

기계 학습

Part II

4. 결정 트리

4.1 결정트리의 형태

4.2 결정트리 학습 알고리즘

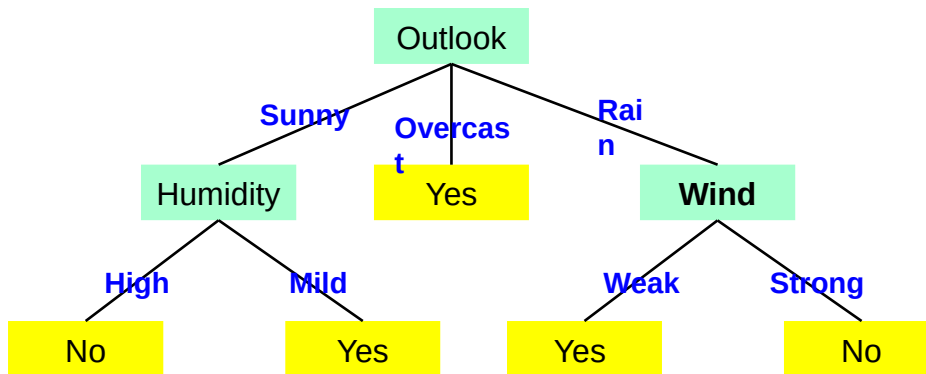
4.3 결정트리를 이용한 회귀

4.1 결정트리의 형태

❖ 결정트리(decision tree)

▪ 트리 형태로 의사결정 지식을 표현한 것

- 내부 노드(internal node) : 비교 속성
- 간선(edge) : 속성 값
- 단말 노드(terminal node) : 부류(class), 대표값



Day 날짜	Outlook 조망	Temperature 기온	Humidity 습도	Wind 바람	PlayTennis 테니스 여부
Day1	Sunny	Hot	High	Weak	No
Day2	Sunny	Hot	High	Strong	No
Day3	Overcast	Hot	High	Weak	Yes
Day4	Rain	Mild	High	Weak	Yes
Day5	Rain	Cool	Normal	Weak	Yes
Day6	Rain	Cool	Normal	Strong	No
Day7	Overcast	Cool	Normal	Strong	Yes
Day8	Sunny	Mild	High	Weak	No
Day9	Sunny	Cool	Normal	Weak	Yes
Day10	Rain	Mild	Normal	Weak	Yes
Day11	Sunny	Mild	Normal	Strong	Yes
Day12	Overcast	Mild	High	Strong	Yes
Day13	Overcast	Hot	Normal	Weak	Yes
Day14	Rain	Mild	High	Strong	No

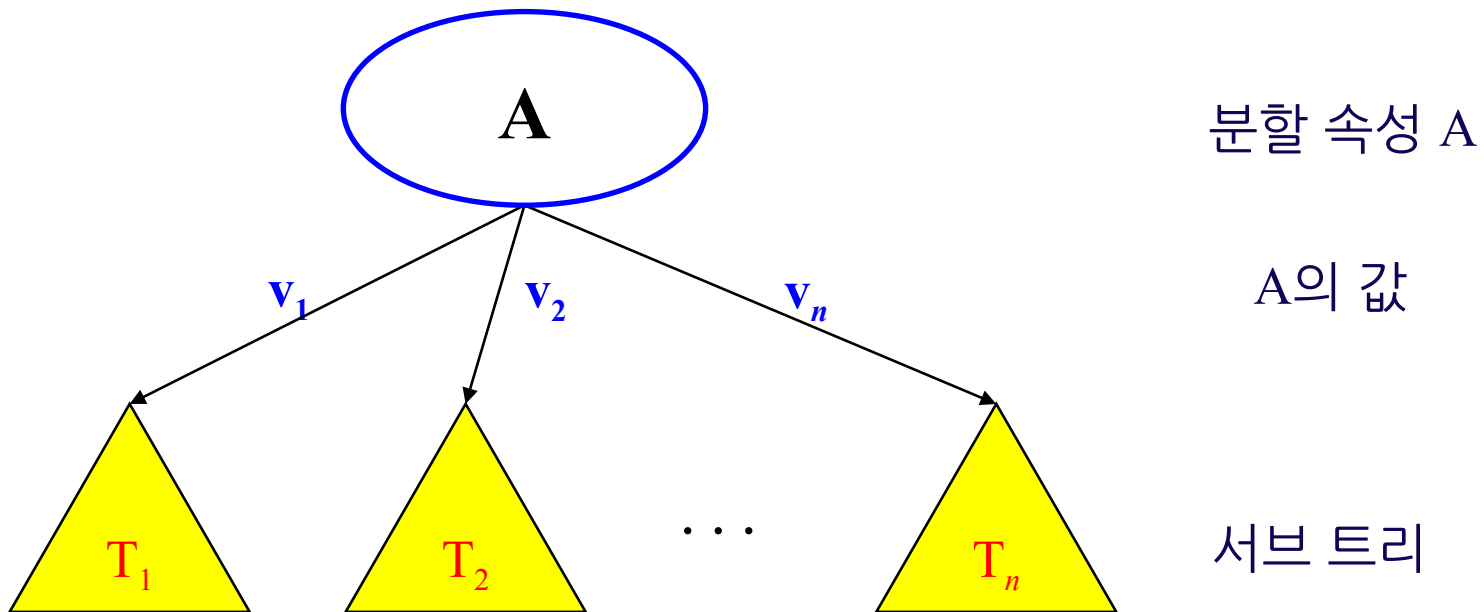
IF Outlook = Sunny AND Humidity = High THEN Answer = No

Outlook 조망	Temperature 기온	Humidity 습도	Wind 바람	PlayTennis 테니스 여부
Sunny	Hot	Mild	Weak	?
Rain	Hot	High	Weak	?

4.2 결정트리 학습 알고리즘

❖ 결정 트리 (decision tree) 알고리즘

- 모든 데이터를 포함한 하나의 노드로 구성된 트리에서 시작
- 반복적인 노드 분할 과정
 1. 분할 속성 (splitting attribute)을 선택
 2. 속성값에 따라 서브트리 (subtree)를 생성
 3. 데이터를 속성값에 따라 분배

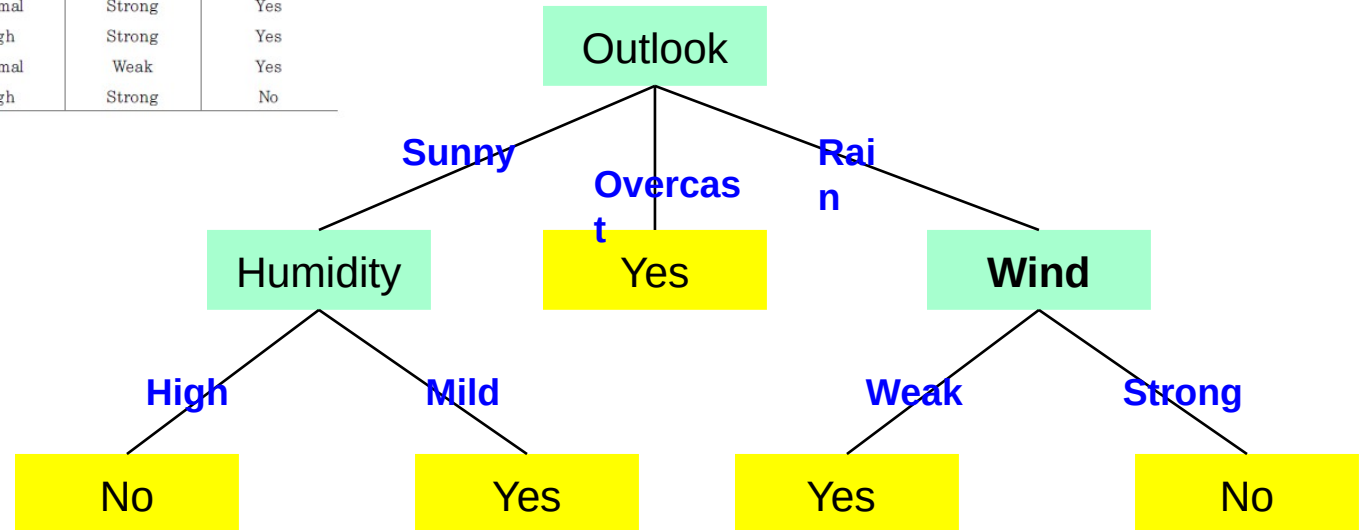


결정트리 학습 알고리즘

❖ 결정 트리 (decision tree)

■ 간단한 트리

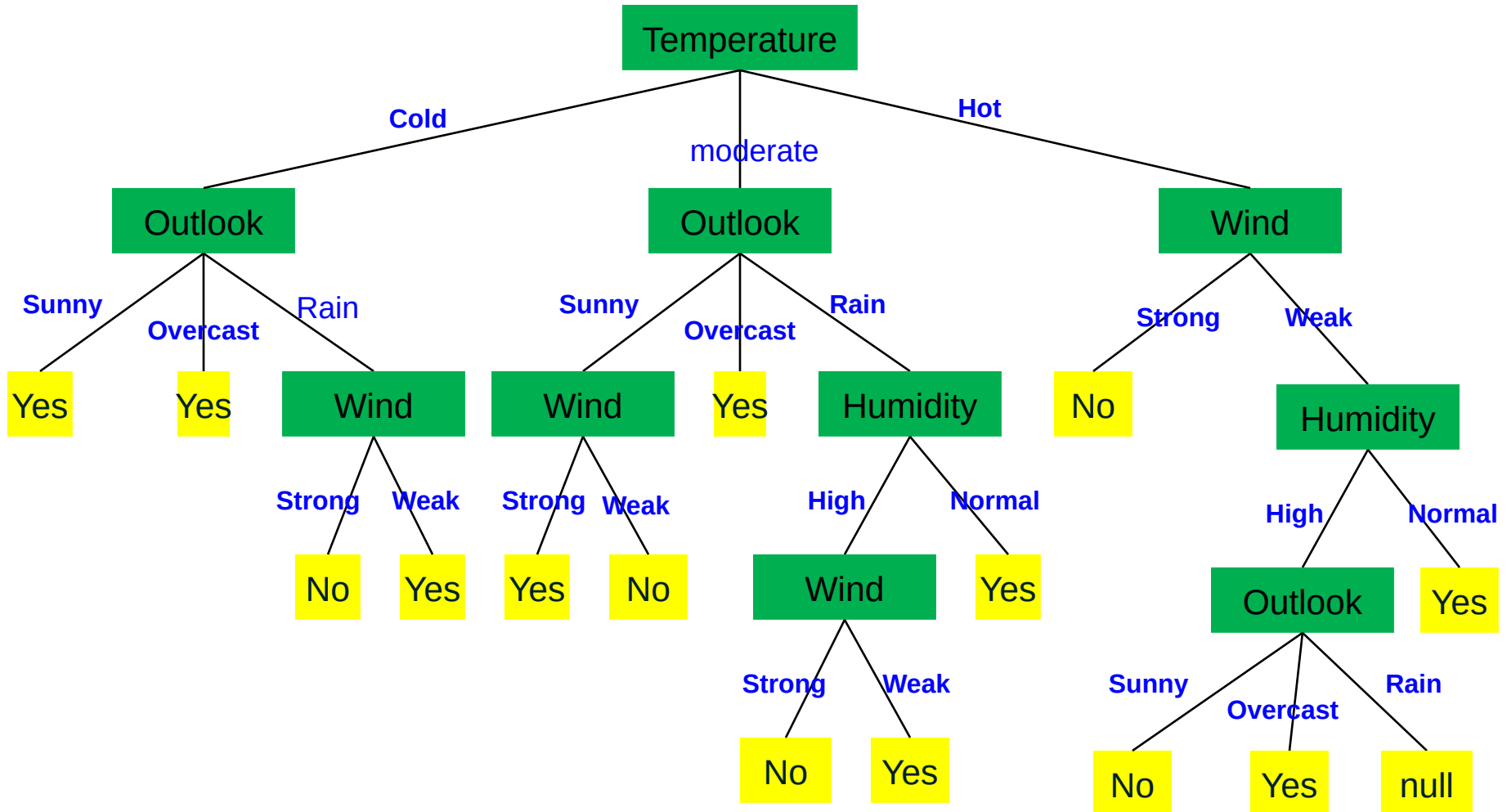
Day 날짜	Outlook 조망	Temperature 기온	Humidity 습도	Wind 바람	PlayTennis 테니스 여부
Day1	Sunny	Hot	High	Weak	No
Day2	Sunny	Hot	High	Strong	No
Day3	Overcast	Hot	High	Weak	Yes
Day4	Rain	Mild	High	Weak	Yes
Day5	Rain	Cool	Normal	Weak	Yes
Day6	Rain	Cool	Normal	Strong	No
Day7	Overcast	Cool	Normal	Strong	Yes
Day8	Sunny	Mild	High	Weak	No
Day9	Sunny	Cool	Normal	Weak	Yes
Day10	Rain	Mild	Normal	Weak	Yes
Day11	Sunny	Mild	Normal	Strong	Yes
Day12	Overcast	Mild	High	Strong	Yes
Day13	Overcast	Hot	Normal	Weak	Yes
Day14	Rain	Mild	High	Strong	No



결정트리 학습 알고리즘

❖ 결정 트리 (decision tree)

