1. 회귀 모델

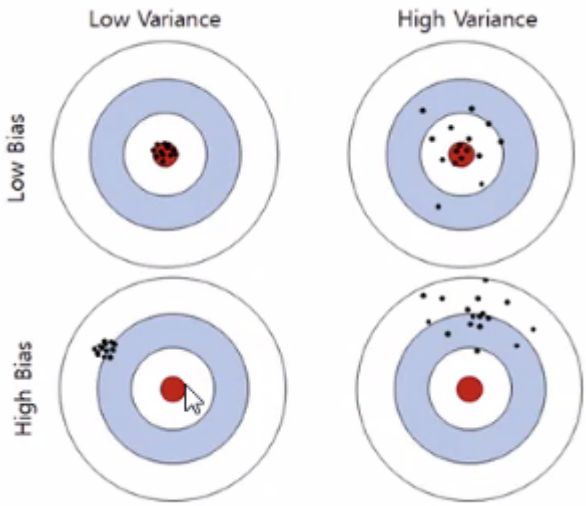
1.1 회귀의 목적

- 회귀
  - loss 최소화하고
  - 가중치 최대화할 수 있는 파라미터를 회기식화해야함 --> 회귀 계수(w1,regression coefficients) 찾기
  - 피쳐(x)와 결정값(y)으로 학습해 최적의 회귀 계수(w)를 찾는 것 --> 오류 최소화
  - 잘 찾았다 = 모든 데이터를 수용할 수 있는 회기선을 찾아낸다

1.2 편향(bias)과 분산(variance)

- 모델이 복잡해지면 overfitting, 단순해지면 underfitting
- trade-off , 반대의 관계 (반비례와는 다름)

- 편향↑, 분산↓ : underfitting (좌측 아래 그림)
- 편향↓, 분산↑ : overfitting (우측 위 그림)
- 최적 모델 (좌측 위 그림) : 편향과 분산이 적절하다
-



- 정리
  - 고편향: 피쳐 하나만 가지고 이야기함, 한 방향성으로 치우침
  - 고분산 : 지나치게 높은 변동성(=리스크가 크다) * 분산: 평균으로부터 얼마나 떨어져 있는가(편차)

- 회기계수가 높다
- 오버피팅

- 이 현상을 조절하기 위한 것이 "규제"

1.2 규제(L1,L2)

규제의 목적

- 다차원의 오버피팅을 막기 위해 주요피쳐만을 다룸
  - 모델이 복잡해질수록 오버피팅이 남
  - 필요없는 피쳐 학습에 약하게 가담(주요 피쳐만 학습에 가담)시켜 올바른 학습을 가능하게 함
  - 일반화 한다
- 규제의 필요성 : cost 비용 조절 --> 정상적 학습 가능하게 하기 위함
  - 회기계수(w)가 커질 경우 머신러닝은 편향되게 학습한다(고편향) --> 규제 상수 값을 작게 만들면 최소 비용을 유지가능
  - 회기계수가 너무 작을 경우 학습이 제대로 되지 않음 --> 규제 상수 값을 크게
- 규제가 있는 모델은 고급 모델 : 가중치가 지나치게 커지는 현상을 막기 위함, 편향된 학습을 하지 않게 함 --> 비용 최소화
- W가 크면 규제상수(a)를 낮춰야 오버피팅을 피할 수 있다.+

L1과 L2

- 벡터(Vector) :크기와 방향을 가지고 있는 기하학적 수치
- 노름(norm) : 백터의 크기+를 측정하는 방법
  - 백터의 크기(magnitude) : 1차(scaler), 2차(평면), 3차(공간)....
  - 백터의 길이(length) : 백터와 백터 사이의 길이
    - 유클리디안 거리(피타고라스 정의,삼각함수: a^2+b^2=c^2)-->머신러닝 default
    - 맨하튼 거리(taxicab)

1. L1(Lasso)

- 변별력 최소화, 필요없는 피쳐 삭제
- 맨하튼 거리(taxicab)로 백터 길이 측정
  - w 절대값 합 --> loss 값 그대로 사용 --> outlier에 민감하지 않음, outlier가 심한 경우 사용

2. L2(Lidge)

- 중요하지 않은 피쳐 약하게,가중치(기울기) 0에 가깝게 (y값에 주는 영향 최소화)
  - 기울기가 크면 클수록 주요 피쳐로 여김

- 유클리디안 거리(루트)로 백터 길이 측정
  - w의 제곱합 --> 실제값과 예측값이 차이가 커지면 가중치가 커짐

    ---> outlier가 심한 경우 사용 불가(일반적인 경우에 사용)

3. Elastic-Net
   - L1+L2

1.3 경사 하강

- 편미분
  - y값 변화량/x축 변화량 = 순간기울기=미분
  - 원하는 상수만을 가지고 미분을 한다(다른 상수 신경 쓰지 않음) = 편미분
  - 비용함수(SSE(잔차=(예측값-실제값)^2)->편미분->순간기울기(가중치,w) 구하기
- 경사 하강(convex)
  - learnign rate
    - 발산한다 : 목표지점을 넘어감
    - 적정하게 learning rate를 구하는 것이 머신러닝에서 중요
    - 보통 default 0.001로 잡음
    - 최소 비용함수를 찾는 알고리즘
  - x= 가중치, y=cost --> cost가 최소가 되는 (0) x의 값을 찾아라

1.4 변수

- 선형 : 차수 1차
- 비선형 : 차수가 다차, 복잡한 형태의 수식(모델)

- 독립변수 (x) : 변량
  - 단변량 : 학습에 가담하는 독립변수가 1개
  - 다변량 : 학습에 가담하는 독립변수가 여러개
    - 다변량 선형 회귀
      - 각 피쳐에 서로 다른 가중치를 계산하여 합산한 값을 y
      - 행열 내적: H(X)=XW --> ㄱ 방향으로 계산

2. 검증

- 회귀에서 target은 스케일링 필수 --> 일반적으로 로그스케일링
- 원핫인코딩
  - 원본에서 상관도가 없다고 나오는 피쳐들 중에서 원핫인코딩으로 값들을 피쳐화시키면 피쳐와 피쳐간 의미있는 상관도가 생길 수 있음(원본 피쳐 값 중 특정 몇개의 값만 상관도가 높을 경우)
  - 바이너리성 데이터(ex.남/여) 원핫인코딩 시 반드시 하나 삭제 --> 다중 공선 걸림(overfitting) => 그냥 하지 말기
  - 다중공선이 걸렸음에도 예측에 지대한 영향을 미치면 삭제하면 안됨 --> 피쳐상태로는 의미가 없었지만 원핫인코딩 후 피쳐가 된 특정 값이 영향을 많이 미치는 경우
    - EDA로 반드시 확인해보기
    - 원핫 인코딩 후 다중공선이 걸리는 피쳐들을 확인한 후 영향도가 높은 것만 원핫인코딩 처리하면 됨