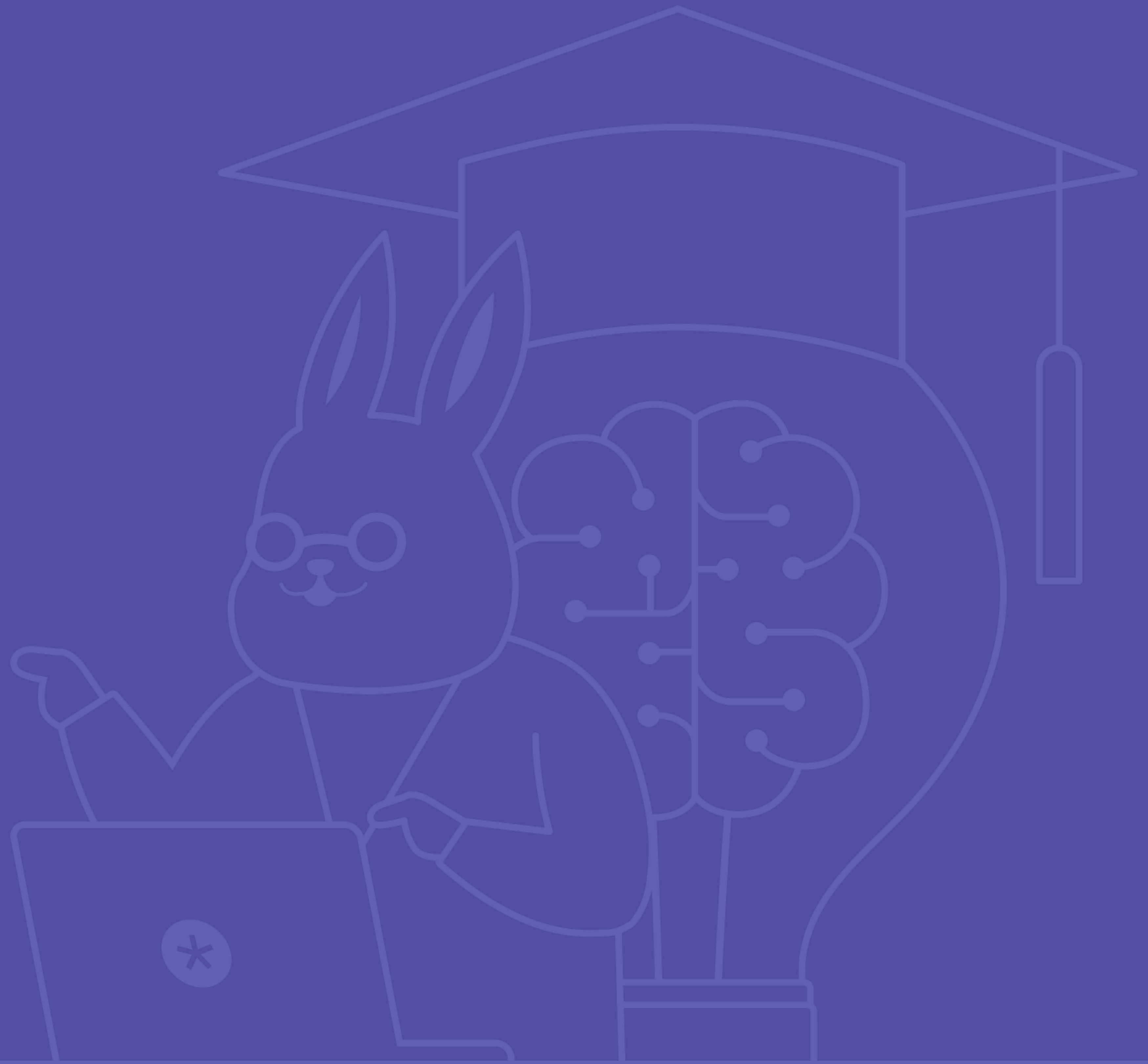
Classification

2장 분류

고혜선 선생님



/* elice */

Contents

Curriculum

What is Classification?

분류 (Classification)의 의미와 특성을 이해하고,
회귀 (Regression)과의 차이점을 알아본다.

Logistic / Softmax Regression

Logistic Regression과 Softmax Regression의 의미와
특성을 이해한다.

Evaluation Metrics

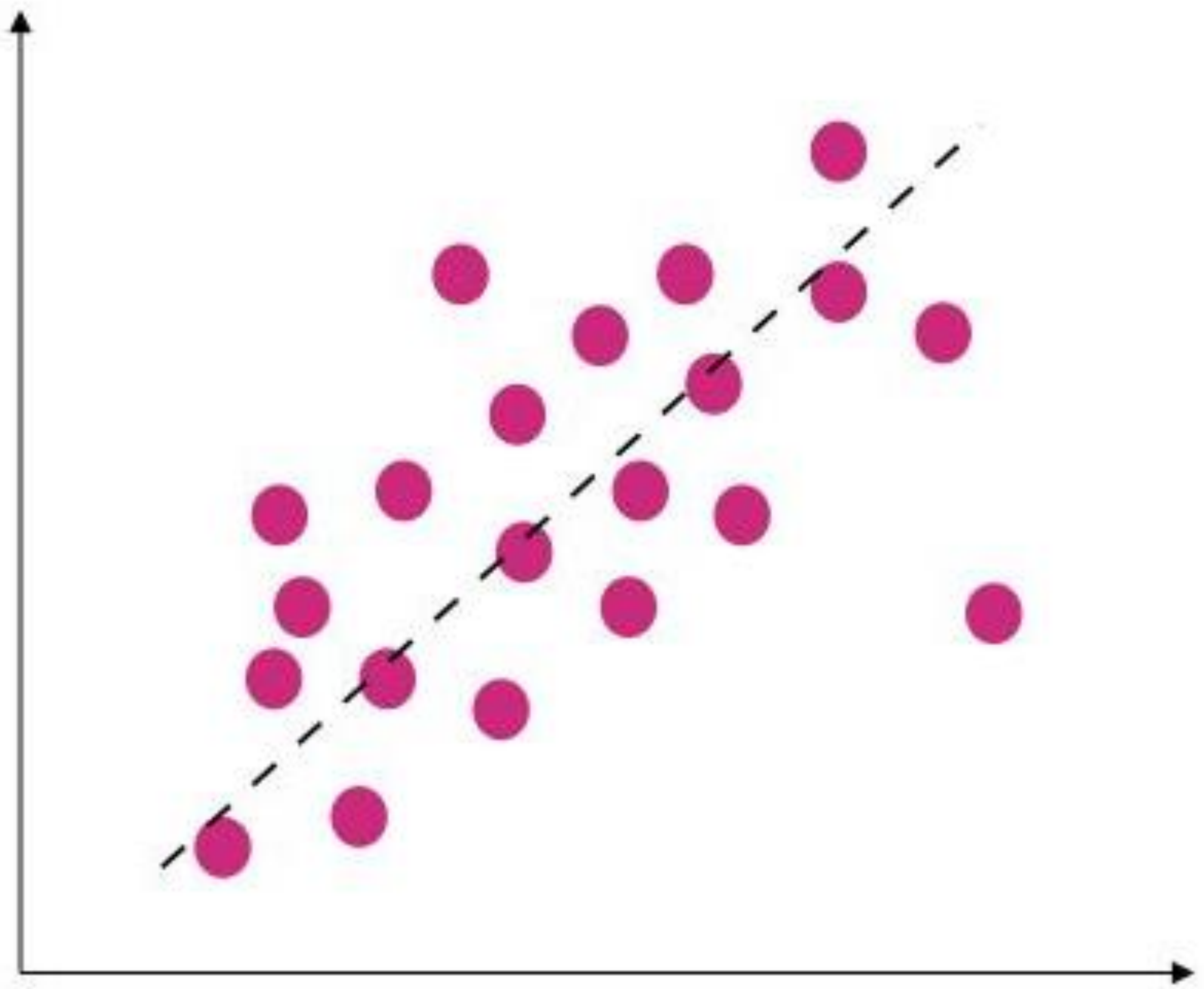
혼동행렬 (Confusion Matrix)를 이해하고, 분류에
사용되는 여러 지표에 대해 알아본다.

01

What is Classification?

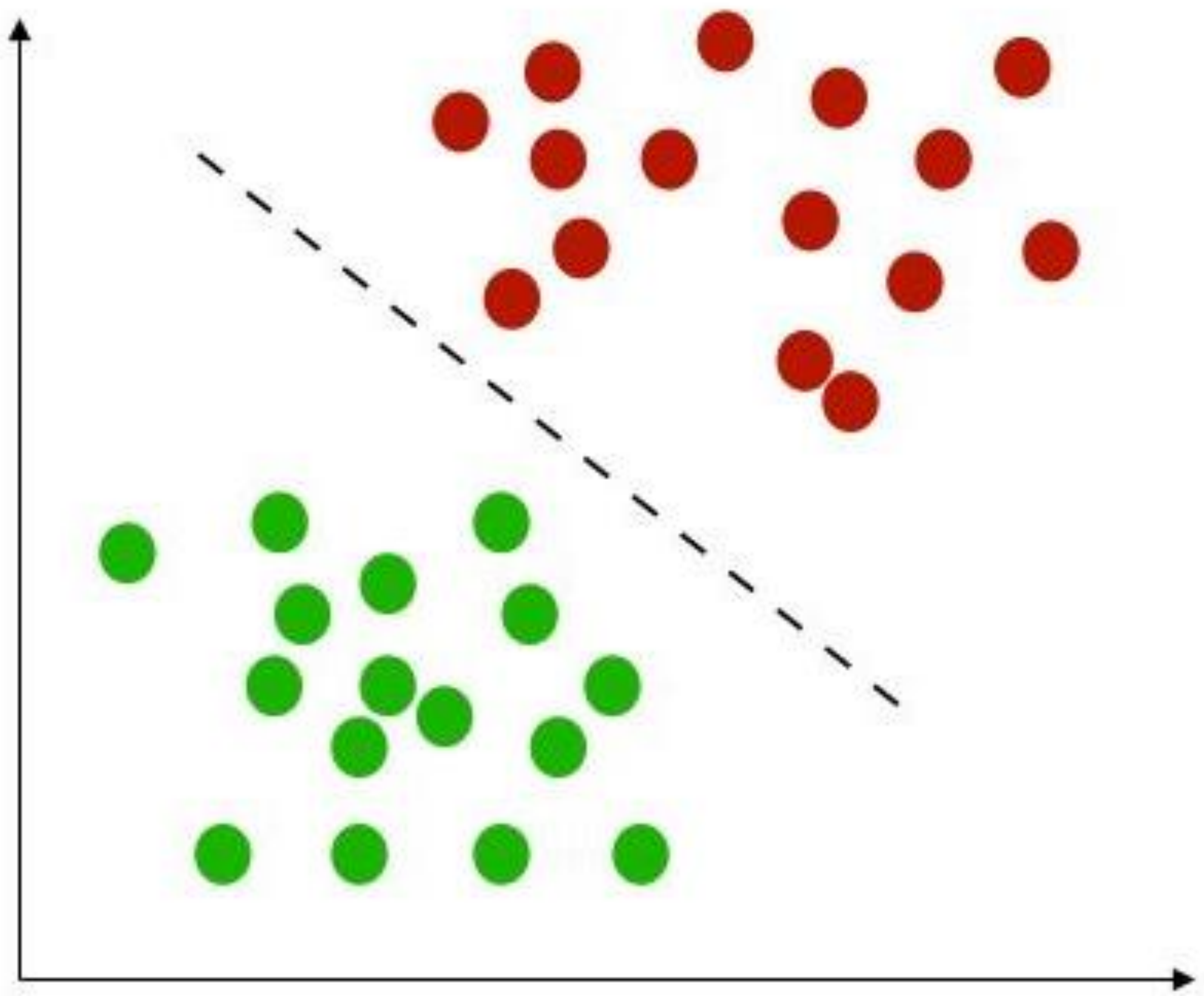
Recall: Regression

주어진 데이터를 가장 잘 근사(표현)하는 선을 통해 “**continuous output** " 을 예측하는 것



What is Classification?

어느 카테고리에 있는지 가장 잘 분류하는 선을 통해 “**discrete output** " 을 예측하는 것



Example of classification 1

VS

Example of classification 1

$f($  $) = ?$

Example of classification 1

$f($  $) = \text{cat}$

Example of classification 1

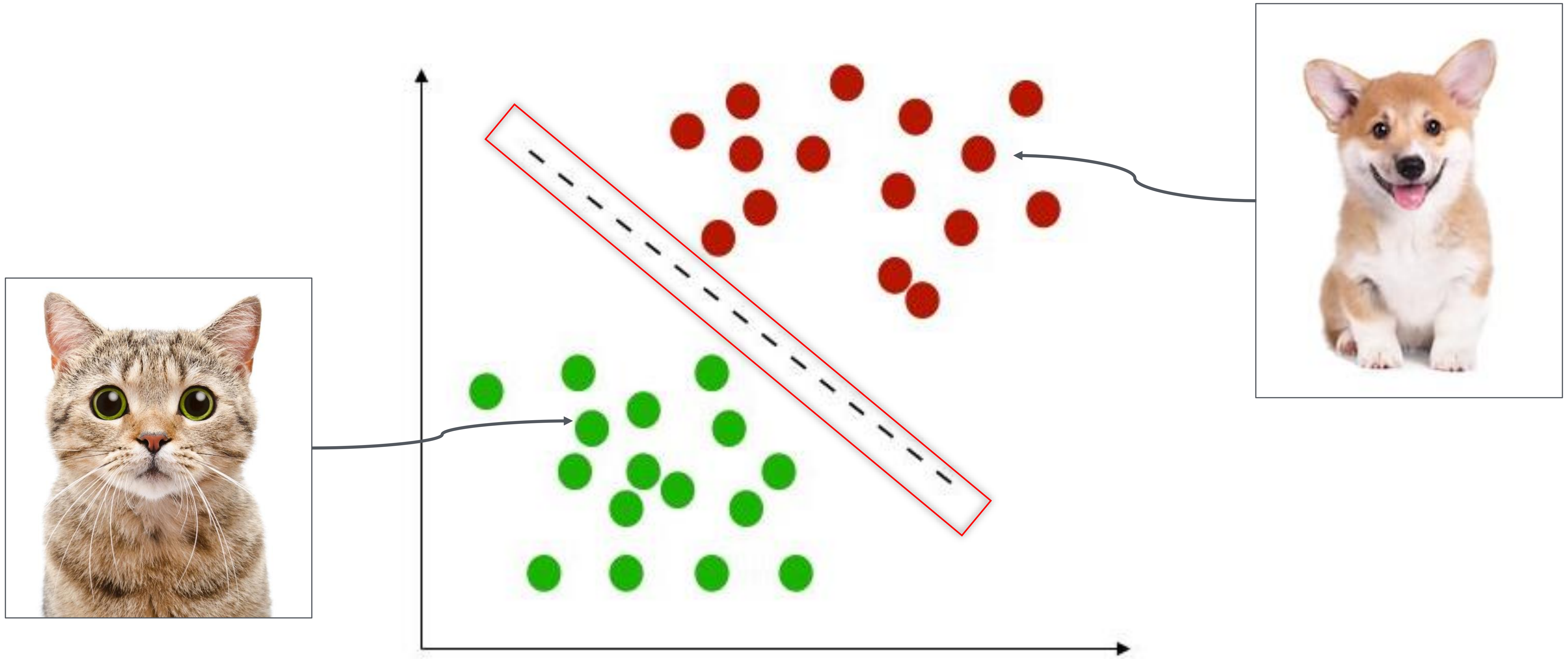
$f($  $) = ?$

Example of classification 1

$f($  $)$ = dog

Classification이란

어느 카테고리에 있는지 가장 잘 분류하는 선을 통해 “**discrete output** " 을 예측하는 것



Example of classification 2

VS

Example of classification 2

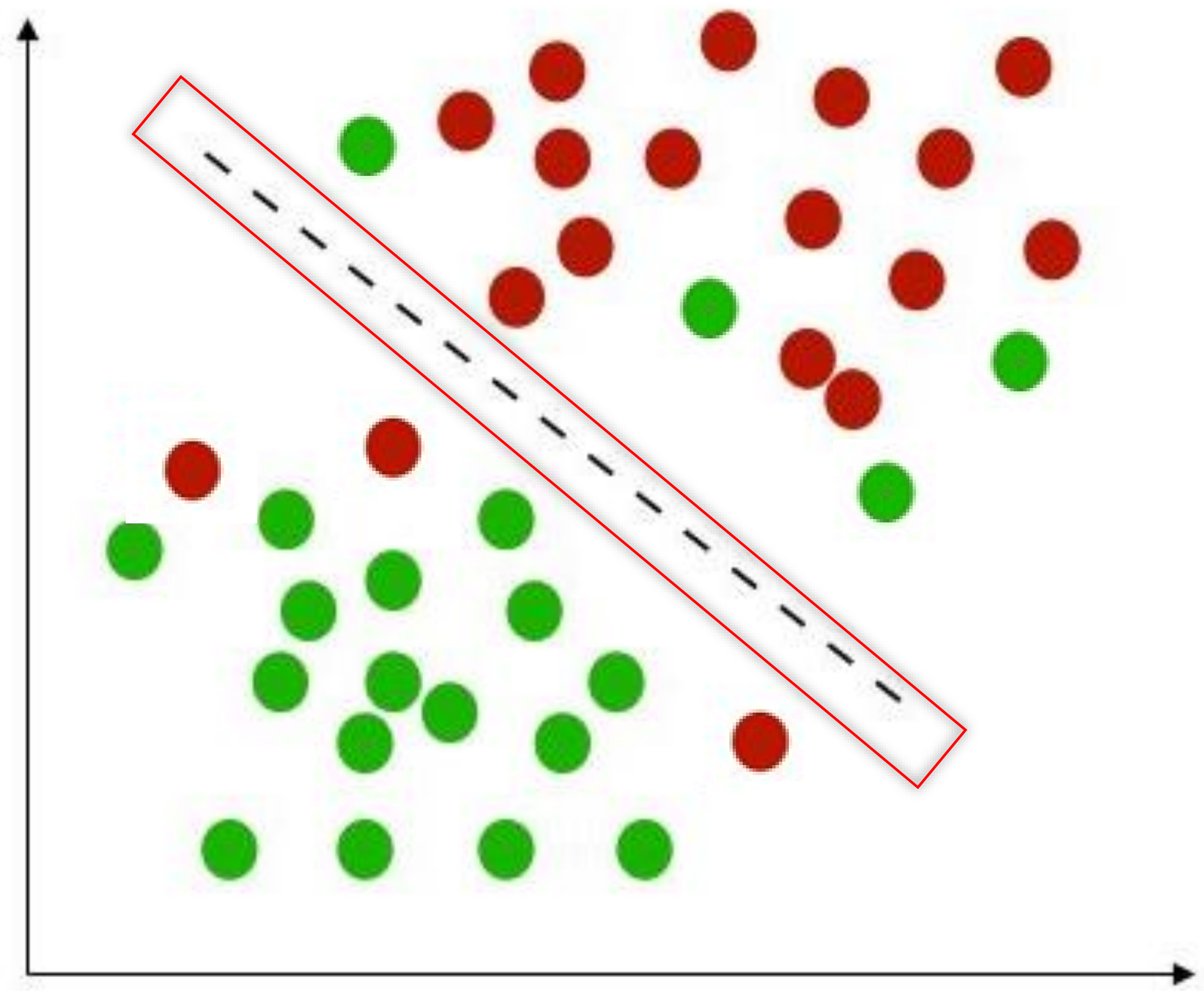
$f($  $) = ?$

Example of classification 2

$f($  $) = ?$

Classification이란

어느 카테고리에 있는지 가장 잘 분류하는 선을 통해 “**discrete output** " 을 예측하는 것



Examples of (Binary) Classification

- **Spam classification:** Spam vs. Not Spam
- **Image classification:** Dog vs. Cat
- **Cancer Diagnosis:** Benign vs. Malignant
- **Sentimental Analysis:** Positive vs. Negative

$$y \in \{0, 1\}$$

0 : Negative Class (e.g., Benign Tumor)

1: Positive Class (e.g., Malignant Tumor)

Regression vs. Classification: $y = f(x)$

- **Output** = Continuous values
- **Evaluation** = RMSE, RMAE, R2, AR2
- **Prediction** = ordered

- **Output** = Discrete values (pre-defined classes)
- **Evaluation** = Accuracy, Precision, Recall
- **Prediction** = unordered

Types of Classification

- Logistic Regression
- Softmax Regression
- Naïve Bayes Classifier
- Support Vector Machines (SVM)
- K-Nearest Neighbors (KNN)
- Decision Tree
- Multi-layer Perceptron (MLP)
- More..

