

4차시 딥러닝이란?

A - 시 대발자

학습목표

딥러닝의 개념에 대해 학습하고, 퍼셉트론과 신경망의 구조를 이해한다.

목차

- 1. 딥러닝의 등장 배경
- 2. 퍼셉트론과 신경망

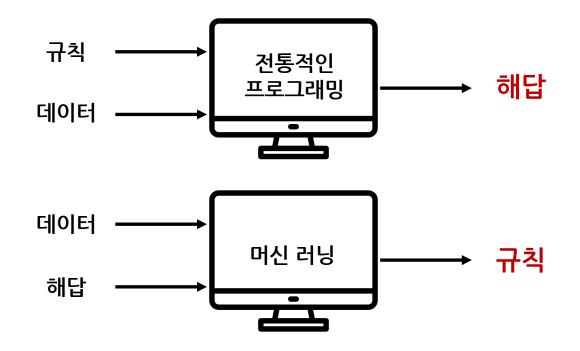


목차

1. 딥러닝의 등장 배경



답러닝이란 무엇일까? 우리는 앞선 수업에서 인공지능, 머신러닝, 딥러닝이 무엇인지 배웠습니다. 딥러닝에 대해 보다 자세히 알아보기에 앞서, 전통적인 프로그래밍으로 만든 지능형 프로그램과 머신러닝 알고리즘으로 만든 인공지능 프로그램의 차이를 알아봅시다.



명시적 프로그래밍은 입력 데이터로부터 원하는 해답을 도출하도록 사람이 규칙을 설계하여 컴퓨터 프로그램으로 만듦

머신러닝은 컴퓨터가 데이터로부터 학습하여 문제를 해결하는 규칙을 도출



머신러닝을 하기 위해서는 세 가지가 필요합니다.

입력 데이터, 기대 출력값, 알고리즘의 평가와 피드백 방법

아래 세 가지가 있어야 컴퓨터가 규칙을 학습할 수 있습니다.



입력 데이터

컴퓨터에게 주어지는 모든 종류의 데이터입니다. 주어진 문제가 음성 인식이라면, 사람의 대화가 녹음된 사운드파일, 객체 탐지라면, 사진입니다.



기대 출력값

학습시킨 후 기대하는 출력값입니다. 음성 인식 문제라면 대화를 듣고 옮긴 글이 되고, 객체 탐지라면 강아지, 고양이 등과 같은 태그입니다.



알고리즘의 평가와 피드백 방법

현재의 출력값과 기대 출력값 간의 차이를 결정하기 위해 알고리즘을 평가하고, 고도화 시키기 위해 피드백을 진행합니다.

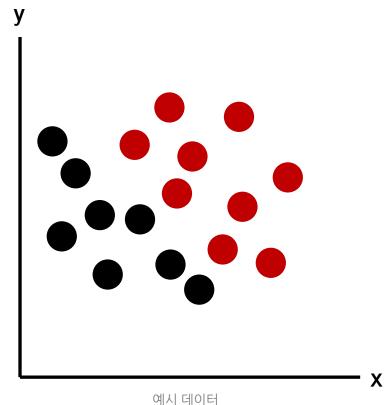


의 등장 배경

입력 데이터를 의미 있는 데이터로 바꾸어 표현한다는 것이 어떤 의미일까요? 간단한 예시 데이터를 통해 이해해 보겠습니다.

검은색 구슬과, 빨간 구슬의 좌표를 입력 데이터로 받고, 그 구슬이 검은색인지 빨간색인지

출력하는 알고리즘을 개발하려고 합니다.



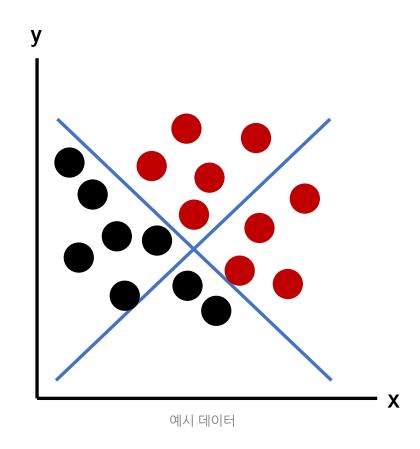
입력 데이터: 구슬의 좌표(x, y)

Ⅰ 기대 출력값 : 구슬의 색깔

알고리즘의 평가 기준: 정확히 분류한 구슬의 비율



머신러닝이 학습을 통해 아래와 같이 새로운 좌표를 만들어 구분한다면 어떨까요?



머신러닝이 찾은 알고리즘 X > 0이라면 빨간 구슬이다. X < 0이라면 검은 구슬이다.

