

2020 D&A

MACHINE LEARNING SESSION

2주차

Contents

1. 경사하강법

2. 과적합

경사하강법(Gradient Descent)

머신러닝 및 딥러닝 알고리즘을 학습시킬때 사용하는 방법 중 하나

학습을 통해 모델의 최적 파라미터를 찾는 것이 목표

손실 함수 (loss function)을 최소화하기 위해 매개 변수를 반복적으로 조정하는 과정

경사하강법-손실함수(loss function)

학습을 통해 얻은 데이터의 추정치와 실제 데이터의 차이를 평가하는 지표

값이 작을수록 완벽하게 예측했다고 할 수 있음

손실함수에는 여러 종류가 있음(MSE, Cross Entropy)

손실함수 = 목적함수 = 비용함수

경사하강법-손실함수(loss function)

평균 제곱 오차(Mean Squared Error, MSE)

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

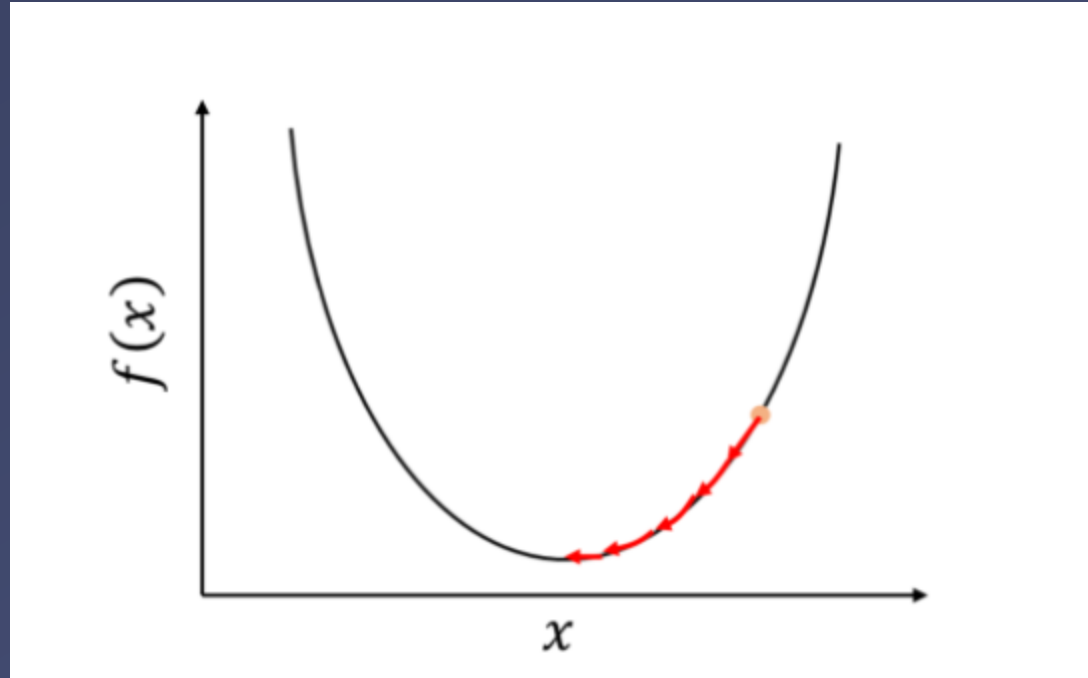
(Y_i : 실제 값, \hat{Y}_i : 예측값)

