딥러닝 스터디

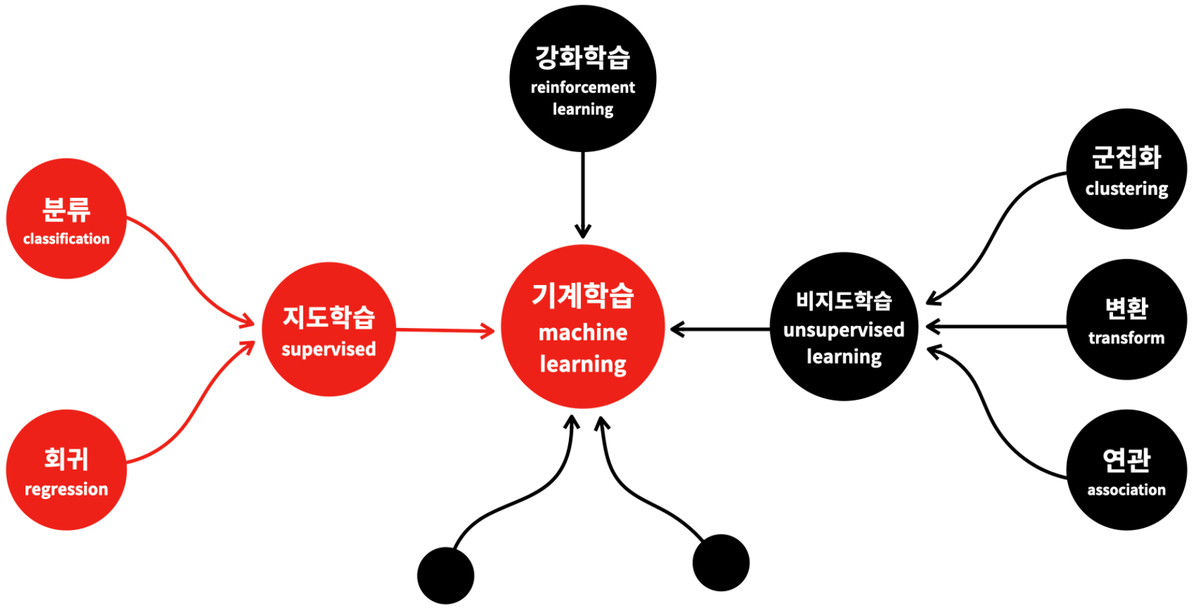
4주차 지도학습 .

20기 시각화 노승혜
20기 분 석 이민선
19기 시각화 정다운

3.1
지도학습

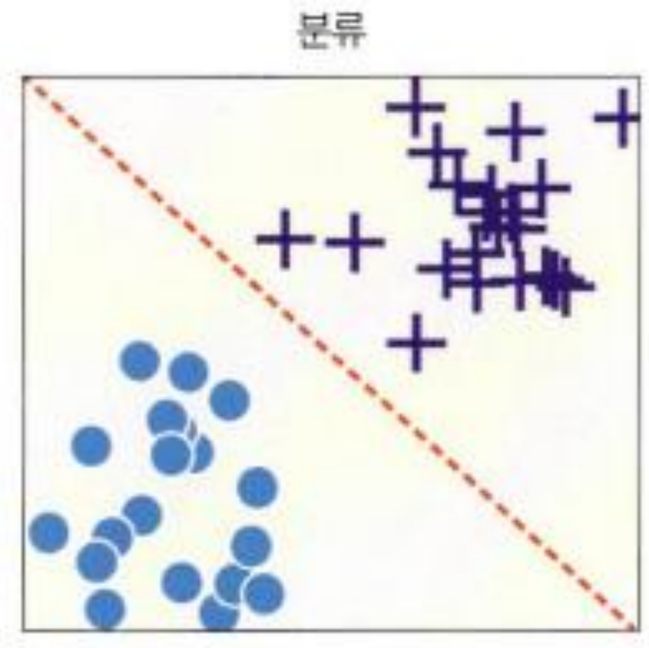
• 지도학습(Supervised Learning)

- 정답이 있는 데이터를 활용해 데이터를 학습시키는 것
- 입력 값(X data)가 주어지면 입력 값에 대한 Label(Y data)를 주어 학습
- 분류(KNN, Decision Tree, SVM, 로지스틱 …), 회귀(선형회귀)



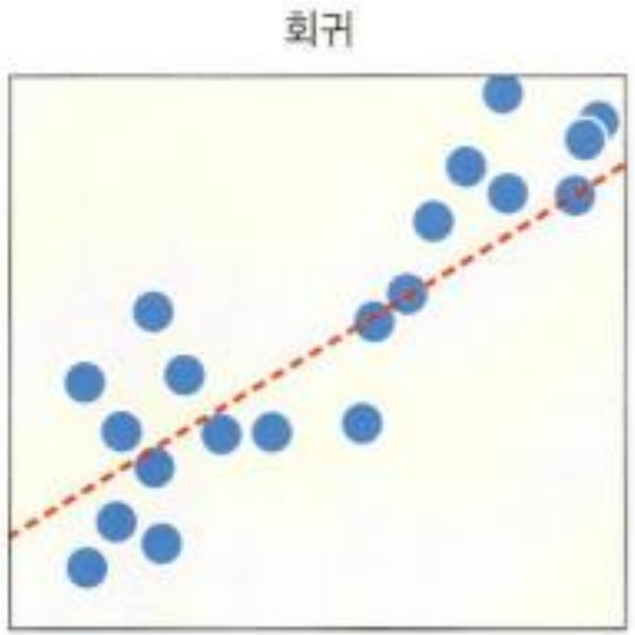
• 분류(Classification)

- 주어진 데이터를 정해진 카테고리에 따라 분류
- 예측 결과가 이산형
- 이진분류(Yes or No), 다중분류(고양이 or 사자 or 강아지)



