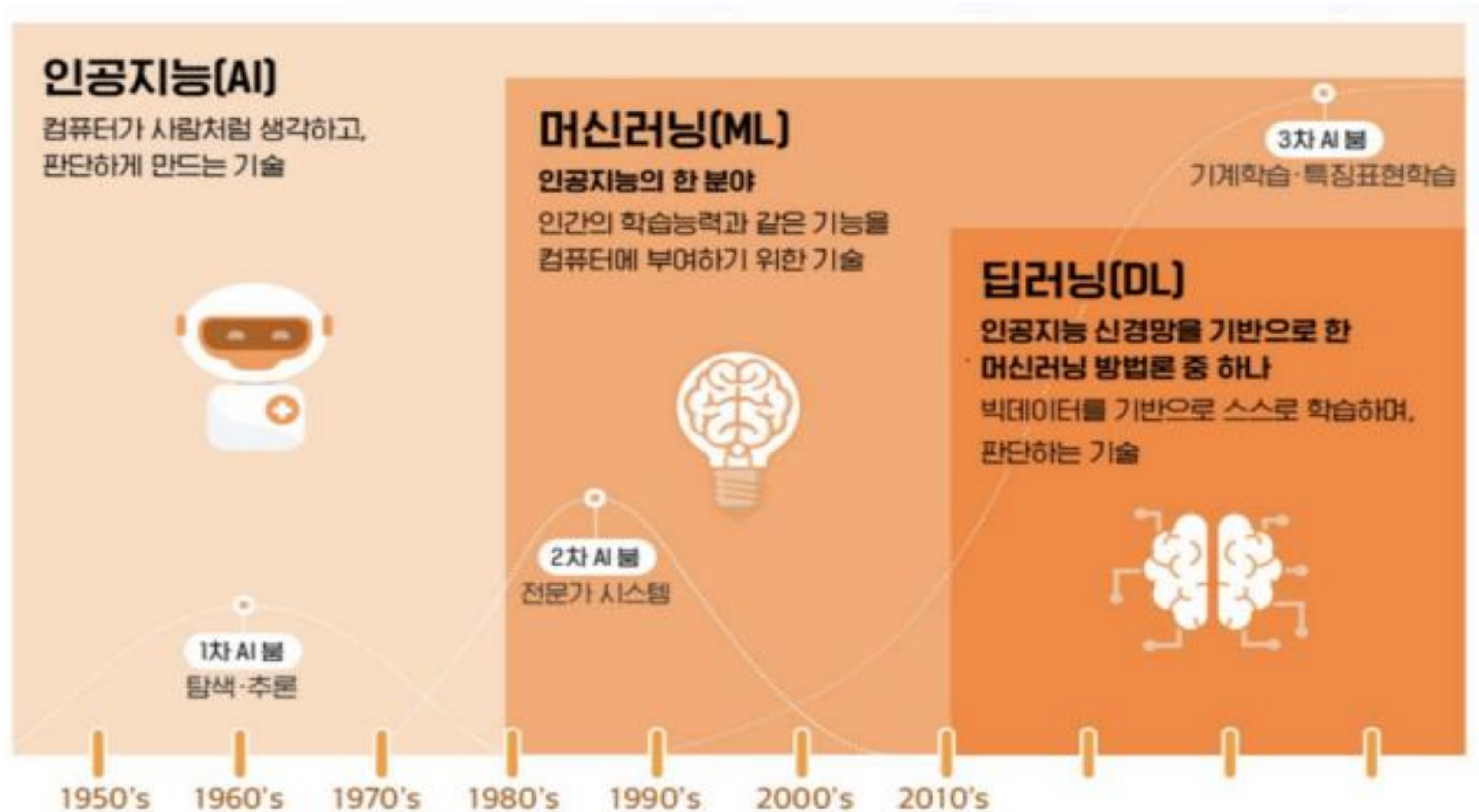
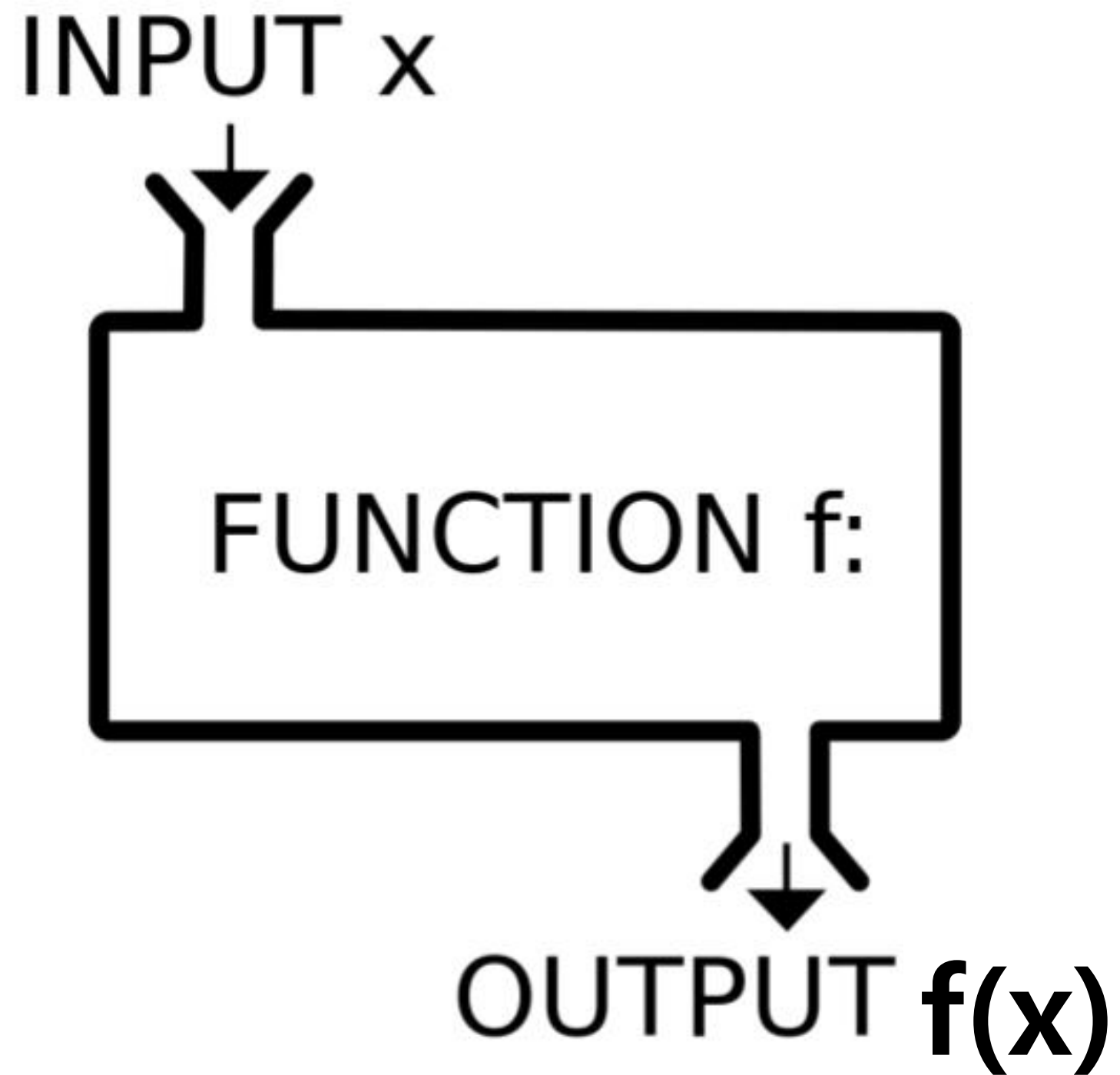


인공지능 알고리즘

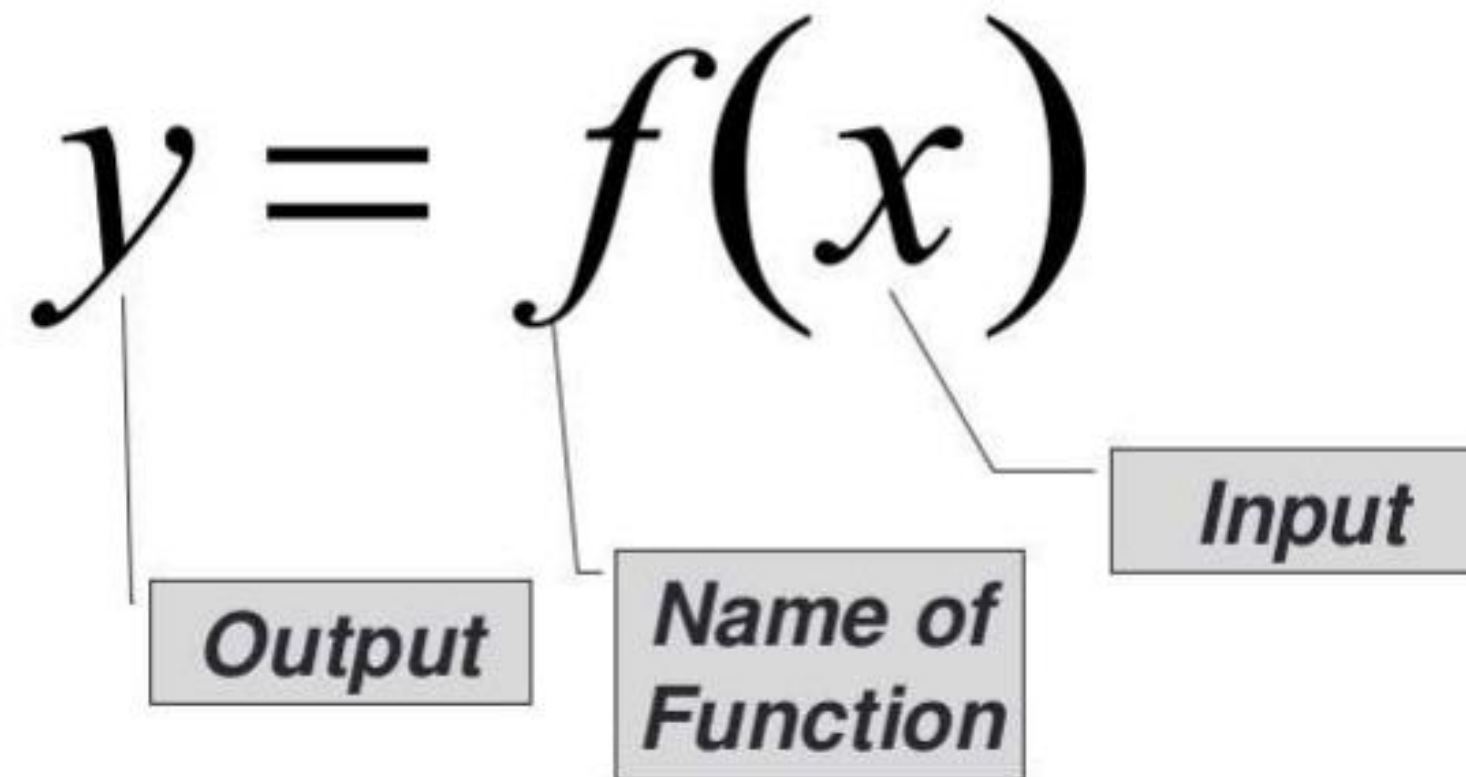
- 분류 -



✓ Computer Science

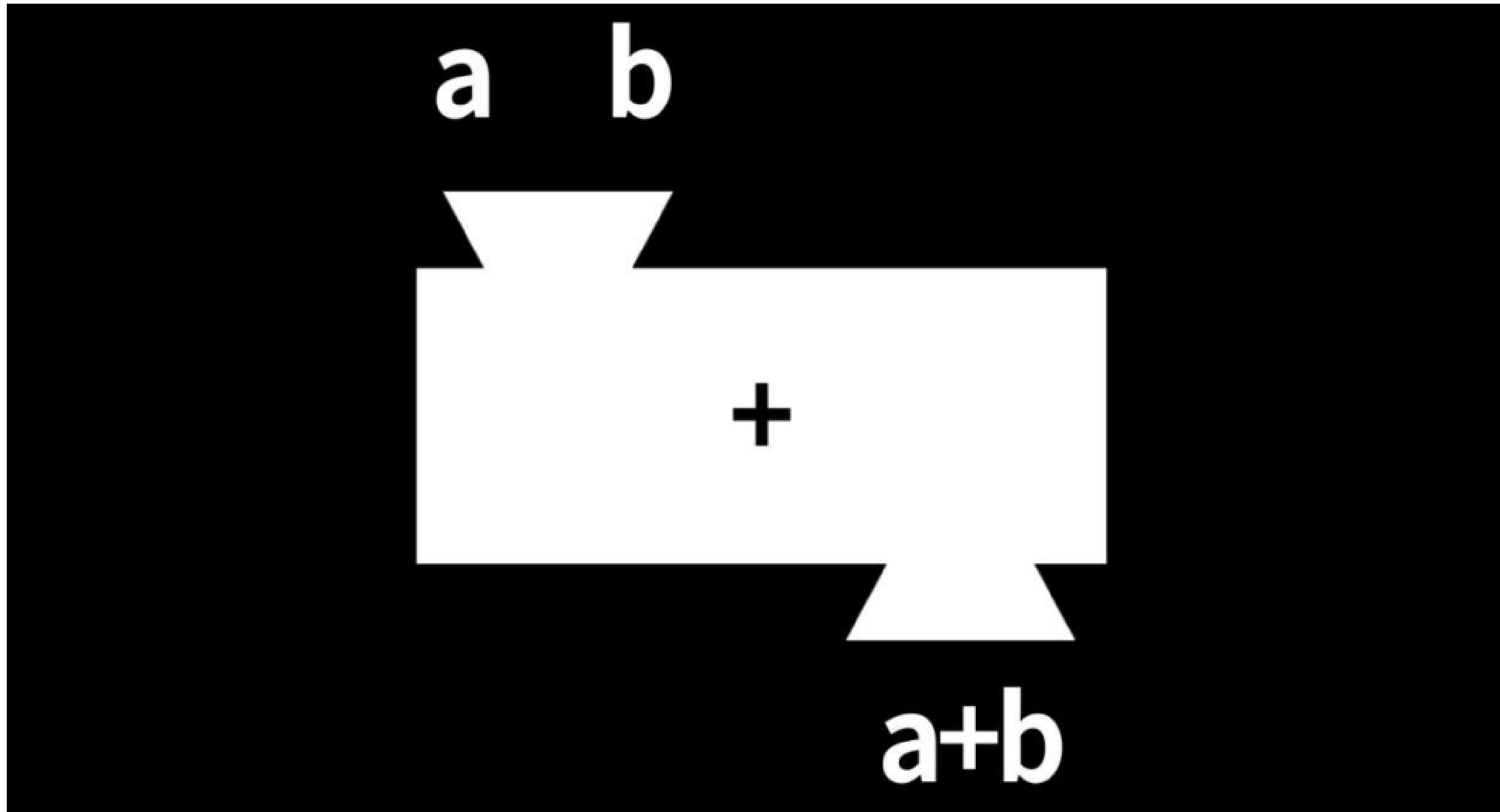


Function Notation

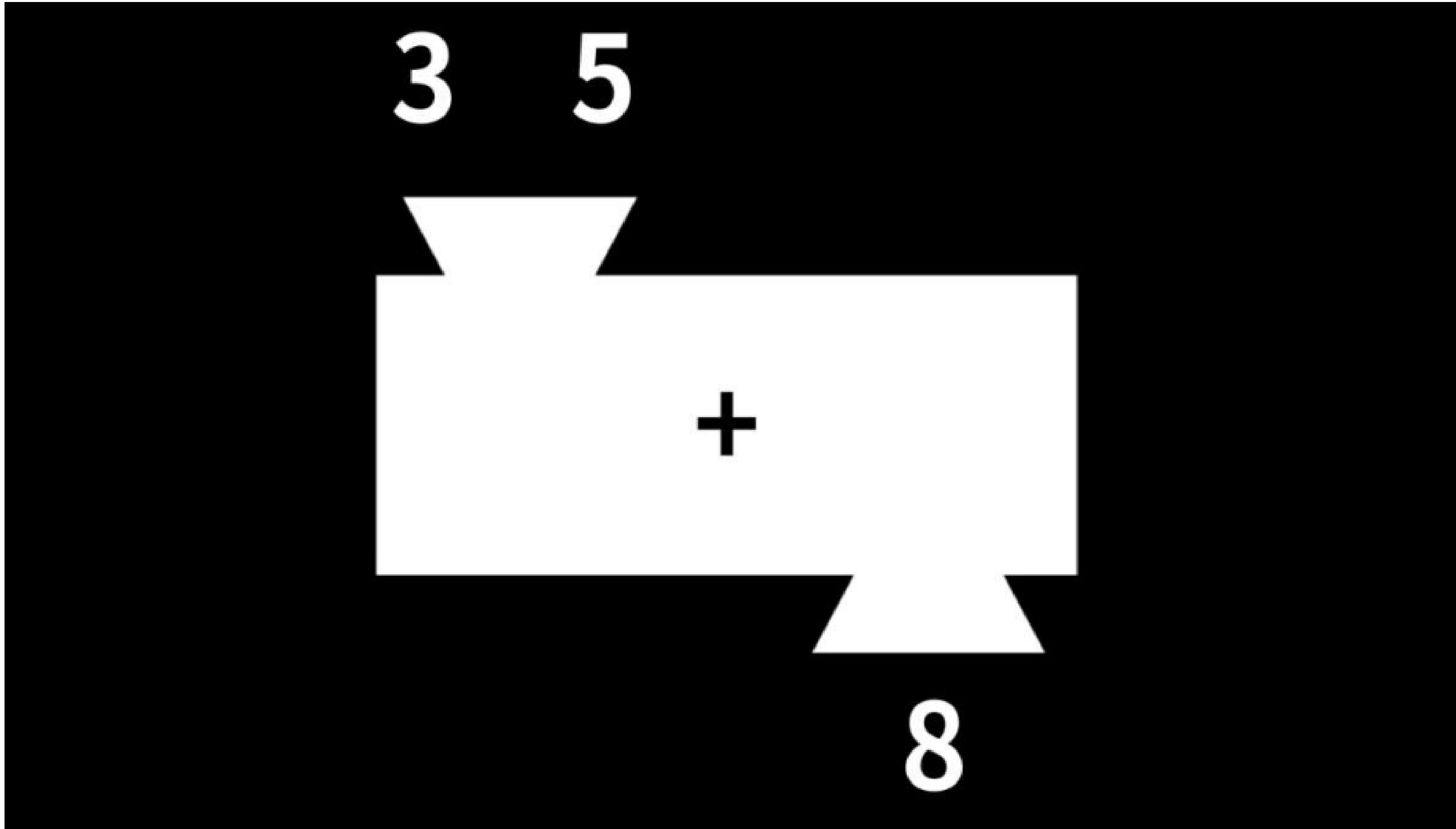
$$y = f(x)$$


A diagram illustrating function notation. The equation $y = f(x)$ is shown. Below the equation, three labels in gray boxes are connected to the symbols by lines: "Output" is connected to y , "Name of Function" is connected to f , and "Input" is connected to x .

