기계학습 개론

- Unsupervised Learning (비지도 학습)

정보통신공학과

Prof. Jinkyu Kang

MYONGJI UNIVERSITY

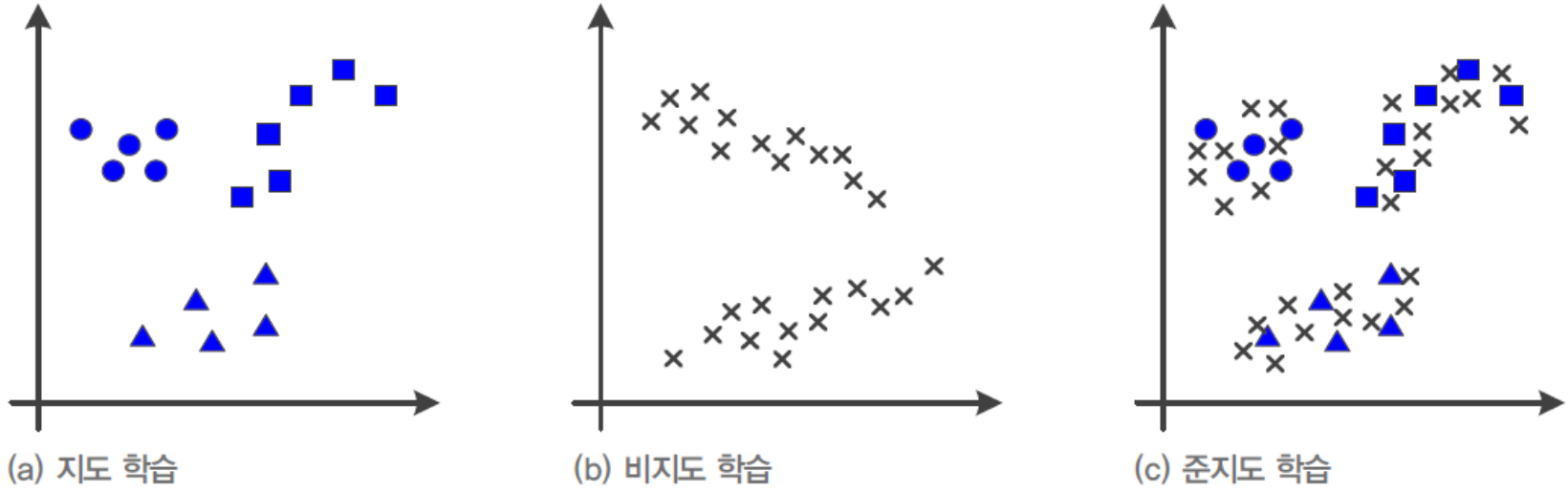
이번주 수업의 목차

- 비지도 학습 소개
- 군집화
  - 거리와 유사도
  - 군집화 알고리즘의 분류
  - 분할 군집화
    - 순차 알고리즘
    - k-means 알고리즘
    - 모델 기반 알고리즘
  - 계층 군집화
    - 응집 계층 알고리즘
    - 분열 계층 알고리즘
  - 신경망

MYONGJI UNIVERSITY

비지도 학습 소개
| 지도 학습과 비지도 학습, 준지도 학습

- 세 가지 유형의 학습
  - 지도 학습: 모든 훈련 샘플이 레이블 정보를 가짐
  - 비지도 학습: 모든 훈련 샘플이 레이블 정보를 가지지 않음
  - 준지도 학습: 레이블을 가진 샘플과 가지지 않은 샘플이 섞여 있음



기계 학습의 유형(속이 찬 샘플은 레이블이 있고, x 표시된 샘플은 레이블이 없음)

비지도 학습 소개
| 지도 학습과 비지도 학습, 준지도 학습

- 지도 학습 과 비지도 학습
  - 지도 학습supervised learning
    - 이전에 공부한 분류기 학습 (베이시언 분류기, MLP, SVM 등)
    - 각 샘플이 그가 속한 부류를 알고 있다.
      (훈련 집합 $X=\{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \cdots, (\mathbf{x}_N, t_N)\}$으로 표기)
  - 비지도 학습 unsupervised learning
    - 샘플은 부류 정보가 없다. ($X=\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$으로 표기)
    - 군집화는 비지도 학습에 해당
    - 군집이 몇 개인지 모르는 경우도 많다.
    - 군집화를 부류 발견 작업이라고도 부른다.