- 복잡한 트리



결정트리 학습 알고리즘

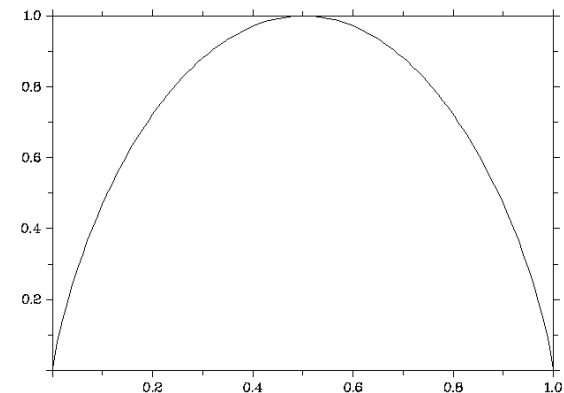
✧ 분할 속성(splitting attribute) 결정

- 어떤 속성을 선택하는 것이 효율적인가
 - 분할한 결과가 **가능하면 동질적인(pure) 것으로** 만드는 속성 선택
- **엔트로피**(Entropy)
 - 동질적인 정도 측정 가능 척도
 - 원래 정보량(amount of information) 측정 목적의 척도

$$I = - \sum_c p(c) \log_2 p(c)$$

– $p(c)$: 부류 c 에 속하는 것의 비율

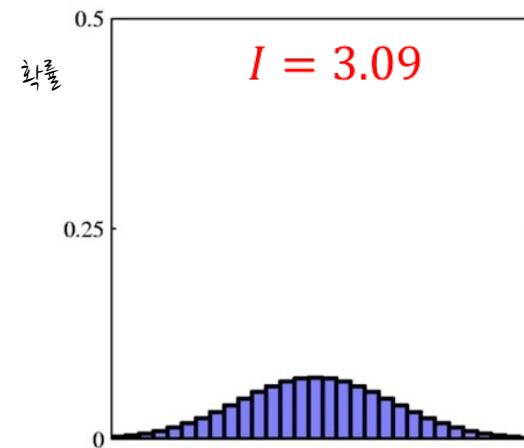
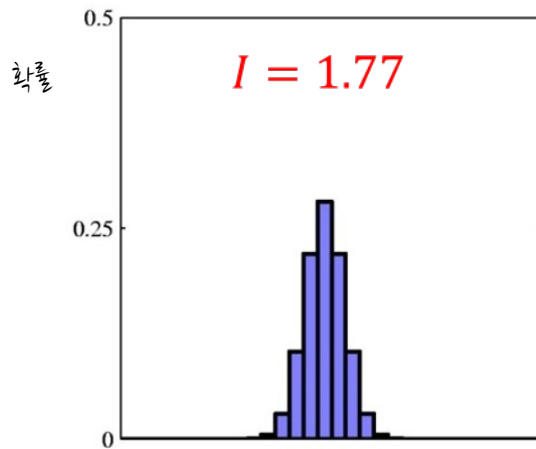
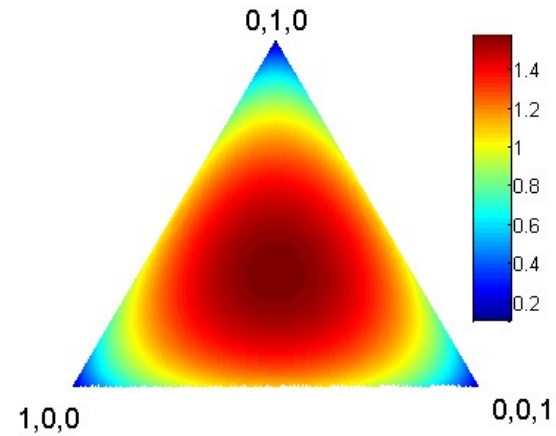
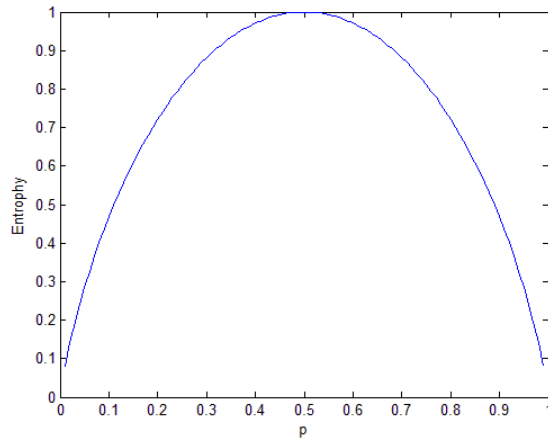
- 2개 부류가 있는 경우 엔트로피



결정트리 학습 알고리즘

❖ 엔트로피의 특성

- 섞인 정도가 클수록 큰 값



결정트리 학습 알고리즘

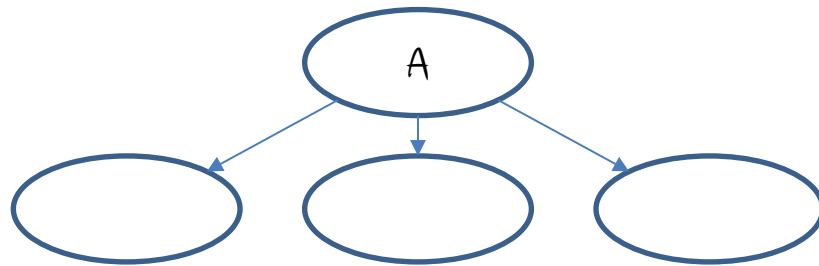
❖ 정보 이득 (information gain)

- $IG = I - I_{res}$

- I_{res} : 특정 속성으로 분할한 후의 각 부분집합의 정보량의 가중평균

$$I_{res} = - \sum_v p(v) \sum_c p(c|v) \log_2 p(c|v)$$

$$IG = I - I_{res}(A) = - \sum_c p(c) \log_2 p(c) + \sum_v p(v) \sum_c p(c|v) \log_2 p(c|v)$$



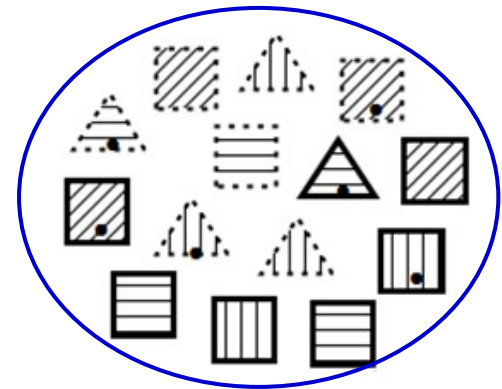
- 정보이득이 클 수록 우수한 분할 속성

결정트리 학습 알고리즘

❖ 학습 데이터의 예

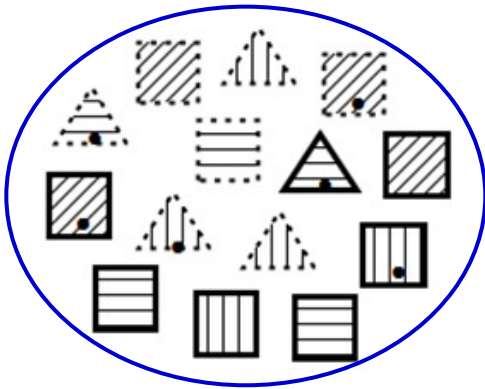
- 부류(class) 정보가 있는 데이터

	속성			부류
	<i>Pattern</i>	<i>Outline</i>	<i>Dot</i>	<i>Shape</i>
1	수직	점선	무	삼각형
2	수직	점선	유	삼각형
3	대각선	점선	무	사각형
4	수평	점선	무	사각형
5	수평	실선	무	사각형
6	수평	실선	유	삼각형
7	수직	실선	무	사각형
8	수직	점선	무	삼각형
9	대각선	실선	유	사각형
10	수평	실선	무	사각형
11	수직	실선	유	사각형
12	대각선	점선	유	사각형
13	대각선	실선	무	사각형
14	수평	점선	유	삼각형



결정트리 학습 알고리즘

❖ 엔트로피 계산



- 9 □ (사각형)

- 5 △ (삼각형)

- **부류별 확률(class probability)**

$$p(\square) = \frac{9}{14}$$

$$p(\triangle) = \frac{5}{14}$$

- **엔트로피(entropy)**

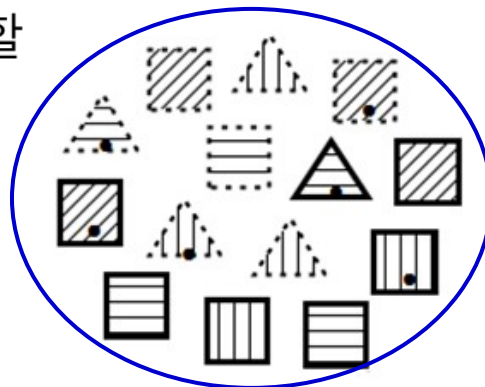
$$I = - \sum_c p(c) \log_2 p(c)$$

$$I = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14} = 0.940 \text{ bits}$$

결정트리 학습 알고리즘

❖ 데이터 집합 분할과 정보이득

▪ Pattern 기준 분할

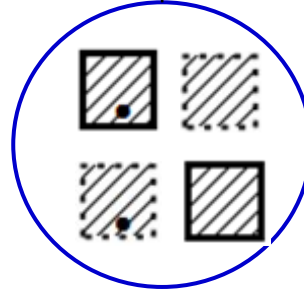


$$I = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14} = 0.940$$

수평

Pattern
대각선

수직



$$I_{horizontal} = -\frac{2}{5} \log_2 \frac{2}{5} - \frac{3}{5} \log_2 \frac{3}{5} = 0.971 \quad I_{diagonal} = -\frac{0}{4} \log_2 \frac{0}{4} - \frac{4}{4} \log_2 \frac{4}{4} = 0.0 \quad I_{vertical} = -\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5} = 0.971$$

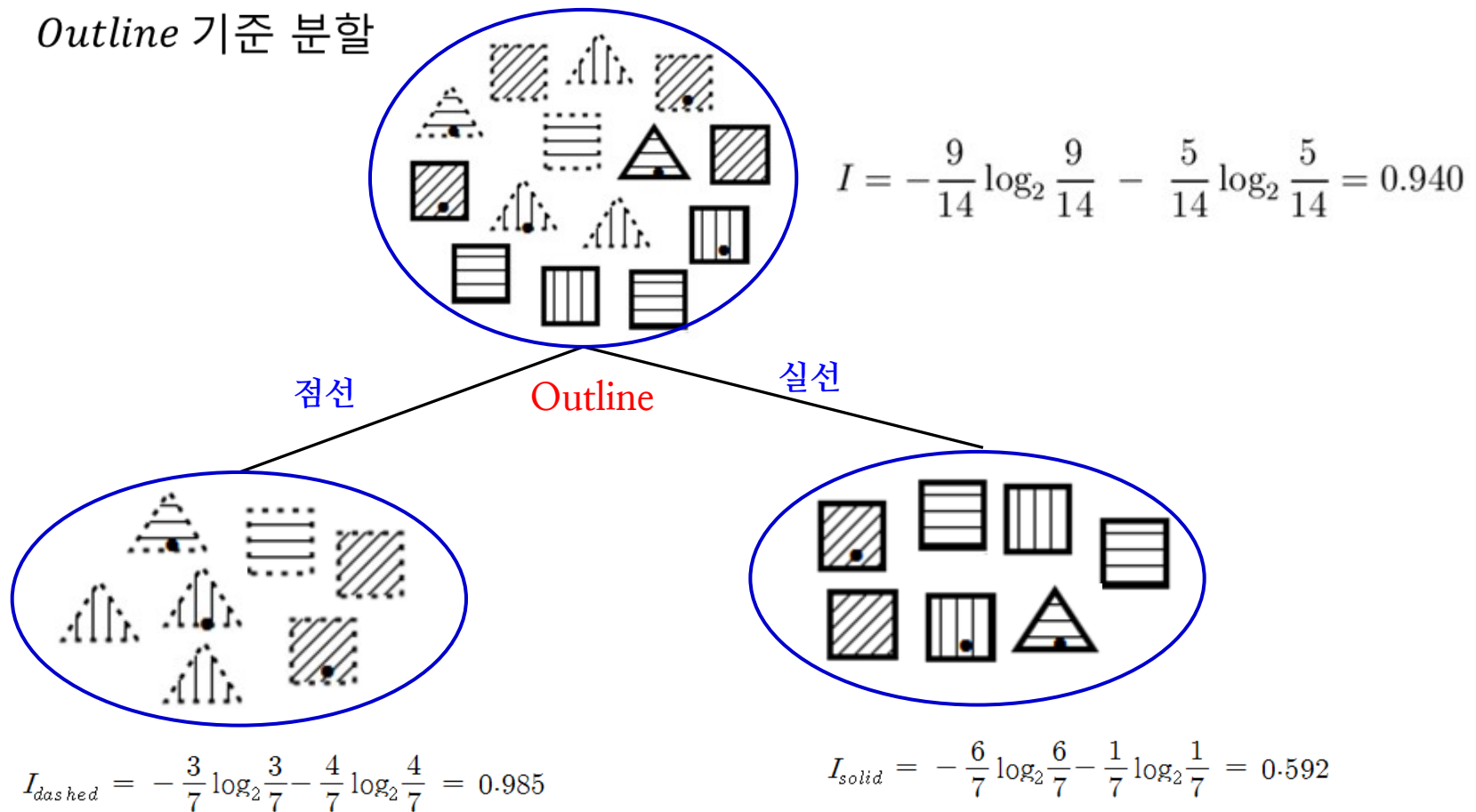
$$I_{res}(Pattern) = \sum p(v) I(v) = \frac{5}{14} \cdot 0.971 + \frac{4}{14} \cdot 0 + \frac{5}{14} \cdot 0.971 = 0.694$$

