

4. Deep Nerual Network with MLE : Equations

MLE Equations

dataset x, y 에서 negative log likelihood를 최소화하는 seta를 gradient descent를 통해 찾는 일련의 과정을 식을 통해 표현하였다.

$$D = \{(x_i, y_i)\}_{i=1}^N$$

dataset θ 라는 함수를 통해 x_i 가 y_i 값을 내

$$\hat{\theta} = \underset{\theta \in \Theta}{\operatorname{argmax}} \sum_{i=1}^N \log P(y_i | x_i; \theta)$$

양호 함수

$$= \underset{\theta \in \Theta}{\operatorname{argmin}} - \sum_{i=1}^N \log P(y_i | x_i; \theta)$$

※ θ : layer의 parameter
 $x_i \sim P(x)$
 $y_i \sim P(y | x = x_i)$
 $x, y \sim \theta$ (θ가 변할 때 결과도) → 과정

$$\theta \leftarrow \theta - \alpha \nabla_{\theta} \mathcal{L}(\theta) \leftarrow \text{gradient descent}$$

MLE를 신경망에서 구현하는 방법

신경망에 x_i 를 넣어서 y_i 에 대한 추정값을 구할 수 있다면,
 앞에서 구했던 Negative Log Likelihood 식을 cross entropy error 식으로 볼 수 있다.
 (softmax layer를 거쳤으며 참값은 one-hot encoding 되어 있다고 가정한다.)

$$D = \{(x_i, y_i)\}_{i=1}^N$$

$$\hat{\theta} = \underset{\theta \in \Theta}{\operatorname{argmin}} - \sum_{i=1}^N \log P(y_i | x_i; \theta)$$



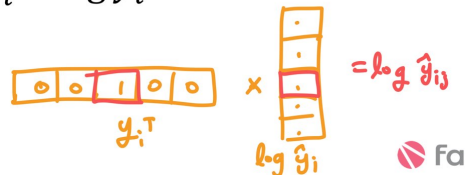
By implement (구현)

$$\hat{y}_i = f_{\theta}(x_i) \quad \Leftrightarrow \text{Neural network 라는 함수에 입력값 } x_i \text{를 넣어서 softmax layer의 결과값을 반환}$$

$$- \sum_{i=1}^N \log P(y_i | x_i; \theta) = - \sum_{i=1}^N y_i^T \cdot \log \hat{y}_i$$

위의 점곱이 어떻게 이루어지는지에 대해 그림으로 표현해 놓았다.

$$y_i^T \cdot \log \hat{y}_i$$



Cross Entropy error

최종적으로 classification에서 softmax layer를 사용해 손실함수를 Cross Entropy error를 사용하여 손실을 최소로 만드는 것은 MLE 관점에서 주어진 데이터를 가장 잘 나타내는 파라미터값들을 찾는 것으로 볼 수 있다.

$$\begin{aligned} \text{CE}(y_{1:N}, \hat{y}_{1:N}) &= -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^d y_{i,j} \log \hat{y}_{i,j} \\ &= -\frac{1}{N} \sum_{i=1}^N y_i^T \cdot \log \hat{y}_i \end{aligned}$$

where $y_{1:N} \in \mathbb{R}^{N \times d}$, $\hat{y}_{1:N} \in \mathbb{R}^{N \times d}$.

Handwritten notes and diagrams:
 - A diagram shows a matrix $\hat{y}_{1:N}$ with dimensions N (rows) and $d=|c|$ (columns).
 - An arrow points from the $\sum_{j=1}^d$ term in the equation to the matrix diagram.
 - A note above the second equation says "출력값의 개수" (Number of output values) pointing to d .
 - A note to the right of the second equation says "위 슬라이드와 '1/N' 차이가 있지만 gradient-descent에서 미분때문에 상수이므로 무시해도 좋아서 상관없음" (Although there is a difference of '1/N' from the previous slide, it is a constant and can be ignored in gradient descent, so it doesn't matter).
 - A small red circle with 'Fa' is also present.

