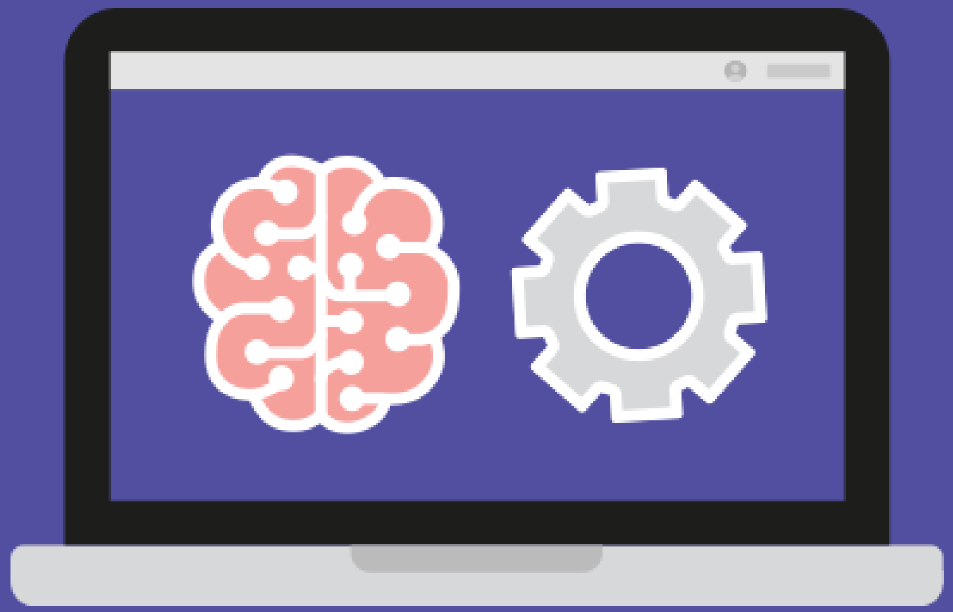
/* elice */

양재 AI School 인공지능 캠프

Lecture 6

분류 (Classification)



박상기 선생님

수업 목표

1 분류 (Classification)의 의미를 이해한다

Classification의 특성과 알고리즘, 그리고
Regression과의 차이점을 알아봅시다.

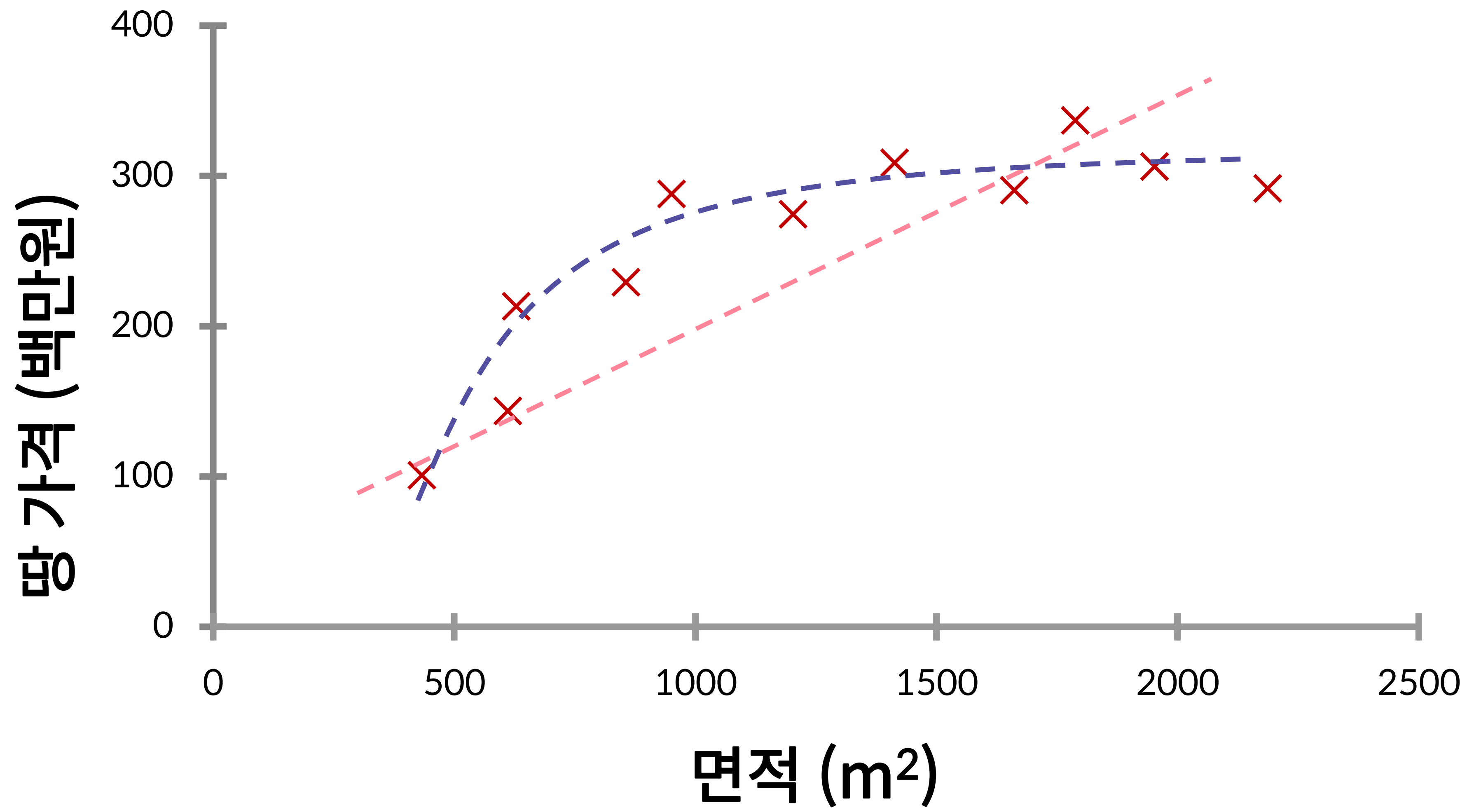
2 Classification의 다양한 방식을 알아본다

Logistic Regression, Softmax, Naïve Bayes, SVM
등의 분류 방식을 알아봅시다

Classification의 개념

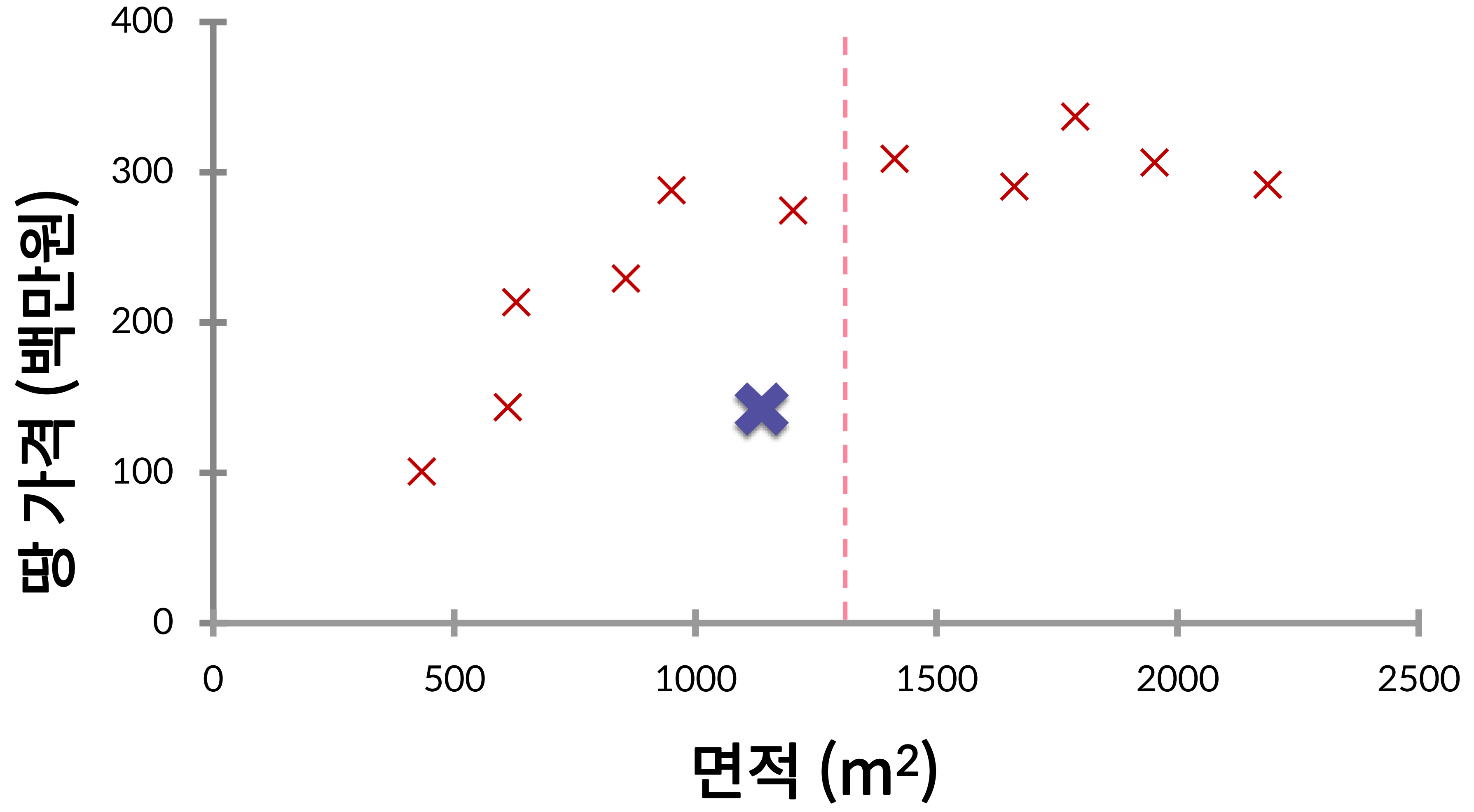
Regression vs Classification

Regression



- 분포된 데이터를 **방정식(Hypothesis)**을 통해 결과의 **값**을 예측

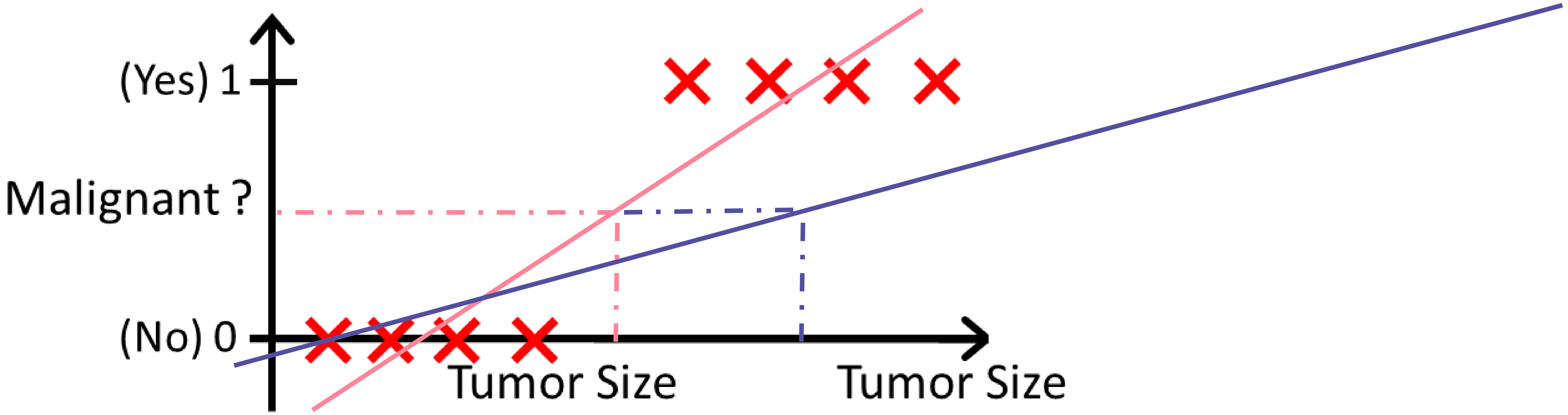
Supervised Learning



- **Classification** : 주어진 input이 어느 카테고리에 있는지 판별

Classification

- Regression이 결과값을 추정하는 방식이었다면,
  Classification은 카테고리에 분류하는 방식



Linear regression은 Classification에 사용할 수 없다!

