

## [4장. 딥러닝 시작]

### 4.1. 인공 신경망의 한계와 딥러닝 출현

### 4.2. 딥러닝 구조

### 4.3. 딥러닝 알고리즘

### 4.4. 우리는 무엇을 배워야 할까?

### 4.1. 인공 신경망의 한계와 딥러닝 출현

- 오늘날 인공 신경망에서 이용하는 구조는 프랭크 로젠블라트가 고안한 퍼셉트론이라는 선형 분류기 (딥러닝의 기원이 되는 알고리즘)
- 흐른다 1 / 안흐른다 0

- 1) And 게이트는 모든 입력이 1일 때 작동
- 2) OR 게이트는 둘 중 하나라도 1이면 작동
- 3) XOR 게이트는 배타적 논리합 (하나만 1이어야 함) → 비선형적 데이터, 분류 어렵
  - 해결책: 입력층과 출력층 사이에 하나 이상의 중간층을 두어 다층 퍼셉트론 고안
  - 딥러닝 = 심층신경망 : 입력층과 출력층 사이에 은닉층이 여러 개 있는 신경망

### 4.2. 딥러닝 구조

#### 1) 용어

구분	구성 요소	설명
층	입력층(input layer)	데이터를 받아들이는 층
	은닉층(hidden layer)	모든 입력 노드부터 입력 값을 받아 가중합을 계산하고, 이 값을 활성화 함수에 적용하여 출력층에 전달하는 층
	출력층(output layer)	신경망의 최종 결과값이 포함된 층

구분	구성 요소	설명
가중치(weight)		노드와 노드 간 연결 강도
바이어스(bias)		가중합에 더해 주는 상수로, 하나의 뉴런에서 활성화 함수를 거쳐 최종적으로 출력되는 값을 조절하는 역할을 함
가중합(weighted sum), 전달 함수		가중치와 신호의 곱을 합한 것
함수	활성화 함수(activation function)	신호를 입력받아 이를 적절히 처리하여 출력해 주는 함수
	손실 함수(loss function)	가중치 학습을 위해 출력 함수의 결과와 실제 값 간의 오차를 측정하는 함수

- (1) 가중치: 입력 값이 연산 결과에 미치는 영향력을 조절하는 요소  
(2) 가중합=전달 함수: 각 노드에서 들어오는 신호에 가중치를 곱해서 다음 노드로 전달, 이 값들 모두 더한 합계를 의미

$$\sum_i w_i x_i + b$$

( $w$ : 가중치,  $b$ : 바이어스)

- (3) 활성화 함수: 전달 함수에서 전달받은 값을 출력할 때 일정 기준에 따라 출력 값을 변화시키는 비선형 함수

Ex) 시그모이드, 하이퍼볼릭 탄젠트, 렐루 함수 등

① 시그모이드 함수

선형 함수의 결과를 0~1 사이에서 비선형 형태로 변형해줌

주로 로지스틱 회귀 같은 분류 문제를 확률적으로 표현하는데 use

기울기 소멸 문제 발생

$$f(x) = \frac{1}{1 + e^{-x}}$$

② 하이퍼볼릭 탄젠트 함수

선형 함수의 결과를 -1~1 사이에서 비선형 형태로 변형해줌

시그모이드에서 결괏값의 평균이 0이 아닌 양수로 편향된 문제 해결하는데 사용

기울기 소멸 문제 여전

③ 렐루 함수

최근 활발히 사용

입력이 음수면 0, 양수면  $x$  출력

경사 하강법에 영향 노노 → 학습 속도 빠름 → 기울기 소멸 문제 발생 X  
일반적으로 은닉층에서 사용

하이퍼볼릭 탄젠트 함수 대비 학습 속도 6배 빠름

음수값 입력받으면 항상 0 출력 → 학습 능력 감소

해결책: 리키 렐루

④ 소프트맥스 함수

입력값을 0~1 사이에 출력되도록 정규화

총합은 항상 1

보통 딥러닝에서 출력 노드의 활성화 함수로 많이 사용

$$y_k = \frac{\exp(a_k)}{\sum_{i=1}^n \exp(a_i)}$$

(4) 손실 함수

학습을 통해 얻은 데이터의 추정치가 실제 데이터와 얼마나 차이가 나는지 평가하는 지표

값이 틀수록 많이 틀린 것임

① 평균 제곱 오차

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$\hat{y}_i$ : 신경망의 출력(신경망이 추정한 값)  
 $y_i$ : 정답 레이블  
 $i$ : 데이터의 차원 개수