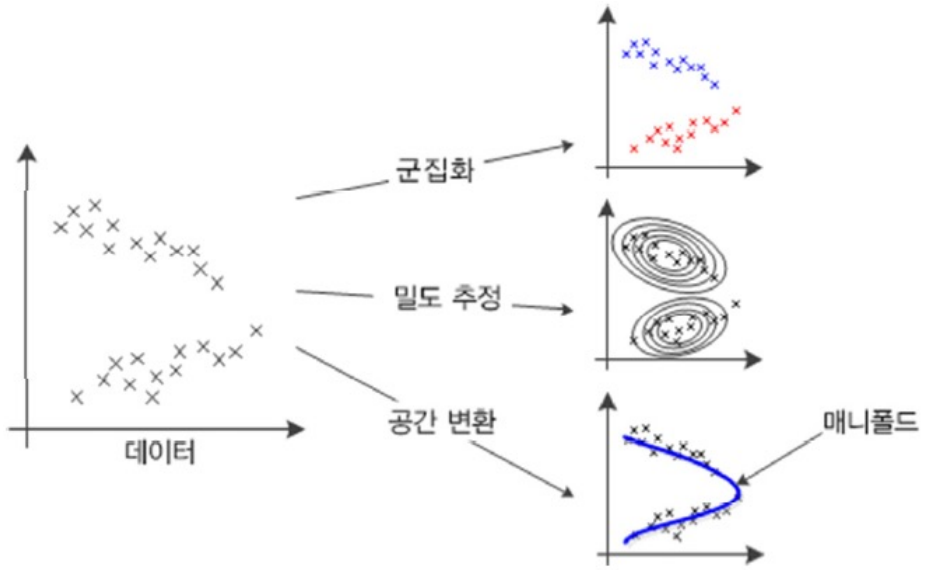
비지도 학습 소개
| 지도 학습과 비지도 학습, 준지도 학습

- 기계 학습이 사용하는 두 종류의 지식
  - 훈련집합
  - 사전 지식prior knowledge(세상의 일반적인 규칙)

- 비지도 학습과 준지도 학습은 사전 지식을 더 명시적으로 사용

비지도 학습 소개 | 비지도 학습의 일반 과업

- 세 가지 일반 과업
  - 군집화: 유사한 샘플을 모아 같은 그룹으로 묶는 일
  - 밀도 추정: 데이터로부터 확률분포를 추정하는 일
  - 공간 변환: 원래 특징 공간을 저차원 또는 고차원 공간으로 변환하는 일
- 데이터에 내재한 구조를 잘 파악하여 새로운 정보를 발견해야 함



비지도 학습의 군집화, 밀도 추정, 공간 변환 과업이 발견하는 정보

비지도 학습 소개 | 비지도 학습의 응용 과업

- 아주 많은 응용(서로 밀접하게 연관)
  - 군집화의 응용
    - 맞춤 광고, 영상 분할, 유전자 데이터 분석, SNS 실시간 검색어 분석하여 사람들의 관심 파악 등
  - 밀도 추정의 응용
    - 분류, 생성 모델 구축 등
  - 공간 변환의 응용
    - 데이터 가시화, 데이터 압축, 특징 추출(표현 학습) 등

군집화

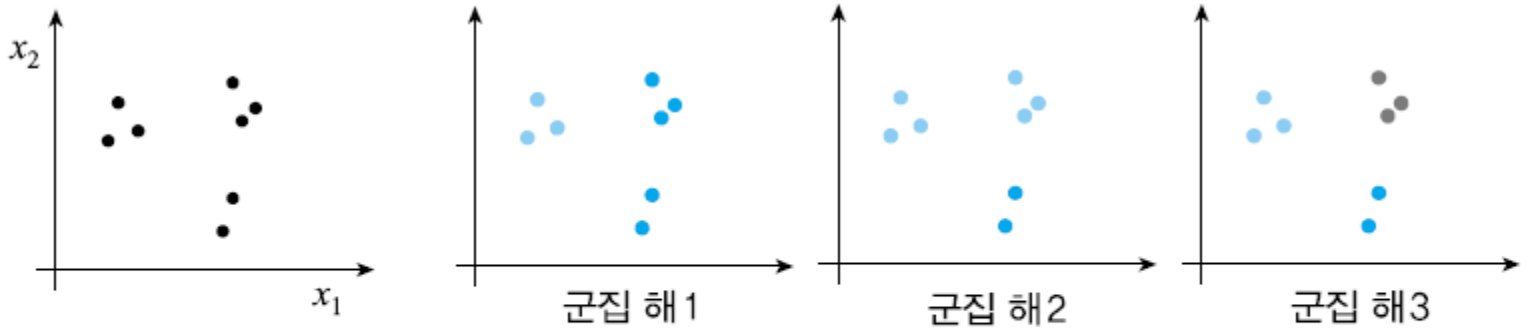
재미있는 응용 시나리오를 생각해 보자. 지난 몇 년간 온라인 쇼핑몰을 성공적으로 운영했는데 이제 제 2의 도약을 꿈꾸고 있다. 그 동안 한 종류의 홍보 팜플렛을 만들어 발송했는데 이제부터는 고객의 취향을 분석하여 4~6종의 팜플렛을 만들어 맞춤 홍보를 하려 한다.[1] 일종의 개인화 personalization 홍보 전략이다. 고객에 대한 각종 정보는 데이터베이스에 저장되어 있어 이것을 기초 자료로 활용하면 된다. 고객 정보는 월평균 구매액, 선호하는 물품의 종류와 수준, 결재 방법, 반품 성향, 직업, 성별, 나이, 거주 지역 등 아주 다양하다. 하지만 수백 만 명이나 되는 고객을 어떻게 4~6개의 그룹으로 분류할 수 있을까?

- 기계학습 문제로 공식화 가능
  - 고객이 샘플, 샘플은 특징 벡터 $\mathbf{x}=(x_1,x_2,\cdots,x_d)^T$로 표현
  - 직업, 월평균 구매액 등이 특징이 됨
  - 유사한 (거리가 가까운) 샘플 집합을 군집이라 함
- 군집화clustering 구현에는 두 가지 필요
  - 1) 거리 척도, 2)유사한 샘플을 군집으로 만드는 알고리즘

MYONGJI UNIVERSITY

군집화

- 군집화의 특성
  - 주관성: 군집화 결과의 품질은 응용이 처한 상황과 요구 사항에 따라 다름



(a) 샘플 집합 (b) 세 가지 군집화 결과를 어떻게 평가할까?