Classification

- Linear regression은 사용 불가능
- Classification은 0, 또는 1값만 가지는데, linear regression은 그 범위 이상의 값을 가질 수 있다.
- 0 또는 1 사이의 값만 내보내는 Hypothesis 함수가 필요

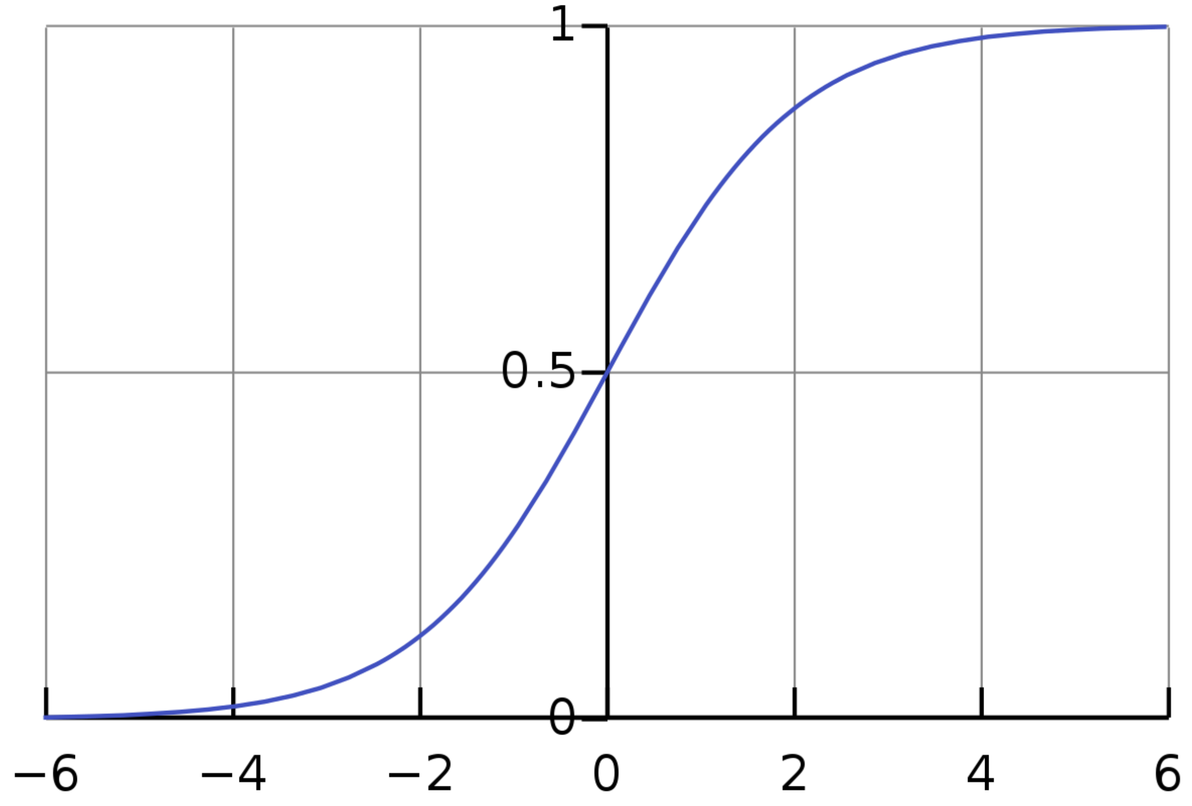
Logistic Regression

Logistic function을 사용한 분류

Logistic Regression

- 0~1 사이의 값만 내보내는 함수가 필요하다
- $0 \leq h(x) \leq 1$
- Sigmoid (logistic) function

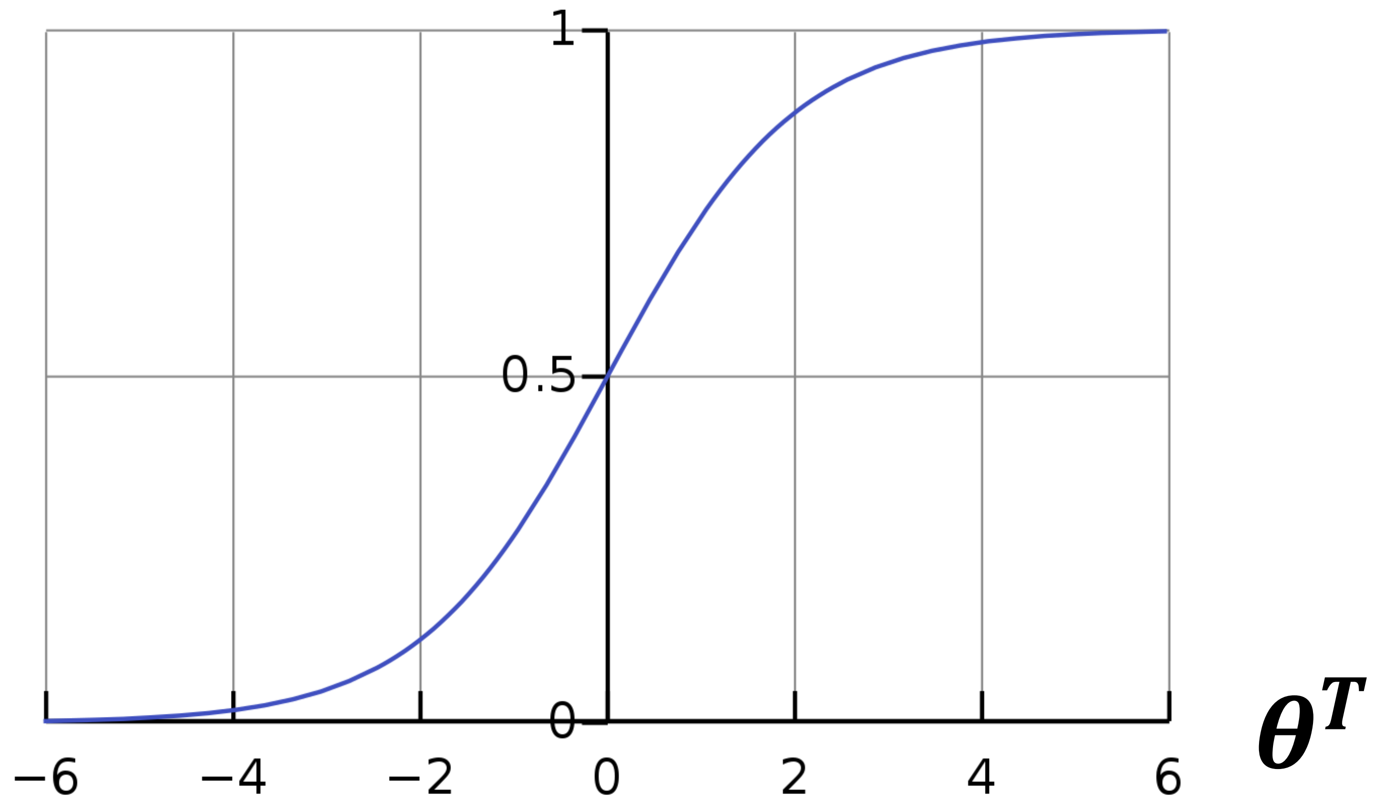
$$h(x) = g(\boldsymbol{\theta}^T \boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^T \boldsymbol{x}}}$$



$\boldsymbol{\theta}^T$

Logistic Regression

$$h(x) = g(\boldsymbol{\theta}^T \boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^T \boldsymbol{x}}}$$



$$\boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \dots \\ \theta_n \end{bmatrix}, \boldsymbol{x} = \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_n \end{bmatrix}, \boldsymbol{\theta}^T \boldsymbol{x} = \theta_0 x_0 + \theta_1 x_1 + \cdots \theta_n x_n$$

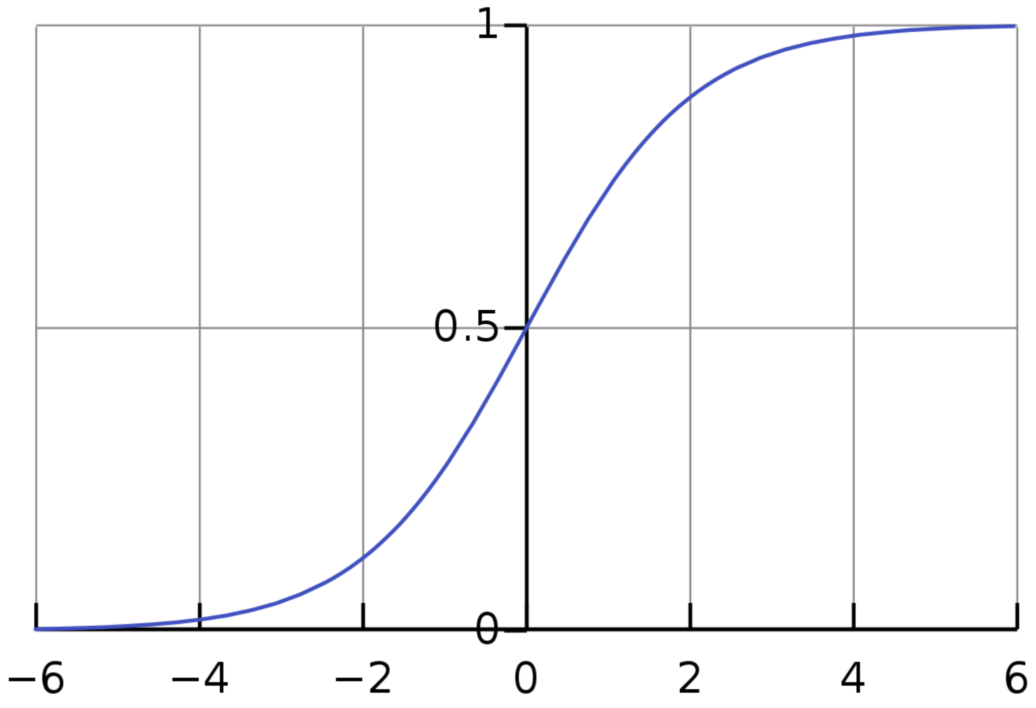
Logistic Regression

- $h(x) = P(y = 1|x; \theta)$
- 주어진 input이 x라는 값을 가질 때 1번 class에 들어갈 확률
- 1번 class일 확률과 0번 class 일 확률의 합은 항상 1이 되어야 한다
- $P(y = 1|x; \theta) + P(y = 0|x; \theta) = 1$