Types of Classification

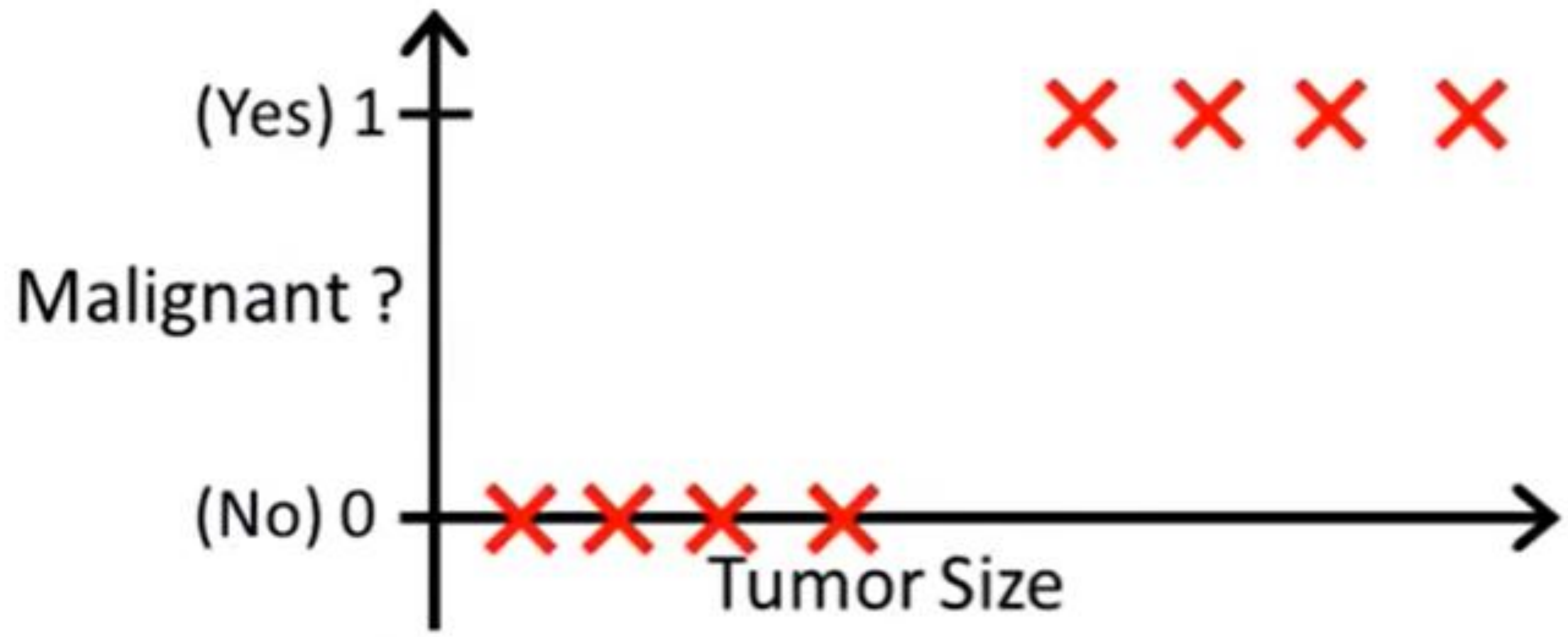
- Logistic Regression
- Softmax Regression
- Naïve Bayes Classifier
- Support Vector Machines (SVM)
- K-Nearest Neighbors (KNN)
- Decision Tree
- Multi-layer Perceptron (MLP)
- More..

02

Logistic / Softmax Regression

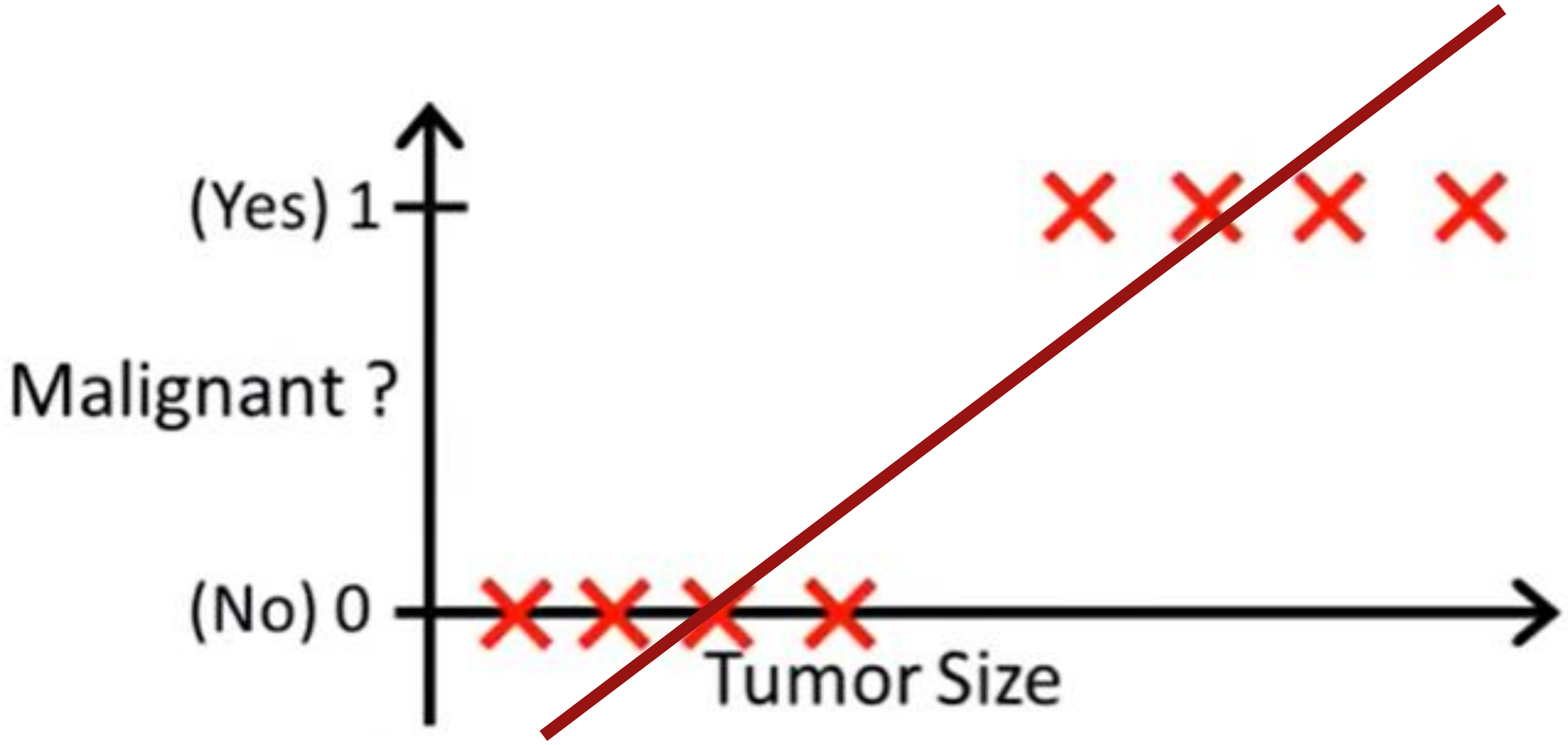
Logistic Regression이란

- Benign과 Malignant를 잘 분류하는 선을 찾는 것



Logistic Regression이란

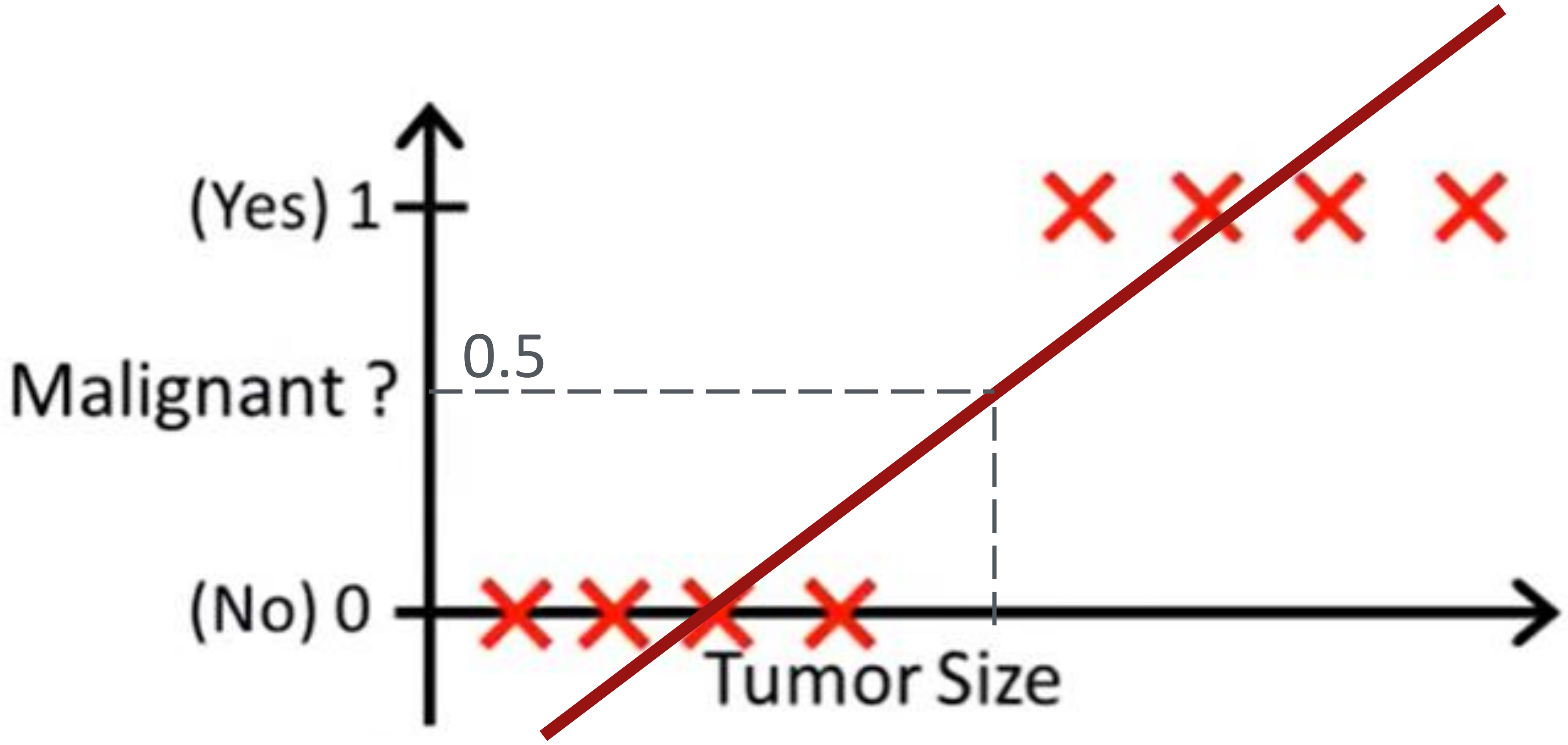
- Linear Regression 식을 그대로 적용한다면?



$$\mathrm{H}(x) = w_0 + w_1 x$$

Logistic Regression이란

- Linear Regression 식을 그대로 적용한다면?



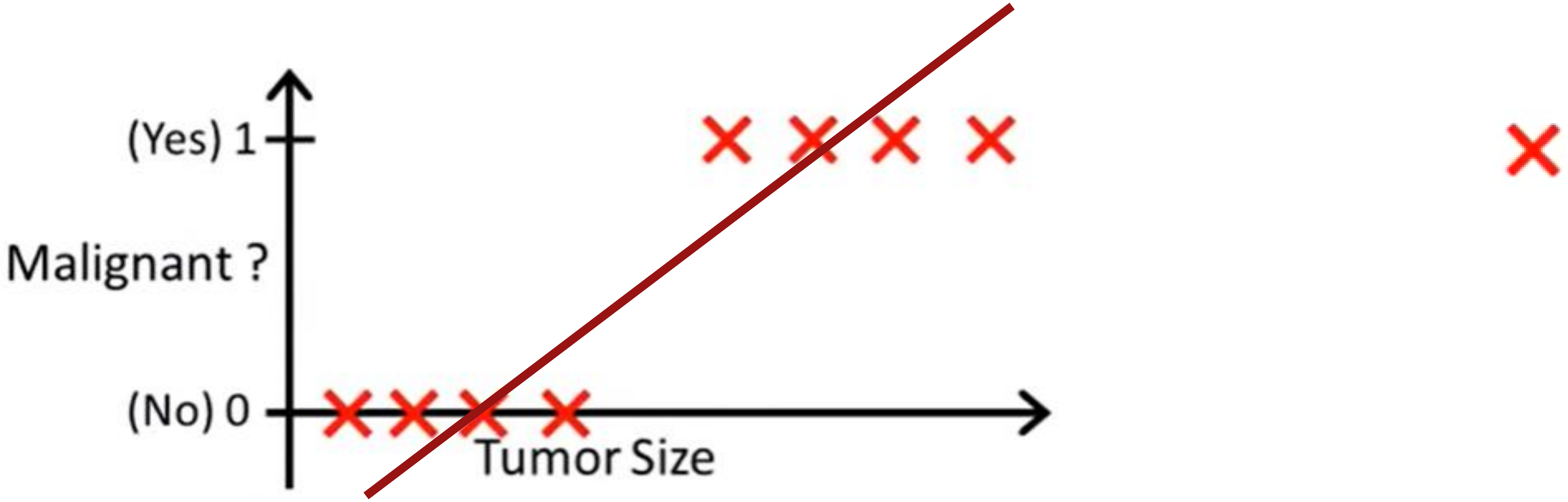
$$\mathrm{H}(x) = w_0 + w_1 x$$

$$H(x) \geq \mathbf{0.5} \Rightarrow y = 1$$

$$H(x) < \mathbf{0.5} \Rightarrow y = 0$$

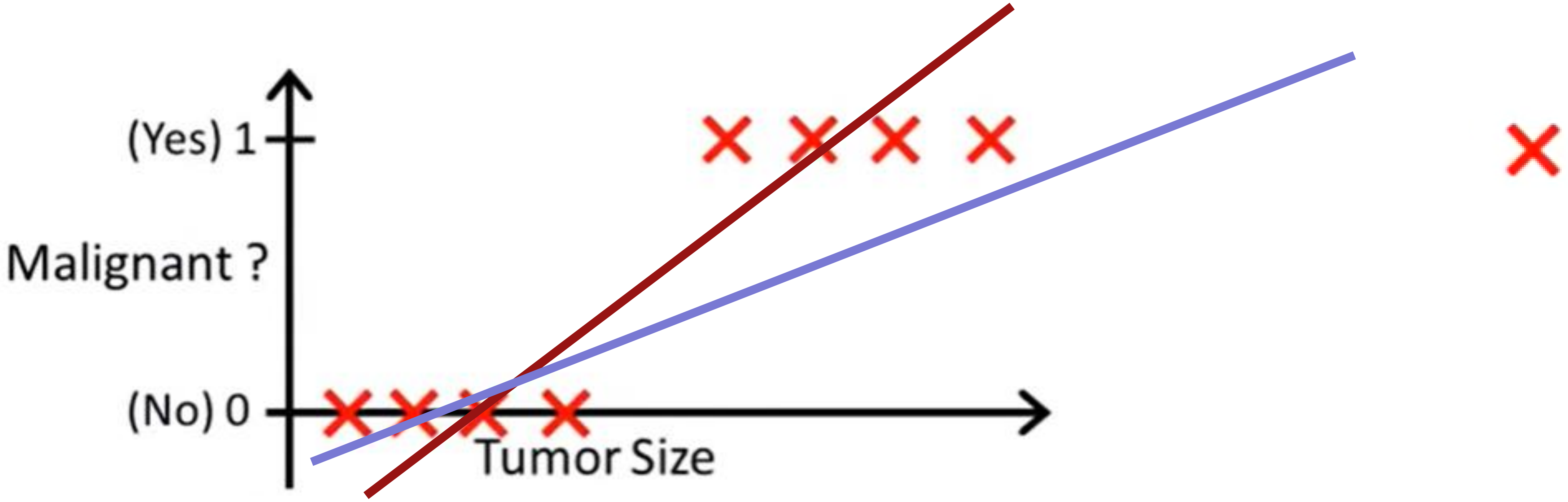
Logistic Regression이란

- Linear Regression 식을 그대로 적용한다면?