그림 10.1 군집화의 주관성

군집화 | 정의

- **군집화란?** 샘플 집합 $X = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$이 주어졌다고 하자. $\mathbf{x}_i$는 i 번째 샘플로 d 차원 특징 벡터 $\mathbf{x}_i = (x_{i1}, x_{i2}, \cdots, x_{id})^T$로 표기한다. 이 입력에 대해 아래 조건을 만족하는 k 개의 군집으로 구성되는 군집 해 clustering solution $C = \{c_1, c_2, \cdots, c_k\}$를 찾아라. 보통 군집의 개수 k는 N에 비해 매우 작다. 상황에 따라 k가 주어지는 경우도 있고 그렇지 않고 이 값을 찾아야 하는 경우도 있다.

$$\left.\begin{array}{l} c_i \neq \varnothing, i = 1, \cdots, k \\ \cup_{i=1,k}\, c_i = X \\ c_i \cap c_j = \varnothing, i \neq j \end{array}\right\} \quad (10.1)$$

MYONGJI UNIVERSITY

거리와 유사도

군집화가 추구하는 본질적인 목표는 같은 군집 내의 샘플은 서로 가깝고 다른 군집에 속한 샘플 사이의 거리는 멀게 하는 것이다. 따라서 군집화에서 거리 개념은 매우 중요하고 여러 가지 계산 방법이 개발되어 있다. 어떤 계산 방법을 사용하느냐에 따라 군집화 결과는 달라지고 상황에 적합할 수도 있고 그렇지 않을 수도 있다. 따라서 주어진 문제에 적합한 거리 측정 방법을 선택하는 것이 매우 중요하다. 거리와 distance 유사도는 similarity 반대 개념이고 하나를 알면 다른 것은 공식을 이용하여 쉽게 계산할 수 있다.

MYONGJI
UNIVERSITY

거리와 유사도 | 특징 값의 종류

- 특징 값의 종류는 다양하다. (군집화를 공부하는데 이에 대한 이해가 필요하다.)

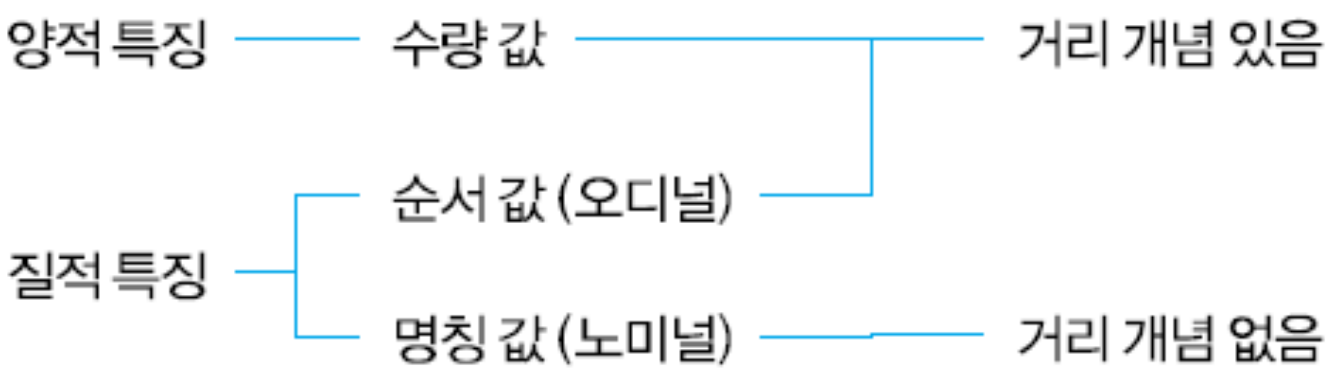


그림 10.3 특징의 유형

MYONGJI UNIVERSITY

거리와 유사도 | 특징 값의 종류

예제 10.1 온라인 쇼핑 몰의 개인화 홍보

고객 레코드 (샘플)을 12개의 특징으로 표현한다고 하자.

$\mathbf{x}$ = (나이, 직업, 연봉, 성별, 월평균 구매액, 반품 성향, 선호하는 물품 수준, 의류 선호도, 전자제품 선호도, 식품 선호도, 팬시 선호도, DVD 선호도)^T

나이: [1,100]의 정수

직업: [1,10]의 정수 (1 = 회사원, 2 = CEO, 3 = 교사, …)

연봉: [1,5]의 정수 (1 = 2천만원 미만, 2 = 2천~3천만원, …, 5 = 1억 이상)

성별: [0,1]의 정수 (0 = 여자, 1 = 남자)

월평균 구매액: 평균을 계산하여 실수로 표현

반품 성향: [1,4]의 정수 (1 = 년간 반품 횟수 0, 2 = 2회 미만, …)

선호하는 물품 수준: [1,4]의 정수 (1 = 저가, 2 = 보통, 3 = 고가, 4 = 명품)

제품 선호도: [1,5]의 정수 (1 = 구매한 적 없음, …, 5 = 아주 선호)

수량 값을 갖는 특징: 나이, 연봉, 월평균 구매액

순서 값을 갖는 특징: 반품 성향, 선호하는 물품 수준, 제품 선호도

명칭 값을 갖는 특징: 직업, 성별

거리와 유사도 | 거리와 유사도 측정

- Minkowski 거리 (10.4)
  - 두 점 $\mathbf{x}_i=(x_{i1},\cdots,x_{id})^T$와 $\mathbf{x}_j=(x_{j1},\cdots,x_{jd})^T$간의 거리 척도
  - p=2면 유클리디언 거리 (10.5), p=1이면 도시블록 거리 (10.6)

$$d_{ij}=\left(\sum_{k=1}^{d}\left|x_{ik}-x_{jk}\right|^{p}\right)^{1/p} \quad (10.4)$$

$$\text{유클리디언 거리 } (p=2)\quad d_{ij}=\sqrt{\sum_{k=1}^{d}(x_{ik}-x_{jk})^2} \quad (10.5)$$

$$\text{맨하탄 거리 } (p=1)\quad d_{ij}=\sum_{k=1}^{d}\left|x_{ik}-x_{jk}\right| \quad (10.6)$$

