

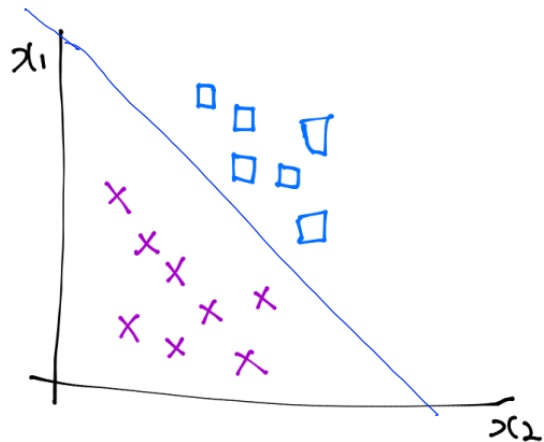
# 2주차 ML/DL 스터디 발표

GDSC Hanyang

ML/DL core  
김남호

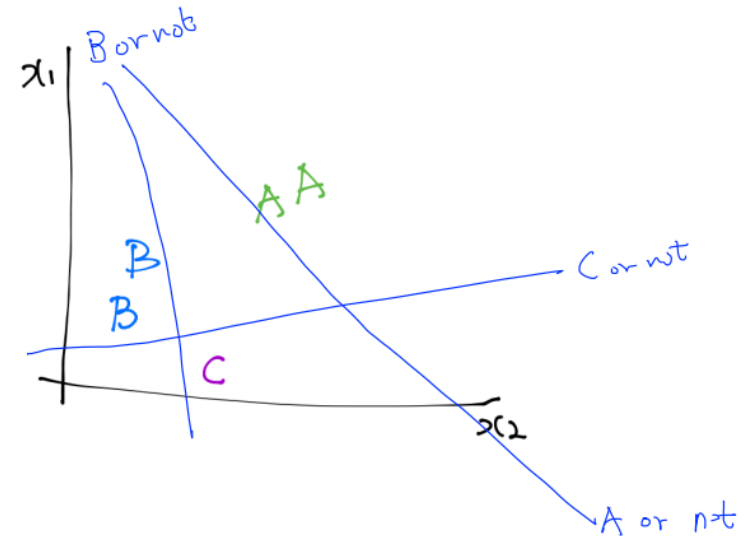
# Logistic regression

- 선형 회귀 직선에 data set을 넣음
- 그 결과를 또 g함수(classification을 위한 함수)에 넣음
- Training data set을 구분할 수 있게 학습->



# Multinomial classification

- 변수도 여러 개. 분류도 이진분류가 아님.(Ex: 학점(A,B,C..))
- 어떻게 분류할래?
- 각 분류값(A,B,C)마다 T,F식으로 구분.



# Multinomial classification

- 행렬로 작성. (다변수이므로)
- 또 총 세개의 구별할 수 있는 가설을 만들어야 함.

$$\begin{aligned} [w_1 \ w_2 \ w_3] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} &= [w_1 x_1 + w_2 x_2 + w_3 x_3] \\ [w_1 \ w_2 \ w_3] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} &= [w_1 x_1 + w_2 x_2 + w_3 x_3] \\ [w_1 \ w_2 \ w_3] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} &= [w_1 x_1 + w_2 x_2 + w_3 x_3] \end{aligned}$$

$$\begin{bmatrix} w_{A1} & w_{A2} & w_{A3} \\ w_{B1} & w_{B2} & w_{B3} \\ w_{C1} & w_{C2} & w_{C3} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} w_{A1}x_1 + w_{A2}x_2 + w_{A3}x_3 \\ w_{B1}x_1 + w_{B2}x_2 + w_{B3}x_3 \\ w_{C1}x_1 + w_{C2}x_2 + w_{C3}x_3 \end{bmatrix} = \begin{bmatrix} \bar{y}_A \\ \bar{y}_B \\ \bar{y}_C \end{bmatrix}$$

$H_A(x)$   
 $H_B(x)$   
 $H_C(x)$

# Multinomial classification

- Where is sigmoid?