$$IG(Pattern) = I - I_{res}(Pattern) = 0.940 - 0.694 = 0.246$$

결정트리 학습 알고리즘

❖ 데이터 집합 분할과 정보이득

- *Outline* 기준 분할



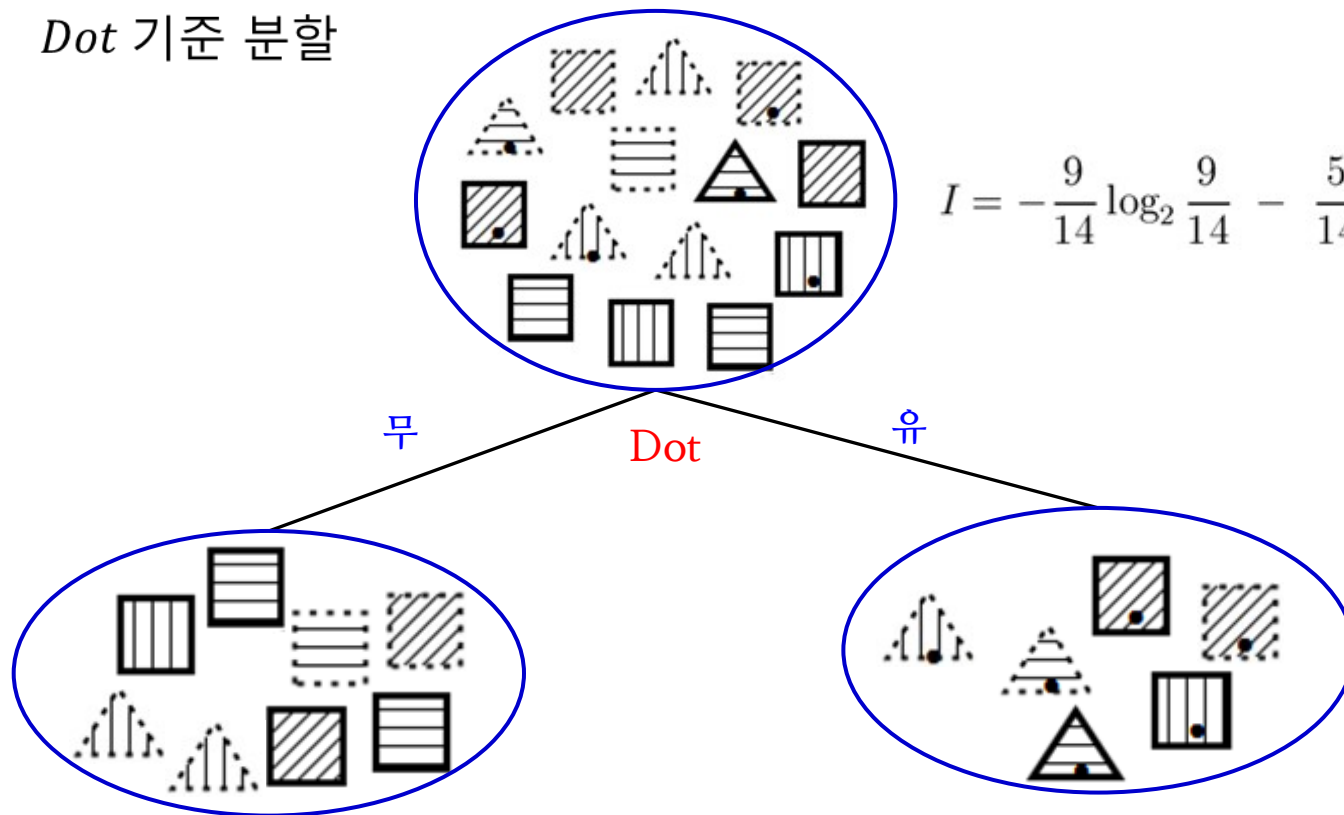
$$I_{res}(Outline) = \sum p(v) I(v) = \frac{7}{14} \cdot 0.985 + \frac{7}{14} \cdot 0.592 = 0.789$$

$$IG(Outline) = I - I_{res}(Outline) = 0.940 - 0.789 = 0.151$$

결정트리 학습 알고리즘

❖ 데이터 집합 분할과 정보이득

- Dot 기준 분할



$$I = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14} = 0.940$$

$$I_{no} = -\frac{2}{8} \log_2 \frac{2}{8} - \frac{6}{8} \log_2 \frac{6}{8} = 0.811$$

$$I_{yes} = -\frac{3}{6} \log_2 \frac{3}{6} - \frac{3}{6} \log_2 \frac{3}{6} = 1.000$$

$$I_{res}(Dot) = \sum p(v)I(v) = \frac{8}{14} \cdot 0.811 + \frac{6}{14} \cdot 1.000 = 0.892$$

$$IG(Dot) = I - I_{res}(Dot) = 0.940 - 0.892 = 0.048$$

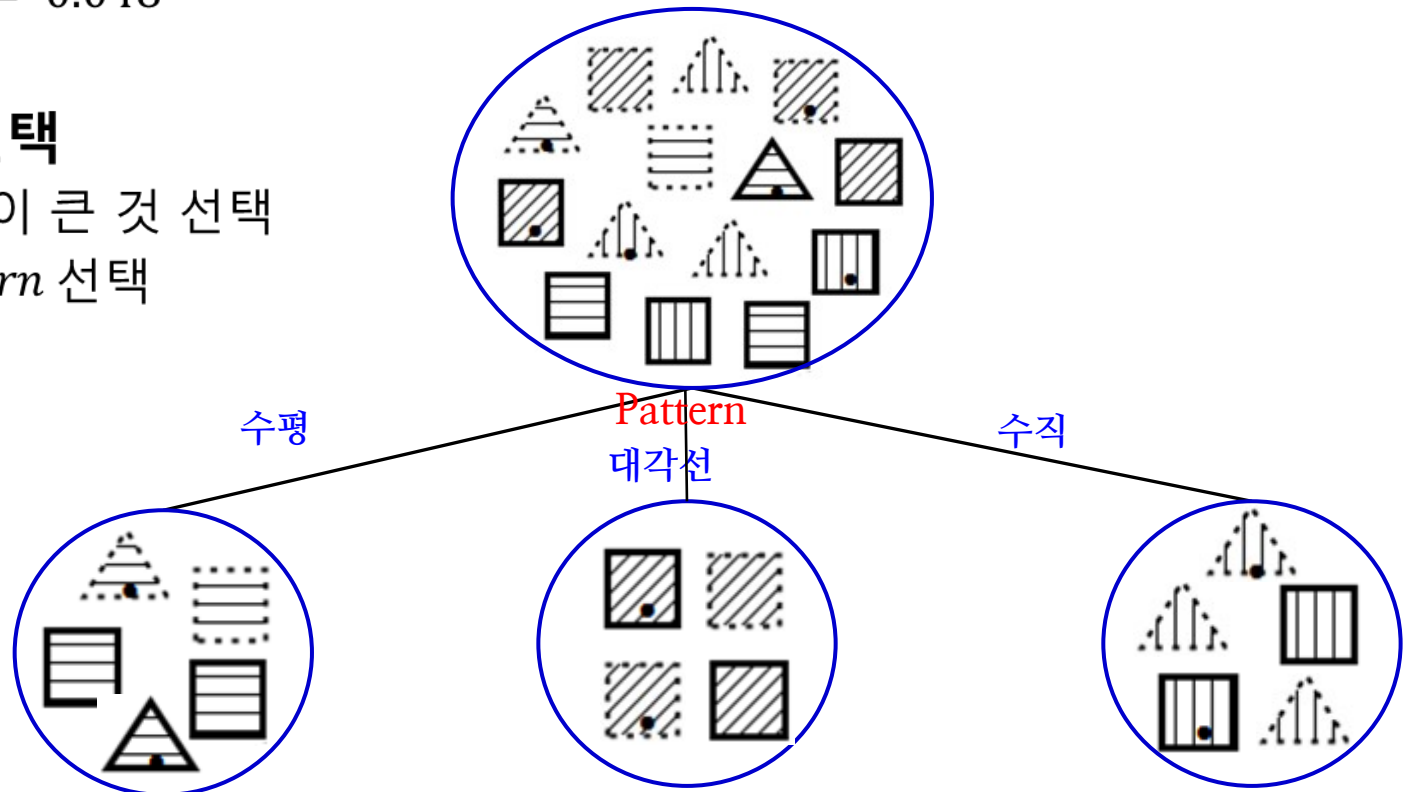
결정트리 학습 알고리즘

❖ 속성별 정보 이득

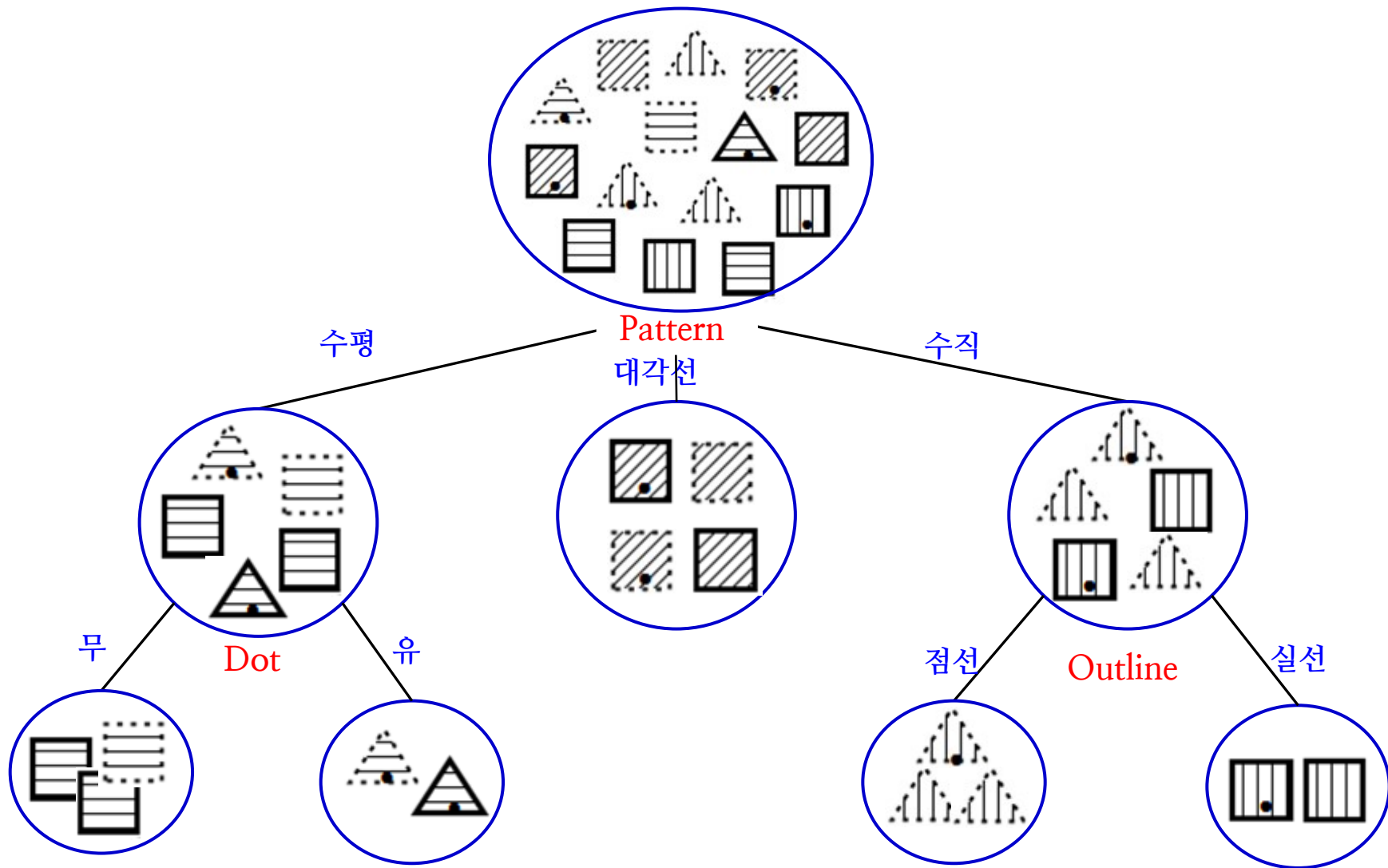
- $IG(Pattern) = 0.246$
- $IG(Outline) = 0.151$
- $IG(Dot) = 0.048$