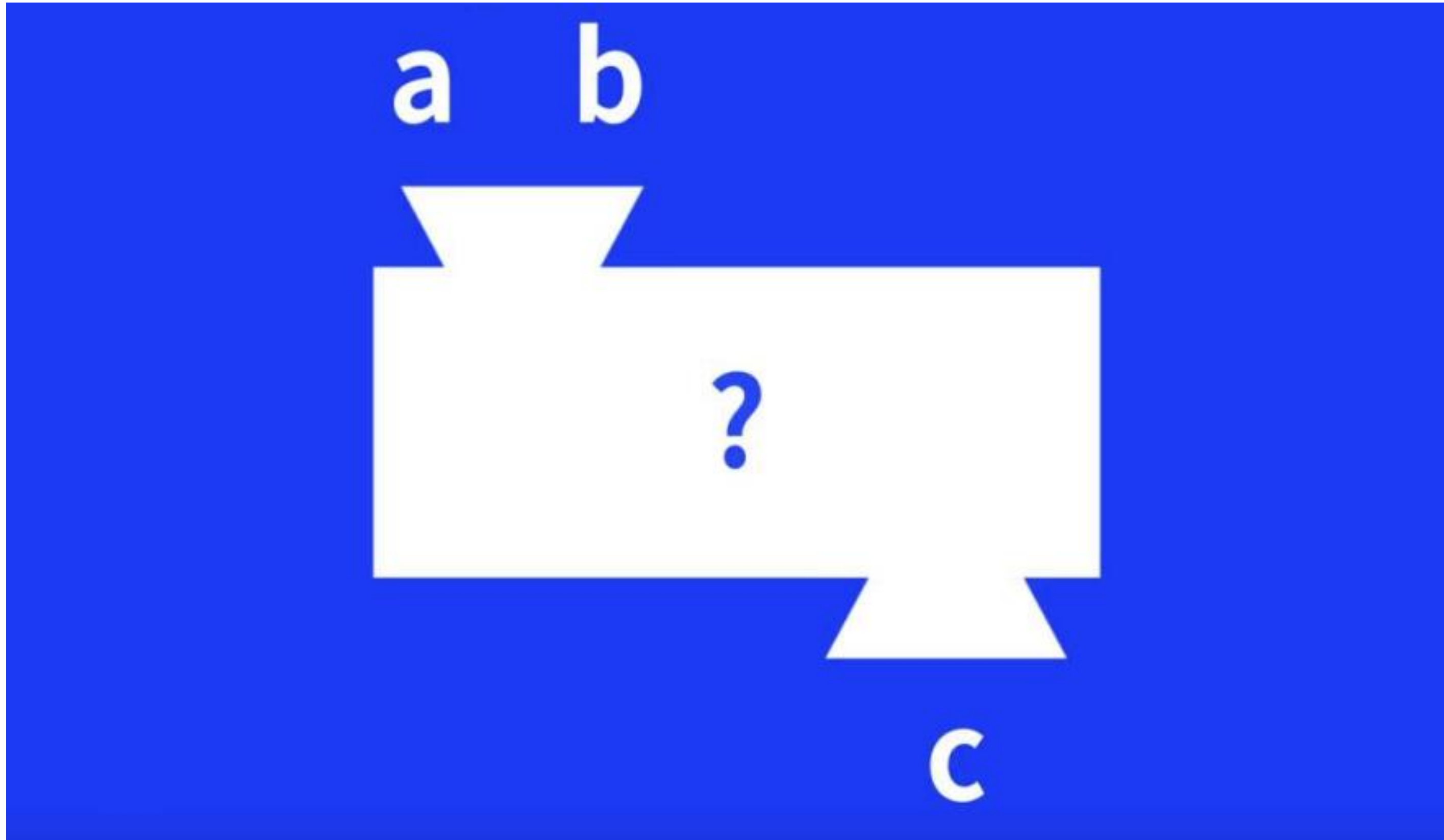
✔ Computer Science



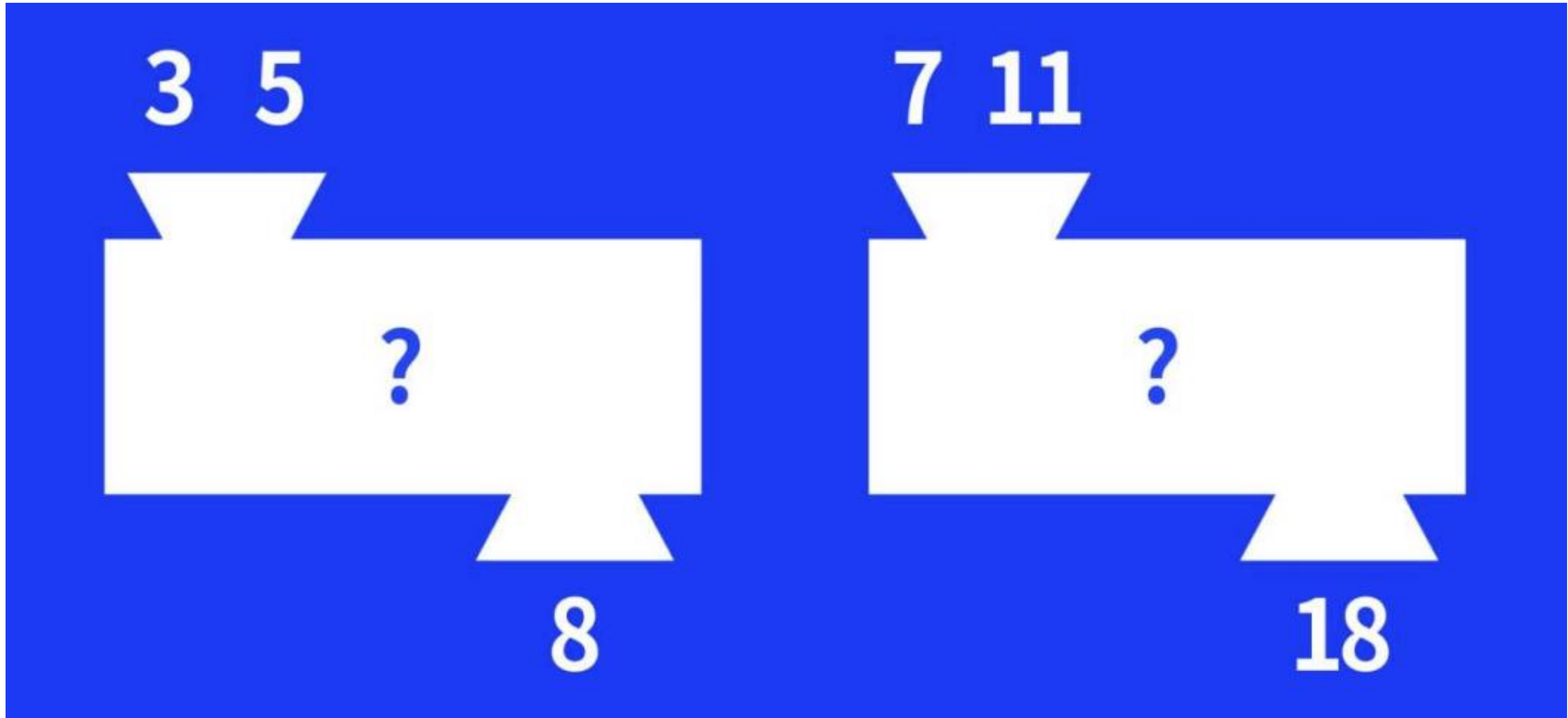
✔ Computer Science



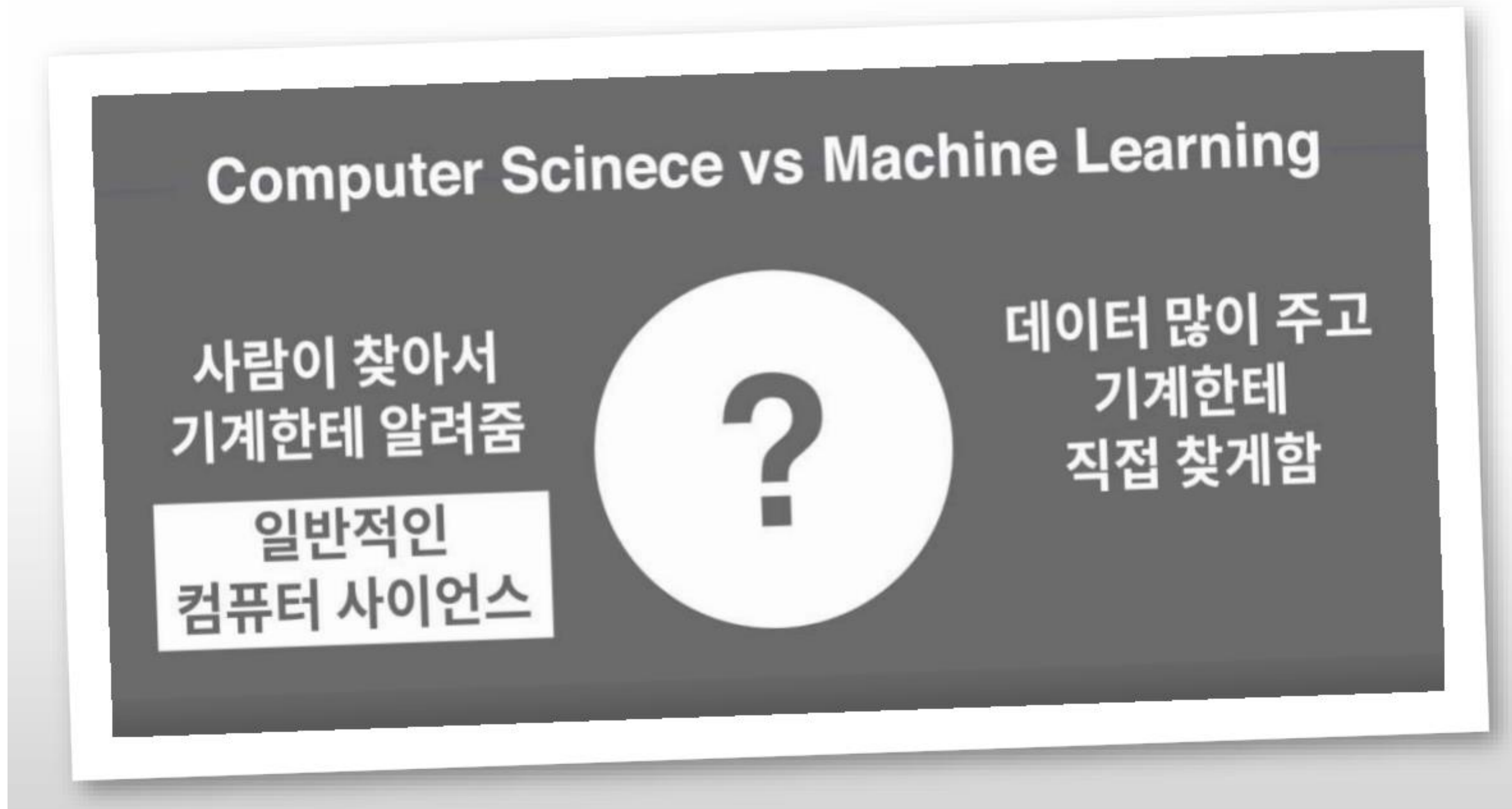
✔ Machine Learning



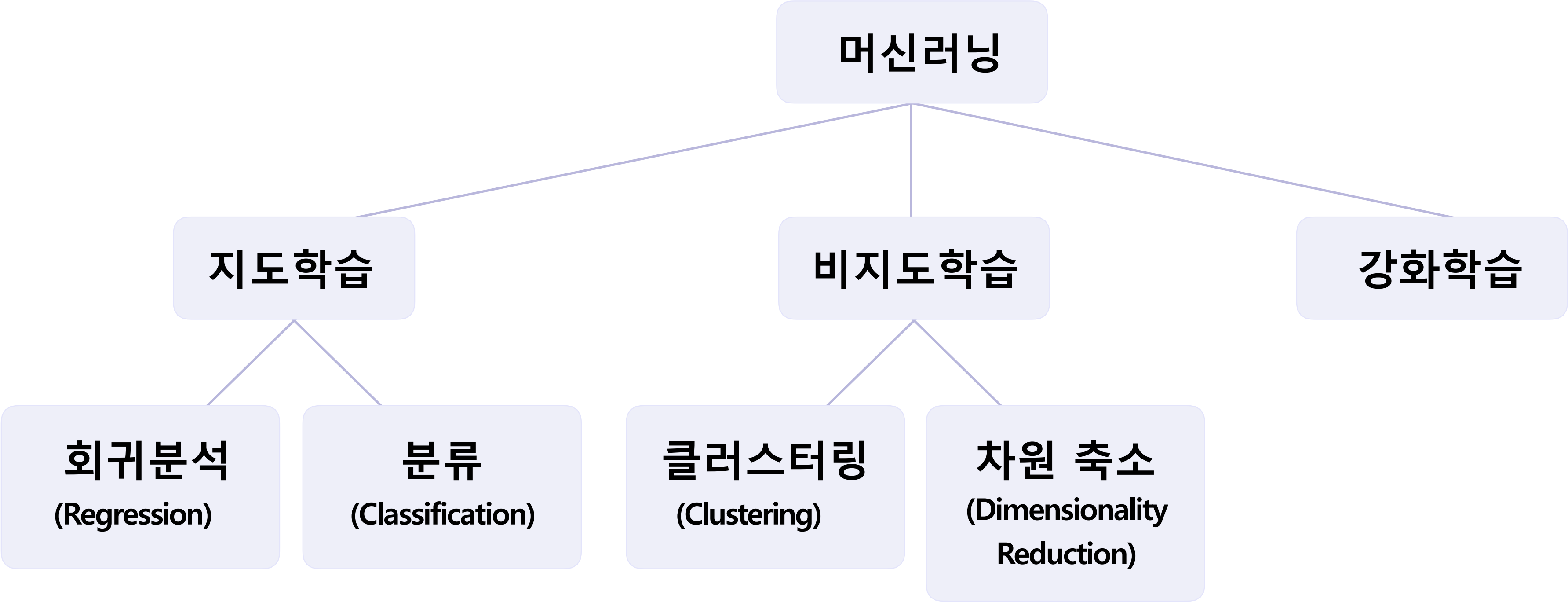
✔ Machine Learning

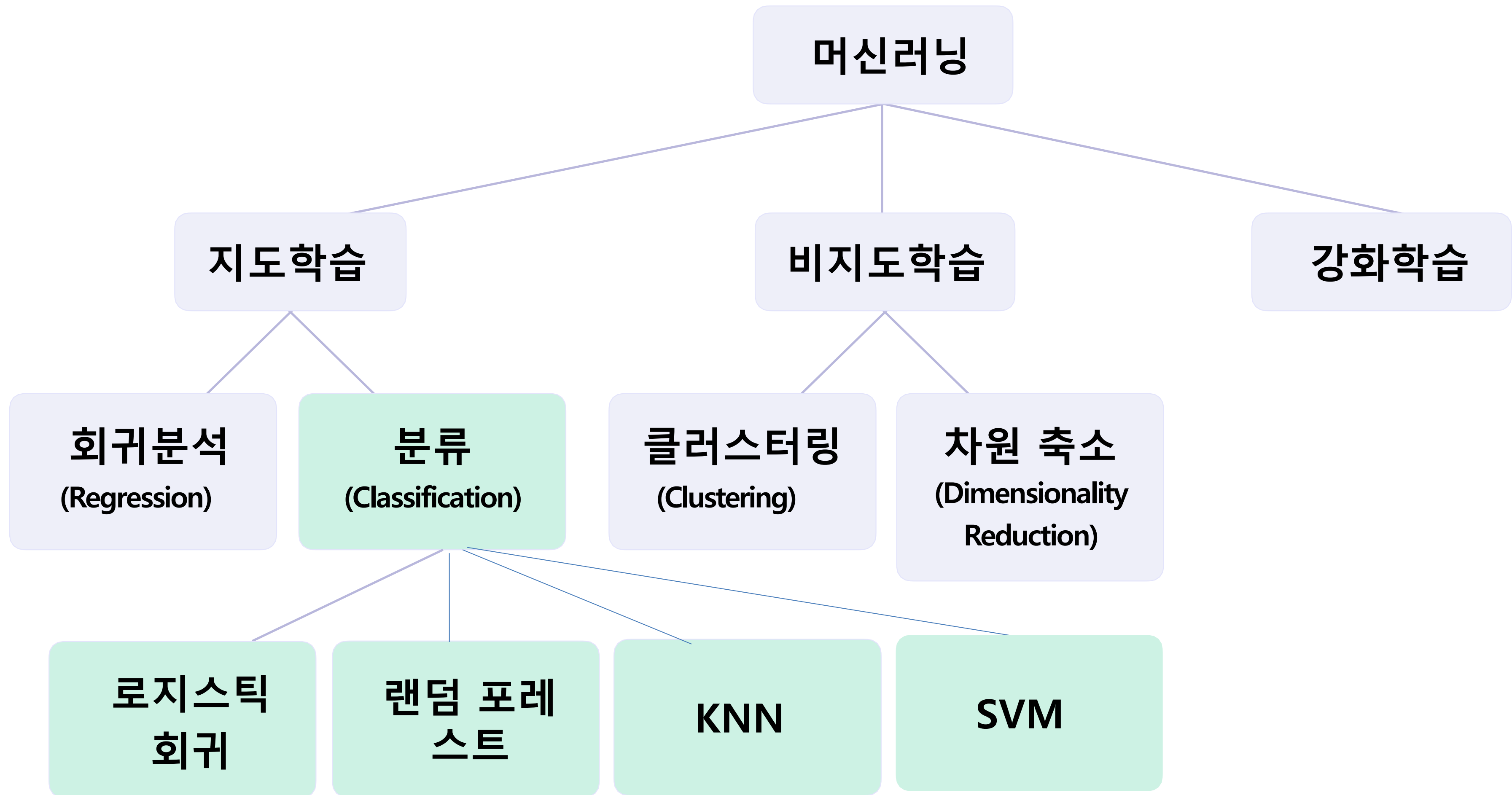


✔ Computer Science VS Machine Learning



머신러닝 유형





인공지능 학습 여부를 예측하기 위한 데이터

오늘 나온 신작 드라마 수 (x_1)	확보한 여가 시간 (x_2)	학습 여부 (y)
2	4	1
5	4	1
7	1	0
3	0	0
0	2	1
4	1	0
⋮	⋮	⋮

분 류

✔ 가정해보기

해외 여행을 준비하고 있다고 가정하기

완벽한 여행을 위해 항공 지연을 피하고자 함

기상 정보(구름양, 풍속)를 활용하여
해당 항공의 지연 여부를 예측할 수 있다면?



분 류

✔ 문제 정의와 해결 방안

• 문제 정의

데이터 : 과거 기상 정보(풍속)와 그에 따른 항공 지연 여부

목표 : 기상 정보에 따른 항공 지연 여부 예측하기

• 해결 방안

분류 알고리즘 활용

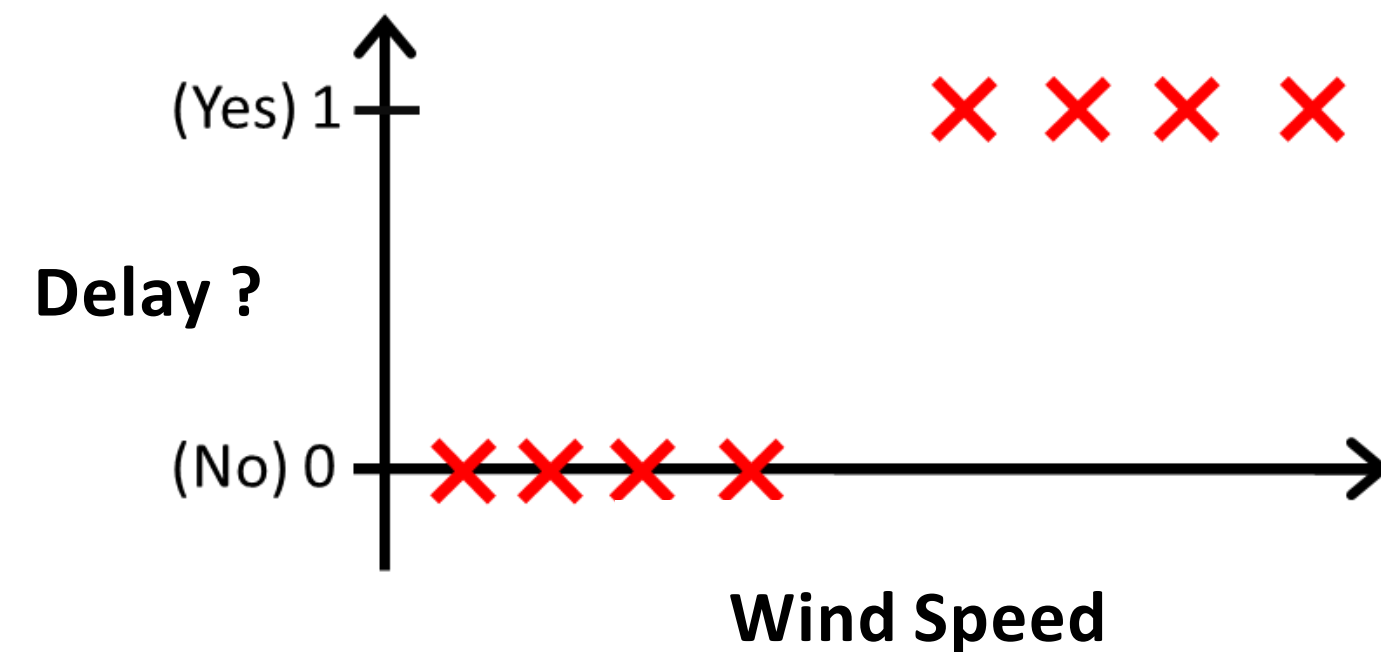
X	Y
풍속(m/s)	지연 여부
2	No
4	Yes
3	No
1	No

분 류

✓ 분류란?