- 단순 scalar 실수값으로 나오면 예측이 안된다.  
(A,B,C grade를 매기기 어렵다)

$$\begin{bmatrix} w_{A1}x_1 + w_{A2}x_2 + w_{A3}x_3 \\ w_{B1}x_1 + w_{B2}x_2 + w_{B3}x_3 \\ w_{C1}x_1 + w_{C2}x_2 + w_{C3}x_3 \end{bmatrix} = \begin{bmatrix} \bar{y}_A \\ \bar{y}_B \\ \bar{y}_C \end{bmatrix} \begin{bmatrix} 2.0 \\ 1.0 \\ 0.1 \end{bmatrix}$$

- Sigmoid와 같은 함수를 통해 **0~1 사이의 값**을 만들어야 한다.

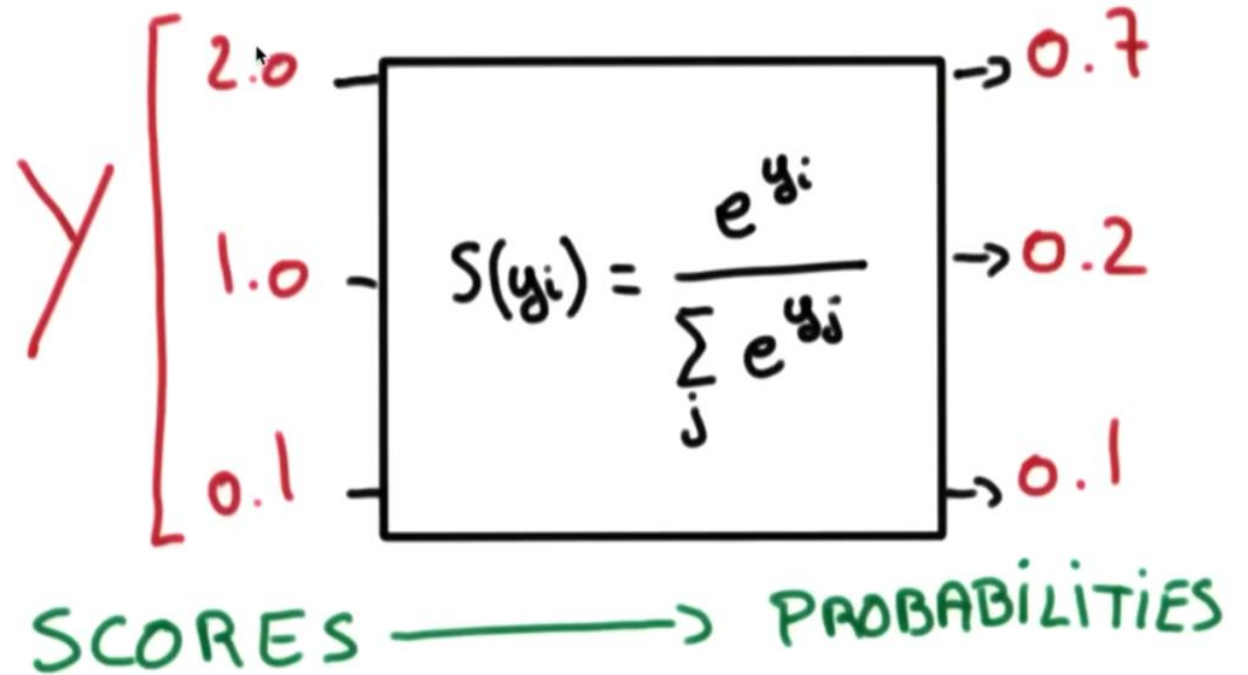
**벡터!**

# SoftMax 함수

$$\text{Softmax}(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

- 확률 분포

SOFTMAX



# One-hot encoding

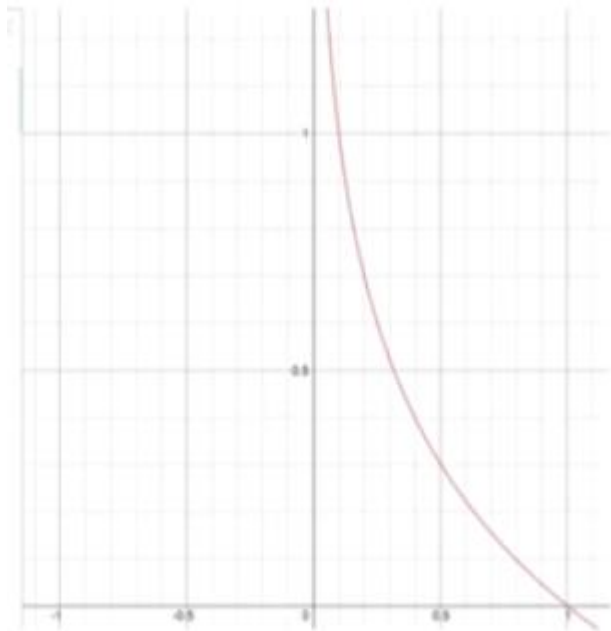
- ~~Softmax함수의 결과를~~

범주형 데이터를 1또는 0(이진변수)로 변환함.

즉, 예측을 하기 위해 실제 training 데이터 셋을 수치형 데이터로 변환하는 것.

- Ex) A등급은 1 나머지 등급(B,C)는 0
- 다중클래스 분류의 Hypothesis 완성

# Cross-entropy



CROSS-ENTROPY

$S(Y) = \bar{Y}$

$L = \bar{Y}$

$D(S, L) = - \sum_i L_i \log(S_i)$

0.7
0.2
0.1

1.0
0.0
0.0

