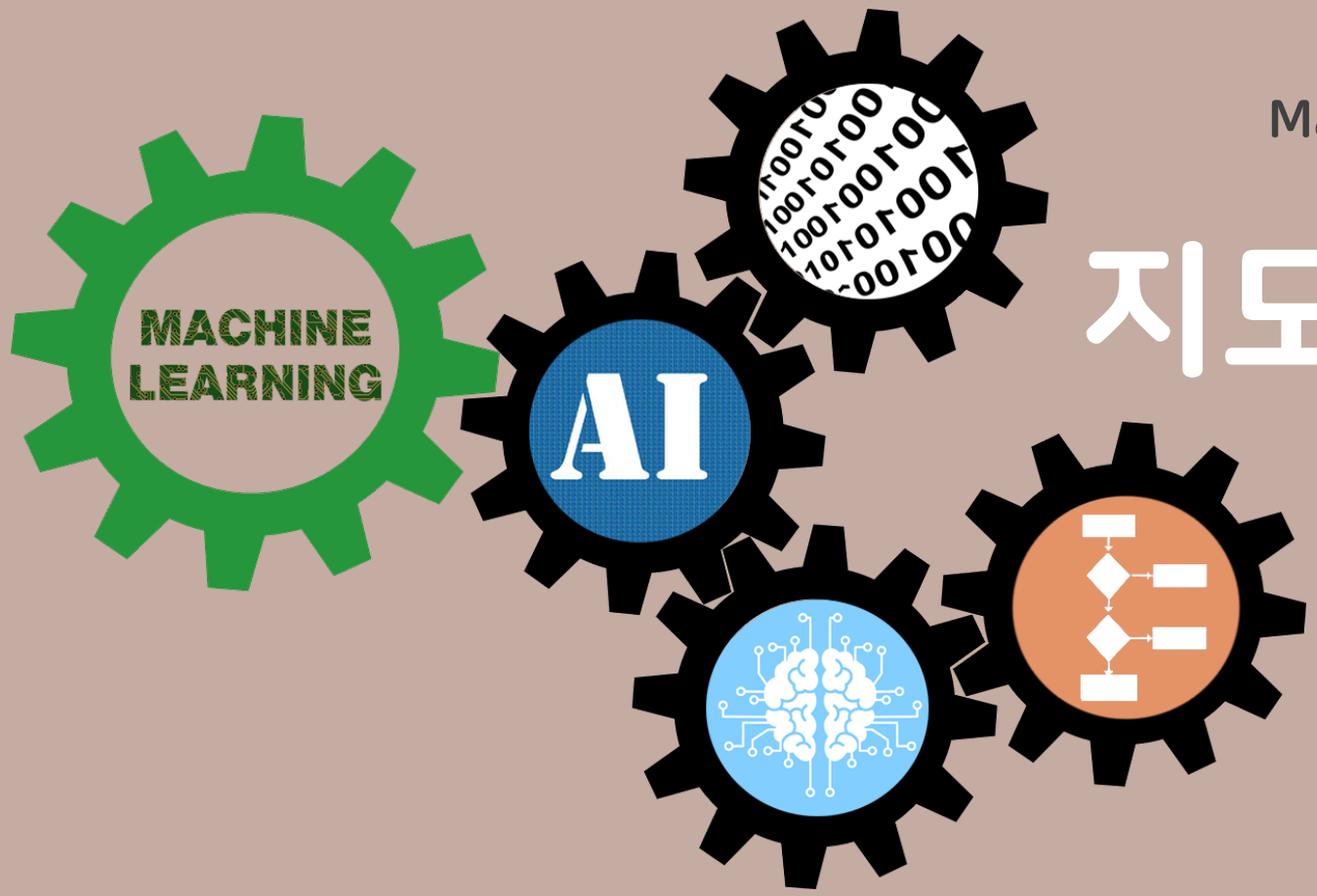
Machine Learning 2강

지도학습: 분류



컴퓨터과학과 이관용 교수

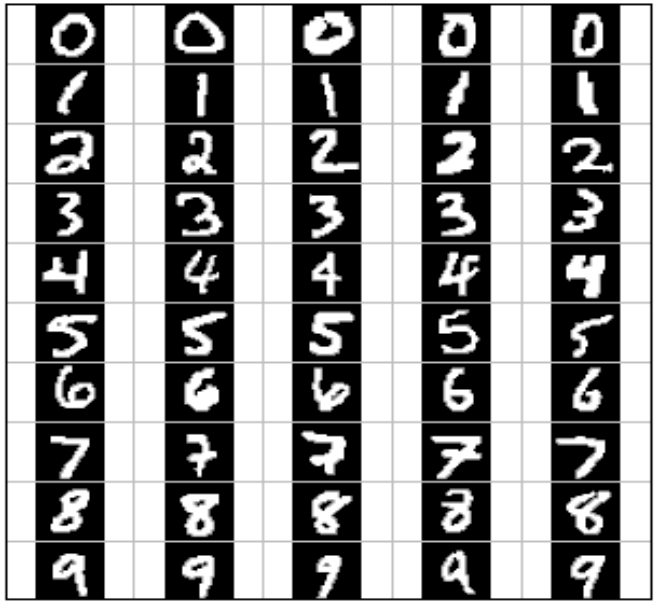
국립 한국방송통신대학교 Korea National Open University

학습목차

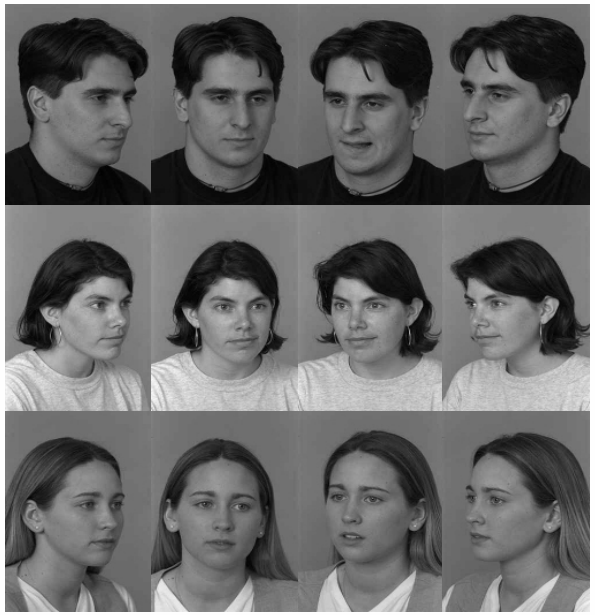
1 분류의 개념

데이터 분류

- 입력 데이터를 이미 정의된 몇 개의 클래스로 구분하는 문제
  - 예: 숫자인식, 얼굴인식 등



[MNIST database]



[FERET database]

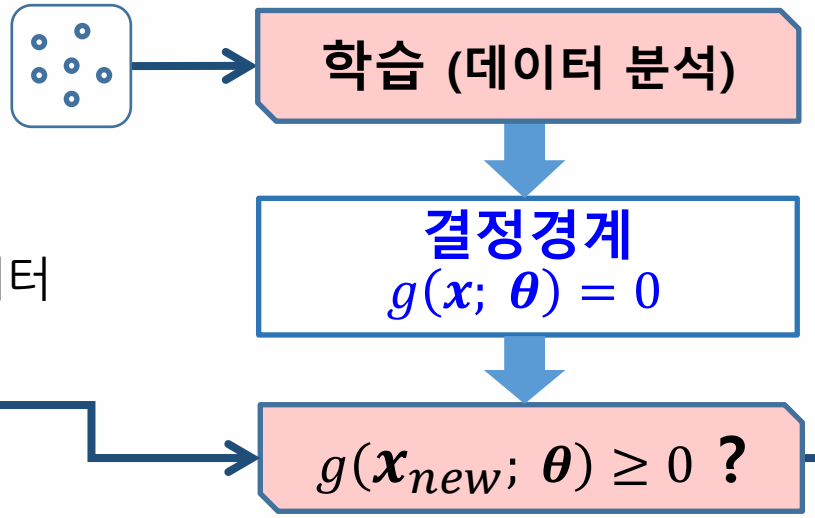
  - 베이즈 분류기, K-최근접이웃 분류기, 결정 트리, 랜덤 포레스트, SVM, 신경망(MLP, CNN, LSTM 등)