• 회귀(Regression)

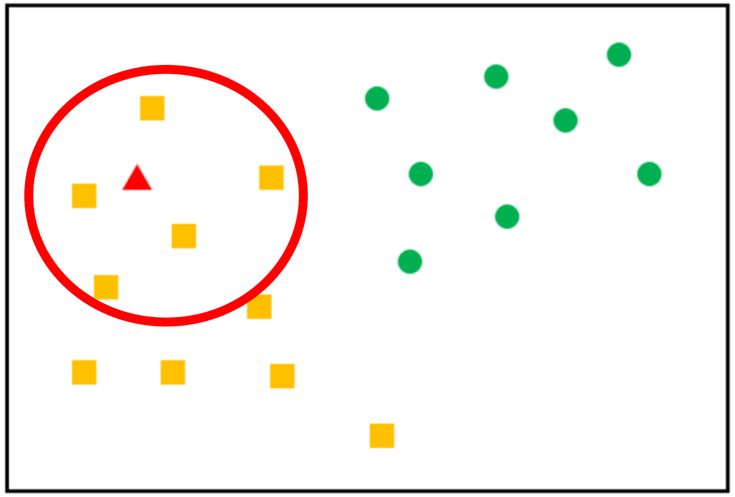
- 데이터들의 feature를 기준으로, 연속된 값을 예측
- 수치나 통계학적 방법으로 답을 도출해내는 방법
- feature와 label 사이의 상관관계를 함수식으로 표현

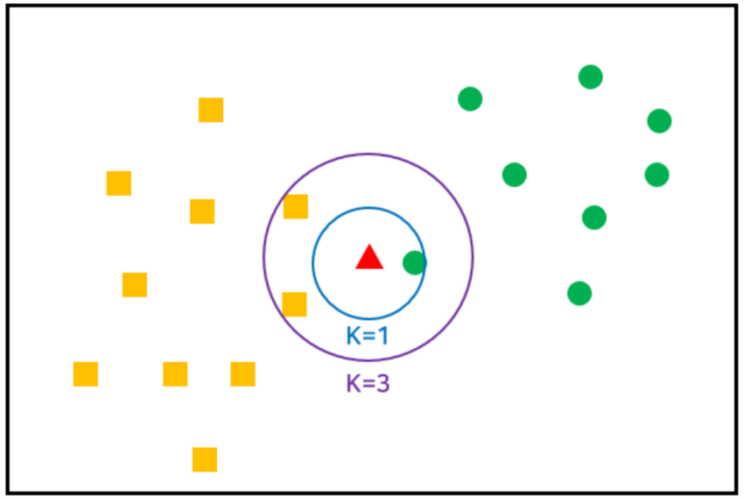


3.1.1
K-최근접 이웃

• K-최근접 이웃(K-nearest neighbor)

- 데이터로부터 거리가 가까운 K개의 주변 데이터를 참조해 데이터가 속할 그룹 분류하는 알고리즘
- 기존 데이터와 단순 비교, 별도의 모델을 학습하지 않음
- K는 항상 분류가 가능하도록 홀수로 설정
- 가장 적절한 K는 일반적으로 총 데이터 수의 제곱근 값을 사용
- ex. K = 5, 새로운 데이터와 가장 가까운 5개의 클래스를 분석해 클래스를 할당





• K-최근접 이웃(K-nearest neighbor)

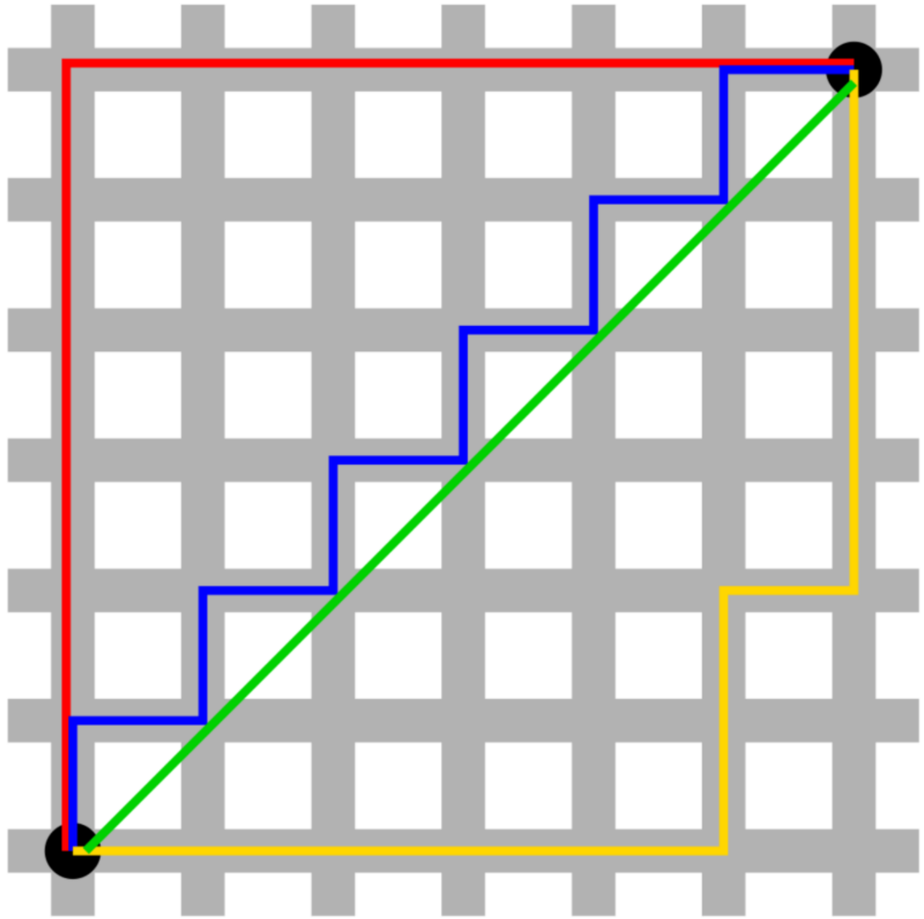
◦ 거리 측정 방법

- 유클리디안 거리(Euclidean Distance)
  - 일반적인 KNN 알고리즘 거리 측정 방법
  - 점과 점 사이의 직선거리

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_i |x_i - y_i|^2}$$

- 맨해튼 거리(Manhattan Distance)
  - 건물을 가로지르지 않고 갈 수 있는 최단거리
  - 가로와 세로의 길이의 합

$$d(\mathbf{x}, \mathbf{y}) = \sum_i |x_i - y_i|$$



• K-최근접 이웃(K-nearest neighbor)