가장 많이 사용되는 손실함수

모델의 출력 값과 실제값의 거리 차이를 오차로 사용

실제 오차보다 줄어드는 경우를 막기 위해 제곱하여 평균

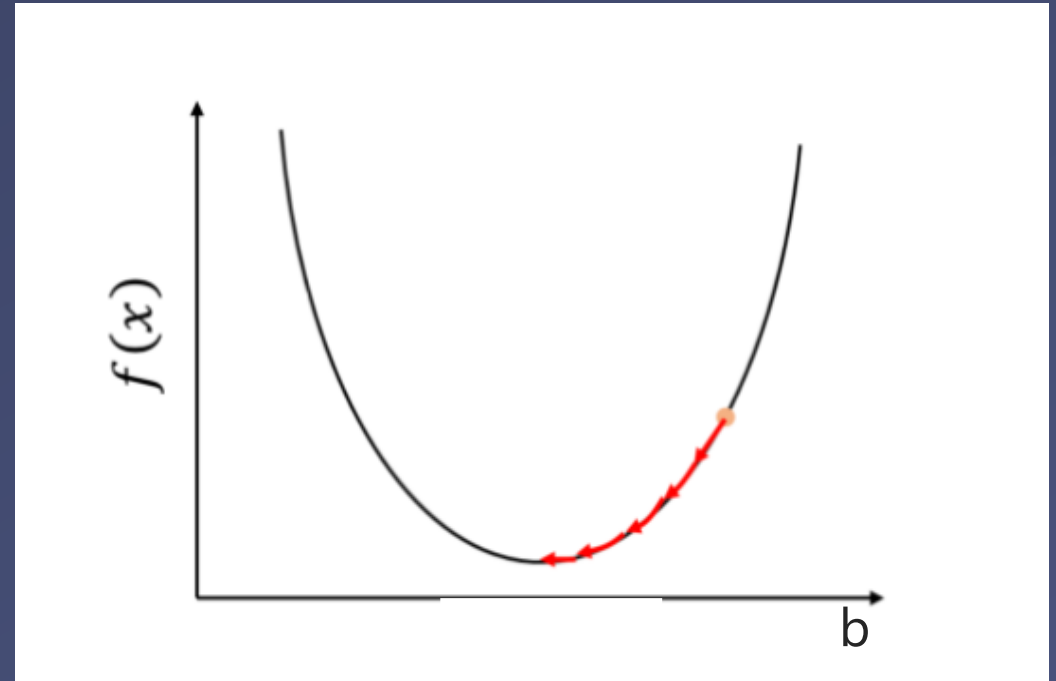
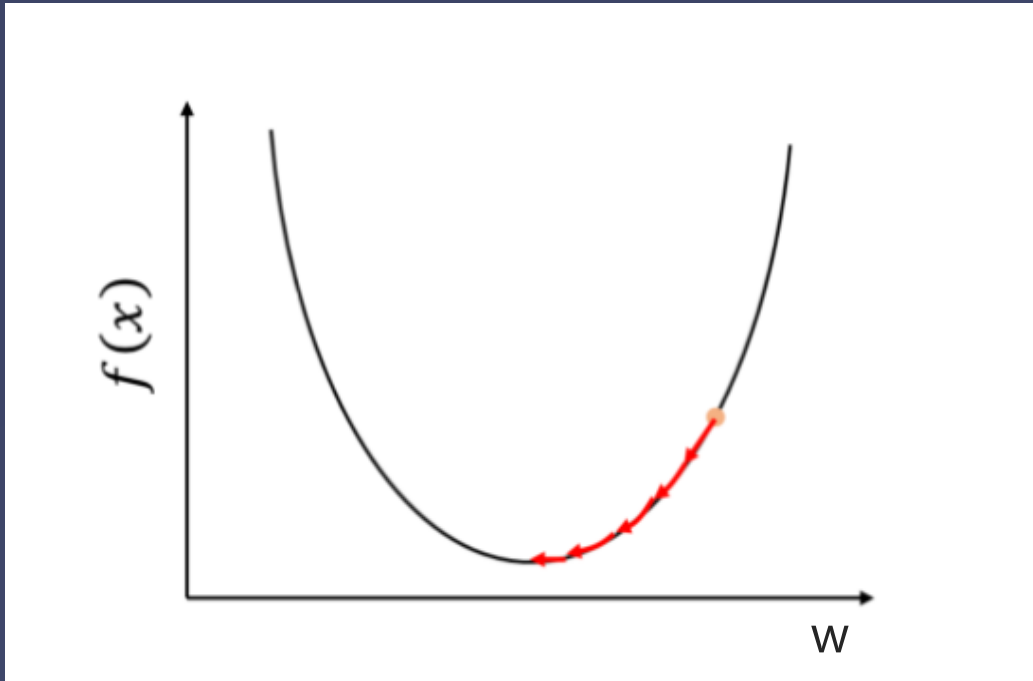
경사하강법(Gradient Descent)



경사를 따라 여러 번의 스텝을 통해 최적점으로 다가감

경사는 미분(gradient)을 이용해 계산

경사하강법(Gradient Descent)