❖ 분할속성 선택

- 정보이득이 큰 것 선택
 - *Pattern* 선택

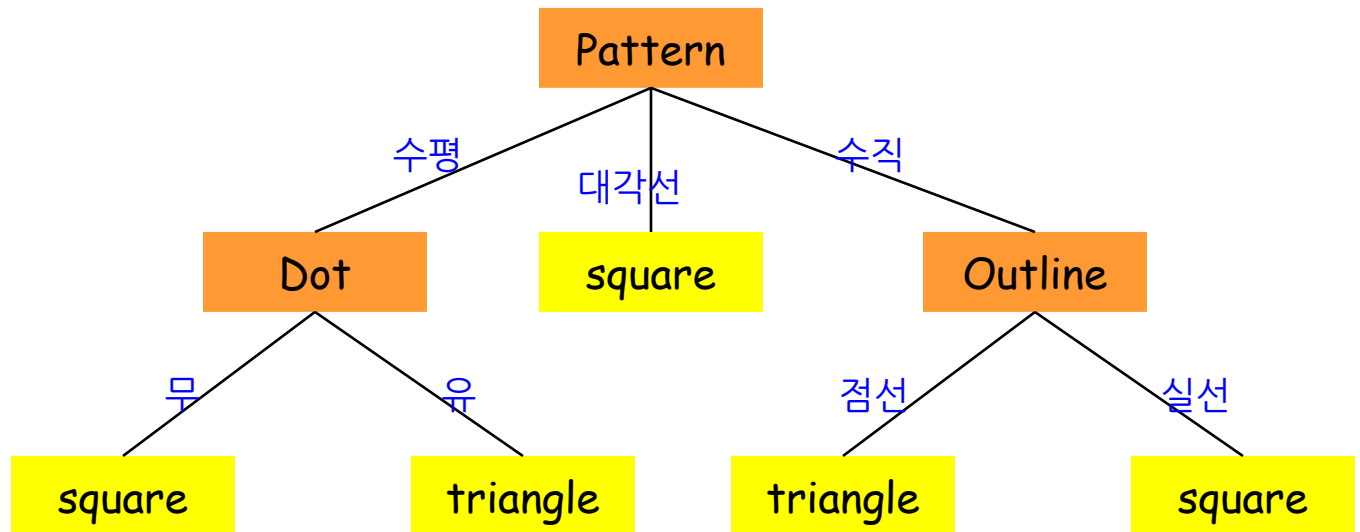


결정트리 학습 알고리즘



결정트리 학습 알고리즘

❖ 최종 결정트리



결정트리 학습 알고리즘

❖ 정보이득(information gain) 척도의 단점

$$IG = I - I_{res}(A) = - \sum_c p(c) \log_2 p(c) + \sum_v p(v) \sum_c p(c|v) \log_2 p(c|v)$$

- 속성값이 많은 것 선호
 - 예. 학번, 이름 등
- 속성값이 많으면 데이터집합을 많은 부분집합으로 분할
 - 작은 부분집합은 동질적인 경향

❖ 개선 척도

- 정보이득비 (information gain ratio)
- 지니 지수 (Gini index)

결정트리 학습 알고리즘

✧ 정보이득 비(information gain ratio) 척도

- 정보이득(information gain) 척도를 개선한 것
 - 속성값이 많은 속성에 대해 불이익

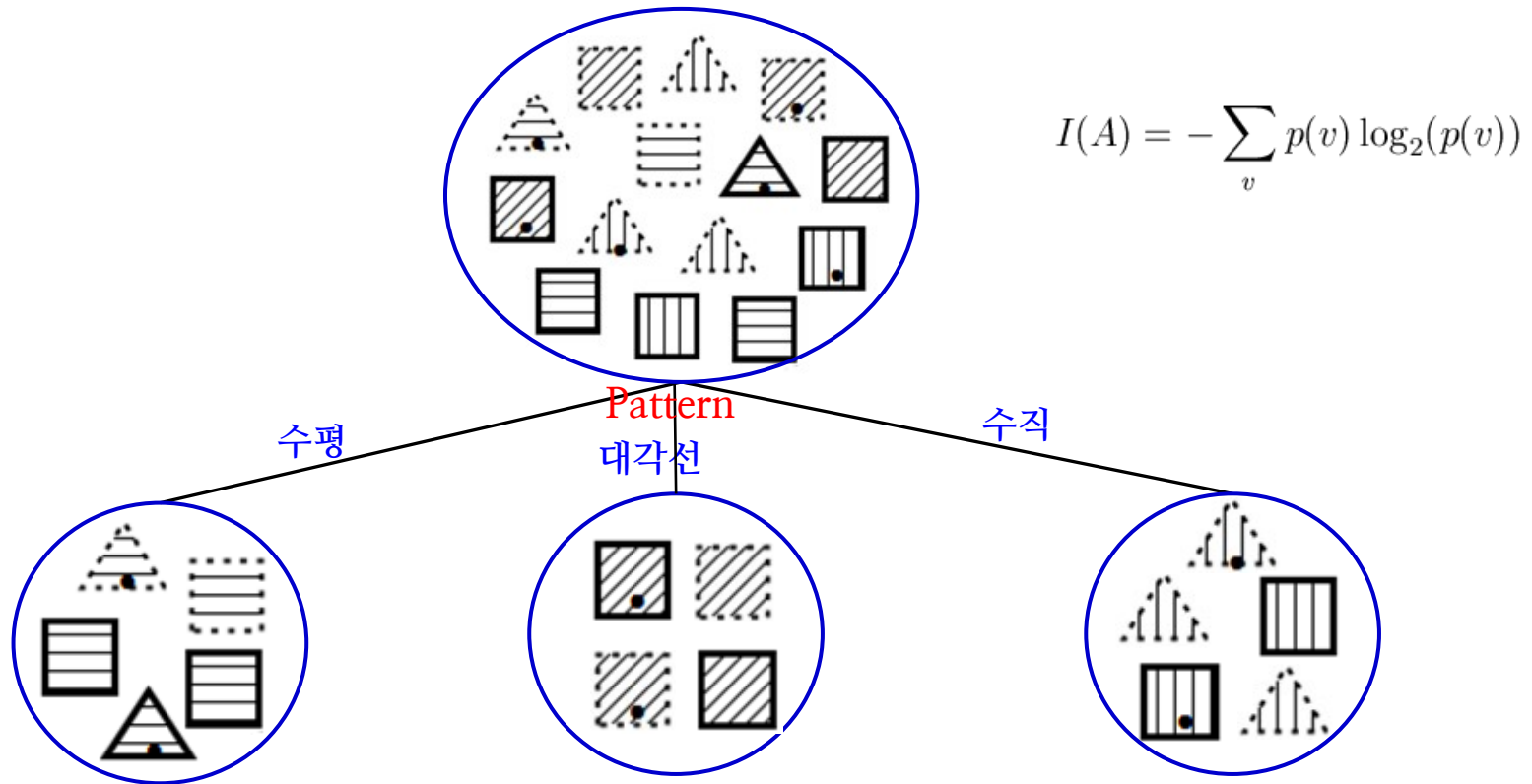
$$GainRatio(A) = \frac{IG(A)}{I(A)} = \frac{I - I_{res}(A)}{I(A)}$$

- $I(A)$
 - 속성 A의 속성값을 부류(class)로 간주하여 계산한 엔트로피
 - 속성값이 많을 수록 커지는 경향

$$I(A) = - \sum_v p(v) \log_2(p(v))$$

결정트리 학습 알고리즘

❖ 정보이득 비



$$I(A) = - \sum_v p(v) \log_2(p(v))$$

$$I(Pattern) = - \frac{5}{14} \log_2 \frac{5}{14} - \frac{5}{14} \log_2 \frac{5}{14} - \frac{4}{14} \log_2 \frac{4}{14} = 1.58$$

$$IG(Pattern) = I - I_{res}(Pattern) = 0.940 - 0.694 = 0.246$$

$$GainRatio(Pattern) = \frac{IG(Pattern)}{I(Pattern)} = \frac{0.246}{1.58} = 0.156$$

결정트리 학습 알고리즘

❖ 정보이득 vs 정보이득 비

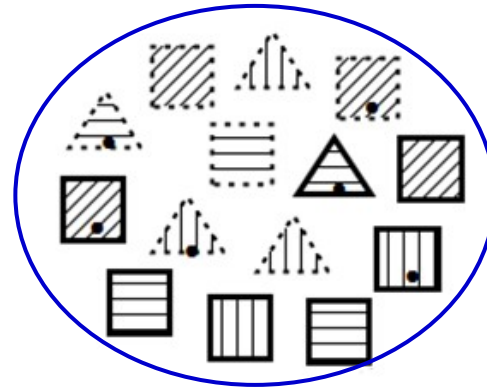
속성	속성의 개수	정보이득	정보이득비
Pattern	3	0.247	0.156
Outline	2	0.152	0.152
Dot	2	0.048	0.049

결정트리 학습 알고리즘

❖ 지니 지수(Gini index)

- 데이터 집합에 대한 지니 값
 - i, j 가 부류(class)를 나타낼 때

$$Gini = \sum_{i \neq j} p(i)p(j)$$



$$p(\square) = \frac{9}{14}$$

$$p(\triangle) = \frac{5}{14}$$

$$Gini = \frac{9}{14} \times \frac{5}{14} = 0.230$$

- 속성 A에 대한 지니 지수값 가중평균

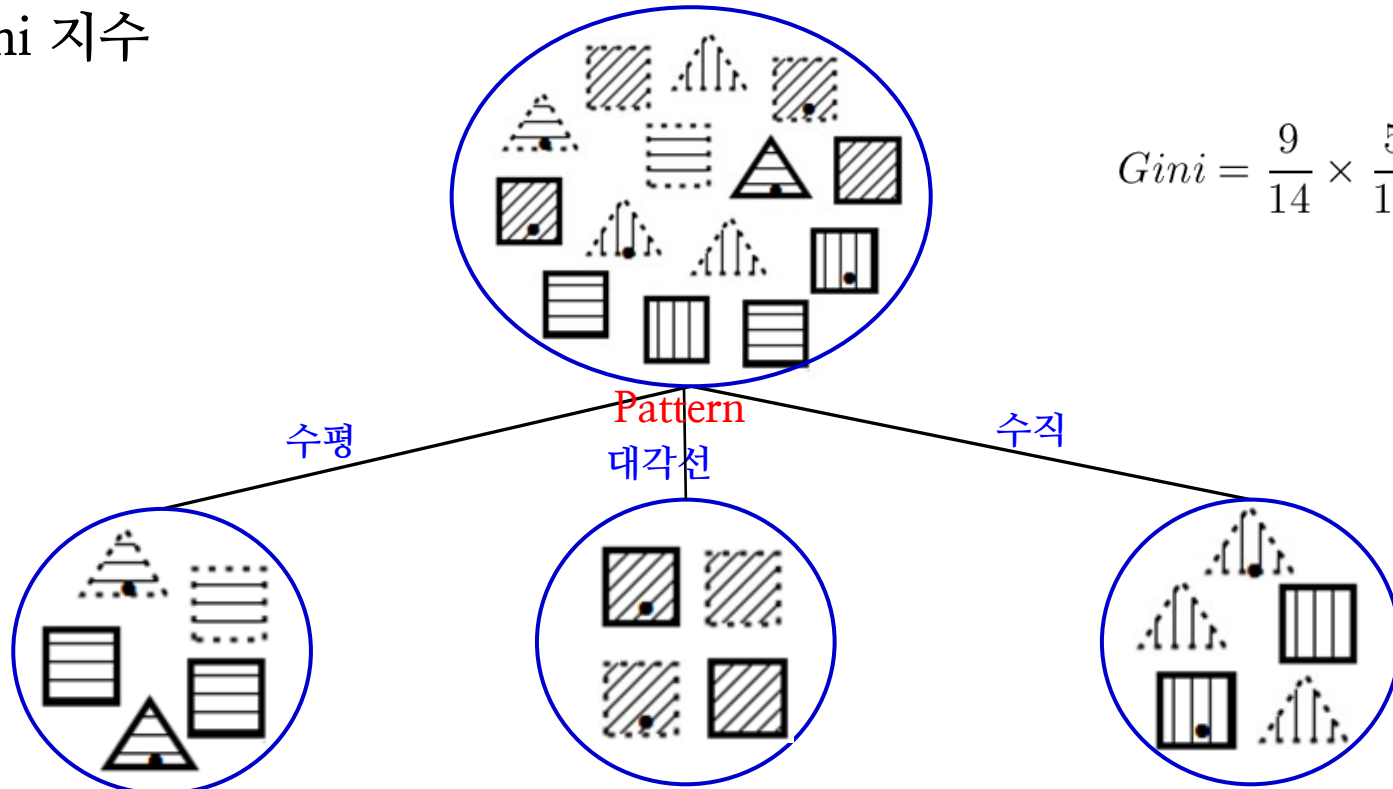
$$Gini(A) = \sum_v p(v) \sum_{i \neq j} p(i|v)p(j|v)$$

- 지니 지수 이득 (gini index gain)

$$GiniGain(A) = Gini - Gini(A)$$

결정트리 학습 알고리즘

❖ Gini 지수



$$Gini = \frac{9}{14} \times \frac{5}{14} = 0.230$$

$$Gini(A) = \sum_v p(v) \sum_{i \neq j} p(i|v) p(j|v)$$

$$Gini(Pattern) = \frac{5}{14} \times \left(\frac{3}{5} \times \frac{2}{5} \right) + \frac{5}{14} \times \left(\frac{2}{5} \times \frac{3}{5} \right) + \frac{4}{14} \times \left(\frac{4}{4} \times \frac{0}{4} \right) = 0.171$$

$$Gini\ Gain(Pattern) = 0.230 - 0.171 = 0.058$$

결정트리 학습 알고리즘

❖ 분할속성 평가 척도 비교

속성	정보이득	정보이득비	지니이득
Pattern	0.247	0.156	0.058
Outline	0.152	0.152	0.046
Dot	0.048	0.049	0.015

결정트리 학습 알고리즘

❖ 결정트리 알고리즘

- ID3 알고리즘
 - 범주형(categorical) 속성값을 갖는 데이터에 대한 결정트리 학습
 - 예. PlayTennis, 삼각형/사각형 문제
- C4.5 알고리즘
 - 범주형 속성값과 수치형 속성값을 갖는 데이터로부터 결정트리 학습
 - ID3를 개선한 알고리즘
- C5.0 알고리즘
 - C4.5를 개선한 알고리즘
- CART 알고리즘
 - 수치형 속성을 갖는 데이터에 대해 적용

