# Logistic Sigmoid는 어쩌다 Activation Function 으로 쓰였나?

Jung, Yucheol(ycjung@postech.ac.kr)

April 10, 2018

뉴럴 네트워크가 인기를 얻으면서 쏟아져 나오는 많은 텍스트들은 활성함수를 뉴런의 역치와 비교하여 설명한다. 어떤 값 이상이 되면 활성화 되고, 어떤 값이 이하라면 비활성화되는 메커니즘 말이다. 그러나 도대체 왜 뉴런 역치와는 전혀 다르게 생긴 함수인 Logistic sigmoid function이 함수가 쓰이는가? 그 이유는 Logistic sigmoid function이 Classification 문제에서 등장하는 과정을 살펴보면 쉽게 알 수 있다. 이 글은 많은 Classification에서 쓰이는 활성 함수 (Activation function) 들인 Logistic sigmoid function 이 유도되는 과정을 소개한다.

어떤 $\vec{x}$ 를 $C_1, C_2$ 둘 중 하나의 클래스로 나누는 문제를 생각해보자. 만약 우리가 어떤 주어진 클래스에서 $\vec{x}$ 가 생성될 확률 $p(\vec{x}|C_k)$ 와 주어진 클래스의 확률 분포 $p(C_k)$ 를 알 수 있다고 하자. 이 상황에서, $\vec{x}$의 클래스를 정하기 위해 그냥 $p(\vec{x}|C_1)$ 과 $p(\vec{x}|C_2)$를 비교하는 것은 충분하지 않다. 다음과 같은 예시를 생각해보자.

*당신은 친구들과 식사를 하기 위해 먹자골목에 있다. 이 먹자골목에는 좋은 신당동 떡볶이 가게와 나쁜 신당동 떡볶이 가게가 있다. 당신은 좋은 신당동 떡볶이 가게를 찾아내고자 한다. 친구들의 말을 들어보자니, 좋은 신당동 떡볶이 가게라면 간판 위에 사장의 사진을 붙여놓을 확률이 매우 높다고 한다. 그 말을 들은 당신은 눈 앞에 보이는 떡볶이 집 중에서 유일하게 간판에 사진이 붙어있는 집으로 들어간다. 그러나 당신과 친구들은 음식을 먹고 나서야 그 집이 나쁜 신당동 떡볶이 집이라는 것을 깨달을 수 있었다. 왜 그런지 궁금해하던 당신은 나중에 다른 친구에게 이 골목에 대해 물어봤다. 그랬더니 친구는 그 먹자골목에는 애초에 좋은 신당동 떡볶이 집이 하나도 없었다는 것을 알려주었다.*

위의 이야기와 같이, 각 클래스 별 확률 분포를 고려하지 않는다면 문제가 생긴다. 우리가 결국 해야 하는 것은 $p(\vec{x}|C_1)$ 으로부터 $p(C_1|\vec{x})$ 를 구하는 것이다.

$p(C_1|\vec{x})$ 는 베이즈 정리를 이용하여 다음과 같이 간단하게 구할 수 있다.

$$p(C_1|\vec{x}) = \frac{p(\vec{x}|C_1)p(C_1)}{p(\vec{x}|C_1)p(C_1) + p(\vec{x}|C_2)p(C_2)}$$

$p(C_1|\vec{x})$ 는 의미상으로 $\vec{x}$ 의 클래스를 결정짓는 증거로 쓰기에 적합하다. $p(C_1|\vec{x})$ 는 주어진 $\vec{x}$ 가 있을 때, 클래스가 $C_1$ 일 확률을 나타낸다. 이제 $p(C_1|\vec{x})$ 와 $p(C_2|\vec{x})$ 를 비교해서 $\vec{x}$ 가 어떤 클래스에 들어가는지 알 수 있다.