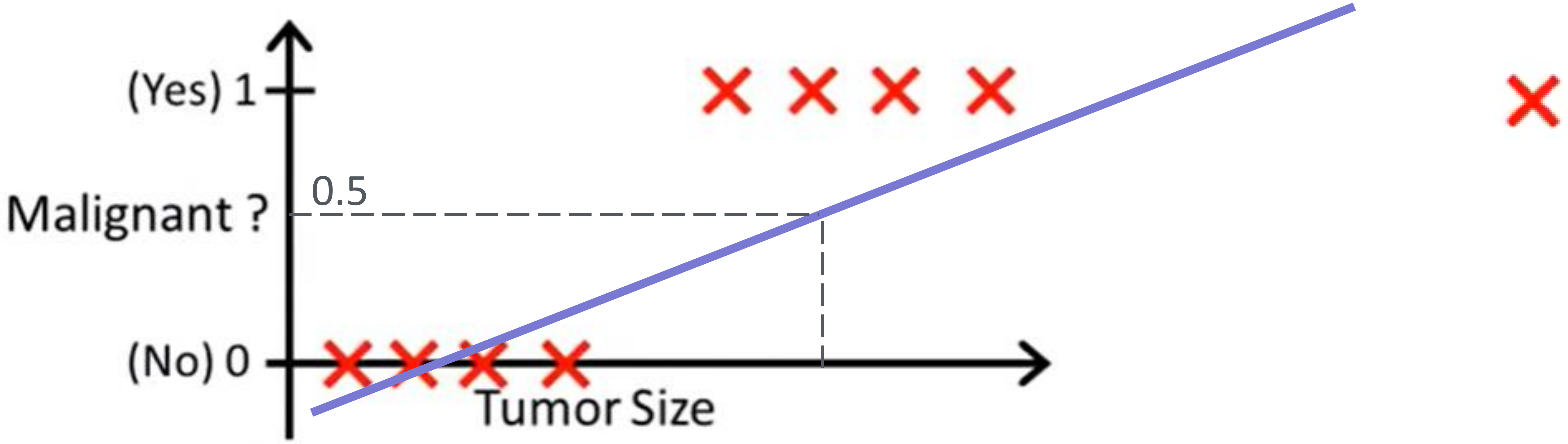
Logistic Regression이란

- Linear Regression 식을 그대로 적용한다면?



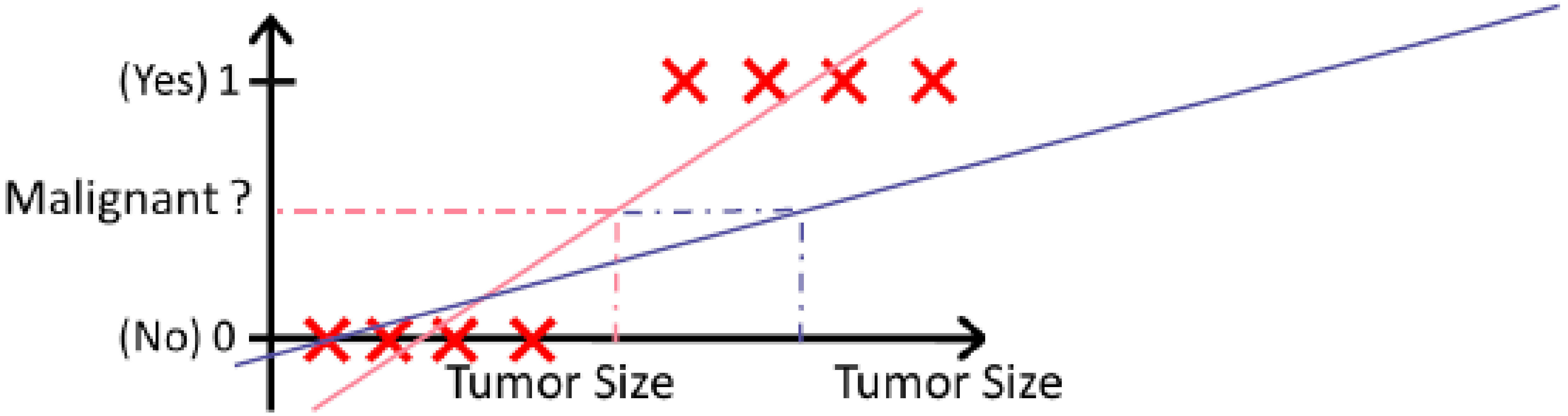
Logistic Regression이란

- Linear Regression 식을 그대로 적용한다면?



Linear Regression을 그대로 적용하면 안된다!

- Binary classification은 0 또는 1 값만 가지는 반면,
  Linear regression은 그 범위 밖(0 이하, 1 이상)의 값을 가질 수 있다.
- 0 또는 1 사이의 값만 내보내는 Hypothesis 함수가 필요하다. $\Rightarrow \mathbf{0} \leq H(X) \leq \mathbf{1}$



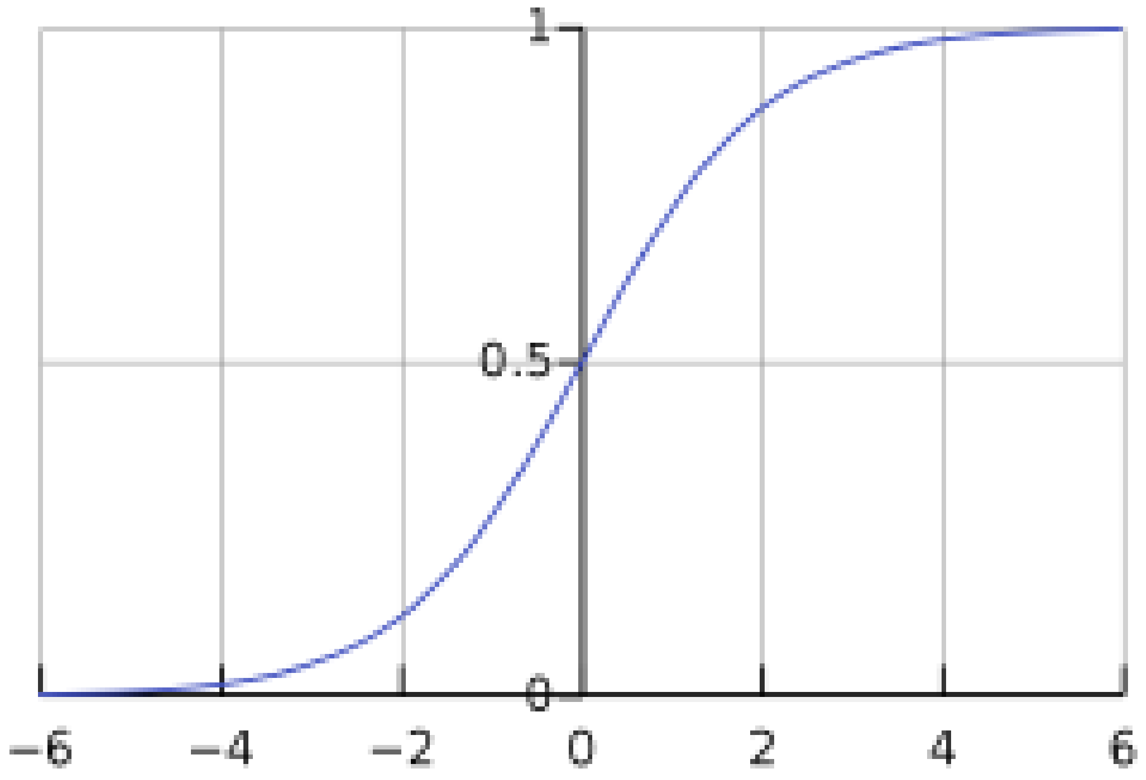
Sigmoid (Logistic) Function

- Goal: $\mathbf{0} \leq H(X) \leq \mathbf{1}$
- 즉, 예측 값을 확률 (0 ~ 1)로 매핑하는 함수가 필요하다.
- **Sigmoid의 성질**

  (1) Bounded [0,1]

  (2) Differential

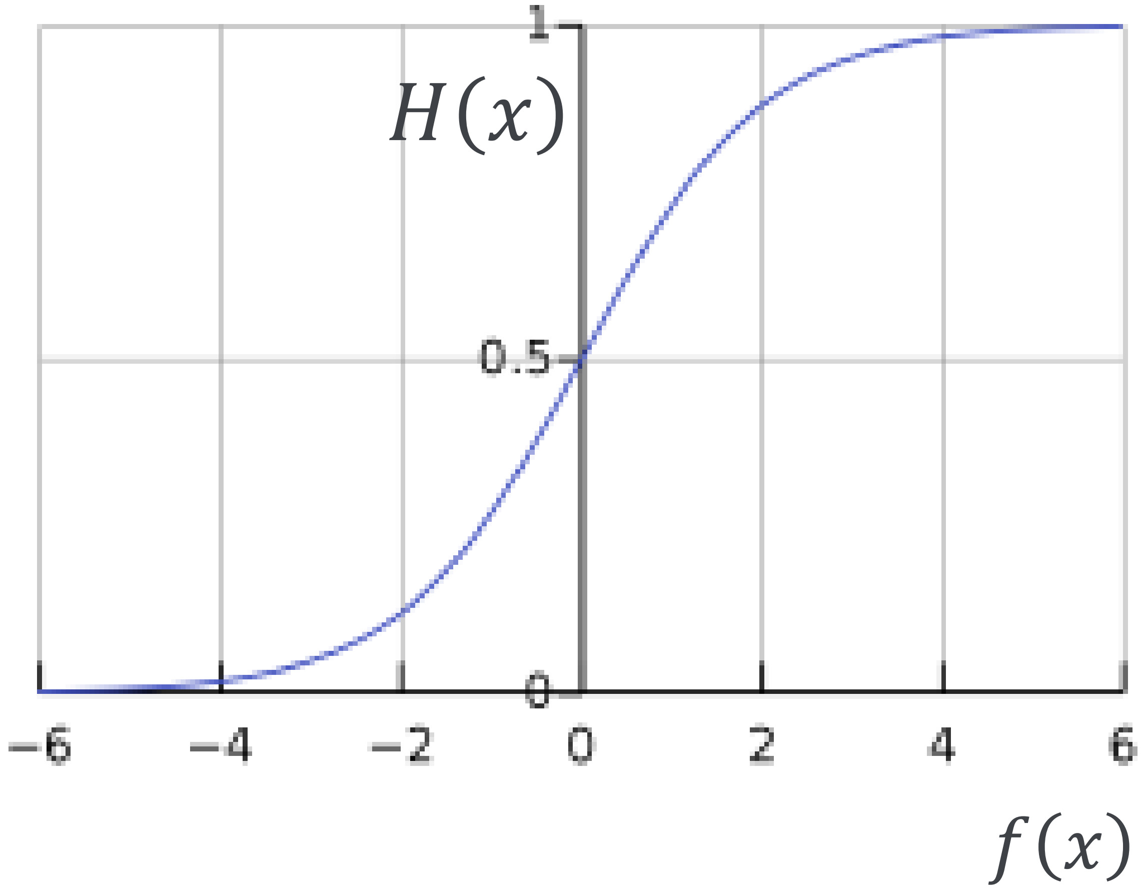
  (3) Defined for all real inputs



$$H(x) = \frac{1}{1+e^{-f(x)}} = \frac{1}{1+e^{-w^T x}}$$

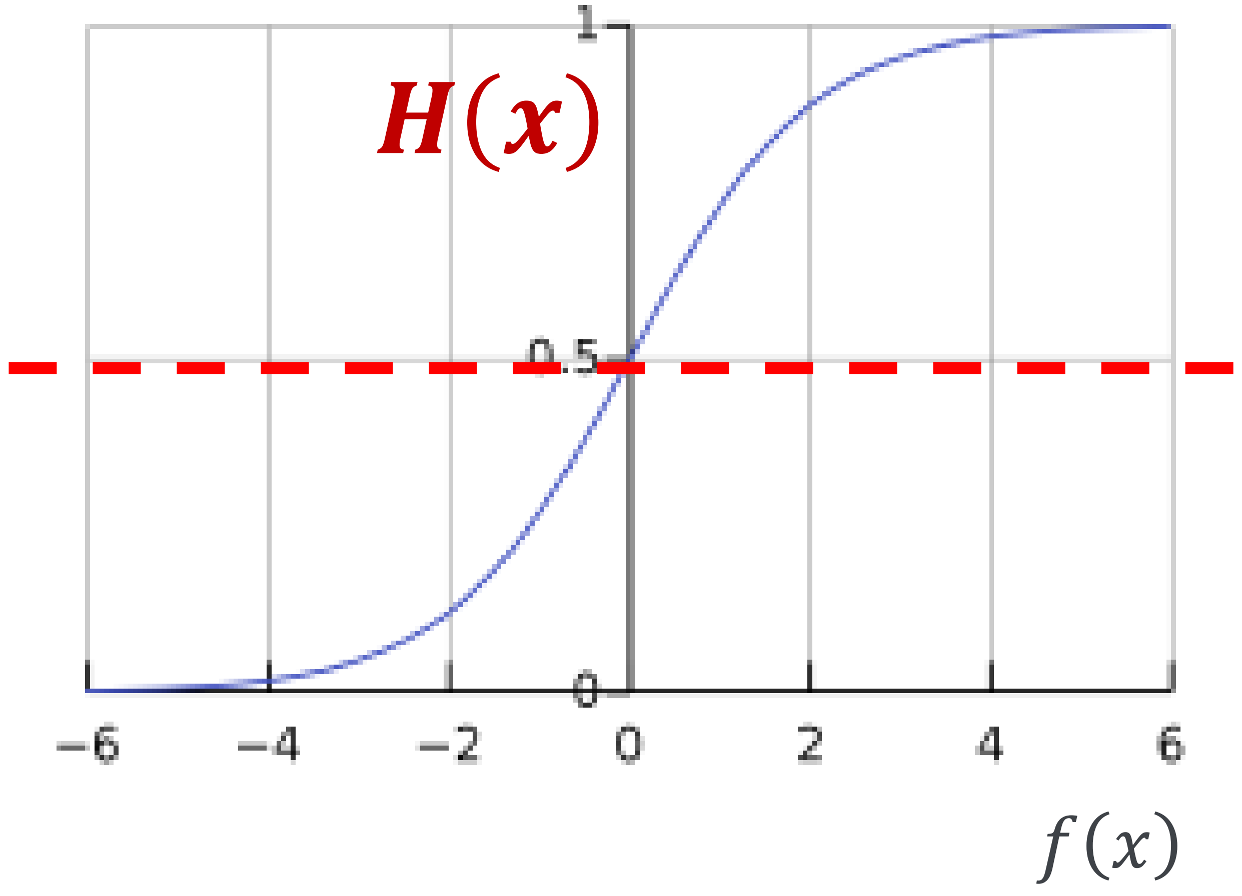
Sigmoid (Logistic) Function

- 언제 y=1이라고 예측할까?