4.3 결정트리를 이용한 회귀

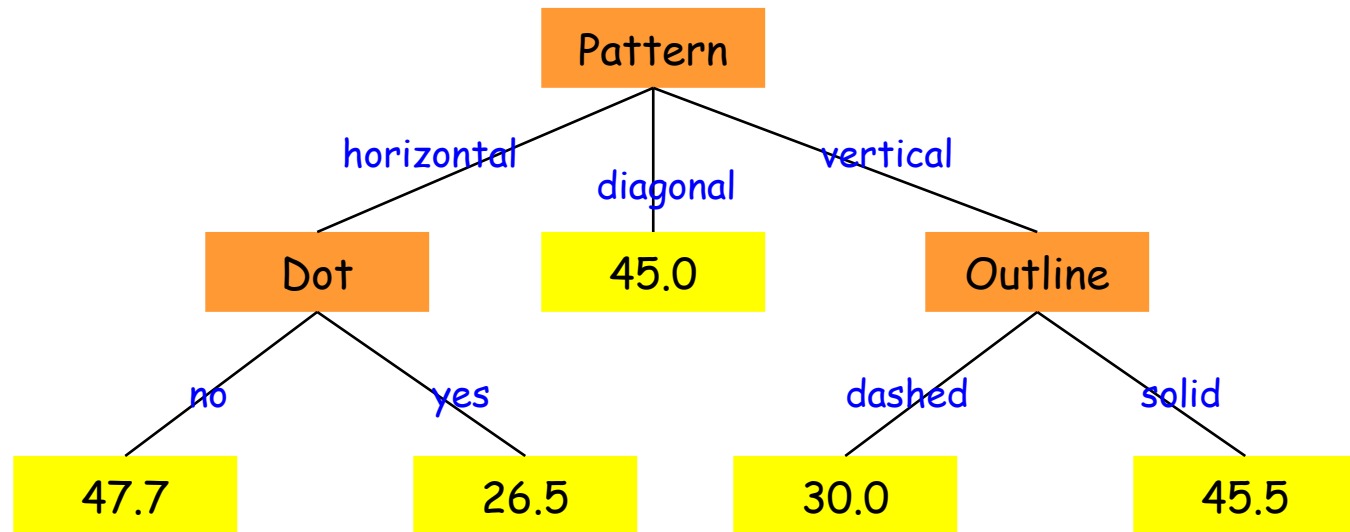
- ❖ 회귀(regression)를 위한 결정트리
 - 출력값이 수치값

표 4.5 도형 면적에 대한 데이터

	속성			면적
	<i>Pattern</i>	<i>Outline</i>	<i>Dot</i>	<i>Area</i>
1	수직	점선	무	25
2	수직	점선	유	30
3	대각선	점선	무	46
4	수평	점선	무	45
5	수평	실선	무	52
6	수평	실선	유	23
7	수직	실선	무	43
8	수직	점선	무	35
9	대각선	실선	유	38
10	수평	실선	무	46
11	수직	실선	유	48
12	대각선	점선	유	52
13	대각선	실선	무	44
14	수평	점선	유	30

결정트리를 이용한 회귀

❖ 회귀(regression)를 위한 결정트리



결정트리를 이용한 회귀

❖ 회귀 (regression)를 위한 결정트리

- 분류를 위한 결정트리와 차이점
 - 단말노드가 부류(class)가 아닌 수치값(numerical value)임
 - 해당 조건을 만족하는 것들이 가지는 대표값
- 분할 속성 선택
 - 표준편차 축소(reduction of standard deviation) SDR 를 최대화하는 속성 선택

$$SDR(A) = SD - SD(A)$$

– 표준편차 $SD = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - m)^2}$ m : 평균

– $SD(A)$

» 속성 A를 기준으로 분할 후의 부분 집합별 표준편차의 가중평균

결정트리를 이용한 회귀

❖ 회귀(regression)를 위한 결정트리

Area
26
30
48
46
62
23
43
36
38
48
48
62
44
30

← $SD = 9.67$

		Area의 표준편차	개수
<i>Pattern</i>	수평	12.15	5
	수직	9.36	5
	대각선	5.77	4
			14

$$SD(Pattern) = \frac{5}{14} \times 12.15 + \frac{5}{14} \times 9.36 + \frac{4}{14} \times 5.77 = 9.05$$

$$SDR(Pattern) = SD - SD(Pattern) = 9.67 - 9.05 = 0.61$$



5. 앙상블 분류기

❖ 앙상블 분류기 (ensemble classifier)

- 주어진 학습 데이터 집합에 대해서 여러 개의 서로 다른 분류기를 만들고, 이들 분류기의 판정 결과를 투표 방식 (voting method)이나 가중치 투표 방식 (weighted voting method)으로 결합
- **붓스트랩** (bootstrap)
 - 주어진 학습 데이터 집합에서 복원추출 (resampling with replacement)하여 다수의 학습 데이터 집합을 만들어내는 기법

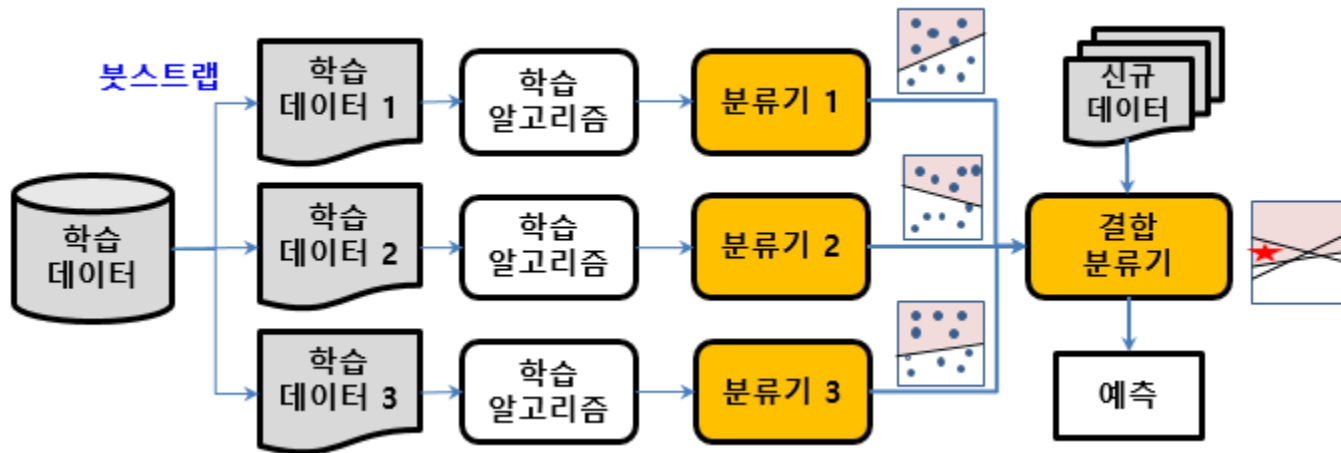


- 배깅 (bagging, bootstrap aggregating)
- 부스팅 (boosting)

5.1 배깅 알고리즘

❖ 배깅 (bagging, bootstrap aggregating)

- 붓스트랩을 통해 여러 개의 학습 데이터 집합을 만들고, 각 학습 데이터 집합별로 분류기를 만들어, 이들이 투표나 가중치 투표를 하여 최종 판정을 하는 기법



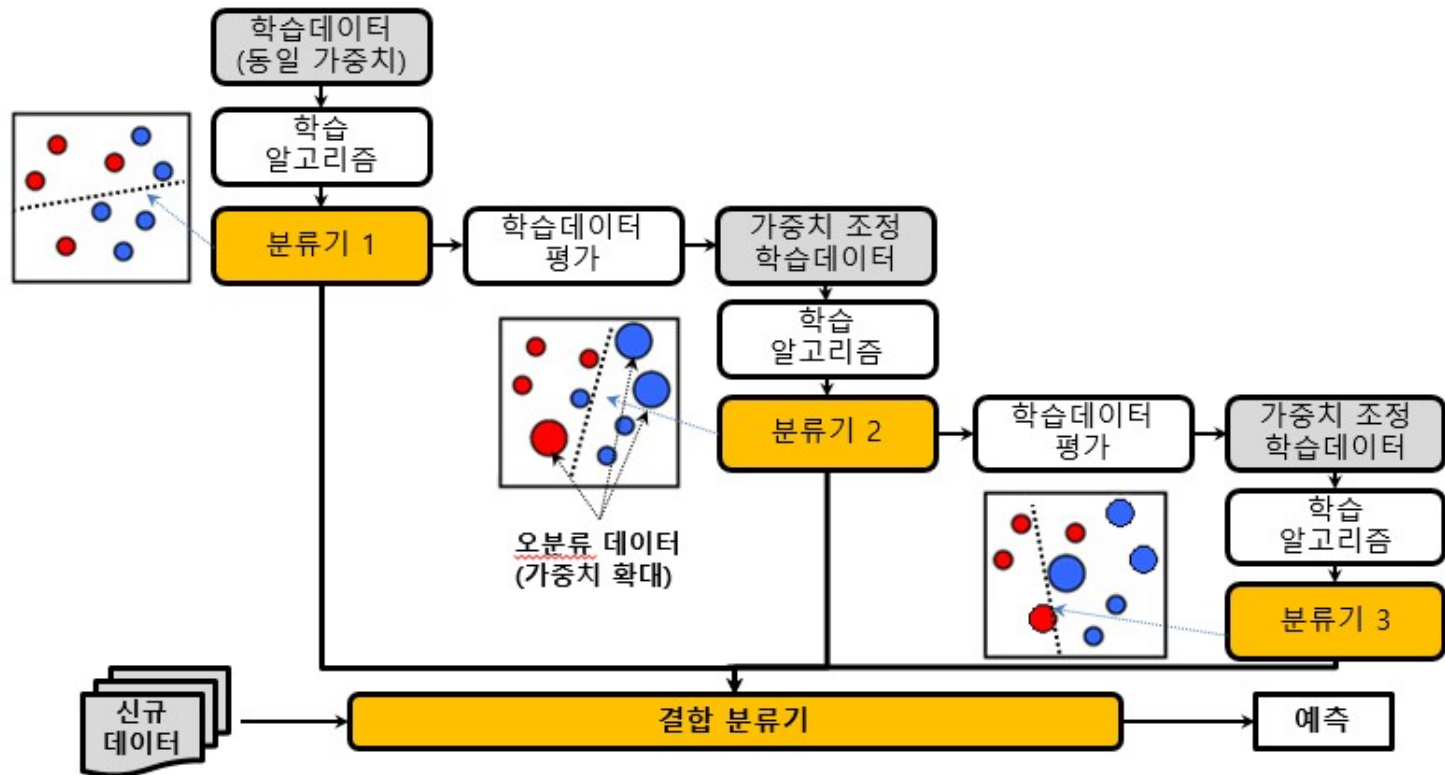
■ 랜덤 포리스트 (random forest) 알고리즘

- 분류기로 결정트리를 사용하는 배깅 기법

5.2 부스팅 알고리즘

❖ 부스팅(boosting)

- K 개의 분류기를 순차적으로 만들어 가는 앙상블 분류기 생성 방법
- 분류 정확도에 따라 학습 데이터에 가중치를 변경해가면서 분류기 생성



- 에이다부스트(AdaBoost)



부스팅 알고리즘

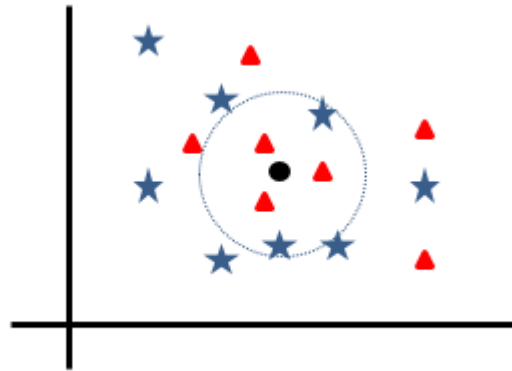
✧ 에이다부스트(AdaBoost)

- N 개의 학습 데이터 d_i 에 대한 초기 가중치 w_i
 - $w_i = \frac{1}{N}$, 가중치의 합 : 1
- 학습 오류값 ϵ
 - 잘못 분류한 학습데이터의 가중치의 합으로 표현
 - 값이 0.5미만인 분류기들만을 사용
- 학습
 - 오류값이 0.5미만인 분류기가 학습되는 경우
 - 분류기 신뢰도 : α
 - $\alpha = 0.5 \ln\left(\frac{1-\epsilon}{\epsilon}\right)$
 - 잘못 판정한 학습 데이터의 가중치는 증대
 - $w_i \leftarrow w_i e^{\alpha}$
 - 제대로 판정한 학습 데이터의 가중치는 축소
 - $w_i \leftarrow w_i e^{-\alpha}$
 - 가중치의 합이 1이 되도록 정규화

6. k-근접이웃 알고리즘

❖ k-근접이웃 (k-nearest neighbor, KNN) 알고리즘

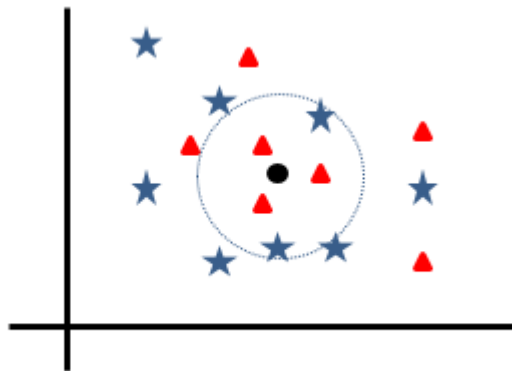
- (입력, 결과)가 있는 데이터들이 주어진 상황에서,
새로운 입력에 대한 결과를 추정할 때
결과를 아는 최근접한 k개의 데이터에 대한 결과정보를 이용하는 방법



- 질의(query)와 데이터간의 거리 계산
- 효율적으로 근접이웃 탐색
- 근접 이웃 k개로 부터 결과를 추정

k-근접이웃 알고리즘

❖ k-nearest neighbor (KNN) 알고리즘 - cont.