◦ 장점

- 단순하기 때문에 다른 알고리즘에 비해 구현이 쉬움
- 훈련 단계가 매우 빠름

◦ 단점

- 모델을 생성하지 않기 때문에 특징과 클래스 간 관계 이해하는데 제한적
- 적절한 K의 선택 필요
- 데이터가 많아지면 분류 단계가 느려짐

• KNN 알고리즘 표준화

◦ 표준화

- KNN 알고리즘과 같은 거리 기반 모델의 경우, 구현 시 변수 값의 범위 재조정 필요
- 분포가 다르면 각 변수의 차이를 해석하기 어렵고, 변수의 중요도를 고르게 해석하기 위해

① 최소 – 최대 정규화 $Z = (X - min(X))/(max(X) - min(X))$

· 변수의 범위를 0%에서 100%

· 예측할 데이터 셋에서 최솟값과 최댓값이 범위를 벗어나는 경우 발생

· 수치형 데이터 중 범위가 한정된 경우에는 사용 가능

② Z – 점수 표준 정규화 $Z = (X - 평균)/표준편차$

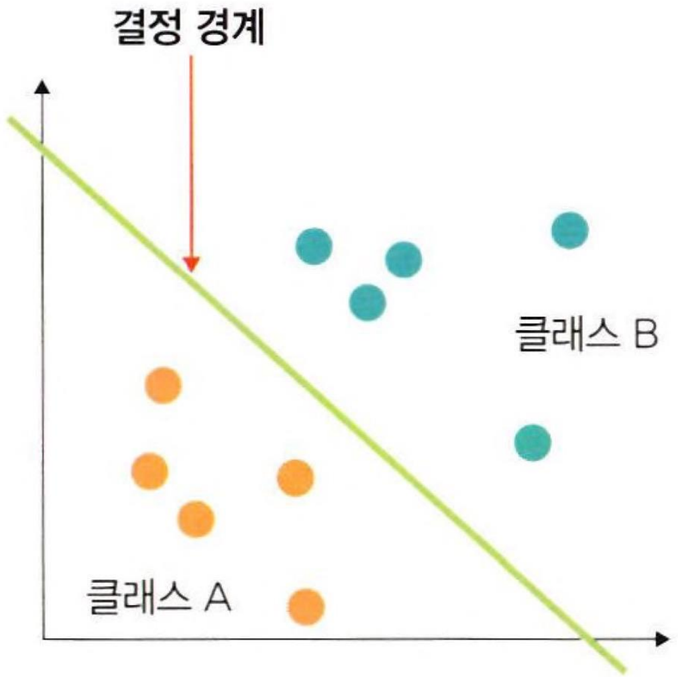
· 변수의 범위 정규분포화

· 평균 = 0, 표준편차 = 1

3.1.2
서포트 벡터 머신

• 서포트 벡터 머신 (Support Vector Machine)

- 분류를 위한 기준선을 정의하는 모델
- 데이터를 분류하기 위한 기준선 -> '결정 경계'
- 새로운 데이터가 나타나면 결정 경계를 기준으로 경계의 어느 쪽에 속하는지를 분류

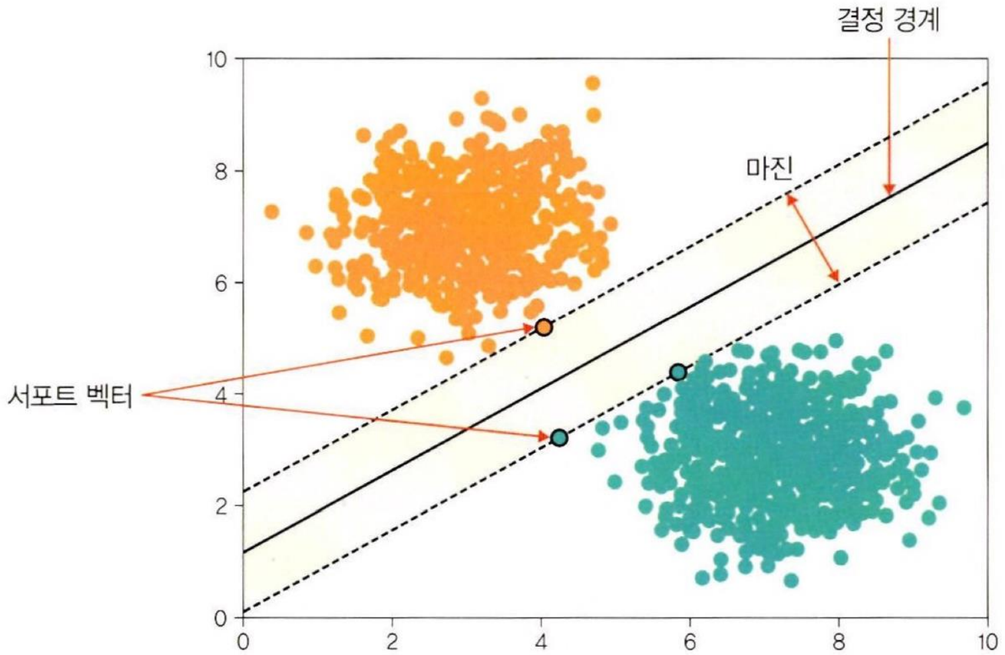


• 서포트 벡터 머신 (Support Vector Machine)

