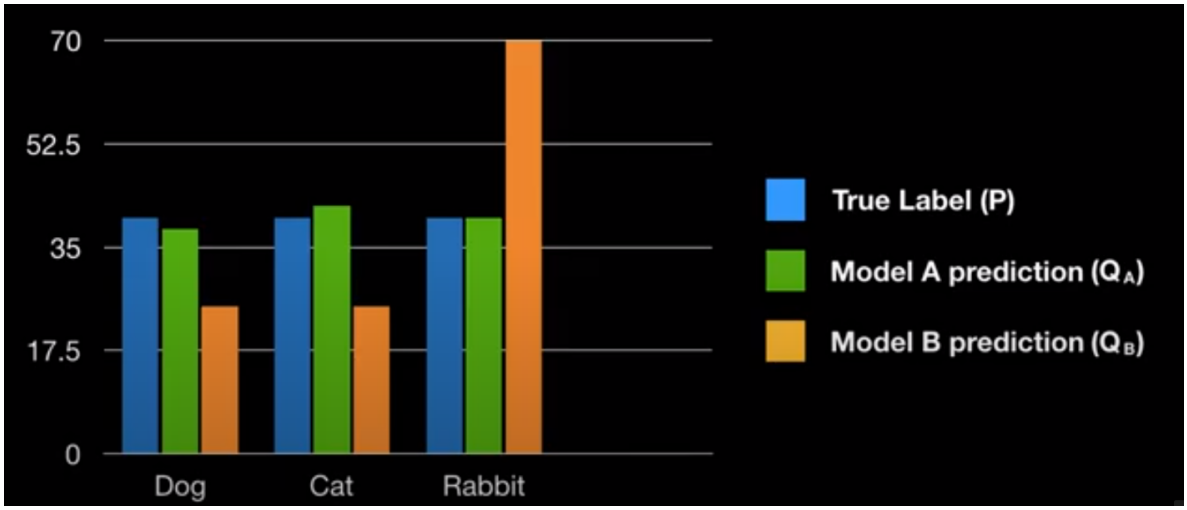
출처가 명시되지 않은 모든 자료(이미지 등)는 조성현 강사님 블로그 및 강의 자료 기반.

[KL Divergence]

1. 아이디어



출처: https://www.youtube.com/watch?v=7GBXCD-B6fo

강아지와 고양이, 토끼를 예측하는 분류 문제에 대해 두 개의 머신러닝 모델 A, B를 만들었다고 하자. 어떤 것이 더 좋은 모델인지 어떻게 평가할 수 있을까?

A 모델의 경우, 예측한 라벨의 분포가 실제 라벨의 분포와 거의 차이가 나지 않는다. 반면, B 모델이 예측한 분포는 실제 라벨의 분포와 매우 차이가 난다. 이 차이를 가지고 예측값의 분포가 더 비슷한 모델을 선택하거나, 혹은 예측값의 개수 중 틀리게 분류한 것의 개수를 계산하는 등 나름의 기준을 세울 수 있겠다.

이제 시각을 *확률 분포*의 차원으로 돌려 보자.

모델의 예측값이 다르다는 것은 모델이 예측에 사용한 확률 분포의 분포가 다르다는 것을 의미한다. 예측을 위해 만들어 낸 모델이 실제 분포와 일치할수록 모델은 정확한 값을 예측해낼 것이다. 따라서 **확률 분포의 차이**를 **수치화**한다면, 모델의 예측력을 평가할 수 있게 된다.

어떤 과정을 거치는지는 상관이 없다. 예컨대 D라는 공식이 있고, 그 D라는 공식에 의해 P와 P의 확률 분포를 수치로 나타내고, Qa와 P의 확률 분포의 차이, 그리고 Qb와 P의 확률 분포의 차이를 수치로 나타낼 수 있다고 해 보자. P와 P의 경우 차이가 없을 테니 0.0이 나오고, 더 비슷한 분포일수록 작은 수치값이 나오도록 공식을 설계하자. 그렇다면, 이 수치에 의해 확률 분포의 차이를 알아내고, 더 작은 차이를 갖는 확률 분포를 정답에 가까운 확률 분포라고 말할 수 있게 될 것이다.

