MLE & MAP

<배경>
- Bayes rule을 이용하여 class를 예측하고자 한다면, likelihood의 확률밀도함수를 알아야 함
-> 데이터로부터 직접 decision policy를 구할 순 없을까?
-> 임의의 parameter로 이루어진 모델로 모델링을 한 후에, 이 모델이 가장 데이터를 잘 설명할 수 있도록 parameter을 구해내는 건 어떨까?
=> MLE(maximum likelihood estimation) & MAP(maximum a posterior)
딥러닝의 loss function에도 이용

<문제 정의>
문제 : regression 문제(x(키) -> y(몸무게))
- 실제 구한 D(데이터 집합-n개의 데이터)
$D = [(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)]$
- 파라미터를 통한 모델링
$y(x|w)$: w라는 파라미터가 있을 때, 해당 파라미를 갖고 x(키)를 넣으면 y(몸무게)를 배출하는 함수(모델) -> *이러한 모델을(선형/이차식) 띈다는 가정이 들어감*
-> 만약 파라미터를 잘 학습하여 완벽한 모델이라면, 해당 모델의 output이 target 값이지만, 그러한 것은 불가능. $t = y(x|w)$
-> 어떤 x에 대한 실제 target 값은 모델을 통해 예측한 예측값을 평균으로 하고, 특정 값을 표준편차로 하는 정규분포를 띄어! $t \sim N(y(x|w), \sigma^2)$

$$P(t|x, w, \sigma) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(t - y(x|w))^2}{2\sigma^2}}$$

- σ : 문제의 특성에 따라서 지정한 값 -> 추후 중요하게 작용할 때가 옴
(만약 input 값이 같더라도 다양한 output 값이 가능한 문제라면 표준편차를 크게, 가능하지 않은 문제라면 표준편차를 작게 설정)

- 그렇다면 $y(x|w)$모델을 통해 우리가 획득한 데이터셋이 나올 확률

$$P(D|w) = \Pi_{i=1}^{N} P(t_i|x_i, \sigma, w) = \Pi_{i=1}^{N} \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(t_i - y(x_i|w))^2}{2\sigma^2}}$$

-> x, 표준편차, w(파라미터)로 각 target값의 분포를 추정했고, 이 분포에 따랐을 때, 실제 target 값일 확률의 결합분포확률 -> 이 모델에 따랐을 때, 실제 데이터셋의 확률
-> 가장 좋은 모델, 파라미터는 이 $P(D|w)$를 최대로 만드는 모델. -> 이러한 모델을 찾자.
-> x들이 주어졌을 때, y들이 독립이어야 함(앞 y의 영향을 받고 그러면 안됨) -> 오차간의 독립성 가정

<Prior, Likelihood, Posterior 정의>
- 구하고자 하는 것 : weight(파라미터)

- 가지고 있는 것 : Dataset
- Posterior : 주어진 대상이 주어졌을 때, 구하고자 하는 대상의 확률분포 : $P(w|D)$
- Likelihood : 구하고자 하는 대상을 모르지만 안다고 가정했을 경우, 주어진 대상의 분포 : $P(D|w)$
- Prior : 주어진 대상과 무관하게 , 상식을 통해 우리가 구하고자 하는 대상에 대해 이미 알고 있는 사전 정보. 연구자의 경험을 통해 정해줌. : $P(w)$

<MLE 계산>

문제 : $P(D|w)$을 최대로 하는 weight를 찾는 것

$\arg max_w(likelihood) = \arg max_w(P(D|w))$

= $\arg max_w(\log(likelihood)) = \arg max_w(\log(P(D|w)))$ (log(x)가 증가함수이고, 0<P(D|w)<1 이기 때문)

=> $\log(P(D|w)) = \log(\Pi_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(t_i - y(x_i|w))^2}{2\sigma^2}})$

$$= \sum_{i=1}^{N} \log(\frac{1}{\sqrt{2\pi}\sigma} e^{-(t_i - y(x_i|w))^2})$$

$$= \sum_{i=1}^{N} (\log(e^{-(t_i - y(x_i|w))^2}) - \log(\frac{1}{\sqrt{2\pi}\sigma}))$$

$$= \sum_{i=1}^{N} (-(t_i - y(x_i|w))^2 - \log(\frac{1}{\sqrt{2\pi}\sigma}))$$

$$=> \arg max(\sum_{i=1}^{N} (-(t_i - y(x_i|w))^2 - \log(\frac{1}{\sqrt{2\pi}\sigma})))$$