결정 경계의 위치를 결정하는 방법

- 마진(margin): 결정 경계와 서포트 벡터 사이의 거리
- 서포트 벡터(support vector): 결정 경계와 가까이 있는 데이터들

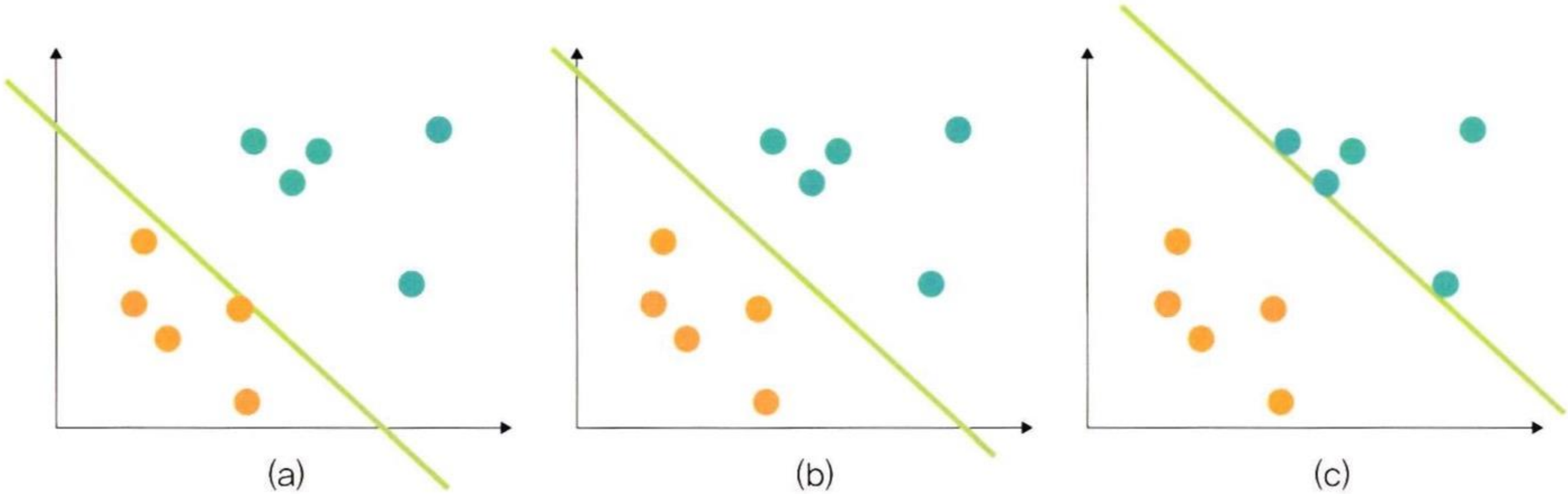
-> 경계를 정의하는 결정적인 역할!



최적의 결정 경계 = 최대의 마진 !

• 서포트 벡터 머신 (Support Vector Machine)

결정 경계의 위치를 결정하는 방법



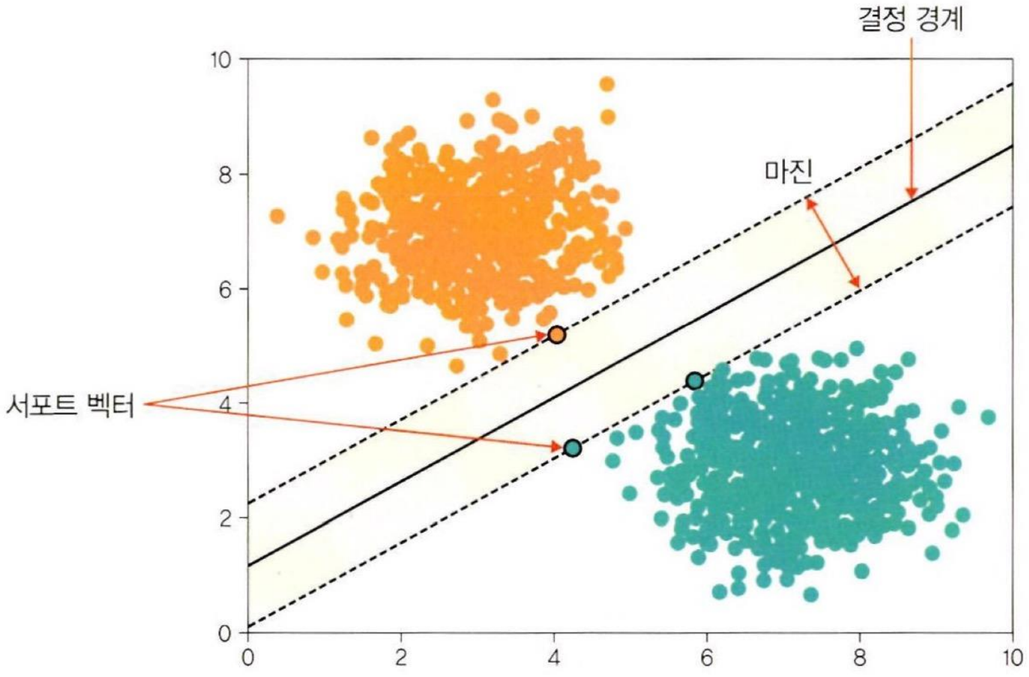
최적의 결정 경계 = 최대의 마진 !