Sigmoid (Logistic) Function

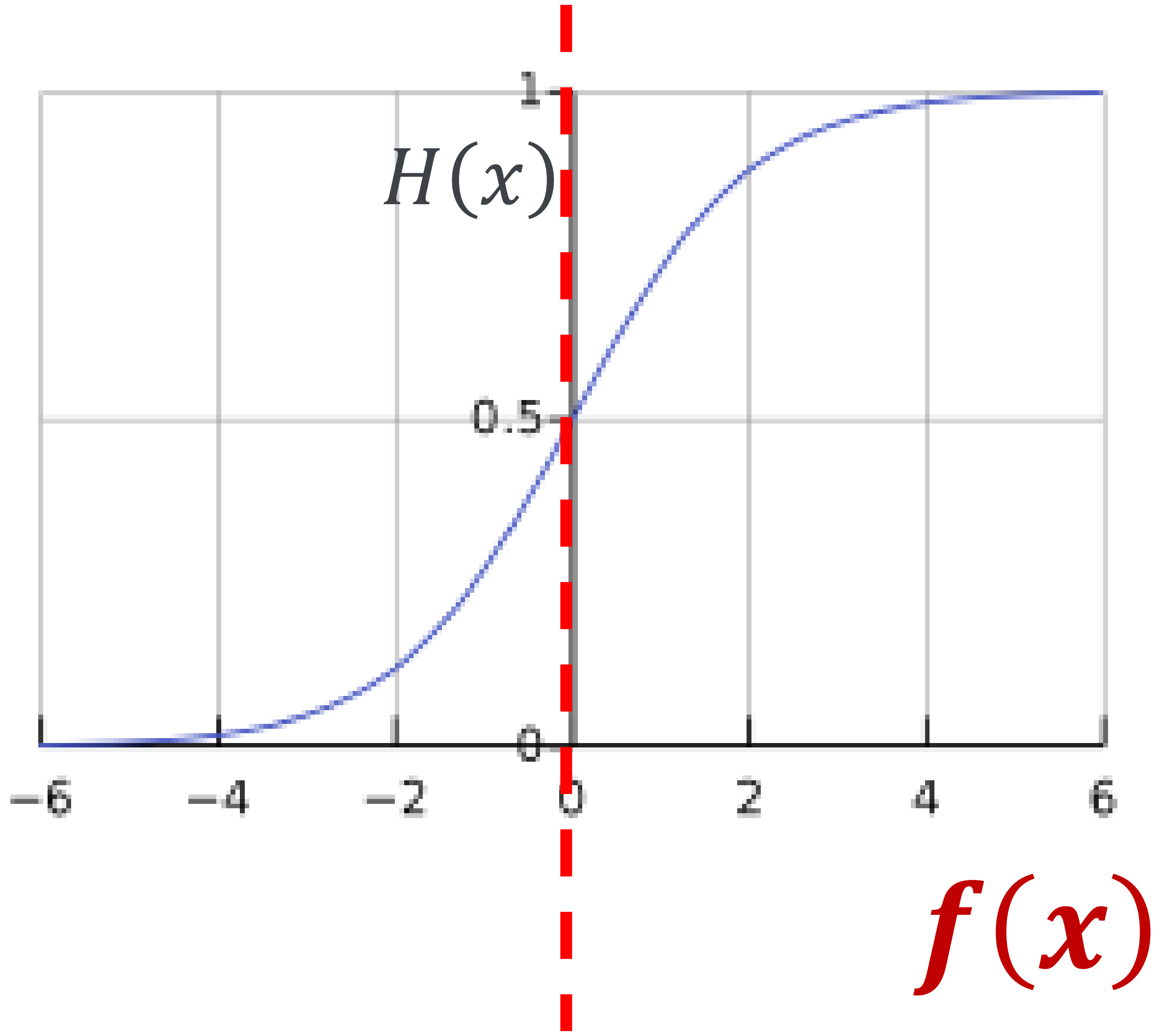
- 언제 y=1이라고 예측할까? ➔ $\boldsymbol{P(y=1|x)}$



$$H(x) \geq \mathbf{0.5} \qquad y = 1$$
$$H(x) < \mathbf{0.5} \qquad \Rightarrow y = 0$$

Sigmoid (Logistic) Function

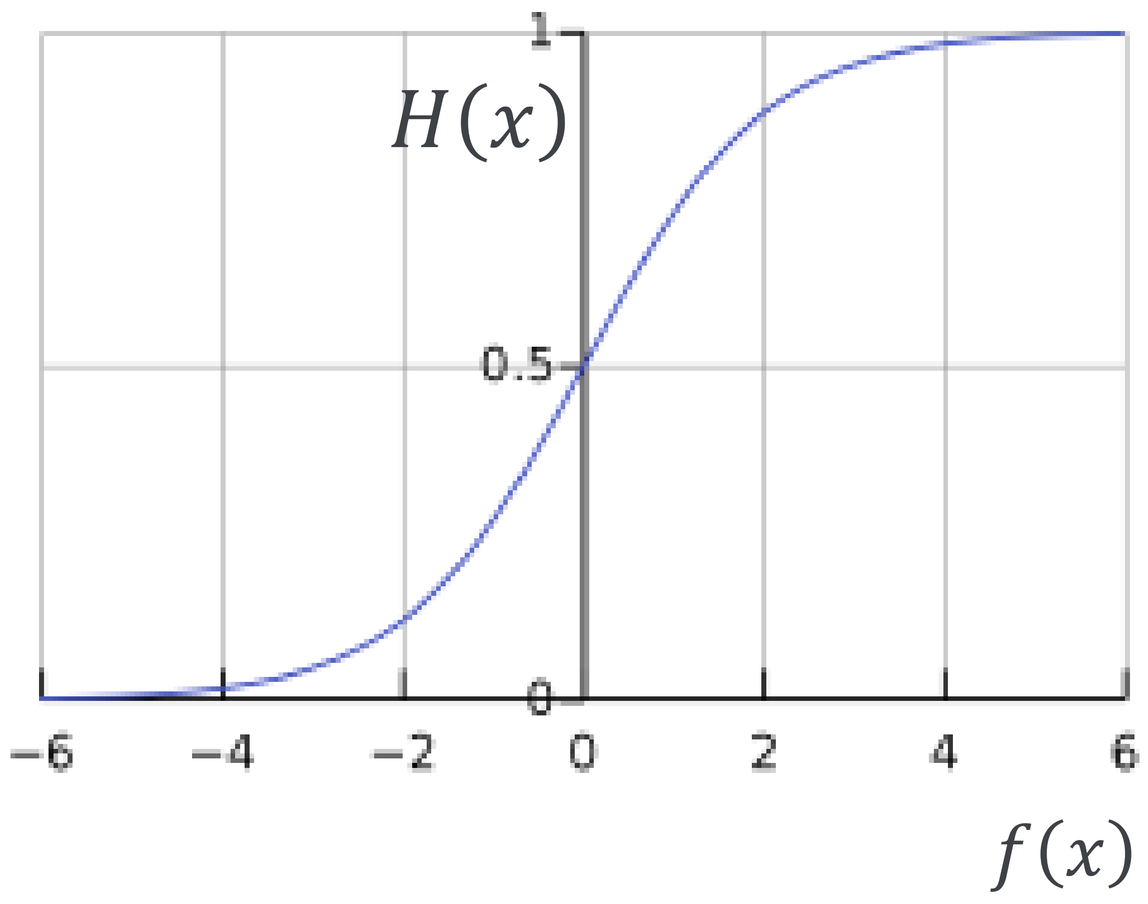
- 언제 y=1이라고 예측할까? ➔ $\boldsymbol{P(y = 1|x)}$



$$H(x) \geq 0.5 \Rightarrow \boldsymbol{f(x)} = w^T x \geq \mathbf{0} \Rightarrow y = 1$$
$$H(x) < 0.5 \Rightarrow \boldsymbol{f(x)} = w^T x < \mathbf{0} \Rightarrow y = 0$$

Decision Boundary

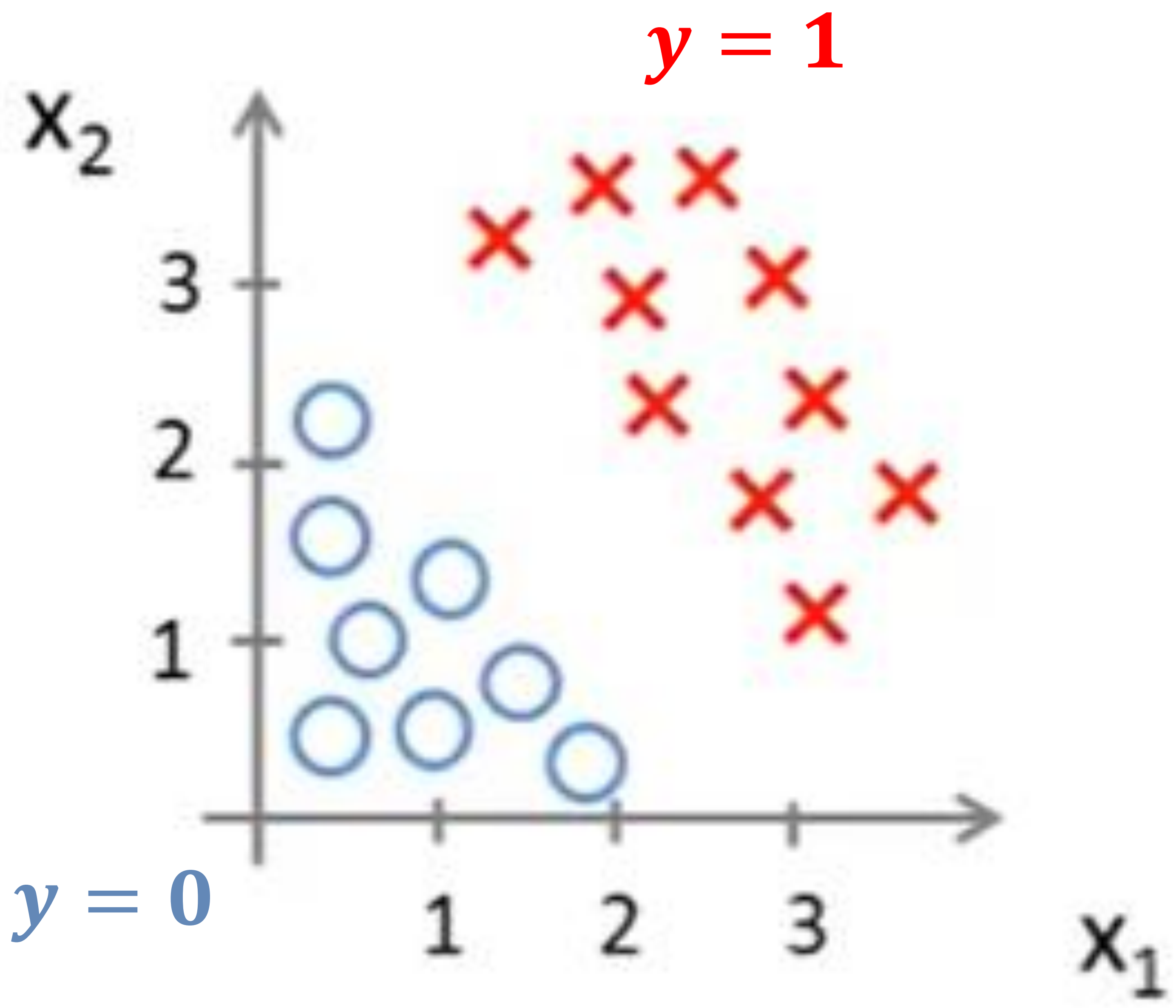
- Hypothesis function 의 값 0.5를 기준으로 분류



$$y = \mathbf{1} \quad \text{if } w^T x \geq \mathbf{0}$$
$$y = \mathbf{0} \quad \text{if } w^T x < \mathbf{0}$$

Linear Decision Boundary

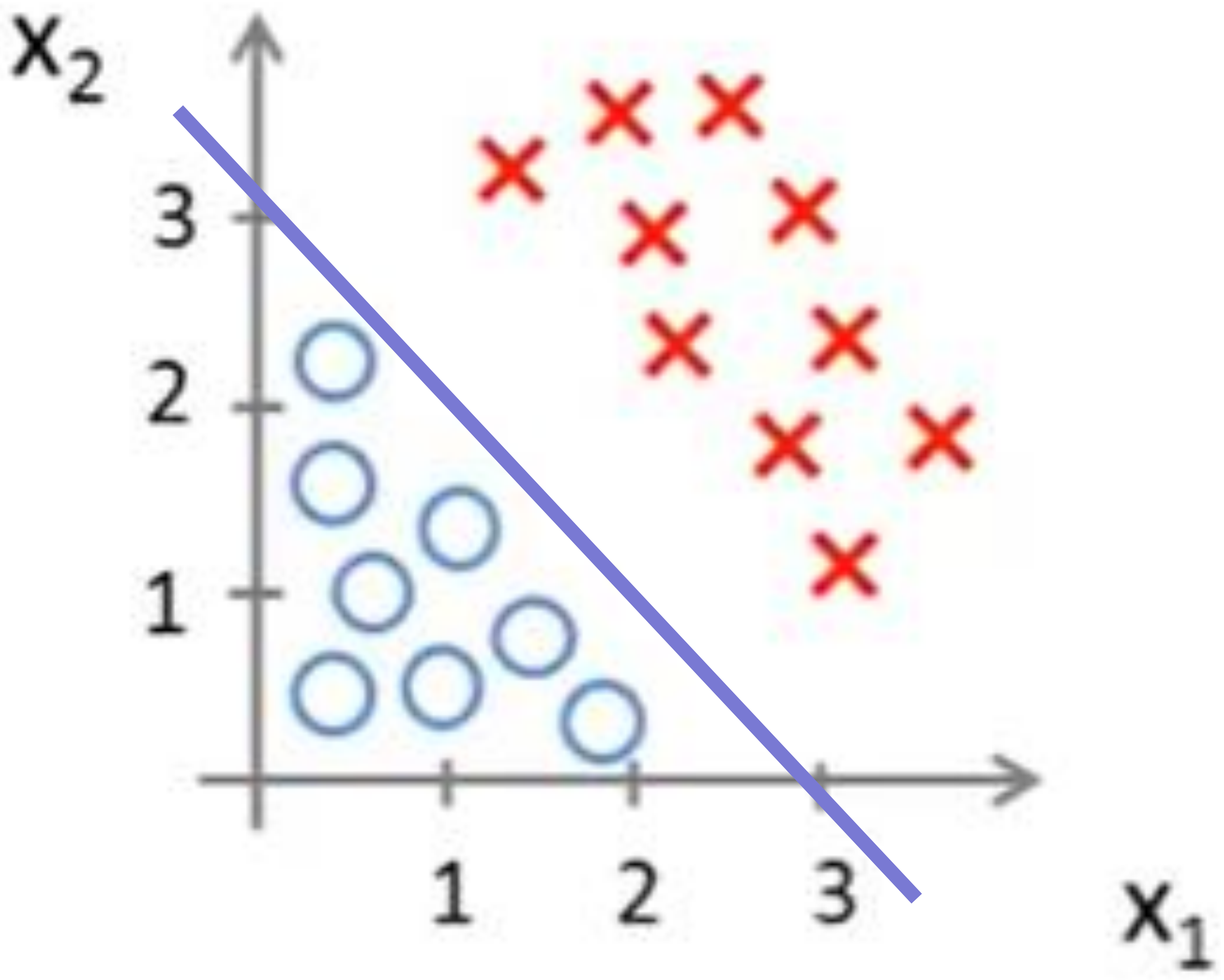
- 예시



$$y = \mathbf{1} \quad \text{if } f(x) = w_0 + w_1 x_1 + w_2 x_2 \geq \mathbf{0}$$

Linear Decision Boundary

- 예시

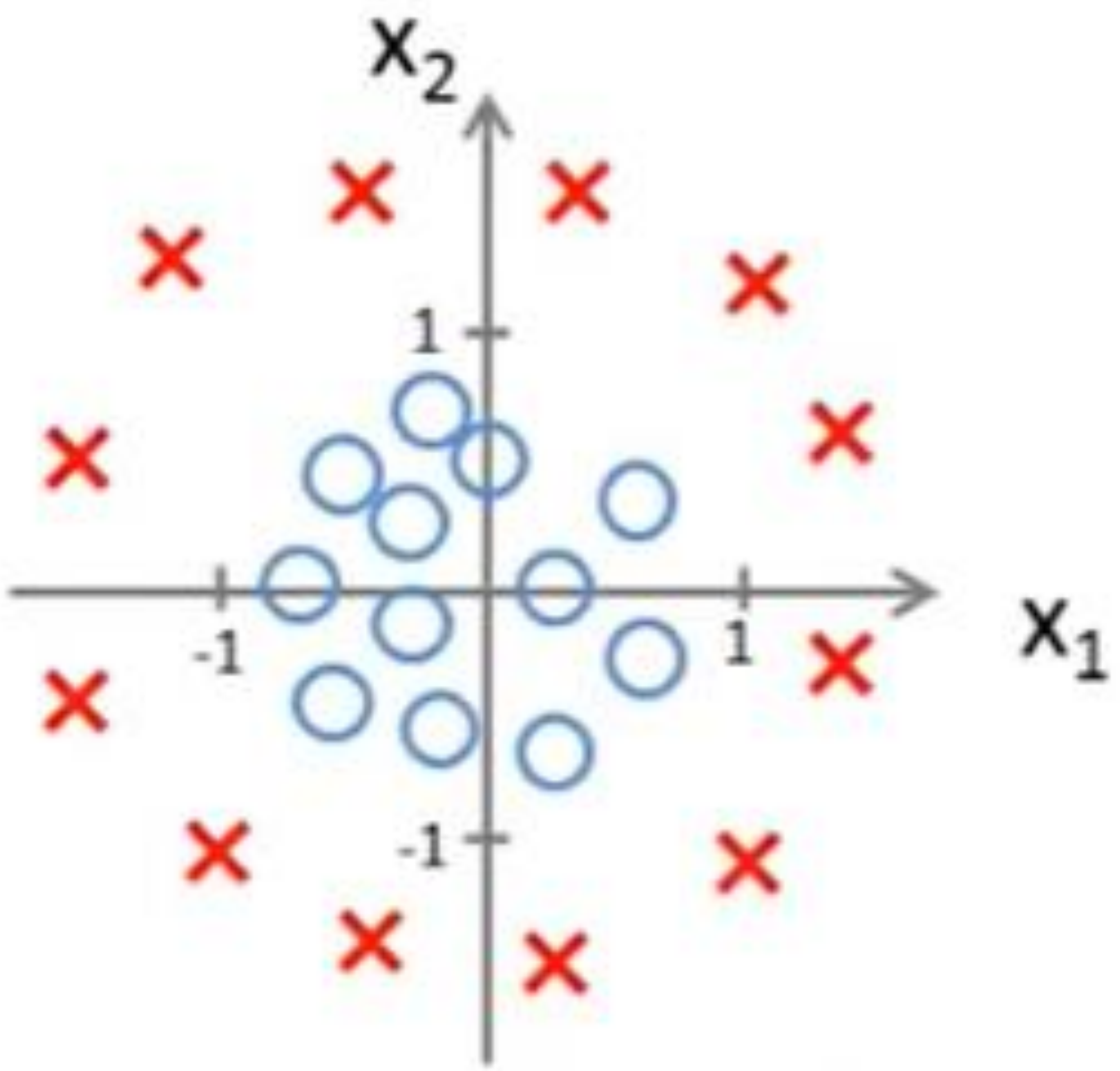


$$y = \mathbf{1} \quad \text{if } f(x) = w_0 + w_1 x_1 + w_2 x_2 \geq \mathbf{0}$$