주어진 입력값이 어떤 클래스에 속할지에 대한 결과값을 도출하는 알고리즘

다양한 분류 알고리즘이 존재하며, 예측 목표와 데이터 유형에 따라 적용

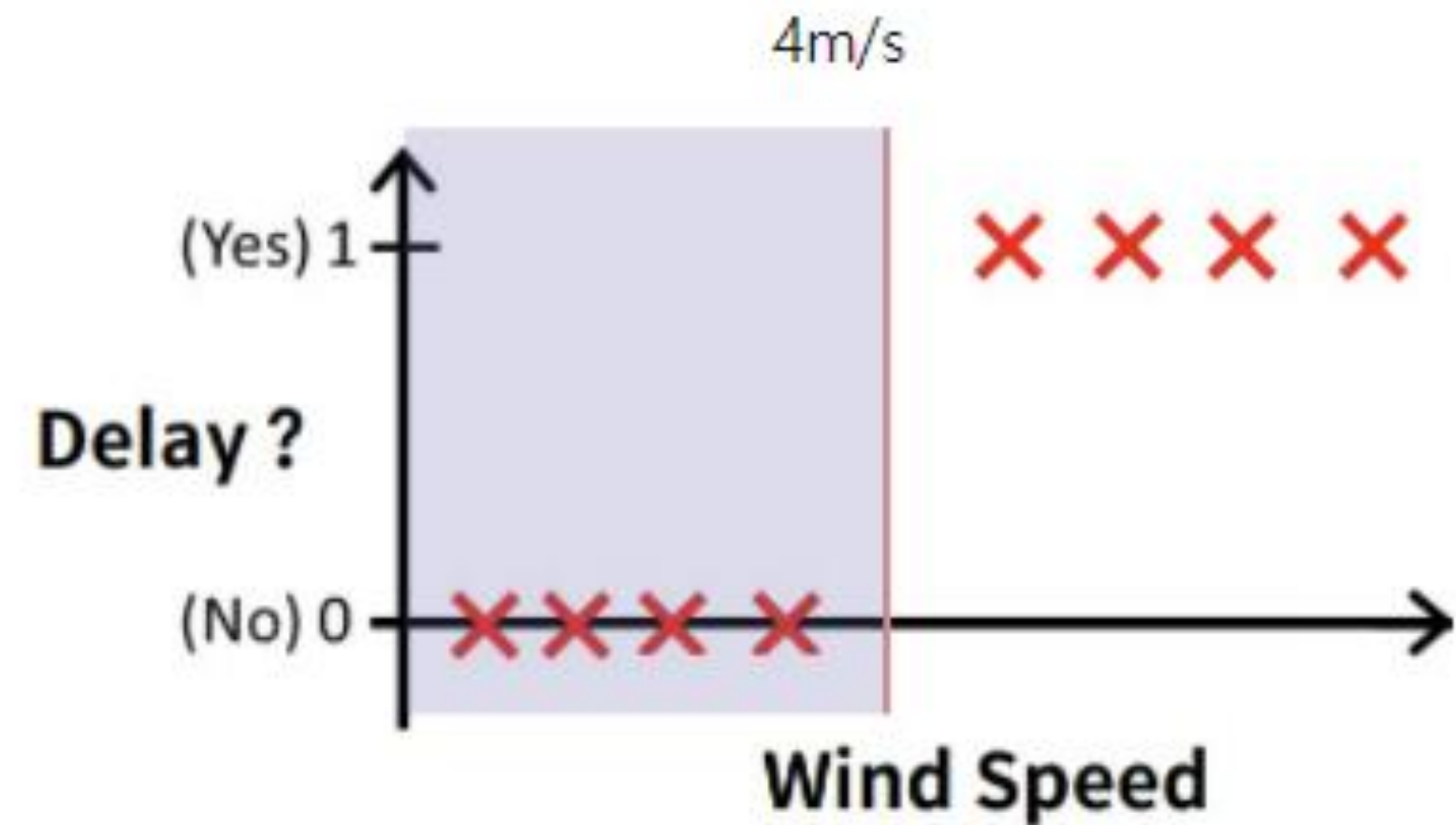
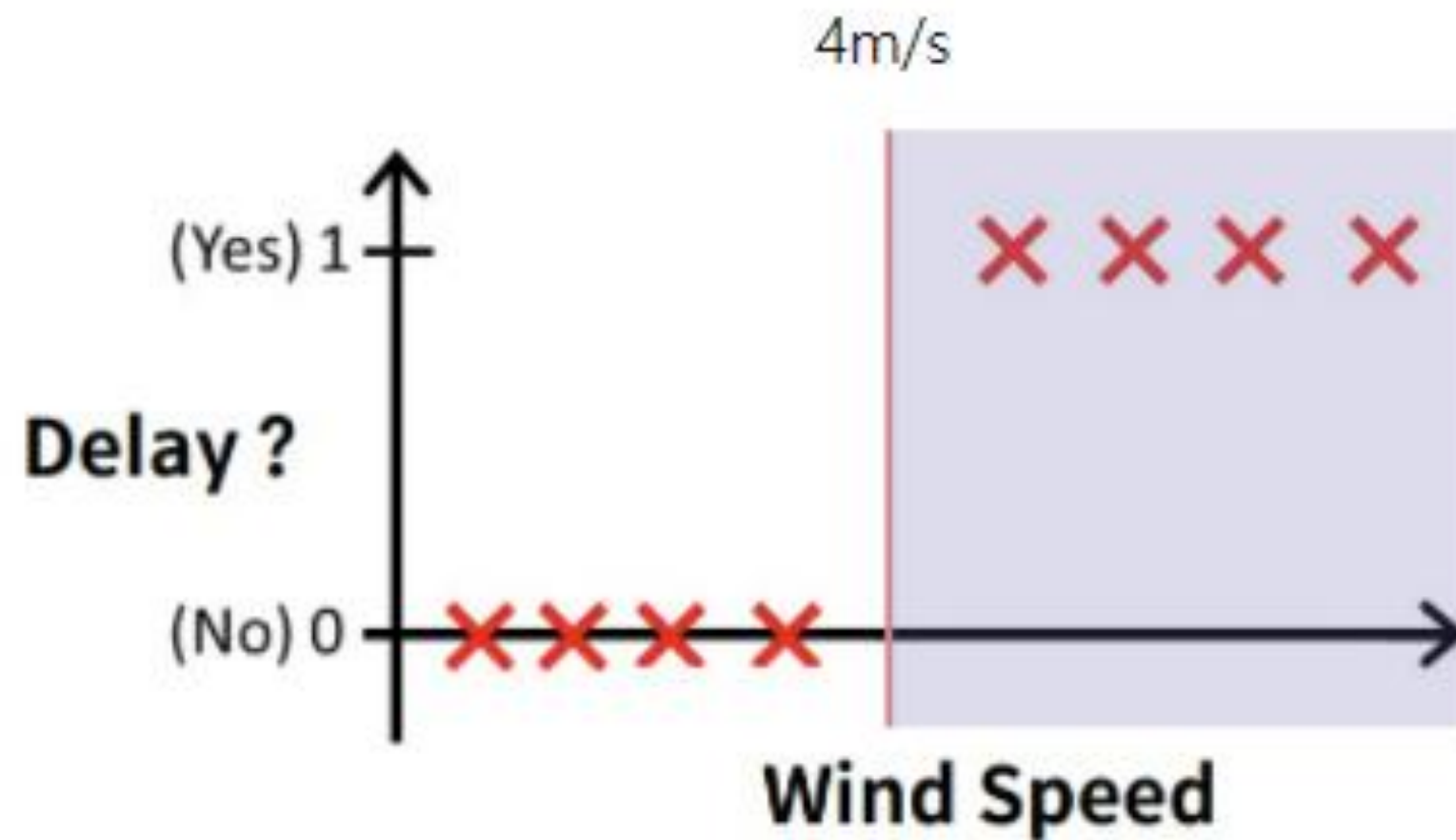


분 류

☑ 항공 지연 문제 해결하기

풍속 4m/s 를 기준으로 지연 여부를 나눠
보자

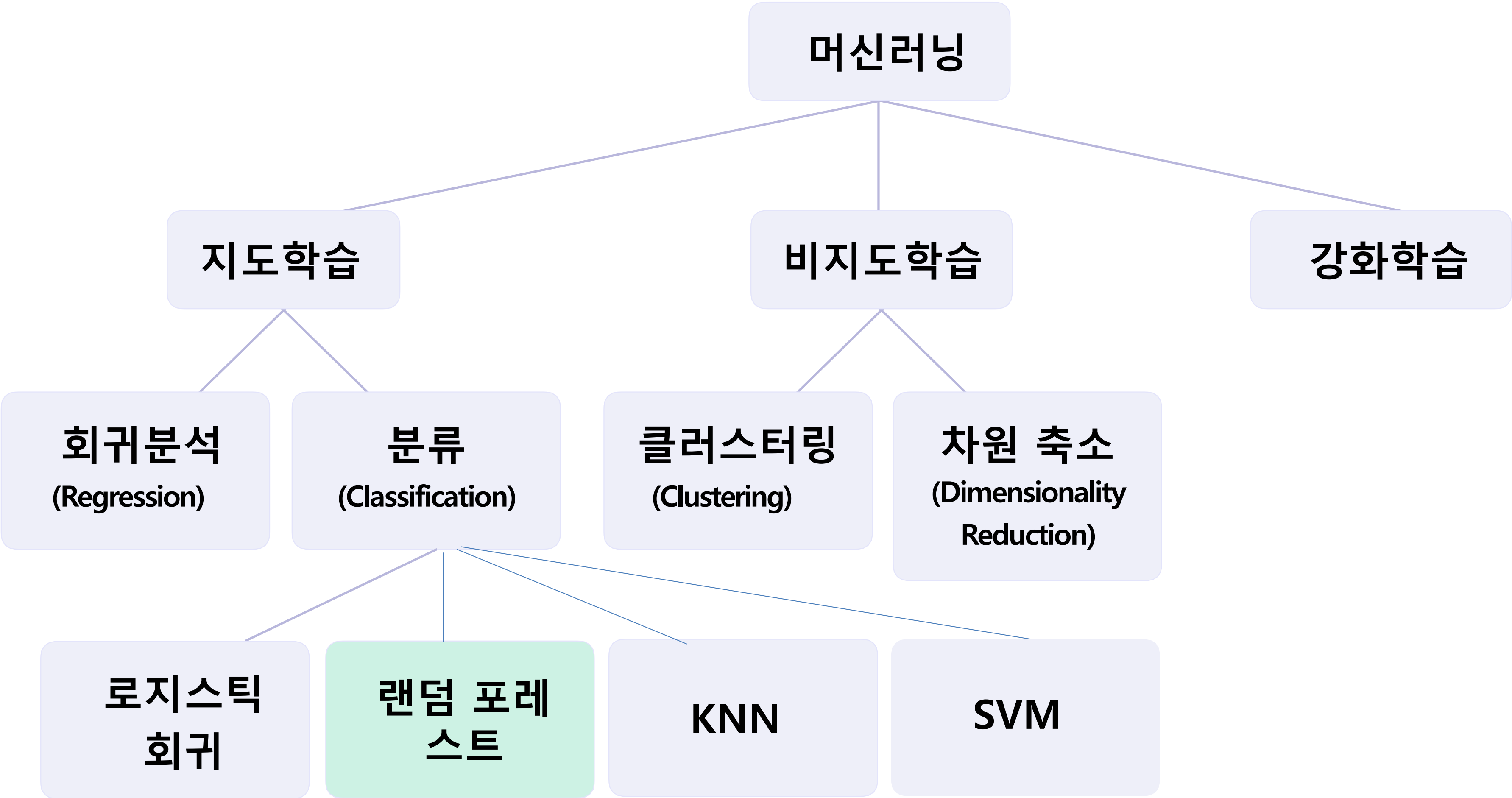
- 풍속 4m/s 보다 크면 지연
- 풍속 4m/s 보다 작으면 지연 없음



✔ 어떠한 분류 알고리즘이 있을까?

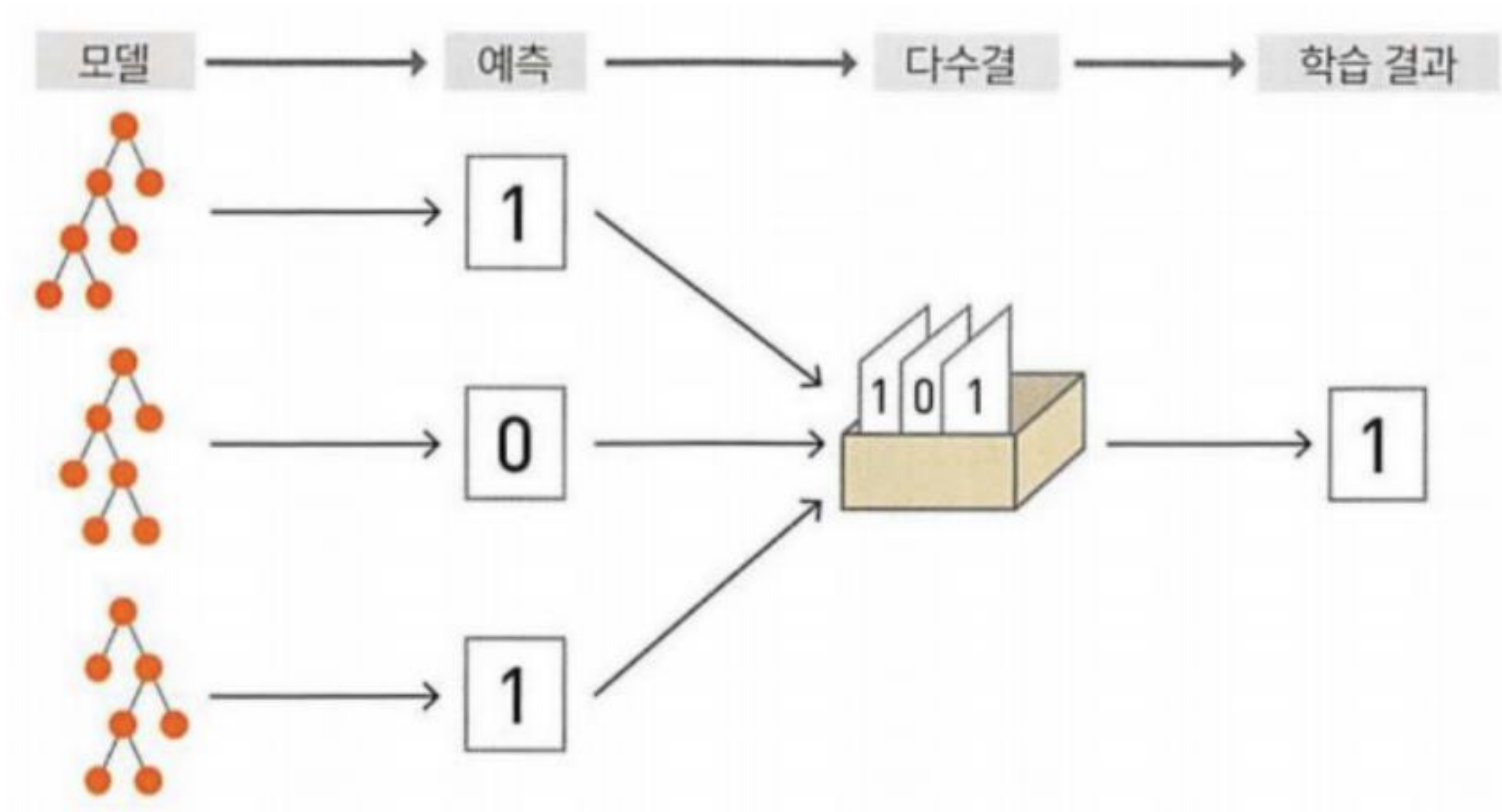
분류 문제에 다양한 머신러닝 모델을 사용하여 해결

트리 구조 기반	의사결정나무, 랜덤포레스트, ...
확률 모델 기반	나이브 베이즈 분류기, ...
결정 경계 기반	선형 분류기, 로지스틱 회귀 분류기, SVM, ...
신경망	퍼셉트론, 딥러닝 모델, ...
...	...



랜덤 포레스트

✔ 분류 문제에 사용하는 랜덤 포레스트

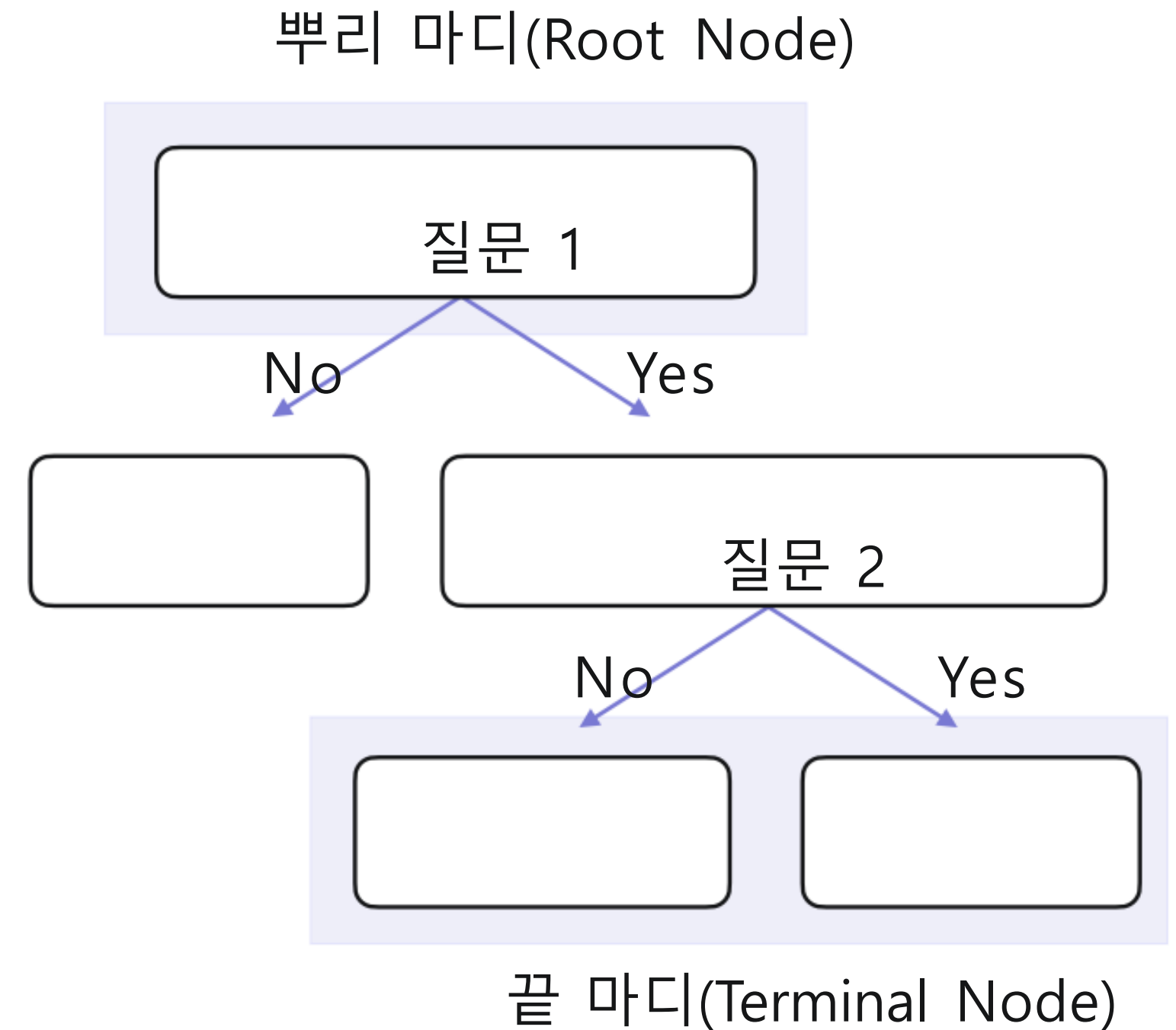


의사결정 나무 - 모델 구조

✔ 의사결정나무(Decision Tree)란

스무고개와 같이 특정 질문들을 통해
정답을 찾아가는 모델

최상단의 **뿌리 마디**에서
마지막 **끝 마디**까지 아래 방향으로 진행



의사결정 나무 - 모델 구조

☑ 의사결정나무로 이해하기

항공 지연 데이터

풍속(m/s)	지연여부
1	No
1.5	No
2.5	No
5	Yes
5.5	Yes
6.5	Yes



뿌리 마디(Root Node)

풍속 4m/s 를 기준으로 분리

풍속(m/s)	지연여부
1	No
1.5	No
2.5	No

끝 마디(Terminal Node)