2. 개념

KL Divergence(Kullback-Leibler Diergence)는 위의 아이디어에서 출발한 함수다. 두 확률분포의 차이를 계산하기 위한 공식으로, 알고자 하는 이상적인 분포(P, 혹은 *알지 못하는 분포*)가 있을 때, 그 분포를 근사하는 다른 분포(Q, 혹은 *알 수 있는 분포*)를 사용해 P를 알아내고자 하는 상황에서 발생할 수 있는 **정보 엔트로피의 차이**를 계산한다.

이를 식으로 나타내면 다음과 같다.

$$D_{KL}(P||Q) = \Sigma(P(x) \times log(\frac{P(x)}{Q(x)}))$$

위의 KL Divergence 식을 이용하면 두 확률분포 P, Q의 유사성을 *정량적으로* 측정할 수 있게 된다. 두 분포가 완전히 같을 때 log 안의 값이 1이 되어 `D_KL` 값은 0이 될 것이므로, KL Divergence 값이 0에 가까울수록 두 확률분포 P와 Q가 비슷한 분포이다.

3. 상대 엔트로피

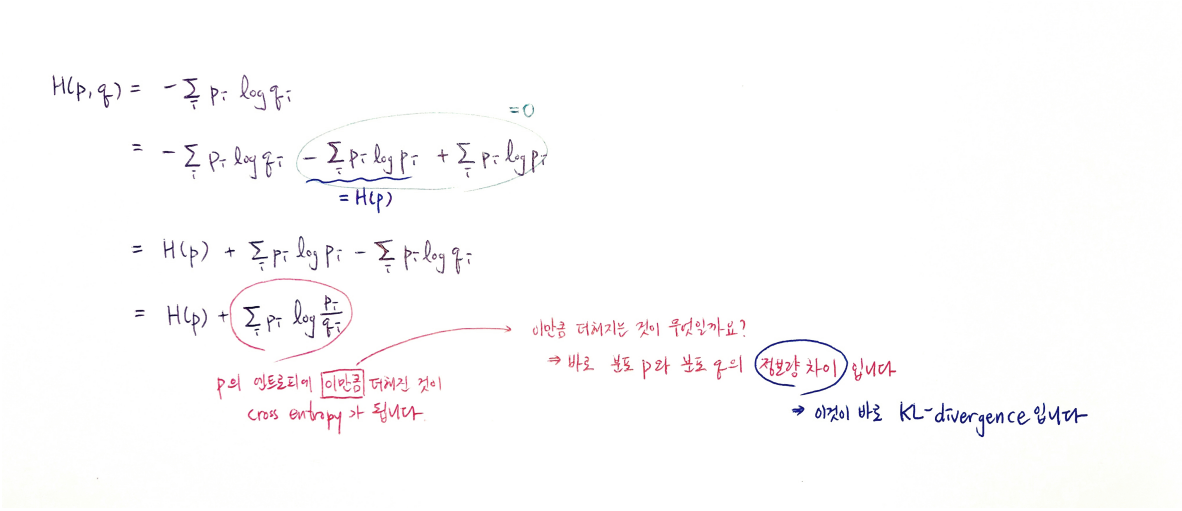
KL Divergence와 엔트로피는 무슨 관련이 있는 것일까?

앞서 두 분포의 크로스 엔트로피가 두 분포 사이의 교차 정보량을 나타내는 것으로서, 실제 정보값에서 불확실한 정보를 포함하고 있는 값이라고 했다. 실제 정보가 가지고 있는 정보량의 크기가 엔트로피이므로, *크로스 엔트로피*는 **실제 정보의 엔트로피 값과 불확실한 정보의 엔트로피 값을 포함하는 값**이 된다.

그렇다면 크로스 엔트로피에서 엔트로피 값을 빼면 정보량의 차이를 계산할 수 있게 된다.

$$\Delta_i = -log(q_i) + log(p_i) \rightarrow E(\Delta_i) = \sum_i p_i \Delta_i = -\sum_i p_i log(q_i) + \sum_i p_i log(p_i) = D_{KL}(p||q)$$: KL Divergence

결과적으로, 크로스 엔트로피에서 엔트로피를 빼면 KL Divergence 값이 된다. 식 자체가 정보량 차이(`△i`)를 나타내고, 크로스 엔트로피에서 엔트로피를 뺀 것이므로 '상대'적이라 하여, 상대 엔트로피라고도 부른다.



출처: [순록킴의 블로그] (https://hyunw.kim/blog/2017/10/27/KL_divergence.html)

위와 같은 KL Divergence는 다음과 같은 성질(수학 증명 참고)을 갖는다.