■ 데이터간의 거리 계산

• 수치 데이터의 경우

- 유클리디언 거리 (Euclidian distance)

$$X = (x_1, x_2, \dots, x_n) \quad Y = (y_1, y_2, \dots, y_n)$$

$$d = (x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2$$

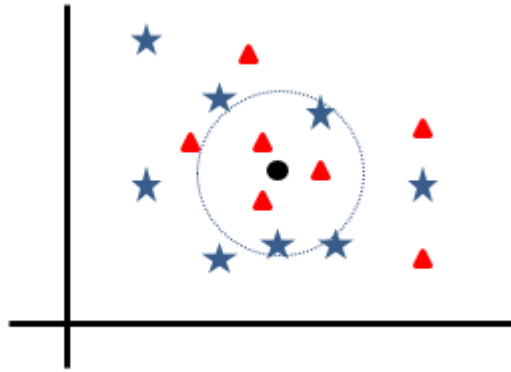
- 응용분야의 특성에 맞춰 개변할

• 범주형 데이터가 포함된 경우

- 응용분야의 특성에 맞춰 개변할

k-근접이웃 알고리즘

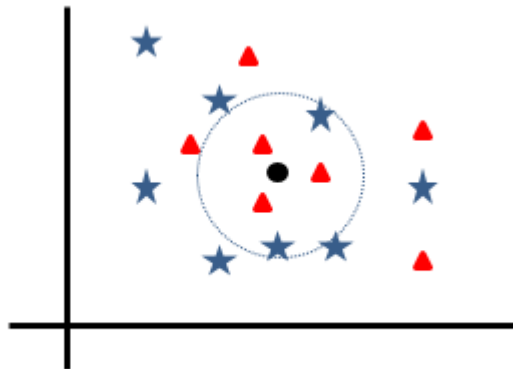
❖ k-nearest neighbor (KNN) 알고리즘 - cont.



- 효율적인 근접 이웃 탐색
 - 데이터의 개수가 많아지면 계산시간 증가 문제
 - 색인(indexing) 자료구조 사용
 - R-트리, k-d 트리 등

k-근접이웃 알고리즘

❖ k-nearest neighbor (KNN) 알고리즘 - cont.



■ 최근접 k개로 부터 결과를 추정하는 방법

- 분류

- 출력이 범주형 값
- 다수결 투표 (majority voting) : 개수가 많은 범주 선택

- 회귀분석

- 출력이 수치형 값
- 평균 : 최근접 k개의 평균값
- 가중합 (weighted sum) : 거리에 반비례하는 가중치 사용

k-근접이웃 알고리즘

❖ k-nearest neighbor (KNN) 알고리즘 - cont.

▪ 특징

- 학습단계에서는 실질적인 학습이 일어나지 않고 데이터만 저장
 - 학습데이터가 크면 메모리 문제
 - 게으른 학습(lazy learning)
- 새로운 데이터가 주어지면 저장된 데이터를 이용하여 학습
 - 시간이 많이 걸릴 수 있음



7. 군집화 알고리즘

❖ 군집화(clustering) 알고리즘

- 데이터를 유사한 것들끼리 묶는 것
- 군집 간의 유사도(similarity)는 크게, 군집 내의 유사도는 작게

❖ 계층적 군집화 (hierarchical clustering)

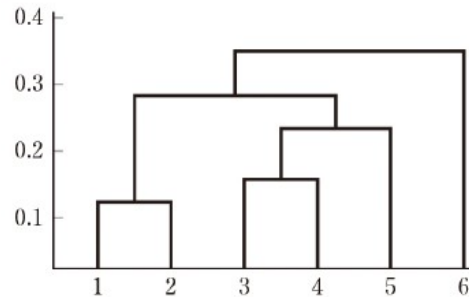
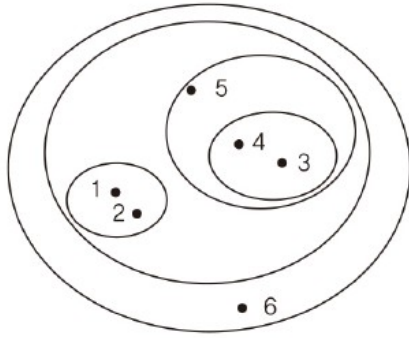
- 군집화의 결과가 군집들이 계층적인 구조를 갖도록 하는 것
- 병합형(agglomerative) 계층적 군집화
 - 각 데이터가 하나의 군집을 구성하는 상태에서 시작하여, 가까이에 있는 군집들을 결합하는 과정을 반복하여 계층적인 군집 형성
- 분리형(divisive) 계층적 군집화
 - 모든 데이터를 포함한 군집에서 시작하여 유사성을 바탕으로 군집을 분리하여 점차 계층적인 구조를 갖도록 구성

❖ 분할 군집화 (partitioning clustering)

- 계층적 구조를 만들지 않고 전체 데이터를 유사한 것들끼리 나누어서 묶는 것
- 예. k-means 알고리즘

군집화 알고리즘

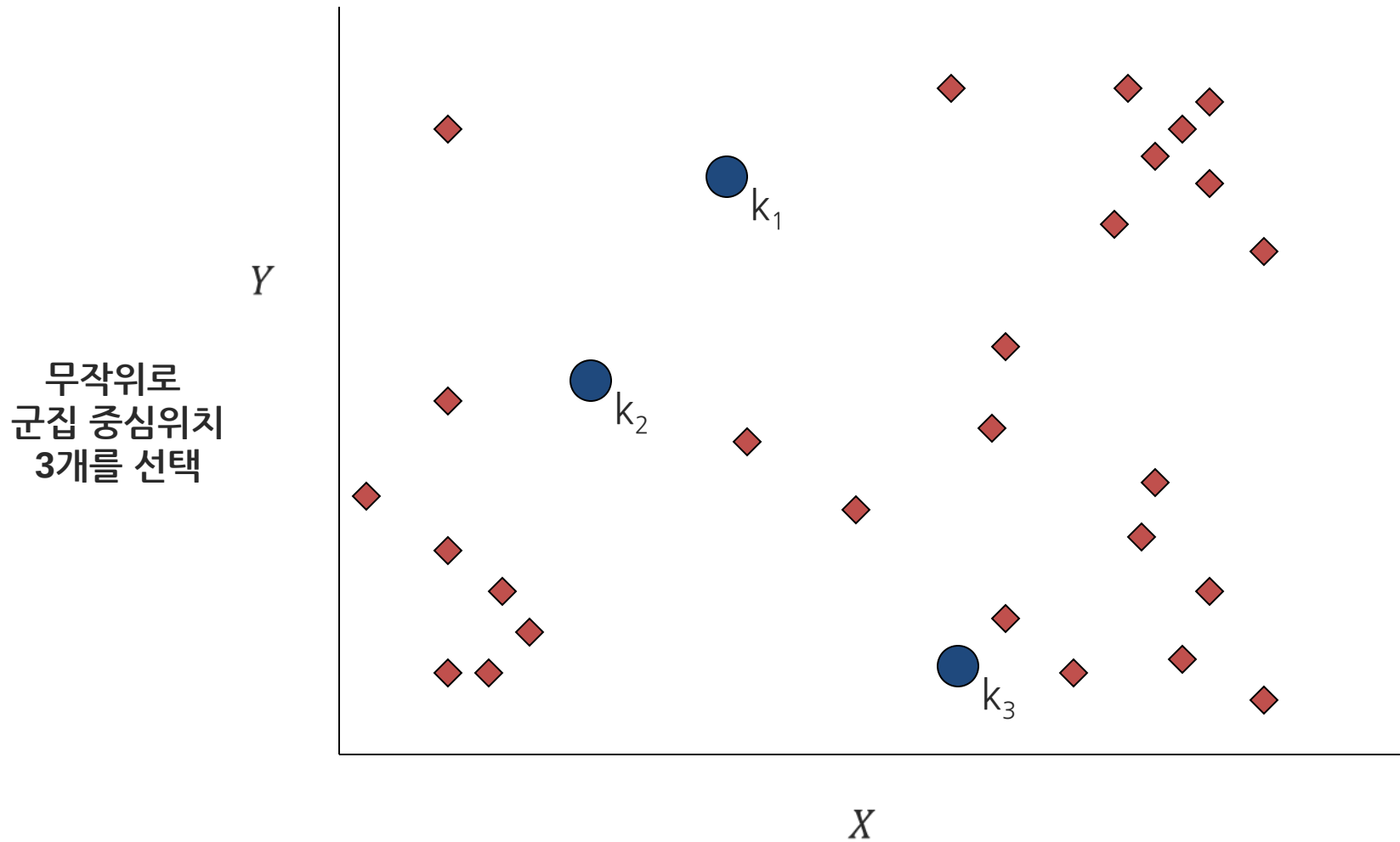
❖ 계층적 군집화와 덴드로그램(dendrogram)



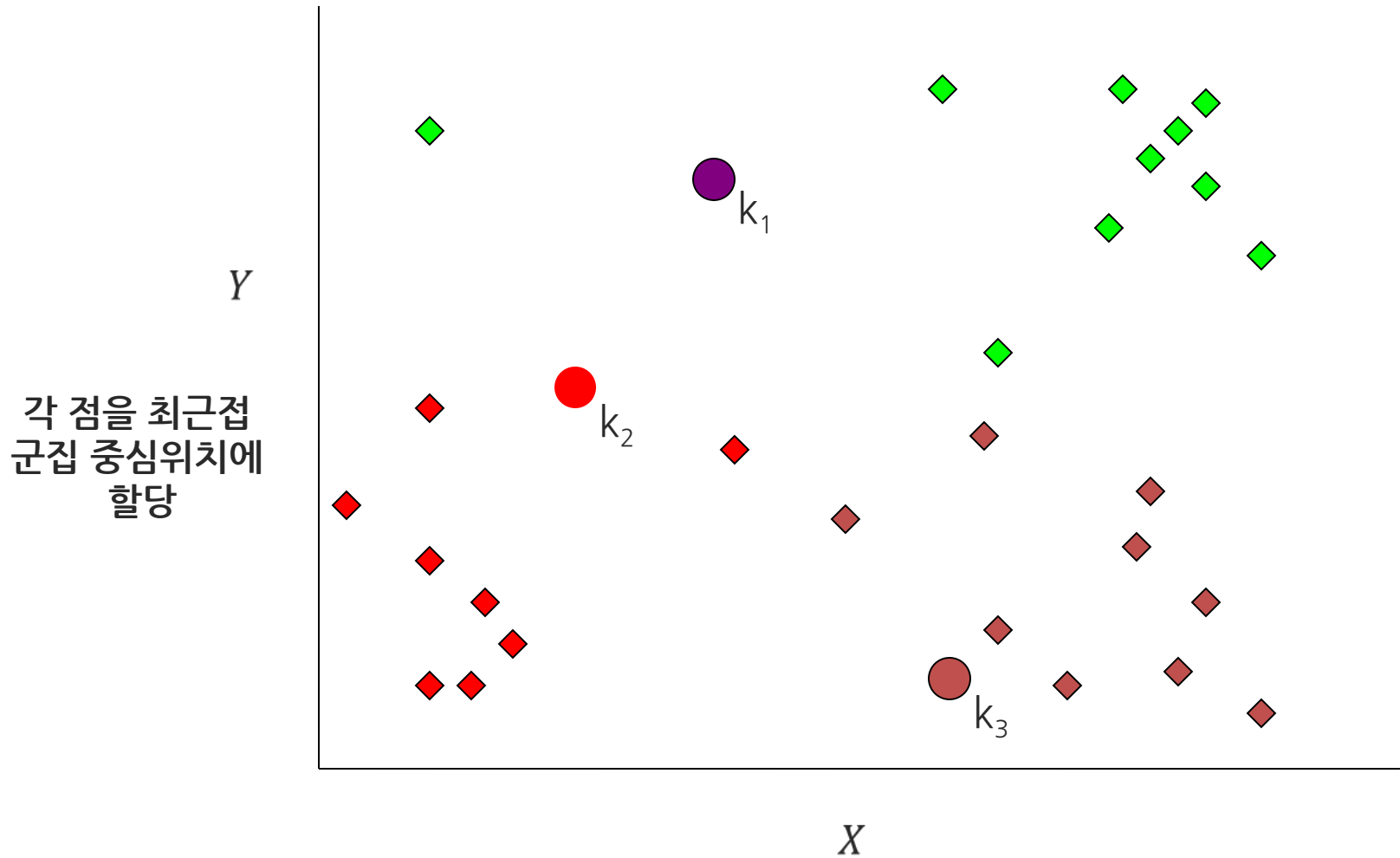
❖ k-means 알고리즘

- 군집화 알고리즘
- 군집화 과정
 1. 군집의 중심 위치 선정
 2. 군집 중심을 기준으로 군집 재구성
 3. 군집별 평균 위치 결정
 4. 군집 평균 위치로 군집 중심 조정
 5. 수렴할 때까지 2-4 과정 반복