- Hamming 거리
  - 이진 특징 벡터에 적용 가능 (서로 다른 비트의 개수)
  - 예) $(1,0,1,0,0,0,1,1)^T$과 $(1,0,0,1,0,0,1,0)^T$의 해밍 거리는 3

MYONGJI UNIVERSITY

거리와 유사도 | 거리와 유사도 측정

- 코사인 유사도
  - 문서 검색 응용에서 주로 사용 (단어가 특징, 출현 빈도가 특징 값)

$$s_{ij} = \cos\theta = \frac{\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j}{\|\mathbf{x}_i\|\|\mathbf{x}_j\|} \quad (10.8)$$

- 이진 특징 벡터의 유사도

$$s_{ij} = \frac{n_{00} + n_{11}}{n_{00} + n_{11} + n_{01} + n_{10}} \quad (10.9)$$

$$s_{ij} = \frac{n_{11}}{n_{11} + n_{01} + n_{10}} \quad (10.10)$$

- 유사도와 거리는 쉽게 상호 변환할 수 있다.

$$s_{ij} = d_{\max} - d_{ij} \quad (10.7)$$

$$d_{ij} = s_{\max} - s_{ij} \quad (10.11)$$

MYONGJI UNIVERSITY

거리와 유사도 | 점 집합을 위한 거리

- 점과 군집 (점 집합) 또는 두 군집 간의 거리
  - 점 $\mathbf{x}_i$와 군집 (점 집합) c_j 간의 거리를 $D(\mathbf{x}_i, c_j)$로 표기
  - 두 군집 c_i와 c_j 간의 거리는 $D(c_i, c_j)$로 표기

- 점과 군집 사이의 거리 (모든 샘플이 참여)

$$D_{\max}(\mathbf{x}_i, c_j) = \max_{\mathbf{y}_k \in c_j} d_{ik} \quad (10.12)$$

$$D_{\min}(\mathbf{x}_i, c_j) = \min_{\mathbf{y}_k \in c_j} d_{ik} \quad (10.13)$$

$$D_{\text{ave}}(\mathbf{x}_i, c_j) = \frac{1}{|c_j|} \sum_{\mathbf{y}_k \in c_j} d_{ik} \quad (10.14)$$

  - d_{ik}는 $\mathbf{x}_i$와 $\mathbf{y}_k$는 간의 거리 ($\mathbf{y}_k$는 c_j에 속한 샘플)
  - $D_{\max}$와 $D_{\min}$은 외톨이에outlier 민감

MYONGJI UNIVERSITY

거리와 유사도 | 점 집합을 위한 거리

- 점과 군집 사이의 거리 (대표 샘플만 참여)
  - 평균을 대표로 삼음

$$\left.\begin{aligned} & D_{\text{mean}}(\mathbf{x}_i, c_j) = d_{i,mean} \\ & \text{여기서 } \mathbf{y}_{mean} = \frac{1}{|c_j|}\sum_{\mathbf{y}_k \in c_j} \mathbf{y}_k \end{aligned}\right\} \quad (10.15)$$

  - 다른 것들과 가장 가까운 샘플을 대표로 삼음

$$\left.\begin{aligned} & D_{\text{rep}}(\mathbf{x}_i, c_j) = d_{i,rep} \\ & \text{여기서 } \sum_{\mathbf{y}_k \in c_j} d_{rep,k} \le \sum_{\mathbf{y}_k \in c_j} d_{lk}, \forall \mathbf{y}_l \in c_j \end{aligned}\right\} \quad (10.16)$$

MYONGJI UNIVERSITY

거리와 유사도 | 점 집합을 위한 거리

예제 10.2 점과 군집 사이의 거리

$$c_j = \{\mathbf{y}_1 = (1,1)^T,\ \mathbf{y}_2 = (1,2)^T,\ \mathbf{y}_3 = (2,1)^T,\ \mathbf{y}_4 = (3,1)^T\},\ \mathbf{x}_i = (4,2)^T$$

$$D_{\max} = \max(3.162,\ 3.0,\ 2.236,\ 1.414) = 3.162$$

$$D_{\min} = \min(3.162,\ 3.0,\ 2.236,\ 1.414) = 1.414$$

$$D_{\text{ave}} = (3.162+3.0+2.236+1.414)/4 = 2.453$$

$$\mathbf{y}_{\text{mean}} = ((1,1)^T + (1,2)^T + (2,1)^T + (3,1)^T)/4 = (1.75,1.25)^T$$