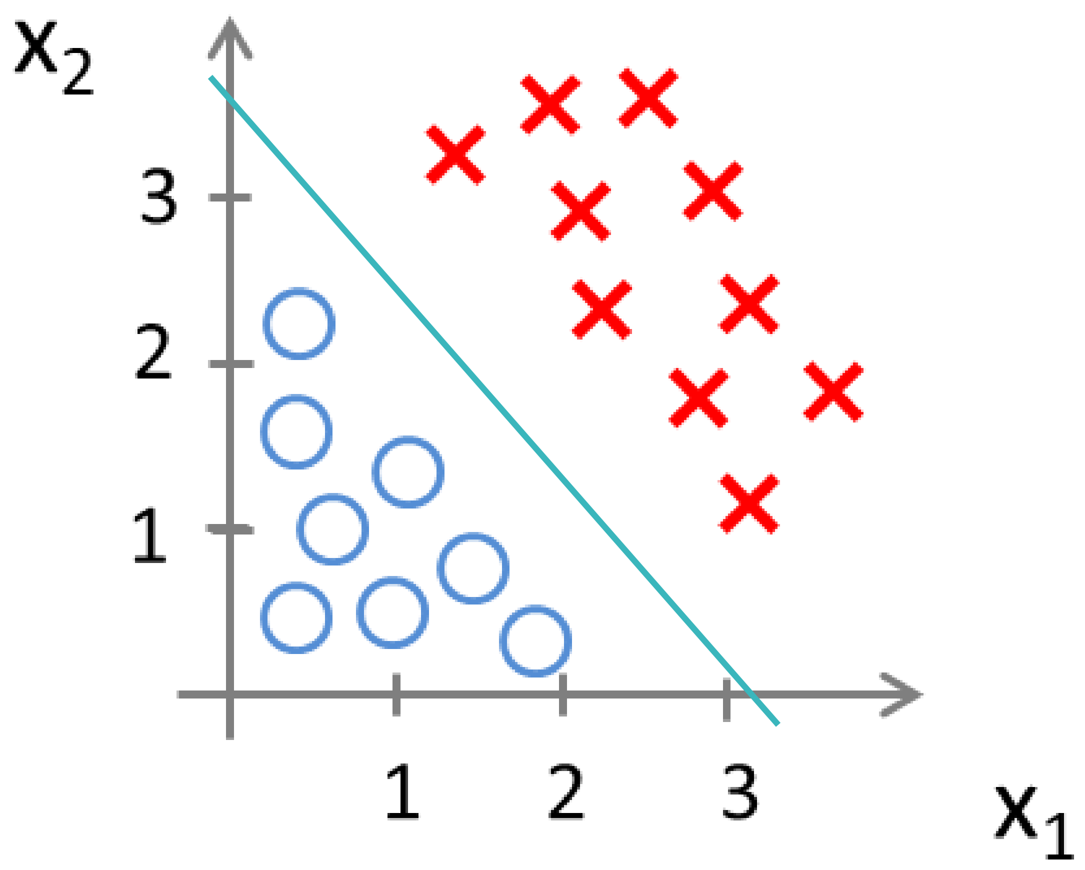
Decision Boundary

- 언제 어떤 클래스에 넣어주어야 할까?
  - Hypothesis function의 값 0.5를 기준으로 분류

- $y = \begin{cases} 1\ if\ \ \boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x} \geq 0 \\ 0\ if\ \ \boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x} < 0 \end{cases}$ $,where\ h(x) = \frac{1}{1+e^{-\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x}}}$



Decision Boundary

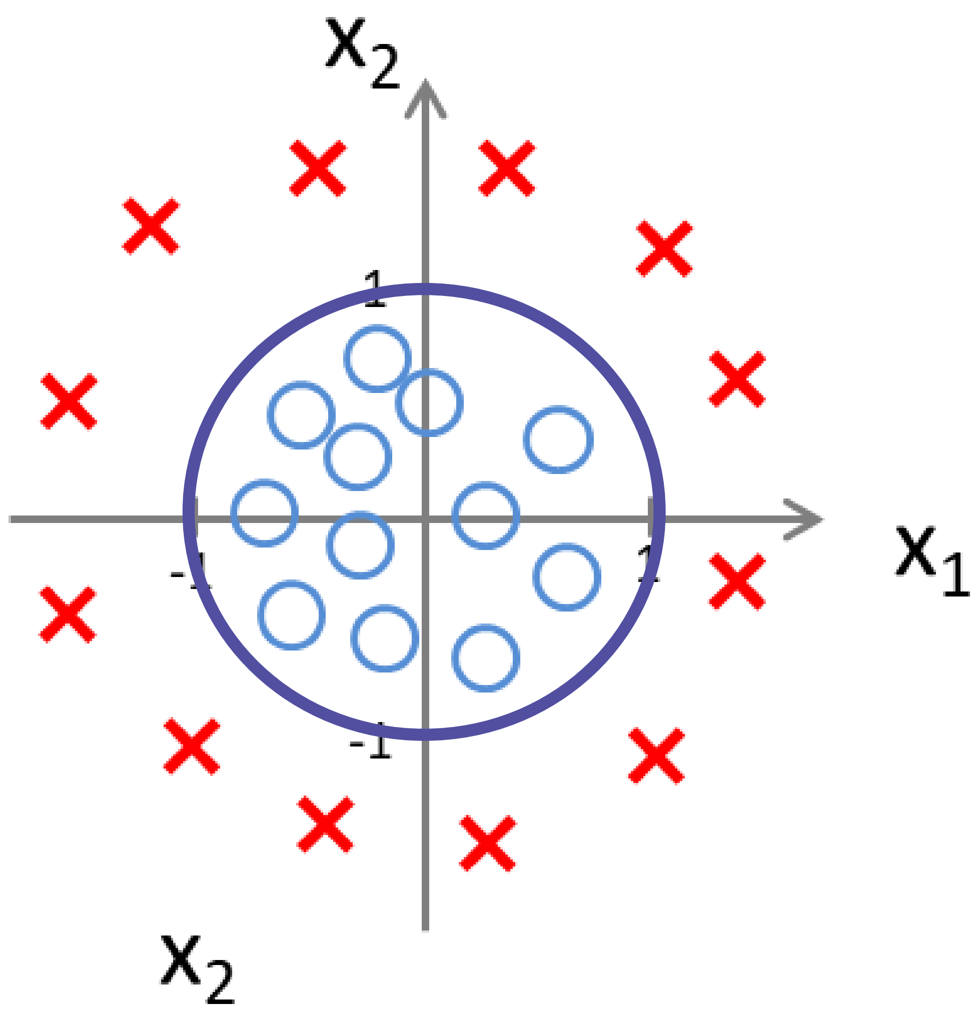


$$h(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$
$$= g(-3 + x_1 + x_2)$$
$$= \frac{1}{1 + e^{-(-3+x_1+x_2)}}$$

$\boldsymbol{\theta}^T\boldsymbol{x} = -3 + x_1 + x_2 \geq 0$ 이면 $y = 1$ 로 예측하자

$h(x) = 0.5$이면 $\boldsymbol{\theta}^T\boldsymbol{x} = 0$

Non-Linear Decision Boundary



$$h(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$$

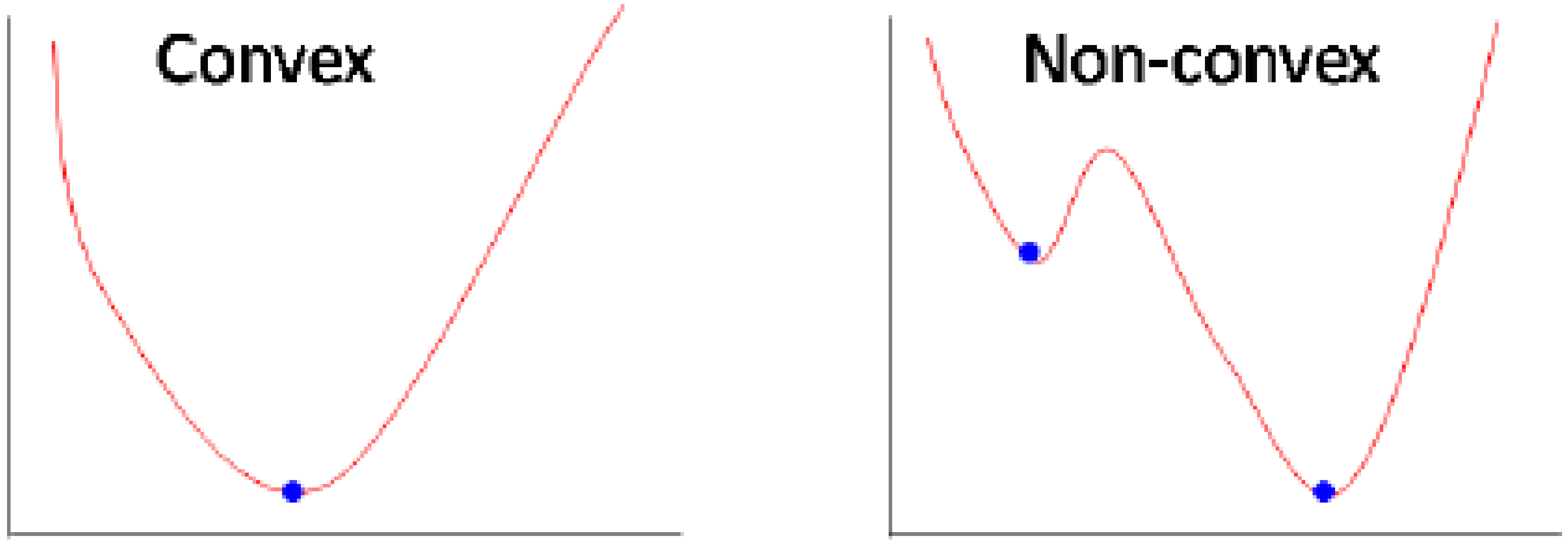
$$= g(-1 + x_1^2 + x_2^2)$$

$\boldsymbol{\theta}^T \boldsymbol{x} = -1 + x_1^2 + x_2^2 \geq 0$ 이면 $y = 1$ 로 예측하자