$$w = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

Non-Linear Decision Boundary

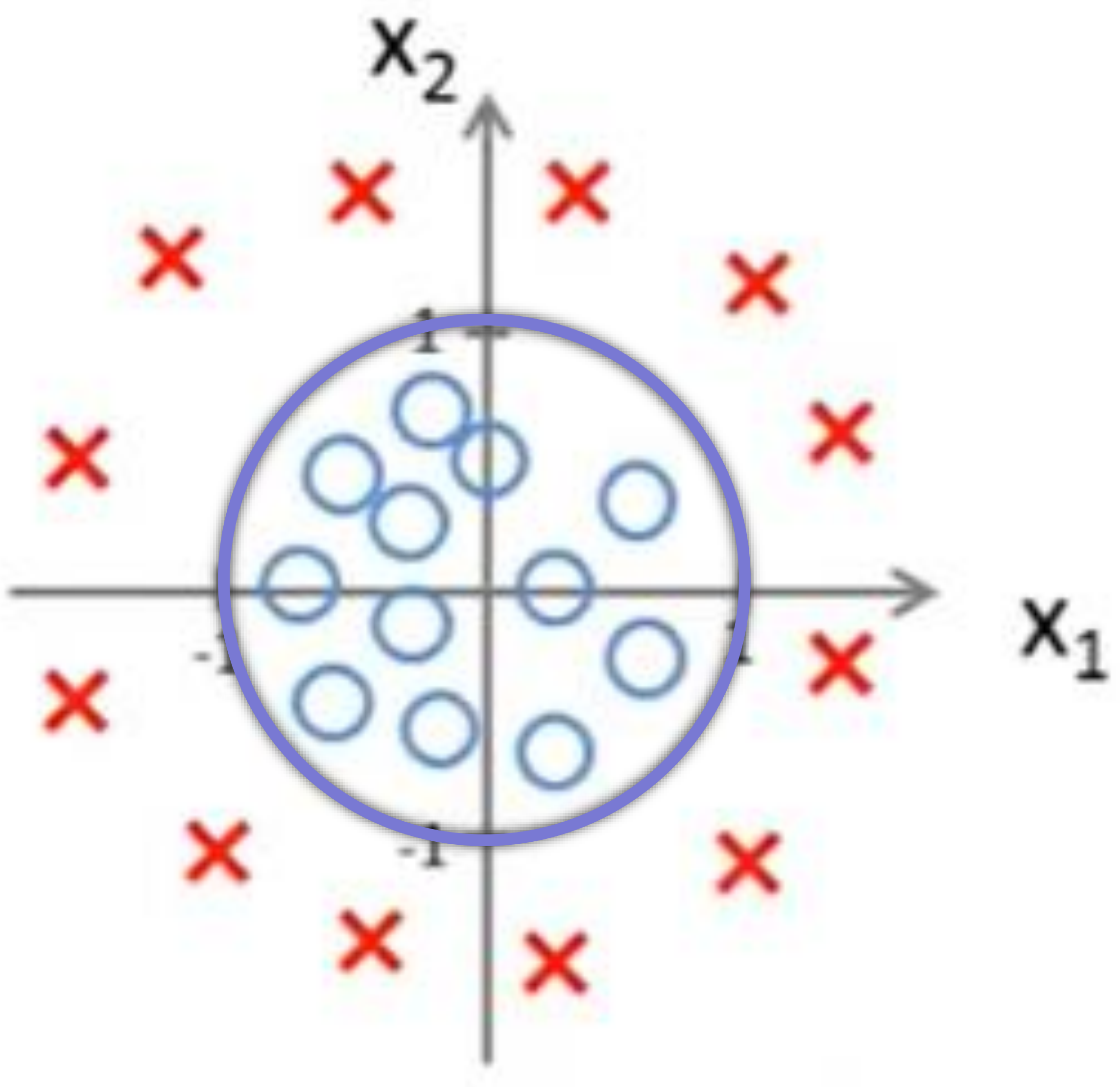
- 예시



$$y = \mathbf{1} \quad \text{if } f(x) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2$$

Non-Linear Decision Boundary

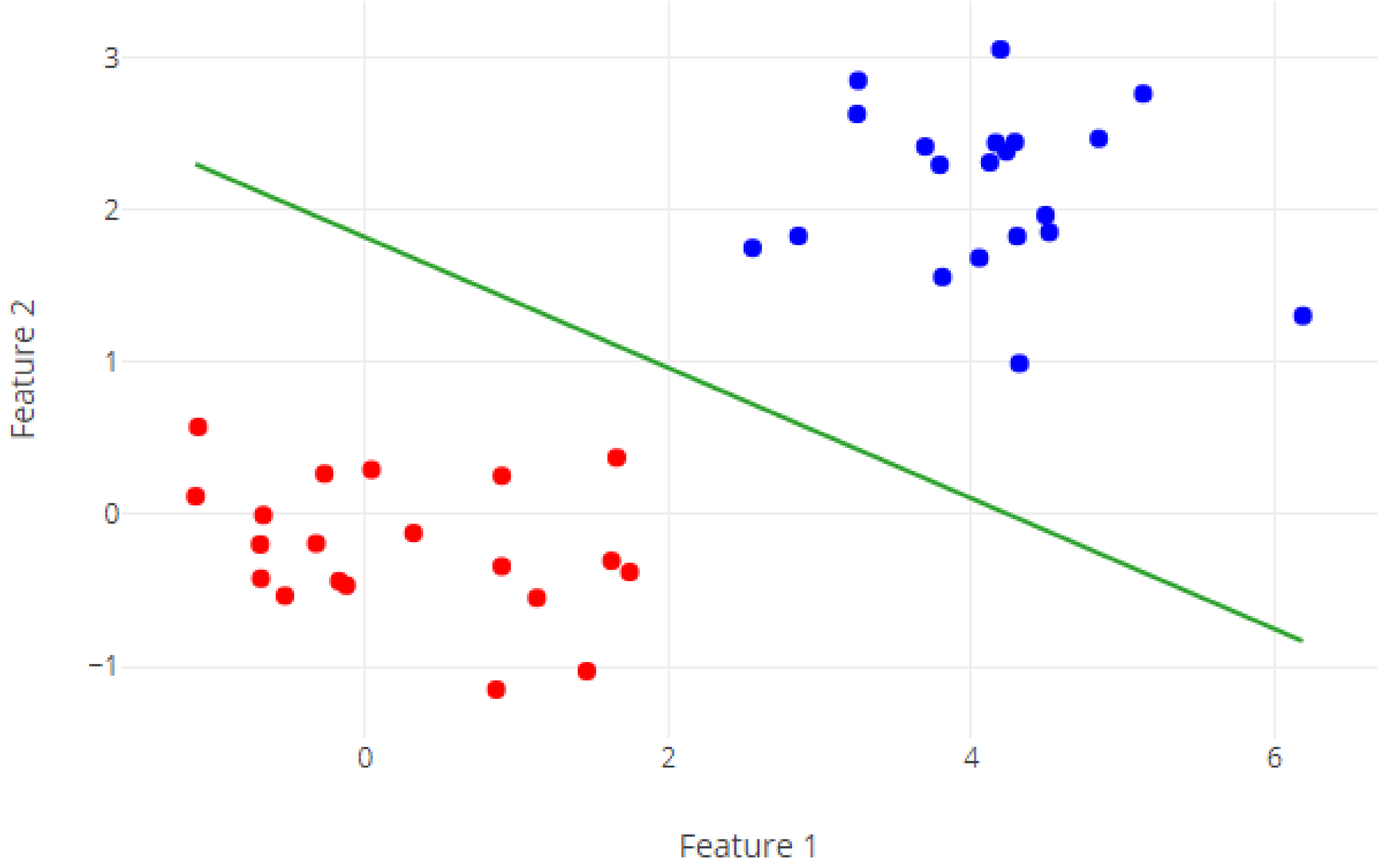
- 예시

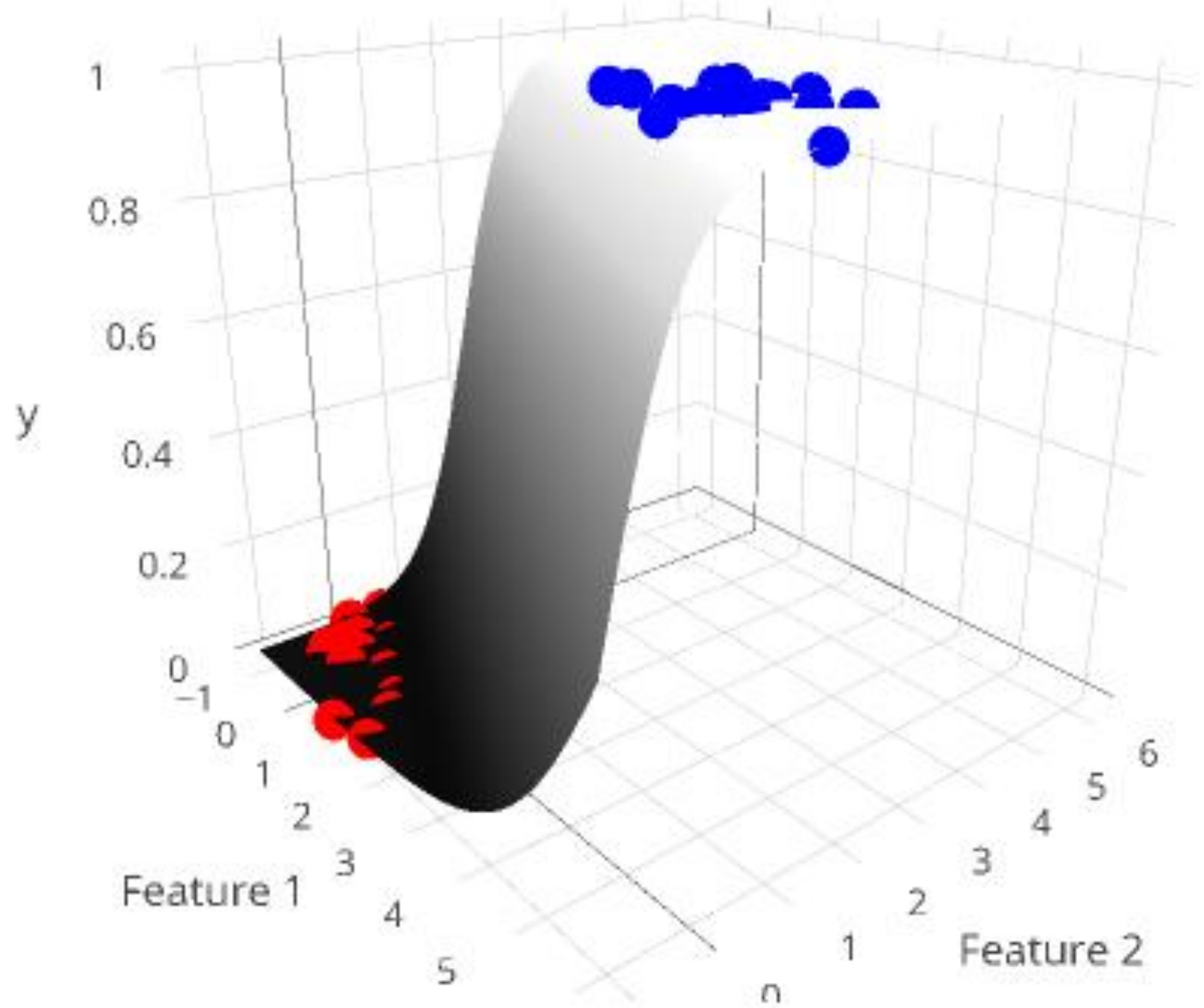


$$y = \mathbf{1} \quad \text{if } f(x) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2$$