$$D_{\text{mean}} = d_{i,mean} = 2.372$$

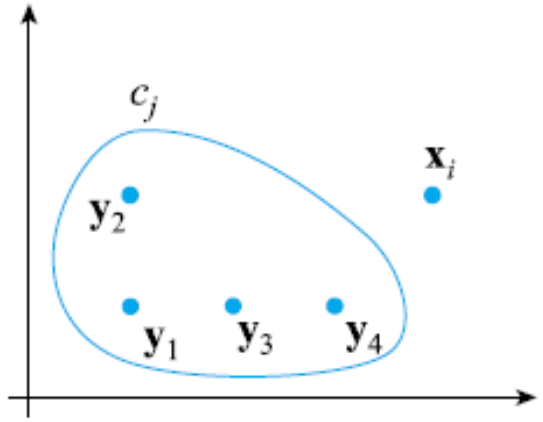


그림 10.4 군집과 점 간의 거리

$$D_{\text{rep}}(\mathbf{x}_i, c_j) = d_{i,rep} = 2.236$$

$$\sum_{\mathbf{y}_k \in c_j} d_{1k} = 1.0+1.0+2.0 = 4.0$$

$$\sum_{\mathbf{y}_k \in c_j} d_{2k} = 1.0+1.414+2.236 = 4.65$$

$$\sum_{\mathbf{y}_k \in c_j} d_{3k} = 1.0+1.414+1.0 = 3.414$$

$$\sum_{\mathbf{y}_k \in c_j} d_{4k} = 2.0+2.236+2.0 = 6.236$$

(*rep*=3이 됨)

MYONGJI UNIVERSITY

거리와 유사도 | 점 집합을 위한 거리

- 두 군집 사이의 거리
  - d_{kl}는 $\mathbf{x}_k$와 $\mathbf{y}_l$ 간의 거리 ($\mathbf{x}_k$는 c_i, $\mathbf{y}_l$은 c_j에 속한 샘플)

$$D_{\max}(c_i, c_j) = \max_{\mathbf{x}_k \in c_i, \mathbf{y}_l \in c_j} d_{kl} \tag{10.17}$$

$$D_{\min}(c_i, c_j) = \min_{\mathbf{x}_k \in c_i, \mathbf{y}_l \in c_j} d_{kl} \tag{10.18}$$

$$D_{\text{ave}}(c_i, c_j) = \frac{1}{|c_i||c_j|} \sum_{\mathbf{x}_k \in c_i} \sum_{\mathbf{y}_l \in c_j} d_{kl} \tag{10.19}$$

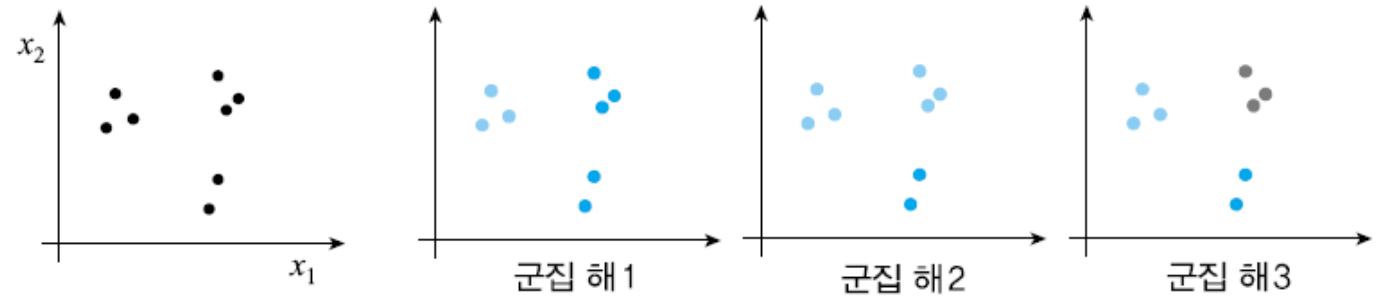
$$\left.\begin{aligned} &D_{\text{mean}}(c_i, c_j) = d_{mean1,mean2} \\ &\text{여기서 } \mathbf{x}_{mean1} = \frac{1}{|c_i|} \sum_{\mathbf{x}_k \in c_i} \mathbf{x}_k, \mathbf{y}_{mean2} = \frac{1}{|c_j|} \sum_{\mathbf{y}_l \in c_j} \mathbf{y}_l \end{aligned}\right\} \tag{10.20}$$

$$\left.\begin{aligned} &D_{\text{rep}}(c_i, c_j) = d_{rep1,rep2} \\ &\text{여기서 } \sum_{\mathbf{x}_k \in c_i} d_{rep1,k} \le \sum_{\mathbf{x}_k \in c_i} d_{pk}, \forall \mathbf{x}_p \in c_i, \sum_{\mathbf{y}_l \in c_j} d_{rep2,l} \le \sum_{\mathbf{y}_l \in c_j} d_{pl}, \forall \mathbf{y}_p \in c_j \end{aligned}\right\} \tag{10.21}$$

MYONGJI UNIVERSITY

군집화 알고리즘의 분류

- 매우 다양한 알고리즘
  - 군집화 문제의 본질적인 성질에 기인 (주관이 많이 개입되는 성질)



(a) 샘플 집합 (b) 세 가지 군집화 결과를 어떻게 평가할까?