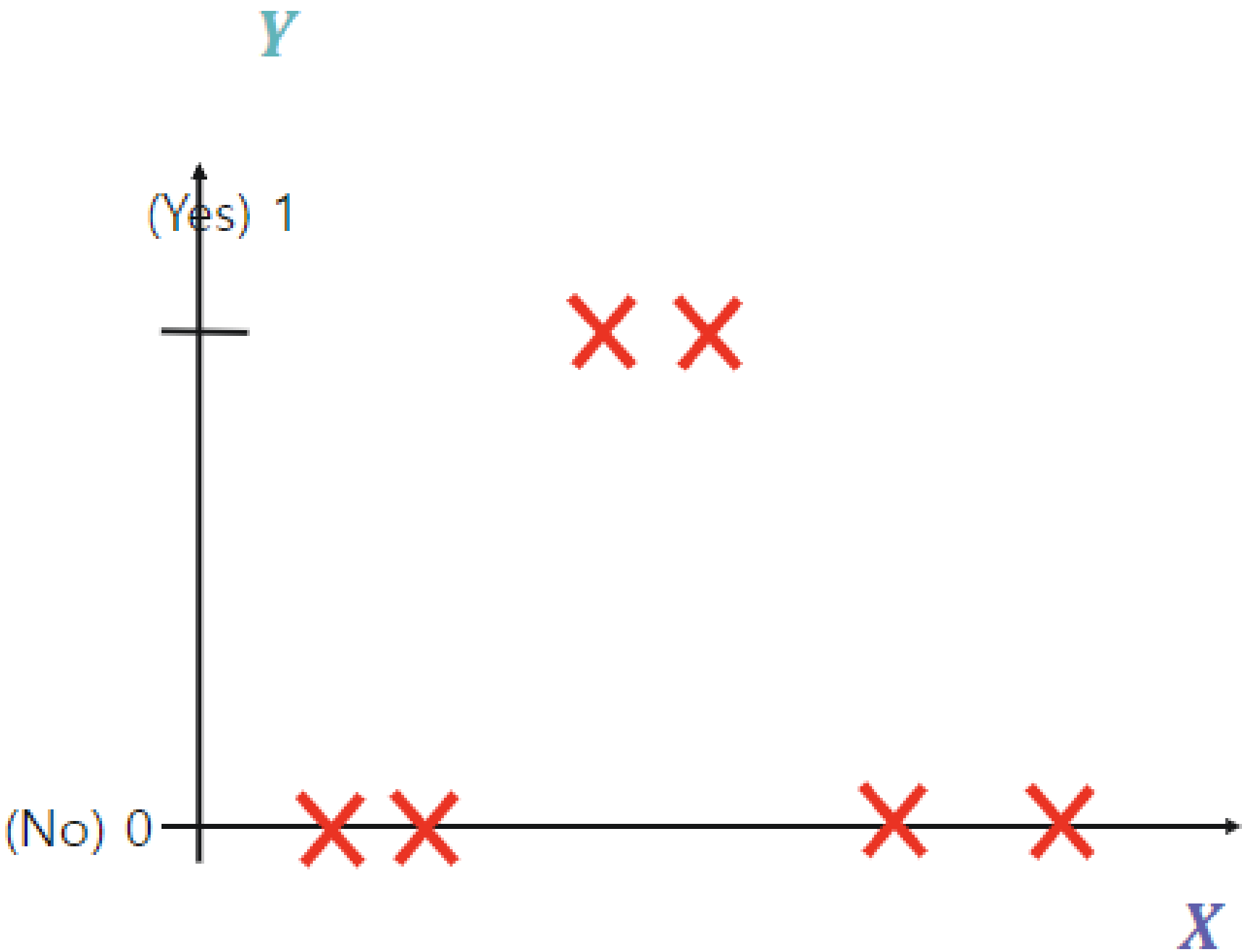
풍속(m/s)	지연여부
5	Yes
5.5	Yes
6.5	Yes

끝 마디(Terminal Node)

중간 마디 추가하기

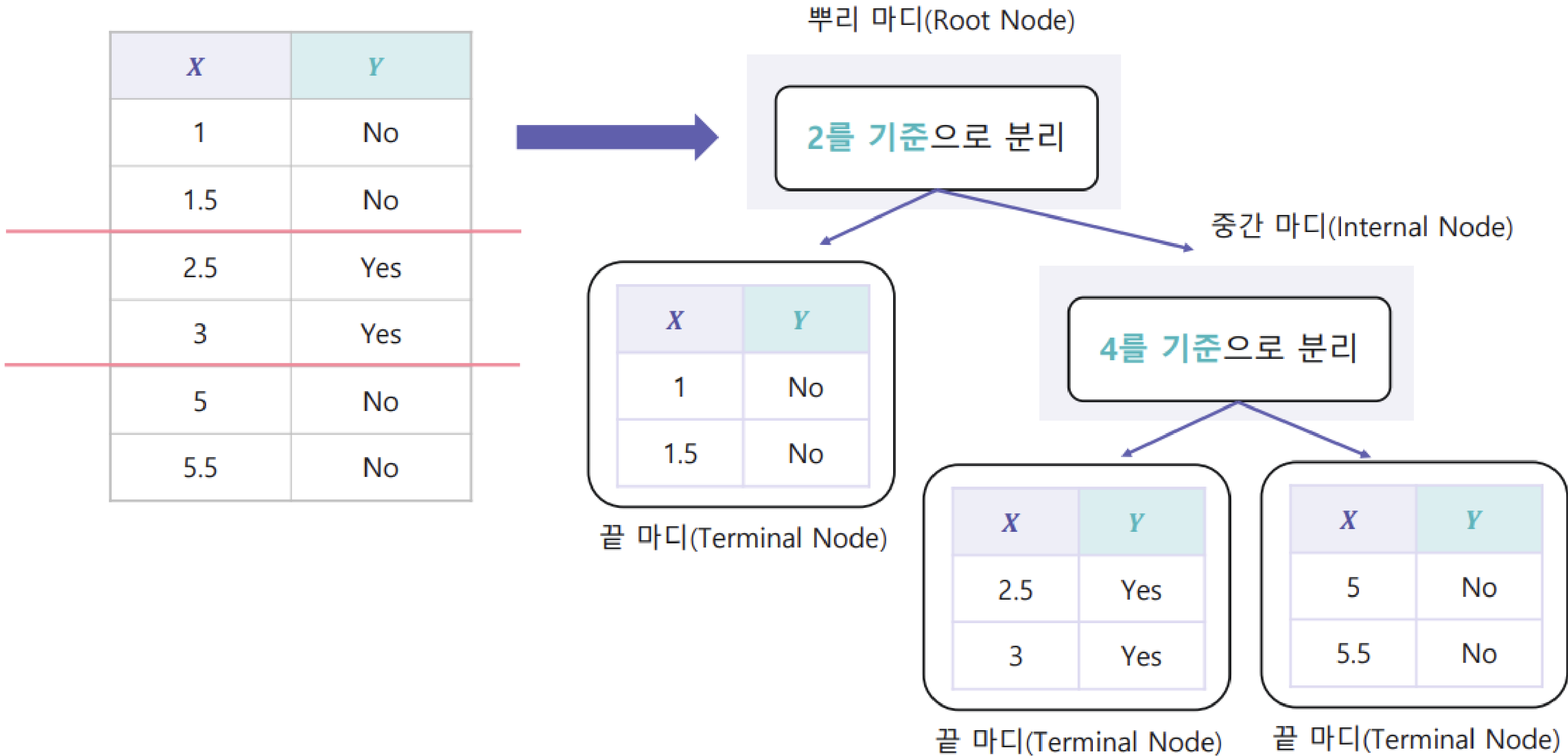
아래와 같은 데이터는 어떻게 나눠야 할까?

<i>X</i>	<i>Y</i>
1	No
1.5	No
2.5	Yes
3	Yes
5	No
5.5	No



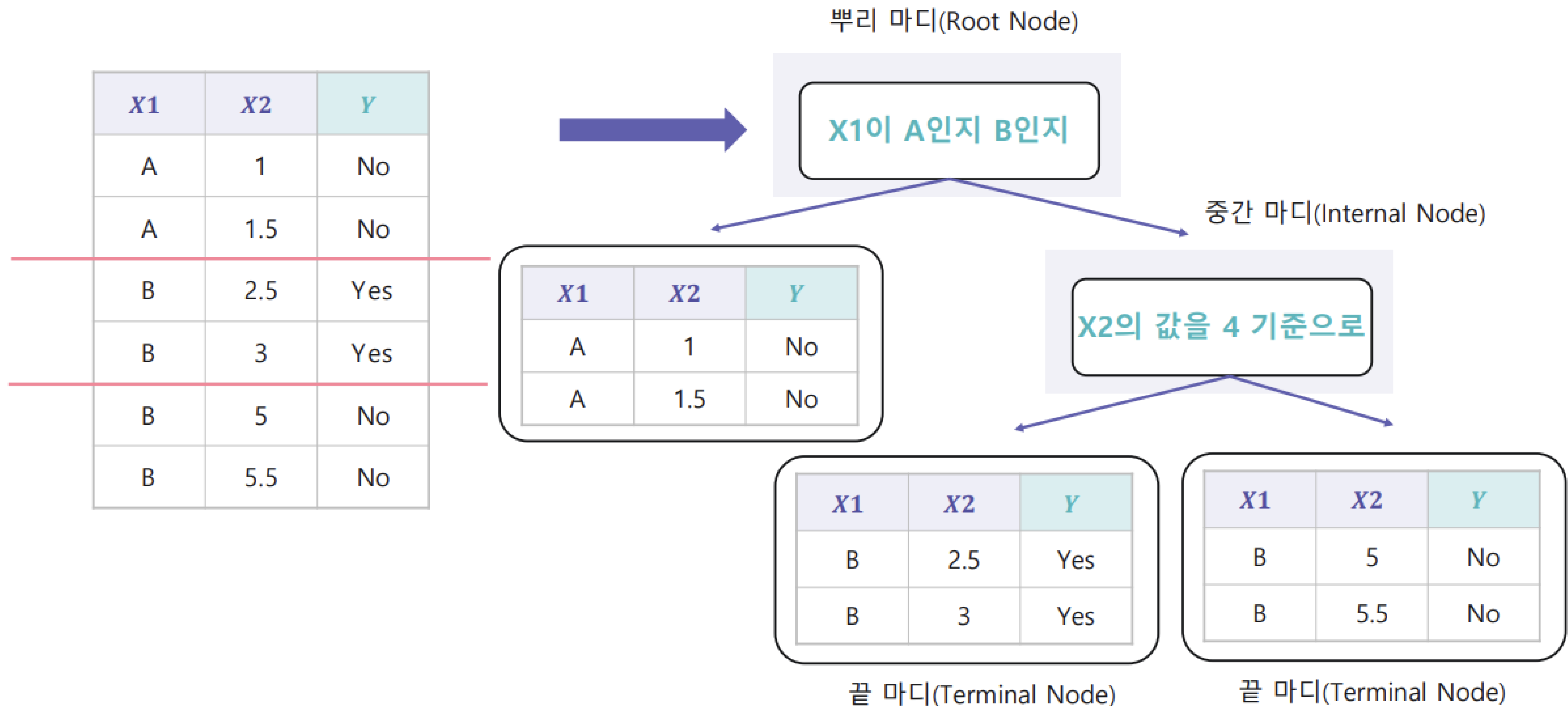
의사결정 나무 - 모델 구조

중간 마디 추가하기



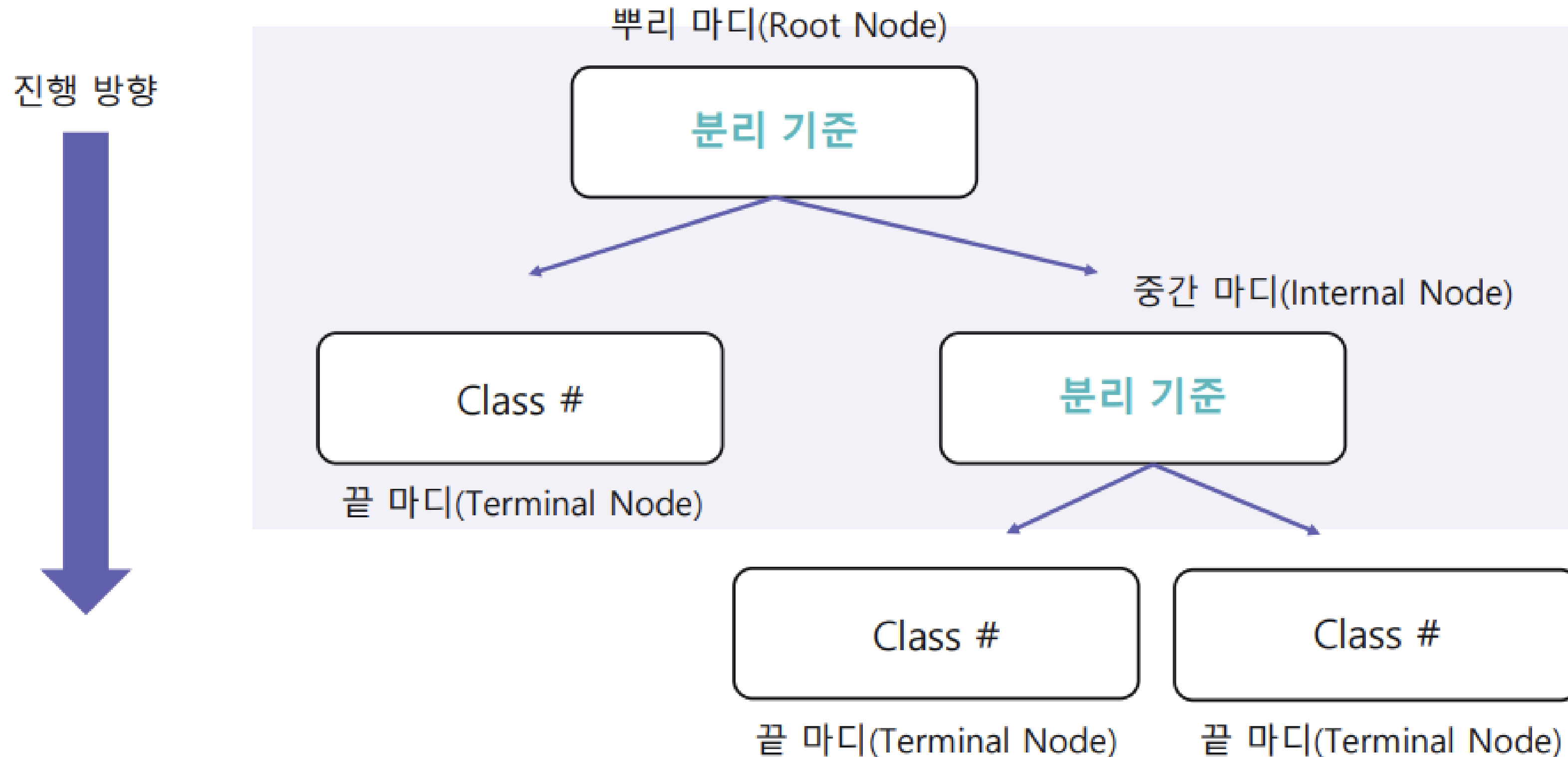
의사결정 나무 - 모델 구조

✔ 2개 이상의 feature 데이터의 경우



의사결정 나무 - 모델 구조

☑ 의사결정나무 구조 살펴보기

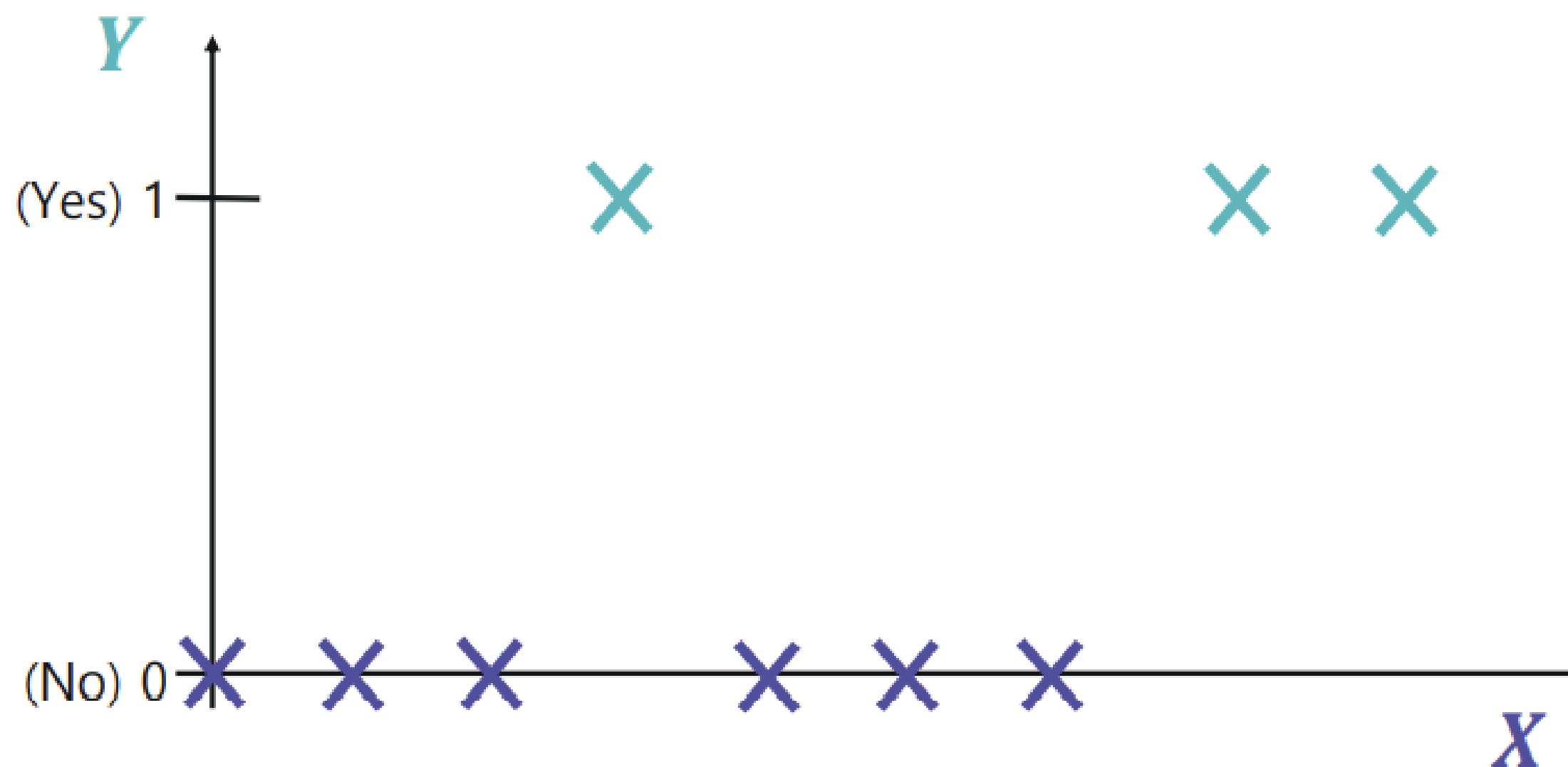


의사결정 나무 - 불순도

✔ 의사결정나무 분리 기준 알아보기

아래와 같은 데이터는 어떻게 나눠야 할까?

X	Y
1	No
2	No
3	Yes
4	No
5	No
6	No
7	Yes
8	Yes

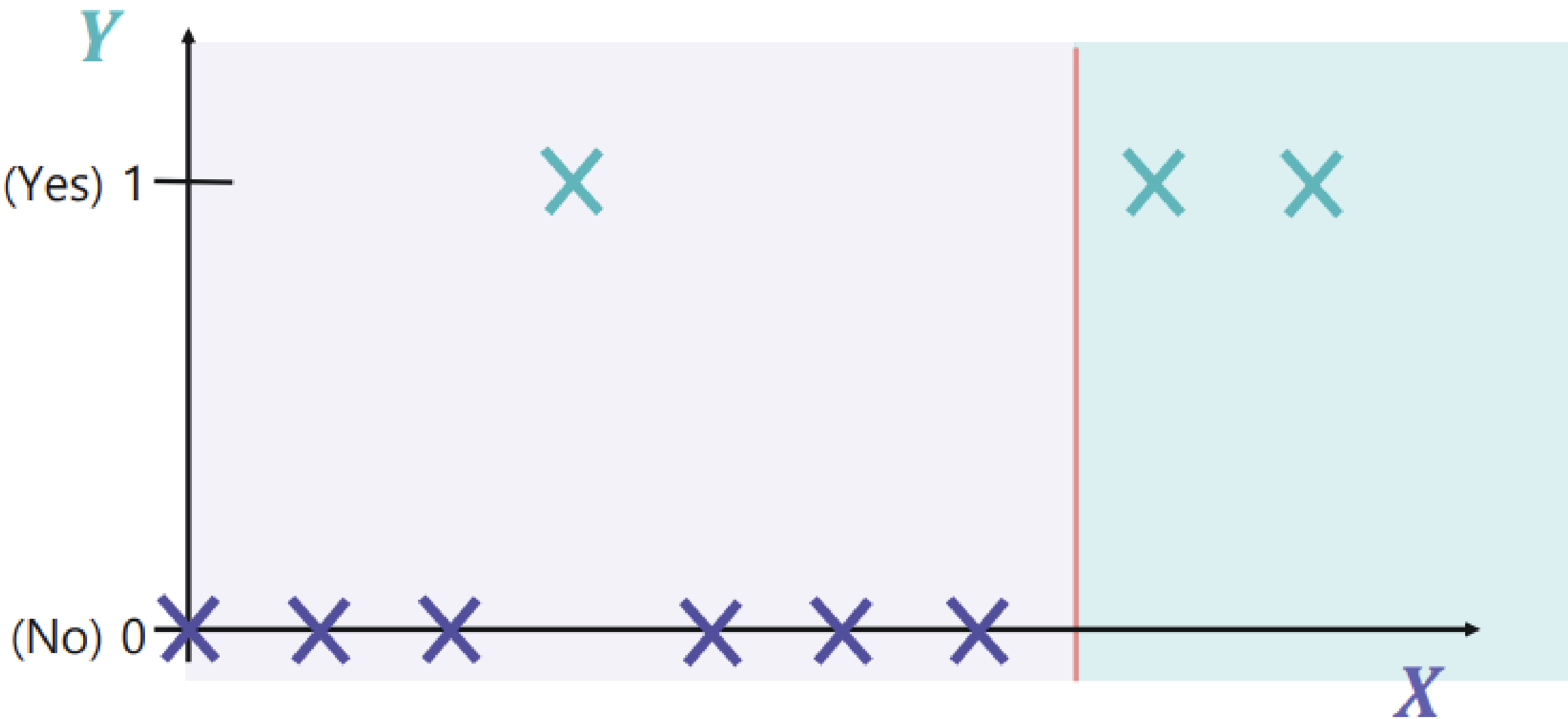


의사결정 나무 - 불순도

☑ 의사결정나무 분리 기준 알아보기

데이터의 불순도(Impurity)를 최소화하는 구역으로 나누자!