첫째, 항상 0보다 크거나 같다.

$$\forall\, 0 < x < 1, lnx \leq x - 1$$
$$\forall\, y > 0, ylnx \leq y(x-1)$$
$$\Downarrow\ y = p_i, x = \frac{q_i}{p_i}$$
$$-\Sigma\ (p_i\ ln\frac{q_i}{p_i}) \geq -\Sigma\ (p_i(\frac{q_i}{p_i} - 1))$$
$$-\Sigma\ (p_i\ ln\frac{q_i}{p_i}) \geq -\Sigma q_i + \Sigma p_i$$
$$\Downarrow \Sigma p_i = 1, \Sigma q_i = 1$$
$$-\Sigma\ (p_i\ ln\frac{q_i}{p_i}) = D_{KL}(P||Q) \geq 0$$

둘째, KL Divergence의 값이 0이라면, 확률 분포 P와 Q가 같다.

$$D_{KL}(P||Q) = 0\ \ if\ \ and\ \ only\, if\ P = Q$$

셋째, $DKL(p||q)$는 대칭적 구조가 아니다. 즉, 어떤 분포의 값을 기준으로 할 것인가에 따라 달라진다.

$$generally,\ D_{KL}(P||Q) \neq\ D_{KL}(Q||P)$$

> *강사님의 가르침*
>
> KL Divergence를 대칭적 구조로 변형한 것을 Jenson-Shannon Divergence(JSD)라고 한다. 나중에 GAN 학습 시 다시 등장하게 되는데, 식은 다음과 같다.
>
> $$JSD(p||q) = \frac{1}{2}D_{KL}\left(p||\frac{p+q}{2}\right) + \frac{1}{2}D_{KL}\left(q||\frac{p+q}{2}\right) \qquad JSD(p||q) = JSD(q||p)$$

4. 크로스 엔트로피와의 관계

이제 KL Divergence가 실제 확률 분포와 예측된 확률 분포가 얼마나 다른지 그 정도를 나타내주는 수치임을 알았다.

> 조금 더 정보 이론적인 측면에서 접근하자면, KL Divergence는 실제 정답 값을 예측값과 비교함으로써, Q로 예측한 정보량이 P의 원래 정보량과 같아지기 위해, 어느 정도의 엔트로피가 더 필요한지를 나타내는 값이다.

그렇다면 모델 학습 시 왜 손실함수로서 KL Divergence를 사용하지 않는 것일까?

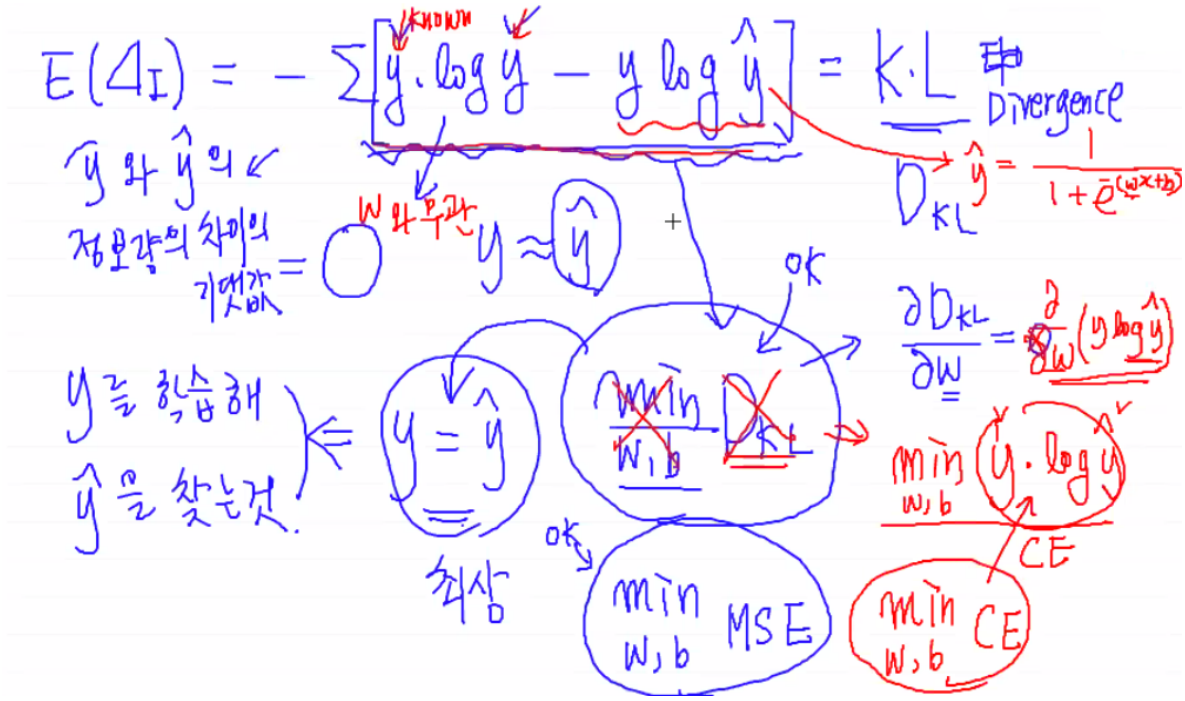
결과적으로는, Cross Entropy나 KL Divergence나 무엇을 손실 함수로 쓰든 최소화의 결과는 같아지기 때문이다.