데이터 분류

분류기의 입·출력 관계

학습 데이터 집합

$D = \{(\boldsymbol{x}_i, y_i)\}_{i=1\ldots N}$



클래스 레이블
(이진 클래스의 경우)

$y_{new} = 0$ if $g(\boldsymbol{x}_{new}) \geq 0$

$y_{new} = 1$ if $g(\boldsymbol{x}_{new}) < 0$

학습 결과 → 결정경계와 결정함수

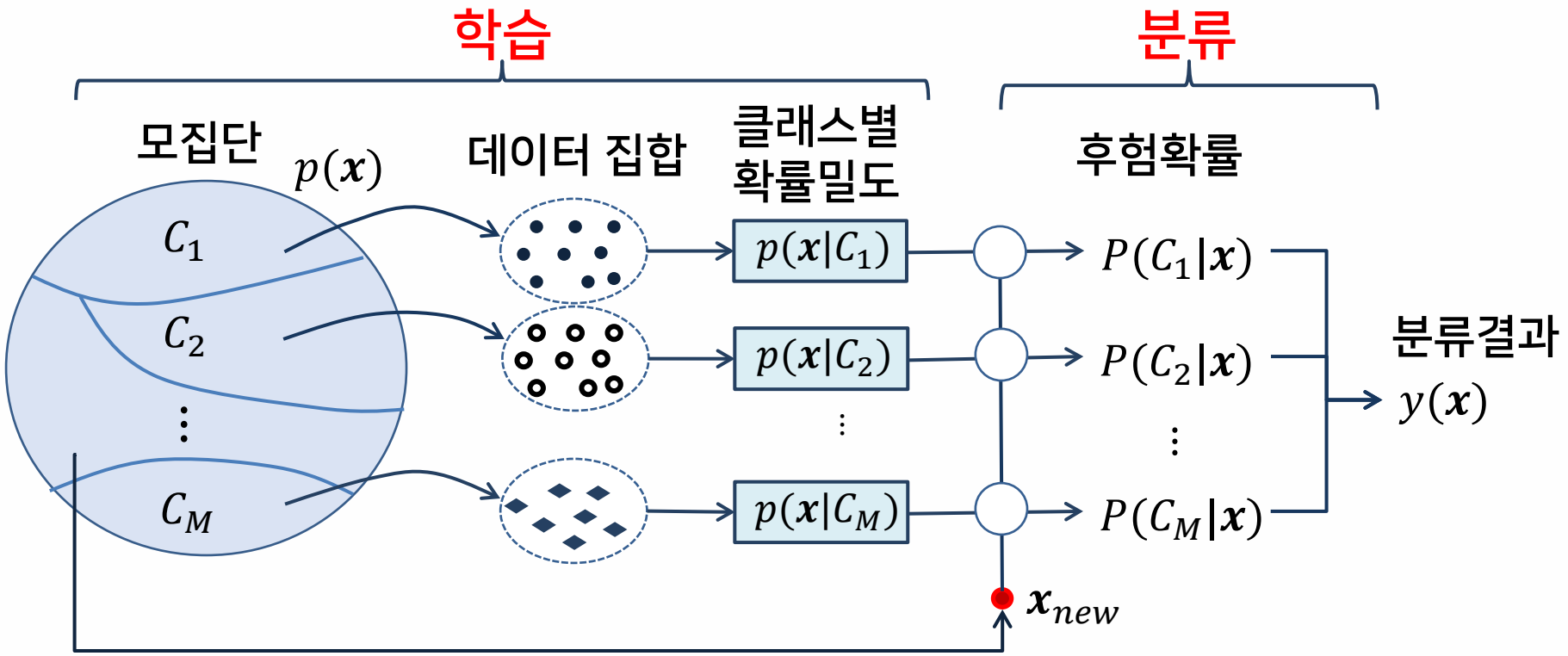
데이터 분류

- 결정경계 $g(\boldsymbol{x};\boldsymbol{\theta})$를 얻는 두 가지 접근법
  - 확률 기반 방법
    - ✓ $P(C_k|\boldsymbol{x})$를 추정하여 분류
    - ✓ 베이즈 분류기
  - 데이터 기반 방법
    - ✓ 데이터 간의 관계를 바탕으로 분류
    - ✓ K-최근접이웃 분류기

2 베이즈 분류기

확률분포에 기반한 분류의 개념

- $\boldsymbol{x}_{new} \rightarrow P(C_k|\boldsymbol{x}_{new}) \rightarrow \boldsymbol{x}_{new} \in C_i$



베이즈 정리를 이용한 결정경계

- 이진 분류 문제 → $\boldsymbol{x} \in C_1$? or $\boldsymbol{x} \in C_2$?

$\boldsymbol{x}$가 각 클래스에 속할 확률 $P(C_1|\boldsymbol{x})$, $P(C_2|\boldsymbol{x})$중 확률값이 큰 클래스로 할당

판별함수 $g(\boldsymbol{x}) = P(C_1|\boldsymbol{x}) - P(C_2|\boldsymbol{x})$

by 베이즈 정리 (p.63)

후험확률 (사후확률) $P(C_k|\boldsymbol{x})$ = 선험확률 (사전확률) $p(\boldsymbol{x}|C_k)p(C_k)$ / $p(\boldsymbol{x})$

$$g(\boldsymbol{x}) = \frac{p(\boldsymbol{x}|C_1)p(C_1)}{p(\boldsymbol{x})} - \frac{p(\boldsymbol{x}|C_2)p(C_2)}{p(\boldsymbol{x})}$$

베이즈 정리를 이용한 결정경계

결정경계

$$g(\boldsymbol{x}) = \frac{p(\boldsymbol{x}|C_1)p(C_1)}{p(\boldsymbol{x})} - \frac{p(\boldsymbol{x}|C_2)p(C_2)}{p(\boldsymbol{x})} = 0$$

분모 제거하고 각 항을 $p(\boldsymbol{x}|C_2)p(C_1)$로 나눈다.

$$g_{LRT}(\boldsymbol{x}) = \frac{p(\boldsymbol{x}|C_1)}{p(\boldsymbol{x}|C_2)} - \frac{p(C_2)}{p(C_1)} = 0$$

결정규칙

$$g_{LRT}(\boldsymbol{x}) = p(\boldsymbol{x}|C_1)p(C_1) - p(\boldsymbol{x}|C_2)p(C_2) > 0 \rightarrow \boldsymbol{x} \in C_1$$

$$g_{LRT}(\boldsymbol{x}) = p(\boldsymbol{x}|C_1)p(C_1) - p(\boldsymbol{x}|C_2)p(C_2) < 0 \rightarrow \boldsymbol{x} \in C_2$$

$$y(\boldsymbol{x}) = \begin{cases} 1 & \text{if } g_{LRT}(\boldsymbol{x}) > 0 \\ -1 & \text{otherwise} \end{cases}$$

베이즈 분류기의 결정경계

$$g_{LRT}(\boldsymbol{x}) = \frac{p(\boldsymbol{x}|C_1)}{p(\boldsymbol{x}|C_2)} - \frac{p(C_2)}{p(C_1)} = 0$$

우도비 likelihood ratio

(각 클래스에서 $\boldsymbol{x}$가 관찰될 확률밀도의 비율)