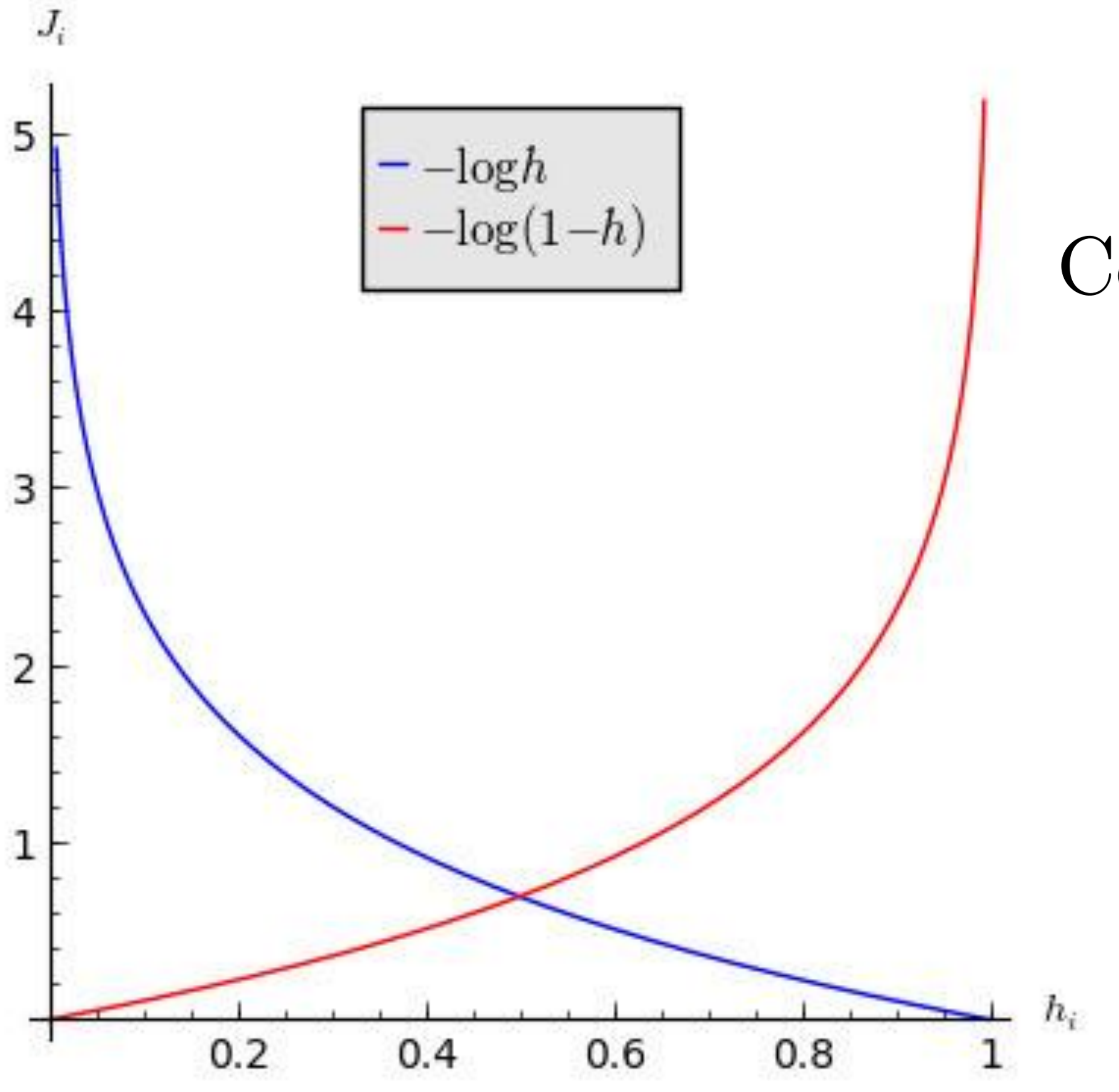
$h(x) = 0.5$이면 $\boldsymbol{\theta}^T \boldsymbol{x} = 0$

Cost Function

- θ parameter 값을 어떻게 정할까?
- Linear regression을 그대로 사용하면 non-convex
- Logistic 만의 cost function이 필요하다



Cost Function



$$\mathrm{Cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$

- $y = \begin{cases} 1 \; if \;\; \boldsymbol{\theta}^T \boldsymbol{x} \geq 0 \\ 0 \; if \;\; \boldsymbol{\theta}^T \boldsymbol{x} < 0 \end{cases} = \begin{cases} 1 \; if \;\; h(x) \geq 0.5 \\ 0 \; if \;\; h(x) < 0.5 \end{cases}$
- $y = 1$ 이고, $h(x) = 0$이면 잘못 분류하고 있다는 뜻 => Penalty

Cost Function

$$\mathrm{Cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$

$$\begin{aligned} J(\theta) &= \frac{1}{m} \sum_{i=1}^{m} \mathrm{Cost}(h_\theta(x^{(i)}), y^{(i)}) \\ &= -\frac{1}{m} [\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log (1 - h_\theta(x^{(i)}))] \end{aligned}$$

Gradient Descent

$$J(\theta) = -\frac{1}{m}[\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log (1 - h_\theta(x^{(i)}))]$$

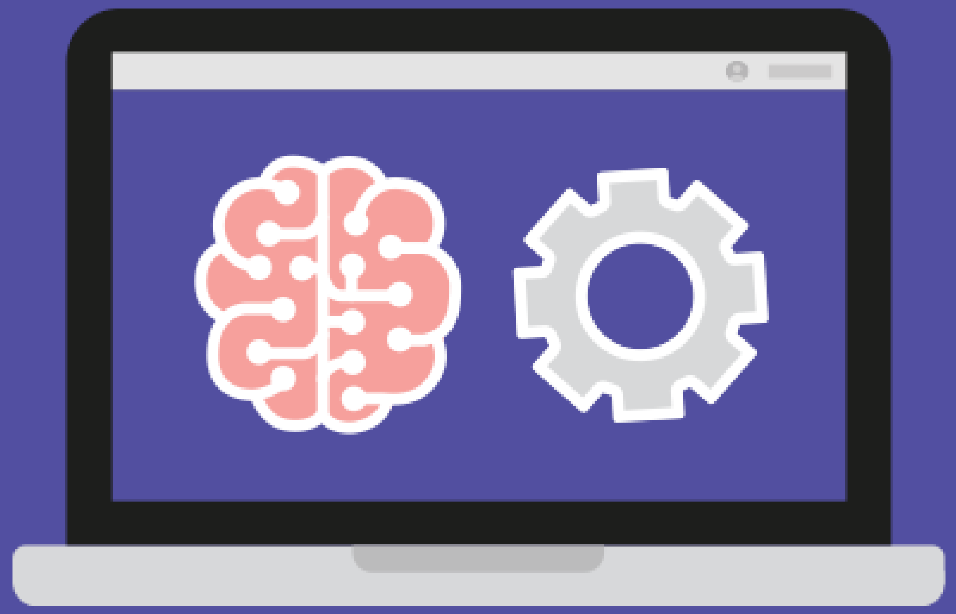
Repeat until convergence, do{

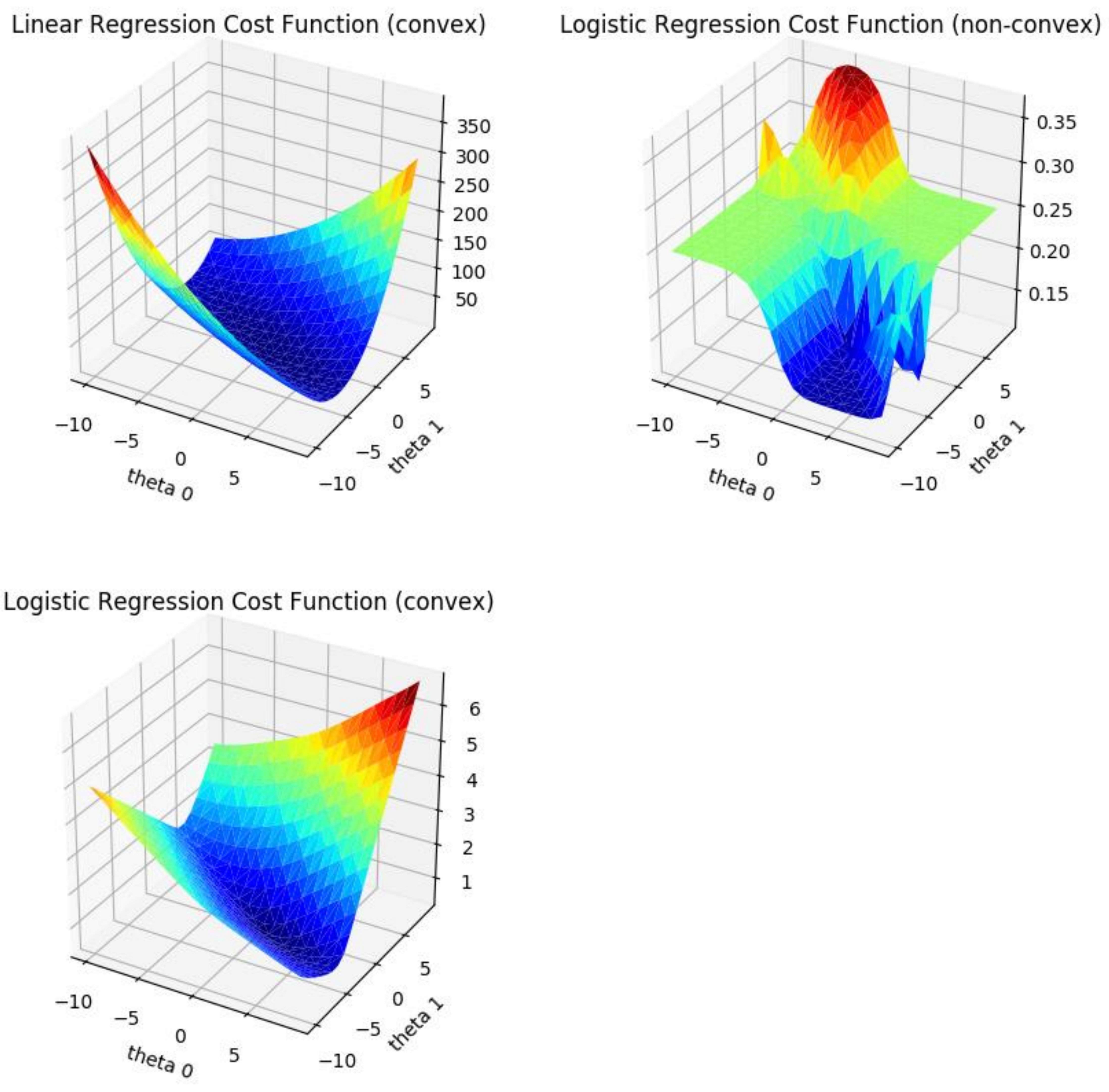
$$\theta_j \coloneqq \theta_j - \alpha\left(\sum_{i=1}^{m}(h(x^{(i)}) - y^{(i)})x_j^{(i)}\right), for\ all\ \theta_j$$

}

$$where, h(x^{(i)}) = \frac{1}{1+e^{-\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x}^{(\boldsymbol{i})}}}$$