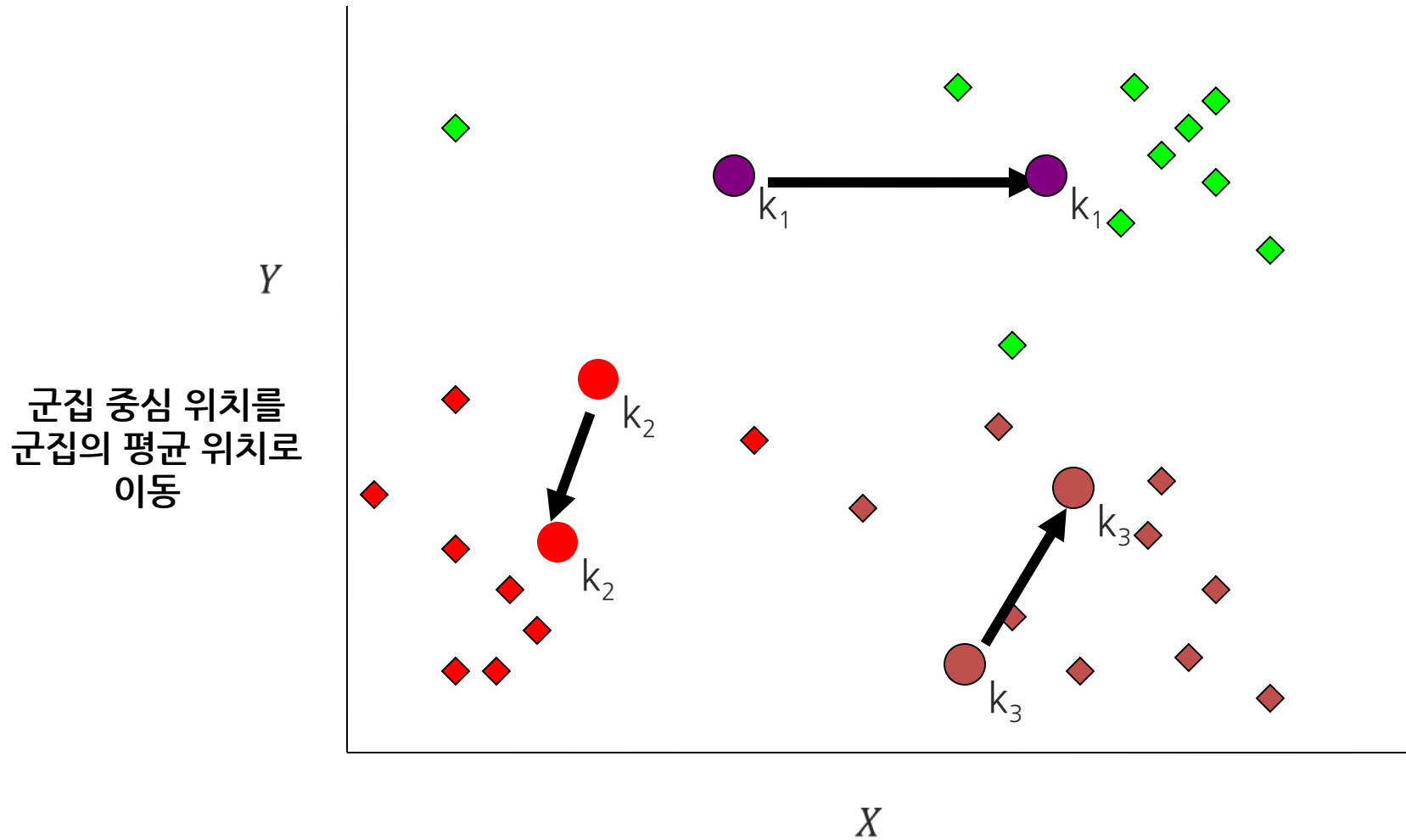
군집화 알고리즘 : K-means 실행과정 1



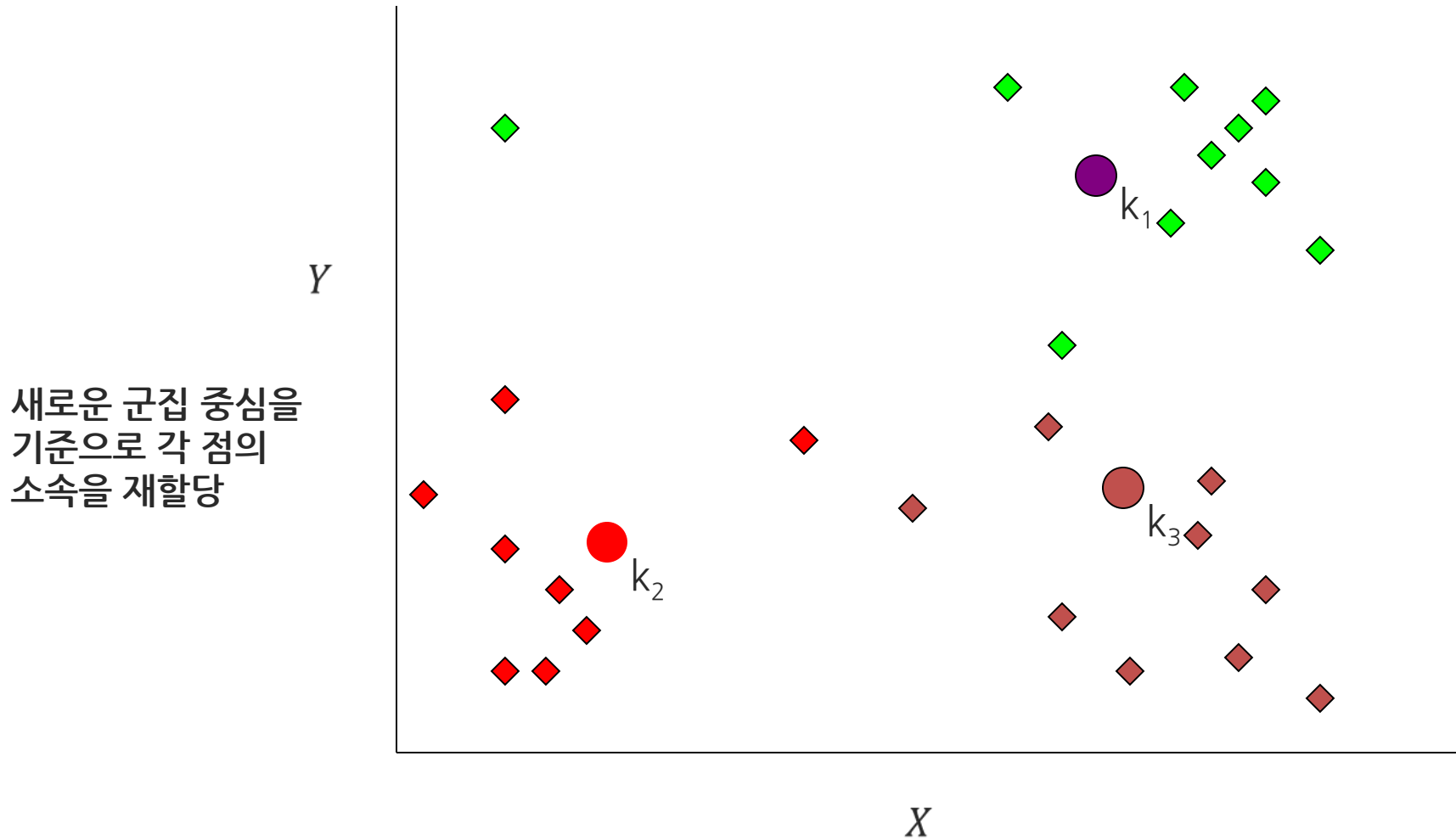
군집화 알고리즘 : K-means 실행과정 2



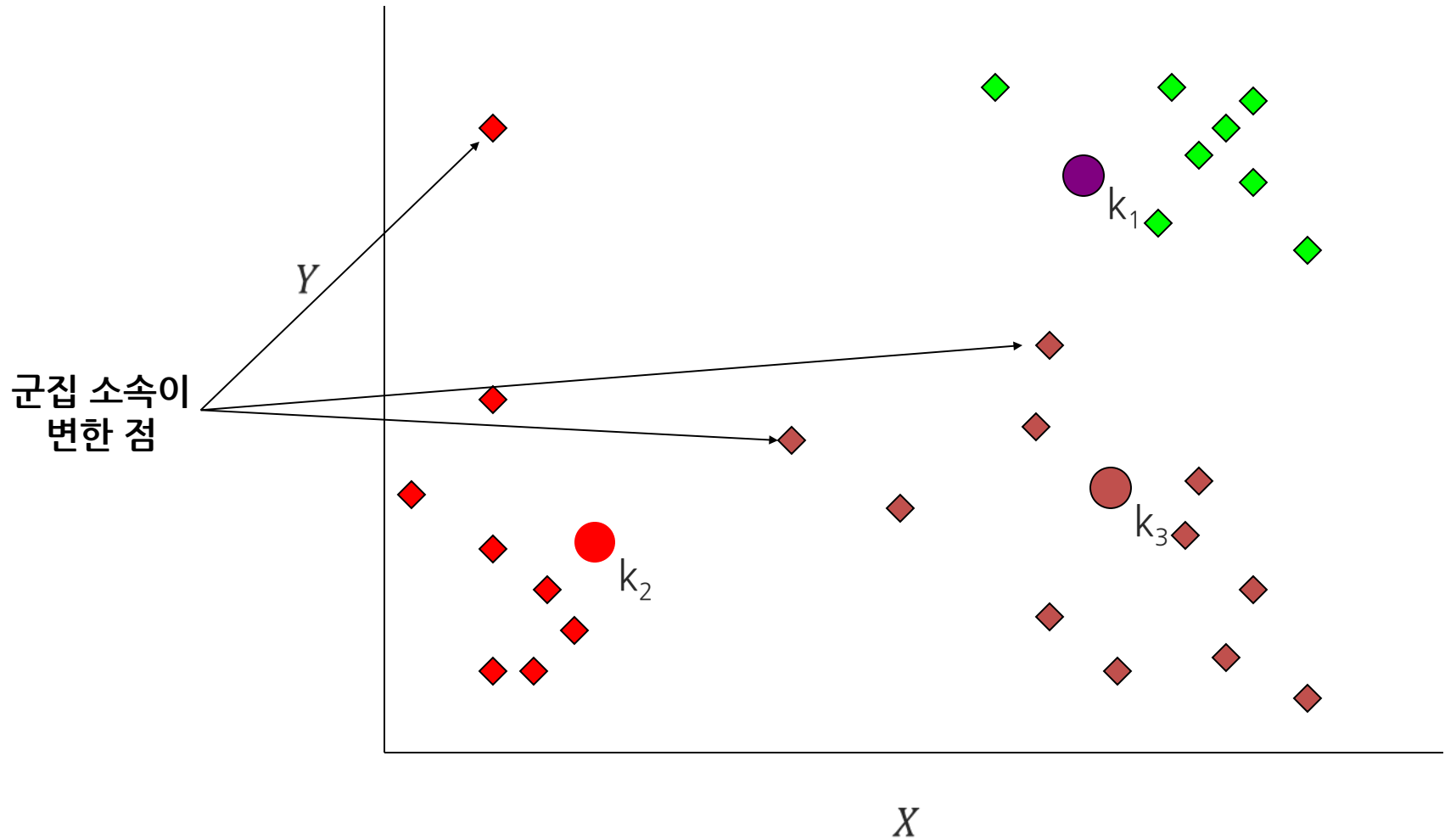
군집화 알고리즘 : K-means 실행과정 3



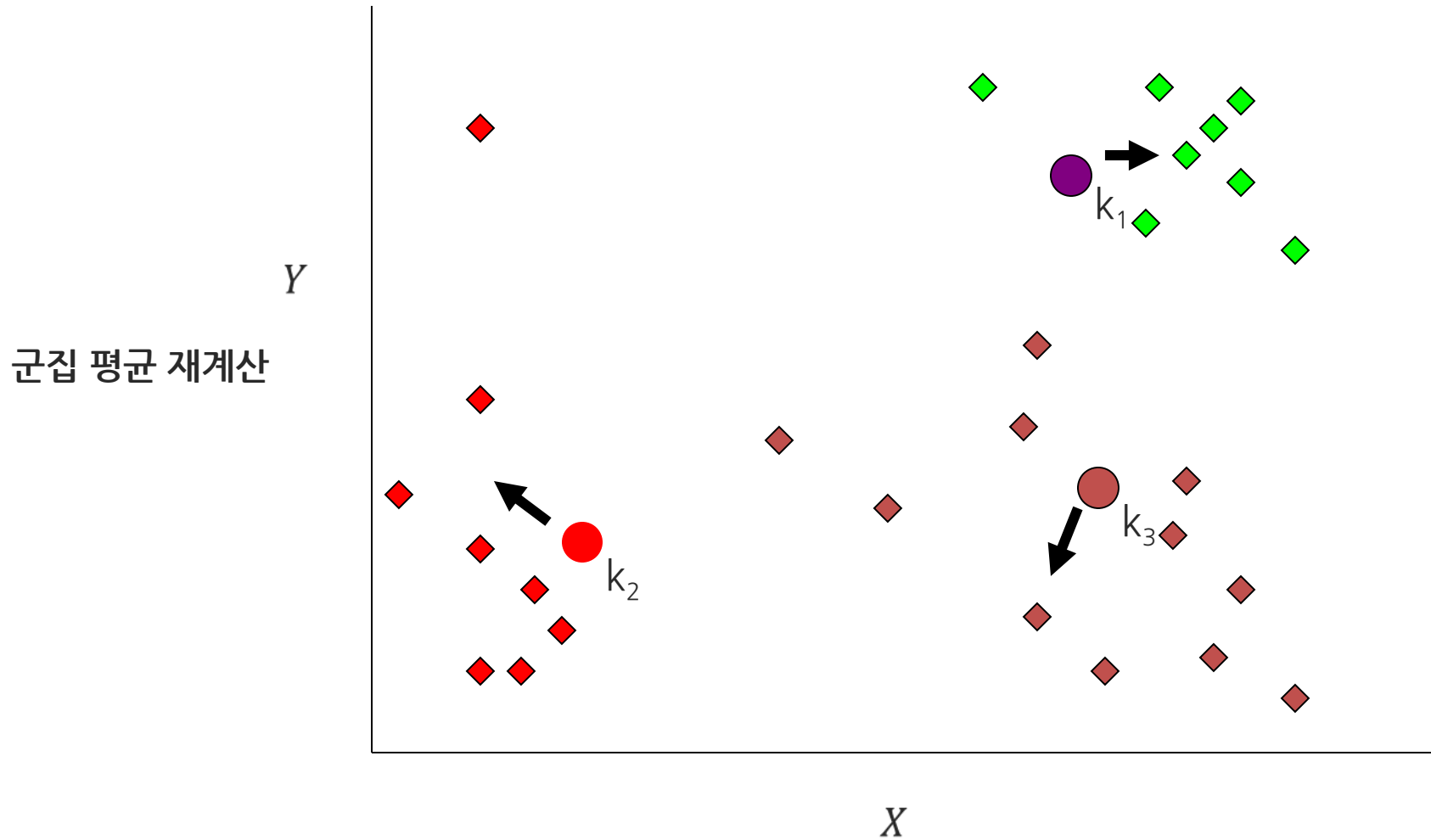
군집화 알고리즘 : K-means 실행과정 4



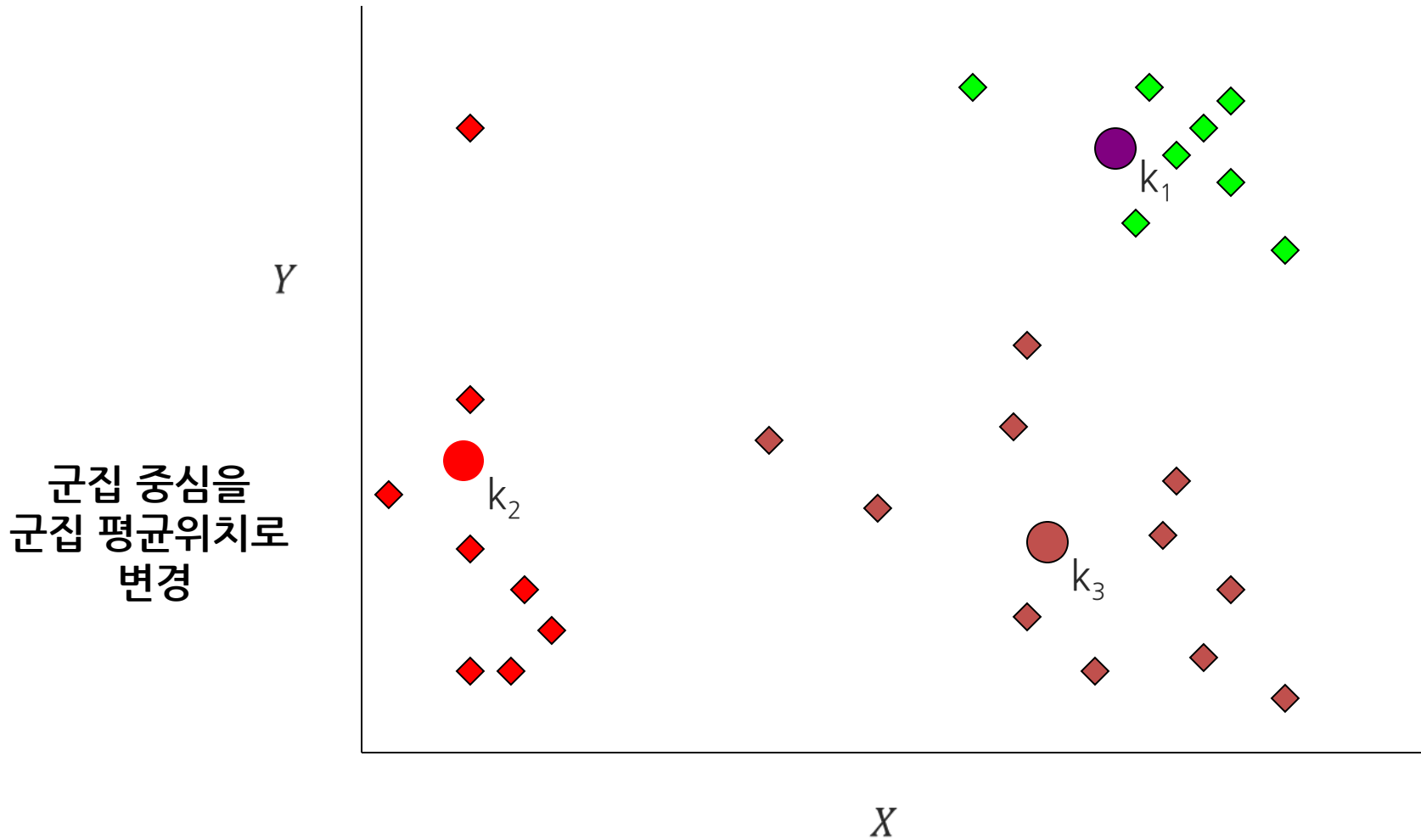
군집화 알고리즘 : K-means 실행과정 5



군집화 알고리즘 : K-means 실행과정 6



군집화 알고리즘 : K-means 실행과정 7



군집화 알고리즘 : K-means 알고리즘

❖ k-means 알고리즘

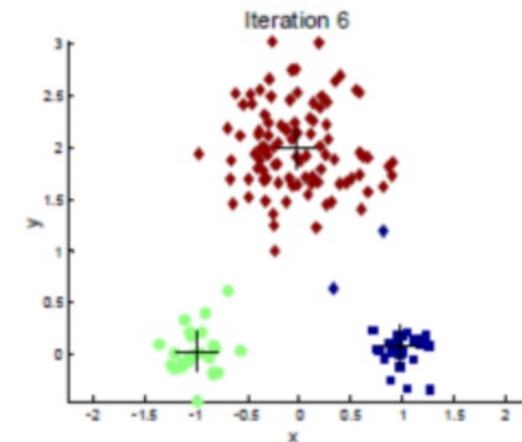
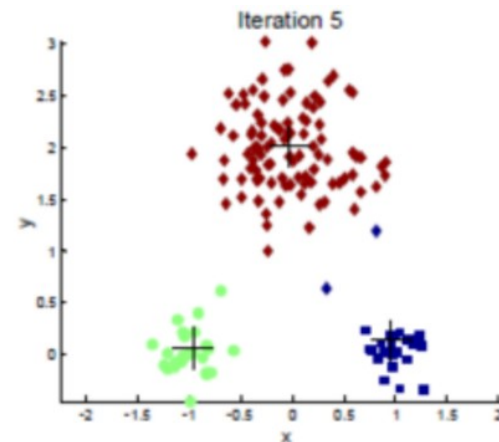
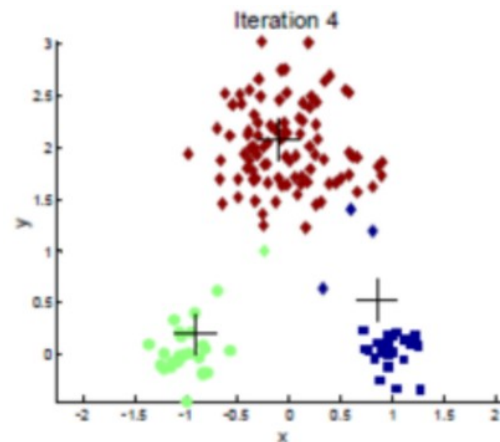
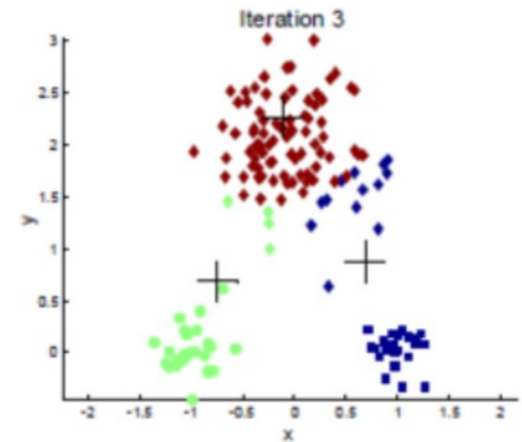
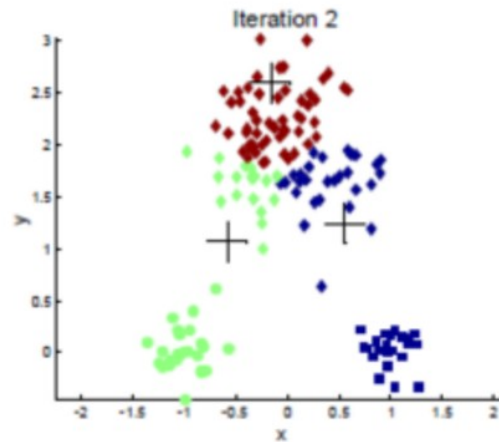
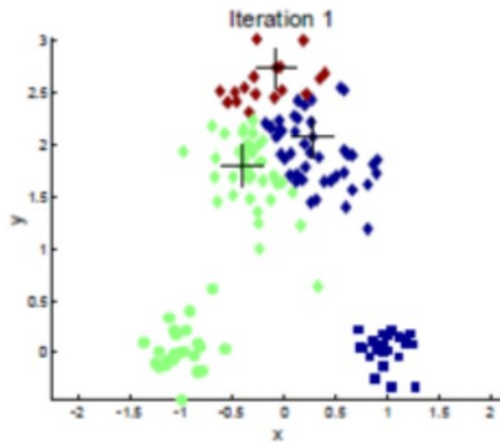
- i 번째 클러스터의 중심을 μ_i , 클러스터에 속하는 점의 집합 S_i 을 라고 할 때, 전체 분산

$$V = \sum_{i=1}^k \sum_{j \in S_i} |x_j - \mu_i|^2$$

- 분산값 V 을 최소화하는 S_i 를 찾는 것이 알고리즘의 목표
- 과정
 1. 우선 초기의 μ_i 를 임의로 설정
 2. 다음 두 단계를 클러스터가 변하지 않을 때까지 반복
 - I. 클러스터 설정: 각 점에 대해, 그 점에서 가장 가까운 클러스터를 찾아 배당한다.
 - II. 클러스터 중심 재조정: μ_i 를 각 클러스터에 있는 점들의 평균값으로 재설정해준다.
- 특성
 - 군집의 개수 k 는 미리 지정
 - 초기 군집 위치에 민감

군집화 알고리즘 : K-means 알고리즘

- 초기 중심값에 대해 민감한 군집화 결과



8. 단순 베이즈 분류기

❖ 단순 베이즈 분류기(naïve Bayes classifier)

- **부류(class)** 결정지식을 **조건부 확률(conditional probability)**로 결정
 - $P(c|x_1, x_2, \dots, x_n)$: 속성값에 대한 부류의 조건부 확률
 - c : 부류
 - x_i : 속성값
- **베이즈 정리 (Bayes theorem)**