↓

우도비 분류

전체 데이터 집합에서
각 클래스가 차지하는 비율

베이즈 분류기 Bayes classifier

베이즈 정리로부터 유도된 결정경계를 이용한 분류

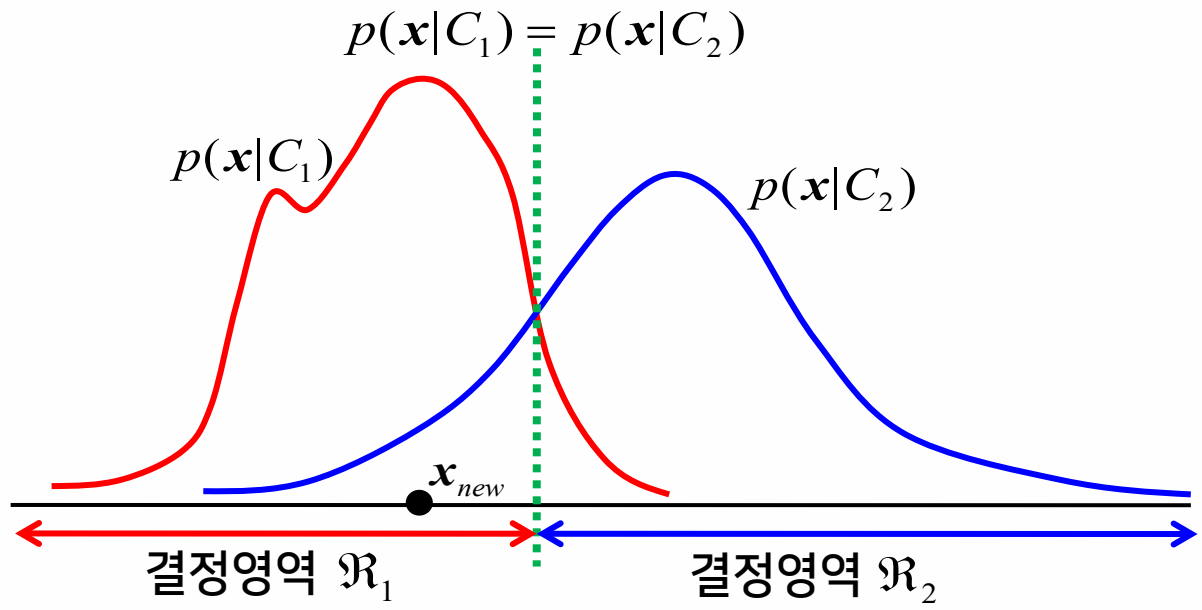
베이즈 분류기: 이진 클래스, $p(C_1) = p(C_2)$인 경우

$$y(\boldsymbol{x}) = \begin{cases} 1 & \text{if } p(\boldsymbol{x}|C_1) > p(\boldsymbol{x}|C_2) \rightarrow \boldsymbol{x} \in C_1 \\ -1 & \text{otherwise} \rightarrow \boldsymbol{x} \in C_2 \end{cases}$$

$$g_{LRT}(\boldsymbol{x}) = p(\boldsymbol{x}|C_1)p(C_1) - p(\boldsymbol{x}|C_2)p(C_2) > 0 \rightarrow \boldsymbol{x} \in C_1$$

$$g_{LRT}(\boldsymbol{x}) = p(\boldsymbol{x}|C_1)p(C_1) - p(\boldsymbol{x}|C_2)p(C_2) < 0 \rightarrow \boldsymbol{x} \in C_2$$