$$w = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

Visualizing Geometric Interpretation

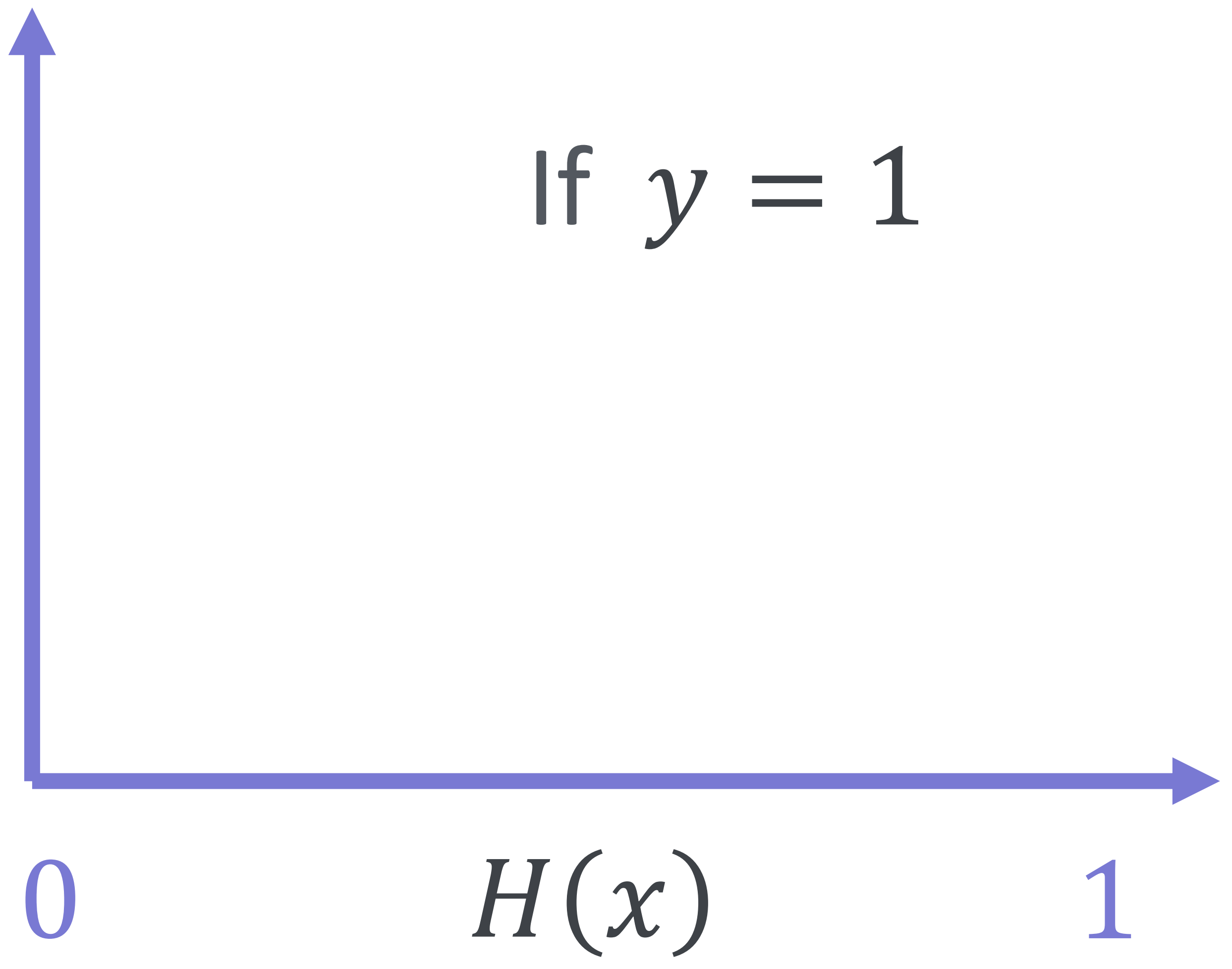


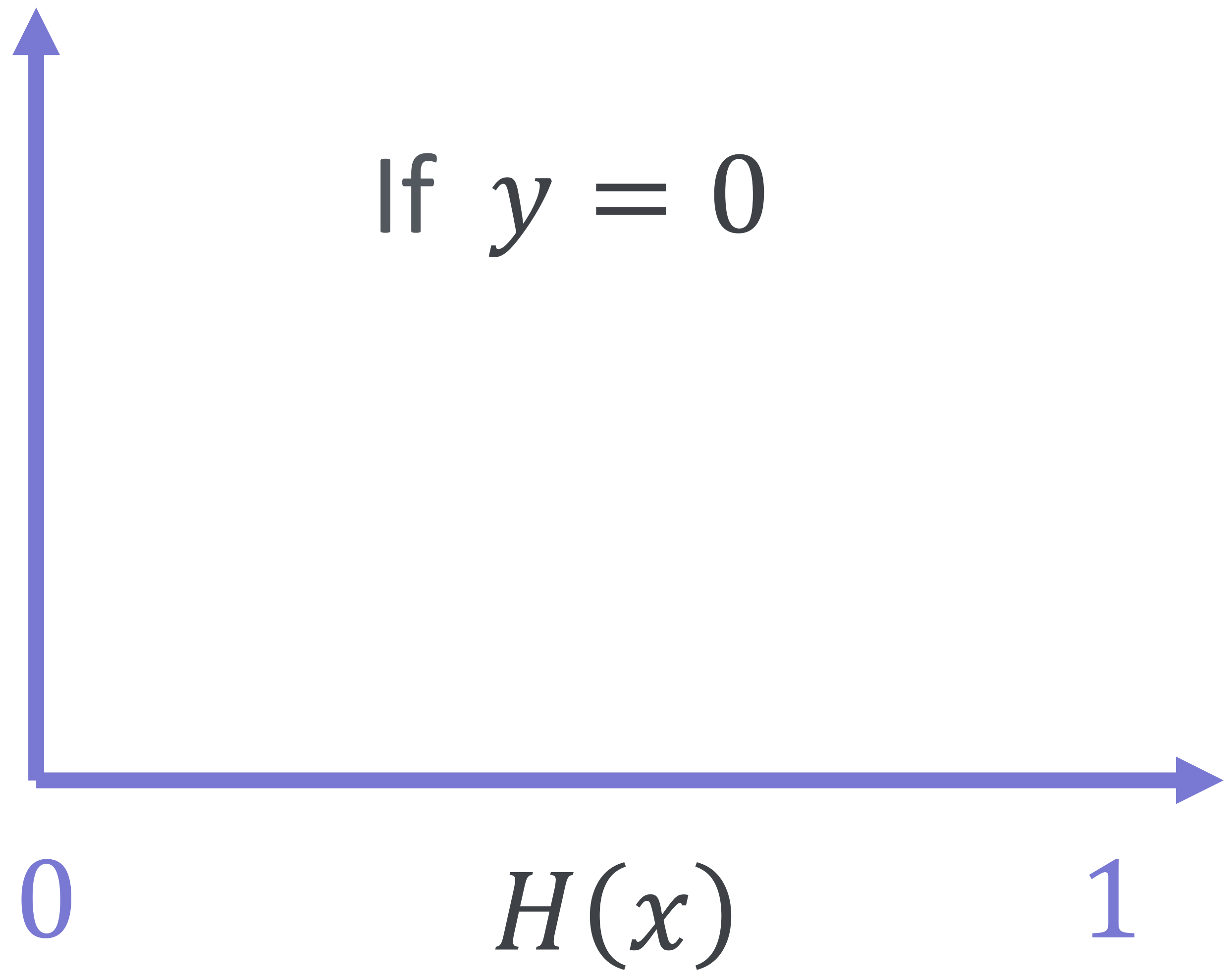


$$f(x) = w_0 + w_1 x_1 + w_2 x_2$$

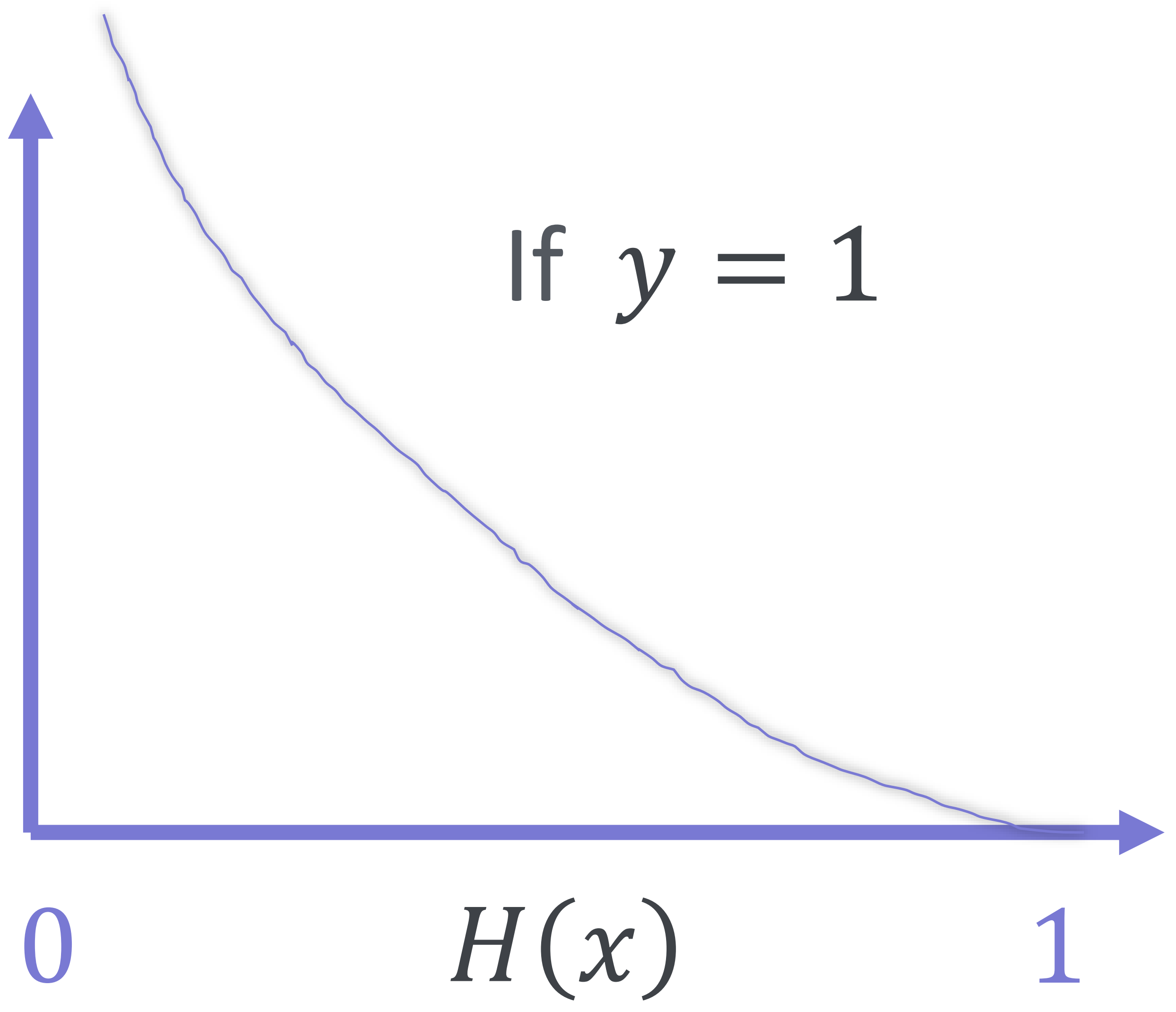
$$H(x) = \frac{1}{1 + e^{-f(x)}}$$

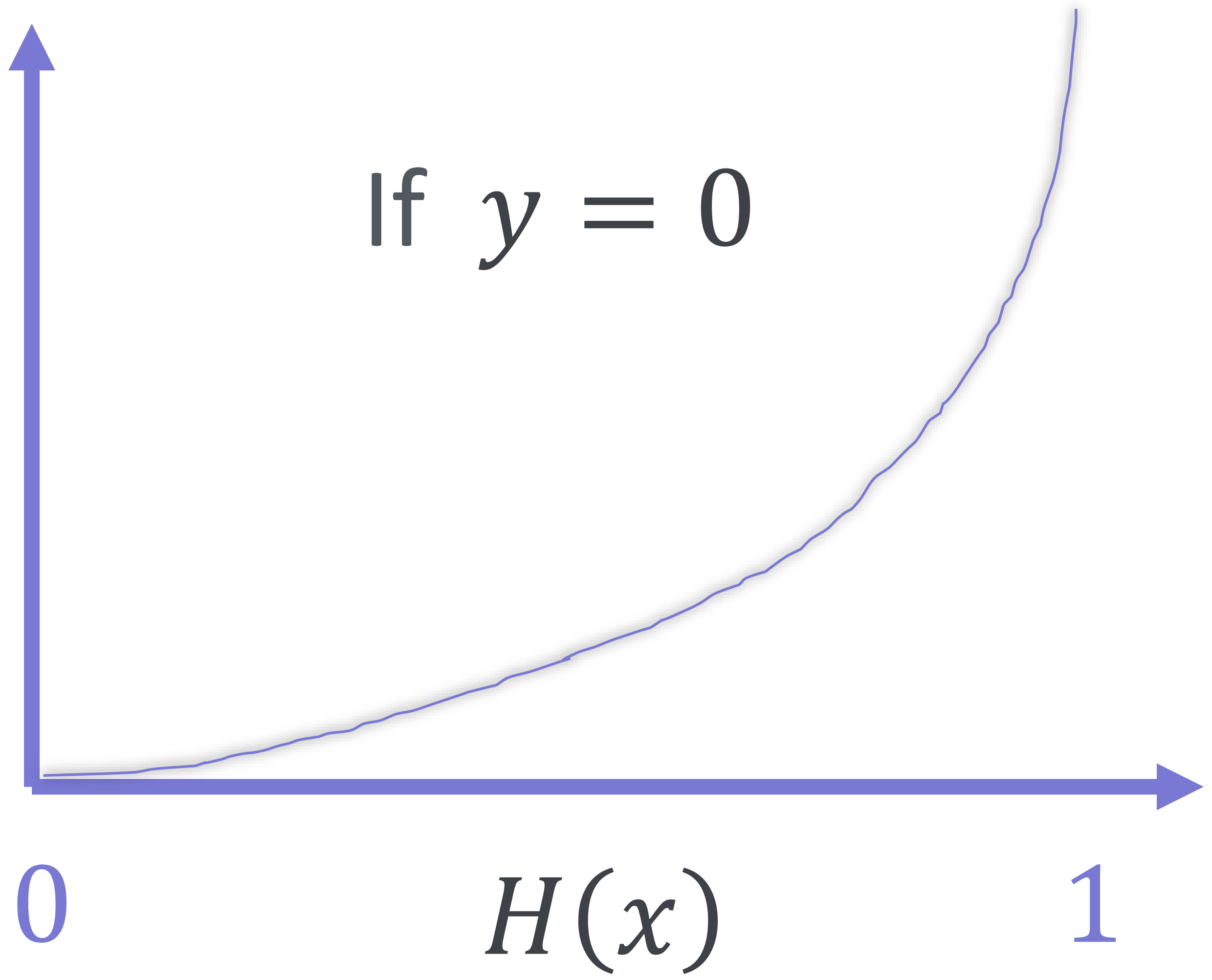
Cost (Loss) Function



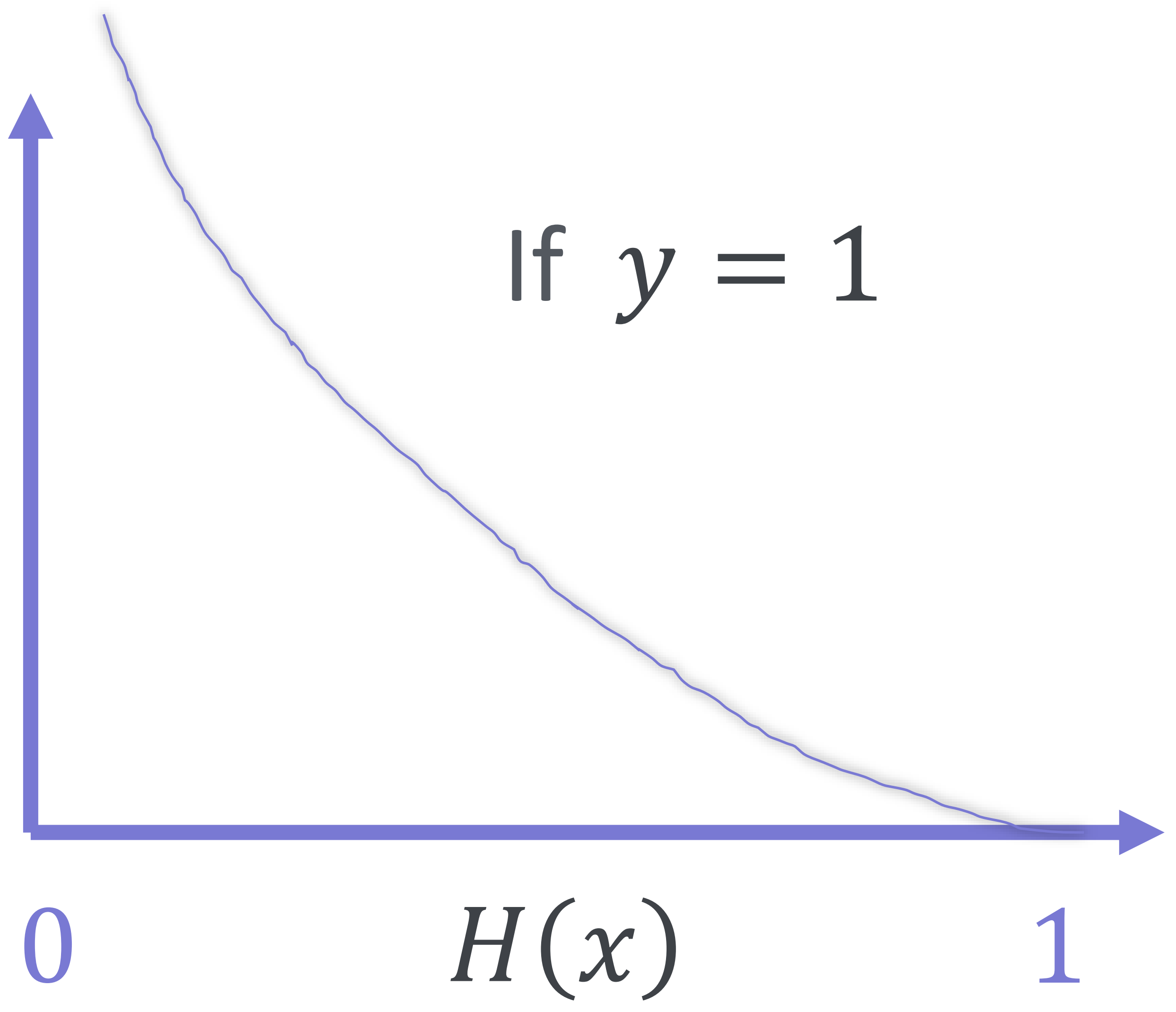


Cost (Loss) Function





Cost (Loss) Function



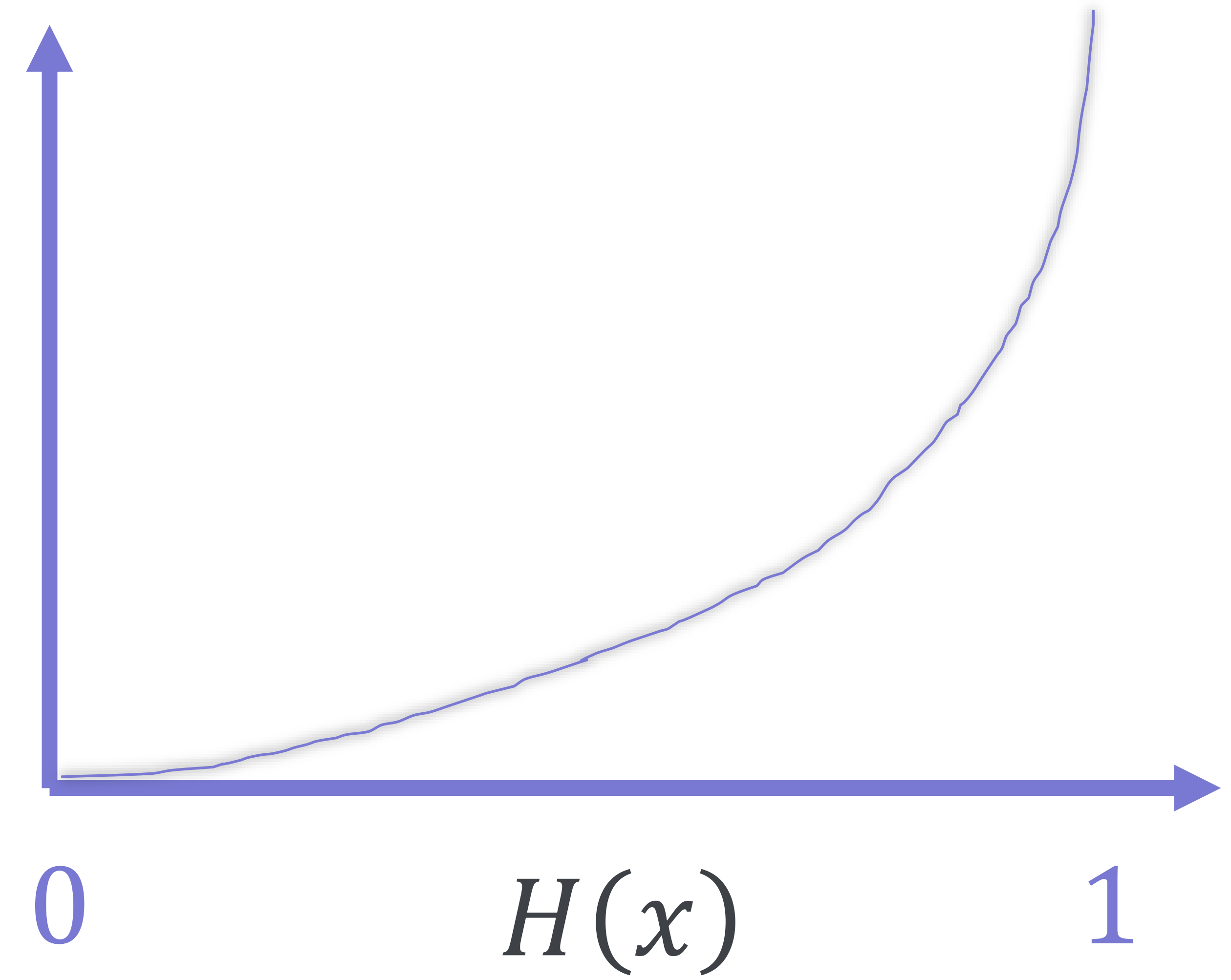
- 만약 <u>정답 y가 **1**번</u>이라면..
- 예측 H(x)가 1번일 때, Cost = 0
- 예측 H(x)가 0번일 때, Cost = ∞

$$\text{Cost}\,(\text{H(x)}, \text{y}) = -\log(\text{H(x)})$$

Cost (Loss) Function

- 만약 <u>정답 y가 **0**번</u>이라면..
- 예측 H(x)가 0번일 때, Cost = 0
- 예측 H(x)가 1번일 때, Cost = ∞

$$\mathrm{Cost}\,(\mathrm{H}(\mathrm{x}), \mathrm{y}) = -\log(1 - \mathrm{H}(\mathrm{x}))$$



Cost Function: Cross Entropy

$$\text{Cost} = \begin{cases} -\log(1 - H(X)) & (y = 0) \\ -\log(H(x)) & (y = 1) \end{cases}$$

CE Loss

$$\text{Cost}\,(H(x), y) = -\text{y}\log(H(x)) - (1 - y)\log(1 - H(X))$$

$$\Rightarrow \frac{1}{N}\sum_{i=1}^{N} -y_i \log(H(X_i)) - (1 - y_i)\log(1 - H(X_i))$$