그림 10.1 군집화의 주관성

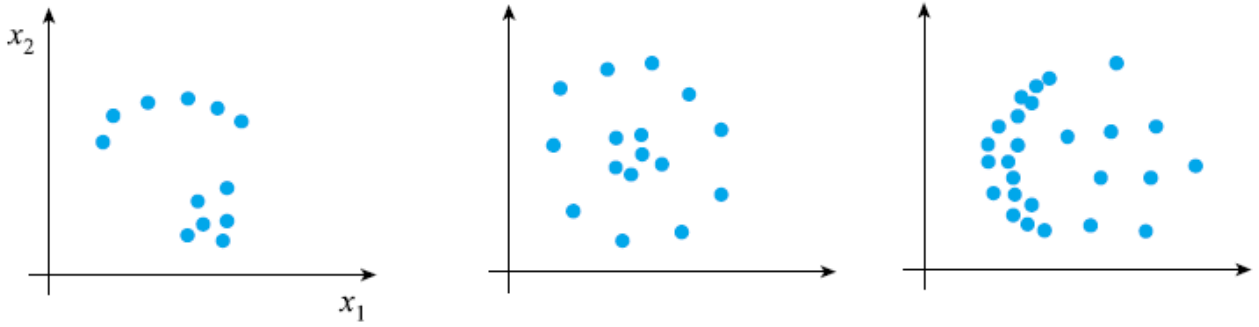
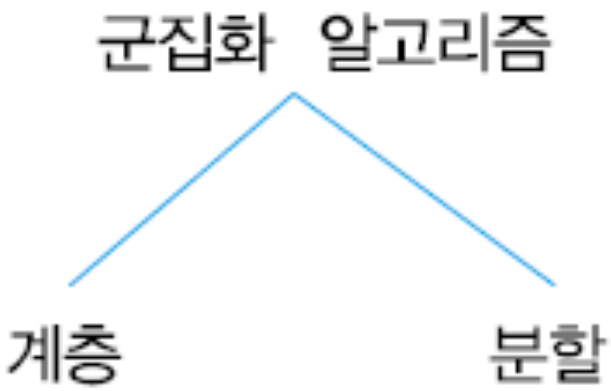


그림 10.5 다양한 군집 상황

군집화 알고리즘의 분류

- 분류 체계
  - 계층 군집화: 군집 결과를 계층을 나타내는 덴드로그램으로 표현
  - 분할 군집화: 각 샘플을 군집에 배정하는 연산 사용



MYONGJI UNIVERSITY

분할 군집화partitional clustering

- 군집화
  - 분할 군집화
    - 순차 알고리즘
    - k-means 알고리즘
    - 모델 기반 알고리즘
  - 계층 군집화
    - 응집 계층 알고리즘
    - 분열 계층 알고리즘
  - 신경망
    - 자기 조직화 맵

분할 군집화 | 순차 알고리즘

- 특성
  - 군집 개수를 찾아준다. (대신 임계값을 설정해주어야 한다.)
  - 순서에 민감하다. 빠르다 (선형 시간).

알고리즘 [10.3] 순차 알고리즘

입력: 샘플 집합 $X = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$, 임계값 T

출력: 군집 해 C, 군집의 개수 k

알고리즘:

1. $m = 1$; // 현재 군집 개수를 뜻한다.
2. $C = \{c_1 = \{\mathbf{x}_1\}\}$; // 한 개의 군집을 가지고 출발한다.
3. **for** $(i = 2$ **to** $N)$ {
4. 아래 식을 이용하여 $\mathbf{x}_i$와 가장 가까운 군집 c_q를 찾는다.

$$D(\mathbf{x}_i, c_q) = \min_{1 \le j \le m} D(\mathbf{x}_i, c_j)$$

5. **if** $(D(\mathbf{x}_i, c_q) > T)$ { m++; $c_m = \{\mathbf{x}_i\}$; $C = C \cup c_m$;}; // c_q로부터 멀기 때문에 새로운 군집을 만든다.
6. **else** $c_q = c_q \cup \mathbf{x}_i$; // 가까우니까 그것에 넣는다.
7. }
8. $k = m$;

MYONGJI UNIVERSITY

분할 군집화 | *k*-means 알고리즘

- 특성
  - 가장 널리 쓰인다. (직관적 이해. 구현 간편)
  - 군집 개수를 설정해주어야 한다.

알고리즘 [10.4] *k*-means 알고리즘

입력: 샘플 집합 $X = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$, 군집의 개수 k

출력: 군집 해 C

알고리즘:

1. k 개의 군집 중심 $Z = \{\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_k\}$를 초기화 한다.
2. **while** (TRUE) {
3. **for** ($i = 1$ **to** N) $\mathbf{x}_i$를 가장 가까운 군집 중심에 배정한다.
4. **if** (이 배정이 이전 루프의 배정과 같음) **break**;
5. **for** ($j = 1$ **to** k) $\mathbf{z}_j$에 배정된 샘플의 평균으로 $\mathbf{z}_j$를 대치한다.
6. }

MYONGJI UNIVERSITY

분할 군집화 | *k*-means 알고리즘

예제 10.5 *k*-means 알고리즘

7개 샘플을 k=3개의 군집으로 만드는 상황

$$\mathbf{x}_1 = (18,5)^T,\ \mathbf{x}_2 = (20,9)^T,\ \mathbf{x}_3 = (20,14)^T,\ \mathbf{x}_4 = (20,17)^T,\ \mathbf{x}_5 = (5,15)^T,\ \mathbf{x}_6 = (9,15)^T,$$
$$\mathbf{x}_7 = (6,20)^T$$