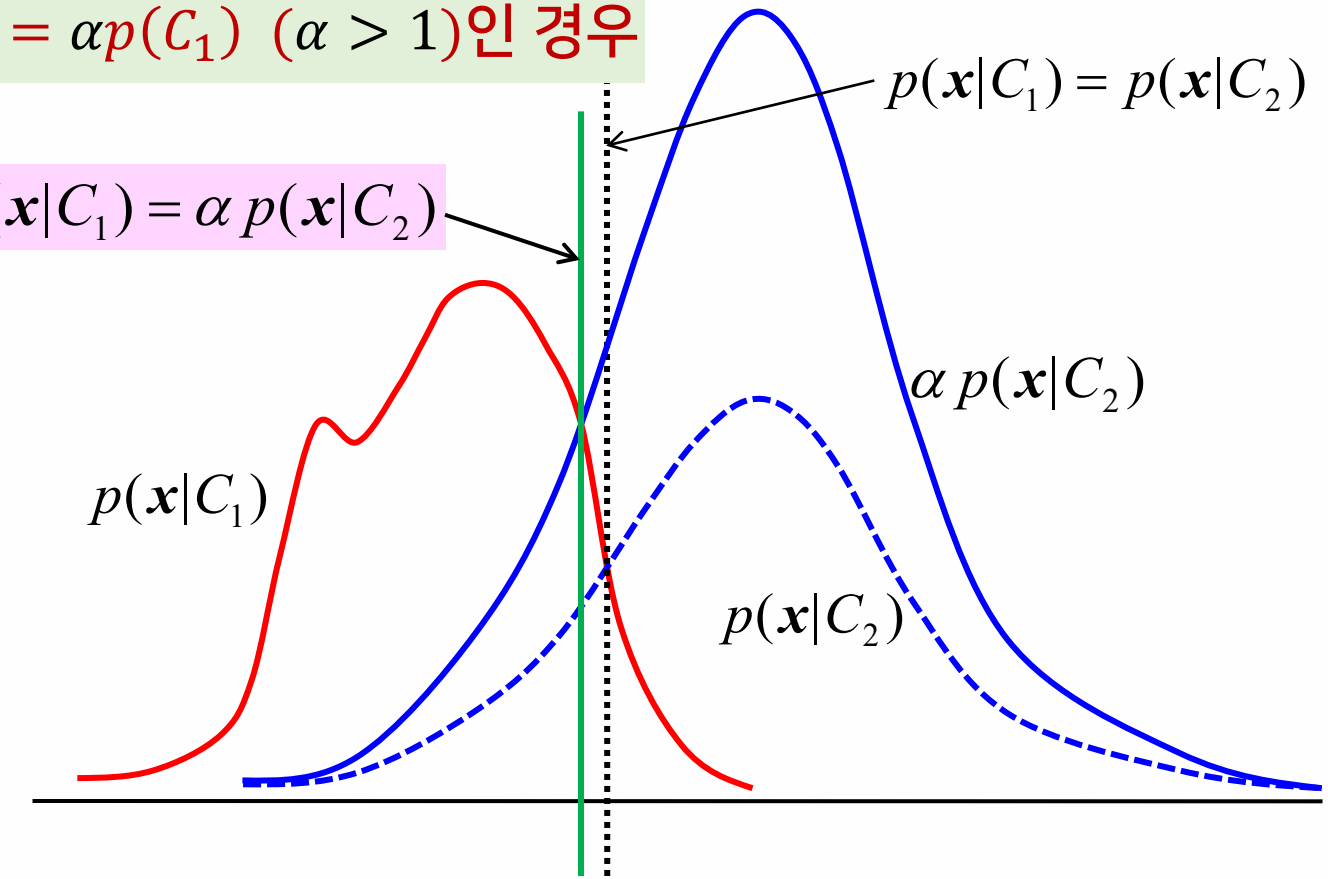
1차원 데이터에 대한 베이즈 분류기의 결정경계



베이즈 분류기: 이진 클래스, $p(C_1) \neq p(C_2)$인 경우

결정경계 → $p(\boldsymbol{x}|C_1)p(C_1) = p(\boldsymbol{x}|C_2)p(C_2)$

$p(C_2) = \alpha p(C_1)$ $(\alpha > 1)$인 경우



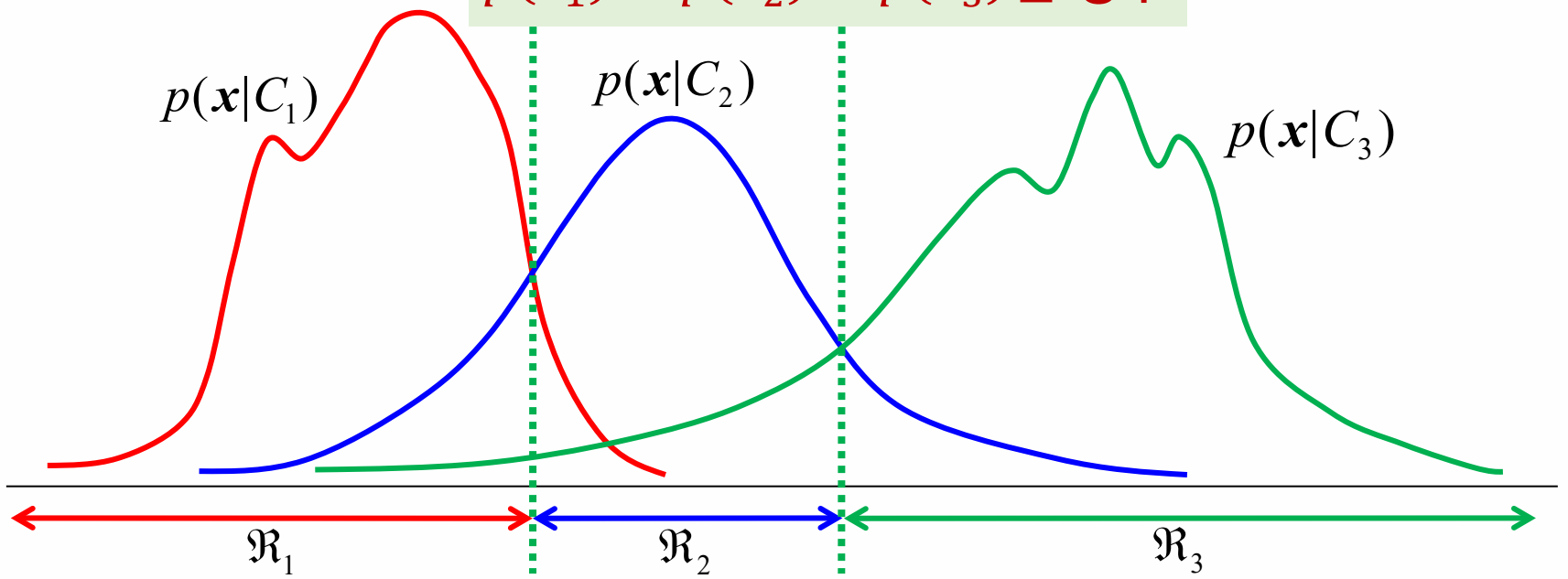
베이즈 분류기: 다중 클래스 문제

3개 클래스 분류기

각 클래스 C_i에 대한 판별함수 → $g_i(\boldsymbol{x}) = p(\boldsymbol{x}|C_i)p(C_i)$

클래스 레이블 $y(\boldsymbol{x})$의 결정규칙 → $y(\boldsymbol{x}) = \text{argmax}_i\{g_i(\boldsymbol{x})\}$

$p(C_1) = p(C_2) = p(C_3)$인 경우



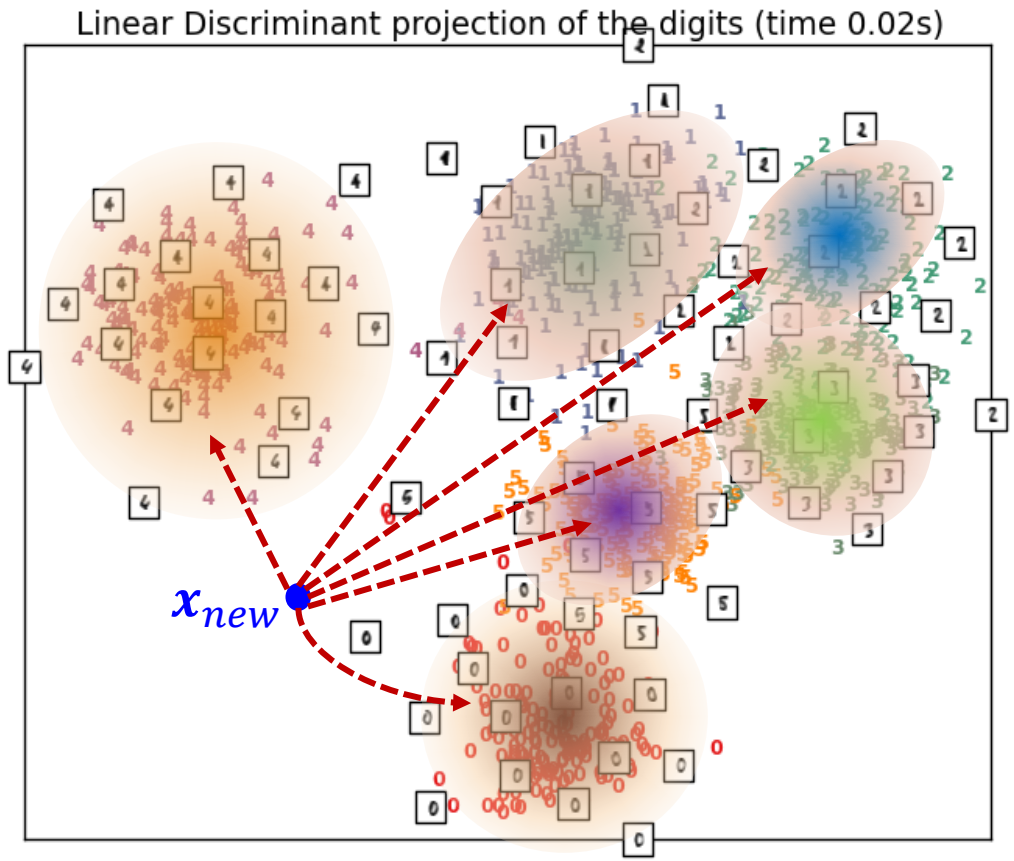
베이즈 분류기: 처리과정

분류 절차

- 학습 데이터 수집
- 학습 데이터로부터 클래스별 분포함수 추정

  $p(\boldsymbol{x}|C_k)$
- 테스트 데이터 $\boldsymbol{x}_{new}$ 입력
- 각 클래스별 판별함수 값 계산

  $g_k(\boldsymbol{x}_{new}) = p(\boldsymbol{x}|C_k)p(C_k)$
- $g_k(\boldsymbol{x}_{new})$가 가장 큰 클래스 k로 할당



베이즈 분류기의 구현

가우시안 확률분포를 가정한 베이즈 분류기

가우시안 분포의 확률밀도함수

$$p(\boldsymbol{x}|C_i) = G(\boldsymbol{x}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) = \frac{1}{\sqrt{(2\pi)^n}\sqrt{|\boldsymbol{\Sigma}_i|}} \exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)\right]$$

다중 클래스 분류를 위한 판별함수 (선험확률 $p(C_i)$가 모두 동일하다고 가정)

$$g_i(\boldsymbol{x}) = p(\boldsymbol{x}|C_i)p(C_i) \longrightarrow \ell_i(\boldsymbol{x}) = \ln g_i(\boldsymbol{x}) = \ln p(\boldsymbol{x}|C_i)$$

$$= -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_i| + const$$

결정규칙

$$y(\boldsymbol{x}) = \text{argmax}_i\{l_i(\boldsymbol{x})\}$$

$$= \text{argmin}_i\{(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i) + \ln|\boldsymbol{\Sigma}_i|\}$$

베이즈 분류기의 구현

공분산행렬의 형태에 따른 판별함수

- 클래스 공통 단위 공분산행렬 $\boldsymbol{\Sigma}_i = \sigma^2 \mathbf{I} \ (i = 1, \ \cdots, \ M)$
  - ✓ 모든 클래스의 공분산이 동일하게 단위행렬의 상수배인 행렬을 가지는 경우

- 클래스 공통 공분산행렬 $\boldsymbol{\Sigma}_i = \boldsymbol{\Sigma}$
  - ✓ 모든 클래스가 동일한 공분산을 갖지만 그 형태가 일반적인 행렬이 되는 경우

- 일반적인 공분산 행렬 $\boldsymbol{\Sigma}_i \neq \boldsymbol{\Sigma}_j$
  - ✓ 각 클래스의 공분산이 서로 다른 일반적인 형태를 가지는 경우