The diagram illustrates the calculation of cross-entropy loss. It shows two probability distributions:  $S$  (predicted) and  $L$  (target). The predicted distribution  $S$  has values [0.7, 0.2, 0.1] and the target distribution  $L$  has values [1.0, 0.0, 0.0]. The cross-entropy loss is calculated as  $D(S, L) = - \sum_i L_i \log(S_i)$ . The formula is shown with arrows indicating that  $S$  and  $L$  are the predicted and target distributions, respectively. The loss is calculated as  $- (1.0 \cdot \log(0.7) + 0.0 \cdot \log(0.2) + 0.0 \cdot \log(0.1))$ .



# Cross-entropy

- 예측하려는  $\hat{Y}$ 와, 그 대상이 되는  $L$ 을 element-wise 곱(행렬산술연산)을 한다.
- 결과: 틀리면 오차가 매우 크게, 맞으면 0으로 나온다.

# Logistic cost VS cross entropy

$$C(H(x), y) = y \log(H(x)) - (1 - y) \log(1 - H(x))$$

$$D(S, L) = - \sum_i L_i \log(s_i)$$

// ?

둘 다 결국 무한대 or 0이렇게 이진값으로 최대한 오차를 양분해서 크게 나타낼 수 있게 표시함. (y와 (1-y)를 이용함)

## Logistic cost VS cross entropy

$$C(H(x), y) = y \log(H(x)) - (1 - y) \log(1 - H(x))$$
$$\mathcal{D}(S, L) = - \sum_i L_i \log(S_i)$$