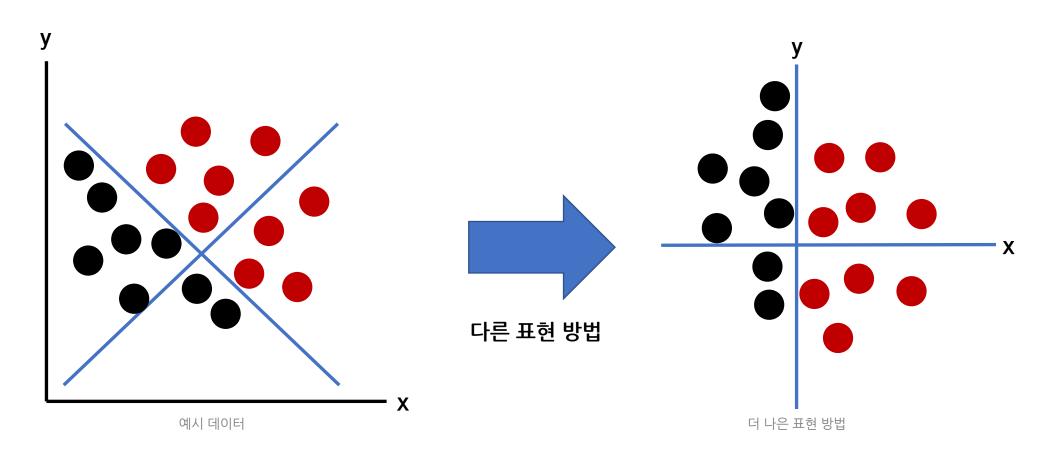


머신러닝이 빨간색, 검은색 구슬의 좌표를 입력받아 출력한 구슬을 구분하는 방법입니다.

머신러닝, 딥러닝의 핵심은 의미 있는 데이터로의 변환입니다.

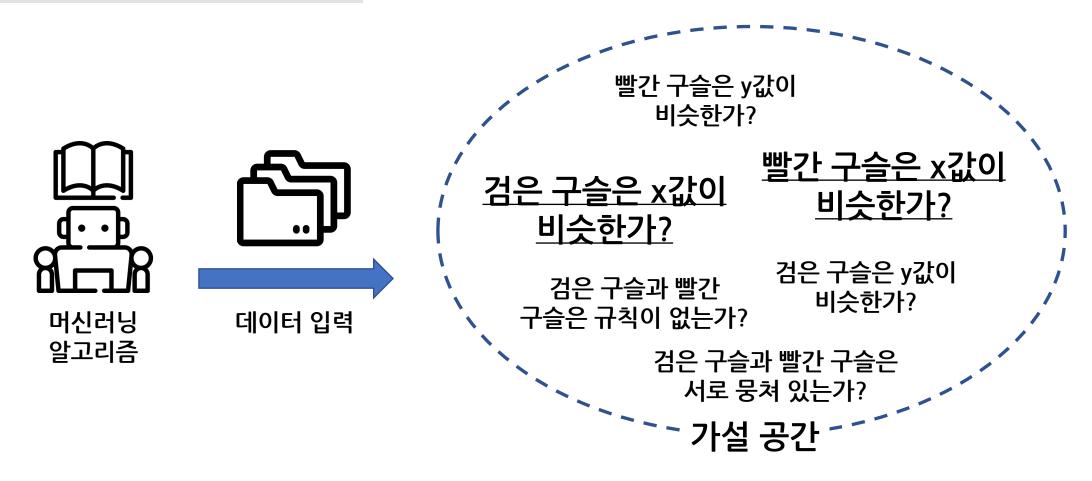
왼쪽처럼 데이터를 표현할 경우 의미를 파악하기 어렵지만,

오른쪽처럼 표현한다면 의미 파악도 쉽고, 우리가 원하는 기대 출력값에 더 가까울 것입니다.