$$P(c|x_1, x_2, \dots, x_n) = \frac{P(x_1, x_2, \dots, x_n|c)P(c)}{P(x_1, x_2, \dots, x_n)}$$

사후 확률 가능도 사전 확률

증거

- **가능도(likelihood)의 조건부 독립(conditional independence)** 가정

$$P(x_1, x_2, \dots, x_n|c) = P(x_1|c) P(x_2|c) \cdots P(x_n|c)$$

$$P(c|x_1, x_2, \dots, x_n) = \frac{P(x_1|c) P(x_2|c) \cdots P(x_n|c)P(c)}{P(x_1, x_2, \dots, x_n)}$$

단순 베이즈 분류기

❖ 단순 베이즈 분류기 - cont.

	속성			부류
	<i>Pattern</i>	<i>Outline</i>	<i>Dot</i>	<i>Shape</i>
1	수직	점선	무	삼각형
2	수직	점선	유	삼각형
3	대각선	점선	무	사각형
4	수평	점선	무	사각형
5	수평	실선	무	사각형
6	수평	실선	유	삼각형
7	수직	실선	무	사각형
8	수직	점선	무	삼각형
9	대각선	실선	유	사각형
10	수평	실선	무	사각형
11	수직	실선	유	사각형
12	대각선	점선	유	사각형
13	대각선	실선	무	사각형
14	수평	점선	유	삼각형

$$P(\text{삼각형}) = \frac{5}{14}$$

$$P(\text{사각형}) = \frac{9}{14}$$

$$P(\text{수직}|\text{삼각형}) = \frac{3}{5}$$

$$P(\text{수평}|\text{삼각형}) = \frac{2}{5}$$

$$P(\text{대각선}|\text{삼각형}) = \frac{1}{5}$$

$$P(\text{점선}|\text{삼각형}) = \frac{4}{5}$$

$$P(\text{실선}|\text{삼각형}) = \frac{1}{5}$$

$$P(\text{유}|\text{삼각형}) = \frac{3}{5}$$

$$P(\text{무}|\text{삼각형}) = \frac{2}{5}$$

$$P(\text{수직}, \text{점선}, \text{무}) = \frac{2}{14}$$

$$P(\text{삼각형}|\text{수직}, \text{점선}, \text{무})$$

$$= \frac{P(\text{수직}|\text{삼각형}) P(\text{점선}|\text{삼각형}) P(\text{무}|\text{삼각형}) P(\text{삼각형})}{P(\text{수직}, \text{점선}, \text{무})} = \frac{3/5 \cdot 4/5 \cdot 2/5 \cdot 5/14}{2/14} = 0.48$$