# Cost function

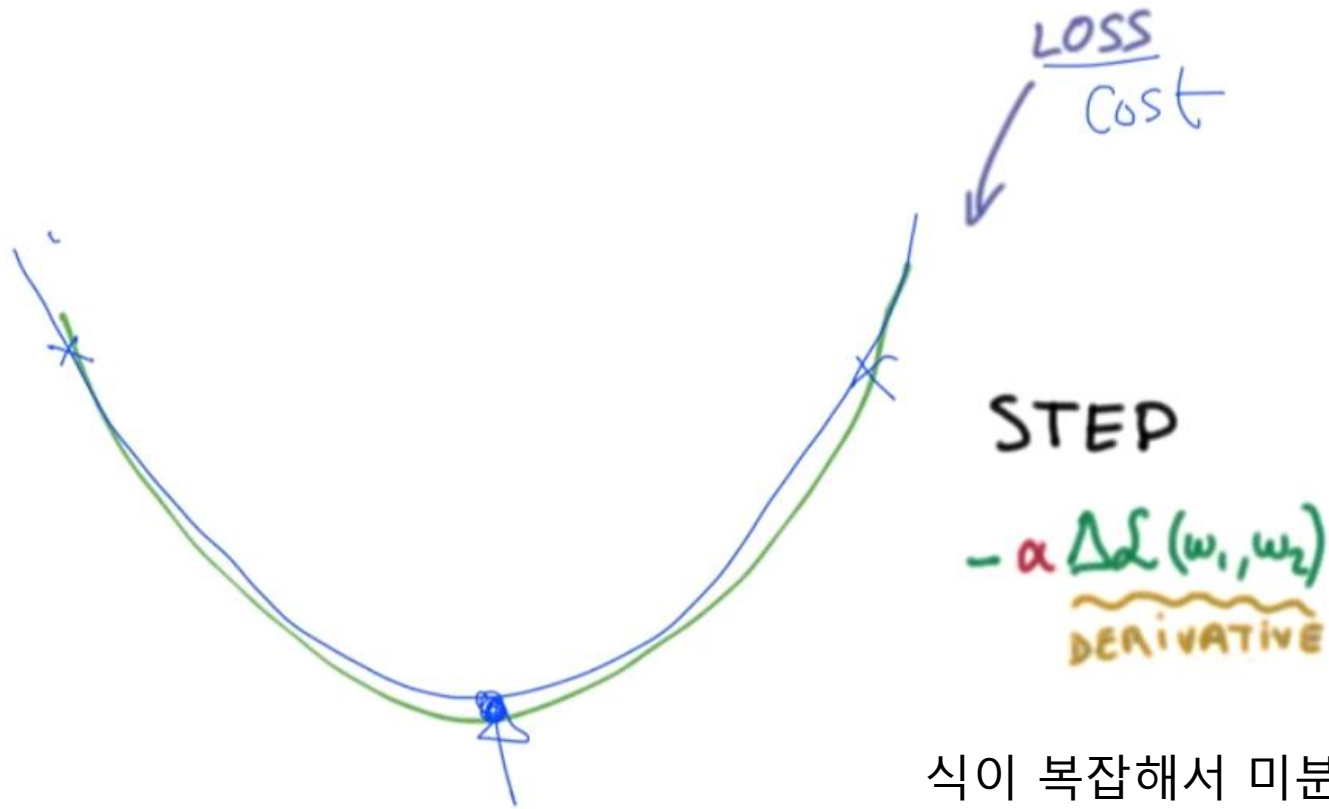
LOSS

$$\mathcal{L} = \frac{1}{N} \sum_i \mathcal{D}(s(wx_i + b), L_i)$$

TRAINING SET

The diagram shows the cost function formula  $\mathcal{L} = \frac{1}{N} \sum_i \mathcal{D}(s(wx_i + b), L_i)$  with several handwritten annotations in blue ink. An arrow labeled 'LOSS' points to the symbol  $\mathcal{L}$ . Another arrow points from the 'TRAINING SET' label to the summation index  $i$ . Two additional arrows originate from the 'TRAINING SET' label: one points to the input  $x_i$  within the function  $s(wx_i + b)$ , and the other points to the target  $L_i$  within the distance function  $\mathcal{D}$ .