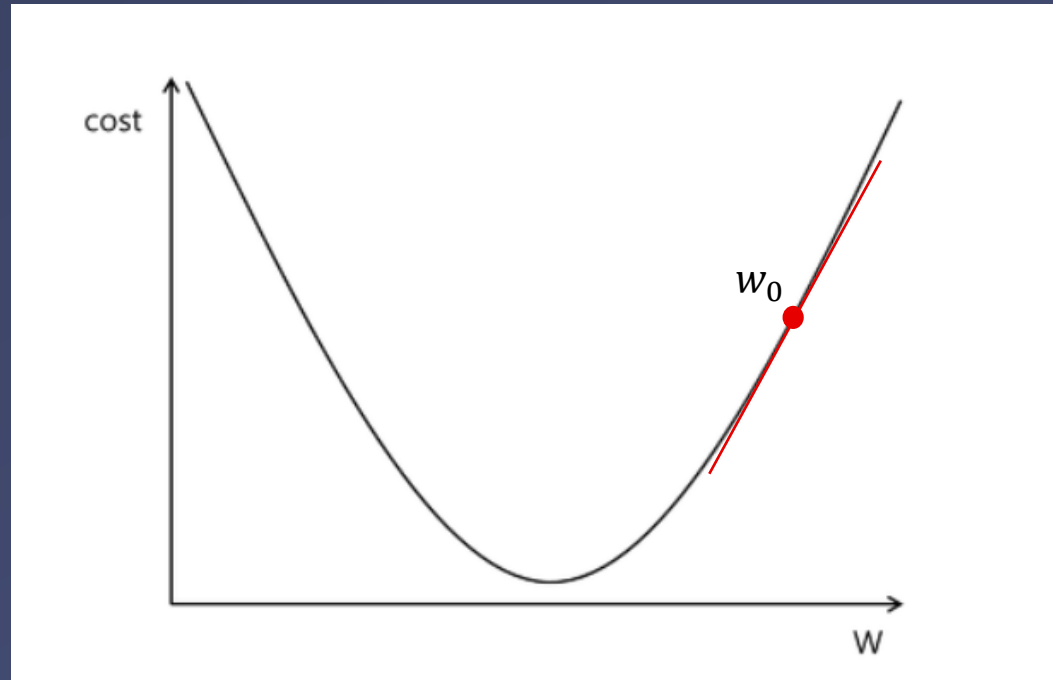


$$y = wx + b$$

최적의 w 와 b 을 찾아내는 것이 목표

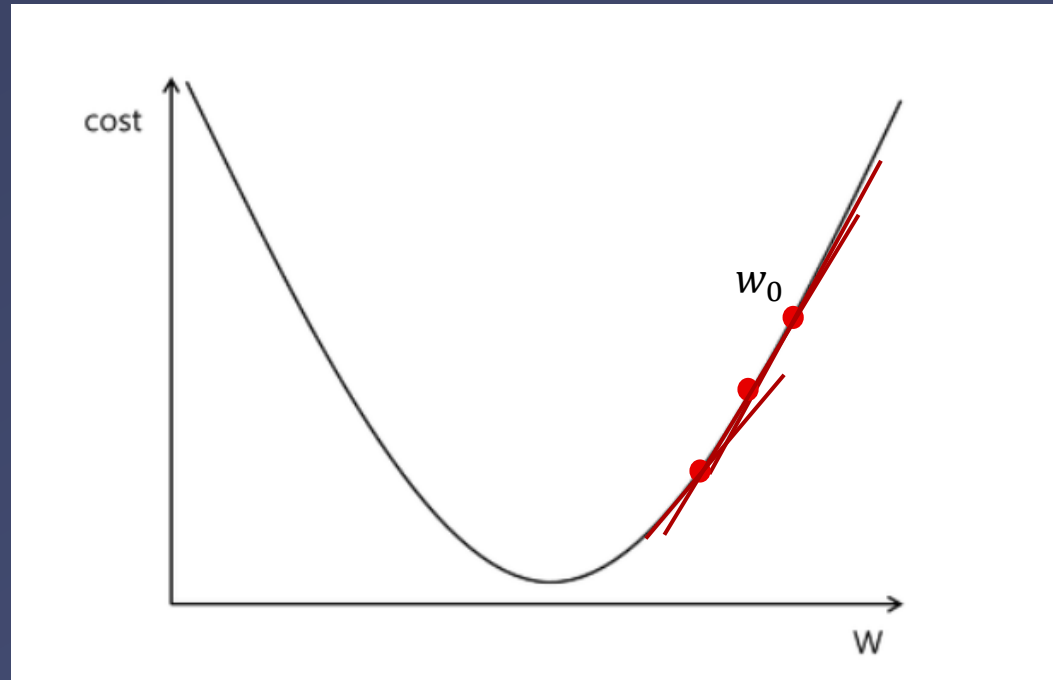
* 편향을 0으로 가정

경사하강법(Gradient Descent)