[이론 6-1] Convex / non-Convex Cost Functions





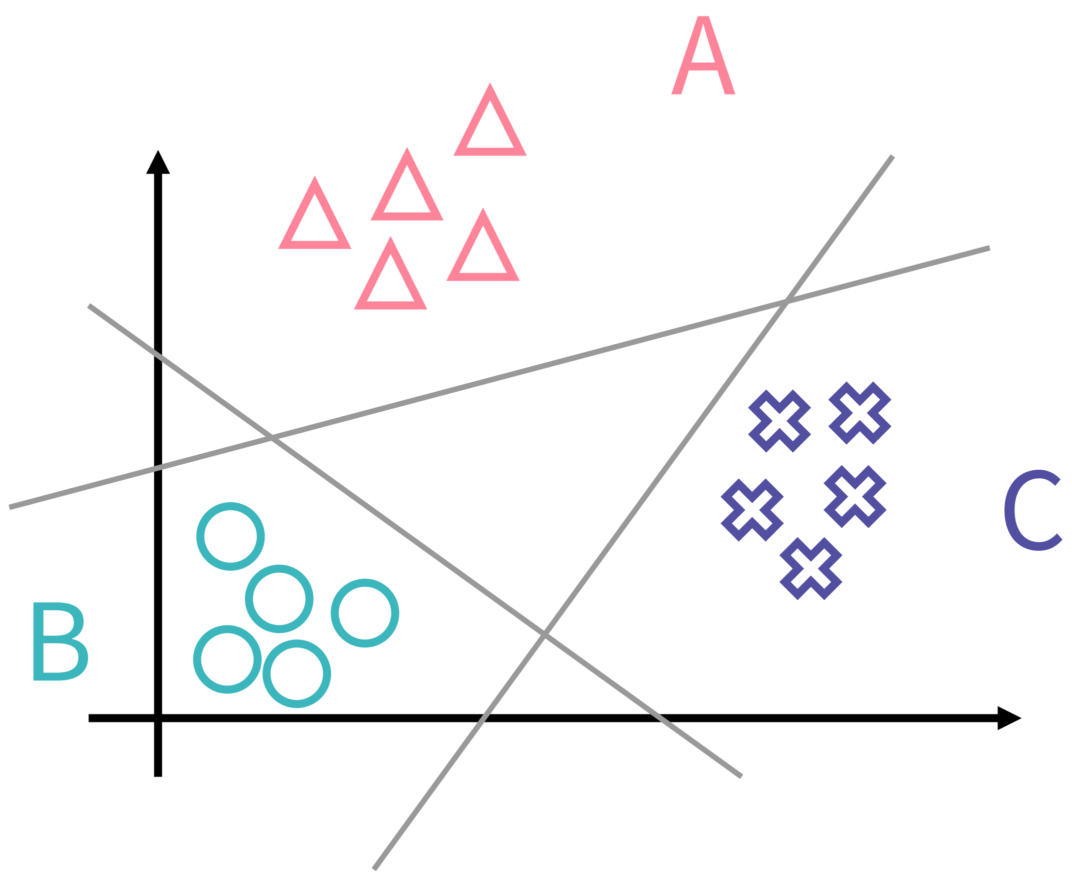
Linear Regression Cost Function (convex)
350
300
250
200
150
100
50
−10
−5
0
5
theta 0
5
0
−5
−10
theta 1
Logistic Regression Cost Function (non-convex)
0.35
0.30
0.25
0.20
0.15
−10
−5
0
5
theta 0
5
0
−5
−10
theta 1
Logistic Regression Cost Function (convex)
6
5
4
3
2
1
−10
−5
0
5
theta 0
5
0
−5
−10
theta 1

Softmax Function

Logistic Regression의 binary class 분류에서

multi-class 분류로 확장

3개 이상의 카테고리?



- A인가 아닌가? -> A 선별
- B인가 아닌가? -> B 선별
- C인가 아닌가? -> C 선별
- 그럼 Cost function은?

Softmax Function

- $x = \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_n \end{bmatrix}, \theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \dots \\ \theta_n \end{bmatrix}$
- $x = \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_n \end{bmatrix}, \theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \dots \\ \theta_n \end{bmatrix}$
- $x = \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_n \end{bmatrix}, \theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \dots \\ \theta_n \end{bmatrix}$

$$\begin{bmatrix} \theta_{00} & \theta_{01} & \theta_{02} \\ \theta_{10} & \theta_{11} & \theta_{12} \\ \theta_{20} & \theta_{21} & \theta_{22} \end{bmatrix} * \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \hat{y}_0 \\ \hat{y}_1 \\ \hat{y}_2 \end{bmatrix}$$

y_i 에 Sigmoid를 적용!

Softmax Function

$$\begin{bmatrix} \hat{y}_0 \\ \hat{y}_1 \\ \hat{y}_2 \end{bmatrix} * \left[\mathbf{sigmoid(y_i)} = \frac{\mathbf{e}^{\hat{y}_i}}{\sum_i e^{\hat{y}_i}}\right] = \begin{bmatrix} \boxed{0.7} \\ 0.1 \\ 0.2 \end{bmatrix}$$

A 그룹에 넣어주자!

- Sigmoid의 결과값은 input이 각 카테고리에 속할 확률
- **결과 행렬의 모든 확률 값을 다 더하면 반드시 1**
- 가장 높은 확률값을 가지는 카테고리에 넣어준다
- Cost function으로는 **Cross-entropy** 사용

Softmax Function

- One hot encoding: 확률이 가장 높은 것만 1, 나머지는 0
- $h(x) = \begin{bmatrix} 0.7 \\ 0.1 \\ 0.2 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$
- $\begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} \Rightarrow \begin{bmatrix} \theta_{00} & \theta_{01} & \theta_{02} \\ \theta_{10} & \theta_{11} & \theta_{12} \\ \theta_{20} & \theta_{21} & \theta_{22} \end{bmatrix} * \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \hat{y}_0 \\ \hat{y}_1 \\ \hat{y}_2 \end{bmatrix} \Rightarrow \left[\mathbf{sigmoid(y_i)} = \frac{e^{\hat{y}_i}}{\sum_i e^{\hat{y}_i}} \right] = \begin{bmatrix} 0.7 \\ 0.1 \\ 0.2 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

Softmax Function

- **Cross-entropy:** 두 변수들의 확률 분포가 얼마나 비슷한지 나타내는 척도
- $J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)}\, ln\left(h\left(x^{(i)}\right)\right)\right),\ where\, h\left(x^{(i)}\right) = \frac{1}{1+e^{-\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x}^{(\boldsymbol{i})}}}$
-

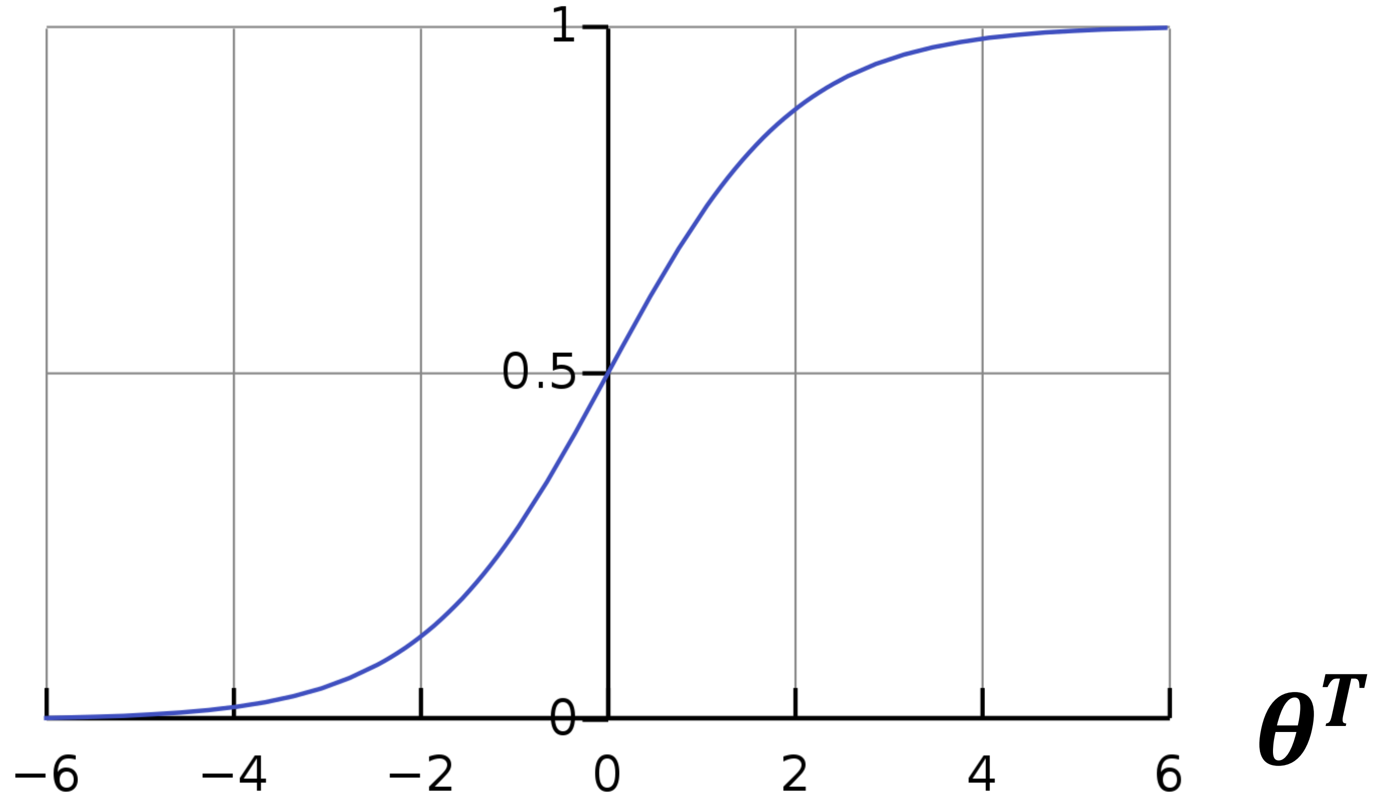
$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} 0.7 \\ 0.1 \\ 0.2 \end{bmatrix}$$

Support Vector Machine (SVM)

비확률적 이진 선형분류모델

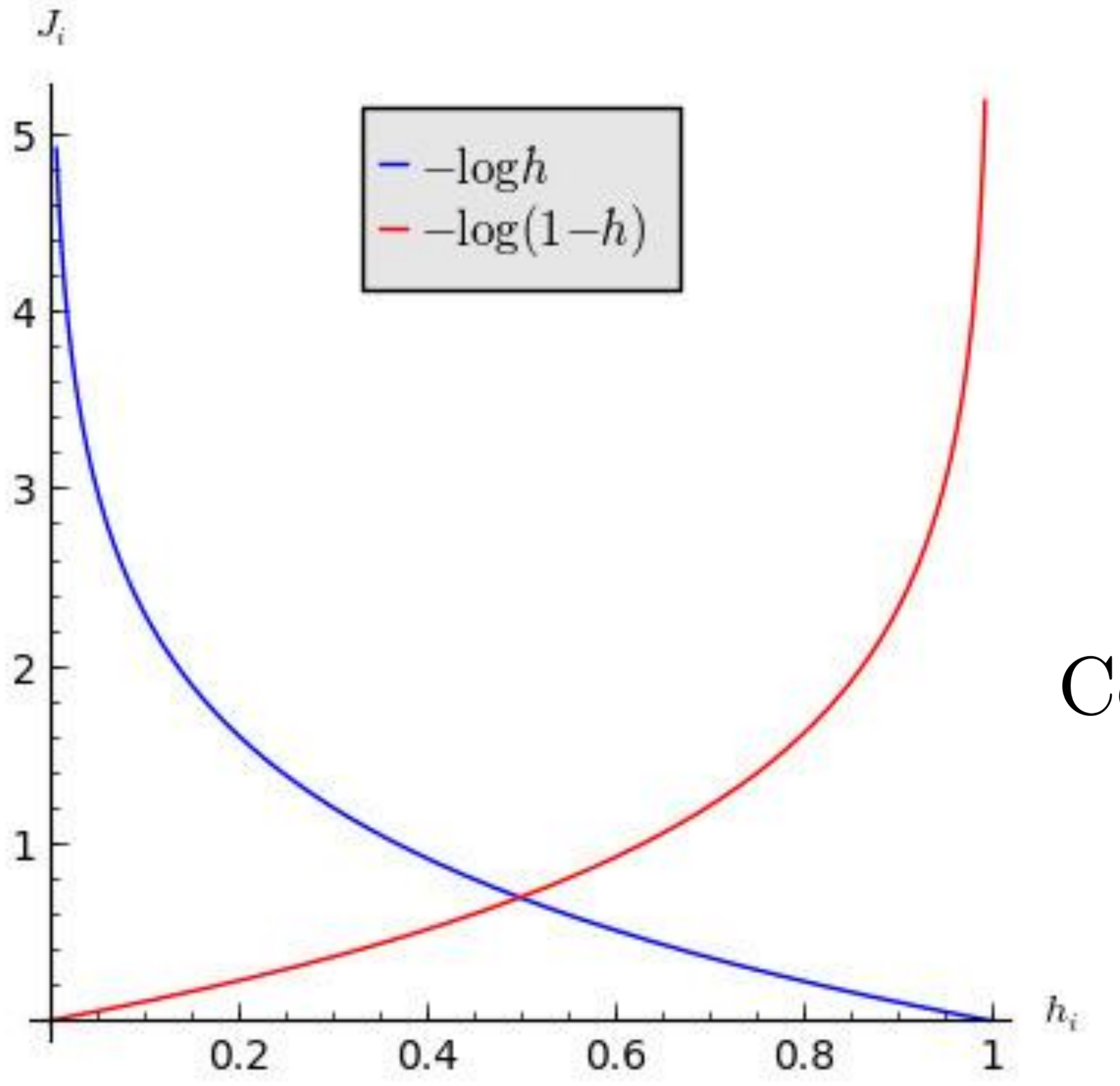
Support Vector Machine (SVM)