이때, 표준편차와 pi는 상수이기 때문에 영향을 주지 않음

=> $\arg max(\sum_{i=1}^{N} -(t_i - y(x_i|w))^2)$

= $\arg\min(\sum_{i=1}^{N}(t_i - y(x_i|w))^2)$

-> 해당 식을 최소화하는 weight 파라미터가 가장 최적 파라미터

-> 해당 식은 L2 loss와 같음 -> 딥러닝, 머신러닝에서 L2 loss를 쓰는 이유

-> 선형회귀분석은 L2 loss를 최소화하여 weight를 구함 -> y의 정규성, 오차의 독립성에 대한 가정이 들어 있는 것임. 오차의 등분산성은 선형회귀모델의 특성과 관련이 있음 -> 따라서 안정성, 정확도와 관련

<MAP>

- posterior과 likelihood의 가장 큰 차이 : prior의 유무
- 만약 prior(사전지식)을 반영하고 싶다면 posterior방식을 사용하는 것이 좋음
- prior을 반영한다는 것 외의 모든 가정(ex t~N(y(x|w),분산))은 동일

▶ 사전지식을 반영하는 것이 좋은 경우

- 매우 강력한 사전 지식을 가지고 있을 때 ex) 연어가 잡힐 확률 : 1/10
- 구하고자 하는 대상에 특정 제약조건을 넣어주고 싶을 때 ex) weight가 0 주변에 분포

<목적식>

$\arg max_w P(w|D)$

$P(w|D) = \frac{P(D|w)P(w)}{P(D)} = \frac{P(D|w)P(w)}{\int P(D\cap w)dw} = \frac{P(D|w)P(w)}{\int P(D|w)P(w)dw}$ *(마찬가지로 해당 모델을 띈다는 가정이 들어가기 때문에 w가 전체망라가 되고 P(D)를 저렇게 표현 가능)*

-> prior인 $P(w)$에 대한 가정 : weight에 대한 사전지식이 없기 때문에 나름대로의 제약 조건을 넣음 > 0 주위에 분포하는 정규분포

$w \sim N(0,\sigma_w^2)$

$P(w) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{w^2}{2\sigma_w^2}}$

$\int P(D|w)P(w)dw$ -> $\int_{-\infty}^{\infty} P(D|w)P(w)dw$는 P(D|w)와 P(w)를 모두 알고 Dataset이 정해져 있고, w에 대한 적분이기 때문에 상수값.

$\eta = \frac{1}{\int P(D|w)P(w)dw}$

=> $\arg max_w P(w|D)$ 로그를 씌워서 변환

$= \arg max_w \eta P(D|w)P(w)$

$=> \arg max_w \log P(w|D)$

$= \arg max_w (\log\eta + \log P(D|w) + \log(P(w))$

이때, logP(D|w)를 최대화 하는 것은 $L(w) = \sum_{i=1}^{N}(t_i - y(x_i|w))^2$을 최소화하는 것과 같음

=> $\arg max_w (\log\eta - L(w) + \log P(w))$ *(argmax로 들어가기 때문에 -L(w))*

logP(w) 식 대입

=> $\arg max_w (\log\eta - L(w) - \log(\sqrt{2\pi}\sigma_w) - \frac{w^2}{2\sigma_w^2})$

η, π, σ_w는 모두 상수이기 때문에 관련 term들 제거

$=> \arg max_w (-L(w) - \frac{w^2}{2\sigma_w^2})$

= argmin$(L(w) + \frac{w^2}{2\sigma_w^2})$

=> 표준편차를 적당히 잡아주면 L2 Regularization과 동일

(weight가 평균이 0인 정규분포를 띈다는 가정을 하면 L2 regularization 방식을 적용한 L2 loss를 최소화 하는 식) - (Laplacian Distribution을 prior로 걸어주면 L1 regularization)

즉 MAP는 해당 dataset이 해당 모델틀을 따르고, 1. y가 정규분포, 2. y들이 서로 독립성(y는 x의 영향만 받아야 하고, 다른 y의 영향은 받으면 안됨, 오차가 서로 독립성), **3. weight의 prior에 대한 가정**(ex w~N(0,분산)) 이 MLE와 달리 추가로 들어감

출처 : https://hyeongminlee.github.io/post/bnn002_mle_map/