가중치 w 에 대한 시작점을 임의로 선택하여 기울기 계산

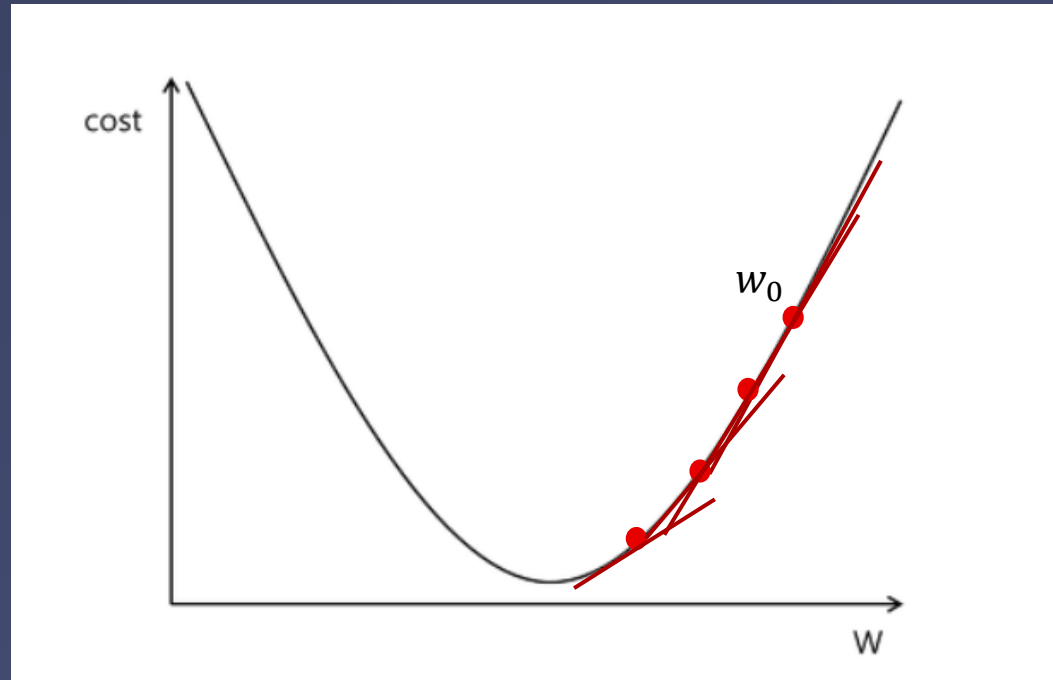
경사하강법(Gradient Descent)



$$w = w - a \frac{\partial}{\partial w} \text{cost}(w)$$

위와 같은 방법이 반복 된 후 최적점으로 수렴함

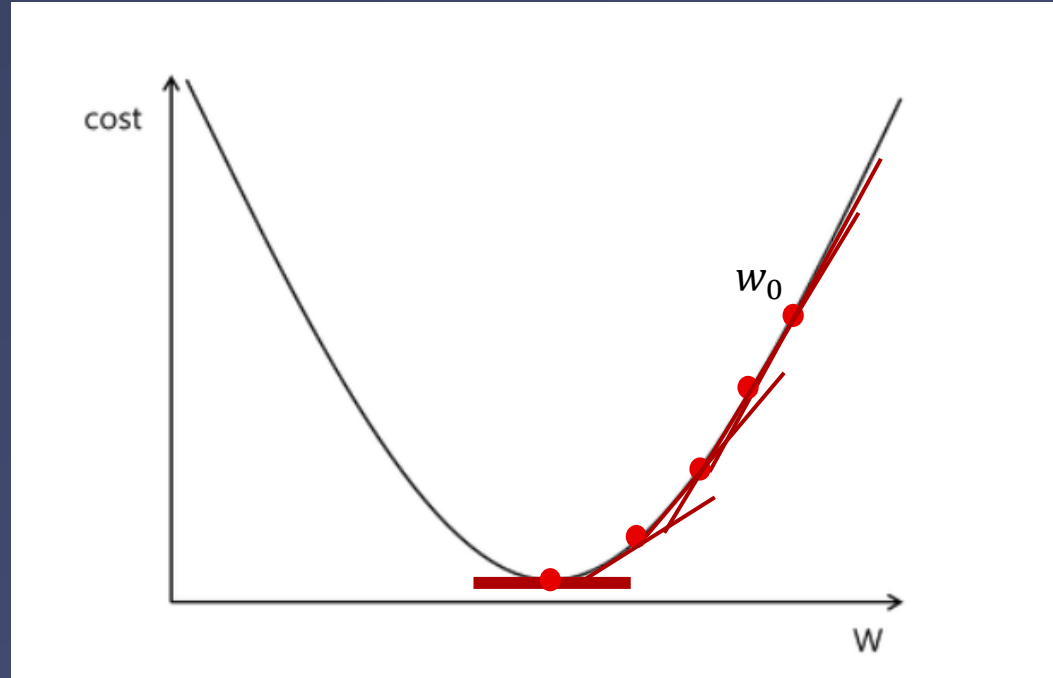
경사하강법(Gradient Descent)



$$w = w - a \frac{\partial}{\partial w} \text{cost}(w)$$

위와 같은 방법이 반복 된 후 최적점으로 수렴함

경사하강법(Gradient Descent)



$$w = w - a \frac{\partial}{\partial w} \text{cost}(w)$$

위와 같은 방법이 반복 된 후 최적점으로 수렴함(기울기가 0)

경사하강법(Gradient Descent)

