

Deep Learning Overview

Artificial Neural Network

Deep Learning

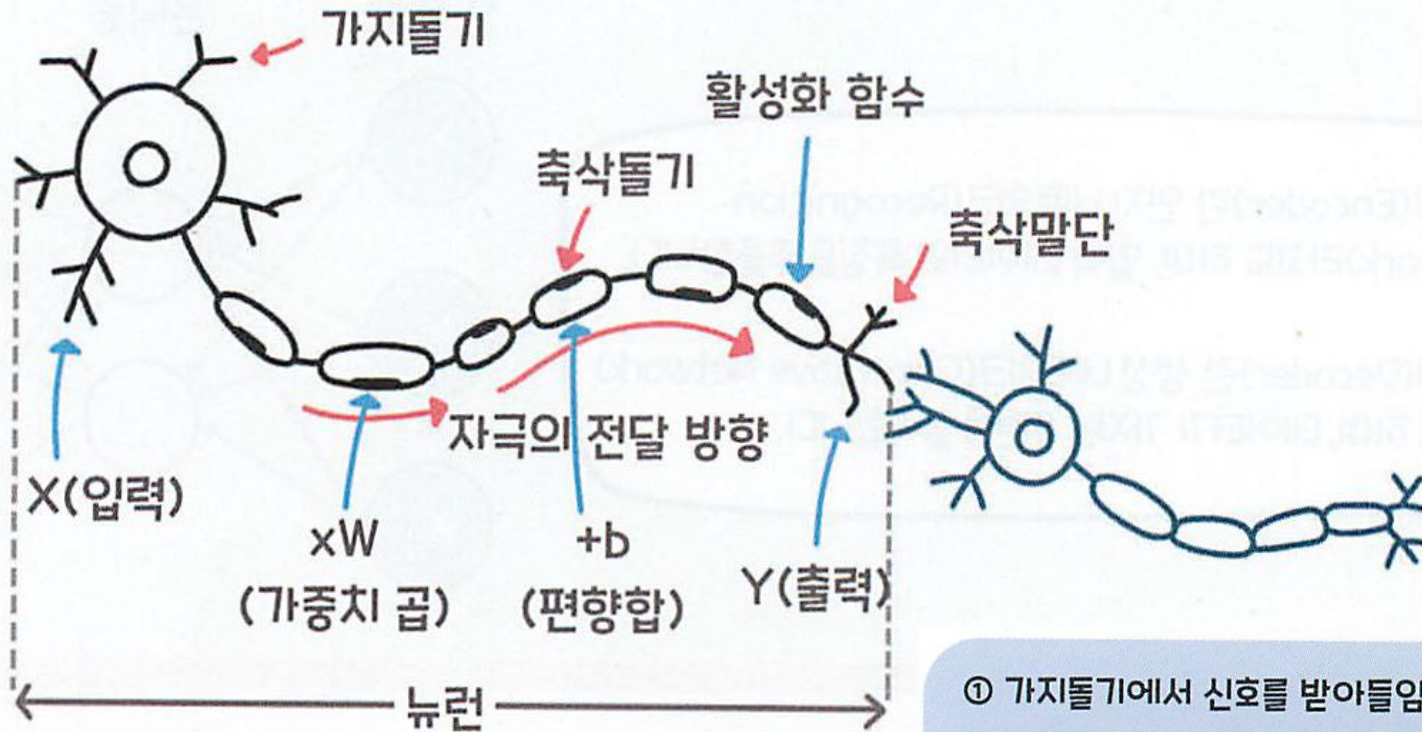


목차

01	인공신경망의 개념
02	딥러닝의 개념
03	딥러닝의 용어 및 학습
04	워드 임베딩
05	GAN

인공신경망의 개념

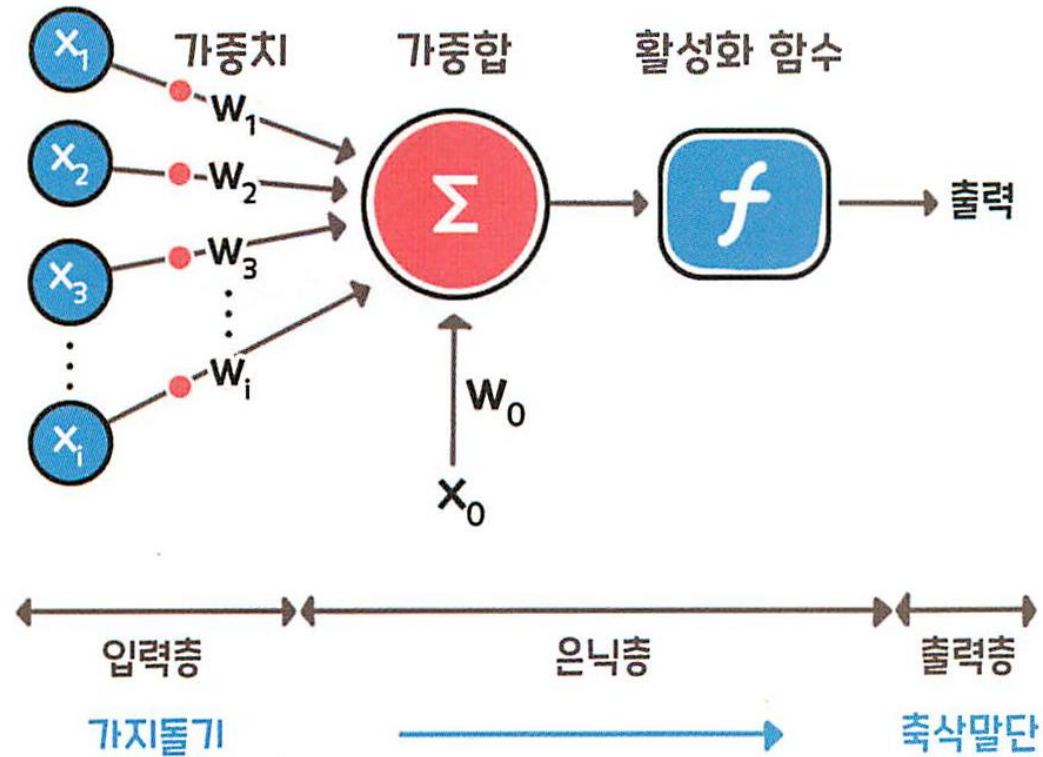
- 사람의 신경망 구조에서 착안함.





- ① 가지돌기에서 신호를 받아들임
- ② 신호가 축삭돌기를 지나 축삭말단으로 전달됨
- ③ 축삭돌기를 지나는 동안 신호가 약해져서 축삭말단까지 전달되지 않거나 강하게 전달되기도 함
- ④ 축삭말단까지 전달된 신호는 다음 뉴런의 가지돌기로 전달됨
- ⑤ 수억 개의 뉴런 조합을 통해 손가락을 움직이거나 물체를 판별하는 등 다양한 조작과 판단 수행 가능