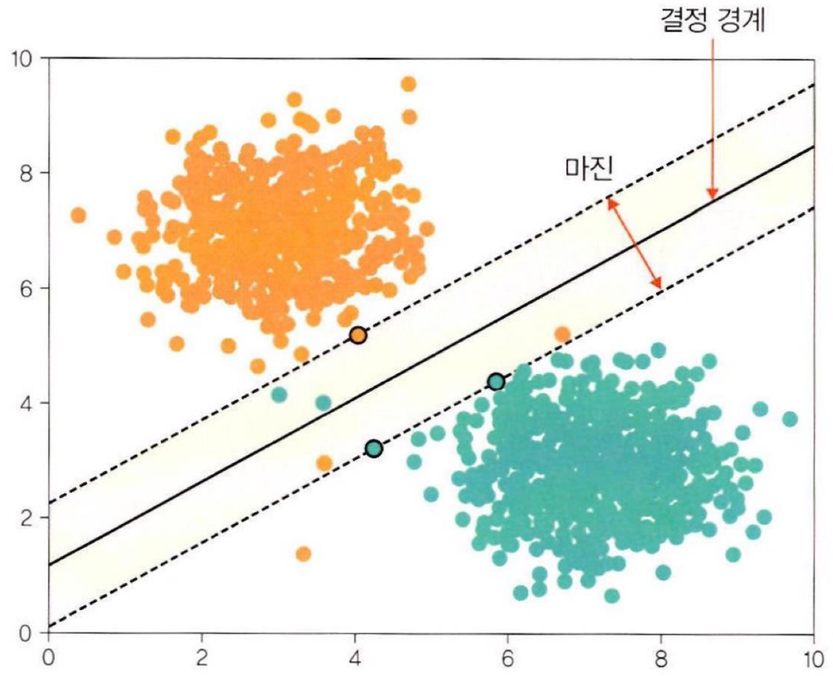
• 서포트 벡터 머신 (Support Vector Machine)

이상치 허용 여부에 따라

하드 마진 vs. **소프트 마진**

이상치들이 마진 안에 포함되는 것을 허용하지 않음 / 허용

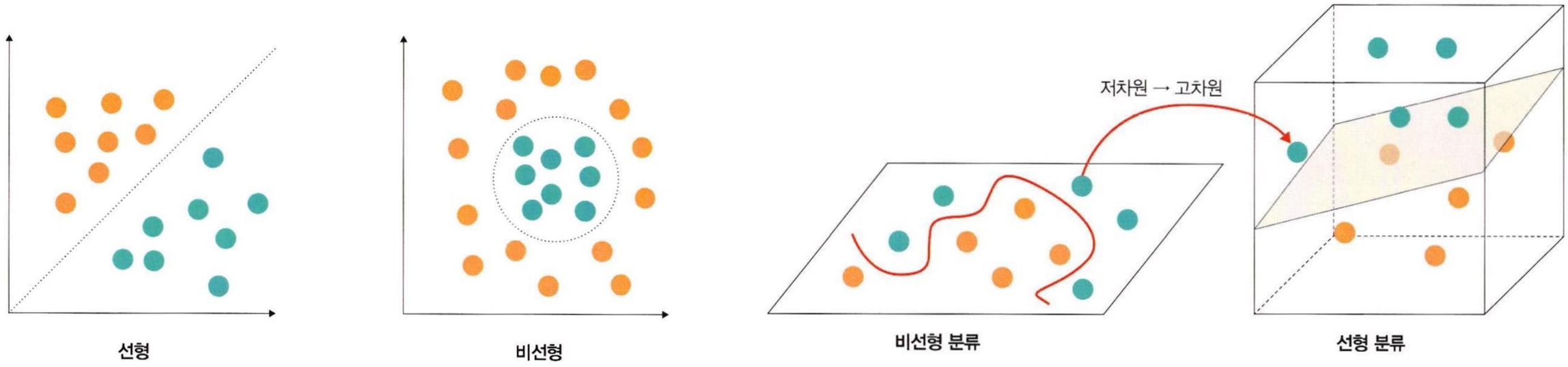




• 서포트 벡터 머신 (Support Vector Machine)

선형 분류와 비선형 분류

비선형 문제 해결 방법: 저차원 데이터를 고차원으로 보내기
-> 그러나 많은 수학적 계산이 필요하기 때문에 성능에 문제를 줄 수 있음



* 저차원 데이터: 특성이 적은 데이터, 고차원 데이터: 특성이 많은 데이터

• 서포트 벡터 머신 (Support Vector Machine)

커널 트릭(Kernel Trick)
: 벡터 내적을 이용해 고차원으로 보내는 방법으로 연산량을 줄임

- 선형 커널: 선형으로 분류 가능한 데이터에 적용
(기본 커널 트릭으로, 커널을 사용하지 않겠다는 의미와 일맥상통함)

$$K(a,b) = a^T \cdot b$$

(a, b: 입력 벡터)

- 다항식 커널: 실제로는 특성을 추가하지 않지만,
다항식 특성을 많이 추가한 것과 같은 결과를 얻을 수 있는 방법
따라서 고차원으로 데이터 매핑 가능

$$K(a,b) = (\gamma a^T \cdot b)^d$$

(a, b: 입력 벡터
γ: 감마
d: 차원, 이때 γ, d는 하이퍼파라미터)