판별함수 형태: 클래스 공통 단위 공분산 행렬

$$\ell_i(\boldsymbol{x}) = \ln g_i(\boldsymbol{x}) = \ln p(\boldsymbol{x}|C_i)$$

$$= -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_i| + const$$

$$\boldsymbol{\Sigma}_i = \sigma^2\mathbf{I}\ (i = 1, \cdots, M)$$

판별함수 $$\ell_i(\boldsymbol{x}) = -\frac{1}{2\sigma^2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T(\boldsymbol{x}-\boldsymbol{\mu}_i) - n\ln\sigma + const$$

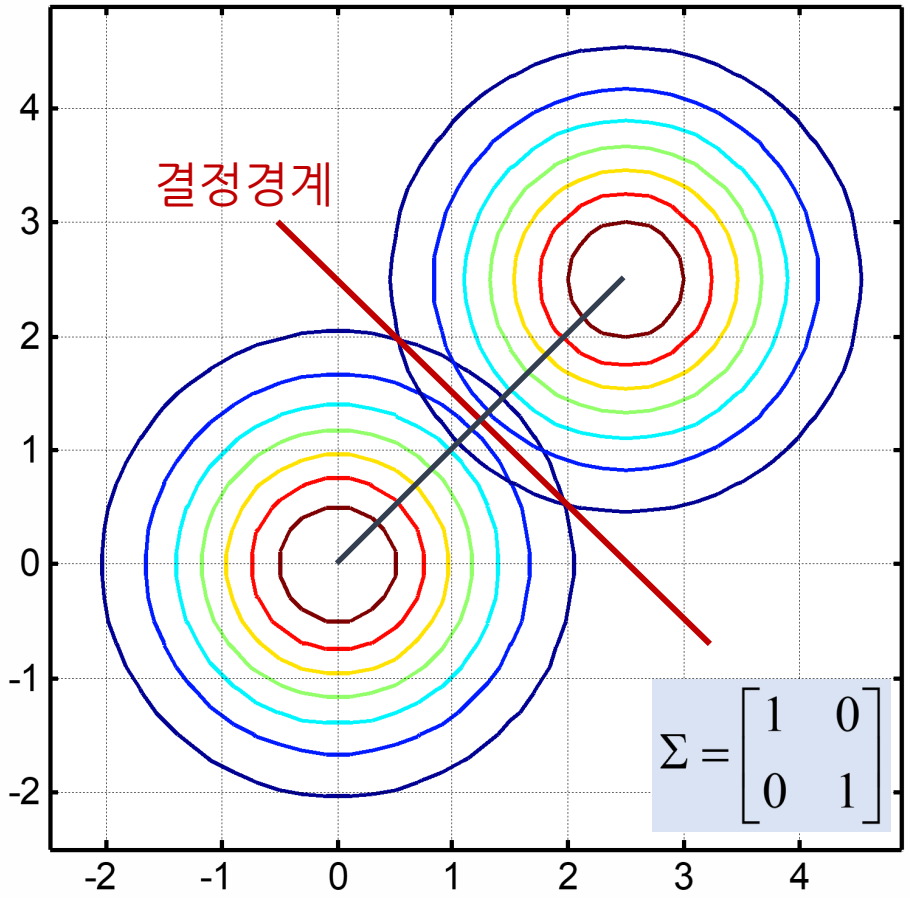
n과 σ는 공통

결정규칙 $$y(\boldsymbol{x}) = \text{argmin}_i\left\{(\boldsymbol{x}-\boldsymbol{\mu}_i)^T(\boldsymbol{x}-\boldsymbol{\mu}_i)\right\}$$

"최소거리 분류기"

minimum distance classifier

판별함수 형태: 클래스 공통 단위 공분산 행렬



판별함수 형태: 클래스 공통 공분산 행렬

$\Sigma_i = \Sigma$ (타원형 형태의 데이터 분포)

판별함수 $$\ell_i(\boldsymbol{x}) = -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)$$

결정규칙 $$y(\boldsymbol{x}) = \text{argmin}_i\left\{(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)\right\}$$

↓

"마할라노비스 거리"

Mahalanobis distance

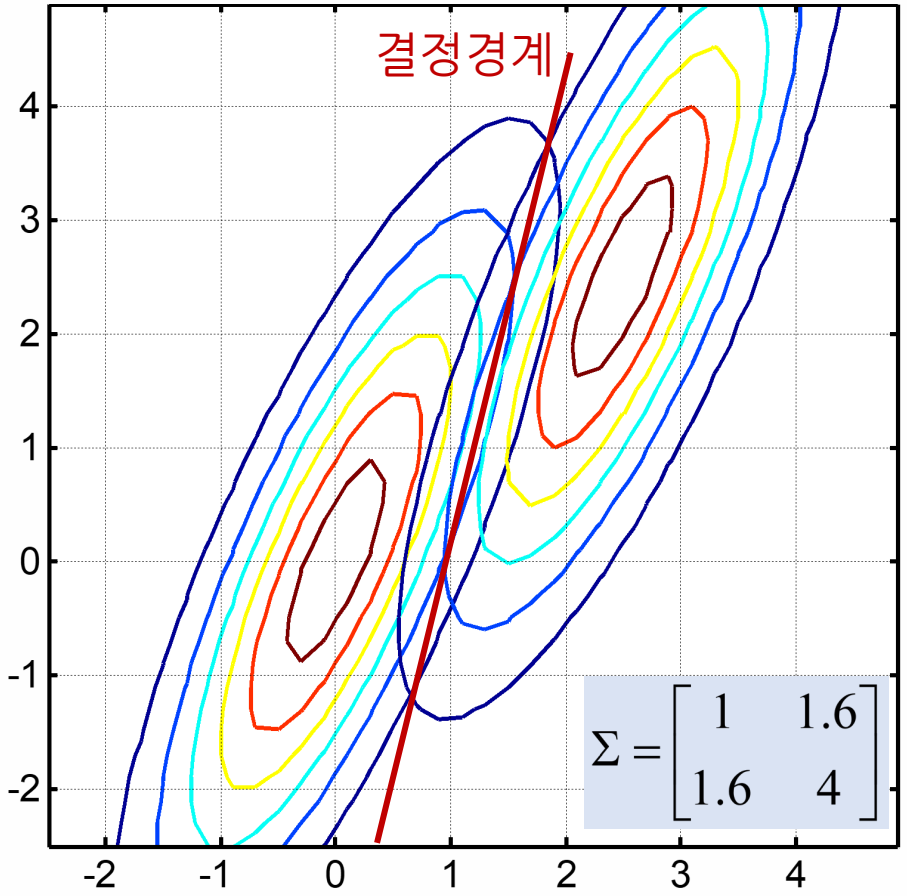
공분산 Σ가 대각행렬이면 ↓

"정규화된 유클리디안 거리"

normalized Euclidean distance

요소별로 표준편차 값으로 나누어 준 후 유클리디안 거리를 계산

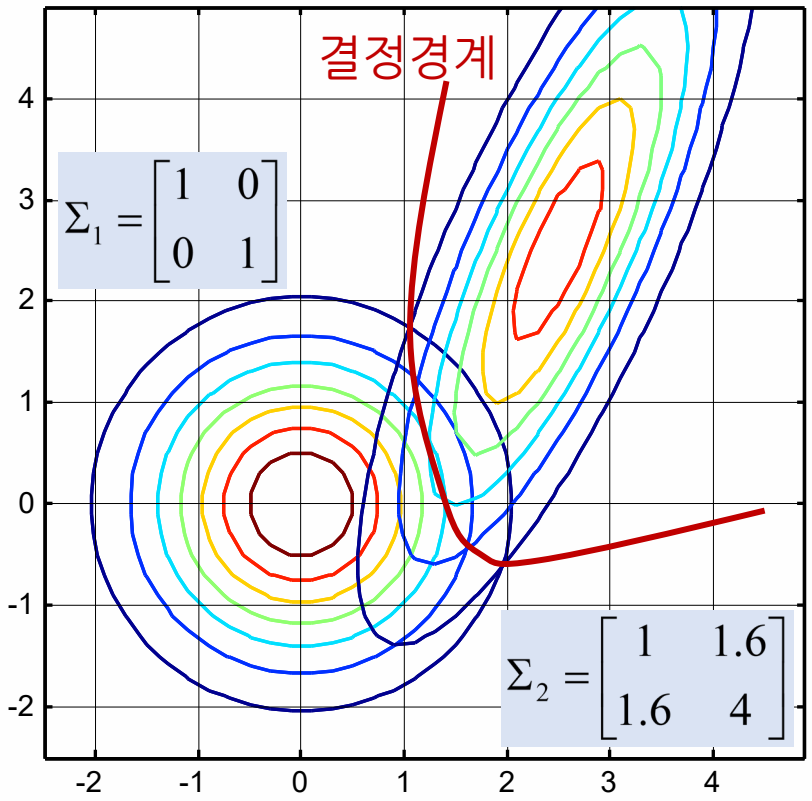
판별함수 형태: 클래스 공통 공분산 행렬



판별함수 형태: 일반적인 공분산 행렬

$\Sigma_i \neq \Sigma_j$ (서로 다른 타원형 분포)