$$h(x) = g(\boldsymbol{\theta}^T \boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^T \boldsymbol{x}}}$$



- $y = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$에서,
  - $y = 1$이면, $h(x) \approx 1, \boldsymbol{\theta}^T \boldsymbol{x} \gg 0$ 일수록 좋고,
  - $y = 0$이면, $h(x) \approx 0, \boldsymbol{\theta}^T \boldsymbol{x} \ll 0$ 일수록 좋다

Support Vector Machine (SVM)



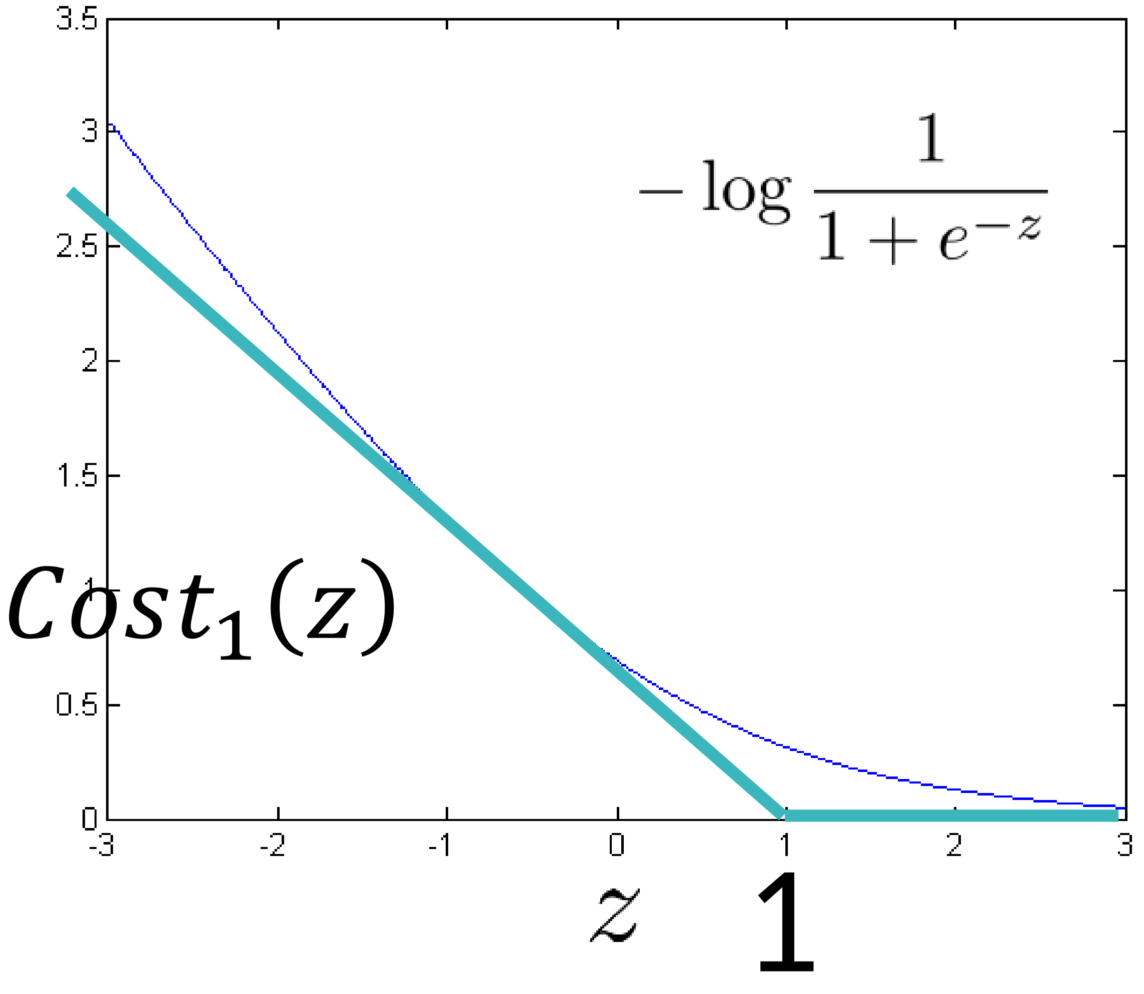
Logistic regression 에서는…

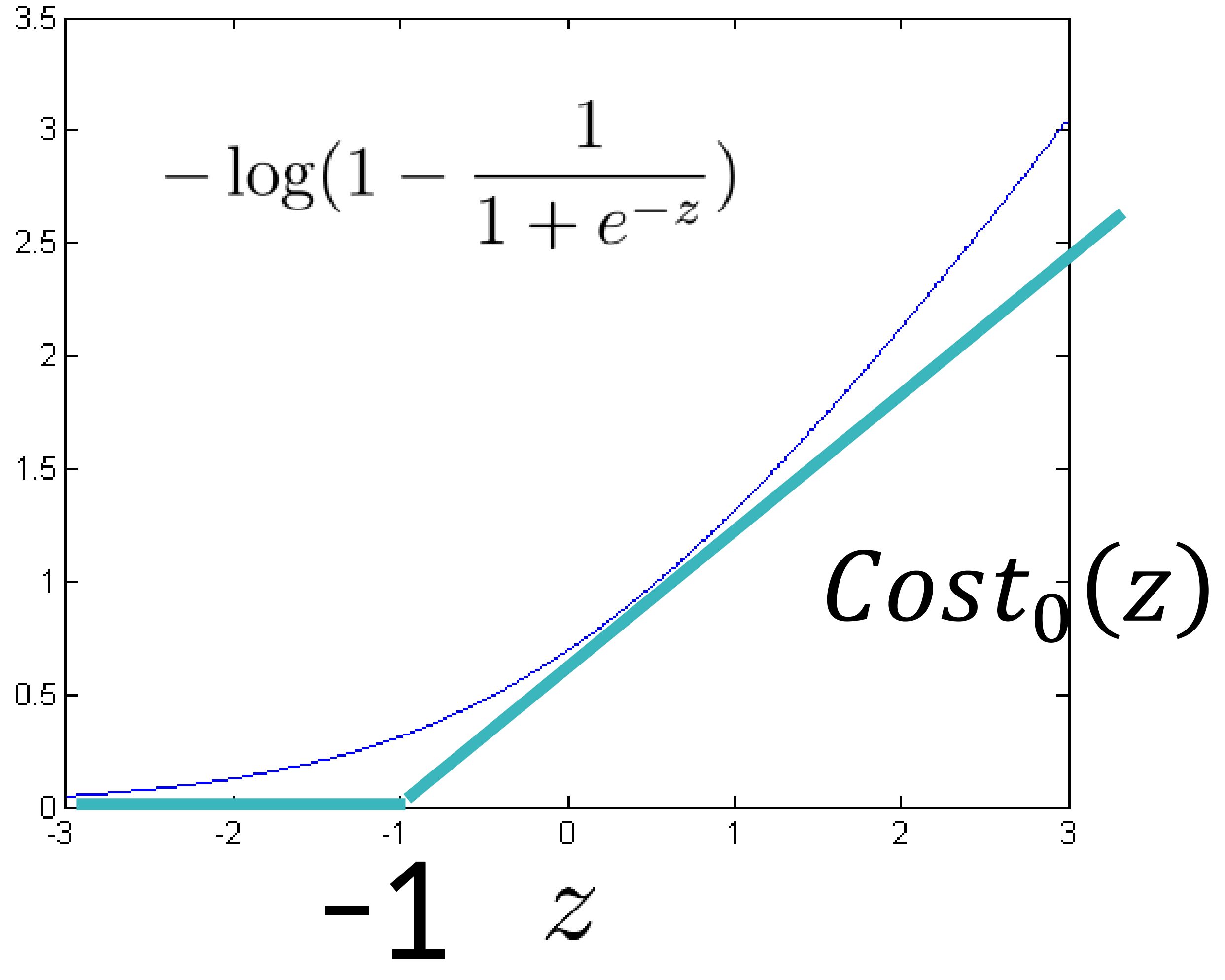
$$\mathrm{Cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$

- $y = \begin{cases} 1 \ if \ \ \boldsymbol{\theta}^T\boldsymbol{x} \geq 0 \\ 0 \ if \ \ \boldsymbol{\theta}^T\boldsymbol{x} < 0 \end{cases} = \begin{cases} 1 \ if \ \ h(x) \geq 0.5 \\ 0 \ if \ \ h(x) < 0.5 \end{cases}$
- $y = 1$ 이고, $h(x) = 0$이면 잘못 분류하고 있다는 뜻 => **Penalty**

Support Vector Machine (SVM)

- **Hinge Function:** 1, -1 을 기준으로 값이 0이 되는 새로운 Cost function을 정의
- $y = 1$이면, $h(x) \approx 1, \boldsymbol{\theta}^{T}\boldsymbol{x} \gg 0$ 일수록 좋고,
- $y = 0$이면, $h(x) \approx 0, \boldsymbol{\theta}^{T}\boldsymbol{x} \ll 0$ 일수록 좋다





Cost Function of SVM

- Logistic regression:

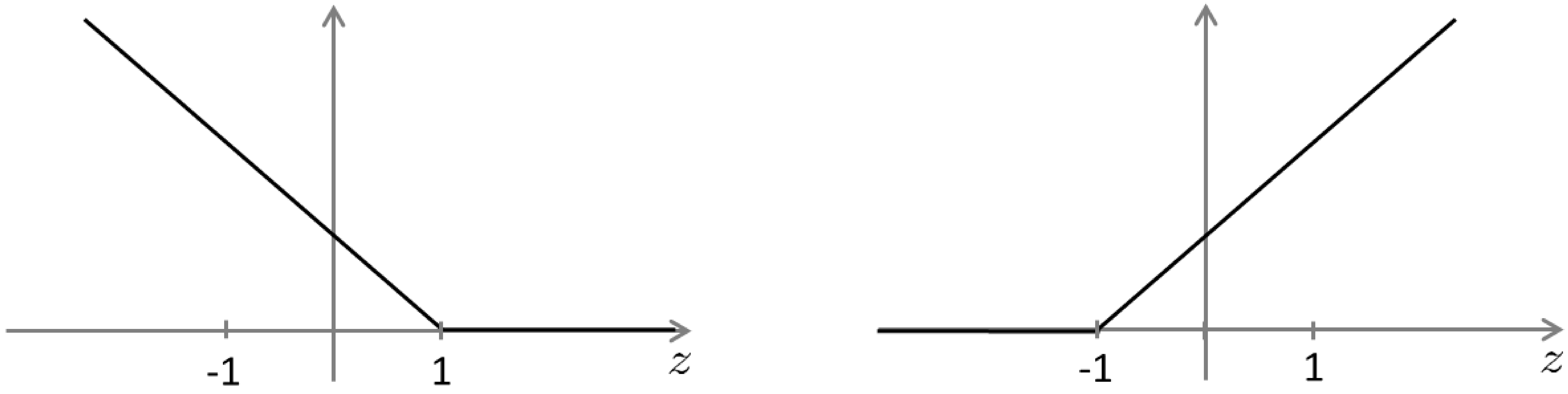
$$\min_{\theta} \frac{1}{m} \left[\sum_{i=1}^{m} y^{(i)} \left(-\log h_\theta(x^{(i)}) \right) + (1 - y^{(i)}) \left((-\log(1 - h_\theta(x^{(i)})) \right) \right]$$

- **SVM:** cost의 평균대신 합을 사용

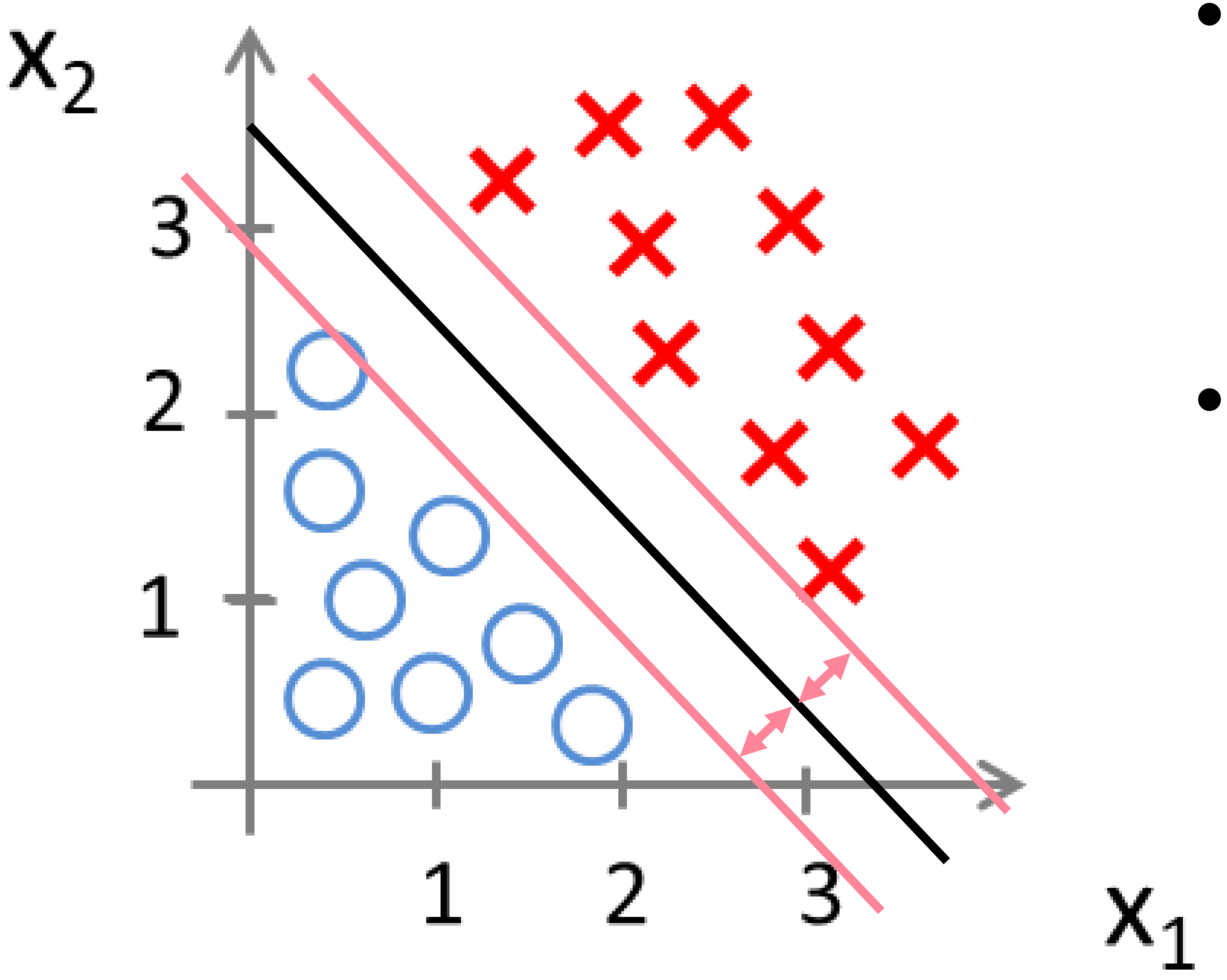
$$\min_{\theta} C \sum_{i=1}^{m} \left[y^{(i)} cost_1(\theta^T x^{(i)}) + (1 - y^{(i)}) cost_0(\theta^T x^{(i)}) \right]$$

Decision Boundary Margin

- $y = 1$이면, $\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x} \geq 1\ (not \geq 0)$
- $y = 0$이면, $\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x} \leq -1\ (not < 0)$