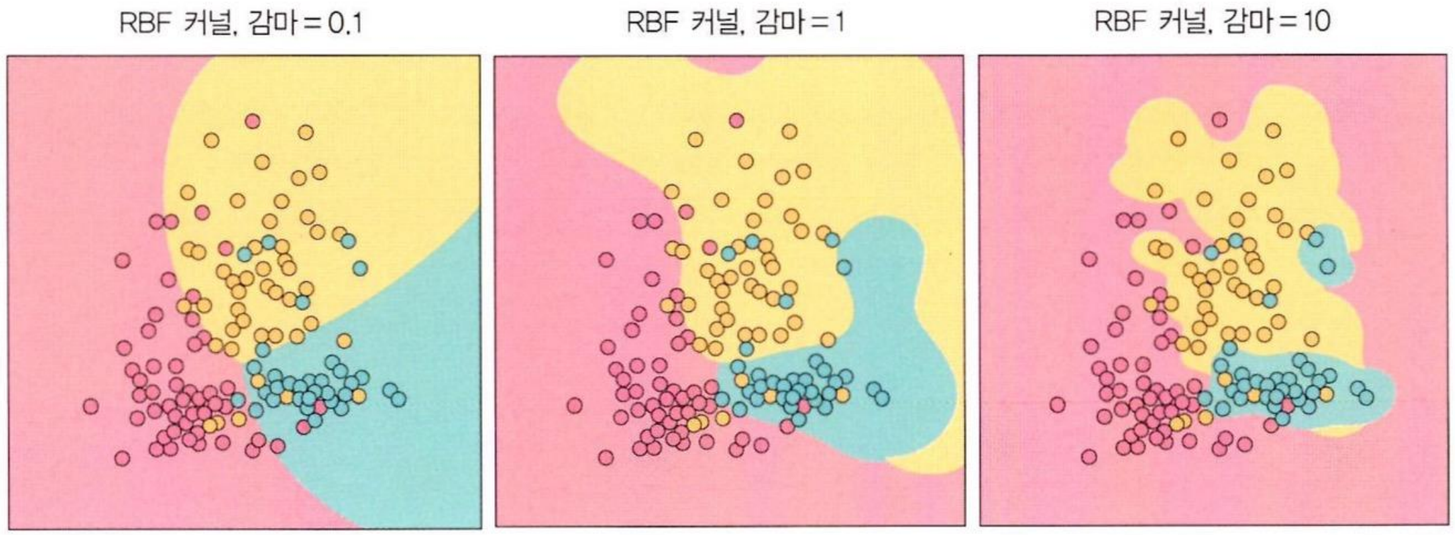
- 가우시안 RBF 커널: 다항식 커널의 확장 버전,
입력 벡터를 차원이 무한한 고차원으로 매핑하여
모든 차수의 모든 다항식을 고려
즉, 다항식 커널은 차수에 한계 존재, 가우시안 RBF는 차수 제한 없음

$$K(a,b) = \exp(-\gamma \| a-b \|^2)$$

(이때 γ는 하이퍼파라미터)

• 서포트 벡터 머신 (Support Vector Machine)

커널 트릭(Kernel Trick)
: 벡터 내적을 이용해 고차원으로 보내는 방법으로 연산량을 줄임

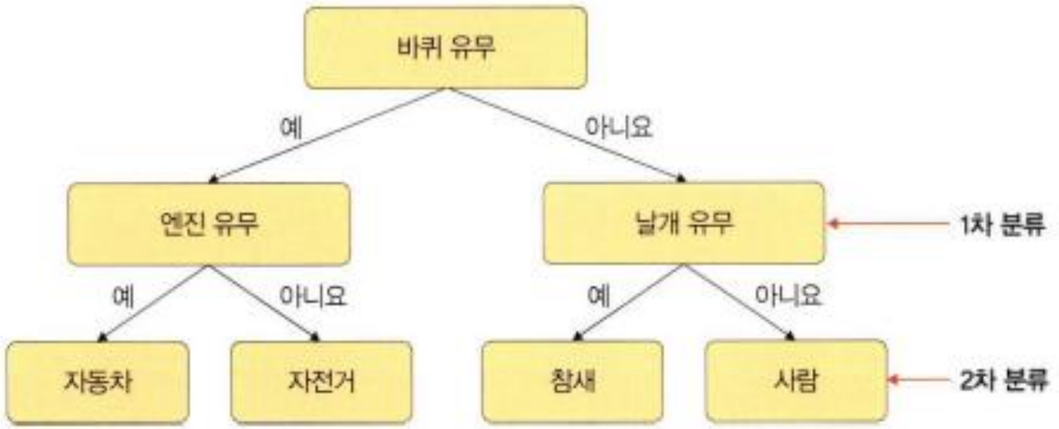


- 감마의 역할: 결정 경계를 얼마나 유연하게 가져갈지
- 감마 값이 클수록 훈련 데이터에 많이 의존하기 때문에 과적합 초래 주의

3.1.3
결정 트리

• 결정 트리 (Decision Tree)

- 대표적인 분류 모델
- 이상치가 많은 값으로 구성된 data set을 다룰 때 사용
- Basic Algorithm (a greedy algorithm 미래를 생각하지 않고 각 단계에서 가장 최선의 선택을 하는 기법)
- 질문을 반복적으로 던짐으로써 규칙을 찾는 Rule based 방식



• 결정 트리 (Decision Tree)

- 데이터 분류 시, 정답률(순도)이 높은 조건으로 구분
- 점점 조건을 세분화하면서 순도 높이기

◦ 불순도(불확실성) 계산 방법

- 엔트로피 (Entropy)
- 지니 계수 (Gini index)

불순도가 낮은 변수로 질문(root node) 생성

• 엔트로피 (Entropy)

- 불확실성을 측정하는 지표

  높은 엔트로피 = 높은 불확실성 = 낮은 순도(정답률)

  낮은 엔트로피 = 낮은 불확실성 = 높은 순도(정답률)

- $$Entropy(A) = -\sum_{k=1}^{m} p_k \log_2(p_k)$$

  $p_k = A$ 영역에 속하는 데이터 가운데 k 범주에 속하는 데이터 비율

- e.g.

  동전 두 번 던져 앞면이 나올 확률 : 1/4

  뒷면이 나올 확률 : 3/4

• 지니 계수 (Gini index)

- 불확실성을 측정하는 지표
- 로그를 계산할 필요가 없어 엔트로피보다 계산이 빠름 → 결정 트리에서 많이 사용
- $G(S) = 1 - \sum_{i=1}^{c} p_i^2$

  S : 이미 발생한 사건의 모음, c : 사건 개수

- e.g.

