

인공 신경망의 원리와 성능 개선 방법

산업 및 에너지 부문

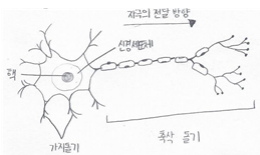
염경중학교 2학년 강준서
염경중학교 2학년 이유진

1 탐구의 동기

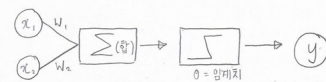


지난 2016년, DeepMind의 딥러닝 기반 바둑 인공지능 Alpha Go와 이 세돌 9단의 대국은 수많은 사람들을, 정치가들을, 기업가들을 인공지능이라는 가능성에 희망을 품게 만들었다. 의 명대사에 비유하자면, 세상은, 대 인공지능 시대를 맞았다. '4차 산업 혁명'이라는 키워드만 들어가도 서점은 대대적으로 홍보해주고, 기업들은 AI 관련 상품들을 마구 내 놓기 시작했다. 우리는 단순히 인공지능의 사회적 파급효과만이 아닌 그 원리와 역사, 그리고 나아가 어떻게 하면 그 성능을 향상시킬 수 있을지에 의문점을 가졌고, 탐구해보고자 하였다.

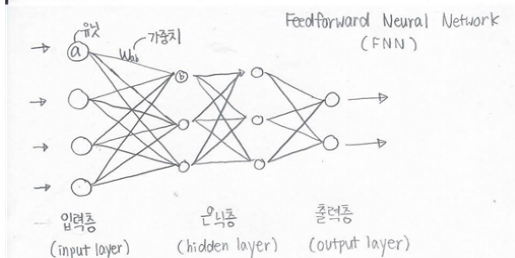
2 인공 신경망



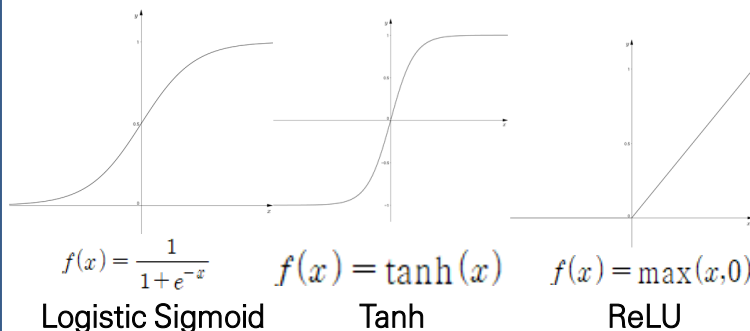
TLU: Threshold Logic Unit



인공 신경망은 사람의 뇌를 모방하여, 전산상으로 구현된 뉴런을 연결하고, 어떠한 문제에 대해 학습시켜 해결하게 한다. 이는 뇌가 시냅스의 연결을 조절하는 과정을 모방하였다.



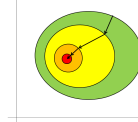
신경망은 각각 입력된 신호에 가중치(신호의 세기를 조절, 시냅스의 역할)를 곱하고, 활성화 함수를 적용하여 다음 계층으로 신호를 발산한다 : **순전파**



다음 계층으로 신호를 발산하기 전에, 가중치와 입력 신호의 곱에 **활성화 함수**를 적용하게 되는데, 이는 축삭돌기가 일정 세기 이상의 신호만을 발산하는 모습을 본 따왔다.

3 인공 신경망의 학습

인공 신경망을 실제 문제 해결에 사용하려면, 가중치를 조절하여 문제 해결에 알맞게 '학습' 시켜야 한다. 학습은 주어진 데이터 세트를 이용해서 수행하는데, 편미분을 통해 주어진 입력에 대한 신경망의 출력에 주어진 기대 출력 값과 가까워지게끔 가중치를 조절한다. 이를 **경사 감소법**이라고 한다($E(u)$ 를 최소화)



$$c = g^T + g^T \text{의 때, 방향은 이해할 수 없다.}$$
$$\Delta c = \frac{\partial c}{\partial \theta} \Delta \theta + \frac{\partial c}{\partial \phi} \Delta \phi \quad (1) \text{ 또한 이 식은 이해할 수 없다.}$$
$$\Delta c = \left(\frac{\partial c}{\partial \theta} \frac{\partial c}{\partial \phi} \right)^T (\Delta \theta, \Delta \phi)^T \quad (2)$$
$$\nabla c = \left(\frac{\partial c}{\partial \theta}, \frac{\partial c}{\partial \phi} \right)^T, \Delta c = (\Delta \theta, \Delta \phi) \text{이라고 하면}$$
$$\Delta c = \nabla c \cdot \Delta u \text{이 된다.}$$