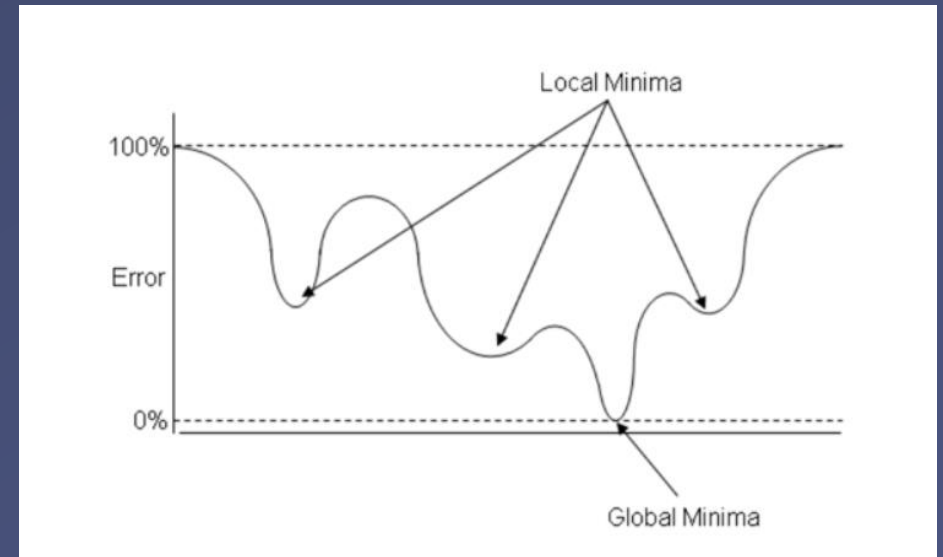
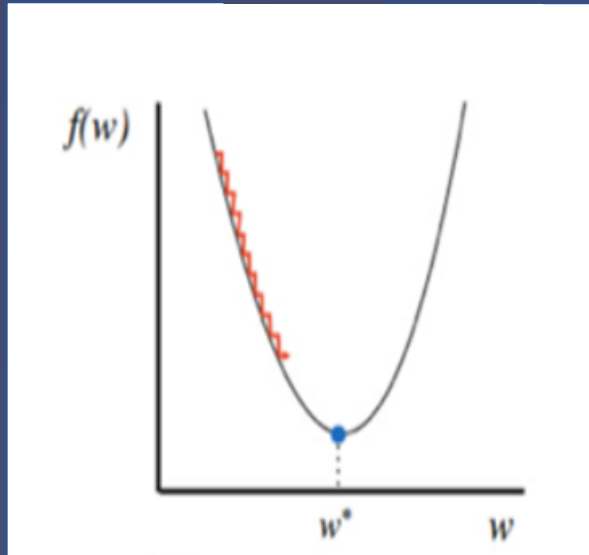
$$w = w - a \frac{\partial}{\partial w} \text{cost}(w)$$

학습률의 크기가 굉장히 중요 !!

학습률이 작을때

반복해야하는 값이 많기 때문에
시간이 오래걸림

지역 최소값에 수렴할 수 있음



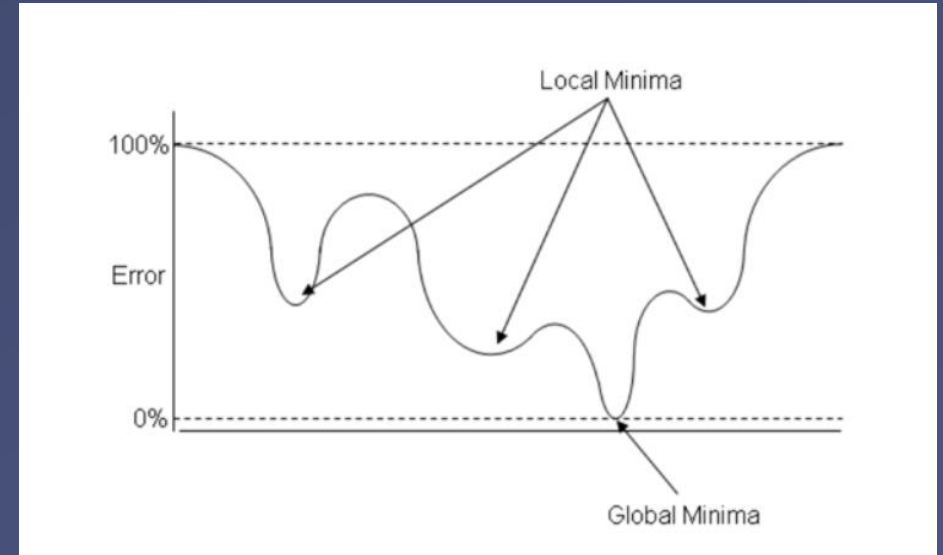
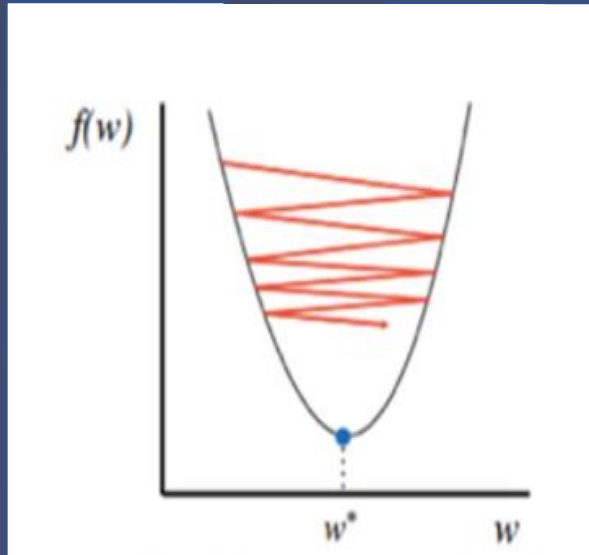
경사하강법(Gradient Descent)

$$w = w - a \frac{\partial}{\partial w} \text{cost}(w)$$

학습률의 크기가 굉장히 중요 !!