인공신경망의 개념

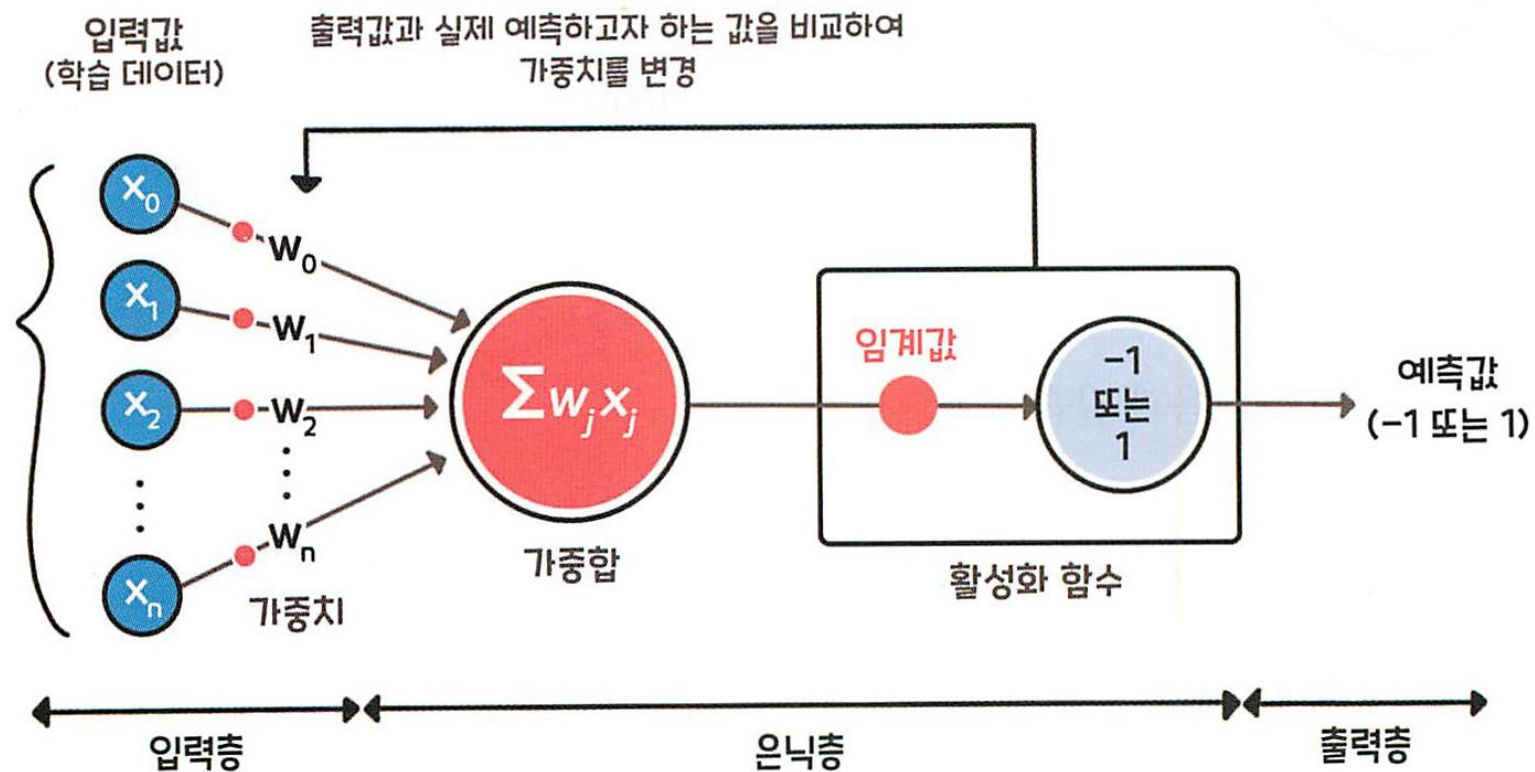
- 축삭말단에 이르기까지 신호의 크기는 인공신경망에서 가중치에 해당



인공신경망	인간의 뇌
	뉴런
	전기·화학적 신호가 흐르는 방향
가중치	신호의 크기

딥러닝의 개념

- 여러 층(특히 은닉층이 여러 개)을 가진 인공신경망을 사용하여 머신러닝 학습을 수행하는 것
- 심층학습이라고도 함

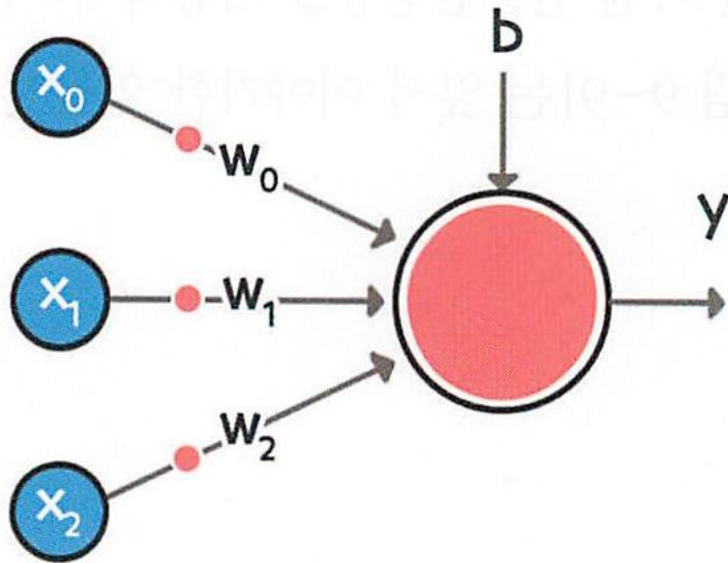


딥러닝 용어

구분	설명
입력층	학습하고자 하는 데이터를 입력받음
은닉층	모든 입력 노드로부터 입력값을 받아 가중합을 계산하고, 이 값을 활성화 함수에 적용하여 출력층에 전달
출력층	최종 결과 출력
가중치	입력 신호가 출력에 미치는 영향을 조절하는 매개변수로, 입력값의 중요도를 결정
편향	가중합에 더하는 상수로 하나의 뉴런에서 활성화 함수를 거쳐 최종적으로 출력되는 값을 조절

가중합

- 입력값과 가중치를 곱한 뒤 편향(Bias)을 더한 값

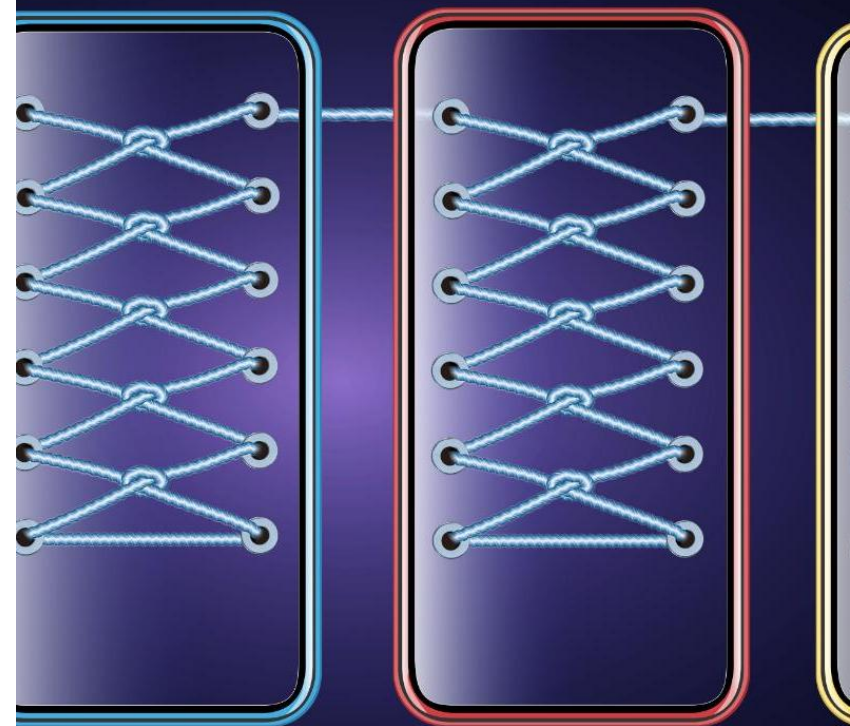


- 원 : 노드
- 화살표 : 신호의 흐름
- x_n : 외부에서 들어오는 신호(Input)
- w_n : 신호를 연결하는 가중치(Weight)
- b : 출력값을 조절하는 편향(Bias)
- y : 외부로 나가는 신호(Output)

$$\bullet \text{ 가중합}(y) = (x_0 \times w_0) + (x_1 \times w_1) + (x_2 \times w_2) + b$$