X	Y
1	No
2	No
3	Yes
4	No
5	No
6	No
7	Yes
8	Yes



의사결정 나무 - 불순도

✓ 불순도 (Impurity)

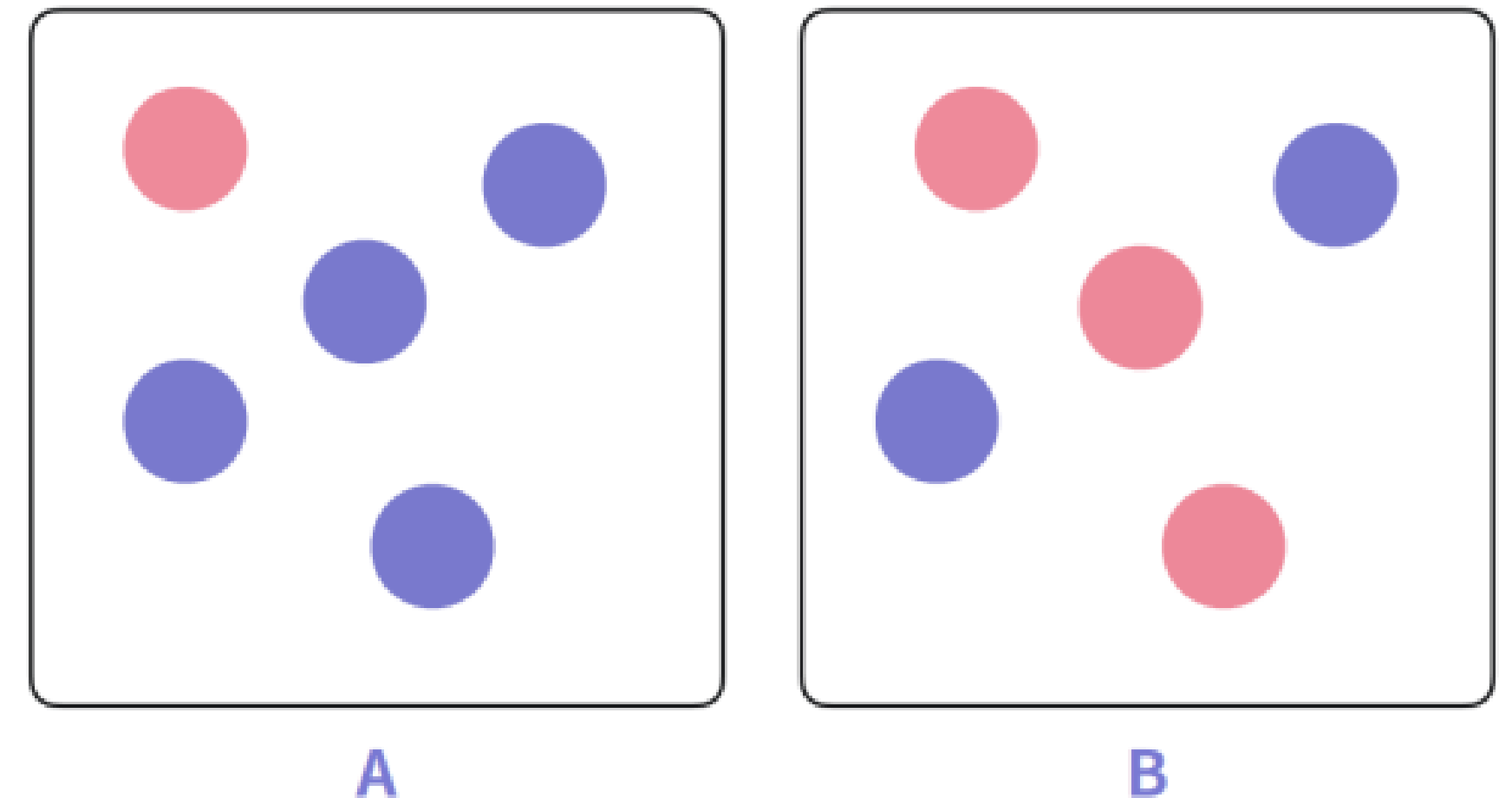
- 불순도

다른 데이터가 섞여 있는 정도

- 데이터 셋 A와 B 중 불순도가 더 낮은 것은?

정답은 A!

데이터의 개수가 적기 때문에 눈으로 확인
그렇다면 수많은 데이터가 존재할 때
불순도는 어떻게 측정할 수 있을까?



의사결정 나무 – 불순도

✓ 불순도 측정 방법, 지니 불순도(Gini Impurity)

- 지니 계수(Gini Index)

해당 구역 안에서 특정 클래스에 속하는 데이터의 비율을 모두 제외한 값
즉, **다양성을 계산**하는 방법

- 지니 불순도(Gini Impurity)

$$\text{Gini Index} = 1 - (\text{yes의 확률})^2 - (\text{no의 확률})^2$$

$$\text{Gini Impurity} = \frac{n_1}{N} \text{Gini}_1 + \frac{n_2}{N} \text{Gini}_2$$

n_i : i번째 자식 마디의 데이터 개수

N: 부모 마디의 데이터 개수

의사결정 나무 - 불순도

✔ 지니 불순도(Gini Impurity) 계산하기

$$\text{Gini Index} = 1 - (\text{yes의 확률})^2 - (\text{no의 확률})^2$$

$$\text{Gini Impurity} = \frac{n_1}{N} \text{Gini}_1 + \frac{n_2}{N} \text{Gini}_2$$

<i>X</i>	<i>Y</i>
1	No
2	No
3	Yes
4	No
5	No
6	No
7	Yes
8	Yes

<i>Yes</i>	<i>No</i>	Gini index
0	2	$1 - \left(\frac{0}{2}\right)^2 - \left(\frac{2}{2}\right)^2 = 0$
3	3	$1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2 = \frac{1}{2}$

$$\text{Gini Impurity} = \frac{2}{8} \times 0 + \frac{6}{8} \times \frac{1}{2} = \frac{3}{8}$$

의사결정 나무 - 불순도

✔ 지니 불순도(Gini Impurity) 계산하기

$$\text{Gini Index} = 1 - (\text{yes의 확률})^2 - (\text{no의 확률})^2$$

$$\text{Gini Impurity} = \frac{n_1}{N} \text{Gini}_1 + \frac{n_2}{N} \text{Gini}_2$$

X	Y
1	No
2	No
3	Yes
4	No
5	No
6	No
7	Yes
8	Yes

Gini Impurity = 0.125

Gini Impurity = 0.457

Gini Impurity = 0.437

Gini Impurity = 0.397

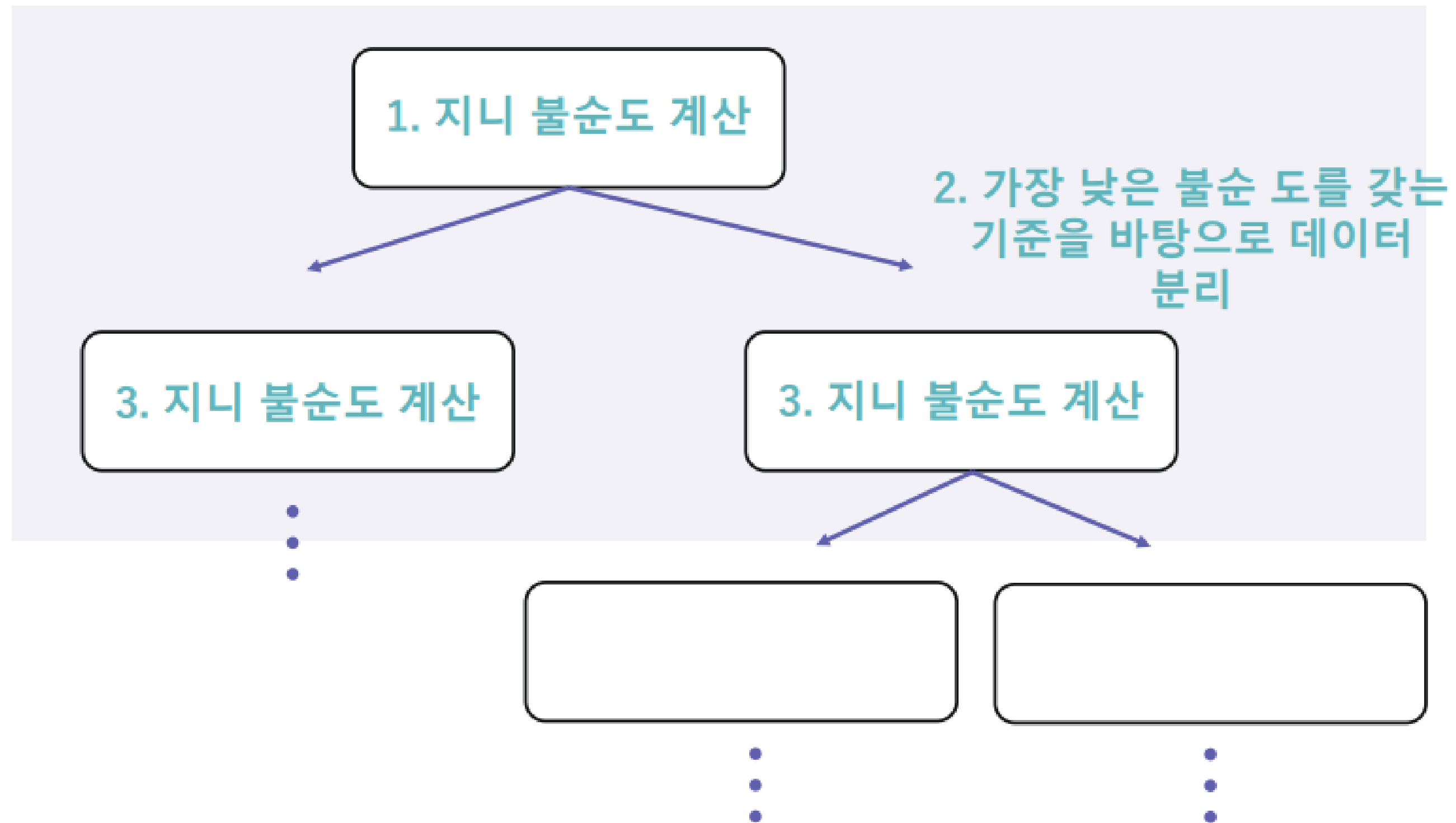
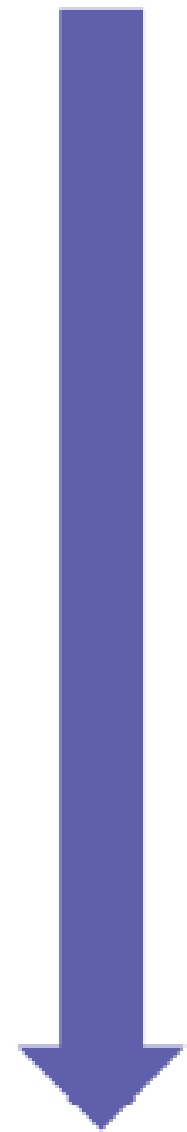
Gini Impurity = 0.069

가장 낮은 Gini 불순도를
갖는 기준을 선택

의사결정 나무 - 불순도

✔ 지니 불순도(Gini Impurity) 적용하기

진행 방향



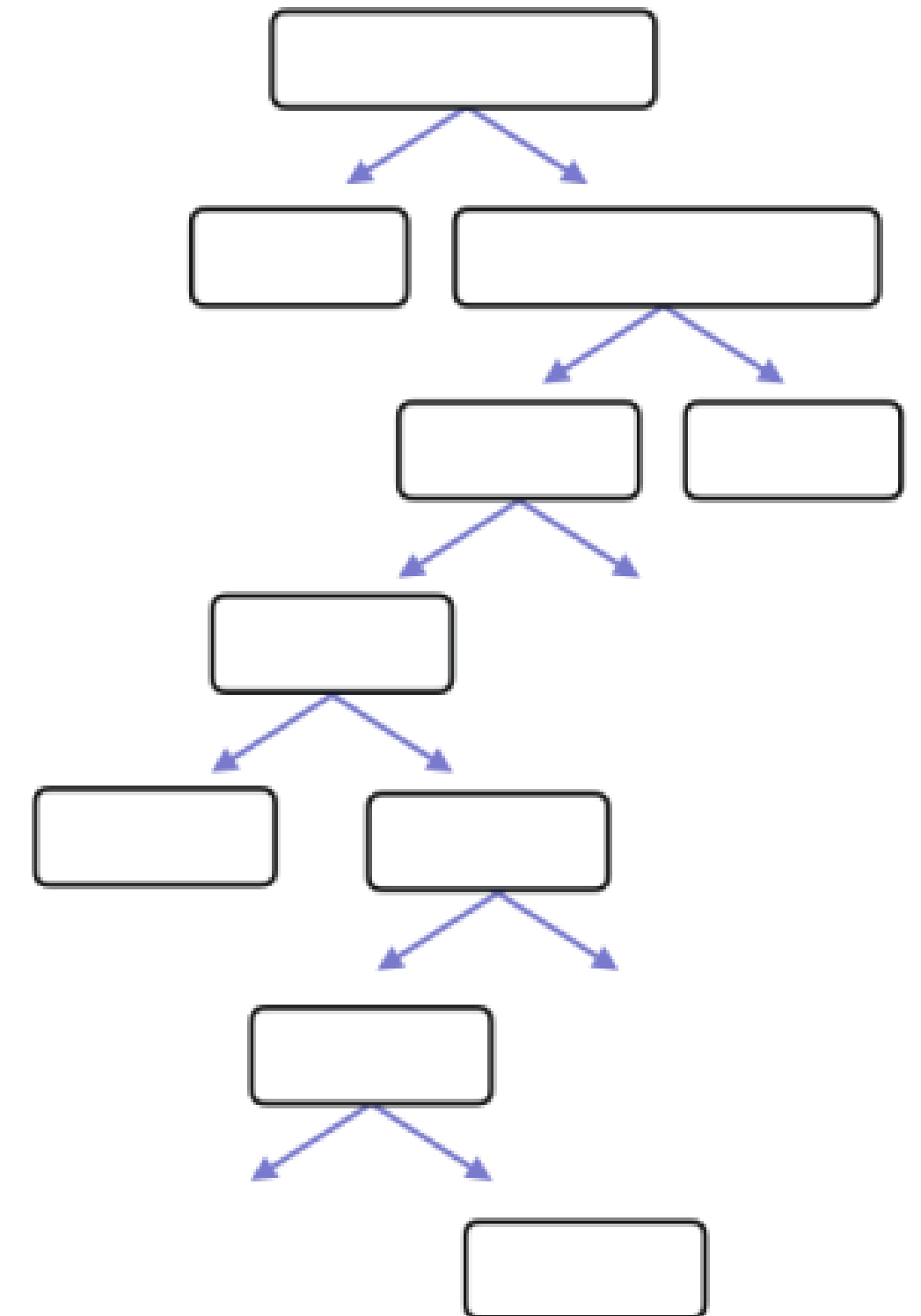
의사결정 나무 - 불순도

✔ 의사결정나무의 깊이의 trade-off

의사결정나무의 깊이가 깊어질 수록 세분화해서 나눌 수 있음

하지만 너무 깊은 모델은 과적합을 야기할 수 있음

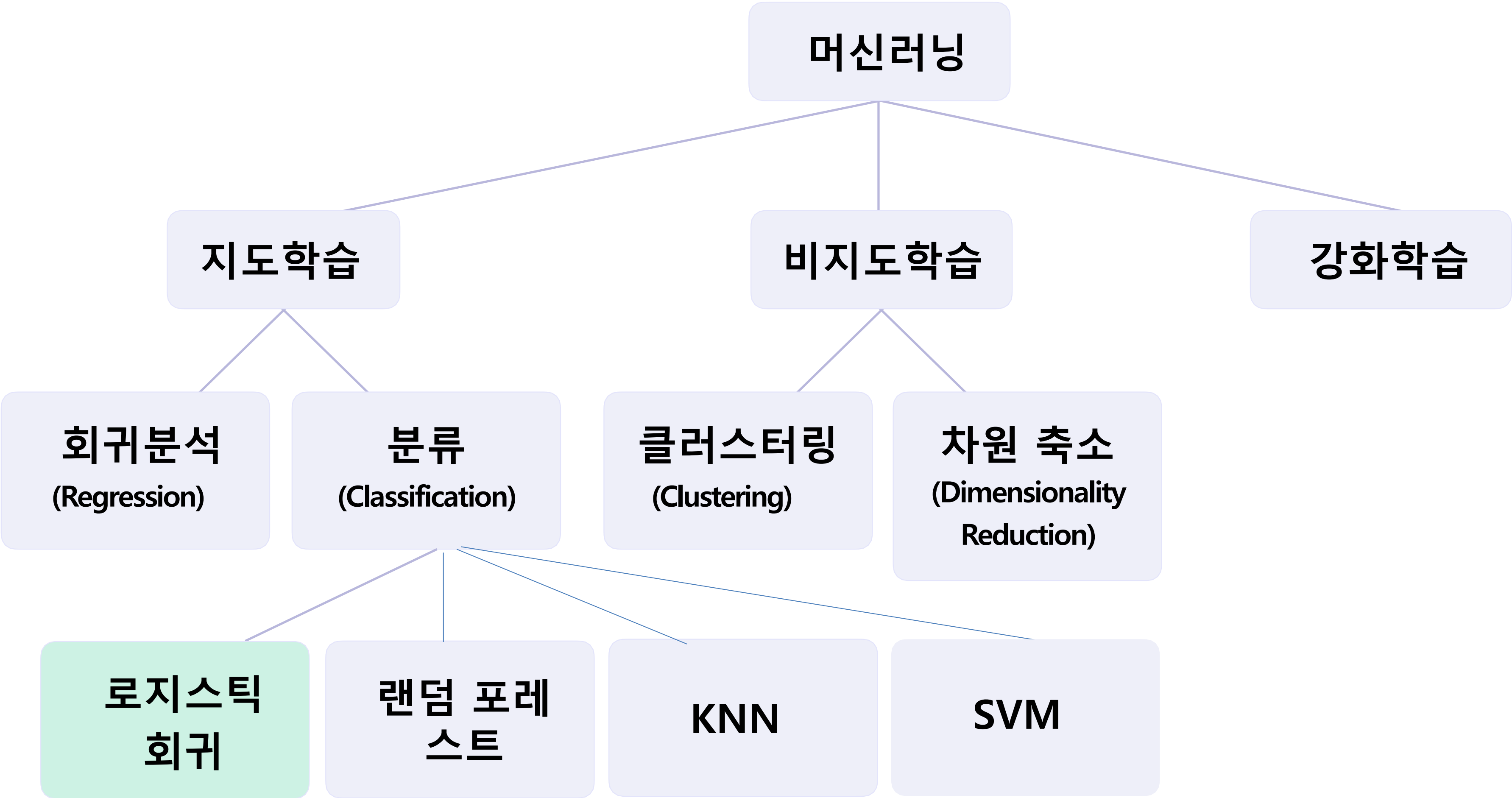
⇒ 데이터에 따라 다를 수 있지만 **너무 깊은 모델은 지양**



의사결정나무의 특징

✔ 의사결정나무의 깊이의 trade-off

- 결과가 직관적이며, 해석하기 쉬움
- 나무 깊이가 깊어질수록 과적합(Overfitting) 문제 발생 가능성이 매우 높음
- 학습이 끝난 트리의 작업 속도가 매우 빠르다



로지스틱 회귀

✓ 분류 문제에 회귀 알고리즘 적용하기

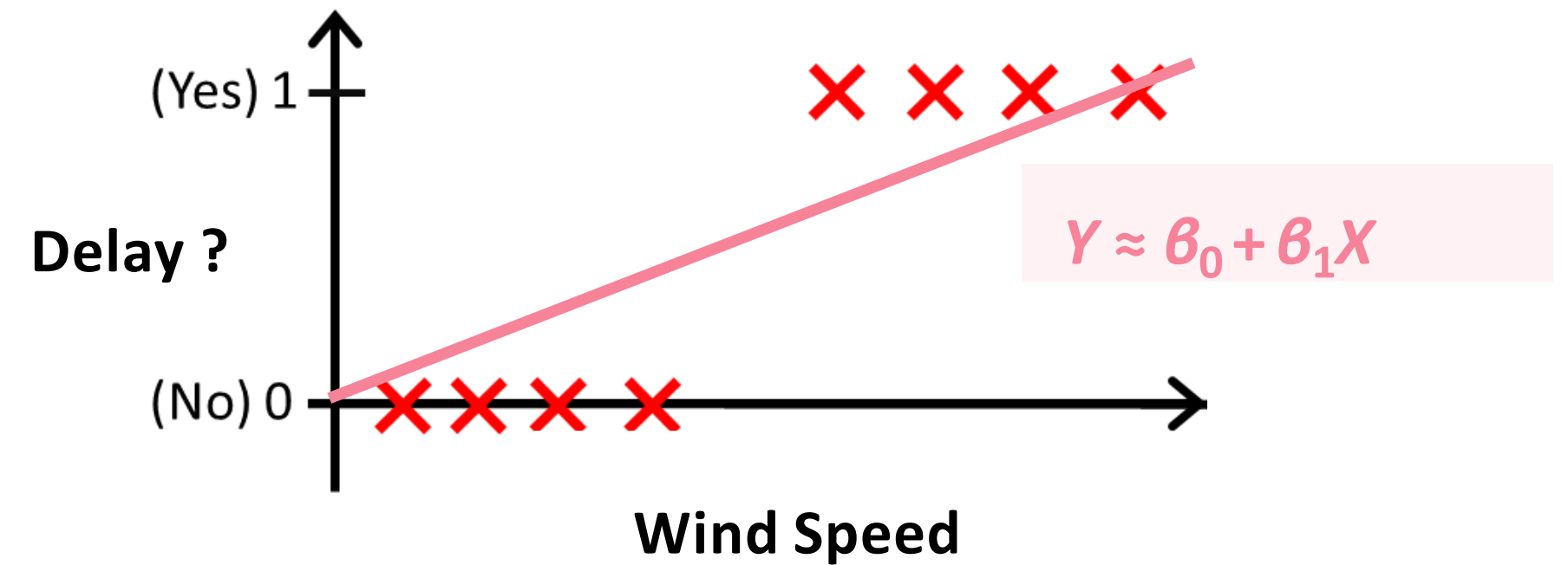
일반적인 회귀 알고리즘은 분류 문제에 그대로 사용할 수 없다!