학습률이 클때

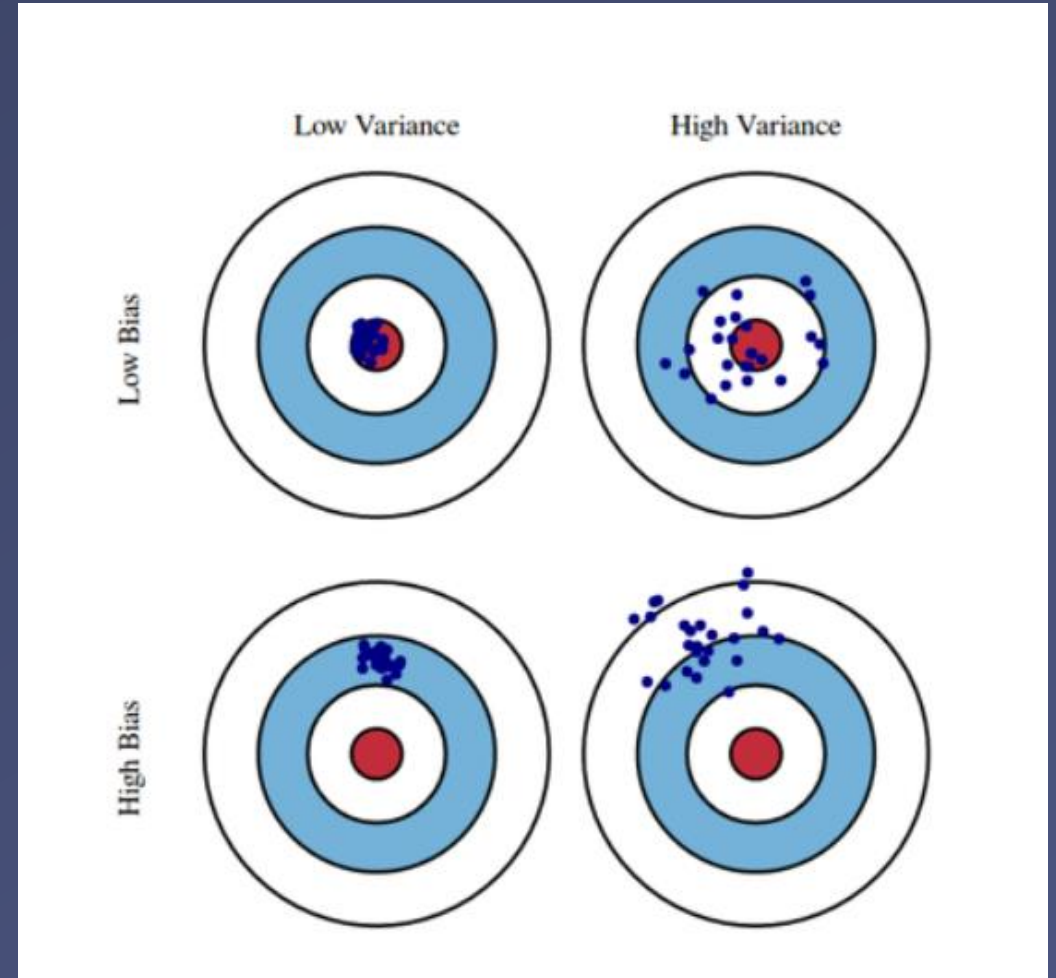
시간은 적게 걸리지만
값이 발산 할 수 있음



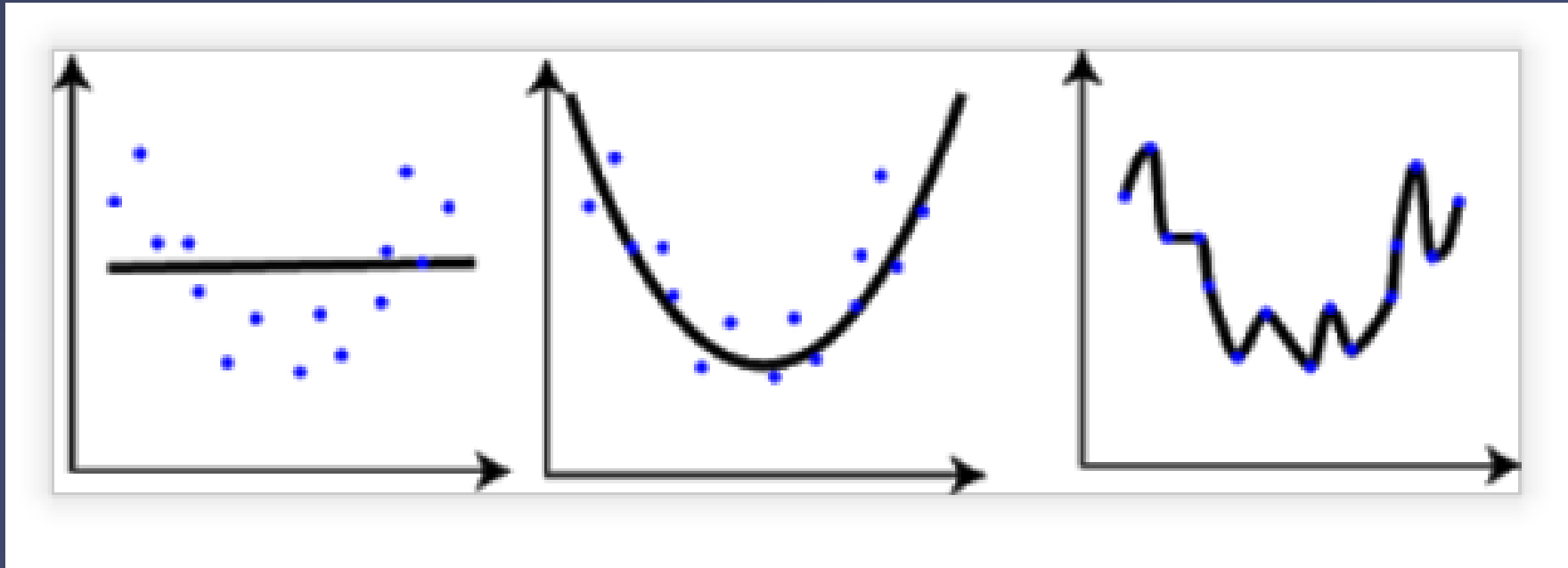
편향과 분산

편향(bias) : 예측이 정답에서 얼마나 떨어져 있는지

분산(variance) : 예측의 변동폭이 얼마나 큰지



과적합(overfitting)



과소적합
(under-fitting)

적정적합
(generalized-fitting)

과대적합
(over-fitting)

과적합(overfitting)

가중치 규제를 추가 - L1 정규화

손실함수에 L1 노름(norm)을 추가

$$\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 + \alpha \sum_{i=1}^N |w_i|$$

가중치(w)의 절댓값의 합을 추가

LASSO

가중치를 0으로 만드는 경향이 강함

과적합(overfitting)

가중치 규제를 추가 - L1 정규화

손실함수에 L1 노름(norm)을 추가

$$L1' = \frac{\partial L1}{\partial w} = \frac{\partial C_0}{\partial w} + \frac{\lambda}{2n} \text{sgn}(w)$$

가중치(w)의 절댓값의 합을 추가

LASSO

가중치를 0으로 만드는 경향이 강함

과적합(overfitting)

가중치 규제를 추가 - L2 정규화

손실함수에 L2 노름(norm)을 추가

$$\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 + \alpha \sum_{i=1}^N |w_i|^2$$

Ridge

가중치를 작게 만드는 경향이 강함

과적합(overfitting)