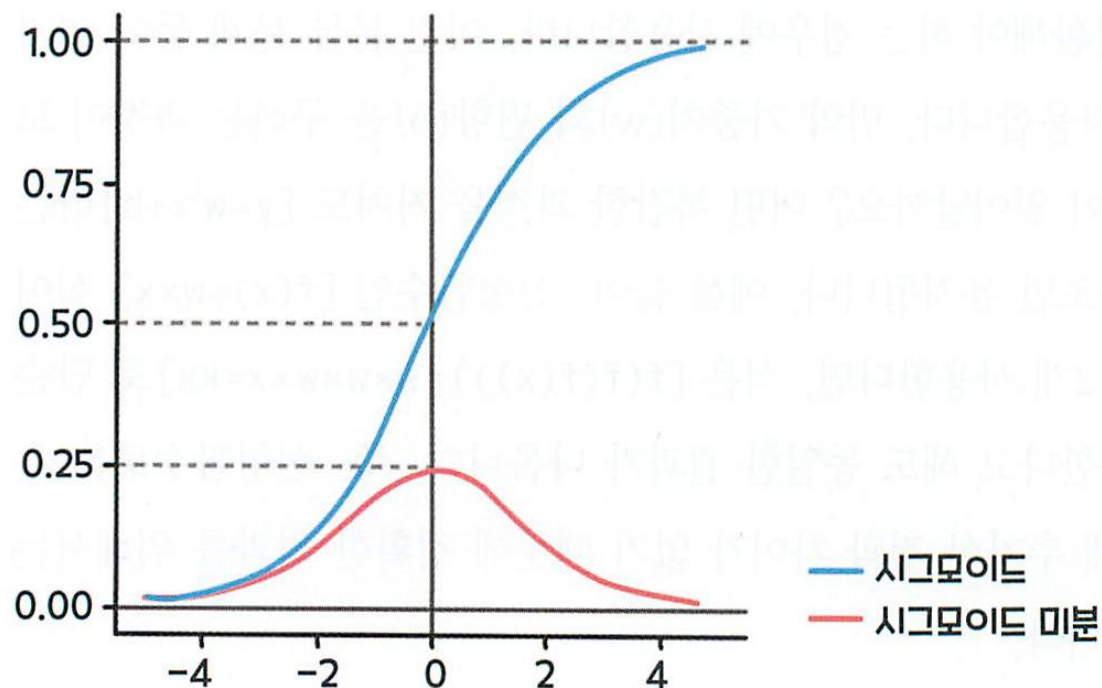
활성화 함수

- 인공신경망에서 각 뉴런이 입력을 받아 출력을 계산하는 함수
 - 출력값을 0~1 사이의 값으로 반환해야 하는 경우에 사용 (이진 분류 문제)
 - 비선형을 위해 사용 (선형함수 만으로는 복잡한 문제를 해결하기 어려움)
- 활성화 함수의 유형으로는
 - 시그모이드
 - 하이퍼볼릭 탄젠트
 - ReLU
 - LeakyReLU
 - SoftMax 등



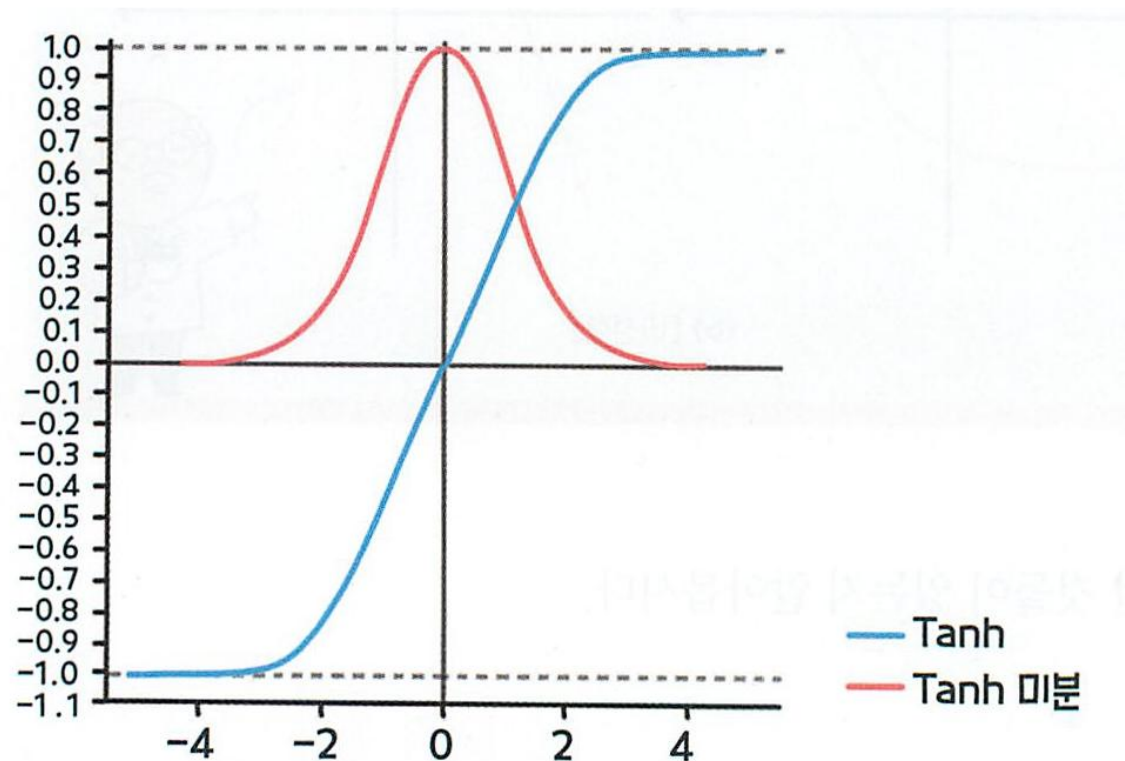
시그모이드 함수

- X값의 변화에 따라 0에서 1까지의 값을 출력하는 S자형 함수 (로지스틱 함수로도 불림)
- 이진 분류 문제에서 많이 사용
- 과거에는 인기가 많은 함수 였지만 딥러닝 모델의 깊이가 깊어질수록 기울기가 사라지는 기울기 소멸 문제가 발생함으로 인하여 현재는 추천하지 않음.