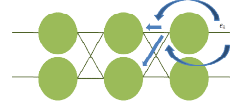
우리가 구해야 할 것은 Δc 변화량을 얼마나 줄일 수 있는 지 변화량이다. 이는 이해할 수 없다.

$$\Delta c = -\alpha \nabla c \quad (3)$$
$$\therefore \Delta c = -\alpha \nabla c \text{이면 } \Delta c < 0$$

경사 감소 법을 이용해 신경망의 오차를 최소화한다. 출력 층의 오차함수는 아래와 같이 구한다(최소제곱 오차)

$$E(u) = (t_u - o_u)^2 \dots u \text{층의 오차}$$

출력 층의 오차함수를 통해 구한 오차를 이용해 가중치를 보정하고, 이 오차를 이용해 이전 계층의 오차를 계산한다. 이전 계층으로 분배되는 오차는 그 유닛에 연결된 가중치에 비례한다 : **역전파**



$$\frac{w_{ij}}{\sum_{i=0}^M w_{ij}} * e_j$$

오차함수를 위의 가중치 보정 식에 대입하면 아래와 같다(로지스틱 시그모이드)

$$\Delta w_{jk} = \alpha \times (e_k * \text{sigm}(o_k)(1 - \text{sigm}(o_k))) \cdot o_j^T$$

위와 같은 과정을 통해 신경망을 학습시킬 수 있었다. 학습된 신경망은 학습 데이터와 같은 유형의 문제를 해결할 수 있다(분류, 회귀 등)

4 인공 신경망의 성능 개선

0) MNIST Dataset

신경망의 성능을 개선하고, 개선된 성능을 평가하기 위해, 신경망을 이용한 이미지 인식 분야의 선구자인 Yann Lecun 교수님의 공공 신경망 학습용 데이터세트인 MNIST 데이터세트를 이용하여 손 글씨를 분류하게끔 학습시키고, 그 성능을 평가 및 비교한다.

1) MiniBatch

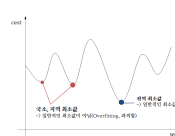


$$E_1(w) = \frac{1}{N_1} \sum_{i=1}^{N_1} E_i(w)$$

미니배치는 데이터세트의 레코드 하나만을 이용하여 가중치를 보정하는 확률적 경사감소법의 속도를 개선하는 방법이다. 데이터세트를 N_t 개의 원소를 갖는 Minibatch 라 불리는 집합으로 나누고, 이 집합마다 오차를 구하여 가중치를 보정하는데, N_t 의 값에 비례하여 속도가 빨라진다.

2) 가중치 감쇠 : Weight Decay

$$B_{\text{weight}}(w) = B(w) + \frac{\gamma}{2} \|w\|^2$$



학습 데이터 양	60,000
주기(epoch)	5회
학습률	0.1
미니배치 수	10
학습 시간	약 6분
정확도	68.69%

과적합은 오차함수를 최적화하는 과정에서 지역 최소 값에서 학습이 끝나버린 경우에 발생하는데, 학습 데이터에 대한 오차는 작지만, 테스트 데이터에 대한 오차는 크게 발생한다. 이는 가중치가 너무 크게 될 때 발생하는데, 이를 억제하기 위해 가중치 감쇠의 방법을 적용한다. 가중치 감쇠는 가중치가 지나치게 커지는 현상을 방지하는데, 가중치의 제곱 합과 감쇠상수의 곱을 오차함수에 더하여 가중치의 크기에 비례하게 작아지도록 한다. 감쇠상수는 주로 10^{-8} 에서 10^{-2} 사이의 값을 사용한다. 벤치마크 결과를 해석하면, 연산량은 늘었지만 전역 최소값에 더 가까이 도달하여 정확도가 올라갔음을 발견할 수 있다.