Why?

선형 회귀는 $-\infty \sim +\infty$ 의 값을 가질 수 있음

Q. 우리의 목표는

지연 여부 판별인데 결과값이 1000이라면?



로지스틱 회귀

✔ 그렇다면 어떻게 해야 할까?

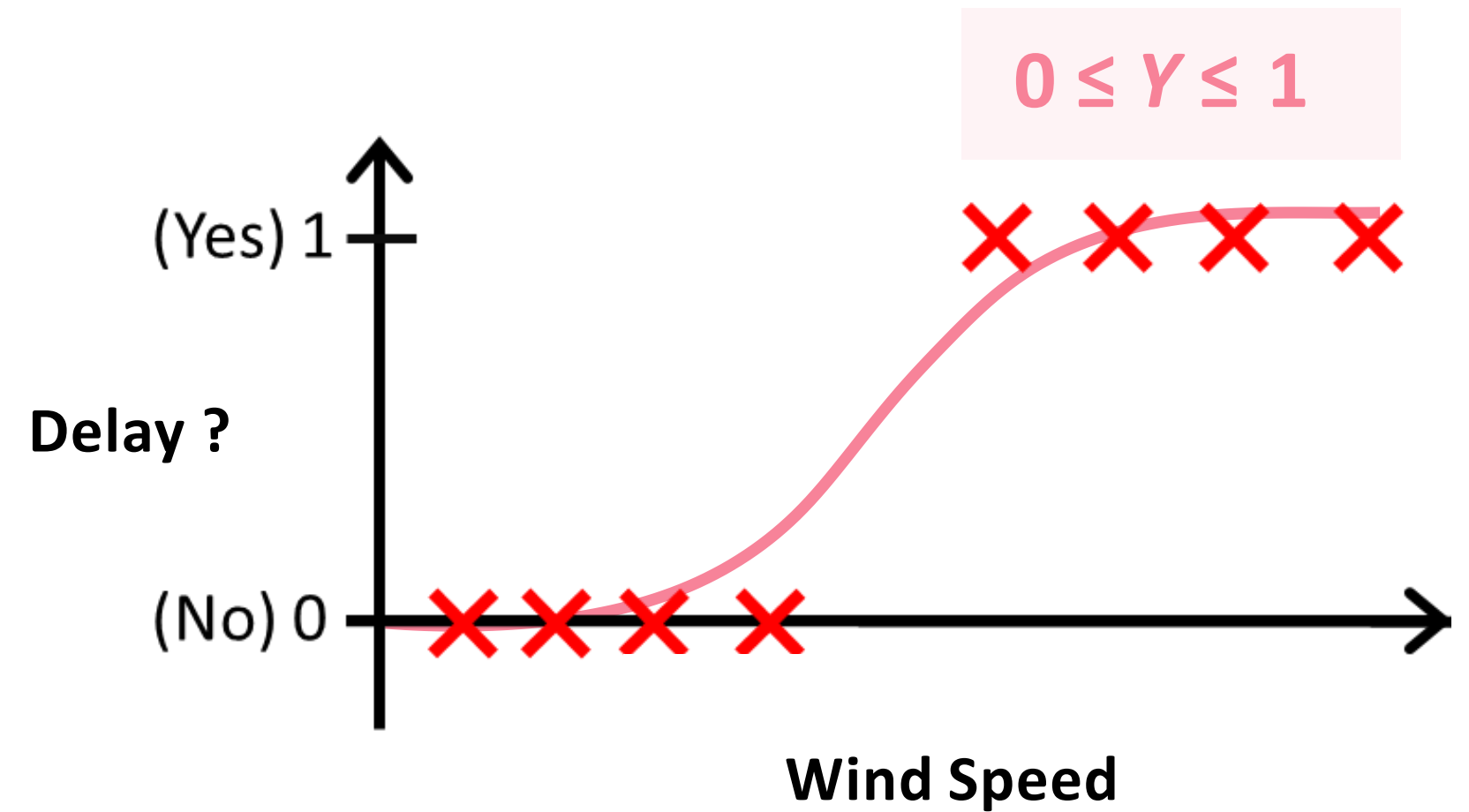
해당 클래스에 속할 확률인

0 또는 1 사이의 값만 내보낼 수 있도록
선형 회귀 알고리즘 수정하기

이처럼 분류 문제에 적용하기 위
해

출력값의 범위를 수정한 회귀를

로지스틱 회귀(Logistic Regression)라고 함

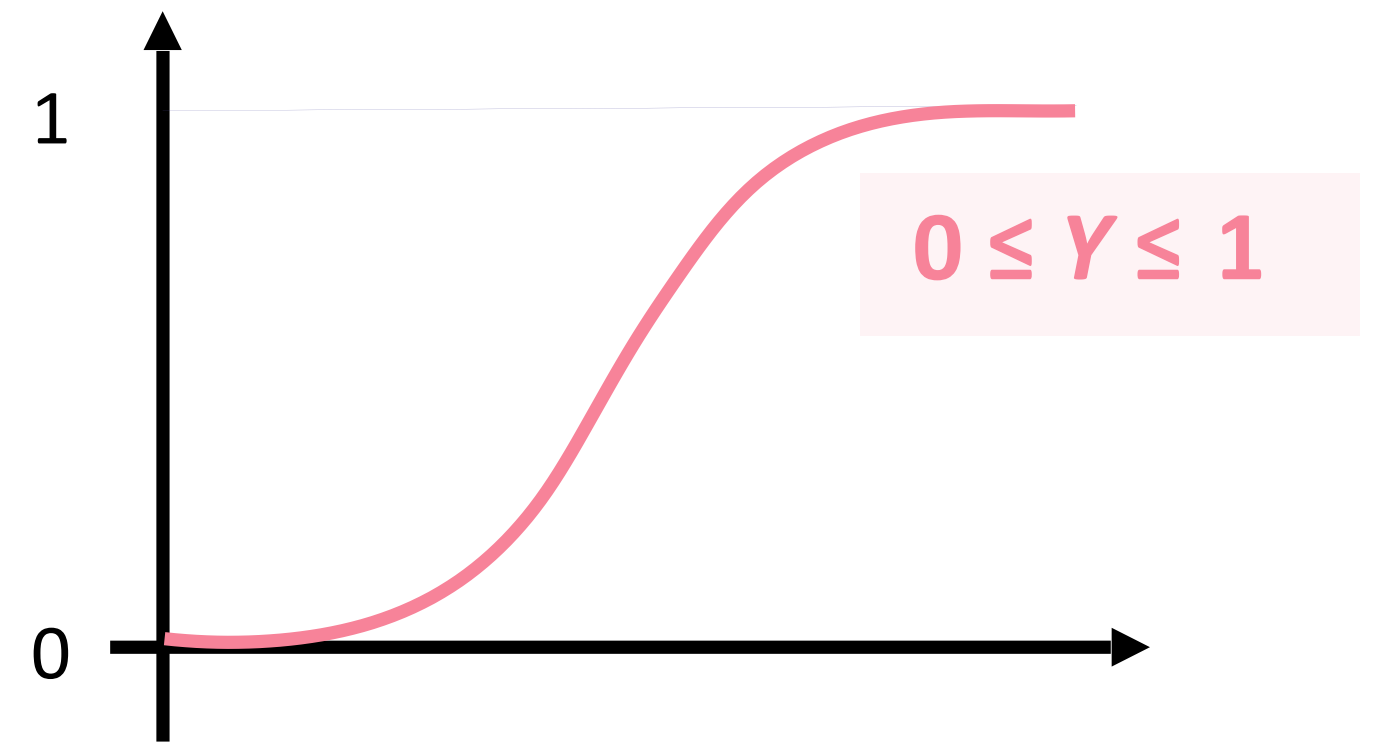


로지스틱 회귀

✓ 분류 문제를 위한 회귀, Logistic Regression

이진 분류(Binary Classification) 문제를 해결
하기 위한 모델

최소값 0, 최대값 1로 결과값을 수렴시키기 위해
Sigmoid(logistic) 함수 사용



로지스틱 회귀

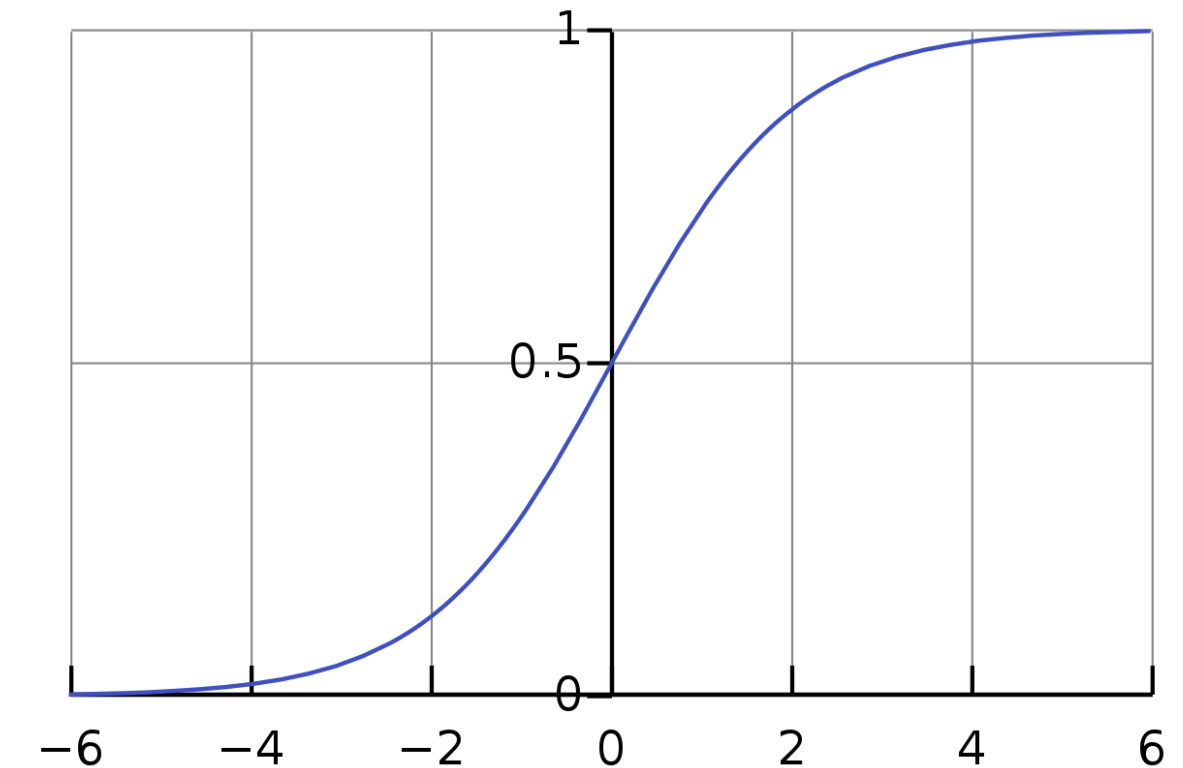
✔ Sigmoid (logistic) 함수

$$g(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{1 + e^x}$$

s자형 곡선을 갖는 함수

값이 커질 경우 $g(x)$ 값은 점점 1에 수렴하고, 값이 작아질 경우 $g(x)$ 값은 점점 0에 수렴함

Sigmoid (logistic) function



로지스틱 회귀

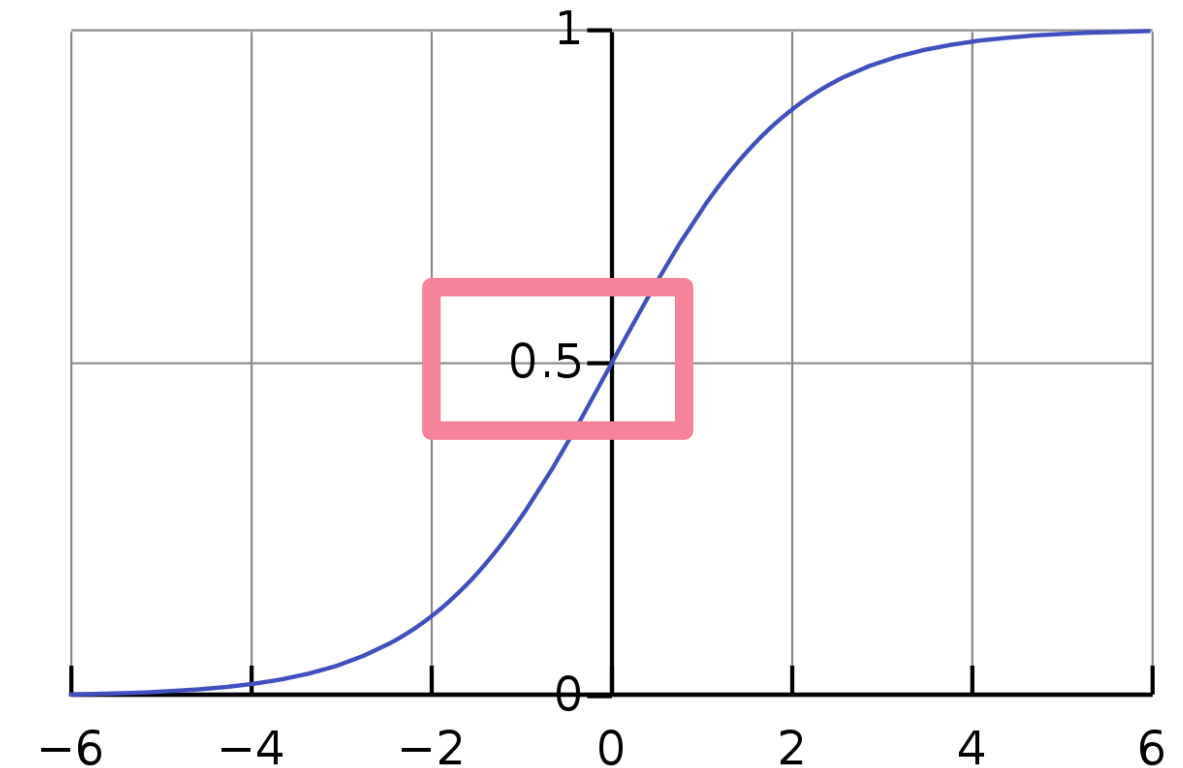
✓ 확률 결과값 판별 방법, 결정 경계(Decision Boundary)

결정 경계란, 데이터를 분류하는 기준값을 의미함

그렇다면, 출력된 확률값을
어떠한 기준으로 클래스에 속한다고 판별해야 할까?

일반적으로 출력값(확률) **0.5**를 기준으로 판별

Sigmoid (logistic) function

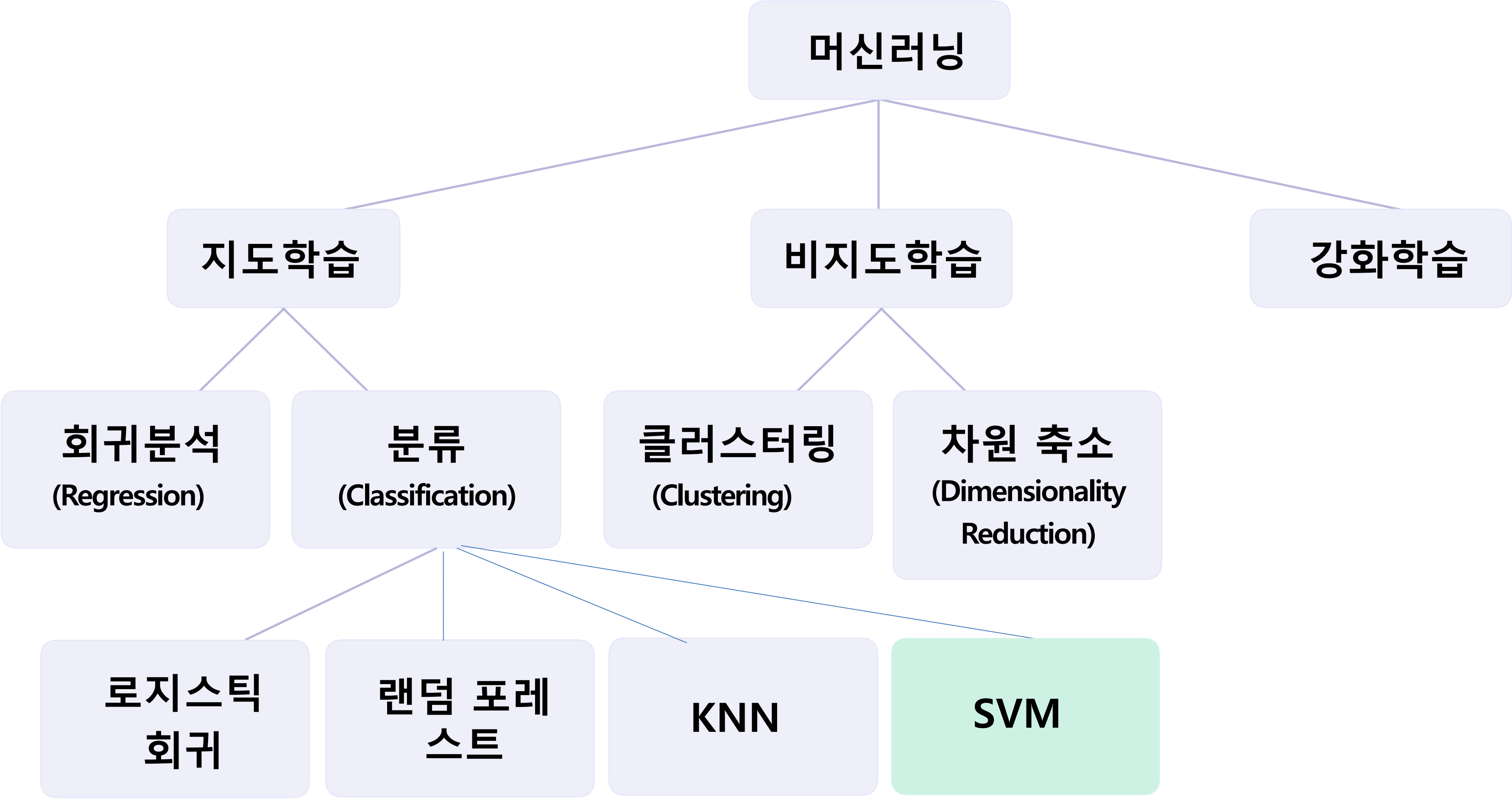


로지스틱 회귀

✔ 로지스틱 회귀 특징

- 주로 2개 값 분류(이진 분류)를 위해 사용
- 선형 회귀를 응용한 분류 알고리즘이기 때문에 선형 회귀의 특징 보유

SVM(Support Vector Machine)



SVM(Support Vector Machine)