결정규칙 $y(\boldsymbol{x}) = \text{argmin}_i \left\{ (\boldsymbol{x}-\boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\boldsymbol{x}-\boldsymbol{\mu}_i) + \ln \left| \boldsymbol{\Sigma}_i \right| \right\}$



베이즈 분류기: 가우시안 모델의 경우

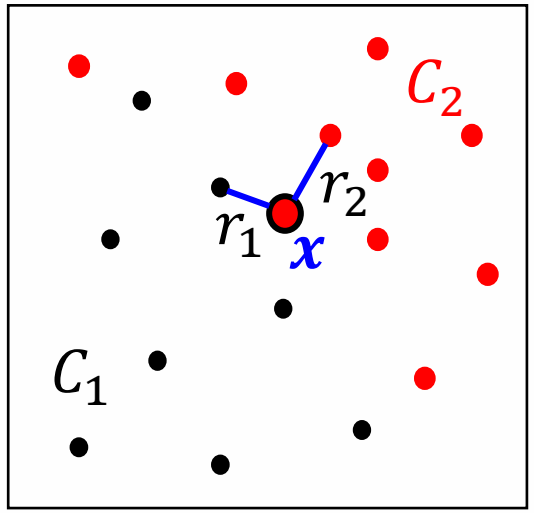
- 데이터 $\boldsymbol{x}$에 대한 클래스별 밀도함수
  - $p(\boldsymbol{x}|C_k) \propto \frac{1}{\det(\boldsymbol{\Sigma}_k)} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)\right\}$
  - 각 클래스의 평균과 공분산행렬을 각각 추정해야 함
- 간소화 방법
  - 공분산행렬 $\boldsymbol{\Sigma}_k$가 모두 단위행렬이라고 가정
    - ✓ $\boldsymbol{x}$와 평균 $\boldsymbol{\mu}_k$와의 거리를 비교하여 가까운 쪽의 클래스로 할당
  - 공분산행렬 $\boldsymbol{\Sigma}_k$가 모두 동일하다고 가정 → 하나의 $\boldsymbol{\Sigma}$만 추정
    - ✓ 평균과의 거리 계산에 활용 → 마할라노비스 거리
  - 일반적으로 공분산행렬이 동일하다고 볼 수 없으나, 계산이 간단하여 널리 사용

3

K-최근접이웃 분류기

K-최근접이웃 분류기 (K=1인 경우)

- "최근접이웃 분류기"



$$y(\boldsymbol{x}) = \text{argmin}\{r_1, r_2\} = 1$$

클래스와 상관없이 모든 데이터 중에서

가장 작은 거리값을 갖는 데이터의 클래스로 할당

$$\boldsymbol{x}_{min} = \text{argmin}_{\boldsymbol{x}_i \in X}\{d(\boldsymbol{x}, \boldsymbol{x}_i)\}$$

$$y(\boldsymbol{x}) = y(\boldsymbol{x}_{min})$$

"최근접이웃 분류기"

nearest neighbor classifier

최근접이웃 분류기

수행 단계

1. 주어진 데이터 $\boldsymbol{x}$와 모든 학습 데이터 $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\}$과의 거리를 계산한다.

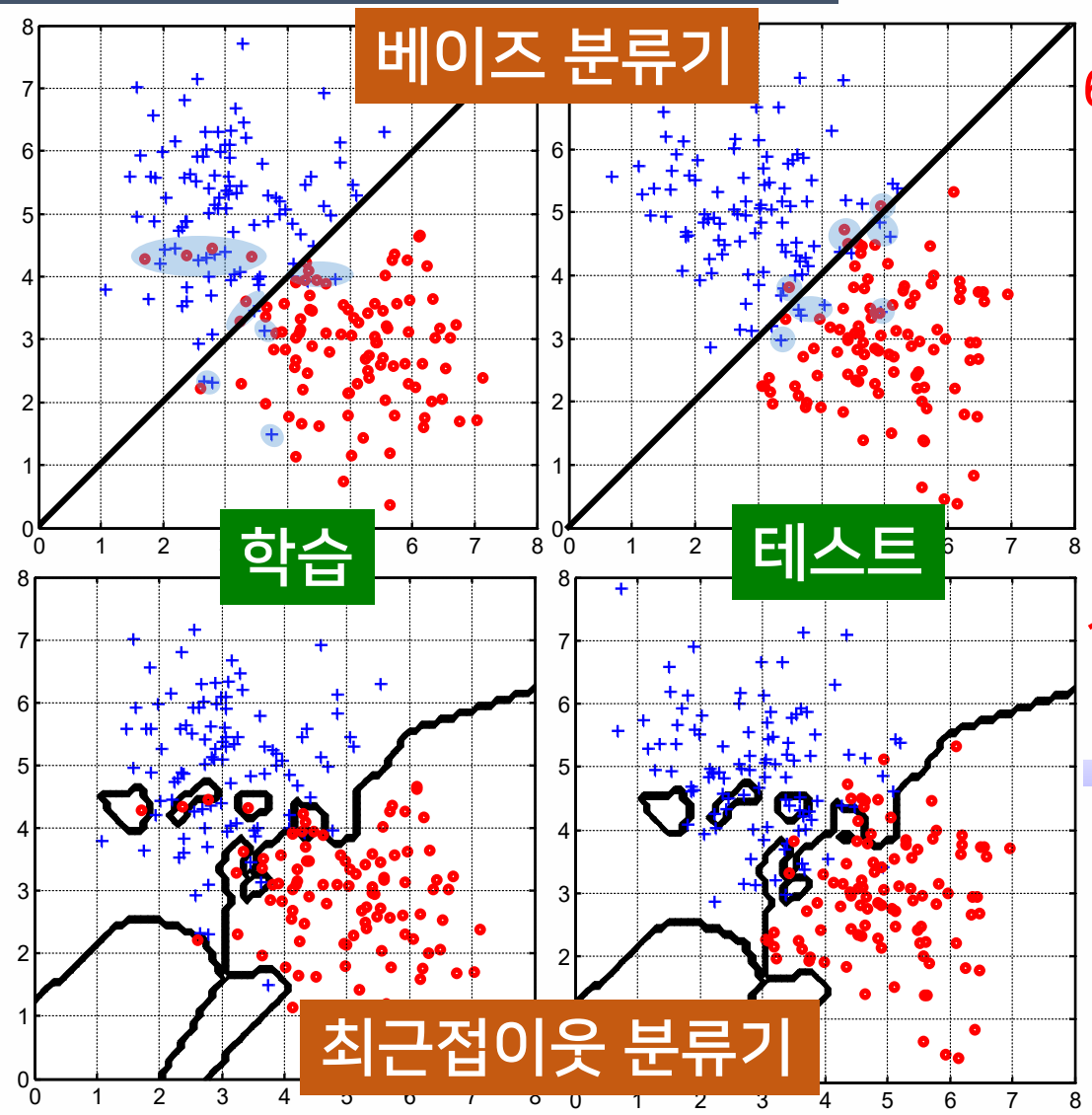
2. 거리가 가장 가까운 데이터를 찾아 $\boldsymbol{x}_{min}$으로 둔다.

$$\boldsymbol{x}_{min} = \text{argmin}_{\boldsymbol{x}_i \in X}\{d(\boldsymbol{x}, \boldsymbol{x}_i)\}$$

3. $\boldsymbol{x}_{min}$이 속하는 클래스에 할당한다.
   즉, $y(\boldsymbol{x}_{min})$과 같은 값을 가지도록 $y(\boldsymbol{x})$를 결정한다.