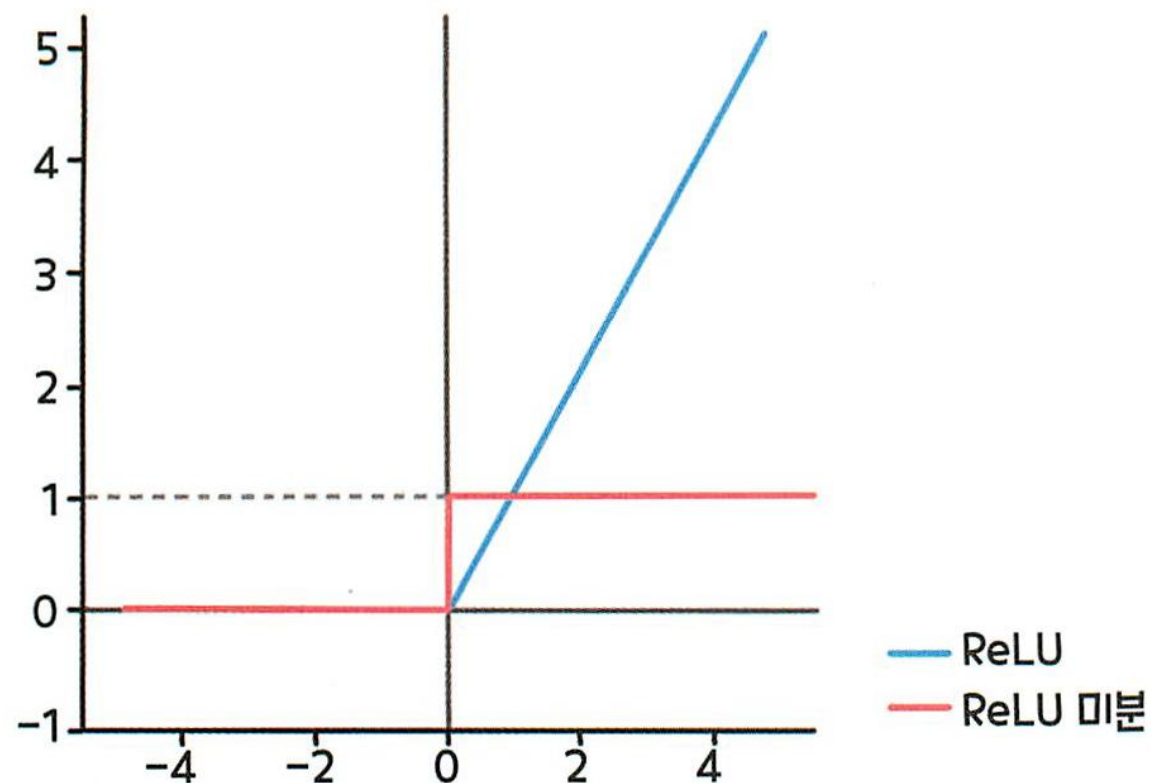
하이퍼볼릭 탄젠트 함수

- 시그모이드 함수와 유사하지만 $-1 \sim 1$ 의 값을 가지면서 데이터의 평균이 0 이라는 점이 다름.
- 거의 모든 측면에서 시그모이드 함수보다 성능이 우수
- 기울기 소멸 문제는 여전히 존재하여 사용시 주의해야 함.



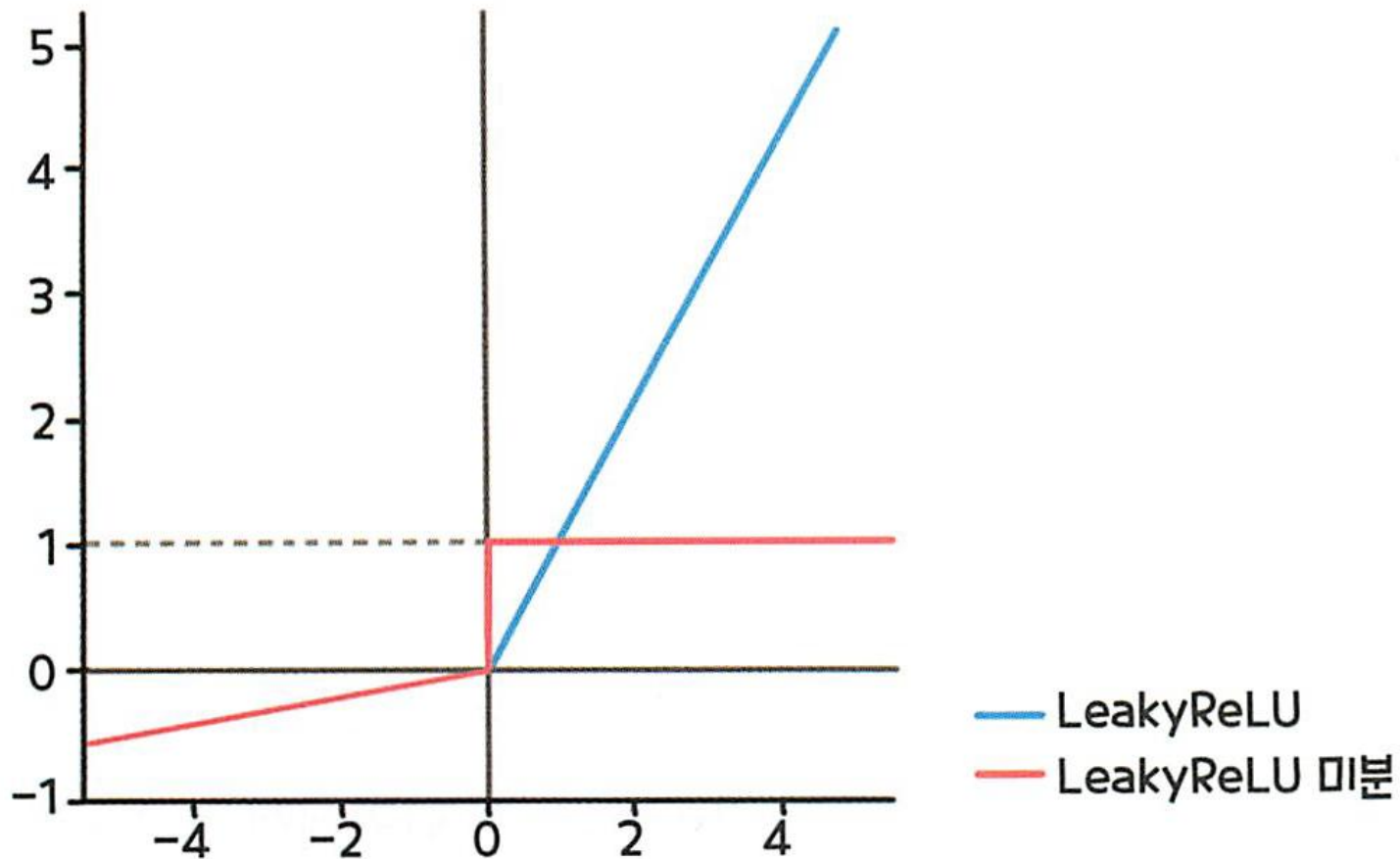
ReLU 함수

- $X < 0 \rightarrow 0$ 을 출력
- $X > 0 \rightarrow X$ 를 출력
- 학습 속도가 매우 빠르며 기울기 소멸 문제도 해결
- 다만, $X < 0$ 이면 기울기는 0 (해당 뉴런은 사용불가)



LeakyReLU 함수

- ReLU와 거의 유사하지만 가중치 곱의 합이 0 보다 작을 때의 값도 고려함.
- ReLU 함수의 단점을 보완



Softmax 함수

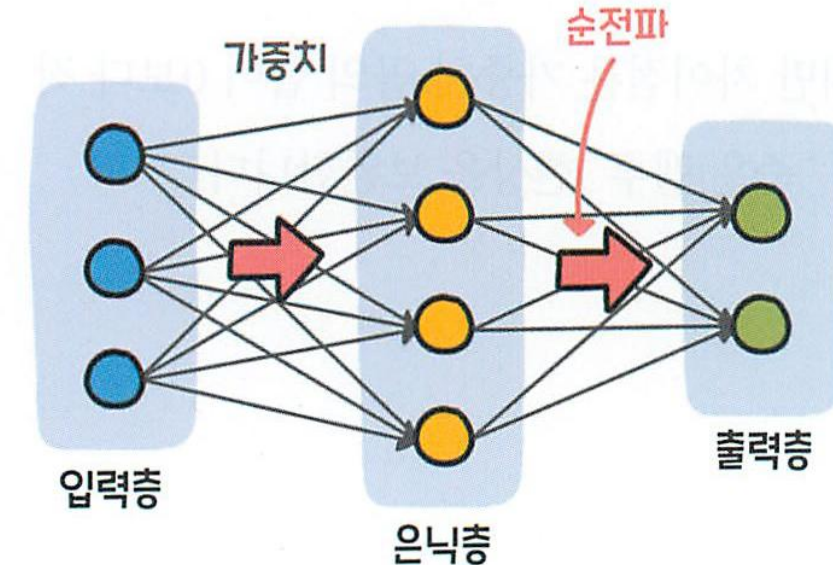
- 입력받은 값이 0 ~ 1 사이의 값으로 출력되도록 정규화하여 출력의 총합이 1이 되는 특성을 가지고 있음.
- 사용이 많이 되는 함수

정규화(Normalization)는 데이터의 범위를 사용자가 원하는 범위로 제한하는 것입니다. 예를 들어, 이미지 데이터의 경우 0~255 사이의 값을 갖는데, 이를 255로 나누면 0~1.0 사이의 값을 갖게 됩니다.