Gradient Descent

$$w_j := w_j - \alpha \, * \left.\frac{\partial L}{\partial w}\right|_{w=w_j}$$

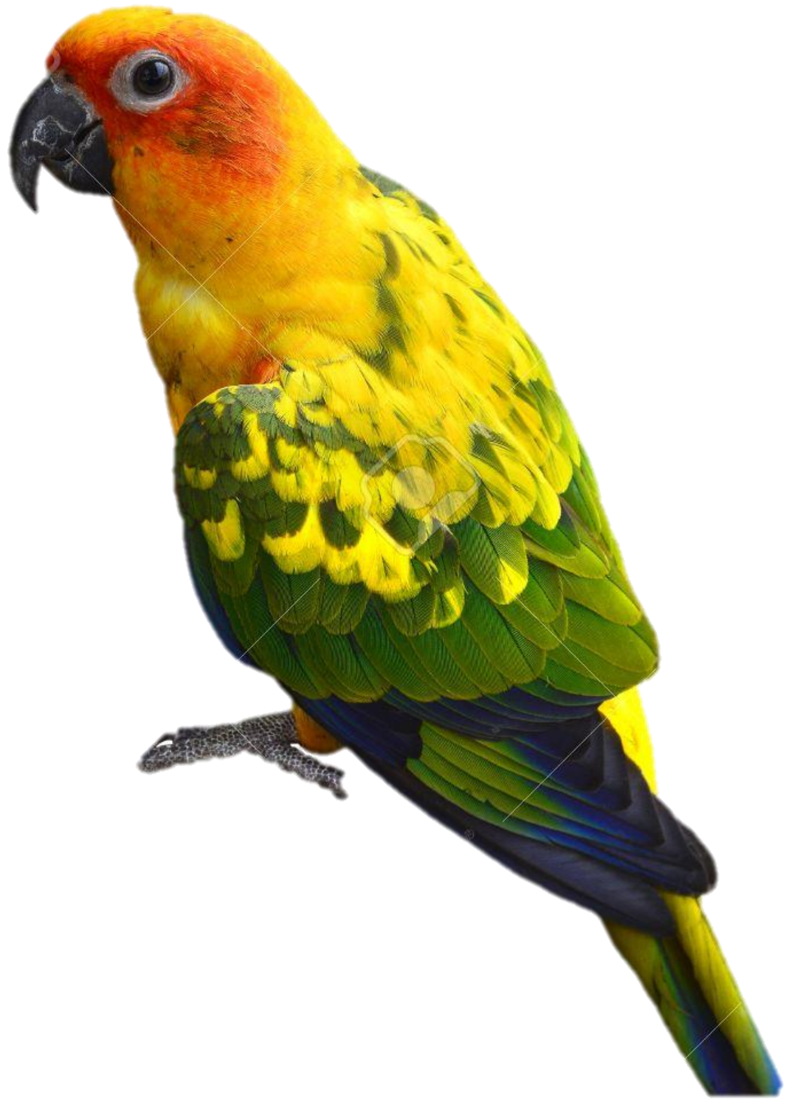
Gradient

$$\left.\frac{\partial L}{\partial w}\right|_{w=w_j} = \frac{1}{N}\sum_{i=1}^{N}(H(X_i) - y_i)X_{i,j}$$

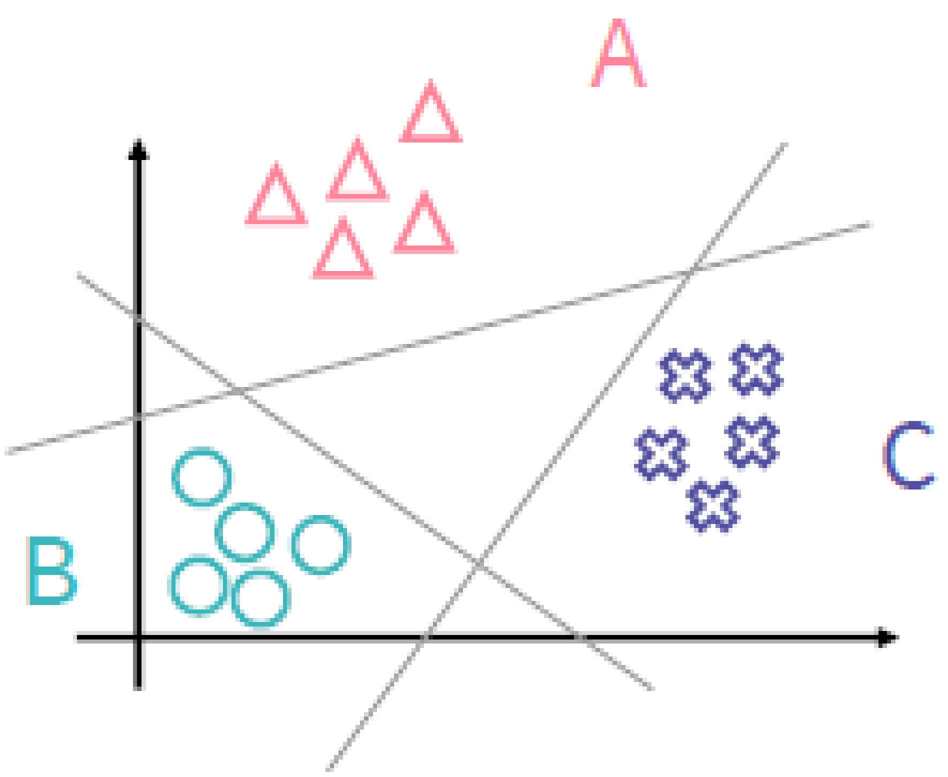
Multi-class Classification

VS

VS

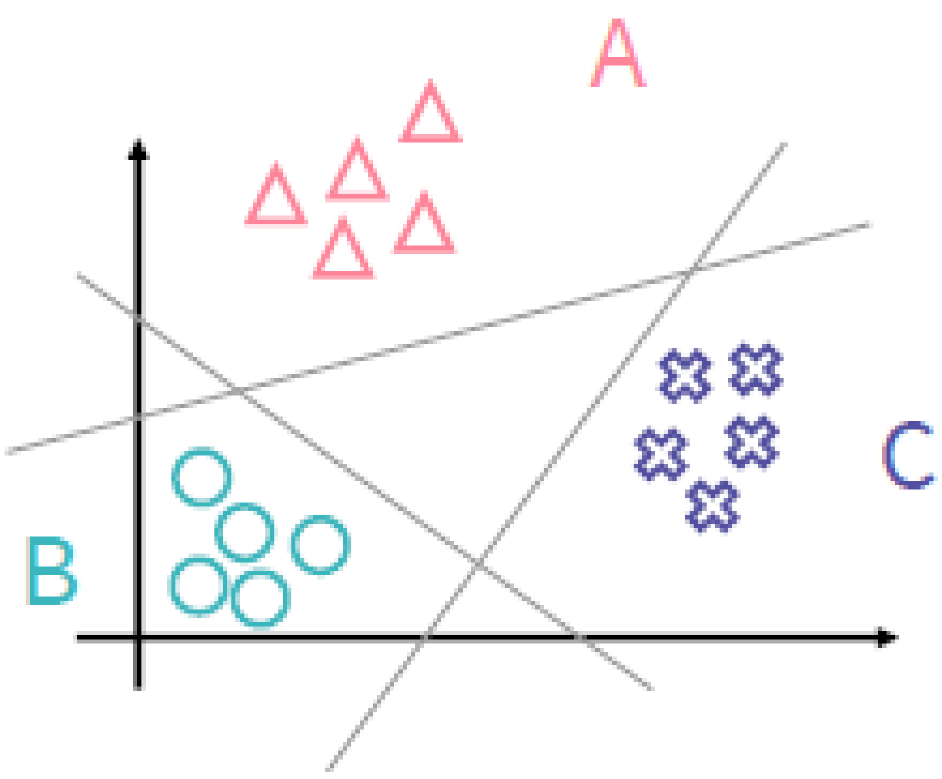


Multi-class Classification



- A인가 아닌가? -> A 선별
- B인가 아닌가? -> B 선별
- C인가 아닌가? -> C 선별

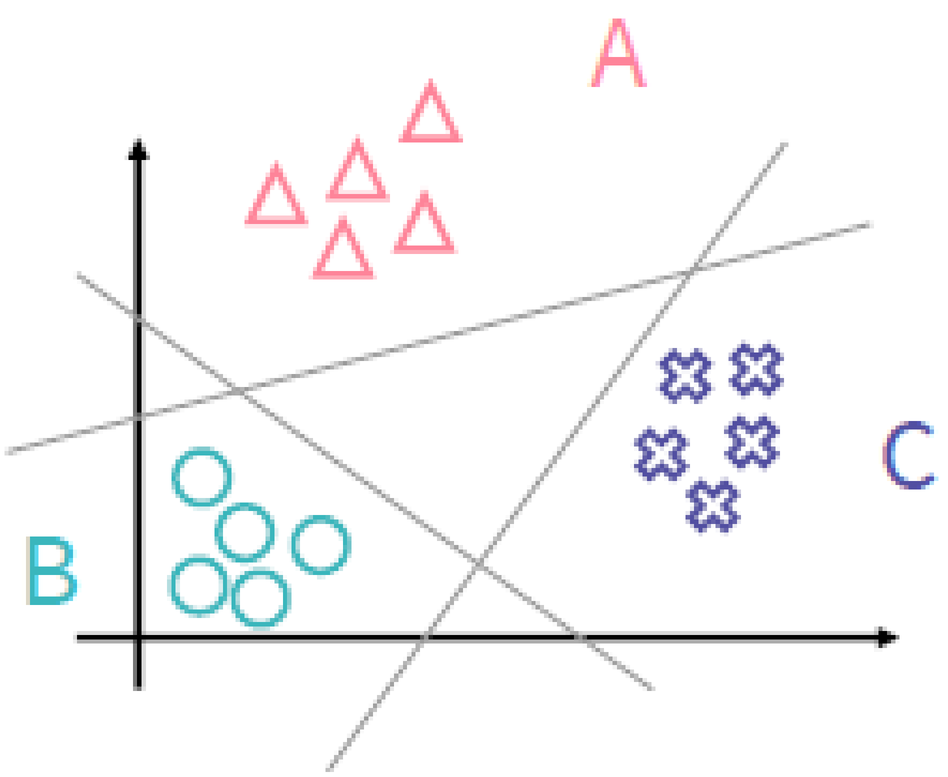
Multi-class Classification



- A인가 아닌가? -> A 선별
- B인가 아닌가? -> B 선별
- C인가 아닌가? -> C 선별

$$w_1 = \begin{bmatrix} w_{1,0} \\ w_{1,1} \\ w_{1,2} \end{bmatrix} \quad w_2 = \begin{bmatrix} w_{2,0} \\ w_{2,1} \\ w_{2,2} \end{bmatrix} \quad w_3 = \begin{bmatrix} w_{3,0} \\ w_{3,1} \\ w_{3,2} \end{bmatrix}$$

Multi-class Classification



- A인가 아닌가? -> A 선별
- B인가 아닌가? -> B 선별
- C인가 아닌가? -> C 선별

$$w_1 = \begin{bmatrix} w_{1,0} \\ w_{1,1} \\ w_{1,2} \end{bmatrix} \quad w_2 = \begin{bmatrix} w_{2,0} \\ w_{2,1} \\ w_{2,2} \end{bmatrix} \quad w_3 = \begin{bmatrix} w_{3,0} \\ w_{3,1} \\ w_{3,2} \end{bmatrix}$$

$$\begin{bmatrix} w_{1,0} & w_{1,1} & w_{1,2} \\ w_{2,0} & w_{2,1} & w_{2,2} \\ w_{3,0} & w_{3,1} & w_{3,2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \widehat{y_0} \\ \widehat{y_1} \\ \widehat{y_2} \end{bmatrix}$$

Softmax Regression 과정

$$\begin{bmatrix} w_{1,0} & w_{1,1} & w_{1,2} \\ w_{2,0} & w_{2,1} & w_{2,2} \\ w_{3,0} & w_{3,1} & w_{3,2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \widehat{y_0} \\ \widehat{y_1} \\ \widehat{y_2} \end{bmatrix}$$

Softmax Regression 과정

$$\begin{bmatrix} w_{1,0} & w_{1,1} & w_{1,2} \\ w_{2,0} & w_{2,1} & w_{2,2} \\ w_{3,0} & w_{3,1} & w_{3,2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \widehat{y_0} \\ \widehat{y_1} \\ \widehat{y_2} \end{bmatrix} = \begin{bmatrix} 2.0 \\ 1.0 \\ 0.1 \end{bmatrix}$$

Softmax Regression 과정

Softmax

$$\begin{bmatrix} w_{1,0} & w_{1,1} & w_{1,2} \\ w_{2,0} & w_{2,1} & w_{2,2} \\ w_{3,0} & w_{3,1} & w_{3,2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \widehat{y_0} \\ \widehat{y_1} \\ \widehat{y_2} \end{bmatrix} = \begin{bmatrix} 2.0 \\ 1.0 \\ 0.1 \end{bmatrix} \rightarrow \boxed{\frac{e^{y_i}}{\sum_j e^{y_j}}} \rightarrow \begin{bmatrix} 0.7 \\ 0.1 \\ 0.2 \end{bmatrix}$$

Softmax Regression 과정

Softmax

$$\begin{bmatrix} w_{1,0} & w_{1,1} & w_{1,2} \\ w_{2,0} & w_{2,1} & w_{2,2} \\ w_{3,0} & w_{3,1} & w_{3,2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \widehat{y_0} \\ \widehat{y_1} \\ \widehat{y_2} \end{bmatrix} = \begin{bmatrix} 2.0 \\ 1.0 \\ 0.1 \end{bmatrix} \rightarrow \boxed{\frac{e^{y_i}}{\sum_j e^{y_j}}} \rightarrow \begin{bmatrix} 0.7 \\ 0.1 \\ 0.2 \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{1} \\ 0 \\ 0 \end{bmatrix}$$

One-Hot encoding

Softmax 구현 시 주의점

$$\frac{e^{y_i}}{\sum_j e^{y_j}}$$

Example

```
import numpy as np

def softmax(x):

    e_x = np.exp(x)
    return  e_x/np.sum(e_x)
```

```
import numpy as np

def revised_softmax(x):

    e_x = np.exp(x – np.max(x))
    return  e_x/np.sum(e_x)
```

Cost Function: Cross Entropy

- 두 변수들의 확률 분포가 얼마나 비슷한지 나타냄

CE Loss

$$\text{Cost}\,(H(x), y) = -\frac{1}{N}\sum_{i=1}^{N} y_i \log\big(H(X_i)\big)$$

$$y_i \rightarrow \begin{bmatrix} \mathbf{1} \\ 0 \\ 0 \end{bmatrix} \qquad H(X_i) \rightarrow \begin{bmatrix} \mathbf{0.7} \\ 0.1 \\ 0.2 \end{bmatrix}$$

03

Evaluation Metrics

Overview

- Confusion Matrix
- Accuracy
- Precision
- Recall
- F-measure
- Average Precision (AP)
- Mean Average Precision (MAP)

Confusion Matrix (혼동행렬)

		Actual	
		Positive	**Negative**
Predicted	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FP)	True Negative (TN)

Confusion Matrix (혼동행렬)

Actual

		Positive	**Negative**
Predicted	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FP)	True Negative (TN)

Confusion Matrix (혼동행렬)

Actual

Predicted

	Positive	Negative
Positive	True Positive (TP)	False Positive (FP)
Negative	False Negative (FP)	True Negative (TN)

Confusion Matrix (혼동행렬)

True Negative

Confusion Matrix (혼동행렬)

True Negative

예측 값

Confusion Matrix (혼동행렬)

True Negative

맞음! 예측 값

Confusion Matrix (혼동행렬)

False Negative

틀림.. 예측 값

Recap: Overview

- Confusion Matrix
- **Accuracy**
- **Precision**
- **Recall**
- **F-measure**
- Average Precision (AP)
- Mean Average Precision (MAP)

Accuracy (정확도)

The fractions of these classifications that are correct

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

Equation

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

Example of Accuracy

Given a query, classify each document as “**Relevant**” or “**Non-Relevant**”

Document	Actual	Predicted
Doc 1	+	+
Doc 2	-	-
Doc 3	-	+
Doc 4	-	+
Doc 5	-	-

	Relevant (+)	Non-Relevant (-)
Relevant		
Non-Relevant		

Accuracy = ?

Example of Accuracy

Given a query, classify each document as "**Relevant**" or "**Non-Relevant**"

Document	Actual	Predicted
Doc 1	+	+
Doc 2	-	-
Doc 3	-	+
Doc 4	-	+
Doc 5	-	-

	Relevant (+)	Non-Relevant (-)
Relevant	1	2
Non-Relevant	0	2

Accuracy = 3/5

Why not just use Accuracy?

만약 class가 imbalanced 되었을 때, (실제 IR에서는 99.9%의 Document가 non-relevant)
무조건 non-relevant하다고 예측한다면 Accuracy는 높지만, 실제로는 쓸모 없다.