✔ 두번째 분류 문제

· 문제 정의

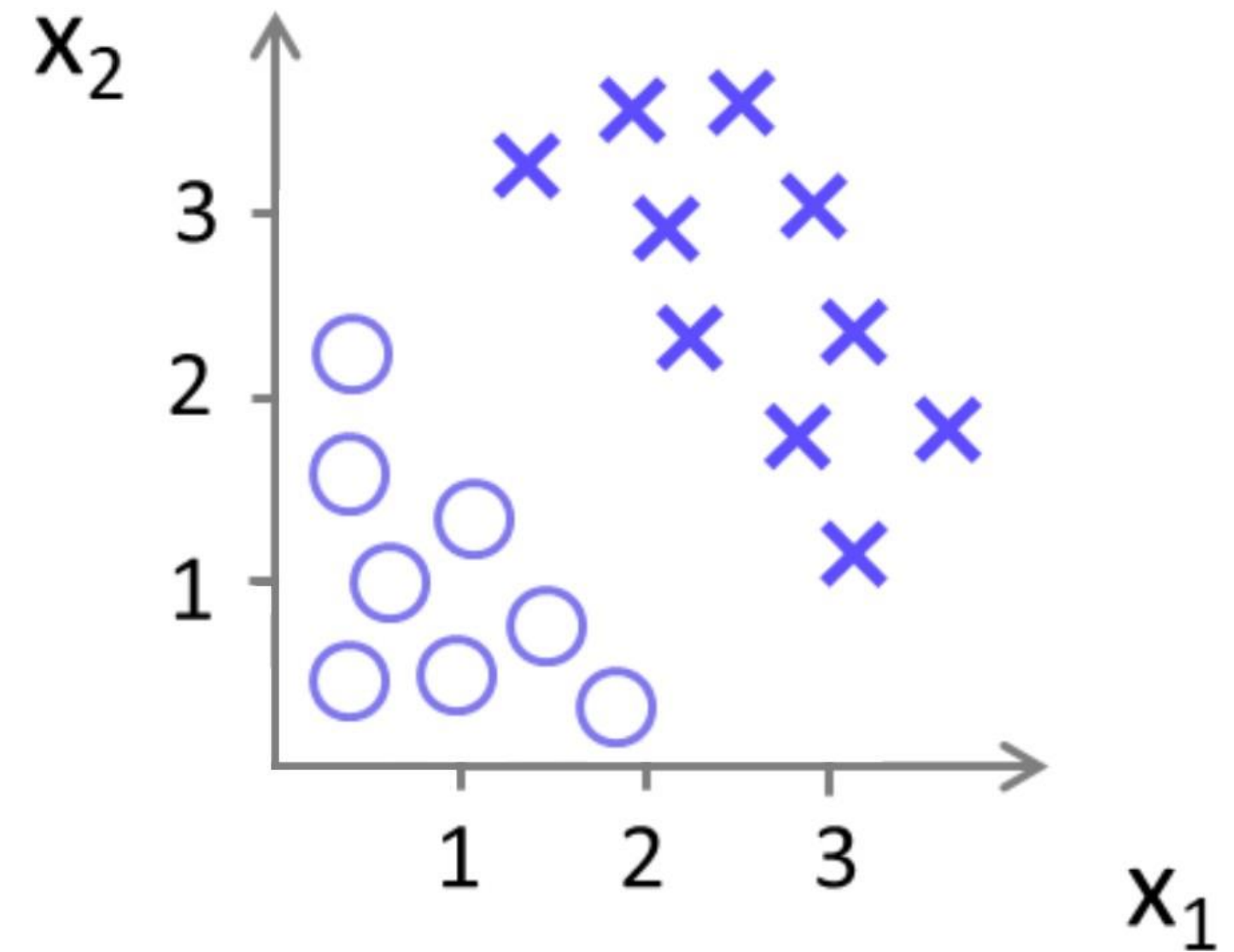
양성(1)과 음성(0)

두 개의 결과 값으로 분류되는 이진분류 문제

ex. 자연 여부 판별, 이상 거래 판별

· 해결 방안

SVM(Support Vector Machine) 분류 알고리즘



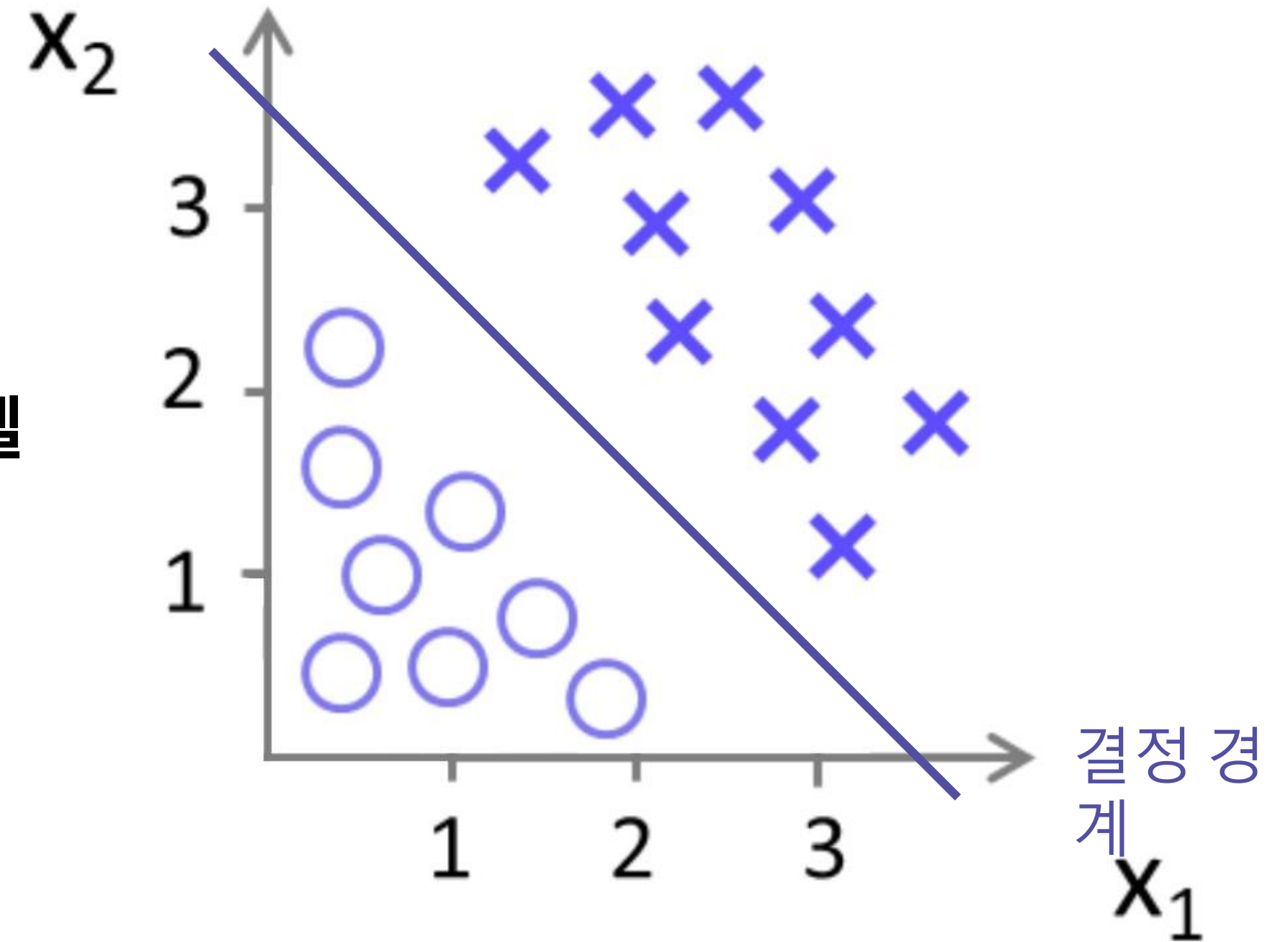
SVM(Support Vector Machine)

✔ SVM(Support Vector Machine)

딥러닝 기술 등장 이전까지
가장 인기 있던 분류 알고리즘

최적의 결정 경계(Decision Boundary)

즉, 데이터를 분류하는 기준 선을 정의하는 모델



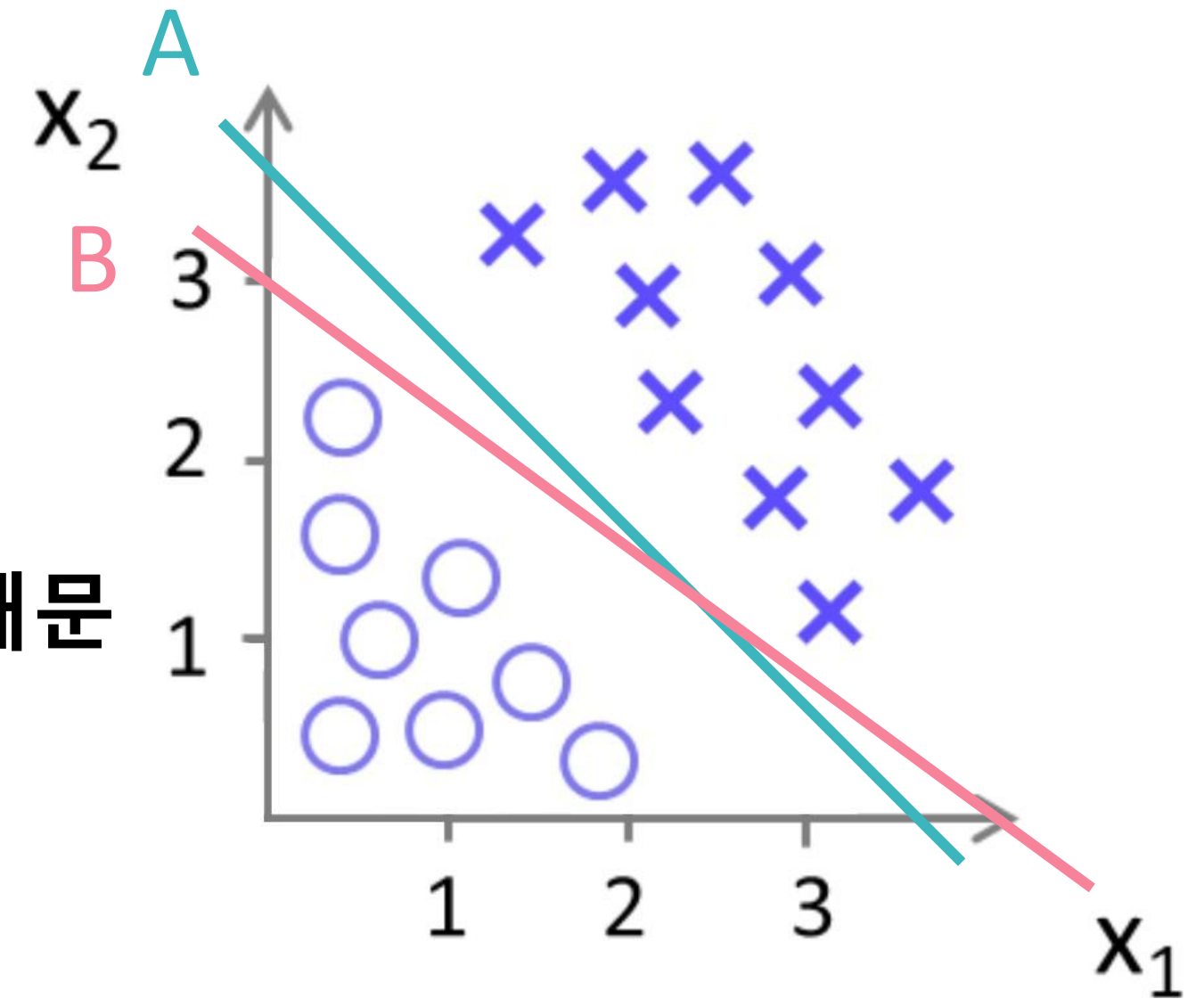
SVM(Support Vector Machine)

✓ 최적의 결정 경계(Decision Boundary)

최적의 결정 경계는
데이터 군으로부터 최대한 멀리 떨어지는 것

Q. A와 B 중 더 최적의 결정 경계는?

정답은 **A**! 데이터로부터 최대한 멀리 떨어져서 분류하기 때문

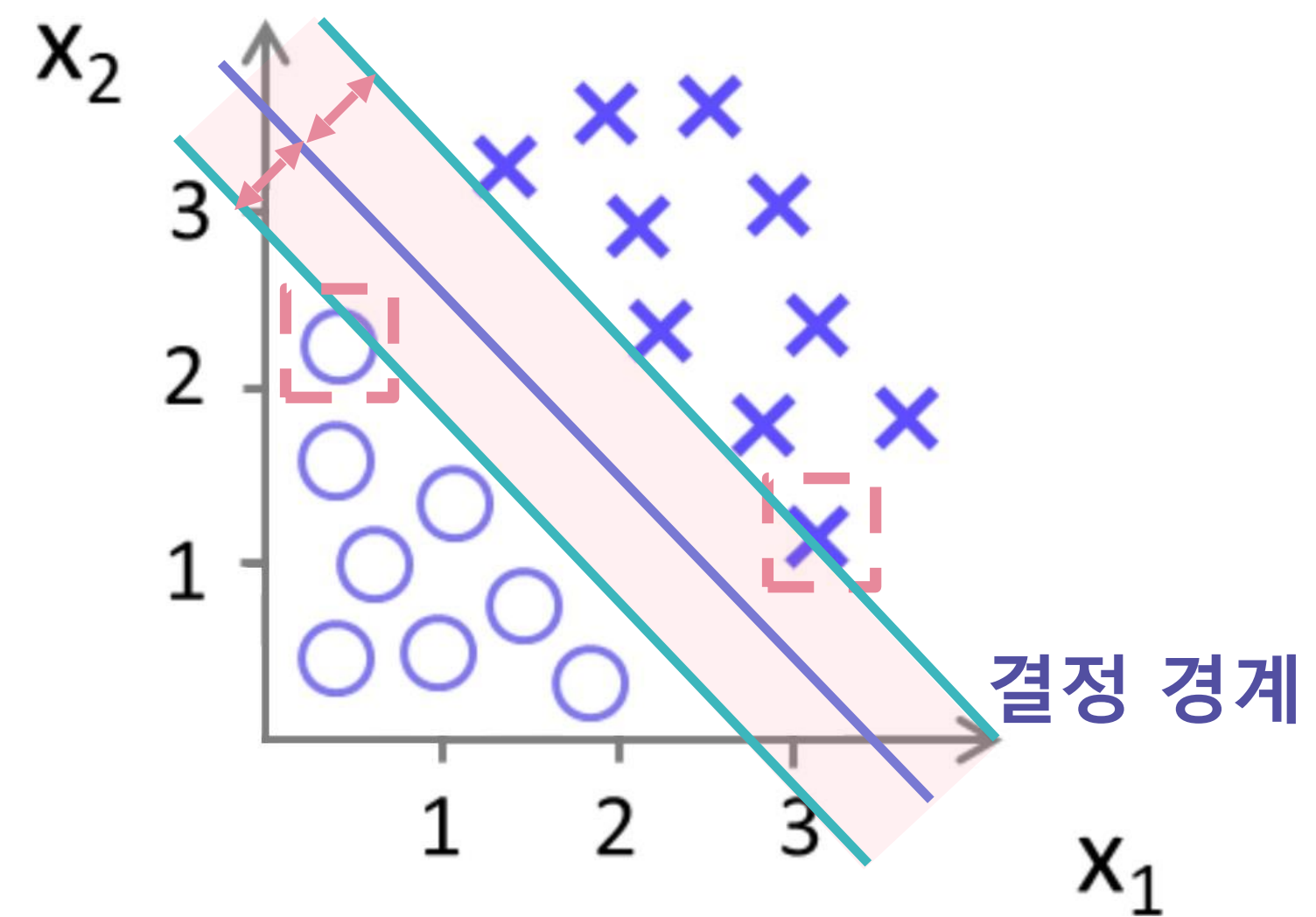
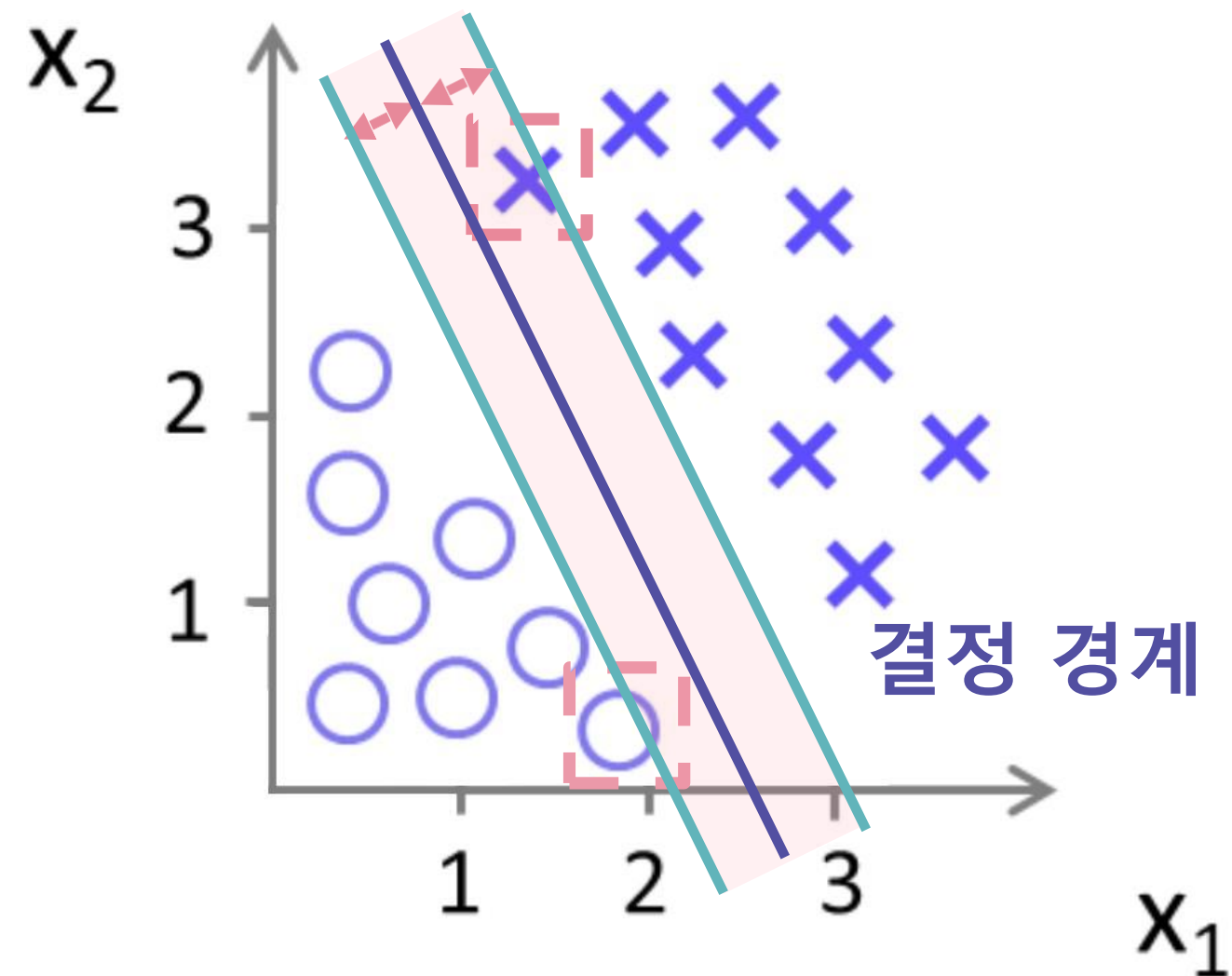


SVM(Support Vector Machine)

✔ 서포트 벡터(Support Vector)

결정 경계와 가장 가까이 있는 데이터 포인트들

Support Vector

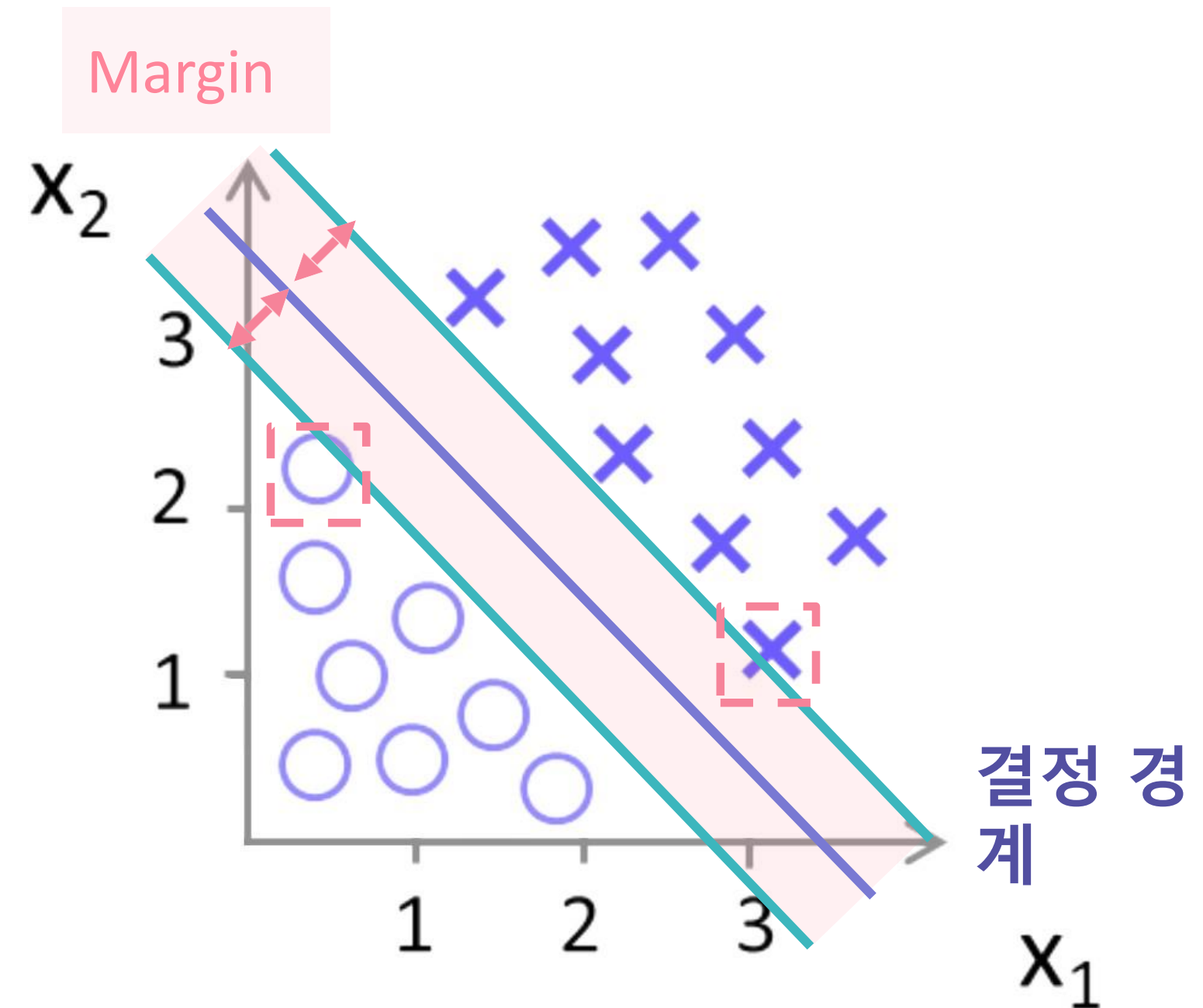


SVM(Support Vector Machine)