# Gradient descent

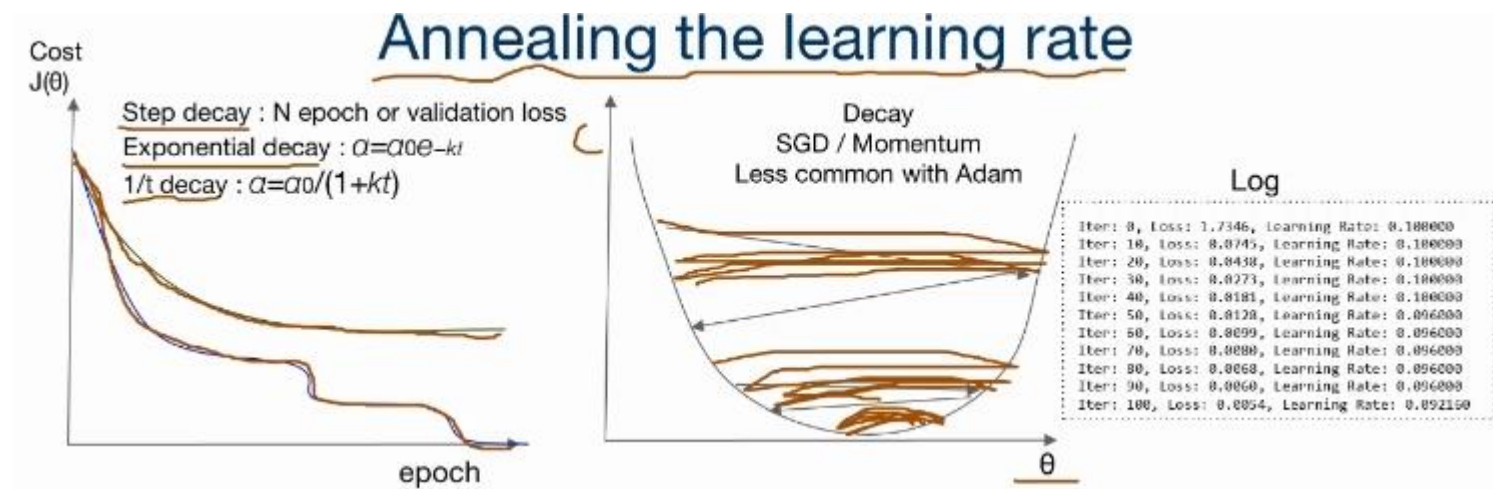


식이 복잡해서 미분하는 것은  
Lec에서 다루지 않음.  
-> 결국  $w$ 에 대해 미분하지 않을까?

(직접 찾아볼것)

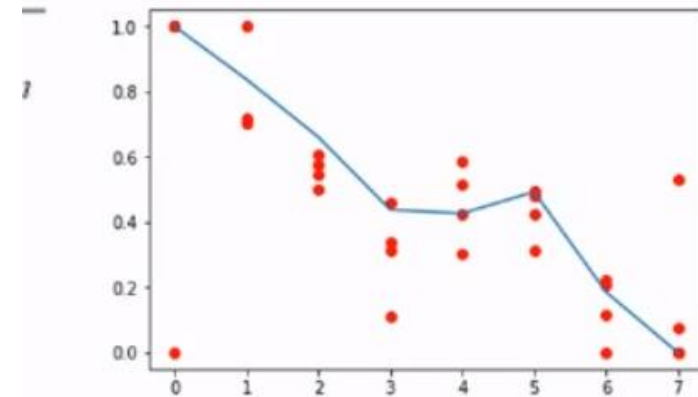
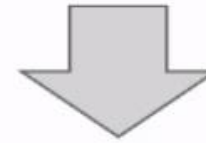
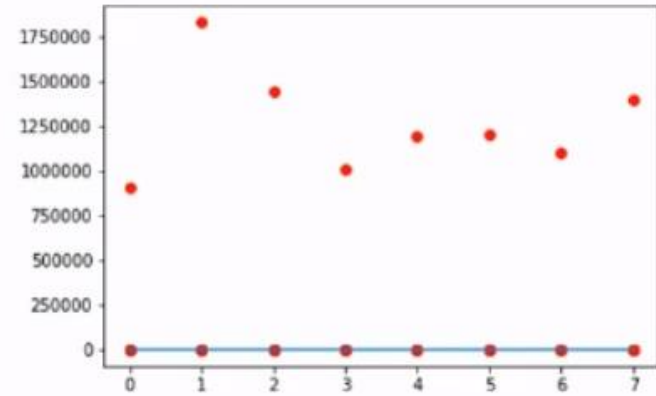
# Learning rate