이제 이 과정이 Logistic sigmoid function 과 연관지어지는 과정을 살펴보자. $p(C_1|x)$ 를 구하는 과정에서 $a = \ln \frac{p(\vec{x}|C_1)p(C_1)}{p(\vec{x}|C_2)p(C_2)}$ 라고 해보자. 그러면 $p(C_1|x)$ 가 다음과 같이 변함을 알 수 있다.

$$p(C_1|\vec{x}) = \frac{p(\vec{x}|C_1)p(C_1)}{p(\vec{x}|C_1)p(C_1) + p(\vec{x}|C_2)p(C_2)} = \frac{1}{1 + \frac{p(\vec{x}|C_2)p(C_2)}{p(\vec{x}|C_1)p(C_1)}} = \frac{1}{1 + exp(-a)}$$

여기에서 우리는 $p(C_1|x)$ 가 $a$ 에 대한 Logistic sigmoid function임을 알 수 있다. 즉, Logistic sigmoid function 은 베이즈 정리를 이용하는 과정에서 자연스럽게 나오게 된다.

그러나 이것만으로는 자연스럽다고 하기 어렵다. $a$ 를 정한 꼴이 매우 인위적이기 때문이다. $p(\vec{x}|C_1)p(C_1)$ 와 $p(\vec{x}|C_2)p(C_2)$ 사이의 비율을 하나로 묶은 것은 그럴듯하다. 이 비율이 1 보다 작느냐 크느냐에 따라 클래스 판정이 나뉘기 때문이다. 그러나 왜 하필 ln을 이 비율에다가 붙였을까? 그 의문에 대한 핵심은 우리가 $p(\vec{x}|C_k)$ 조차 직접적으로 알기 힘들다는 사실에 있다. $p(\vec{x}|C_k)$ 를 정확하게 알기 힘들기 때문에, 사람들은 이 확률이 Gaussian 분포를 가진다고 가정하고 모델링한다. 즉, 다음과 같은 등식이 성립한다고 보는 것이다.

$$p(\vec{x}|C_k) = \mathcal{N}(\mu_k, \Sigma)$$

풀어서 쓰면 다음과 같다.

$$p(\vec{x}|C_k) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} exp\{-\frac{1}{2}(\vec{x} - \mu_k)^T \Sigma^{-1} (\vec{x} - \mu_k)\}$$

여기에서 $\Sigma$ 는 모든 클래스 별로 같다고 가정한다. $D$ 는 $\vec{x}$ 의 차원이다. 이제 이 식을 위에 있는 $p(C_1|\vec{x})$ 속의 $a$ 에 집어 넣어보자. 그러면 Gaussian 분포 식에 있는 $exp$ 가 ln 에 의해 자연스럽게 사라지고, 다음과 같은 표현이 가능해진다.

$$p(C_1|\vec{x}) = \sigma(W^T \vec{x} + w_0)$$

where

$$W = \Sigma^{-1}(\mu_1 - \mu_2)$$

$$w_0 = -\frac{1}{2}\mu_1^T \Sigma^{-1} \mu_1 + \frac{1}{2}\mu_2^T \Sigma^{-1} \mu_2 + \ln \frac{p(C_1)}{p(C_2)}$$

이로써 $p(C_1|\vec{x})$ 가 $\vec{x}$ 에 대한 Linear 한 함수를 Logistic sigmoid로 Activation 한 꼴이라는 것을 알 수 있다.

정리하자면, $p(\vec{x}|C_k)$ 가 각 클래스 별로 같은 Covariance matrix 를 가지는 Gaussian 분포를 따른다고 가정했을 경우, $p(C_k|\vec{x})$ 는 $\vec{x}$ 에 대한 linear function 에 logistic sigmoid 로 activation 한 꼴이 된다. 만약 클래스 별로 Covariance matrix가 다르다고 한다면, $\vec{x}$ 에 대한 quadratic function을 logistic sigmoid 로 activation 한 꼴이 될 것이다. 결론적으로 Activation function 은 우리가 뉴런을 모사하기 위해 억지로 끼워 맞춘 것이 아니라, 베이즈 정리를 이용한 Posterior probability 를 구하는 과정에서 자연스럽게 나오는 함수이다.

참고자료 : Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning. New York, NY. Springer Science+Business Media, LLC.