가중치 규제를 추가 - L2 정규화

손실함수에 L2 노름(norm)을 추가

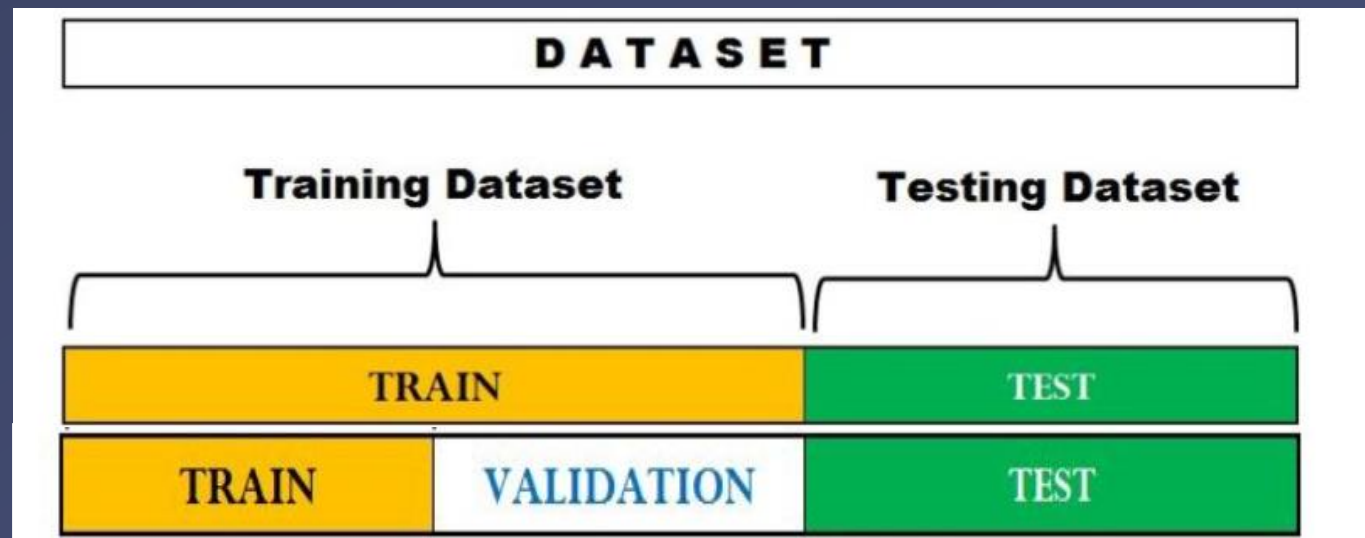
$$L2' = \frac{\partial L2}{\partial w} = \frac{\partial C_0}{\partial w} + \frac{\lambda}{n} w$$

Ridge

가중치를 작게 만드는 경향이 강함

과적합(overfitting)

Hold out

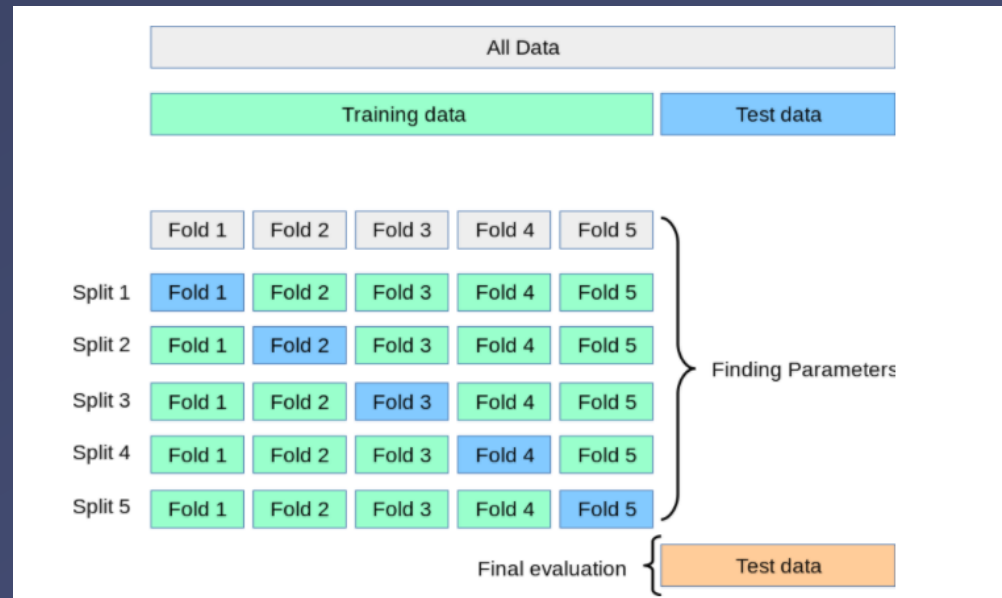


training 데이터를 train data와 validation data로 한번 더 나눔

Validation data로 성능을 확인하며 학습을 진행

과적합(overfitting)

K - fold cross validation



Train data를 k개로 나눔

K-1개는 train으로 나머지 1개의 fold는 validation data로 사용

과제

1. 데이터로 숫자형 변수 만들기

2. 다음주 시험 준비하기

PRODUCT

CLNT_ID	고객 id
SESS_ID	세션 id
HITS_SEQ	하트일련번호
PD_C	상품코드
PD_ADD_NM	상품추가정보
PD_BRA_NM	상품브랜드
PD_BUY_AM	단일상품금액
PD_BUY_CT	구매건수

MASTER

PD_C	상품코드
PD_NM	상품명
CLAC1_NM	상품대분류명
CLAC2_NM	상품중분류명
CLAC3_NM	상품소분류명

SEARCH

CLNT_ID	고객 id
SESS_ID	세션 id
KWD_NM	검색키워드명
SEARCH_CNT	검색건수

SESSION

CLNT_ID	고객 id
SESS_ID	세션 id
SESS_SEQ	세션일련번호
SESS_DT	세션일자
TOP_PAG_VIEW_CT	총페이지조회건수
TOT_SESS_HR_V	총세션시간값
DVC_CTG_NM	기기유형
ZON_NM	지역대분류
CITY_NM	지역중분류

DVC_CTG_NM (1 : desktop, 2: mobile, 3 : tablet)