- 높은 learning rate = 오버슈팅
- 아담 옵티마이저 = 0.0003
- Decay 기법 (Ex: exponential decay)



# 데이터 전처리

- 표준화
  - 통계에서 배운 그대로
- 정규화
  - 0~1사이의 값으로 data 표현



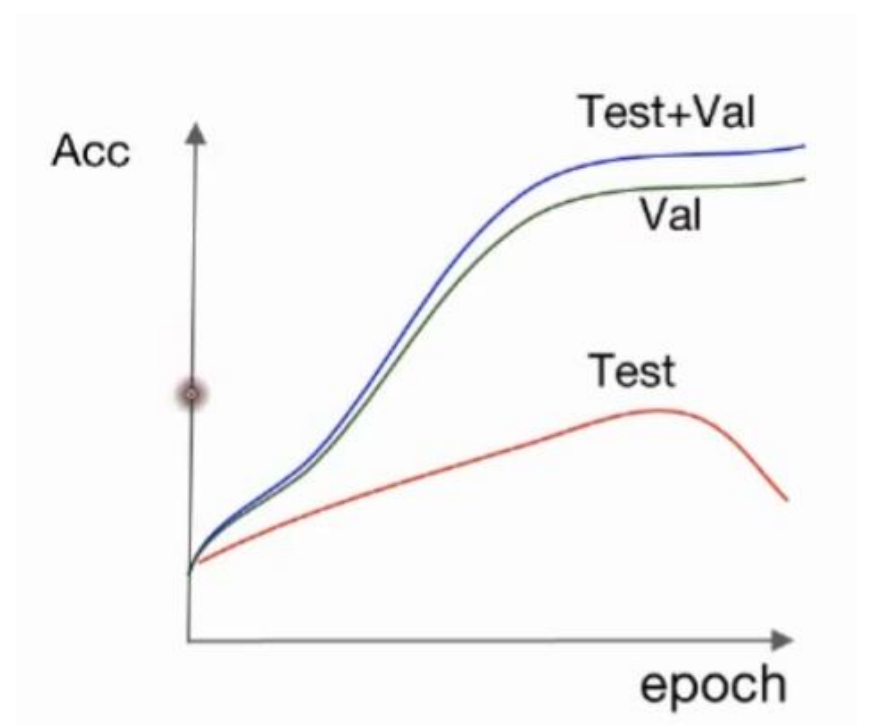
# 데이터 전처리

- Noisy data 처리
  - 수치데이터 (실제 분석에 필요없는, 너무 크거나 작은 값을 배제)
  - NLP : 중요 word만 추출
  - Face image : 얼굴만 추출(머리, 배경 등을 제외)



# Overfitting

- 모델이 훈련 데이터에 너무 과도하게 적합되어 훈련 데이터에 대한 예측 성능이 높지만, 새로운 데이터 또는 테스트 데이터에 대한 성능이 저하되는 현상
- Underfit / just right / overfit(High variance)