Document	Actual	Predicted 1	Predicted 2
Doc 1	+	+	-
Doc 2	-	-	-
Doc 3	-	+	-
Doc 4	-	-	-
Doc 5	-	-	-

Precision (정밀도)

예측한 (+) 것 중에 실제 (+)인 것의 비율 = 모델의 입장

	Positive	Negative
Positive	True Positive (TP)	False Positive (FP)

Equation

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

Recall (재현율)

실제 (+) 것 중에 예측한 (+)의 비율 = 데이터의 입장

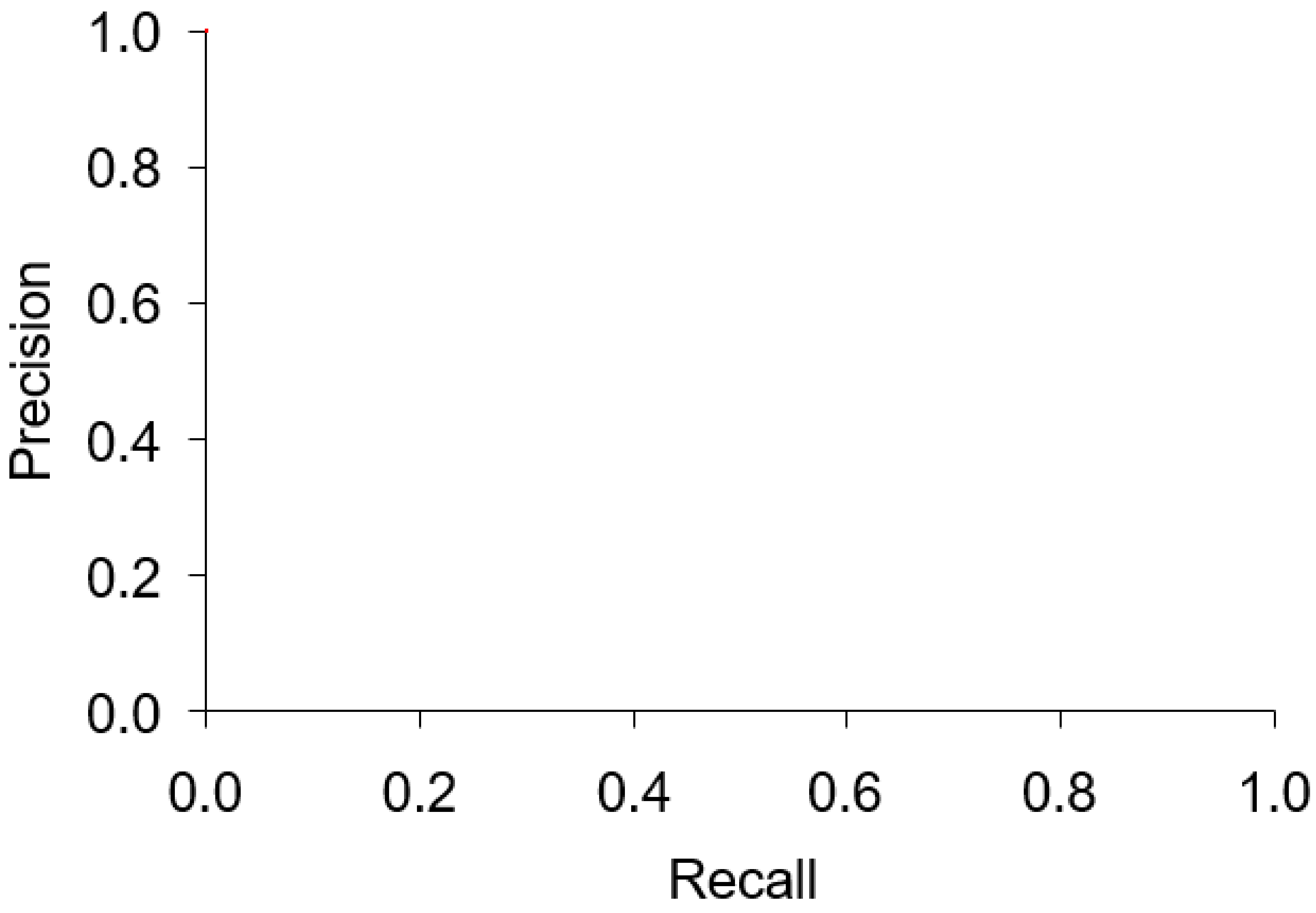
	Positive
Positive	True Positive (TP)
Negative	False Negative (FN)

Equation

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

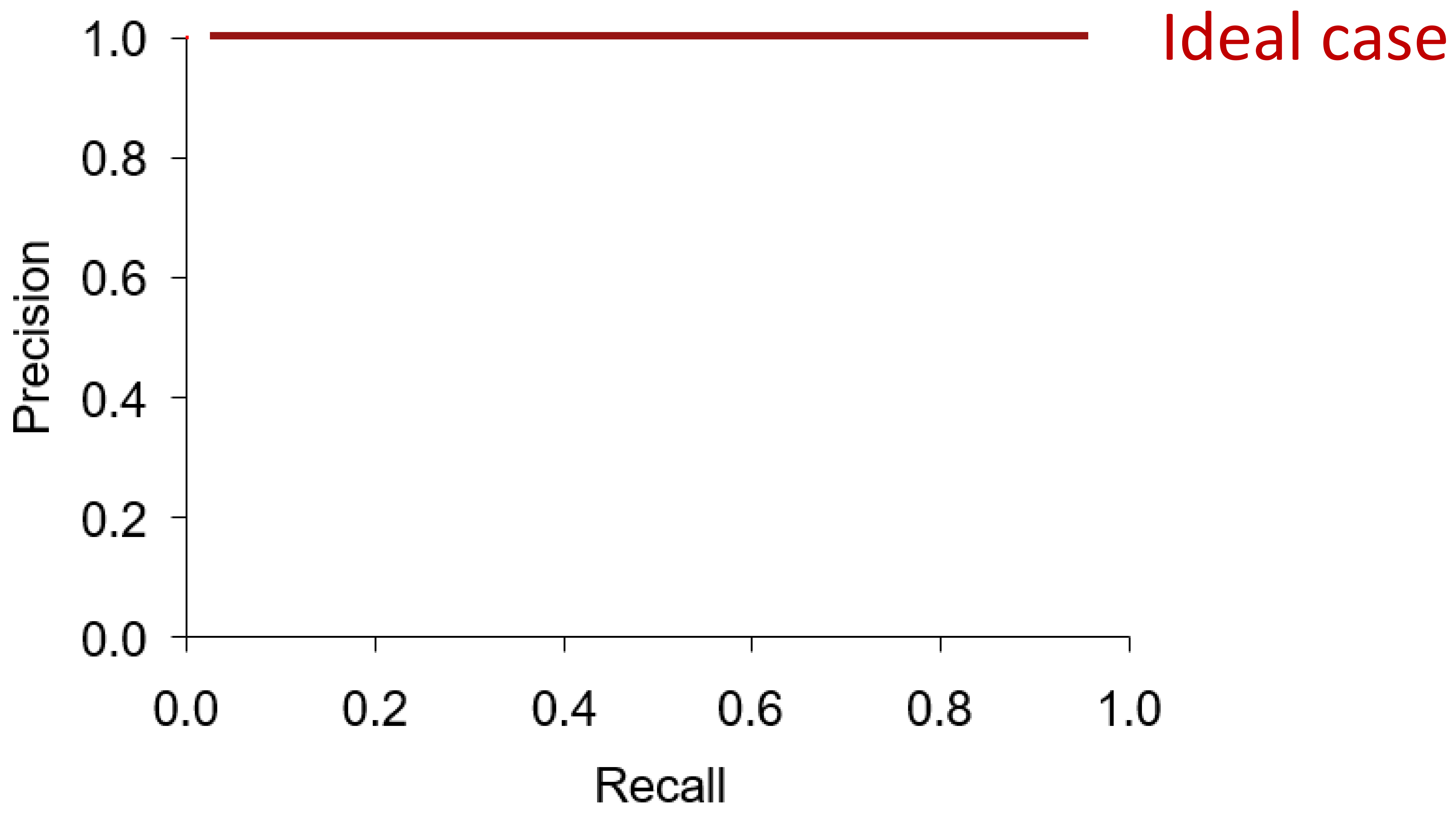
Precision-Recall Curve (PR-curve)

- X축을 **Recall**, y축을 **Precision**으로 하여 시각화한 그래프
- 보통 데이터가 불균형 (imbalanced) 될 때, 사용



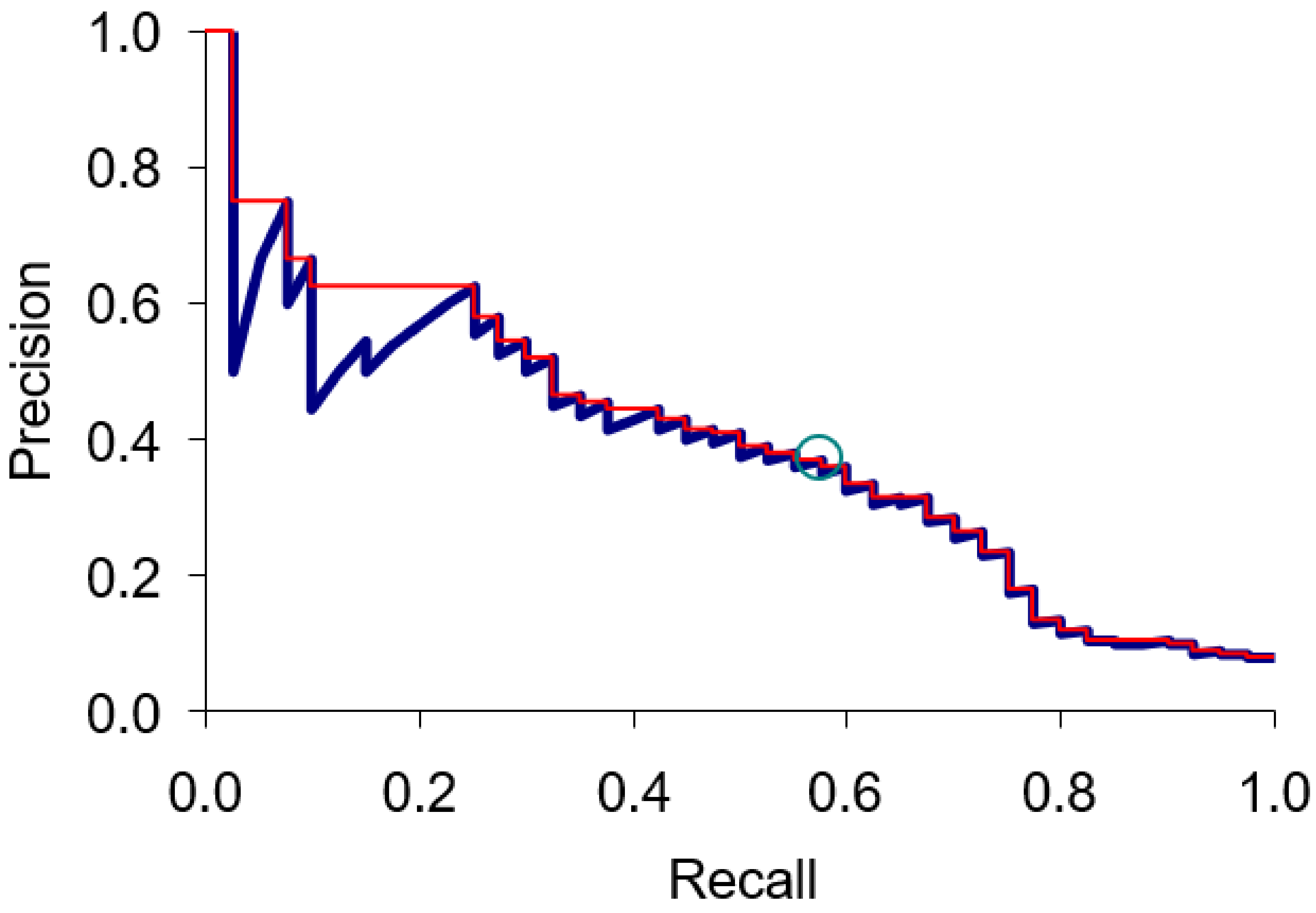
Precision-Recall Curve (PR-curve)

- X축을 **Recall**, y축을 **Precision**으로 하여 시각화한 그래프
- 보통 데이터가 불균형 (imbalanced) 될 때, 사용



Precision-Recall Curve (PR-curve)

- 다음으로 예측한 값이 **정답**에 속할 때, Precision은 (   ), Recall은 (   )이다.
- 다음으로 예측한 값이 **오답**에 속할 때, Precision은 (   ), Recall은 (   )이다.



F-measure

The weighted harmonic mean (조화평균) of Precision and Recall

Equation

$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} \qquad F_1 = \frac{2PR}{P+R} \quad (\text{when} \quad \alpha = 0.5)$$

F-measure

The weighted harmonic mean (조화평균) of Precision and Recall

Q. Why not just use arithmetic mean(산술평균)?

- 산술평균(P, R) >= 기하평균(P, R) >= 조화평균(P, R) >= min(P, R)
- P와 R이 많이 차이가 날 때, 조화평균은 mean(P, R)보다 min(P, R)에 가까워진다.

Example of Precision, Recall, F1

실제 관련 있는 문서 (정답) = 3개 (+)

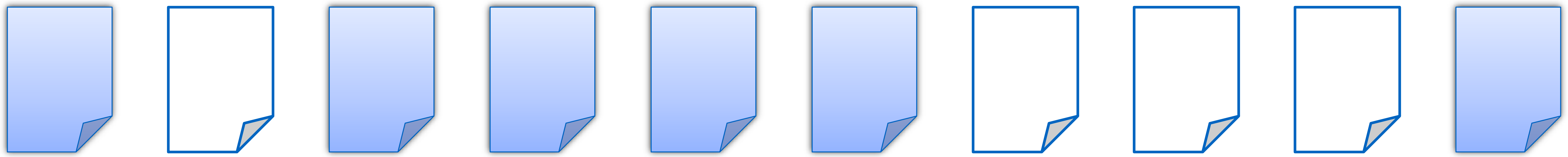
Model 1	Model 2
+	+
+	-
-	-
+	+
-	-

Q. Model 1 & Model 2 의 Precision, Recall, F1 을 계산해보세요!

Average Precision (AP)

The average of the precision values from the rank positions where a relevant document was retrieved (=when recall increases)

Model 1



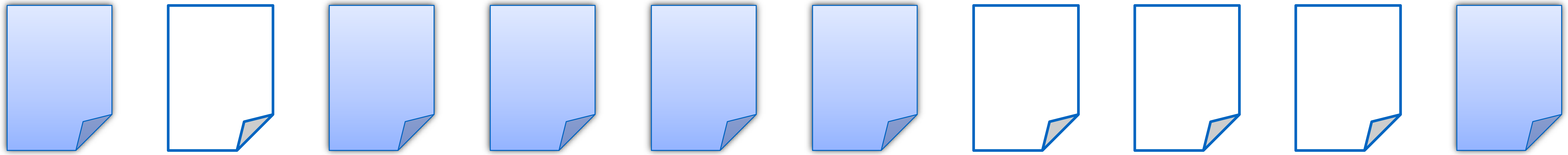
Precision										

Average Precision (AP) =

Average Precision (AP)

The average of the precision values from the rank positions where a relevant document was retrieved (=when recall increases)

Retrieved



Precision	**1.00**	**0.50**	**0.67**	**0.75**	**0.80**	**0.83**	**0.71**	**0.63**	**0.56**	**0.60**

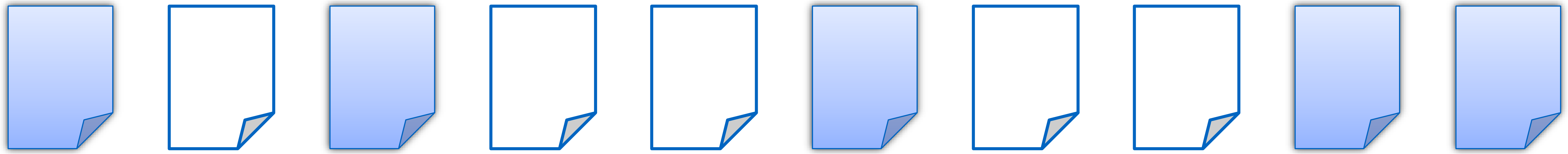
Average Precision (AP) = (1.00 + 0.67 + 0.75 + 0.80 + 0.83 + 0.60) / 6 = 0.78

Mean Average Precision (MAP)

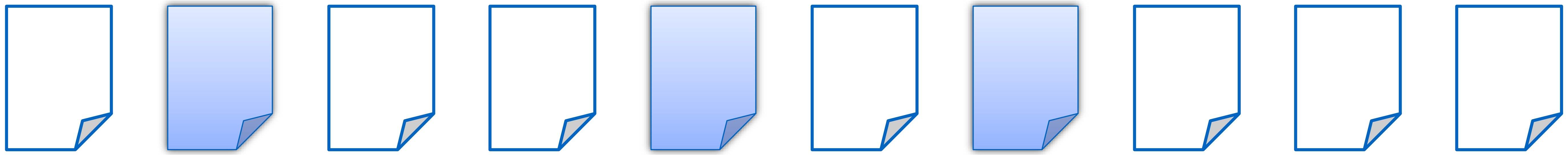
The average of the average precision on each query

$$\text{MAP} = \text{Average}(\ \text{AP}(\text{Q1}), \text{AP}(\text{Q2})\ )$$

Query 1: # of relevant docs = 5



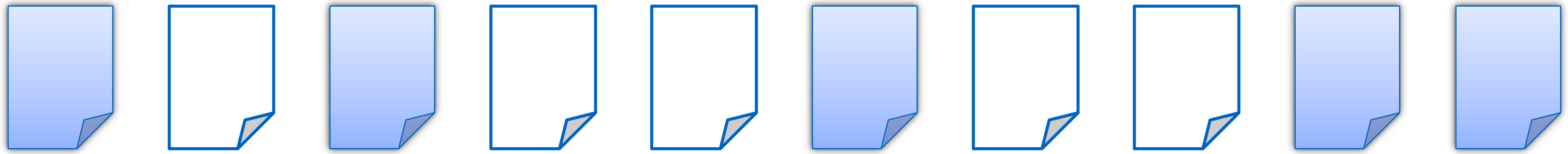
Query 2: # of relevant docs = 3



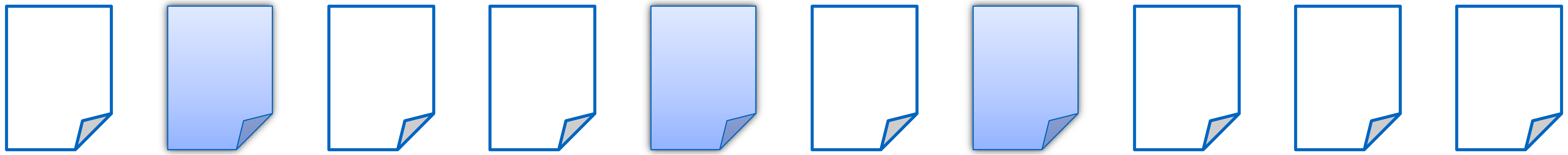
Mean Average Precision (MAP)

- **AP(Q1)** $= (1.00 + 0.67 + 0.50 + 0.44 + 0.50) / 5 = 0.62$
- **AP(Q2)** $= (0.50 + 0.40 + 0.43) / 3 = 0.44$
- **MAP** $= (0.62 + 0.44) / 2 = 0.53$

Query 1: # of relevant docs = 5



Query 2: # of relevant docs = 3



Credit

/* elice */

코스 매니저
이호정

콘텐츠 제작자
고혜선

강사
고혜선

감수자
이해솔

디자인
이해솔

Contact

TEL
070-4633-2015

WEB
https://elice.io

E-MAIL
contact@elice.io