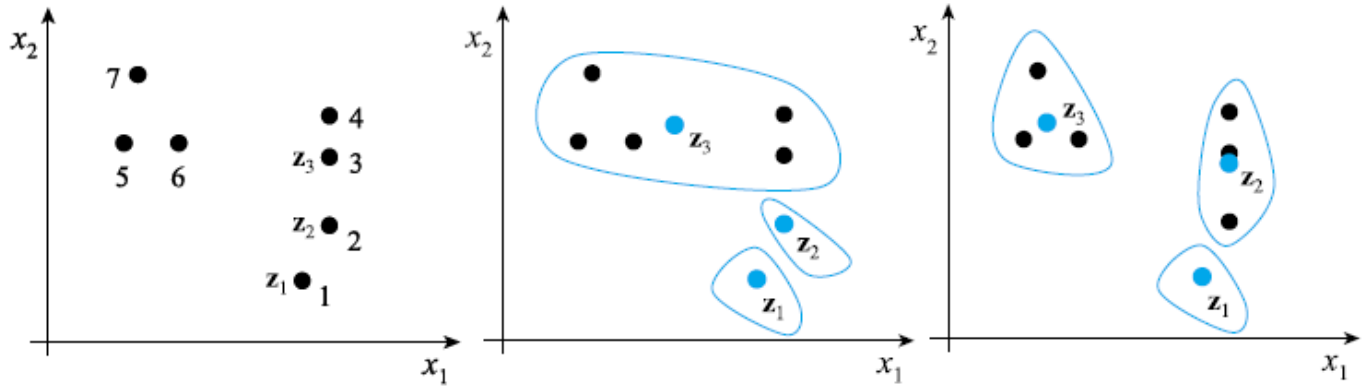
초기화에 의해 (그림 10.12(a)), $\{\mathbf{x}_1\}$은 $\mathbf{z}_1$

$\{\mathbf{x}_2\}$은 $\mathbf{z}_2$

$\{\mathbf{x}_3,\mathbf{x}_4,\mathbf{x}_5,\mathbf{x}_6,\mathbf{x}_7\}$은 $\mathbf{z}_3$

라인 5에 의해 (그림 10.12(b)),

$$\mathbf{z}_1 = \mathbf{x}_1 = (18,5)^T$$
$$\mathbf{z}_2 = \mathbf{x}_2 = (20,9)^T$$
$$\mathbf{z}_3 = (\mathbf{x}_3+\mathbf{x}_4+\mathbf{x}_5+\mathbf{x}_6+\mathbf{x}_7)/5 = (12,16.2)^T$$



(a) 초기 군집 중심 (b) 첫 번째 루프 후 (c) 두 번째 루프 후

그림 10.12 *k*-means의 동작 예

분할 군집화 | *k*-means 알고리즘

예제 10.5 *k*-means 알고리즘

두 번째 루프를 실행하면 (그림 10.12(c)),

$\{\mathbf{x}_1\}$은 $\mathbf{z}_1$

$\{\mathbf{x}_2,\mathbf{x}_3,\mathbf{x}_4\}$은 $\mathbf{z}_2$

$\{\mathbf{x}_5,\mathbf{x}_6,\mathbf{x}_7\}$은 $\mathbf{z}_3$

$$\mathbf{z}_1 = \mathbf{x}_1 = (18,5)^T$$

$$\mathbf{z}_2 = (\mathbf{x}_2+\mathbf{x}_3+\mathbf{x}_4)/3 = (20,13.333)^T$$

$$\mathbf{z}_3 = (\mathbf{x}_5+\mathbf{x}_6+\mathbf{x}_7)/3 = (6.667,16.667)^T$$

세 번째 루프는 그 이전과 결과가 같다. 따라서 멈춘다.

결국 출력은

$$C = \{\{\mathbf{x}_1\}, \{\mathbf{x}_2,\mathbf{x}_3,\mathbf{x}_4\}, \{\mathbf{x}_5,\mathbf{x}_6,\mathbf{x}_7\}\}$$

MYONGJI UNIVERSITY

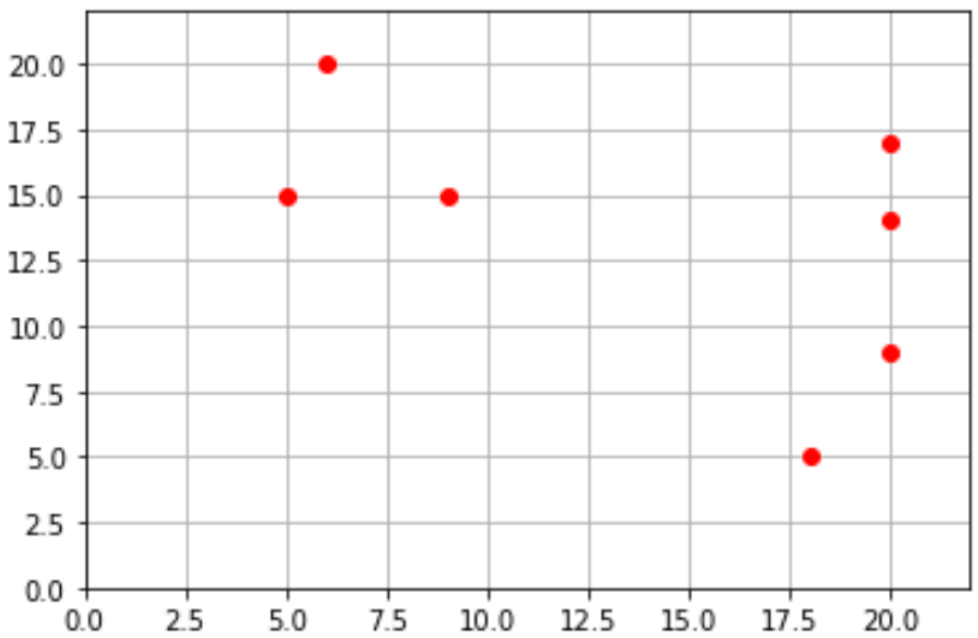
분할 군집화 | *k*-means 알고리즘 (Python)