$$\text{Cross Entropy (CE)} = -\sum_i p_i log(q_i) = H(p,q) \qquad D_{KL}(p||q) = H(p) - H(p,q)$$

KL Divergence 식을 다시 나타내면 엔트로피 식에서 크로스 엔트로피 식을 뺀 것이다. 그런데 모델 학습 시, 학습을 통해 파라미터가 조정되면서 바뀌는 값은 Q이다. 원래 확률 분포인 P는 정해져 있기 때문에, 학습이 진행되는 과정에서 P의 엔트로피는 변할 수가 없다. 상수라는 말이다.

즉, KL Divergence 식에서 엔트로피를 나타내는 부분인 `H(p)` 부분은 예측 확률분포 Q와 *무관*하다. 머신러닝 모델링을 통해 Q 분포를 통해 P 분포를 추정하려면, Q 분포의 파라미터를 조정해 가면서 **크로스 엔트로피가 최소가 되는 지점**을 찾으면 된다는 말이다.

"결과적으로, Cross Entropy를 쓰든, KL Divergence를 쓰든 최소화하고자 하는 것은 같다."



더 공부해야 할 부분

아래에 해당하는 교재 부분은 아직 잘 이해하지 못하겠다. 나중에 다시 공부하는 것으로. ~~(H / ..... 머리아파 힘들어 내...)~~

- y^t 분포가 one-hot encoding인 경우 (classification 문제의 경우) entropy 항 $H(y^t) = 0$ 이 되어 cross entropy (CE)만으로 두 분포의 유사성을 나타낼 수 있다.
- 아래 절차에 의하면 CE를 minimize하는 방향으로 학습시키면 y^p가 y^t에 점점 가까워 짐을 알 수 있다.

$$y^t = [1, 0, 0] \quad \leftarrow True\ value\ (p)$$
$$y^p = [0.7, 0.1, 0.2] \quad \leftarrow Prediction\ value\ (q)$$

$$D_{KL}(y^t || y^p) = -\sum_i y_i^t log(y_i^p) \sim Cross\ Entropy\ (CE)$$

$$Jensen's\ inequality\ for\ \log(x):$$

$$log(E[y_i^p]) \geq E[log(y_i^p)] \rightarrow log\left(\sum_i y_i^t y_i^p\right) \geq \sum_i y_i^t\ log(y_i^p) = -CE$$

$$\sum_i y_i^t y_i^p = 1 * 0.7 + 0 * 0.1 + 0 * 0.2 = 0.7 = Pr(y_1^t = y_1^p)$$

$$log\left(Pr(y_1^t = y_1^p)\right) \geq -CE$$

$$Pr(y_1^t = y_1^p) \geq e^{-CE}$$

$$min(CE) \rightarrow max\left(Pr(y_1^t = y_1^p)\right) \rightarrow [\ y^p \rightarrow y^t\ ]$$

- CE를 작게 만들수록 Low bound가 증가하므로 $y_1^t = y_1^p$ 일 확률이 증가한다. y^p가 y^t에 점점 가까워진다.
- 이 원리에 의해 loss function으로 cross entropy를 사용할 수 있다.
- Classification 문제에 대해서는 MSE 보다 CE를 사용하는 것이 더 성능이 좋다고 알려져 있다.