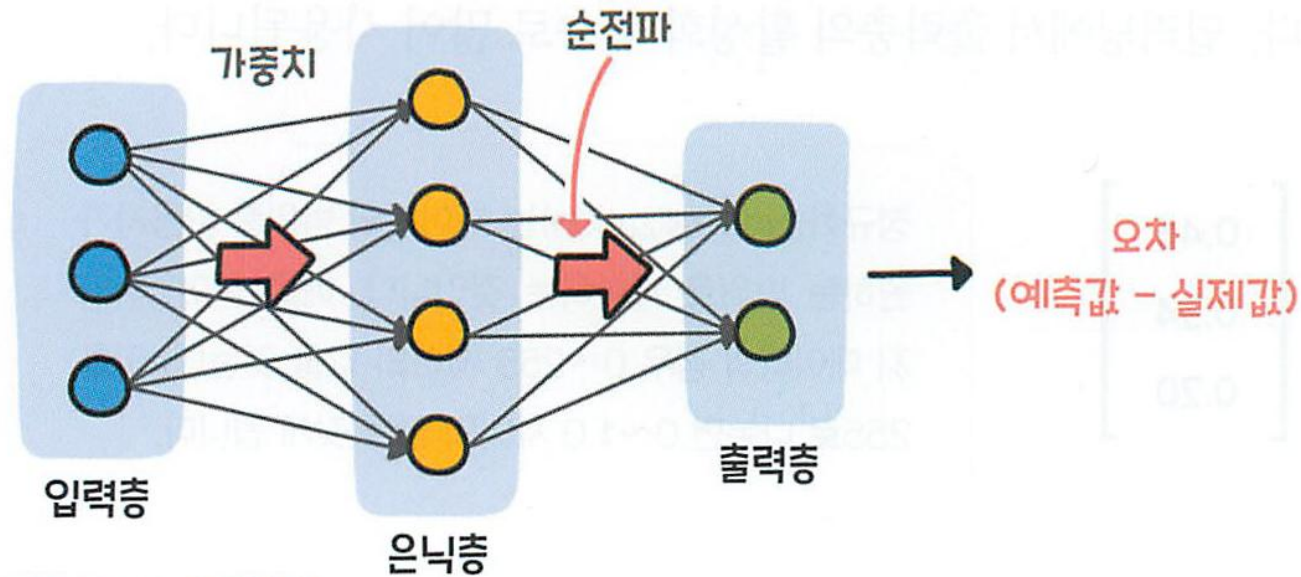
딥러닝 학습 (Forward Propagation)

- 순전파(Forward Propagation), 손실함수(Loss Function), 옵티마이저(Optimizer), 역전파(BackPropagation) 과정을 거치면서 학습
- 네트워크를 구성하기 위해서는 하이퍼파라미터를 결정해야 함.
- 네트워크가 구성된 이후에는 입력값이 은닉층을 거치면서 출력층으로 결과가 향함. (예측값 출력)
- 이 방식은 순전파라고 함.



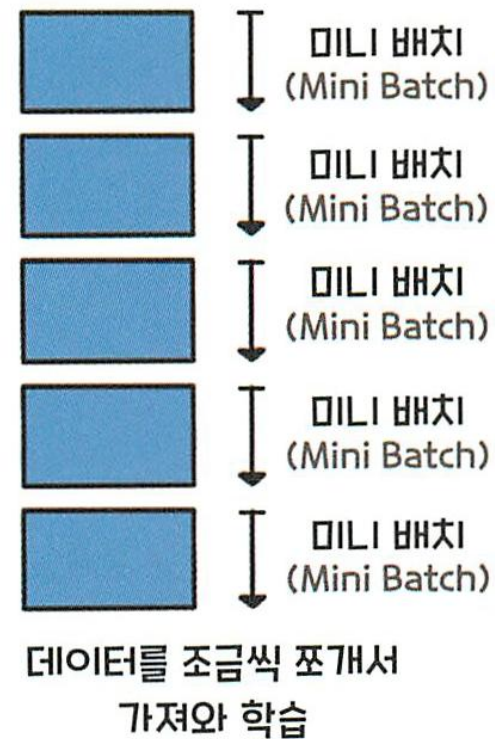
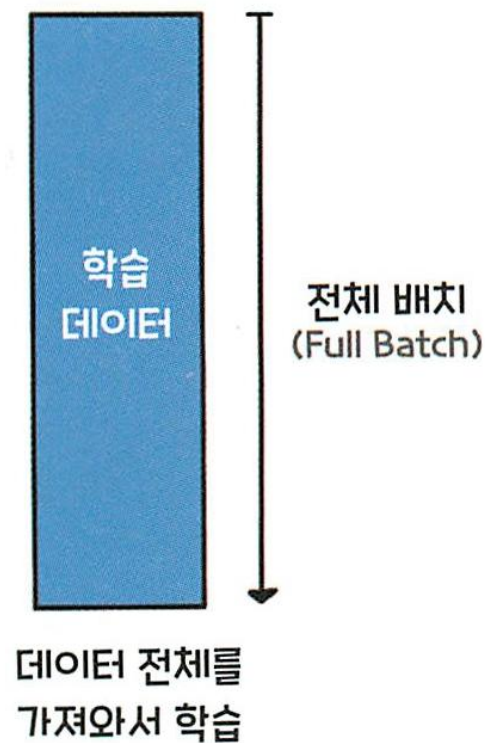
딥러닝 학습 (Loss Function)

- 예측값과 실제값의 차이를 구하는 함수.
- 회귀 문제에서는 평균제곱오차(MSE, Mean Squared Error)
- 분류 문제에서는 Cross Entropy를 손실함수로 사용



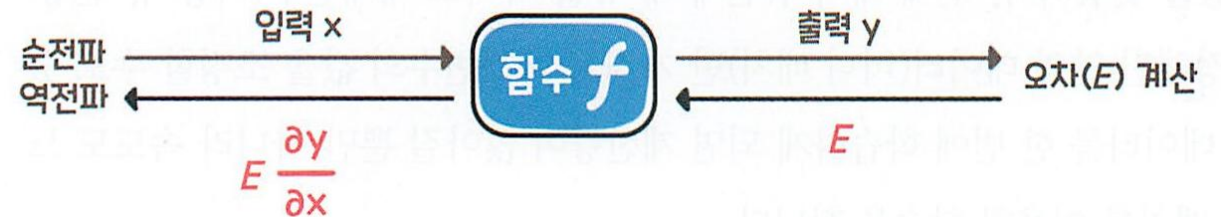
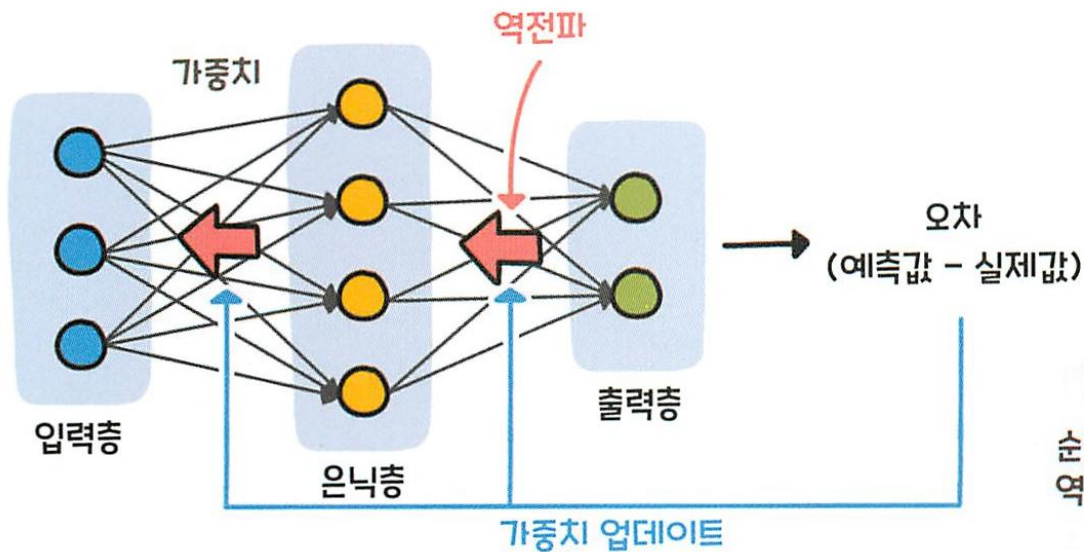
딥러닝 학습 (Optimizer)

- 옵티마이저란 학습속도를 빠르고 안정적이게 만드는 것을 뜻함.
- 배치(Batch)란 가중치 등 매개변수의 값을 조정하기 위해 사용하는 데이터의 양
- 전체 데이터를 Full Batch / 정해진 양의 데이터를 Mini Batch 라고 함.



