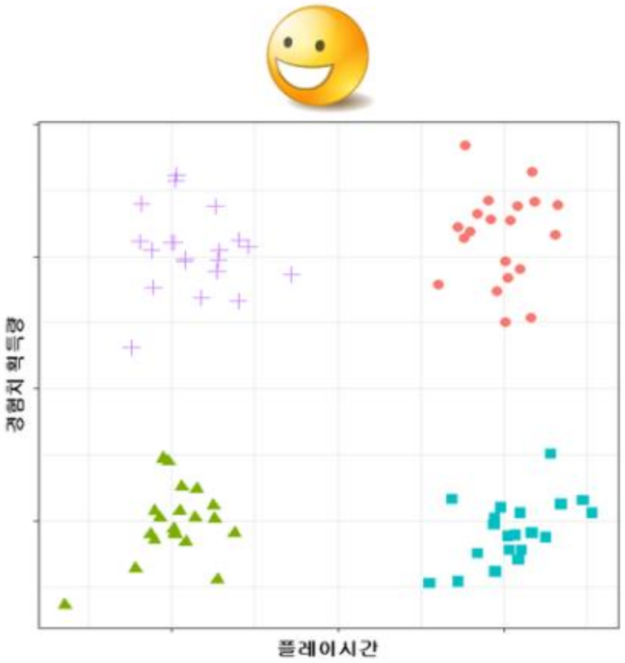
- 데이터의 분포가 특이한 케이스 (non-globular shapes)

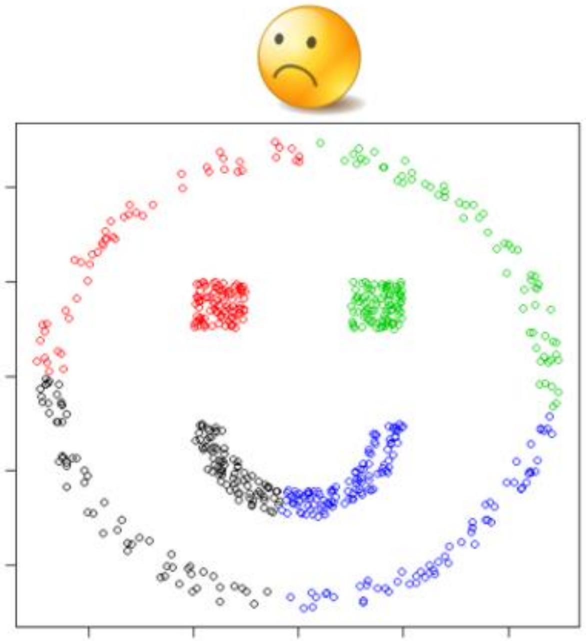


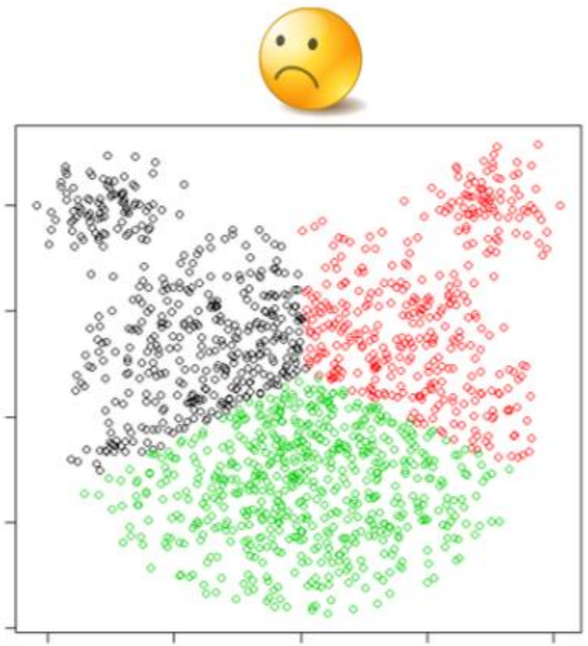
9.4.1 K 평균 클러스터링 알고리즘 K-Means Clustering

▪ K 평균 클러스터링의 한계

- 데이터의 분포가 특이한 케이스 (non-globular shapes)







Etc. 참고자료

- **고려대학교 DSBA - Multivariate Data Analysis 강의**
  - Ch 9. Clustering

- **참고 블로그**
  - ratsgo's blog