Age 변수	변수 발생 확률	Computer = yes	Computer = no
<= 30	$\frac{5}{14}$	2	3
31 … 40	$\frac{4}{14}$	4	0
> 40	$\frac{5}{14}$	3	2

• 혼동 행렬 (confusion matrix)

◦ Accuracy : $\frac{(TP+TN)}{ALL}$

전체 대비 예측 맞춘 비율

(Class가 동일하게 나누어져 있는 경우에 유용)

◦ Error Rate : $\frac{(FP+FN)}{ALL}$

전체 대비 예측 틀린 비율 (1 – Accuracy)

◦ Sensitivity : $\frac{TP}{(TP+FP)}$

실제 Positive 중에 맞춘 것의 비율

◦ Specificity : $\frac{TN}{(TN+FN)}$

실제 Negative 중에 맞춘 것의 비율

		예측 값	
		Positive	Negative
실제 값	Positive	TP	FN
	Negative	FP	TN

• 혼동 행렬 (confusion matrix)

◦ Precision : $\frac{TP}{(TP+FP)}$

Positive라고 예측한 것 중, 맞춘 것 (예측률, 정밀도)

◦ Recall : $\frac{TP}{(TP+FN)}$

실제 Positive 중에서 맞춘 것 (재현율)

◦ F1-score : $\frac{2 \times Precision \times Recall}{Precision+Recall}$

- 정밀도와 재현율의 trade-off 문제를 해결하기 위함
- 둘의 조화 평균을 이용

		예측 값	
		Positive	Negative
실제 값	Positive	TP	FN
	Negative	FP	TN

◦ F_β : $\frac{(1+\beta^2) \times Precision \times Recall}{(\beta^2 \times Precision) + Recall}$

Precision이 더 중요하면 β값 높여주고,

Recall이 더 중요하면 β값 낮춰줌 ($\beta = 1$이면 F1-score와 동일)

3.1.4
로지스틱 회귀와 선형 회귀

• 회귀 (Regression)

◦ 사용

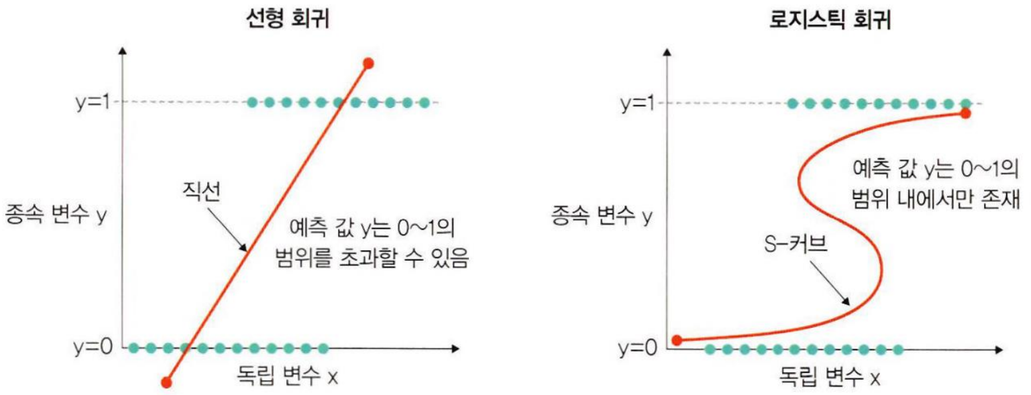
- 변수가 두 개 주어졌을 때 한 변수에서 다른 변수를 예측
- 두 변수의 관계를 규명

◦ 변수 유형

- 독립 변수(예측 변수) : 영향을 미칠 것으로 예상되는 변수
- 종속 변수(기준 변수) : 영향을 받을 것으로 예상되는 변수

◦ 로지스틱 회귀 : 예측값 y는 0~1의 범위 내에서만 존재

◦ 선형 회귀 : 예측값 y는 0~1의 범위를 초과할 수 있음



• 로지스틱 회귀 (Logistic Regression)