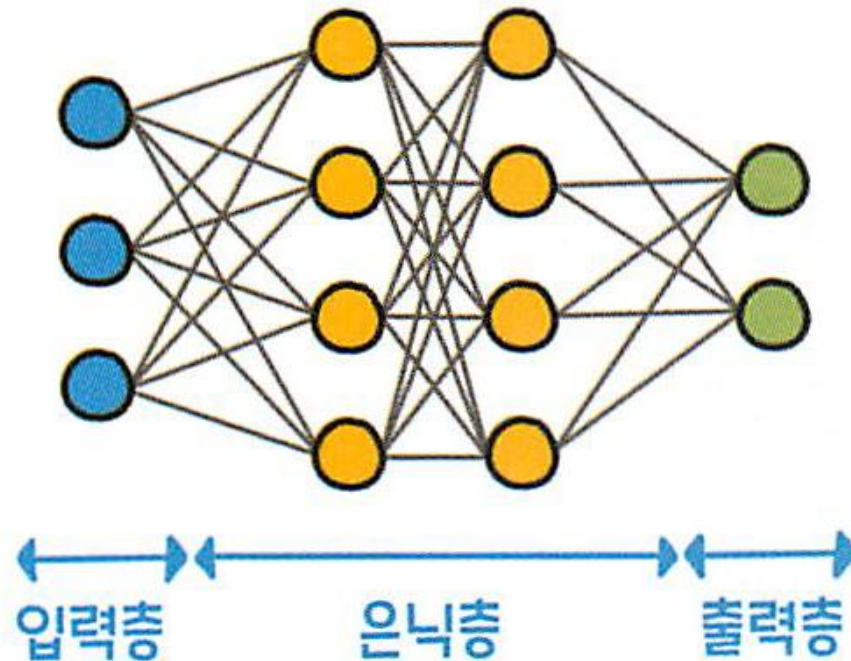
딥러닝 학습 (Backpropagation)

- 오차를 역방향으로 전파 시켜 가중치를 업데이트 하는 것.
- 경사 하강법을 이용해 오차를 줄이는 방향으로 가중치를 수정함.
- 가중치를 수정할 때는 순전파에서 계산한 결과의 편미분값을 오차에 곱해 출력층 -> 은닉층 -> 입력층 순으로 전달
- 이렇게 진행하는 이유는 계산하려는 노드와 연결된 가중치값만을 고려하기 위함.



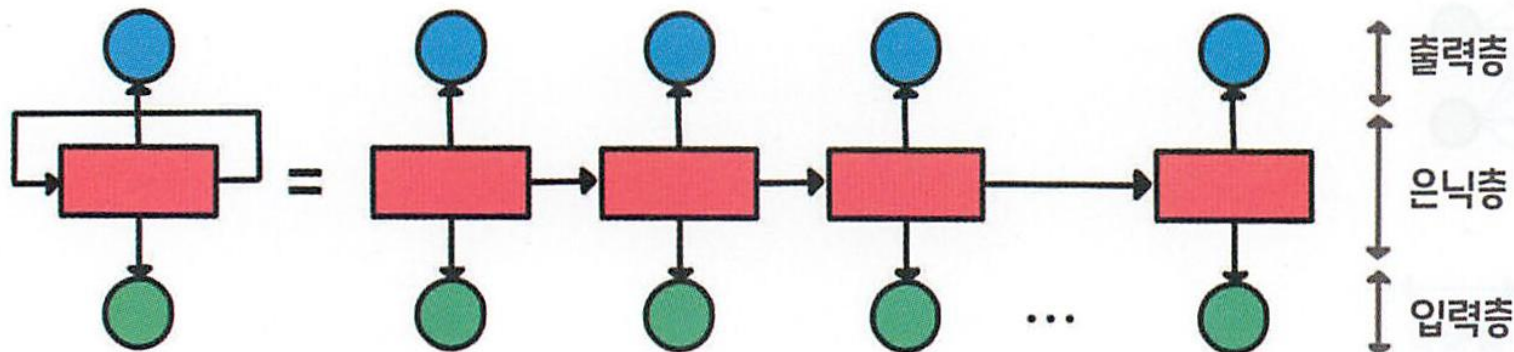
딥러닝의 유형 (심층 순방향 신경망 / DFN)

- 입력층, 은닉층(2개 이상), 출력층으로 구성된 가장 기본적인 딥러닝 모델
- "순방향(Feedforward)"이라는 이름은 데이터가 한 방향으로만 흐르기 때문
- DFN은 입력 데이터의 시간 순서나 종속성을 반영하지 못하는 단점이 있음.



딥러닝의 유형 (순환 신경망 / Recurrent Neural Network)

- 시간적인 연속성이 있는 시계열 데이터를 처리하기 위해 고안된 신경망이다.
- 시계열 데이터란 일정 시간 동안 연속적으로 수집된 데이터를 의미한다. (예: 주식 데이터, 센서 데이터 등.)
- 시계열 데이터는 과거 데이터가 현재 데이터에 영향을 미치므로, 단순히 순차적으로만 처리하는 DFN으로는 정확한 예측이 어렵다.
- RNN은 은닉층 뉴런에 **순환 구조(recurrent connection)**를 추가하여 과거 입력 정보를 현재 예측에 활용 가능
- RNN은 과거 데이터를 고려하여 더 정밀한 예측이 가능하다.
- 그러나, 신경망 층이 깊어 질 수록 먼 과거의 데이터가 현재에 영향을 미치지 못하는 문제 발생 (장기 의존성 문제)



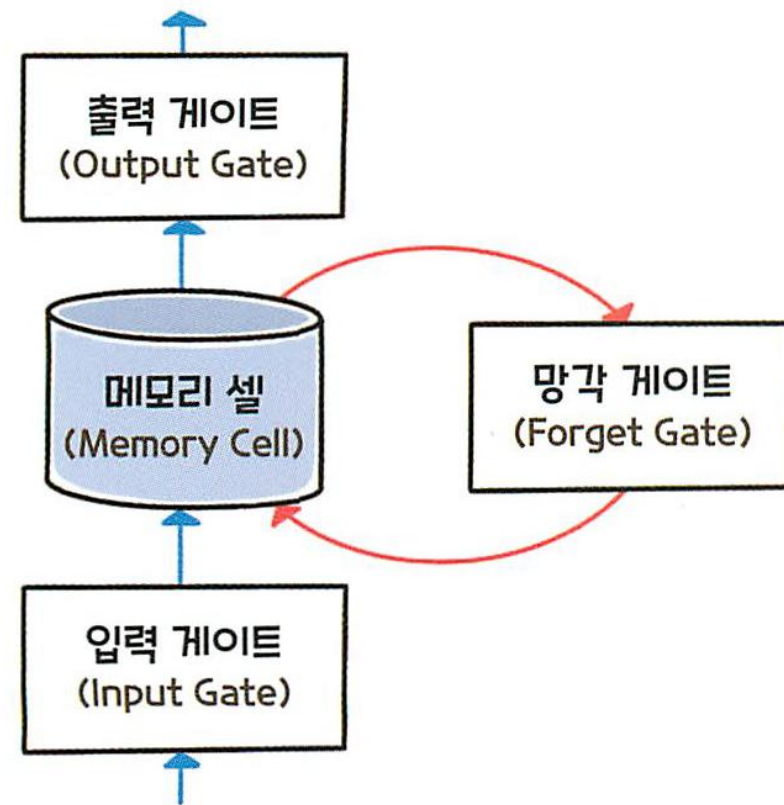
딥러닝의 유형 (Long Short-Term Memory)

- 신경망 내에 메모리를 두어 먼 과거의 데이터도 저장할 수 있도록 고안
- 입출력을 제어하기 위한 논리적인 장치를 추가 (Gate)
- Gate의 역할은 다음과 같음

입력 게이트: 현재 정보를 얼마나 저장할지 결정
→ 시그모이드 + tanh를 통해 현재 정보의 보존량을 조절.