최근접이웃 분류기의 문제점

- 과다적합



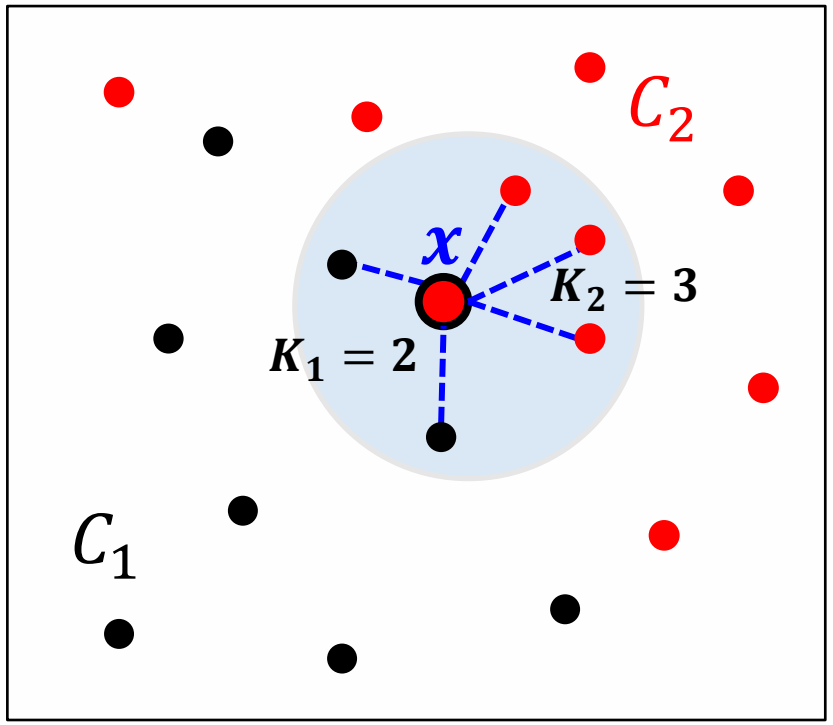
6.5%

13.5%

"K-최근접이웃 분류기"

K-최근접이웃 분류기

- K=5인 경우



$$y(\boldsymbol{x}) = \text{argmax}\{K_1(\boldsymbol{x}), K_2(\boldsymbol{x})\} \mapsto C_2$$

K-최근접이웃 분류기

수행 단계

1. 주어진 데이터 $\boldsymbol{x}$와 모든 학습 데이터 $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\}$과의 거리를 계산한다.

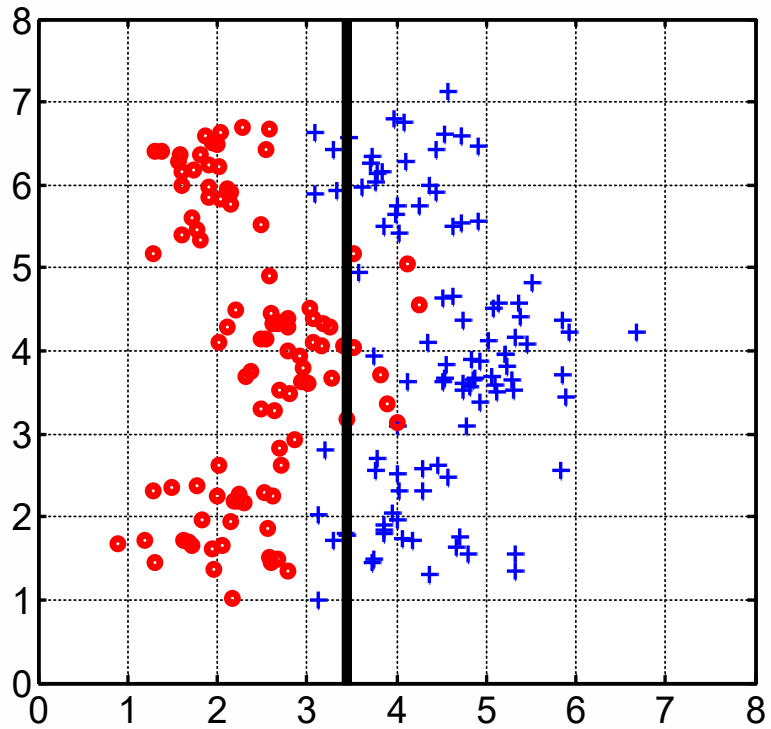
2. 거리가 가장 가까운 것부터 순서대로 K개의 데이터를 찾아 후보 집합 $N(\boldsymbol{x}) = \{\boldsymbol{x}^1, \boldsymbol{x}^2, \cdots, \boldsymbol{x}^K\}$를 만든다.

3. 후보 집합의 각 원소가 어떤 클래스에 속하는지 그 레이블값 $y(\boldsymbol{x}^1), y(\boldsymbol{x}^2), \cdots, y(\boldsymbol{x}^K)$ 을 찾는다.

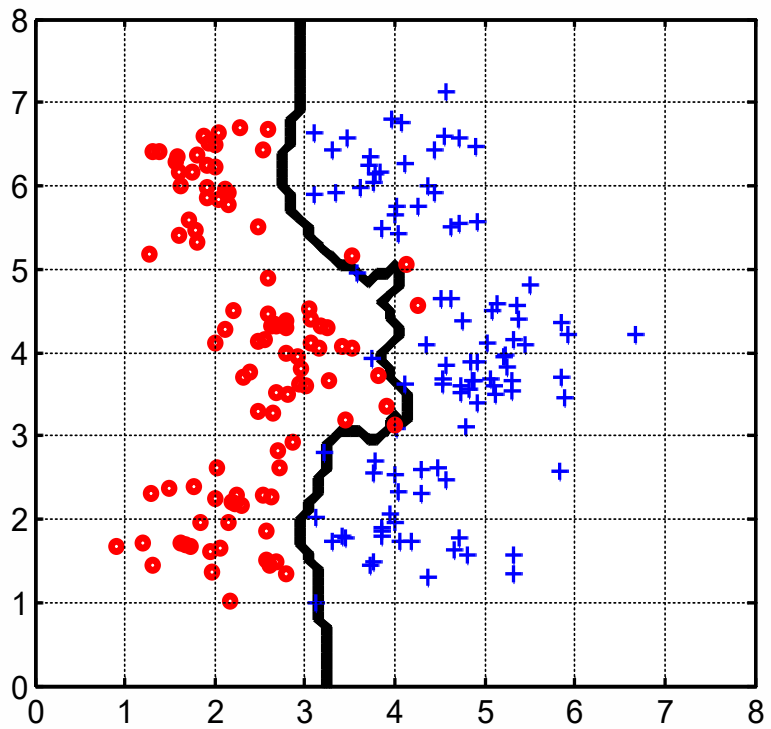
4. 찾아진 레이블값 중 가장 많은 빈도수를 차지하는 클래스를 찾아 $\boldsymbol{x}$를 그 클래스에 할당한다.