✓ 결정경계 여유? Margin

클래스를 분류하는 기준선에
여유(Margin)를 둘 수 있다

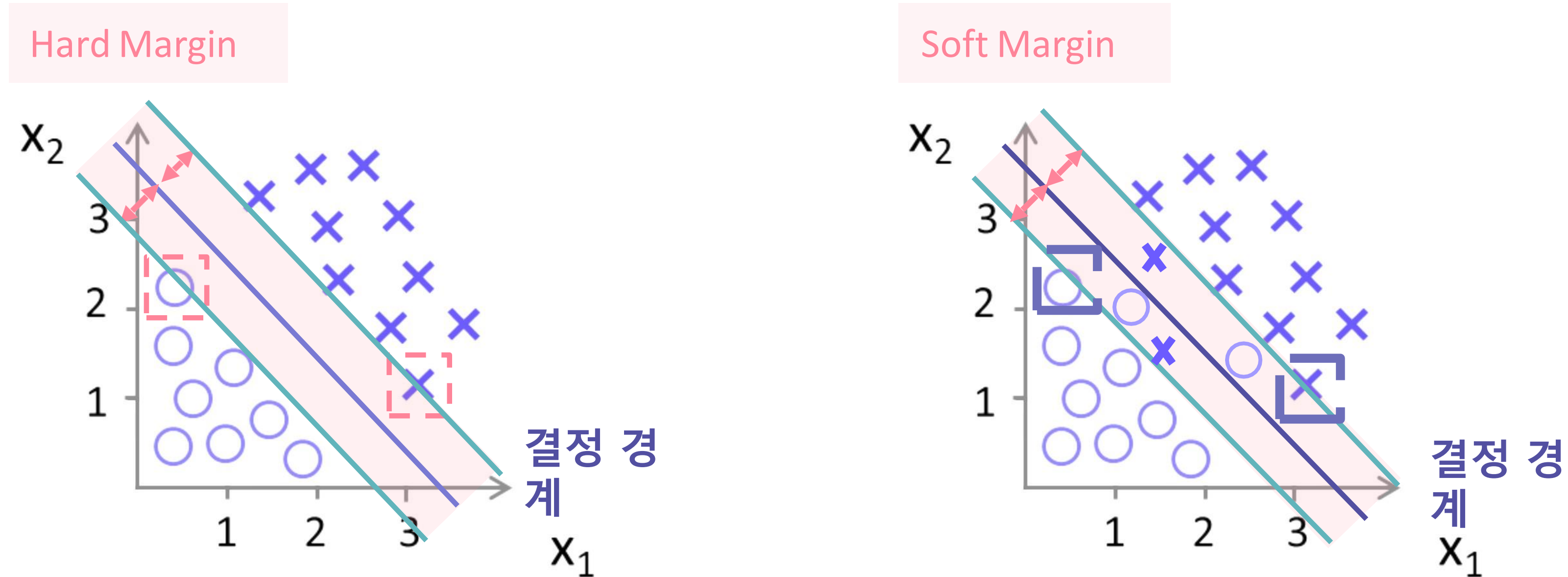
여유(Margin)
= 결정 경계와 서포트 벡터 사이의 거리
Margin을 최대화 하는 결정 경계를 찾음



SVM(Support Vector Machine)

✓ Hard Margin vs Soft Margin

이상치(Outlier) 허용 범위에 따라 Hard Margin과 Soft Margin으로 구분됨

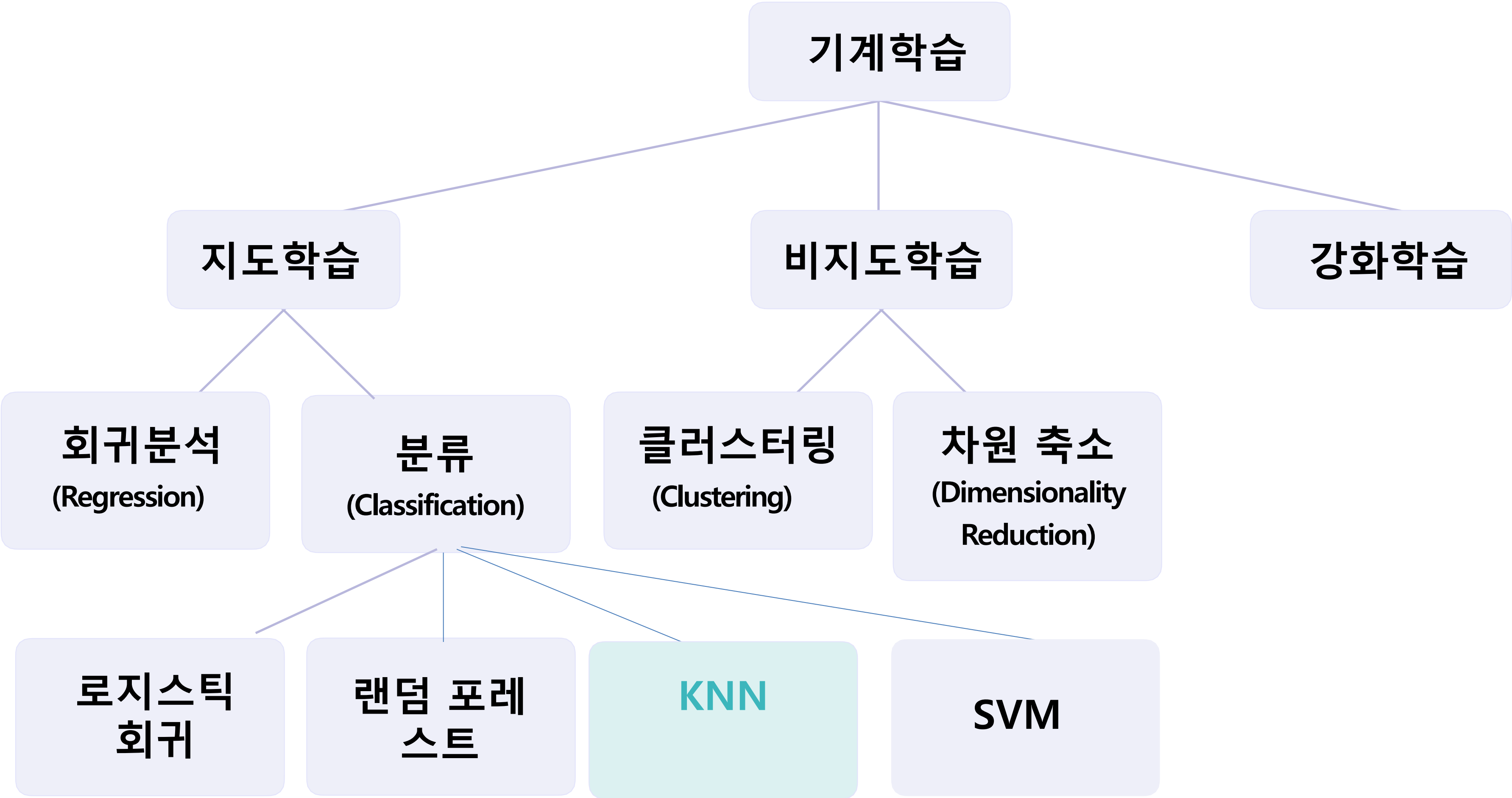


SVM(Support Vector Machine)

✔ SVM 특징

- 선형 분류와 비선형 분류 모두 가능
- 고차원 데이터에서도 높은 성능의 결과를 도출
- 회귀에도 적용 가능

KNN(K-Nearest Neighbor)



KNN(K-Nearest Neighbor)

✔ 문제 정의와 해결 방안

• 문제 정의

고객이 평가한 영화 평점 데이터를 기준으로
기존 보유 고객을 분류한 이후 새로 유입된 고객을 기준에
따라 분류하고자 하는 경우

• 해결 방안

KNN(k-Nearest Neighbor) 알고리즘



KNN(K-Nearest Neighbor)

✔ KNN(K-Nearest Neighbor)

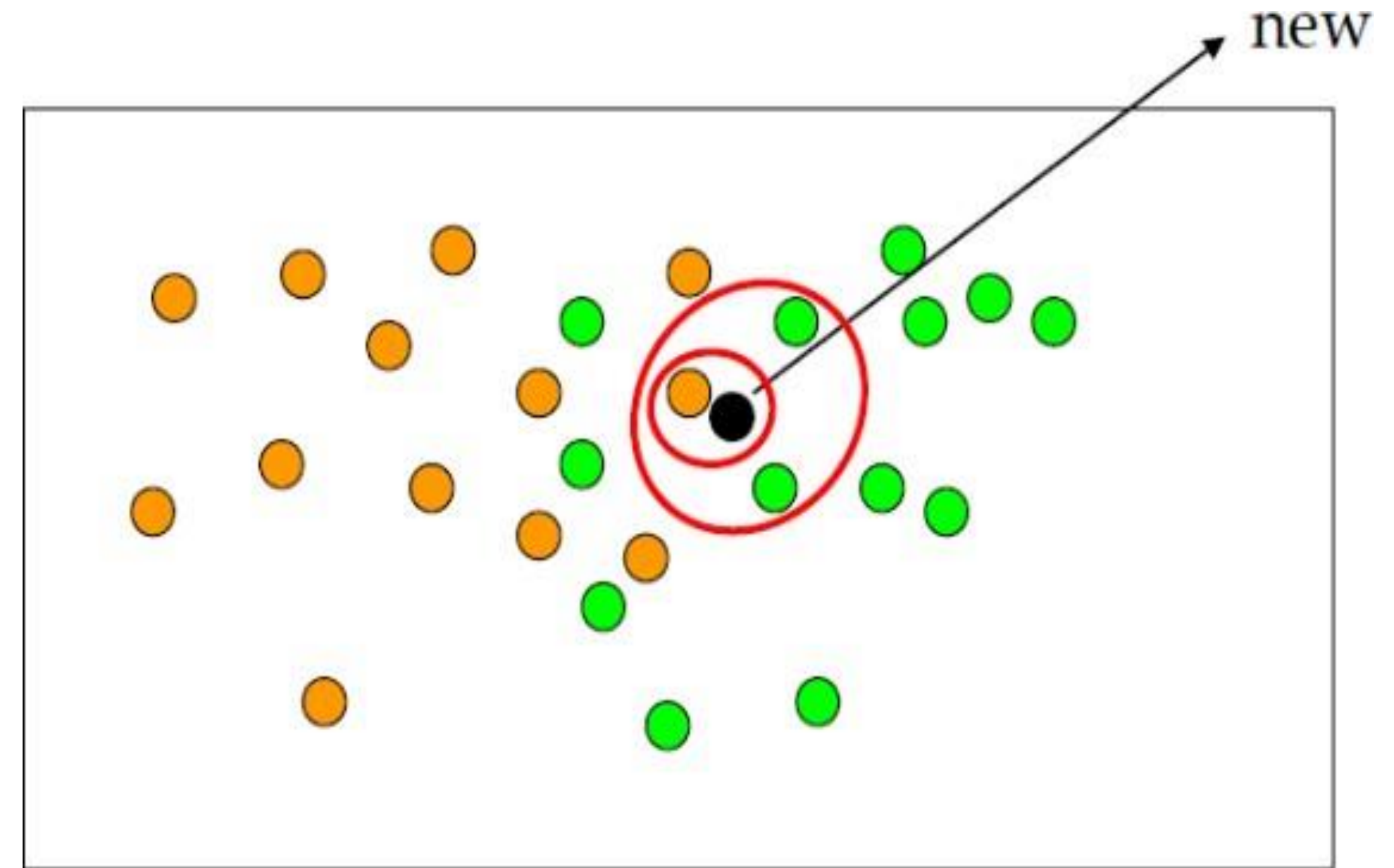
기존 데이터 가운데

가장 가까운 k 개 이웃의 정보로

새로운 데이터를 예측하는 방법론

유사한 특성을 가진 데이터는

유사 범주에 속하는 경향이 있다는 가정 하에 분류



KNN(K-Nearest Neighbor)

✔ KNN(K-Nearest Neighbor) 원리

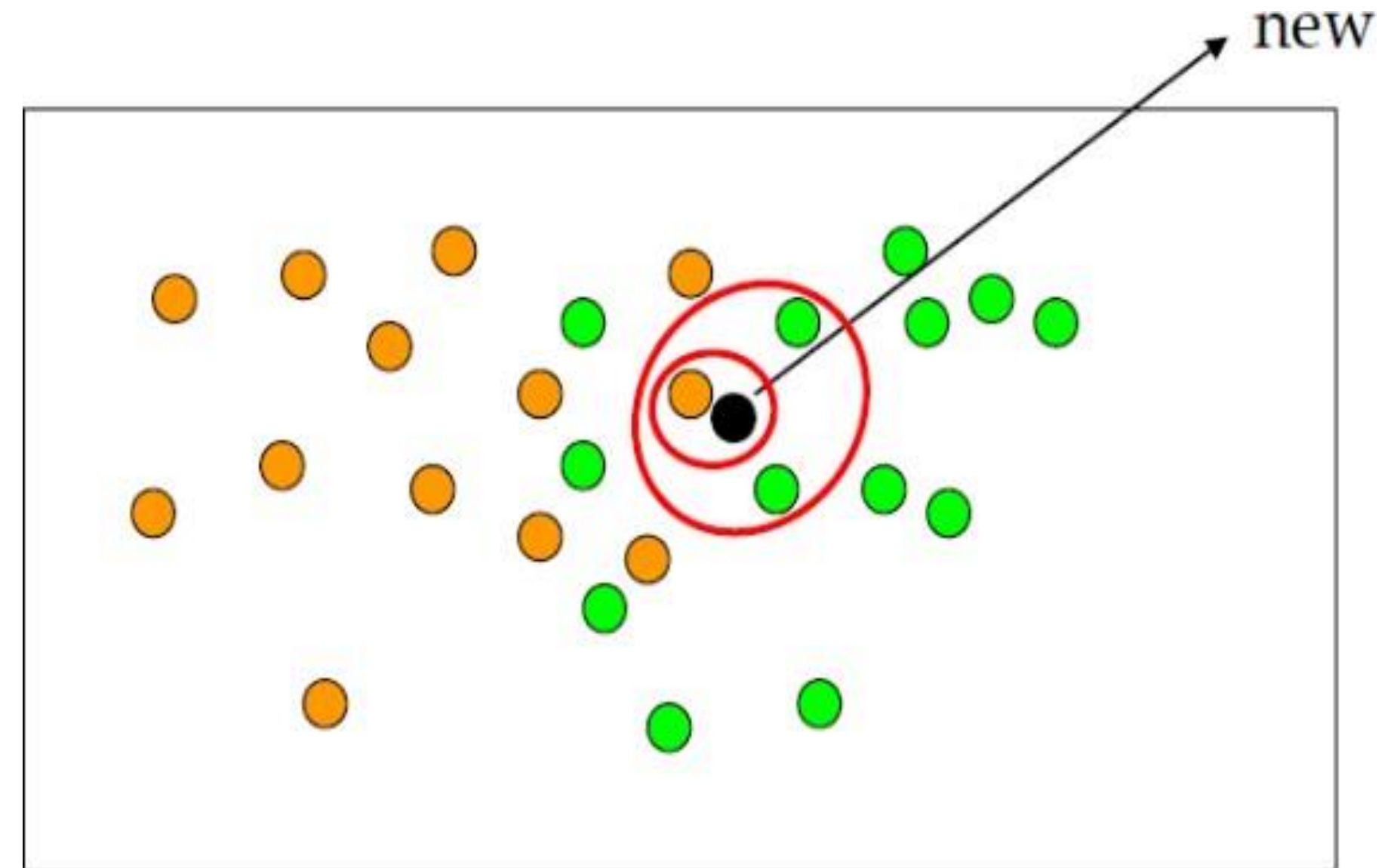
설정된 K값에 따라

가까운 거리 내의 이웃의 수에 따라 분류

새로운 고객 데이터(검정색)이 들어왔을 때 만약

K=1 이면 주황색 클래스로 분류

K=3 이면 초록색 클래스로 분류



KNN(K-Nearest Neighbor)

✓ KNN(K-Nearest Neighbor) 특징

- 직관적이며 복잡하지 않은 알고리즘, 결과 해석이 쉬움
- K값 결정에 따라 성능이 크게 좌우됨
- 딱히 학습이랄 것이 없는 Lazy Model

분류 평가 지표

✓ 혼동 행렬(Confusion Matrix)

분류 모델의 성능을 평가하기 위함

		예측	
		Positive	Negative
실제	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

분류 평가 지표

✓ 혼동 행렬(Confusion Matrix)

True Positive: 실제 **Positive** 인 값을 **Positive** 라고 예측(정답)

True Negative: 실제 **Negative** 인 값을 **Negative** 라고 예측(정답)

False Positive: 실제 **Negative** 인 값을 **Positive** 라고 예측(오답) – 1형 오류

False Negative: 실제 **Positive** 인 값을 **Negative** 라고 예측(오답) – 2형 오류

분류 평가 지표

✓ 정확도(Accuracy)

전체 데이터 중에서 제대로 분류된 데이터의 비율로,
모델이 얼마나 정확하게 분류하는지를 나타냄

일반적으로 분류 모델의 주요 평가 방법으로 사용됨

그러나, 클래스 비율이 불균형 할 경우
평가 지표의 신뢰성을 잃을 가능성이 있음

$$Accuracy = \frac{TP+TN}{P+N}$$

$$P: TP + FN,$$

$$N: TN + FP$$

분류 평가 지표

✓ 정밀도(Precision)

모델이 Positive라고 분류한 데이터 중에서 실제로 Positive인 데이터의 비율

Negative가 중요한 경우

즉, 실제로 Negative인 데이터를 Positive라고 판단하면 안되는 경우 사용되는 지표

$$Precision = \frac{TP}{TP+FP}$$

분류 평가 지표

✔ Negative가 중요한 경우

스팸 메일 판결을 위한 분류 문제

해당 메일이 스팸일 경우 **Positive**,
스팸이 아닐 경우 즉, 일반 메일일 경우 **Negative**

일반 메일을 **스팸 메일(Positive)**로 잘못 예측했을 경우
중요한 메일을 전달받지 못하는 상황이 발생할 수 있음

분류 평가 지표

✓ 재현율(Recall, TPR)

실제로 Positive인 데이터 중에서
모델이 Positive로 분류한 데이터의 비율

Positive가 중요한 경우

즉, 실제로 Positive인 데이터를
Negative라고 판단하면 안되는 경우 사용되는
지표

$$Recall = \frac{TP}{TP+FN} = \frac{TP}{P}$$

분류 평가 지표

✔ Positive가 중요한 경우

악성 종양 여부 판결을 위한 검사

악성 종양일 경우 **Positive**,

악성 종양이 아닐 경우 즉, 양성 종양일 경우 **Negative**

악성 종양(Positive)을 양성 종양(Negative)으로 잘못 예측했을 경우
제 때 치료를 받지 못하게 되어 생명이 위급해질 수 있음

분류 평가 지표

✔ 다양한 분류 지표의 활용

분류 목적에 따라 다양한 지표를 계산하여 평가

- 분류 결과를 전체적으로 보고 싶다면 → 혼동 행렬(Confusion Matrix)
- 정답을 얼마나 잘 맞췄는지 → 정확도(Accuracy)
- FP 또는 FN의 중요도가 높다면 → 정밀도(Precision), 재현율(Recall)

감사합니다.