◦ 사용

- 분류 결과에 대해 확신이 없는 경우
- 향후 추가적으로 훈련 데이터셋을 수집하여 모델을 훈련시킬 수 있는 환경

◦ 절차

- 1단계) 각 집단에 속하는 확률의 추정치 예측
- 2단계) 분류 기준 값 설정 후 특정 범주로 분류

e.g. $P(Y = 1) \geq 0.5 \rightarrow$ 집단 1로 분류

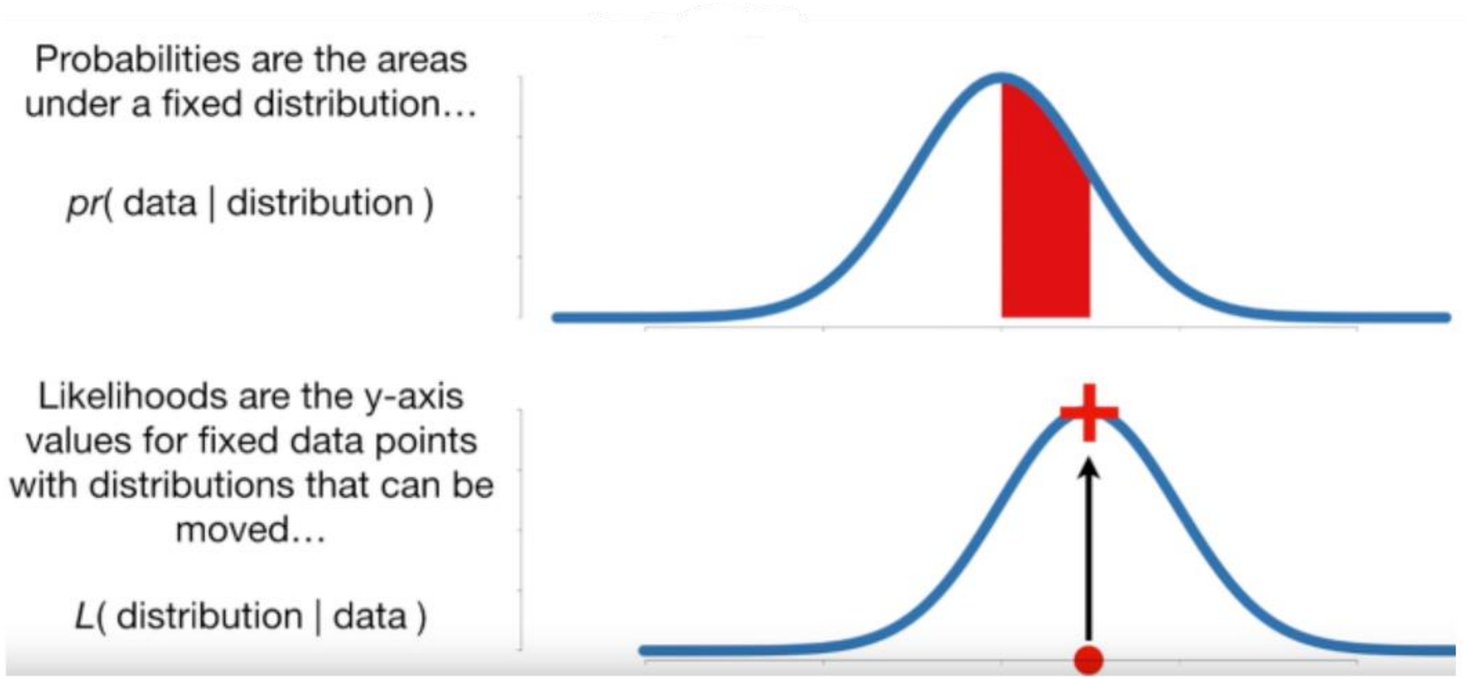
$P(Y = 1) < 0.5 \rightarrow$ 집단 0으로 분류

• 로지스틱 회귀 (Logistic Regression)

◦ 최대우도법(maximum likelihood)

- 우도(likelihood) : 어떤 값이 관측되었을 때, 해당 관측값이 어떤 확률분포로부터 나왔는지에 대한 확률

확률"의 개념과는 반대로 고정되는 요소가 확률분포가 아닌 관측값



• 로지스틱 회귀 (Logistic Regression)

◦ 최대우도법(maximum likelihood)

- 각 관측값에 대한 총 가능도(모든 가능도의 곱)가 최대가 되게 하는 분포를 찾는 방법
- 각 관측값 마다의 x값을 곱한 것 (데이터들의 추출이 독립적으로 연달아 발생)
- likelihood function

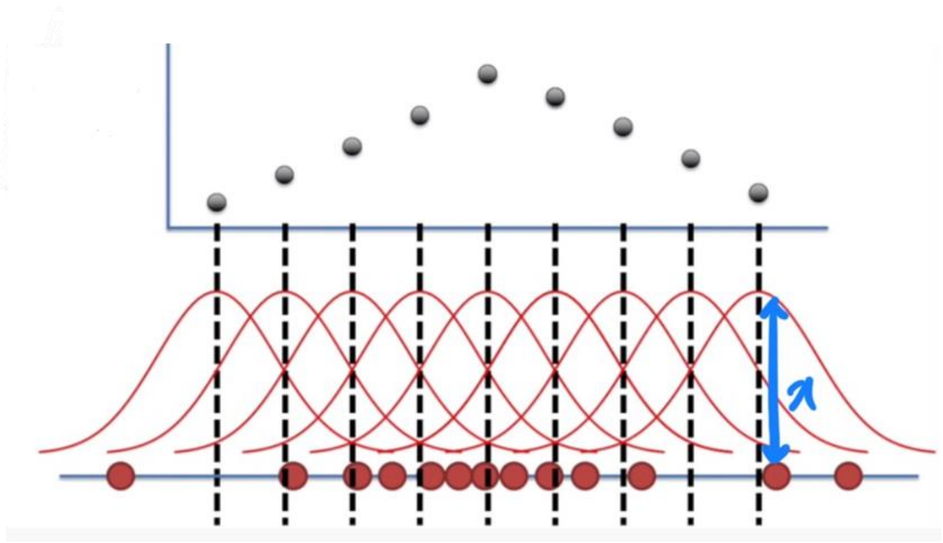
$$P(x|\theta) = \prod_{k=1}^{n} P(x_k|\theta)$$

$$L(\theta|x) = \log P(x|\theta) = \sum_{i=1}^{n} \log P(x_i|\theta)$$

- maximum likelihood function

$$\theta_{ml} = arg\,\max_{\theta}\ P_{model}\,(Y \mid X;\theta)$$

$$\theta_{ml} = arg\,\max_{\theta}\ \sum_{i=1}^{m} log\, P_{model}\,(y_i \mid x_i;\theta)$$



참고자료 : https://jjangjjong.tistory.com/41
https://junha1125.tistory.com/24

• 선형 회귀 (Linear Regression)

◦ 선형 회귀 종류

- 단순 선형 회귀 : 하나의 x값으로 y값 설명
- 다중 선형 회귀 : 여러 개의 x값으로 y값 설명

구분	일반적인 회귀 분석	로지스틱 회귀 분석
종속 변수	연속형 변수	이산형 변수
모형 탐색 방법	최소제곱법	최대우도법
모형 검정	F-테스트, t-테스트	X^2 테스트

◦ 평균제곱법 $MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

◦ 루트 평균제곱법 : $\mathrm{RMSE} = \sqrt{MSE}$

모델 정확도가 높진 않지만, 여전히 합리적으로 좋은 예측할 수 있음을 의미

◦ e.g.

y = 220 * X 에서 x값을 알면 y값을 추정할 수 있는 것과 동일