K-최근접이웃 분류기 vs 가우시안 베이즈 분류기

- 데이터 분포가 복잡한 비선형 구조를 가지는 경우



가우시안 베이즈 분류기



K-최근접이웃 분류기

K-최근접이웃 분류기 vs 가우시안 베이즈 분류기

가우시안 베이즈 분류기

- 각 클래스에 대한 확률분포함수를 미리 가정하고 추정
- 학습 데이터를 통해 평균과 표준편차만 계산하여 활용
  - ✓ 분류 과정에서 학습 데이터가 불필요

K-최근접이웃 분류기

- 확률분포모델을 미리 가정하지 않고 데이터 집합을 이용하여 추정
- 새 데이터가 주어질 때마다 학습 데이터 전체와의 거리 계산이 필요
  - ✓ 항상 학습 데이터를 저장 → 비용(계산량, 메모리) 증가

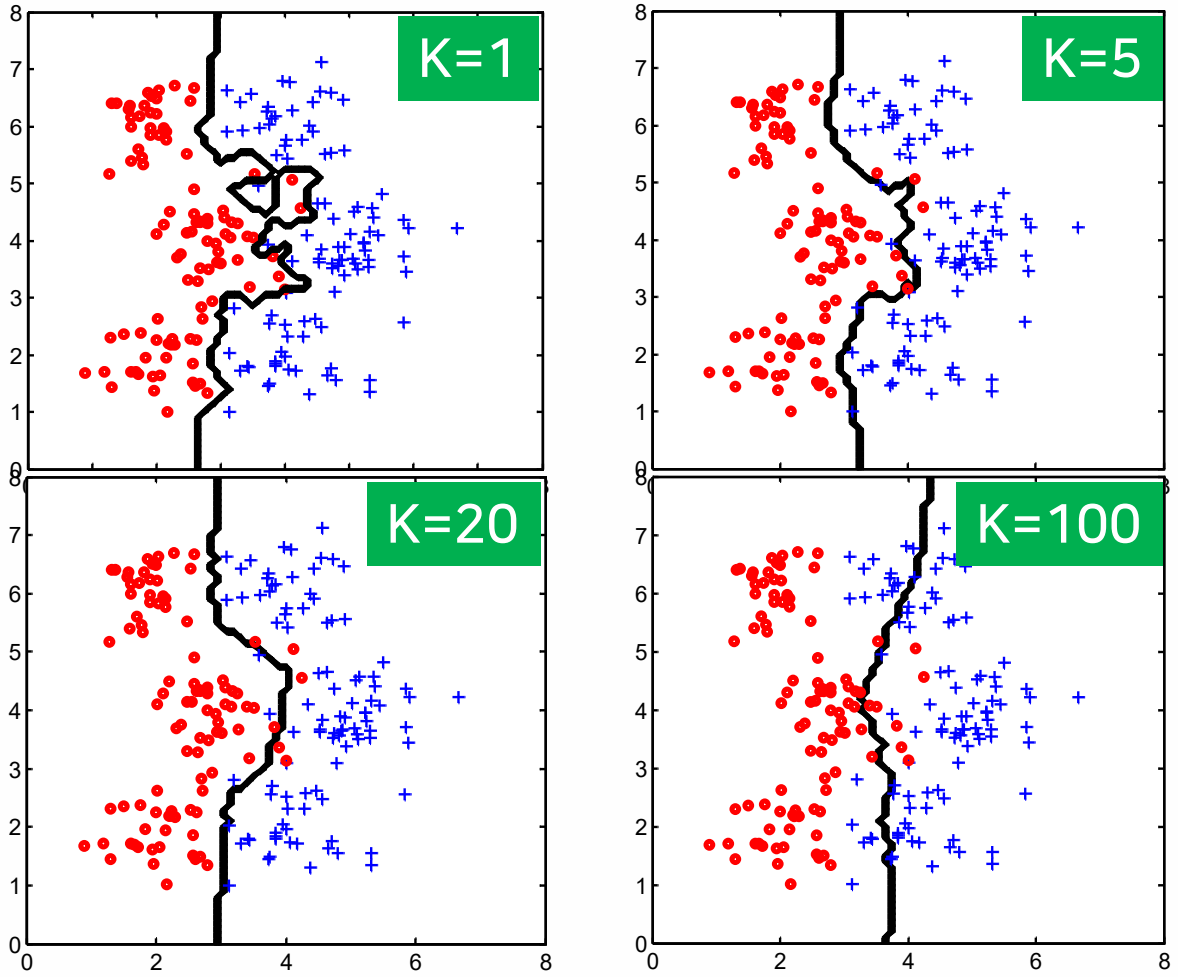
K-NN 분류기의 설계 고려사항

적절한 K값의 결정

- $K = 1$ → 바로 이웃한 데이터에만 의존하여 클래스가 결정
  → 노이즈에 민감, 과다적합 발생
- $K \gg 1$ → 주어진 데이터 주변 영역이 아닌 전체 데이터 영역에서 각 클래스가 차지하는 비율(선험확률)에 의존
- 주어진 데이터의 분포 특성에 의존
  - ✓ 주어진 데이터에 대한 분류를 통해 가장 좋은 성능을 주는 값을 선택

K-NN 분류기의 설계 고려사항

- K값에 따른 결정경계의 변화



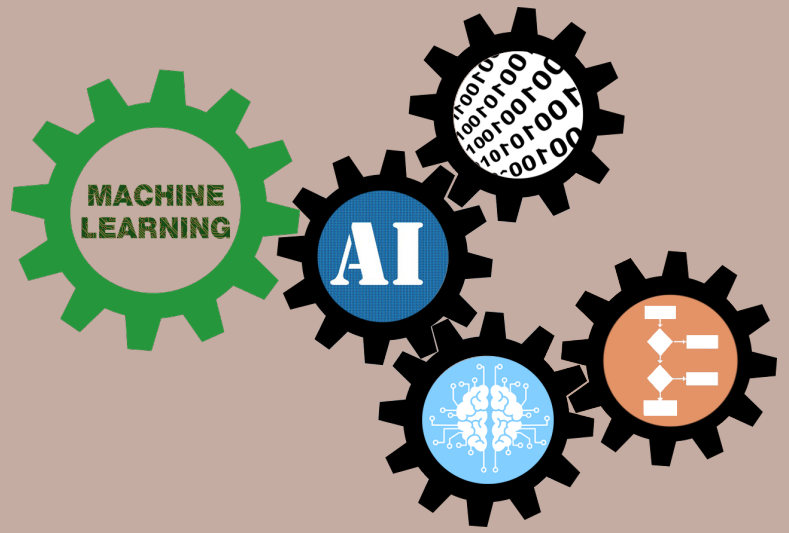
K-NN 분류기의 설계 고려사항

- 거리 함수? → 주어진 데이터와 학습 데이터 간의 거리 계산

2차 노름 (유클리디안 거리)	$d_E(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x}-\boldsymbol{y}\|_2 = \sqrt{(\boldsymbol{x}-\boldsymbol{y})^T(\boldsymbol{x}-\boldsymbol{y})} = \sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$
1차 노름	$d_1(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x}-\boldsymbol{y}\|_1 = \sum_{i=1}^{n}\|x_i-y_i\|$
p차 노름	$d_p(\boldsymbol{x}, \boldsymbol{y}) = \sqrt[p]{\sum_{i=1}^{n}(x_i-y_i)^p}$
내적	$d_{IN}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x}\cdot\boldsymbol{y} = \sum_{i=1}^{n}x_i y_i$
코사인 거리	$d_{\cos}(\boldsymbol{x}, \boldsymbol{y}) = 1 - \frac{\boldsymbol{x}\cdot\boldsymbol{y}}{\|\boldsymbol{x}\|\,\|\boldsymbol{y}\|}$
정규화된 유클리디안 거리	$d_{NE}(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\sum_{i=1}^{n}\frac{(x_i-y_i)^2}{\sigma_i^2}}$ (σ_i^2는 데이터의 분산)
마할라노비스 거리	$d_M(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{(\boldsymbol{x}-\boldsymbol{y})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{y})}$ ($\boldsymbol{\Sigma}$는 데이터의 공분산행렬)

그밖의 분류기들

- 로지스틱 회귀
  - 회귀 기법을 분류 문제로 확장
- 결정 트리
  - 속성들의 정보를 순차적으로 적용하여 분류 → 판단 결과에 대한 설명력이 우수
- 서포트벡터머신(SVM)
  - 결정경계의 마진을 최대화하는 목적함수 사용 → 일반화 성능 우수
- 신경망(딥러닝 모델)
  - 복잡한 결정경계를 신경망 모델로 정의하여 학습
  - 특징추출 단계까지 한 번에 학습



다음시간안내

제3강

지도학습: 회귀