② 크로스 엔트로피 오차

분류 문제에서 원-핫 인코딩 했을 때만 사용 가능한 오차 계산법

$$CrossEntropy = - \sum_{i=1}^n y_i \log \hat{y}_i$$

$\hat{y}_i$ : 신경망의 출력(신경망이 추정한 값)  
 $y_i$ : 정답 레이블  
 $i$ : 데이터의 차원 개수

2) 딥러닝 학습

Step 1: 순전파(네트워크에 훈련 데이터가 들어올 때 발생, 데이터 기반 예측 값 계산하기 위해 은닉층으로 전송

Step2: 손실 함수로 네트워크의 예측 값과 실제 값의 차이(손실, 오차) 추정

손실 함수 비용은 0이 이상적

손실이 계산되면 그 정보는 역으로 전파되므로 역전파라 부름

은닉층의 뉴런은 각 뉴런이 원래 출력에 기여한 상대적 기여도에 따라 값이 달라짐

예측 값과 실제 값 차이를 각 뉴런의 가중치로 미분 후 기존 가중치 값에서 뺌

모든 뉴런에 동일하게 진행

#### 4) 딥러닝의 문제점과 해결방안

포인트는 여러 은닉층을 결합해서 비선형 영역을 표현하는 것

- 은닉층이 많을 때 문제점

과적합(훈련 데이터에 대해 과하게 학습하여 실제 데이터에 대한 오차 증가하는 현상) → 해결책: 드롭아웃

기울기 소멸 문제(출력층에서 은닉층으로 전달되는 오차가 크게 줄어들어 학습이 되지 않는 현상) → 렐루 활성화 함수 사용

성능이 나빠지는 문제 → 확률적 경사 하강법과 미니 배치 경사 하강법 사용  
배치 경사 하강법: 전체 데이터셋에 대한 오류 구한 후 기울기를 한번만 계산해서 모델의 파라미터 업데이트

확률적 경사 하강법: 임의로 선택한 데이터에 대해 기울기를 계산하는 방법으로 적은 데이터 사용해서 속도 높임

미니 배치 경사 하강법: 전체 데이터셋을 미니 배치 여러 개로 나눈 후 한 개 한개 기울기 구하고 그것의 평균 기울기를 이용하여 모델 업데이트에서 학습하는 방법 (다소 안정적)

#### 5) 딥러닝의 이점

특성 추출, 빅데이터의 효율적 활용

### 3. 딥러닝 알고리즘

CNN, RNN, RBM, DBN 으로 분류됨

#### 1) 심층 신경망

입력층과 출력층 사이에 다수의 은닉층을 포함하는 인공 신경망

연산량 많음, 기울기 소멸 문제 등 → 드롭아웃, 렐루 함수, 배치 정규화 등 적용해야 함

#### 2) 합성곱 신경망

합성곱층, 풀링층을 포함하는 이미지 처리 성능이 좋은 인공 신경망 알고리즘

영상 및 사진이 포함된 이미지 데이터에서 객체 탐색 or 객체 위치 찾아내기 유용

#### 3) 순환 신경망

시계열 데이터 같은 시간 흐름에 따라 변화하는 데이터를 학습하기 위한 인공 신경망

순환: 자기 자신을 참조한다

기울기 소멸 문제 → LSTM 사용

4) 제한된 볼츠만 머신

가시층, 은닉층으로 구성됨 둘이만 연결됨

DBN 의 요소로 많이 use

5) 심층 신뢰 신경망

제한된 볼츠만 머신을 블록처럼 여러 층으로 쌓아 올린 구조

레이블이 없는 데이터에 대한 비지도 학습이 가능