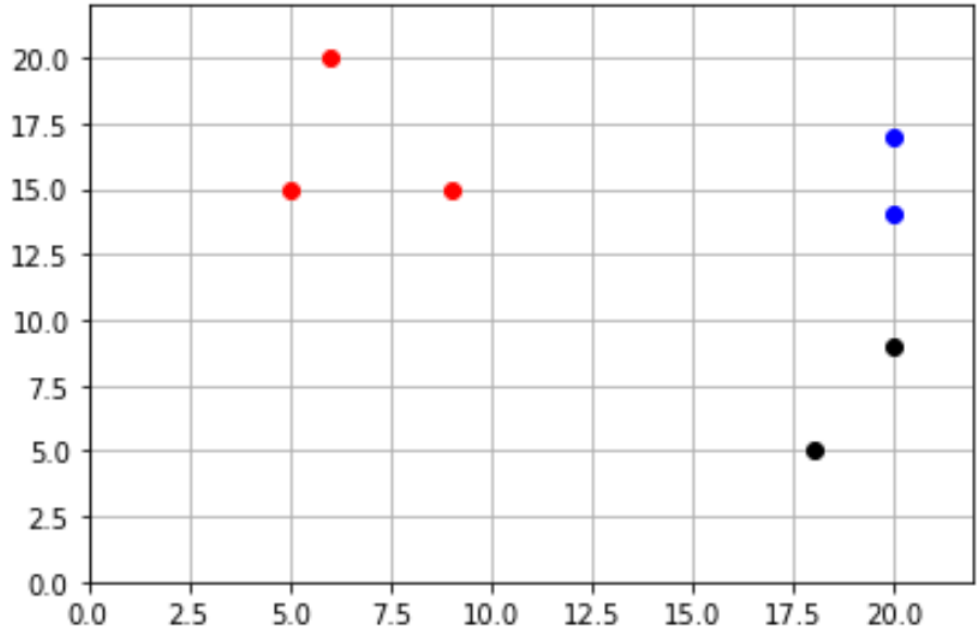
예제 10.5 *k*-means 알고리즘

7개 샘플을 k=3개의 군집으로 만드는 상황

$$\mathbf{x}_1 = (18,5)^T,\ \mathbf{x}_2 = (20,9)^T,\ \mathbf{x}_3 = (20,14)^T,\ \mathbf{x}_4 = (20,17)^T,\ \mathbf{x}_5 = (5,15)^T,\ \mathbf{x}_6 = (9,15)^T,$$

$$\mathbf{x}_7 = (6,20)^T$$



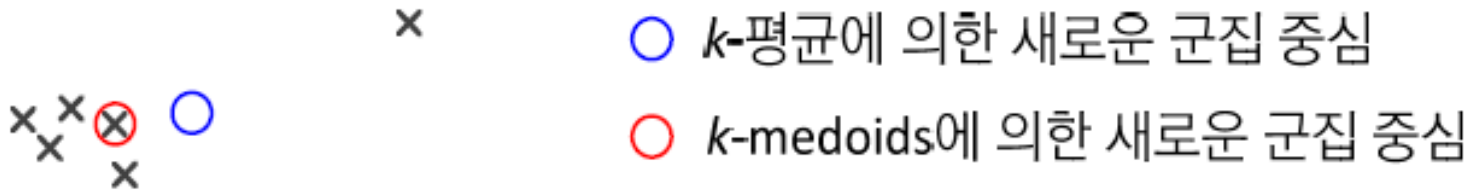


분할 군집화 | *k*-means 알고리즘

- 이론적 배경
  - (10.23)을 비용 함수로 하는 내리막 경사법의 일종

$$J(Z,\mathbf{U}) = \sum_{i=1}^{N}\sum_{j=1}^{k} u_{ji}\left\|\mathbf{x}_i - \mathbf{z}_j\right\|^2 \qquad (10.23)$$

  - 항상 지역 최적점으로 수렴한다. (전역 최적점 보장 못함)
  - 초기 군집 중심에 민감
  - 빠르다.
  - 외톨이에 민감하다. (*k*-medoids는 덜 민감)



k-평균과 *k*-medoids가 군집 중심을 갱신하는 과정

MYONGJI UNIVERSITY

분할 군집화 | 모델 기반 알고리즘

- 샘플로부터 가우시언을 추정하고 그 결과에 따라 군집 배정
  - 가우시언 추정은 EM 알고리즘을 사용할 수 있다.

알고리즘 [10.6] 가우시언 모델 기반 알고리즘

입력: 샘플 집합 $X = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$, 군집의 개수 k

출력: 군집 해 C

알고리즘:

1. X를 가지고 EM 알고리즘을 수행하여 가우시언 G_j, $j = 1, \cdots, k$를 추정한다.
2. (10.24)의 규칙으로 각각의 샘플을 군집에 배정한다.

$$\left.\begin{array}{l} \mathbf{x}_i\text{를 } c_q\text{에 배정한다.} \\ \text{이때 } P(G_q \mid \mathbf{x}_i) > P(G_j \mid \mathbf{x}_i),\ j = 1, \cdots, k,\ j \neq q \end{array}\right\} \quad (10.24)$$

MYONGJI UNIVERSITY

12주차 예제

- 예제 10.5에서 *k*-means 알고리즘을 사용하여 2개의 군집으로 만들 때 어떤 군집 해를 얻게 되는지 알아보자.
  - 1) 초기점을 $\mathbf{x}_3$, $\mathbf{x}_7$로 설정하고 직접 군집 해를 구해보자.
  - 2) Python을 통해 군집 해를 구해보고 위에서 구한 군집 해와 차이가 있는지 알아보자.

Thank you

MYONGJI
UNIVERSITY