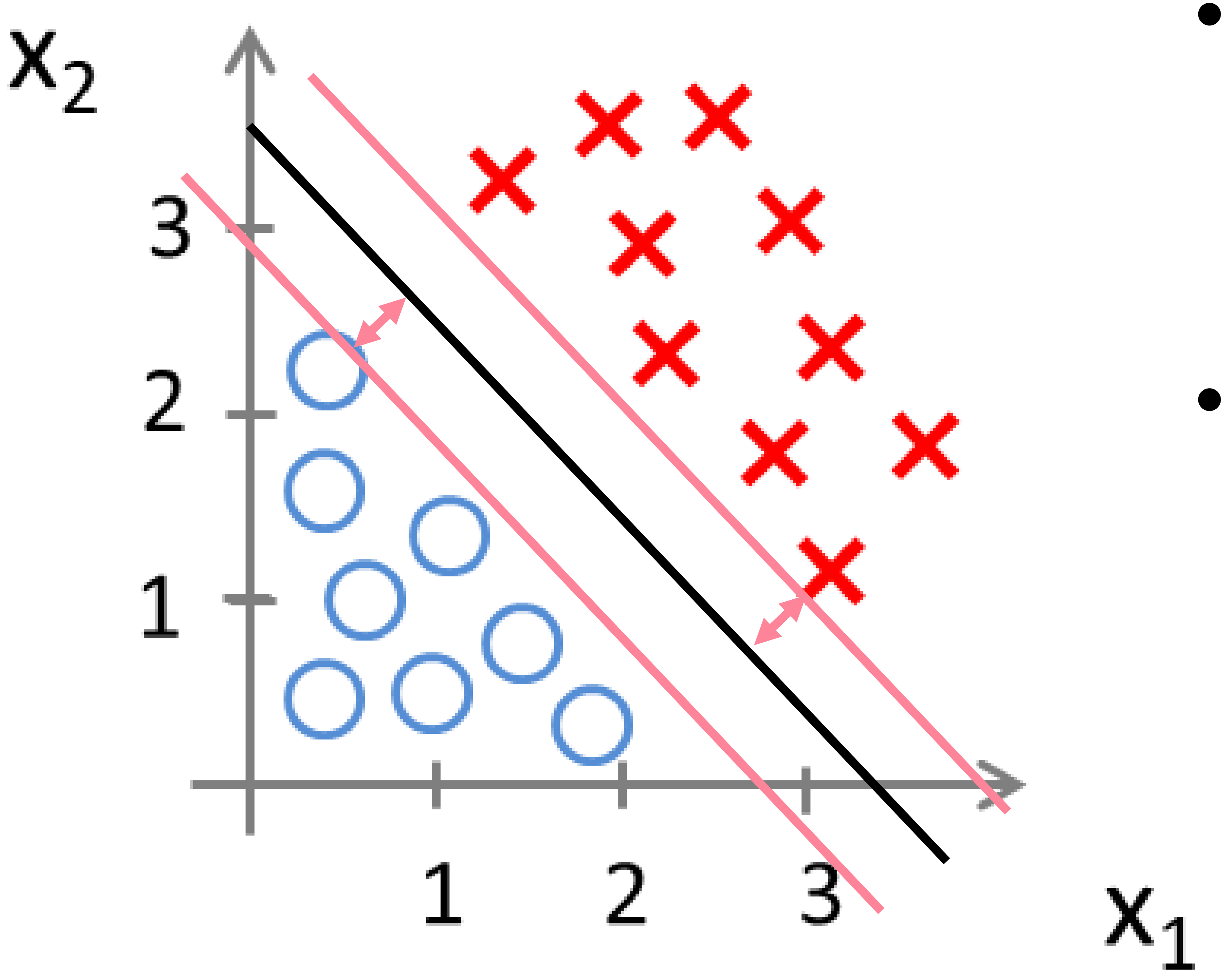


Decision Boundary Margin



- Class를 분류하는 기준선에 여유를 둘 수 있다
- Large Margin Classifier

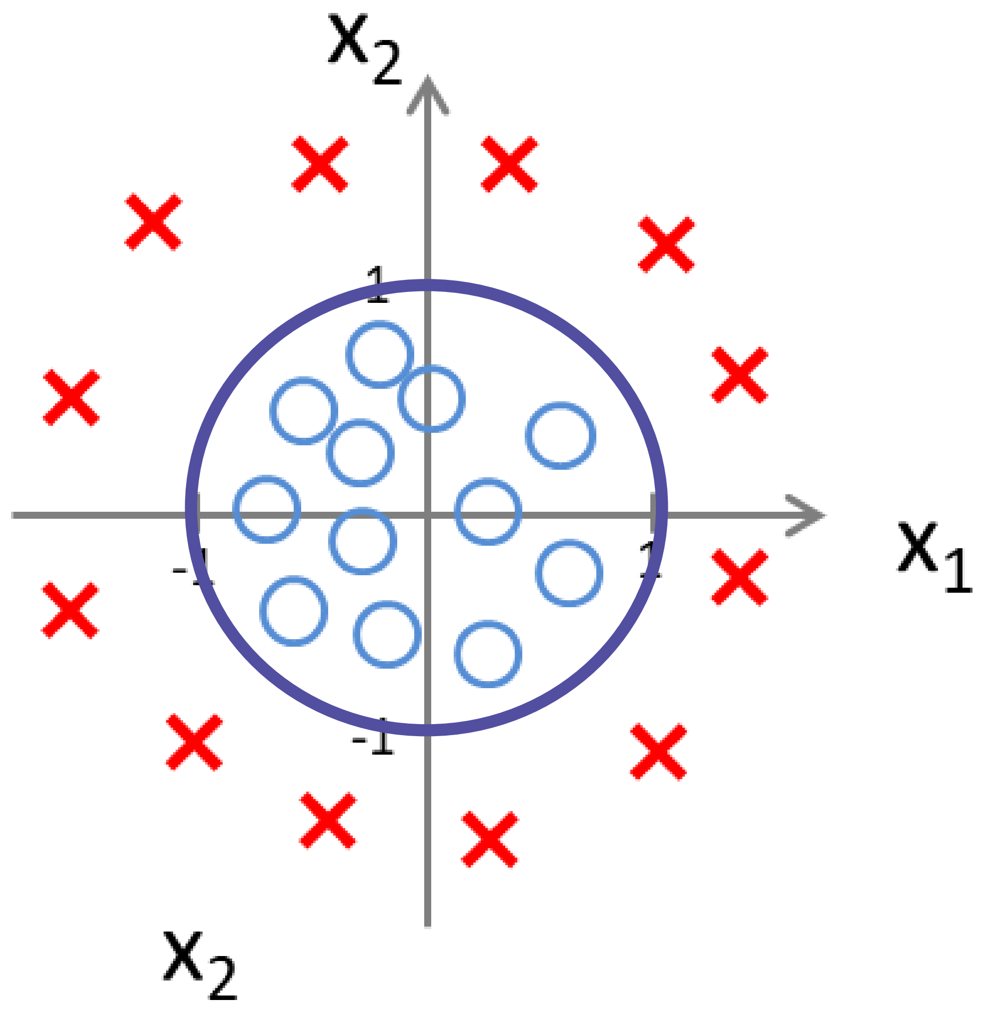
Decision Boundary Margin

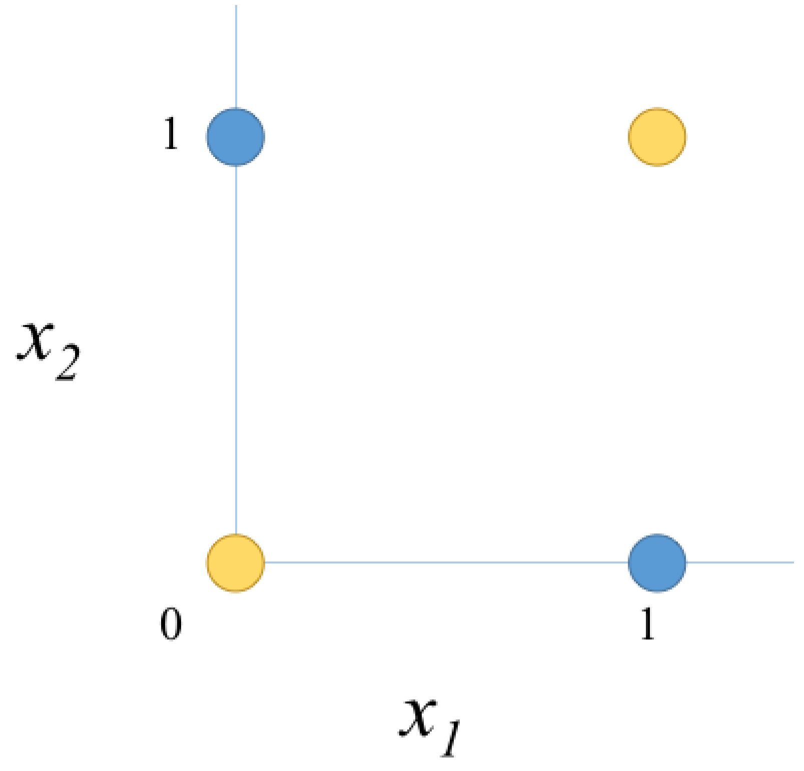


- Support Vector: 결정 경계에 가장 가까운 데이터
- slack 변수를 이용해 어느정도 오차를 허용하기도 함

Kernel SVM

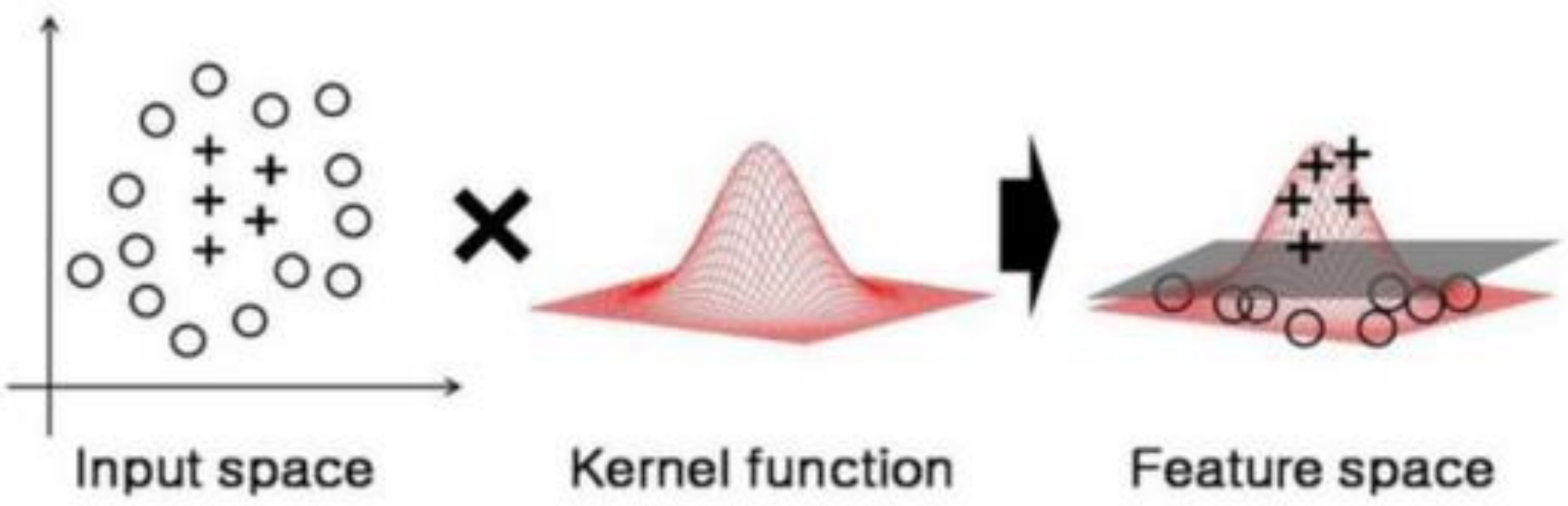
- Non-linear Decision Boundary
  - 데이터를 구분 짓는 ‘선’을 긋기 힘들다





Kernel SVM

- 데이터를 선형으로 구분할 수 있는 공간으로 재배치 한다
- 이때 데이터를 재배치 해주는 함수가 **Kernel**
- Polynomial, Sigmoid, Gaussian RBF …



Naïve Bayes Classifier

통계적 기법을 이용한 클래스 규정 알고리즘

Bayes Rule

- Bayes 정리: $P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B|A)P(A)}{P(B)}$
- $P(A|B)$: 사건 B가 발생했을 때, A도 같이 발생했을 확률

	비가 옴	비가 안옴	
맑은 날	2	8	10
흐린 날	5	5	10
	7	13	20

Bayes Rule

- P(비X|맑)=P(맑|비X)*P(비X)/P(맑) = 0.8

	비가 옴	비가 안옴	
맑은 날	2	P(맑|비X)=8/13	P(맑)=0.5
흐린 날	5	5	10
	7	P(비X)=0.65	20

Naïve Bayes

- 각 특징들이 서로 영향을 미치지 않는다고 가정 (독립적)
- 만약 오늘 **맑고**, 바람이 **세고**, 기압이 **낮고**, 온도가 **낮다**면 비가 올 확률은?

	날씨		바람		기압		온도		
	맑음	흐림	셈	약함	높음	낮음	높음	낮음	계
비 옴	2	6	6	2	8	0	5	3	8
비 안옴	8	4	2	10	2	10	6	6	12
계	10	10	8	12	10	10	11	9	20

Naïve Bayes

$$\frac{P(비|맑음,셈,기압낮,온도낮)}{P(비|맑음,셈,기압낮,온도낮)+P(\sim 비|맑음,셈,기압낮,온도낮)}$$

$$P(비|맑음,셈,기압낮,온도낮) = $$

$$P(맑음|비)P(셈|비)P(기압낮|비)P(온도낮|비)P(비)$$

$$P(\sim 비|맑음,셈,기압낮,온도낮) = $$

$$P(맑음|\sim 비)P(셈|\sim 비)P(기압낮|\sim 비)P(온도낮|\sim 비)P(\sim 비)$$

Naïve Bayes

$$\frac{P(\text{비}|\text{맑음}, \text{셈}, \text{기압낮}, \text{온도낮})}{P(\text{비}|\text{맑음}, \text{셈}, \text{기압낮}, \text{온도낮}) + P(\text{비안옴}|\text{맑음}, \text{셈}, \text{기압낮}, \text{온도낮})} = \frac{0.0018}{0.0018 + 0.0278}$$

$$= 0.0608 = 6.08\%$$

	날씨		바람		기압		온도		
	맑음	흐림	셈	약함	높음	낮음	높음	낮음	계
비 옴	2/8	6	6/8	2	6	2/8	5	3/8	8/20
비 안옴	8/12	4	2/12	10	2	10/12	6	6/12	12/20
계	10	10	8	12	10	10	11	9	20

Naïve Bayes

- 스팸 메일의 분류
  - 스팸인 메일과 정상적인 메일의 단어를 체크
  - 새로운 메일의 단어들에 대한 확률로 스팸메일을 구분

IDX	type	text
1	ham	Hope you are having a good week. Just checking in
2	ham	K..give back my thanks.
3	ham	Am also doing in cbe only. But have to pay.
4	spam	complimentary 4 STAR Ibiza Holiday or 10,000 cash needs your URGENT
5	spam	okmail: Dear Dave this is your final notice to collect your 4* Tenerife Holiday or ……
~		
5559	ham	Shall call now dear having food

idx	check	good	thanks	pay	~
1	1	1	0	0	~
2	0	0	1	1	
3	0	0	0	0	
~					

$P(\text{스팸}|\text{단어}1, \text{단어}2, \text{단어}3 \ldots) > P(\text{정상}|\text{단어}1, \text{단어}2, \text{단어}3 \ldots)$ 이면 스팸

Naïve Bayes

- +) Multi-class 분류에서 쉽고 빠르게 예측 가능
- +) 각 특징들이 독립이라면 다른 분류 방식에 비해 결과가 좋고, 학습 데이터도 적게 필요

- -) 각 특징들이 독립이 아니라면 결과의 신뢰성 하락
- -) 학습 데이터에 없는 범주의 데이터가 들어오면 정상적인 예측 불가능

/* elice */

문의 및 연락처

academy.elice.io
contact@elice.io
facebook.com/elice.io
medium.com/elice