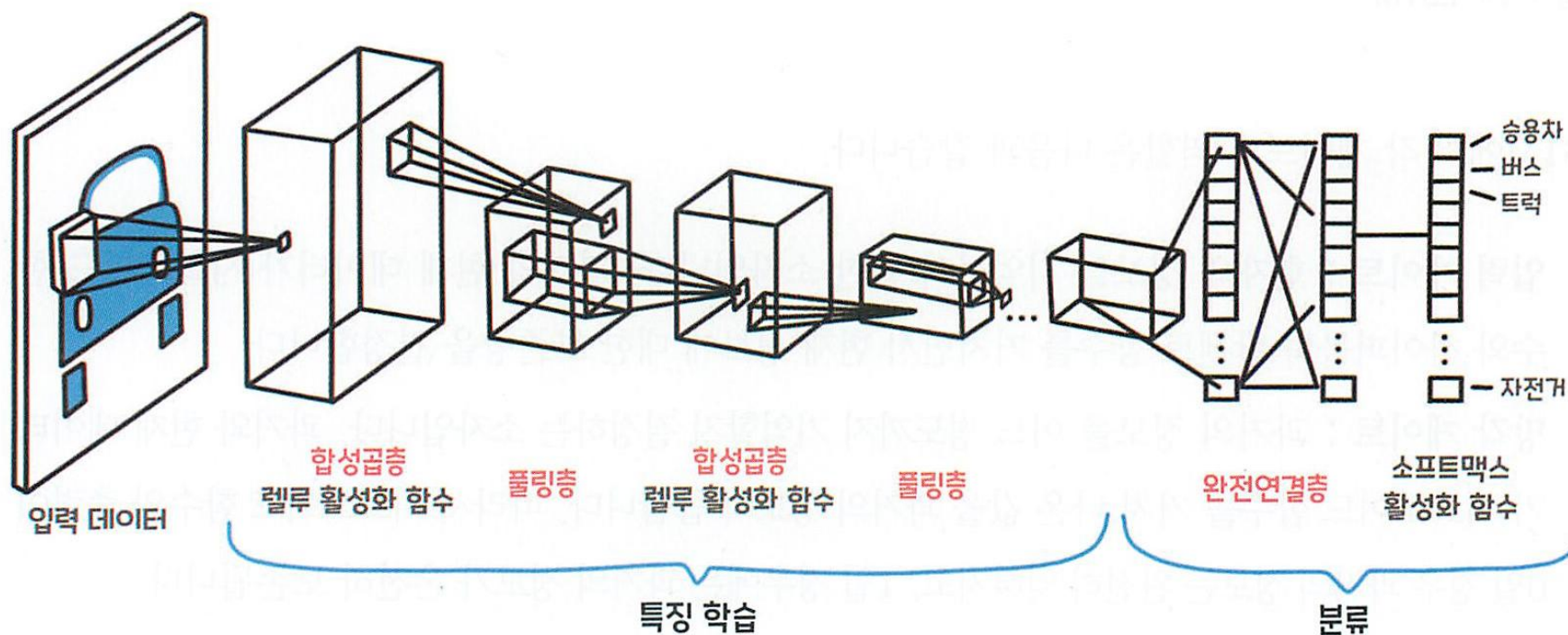
망각 게이트: 과거 정보를 얼마나 기억할지 결정
→ 시그모이드 출력값과 과거 정보를 곱해 기억 여부 조절.
→ 0이면 완전 삭제, 1이면 완전 유지.

출력 게이트: 최종적으로 출력할 정보를 결정
→ 출력층으로 보낼 정보의 양을 조절.



딥러닝의 유형 (합성곱 신경망 / CNN)

- 인간의 시각 처리 방식을 모방한 신경망
- 이미지 처리를 위한 합성곱(Convolution) 연산을 도입



딥러닝의 유형 (CNN - 합성곱층)

- 합성곱층은 이미지에서 분류에 필요한 특징(Feature)을 추출하는 역할을 한다.
- 이때 사용하는 것이 필터(Filter) 또는 커널(Kernel)이며, 이미지의 특정 패턴을 감지한다.
- 여러 필터를 사용하면 다양한 특징을 동시에 추출할 수 있다.
- 결과적으로 필터가 적용된 출력은 '특성 맵(Feature Map)'이라 불린다.

입력 이미지



필터 적용

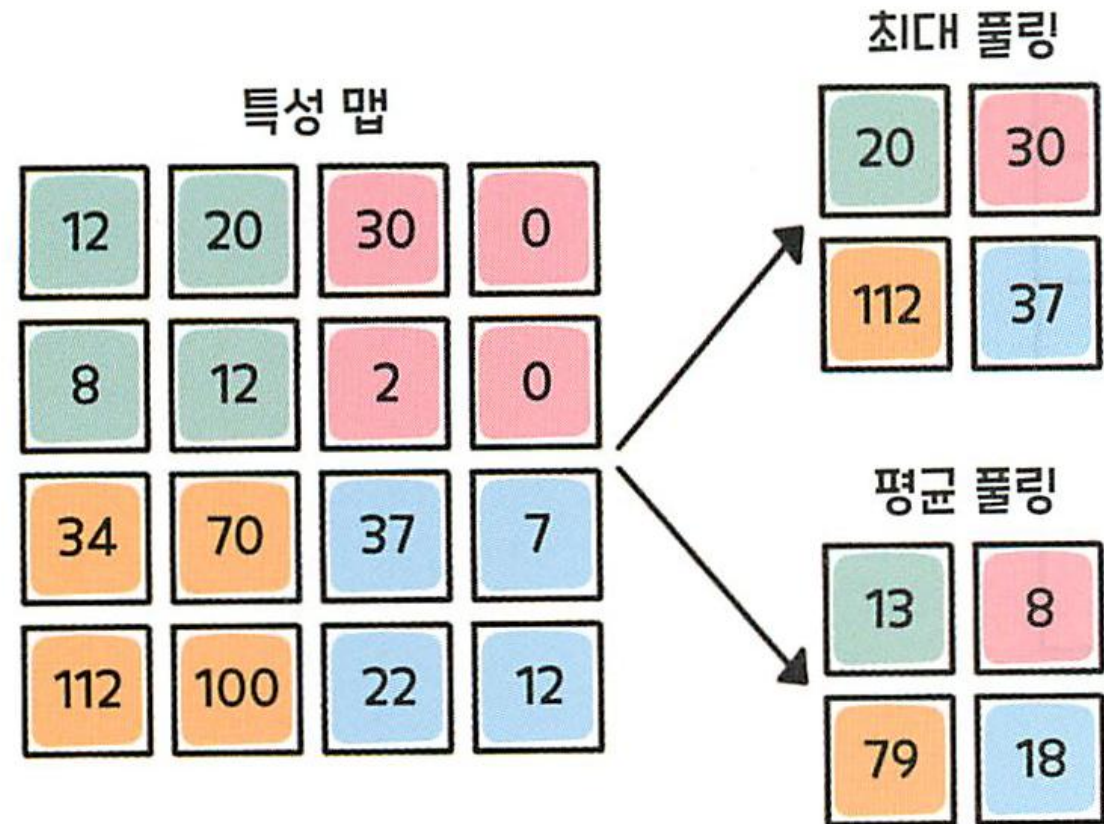
$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