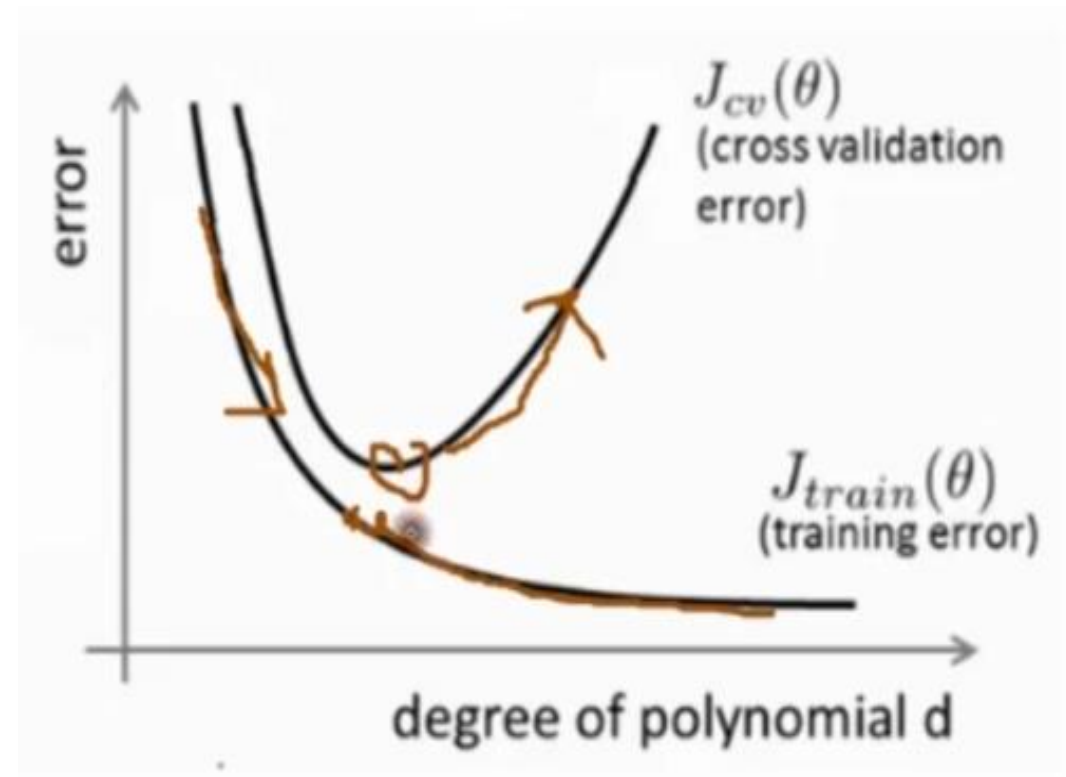


# How to fix overfitting

- 1. Training data를 더 많이.(high variance 해결위해)
- 2. Feature의 차원을 내리는 것 (의미있는 저차원의 공간에서 속성의 의미를 분명히 함) : PCA(sklearn)
- 3. Feature의 수를 증가.(underfitting 해결위해)

# How to fix overfitting

- 학습 횟수를 늘릴때마다 트레이닝 data set의 error는 감소
- 하지만, 새로운 test에는 error가 증가하는 시점.
- => 적절한 최솟값을 찾아서
- **맞춤숫자의 feature set**을 꾸려야.



# How to fix overfitting

- 4. 정규화(loss함수에 term을 추가)
- 특정 변수의 값이 다른 변수들과 큰 차이를 보일 때, 이를 조정해줌. ( $\lambda$ 를 통한 모델의 평균값을 더해줌)
- Overfitting 방지용

- Tf에서도 가능

**Linear regression with regularization**

Model:  $h_{\theta}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m} \sum_{j=1}^m \theta_j^2$$