특성 맵



딥러닝의 유형 (CNN – Pooling Layer)

- 풀링층은 합성곱층의 출력(특성 맵)을 입력으로 받아 데이터 크기를 줄이거나 중요 특성만 강조하는 역할을 한다.
- 주요 방식:
 - 최대 풀링(Max Pooling): 영역 내 최대값 추출
 - 평균 풀링(Average Pooling): 영역 내 평균값 추출
 - 최소 풀링(Min Pooling): 영역 내 최소값 추출



딥러닝의 유형 (CNN – Full Connected Layer)

- 완전연결층(Fully-connected Layer)은 합성곱층(Convolution Layer)과 풀링층(Pooling Layer)을 통해 추출된 특징을 종합하여 최종 분류를 수행하는 계층
- 합성곱 신경망(CNN)은 필터를 통해 중요한 특성만을 학습하기 때문에, DFN이나 RNN에 비해 학습해야 할 매개변수의 수가 적고, 학습 및 추론 속도가 빠르다는 장점을 갖는다.
- 최근에는 CNN의 높은 예측 성능과 연산 효율성을 바탕으로, 기존의 이미지 분석을 넘어 시계열 데이터 등 다양한 도메인에도 응용하려는 연구가 활발히 진행되고 있다.

딥러닝의 유형 (비교 정리)

항목	DFN (순방향 신경망)	RNN (순환 신경망)	CNN (합성곱 신경망)
입력 데이터 유형	정적인 데이터 (고정된 형태)	시계열 데이터 (시간 의존성 있음)	이미지, 음성, 시계열 (공간적 특성 중요)
데이터 흐름 구조	입력 → 은닉층 → 출력층 (단방향)	은닉층에서 순환 연결 (이전 상태 반영)	합성곱 → 풀링 → 완전연결층
특징 처리 방식	전체 입력을 동일한 방식으로 처리	과거 정보(기억 상태)를 현재 처리에 반영	필터(커널)를 통한 지역적 특징 추출
학습 난이도	비교적 쉬움	학습 어려움 (기울기 소실 문제 등)	중간 정도, 병렬처리 가능
대표 사용 예시	간단한 분류/회귀 문제	자연어 처리, 시계열 예측, 음성 인식 등	이미지 분류, 물체 인식, 음성/영상 분석 등
파라미터 수	많음 (층마다 완전 연결)	많음 (시간 단위로 반복 학습)	적음 (필터 공유로 효율적)
연산 효율성	낮음	낮음 (연산 병렬화 어려움)	높음 (GPU 병렬 처리 용이)

워드 임베딩 (One-hot Encoding)

- 워드 임베딩은 단어를 벡터로 표현하는 방법. 기계가 사람의 언어를 이해할 수 있도록 변환
- 원-핫 인코딩 (One-hot Encoding)은 N개의 단어를 각각 N차원의 벡터로 표현하는 방식
- 단어가 포함되는 자리에는 1 / 나머지 자리에는 0

color	color_red	color_blue	color_green
red	1	0	0
green	0	0	1
blue	0	1	0
red	1	0	0

워드 임베딩 (Word2Vec)

- Word2Vec은 비슷한 문맥(context)에서 등장하는 단어들이 유사한 의미를 가진다는 전제 하에, 단어를 벡터로 표현하는 기법이다.
- 대표적인 두 가지 학습 모델:
 - CBOW (Continuous Bag-of-Words)
 - 주변 단어들(context)로부터 중심 단어를 예측
 - 예: 철수는 [??]를 구매했다 → '책' 예측
 - Skip-gram
 - 중심 단어로부터 주변 단어들을 예측
 - 예: [책] → 도서, 구매, 철수... 등 문맥 예측

워드 임베딩 (TF-IDF)

- TF-IDF는 단어마다 가중치를 부여해 문서에서 단어의 중요도를 파악하는 방법이다.

✓ 구성 요소

- TF (Term Frequency): 특정 단어가 문서 내에 등장한 횟수
- DF (Document Frequency): 특정 단어가 등장한 문서의 수
- IDF (Inverse Document Frequency): DF에 반비례하는 값
→ 많이 등장하는 단어일수록 IDF가 낮음

📌 특징

- 조사는 많은 문서에 자주 등장하므로 IDF가 낮고,
- 따라서 중요도가 낮은 단어로 간주된다.
- TF-IDF를 통해 중요한 키워드를 효과적으로 추출할 수 있다.

워드 임베딩 (Fasttext)

- FastText는 페이스북에서 개발한 워드 임베딩 방법으로, 단어를 부분 단어(Subword)로 나누어 벡터화함.
- 이를 위해 N-gram(예: Trigram)을 사용하여 연속된 문자 조각을 생성함.

"Best books about AI" → " Best books about", " books about AI"

- 이 방식은 OOV(Out of Vocabulary) 문제(모르는 단어 처리 문제)를 효과적으로 해결함.
- FastText는 Word2Vec보다 OOV에 강해 임베딩 성능이 뛰어나며, GloVe, ELMo 등 다양한 임베딩 모델과 함께 널리 사용됨.

워드 임베딩 (비교정리)

항목	Word2Vec	FastText	GloVe	ELMo
개발 주체	Google	Facebook	Stanford Univ.	Allen Institute for AI
출시 연도	2013	2016	2014	2018
기본 아이디어	주변 단어로 중심 단어 예측 (CBOW, Skip-gram)	Word2Vec + Subword(N-gram) 구조 추가	단어 간 동시 등장 통계에 기반한 벡터 학습	문장 전체 문맥 기반 동적 벡터 생성
OOV(모르는 단어) 처리	불가능	가능 (부분 단어 기반 조합)	불가능	가능 (문맥 기반 구성)
문맥 반영	❌ (고정 벡터)	❌ (고정 벡터)	❌ (고정 벡터)	✅ (동일 단어라도 문맥 따라 벡터 달라짐)
학습 방식	예측 기반 (Predictive)	예측 기반 + N-gram	카운트 기반 (Co-occurrence)	사전학습된 언어모델 기반
장점	빠른 속도, 직관적 구조	OOV 강인성, 다국어 적합	전체 말뭉치의 통계적 의미 반영	문맥 정보 반영, 문장 수준 의미 이해 가능
단점	OOV 문제, 문맥 미반영	속도가 다소 느릴 수 있음	OOV 문제, 문맥 미반영	속도 느림, 연산량 많음
주 사용 분야	기본 단어 임베딩, NLP 기초	다국어 처리, OOV 회피 필요한 응용	대규모 텍스트 분석, 뉴스 등	문장 의미 분석, QA, 감성분석 등 복잡한 NLP 문제

적대적 생성 신경망 (GAN)

- GAN(Generative Adversarial Network)은 두 개의 신경망, 즉 생성 모델(Generator)과 판별 모델(Discriminator)이 경쟁적 학습(adversarial training)을 통해 고품질의 데이터를 생성하는 강화학습입니다.
- **생성 모델**은 주어진 학습 데이터와 최대한 유사한 가짜 데이터를 생성
- **판별 모델**은 입력된 데이터가 실제(real)인지, 생성된(fake) 것인지를 판별
- 이러한 경쟁 구조를 통해 두 모델은 반복적으로 성능을 개선하게 되며, 특히 이미지 생성, 예술 창작, 고해상도 변환 등의 분야에서 뛰어난 결과를 보이고 있습니다.
- 실제로 2018년에는 GAN으로 생성한 작품이 크리스티 미술품 경매에서 고가(\$430,000)에 낙찰될 정도로 현실성 있는 생성 능력을 인정받고 있습니다.
- GAN의 작동 원리는 위조지폐 제작에 비유할 수 있습니다. 생성 모델은 진짜처럼 보이는 위조지폐를 만들고, 판별 모델은 이를 감별하며 두 모델이 경쟁적으로 성능을 향상시키는 방식으로 학습이 진행됩니다.