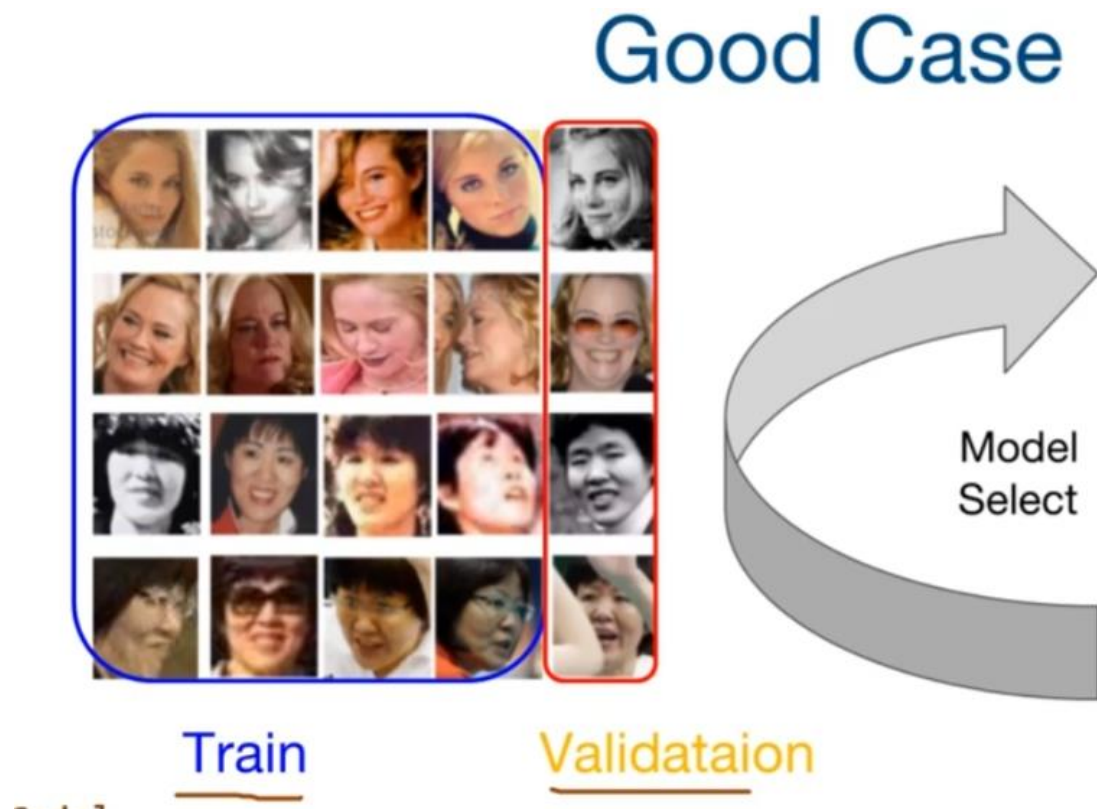
# How to fix overfitting

- Neural network 이해하고도 할 수 있는 추가적인 overfitting solution이 있음.
  - Feature Normalization
  - Regularization
  - More Data (Data Augmentation)
    - Color Jilttering
    - Horizontal Flips
    - Random Crops/Scales
  - Dropout (0.5 is common)
  - Batch Normalization

# DATA SET training

- Mnist data set (숫자 데이터셋)

- 가설 검증



# 이상 감지

- GAN 모델

일반적 data 학습 => 특이 data 감지

=> 피상적내용이었음

# Learning

	Online Learning	Batch(Offline) Learning
Data	Fresh	Static
Network	connected	disconnected
Model	Updating	Static
Weight	Tunning	initialize
Infra(GPU)	Always	Per call
Application	Realtime Process	Stopping
Priority	Speed	Correctness



# Online learning

- Fine tuning
  - 전체 data를 새롭게 training
- Feature extraction
  - 하위 layer 고정, 상위 layer만 fine tuning학습

# Sample data

- Fashion Mnist
  - 패션 샘플데이터
- IMDB
  - 자연어 처리용(영화 평론) 데이터
- CIFAR-100

# 느낀점

- 이번에는 코드를 작성하면서, 직접 실습해보는 느낌이 더 났다.  
(특히 sampler 데이터)
- 이상 감지나 후반부 설명에 있어서 어려운내용이어서 그런지, 선생님이 조금 피상적이고 추상적으로 설명하셔서 아쉬웠다..
- 어려워지고 있다!

# 끝

- 출처: <https://www.boostcourse.org/ai212/lecture/41852?isDesc=false>

감사합니다!