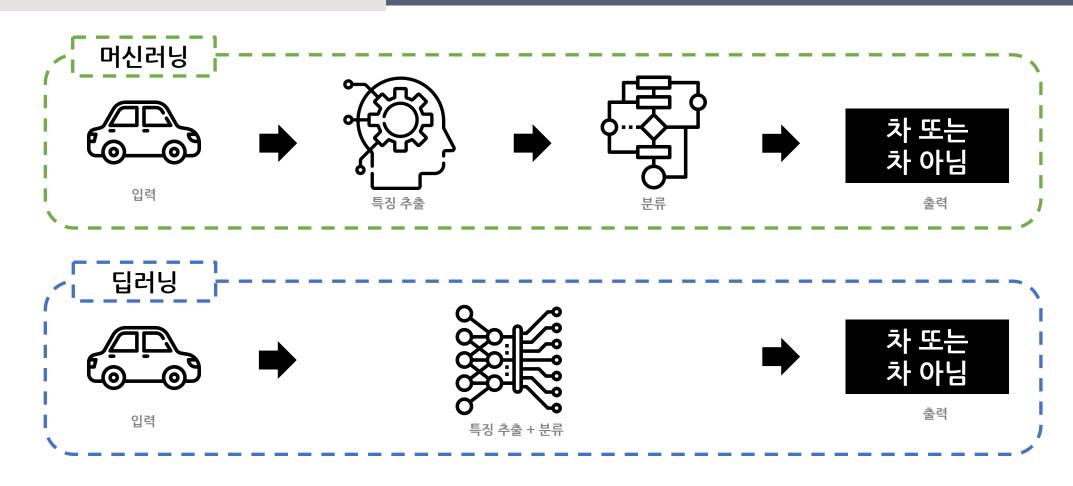
모든 머신러닝 알고리즘은 주어진 문제를 해결하기 위해 데이터를 더 유용한 표현으로 바꾸는 변환을 자동으로 찾아줍니다. 하지만, 고전적인 머신러닝의 알고리즘은 이러한 변환을 찾기위한 창의력은 없었습니다. 사람이 가설을 세워야 했습니다.





머신러닝 알고리즘은 결국 가설 공간을 사전에 정의하고, 이 공간에 대해 입력 데이터에서 유용한 변환을 찾는 것입니다. 즉, 머신러닝은 입력 데이터에 기대 출력값을 가장 잘 매핑(Mapping)시키는 기능을 학습하는 것입니다.

사람이 개입하여 가설을 세우고, 데이터의 어떤 특징을 학습에서 고려할 것인지 알려주는 '특징 추출' 단계가 필요하다는 점은 고전적 머신러닝의 한계였습니다. 바로 이러한 문제를 해결하기 위해 딥러닝이 등장하게 되었습니다.





딥러닝은 인간의 도움 없이 데이터로부터 원하는 출력을 효과적으로 내는데 필요한 특징 추출과 분류를 모두 스스로 해냅니다. 그렇다면 딥러닝은 어떻게 스스로 모든 것을 할 수 있는 것일까요?

목차

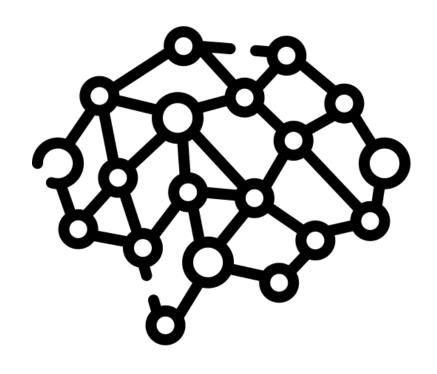
2. 퍼셉트론과 신경망



퍼셉트론』 신경망

딥러닝(Deep Learning)은 1950년대

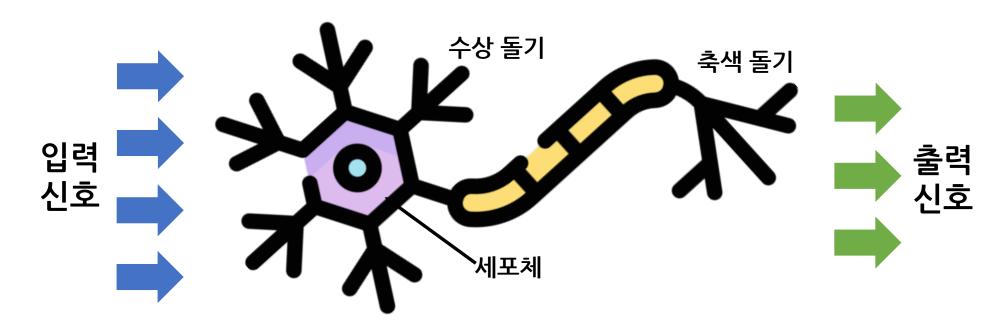
인공 신경망(Artificial Neural Network: ANN) 연구에서 시작되었습니다.



인공 신경망은 사람의 뇌 속 뉴런의 작용을 본떠 패턴을 구상한 컴퓨팅 시스템



퍼셉트론에 대해 이해하기 위해선 먼저 인간의 두뇌를 구성하고 있는 뉴런(Neuron)에 대해 먼저 알아봅시다. 뉴런은 신경계를 구성하는 세포로, 인간의 뇌에는 600억 ~ 1000억 개의 뉴런이 있습니다.

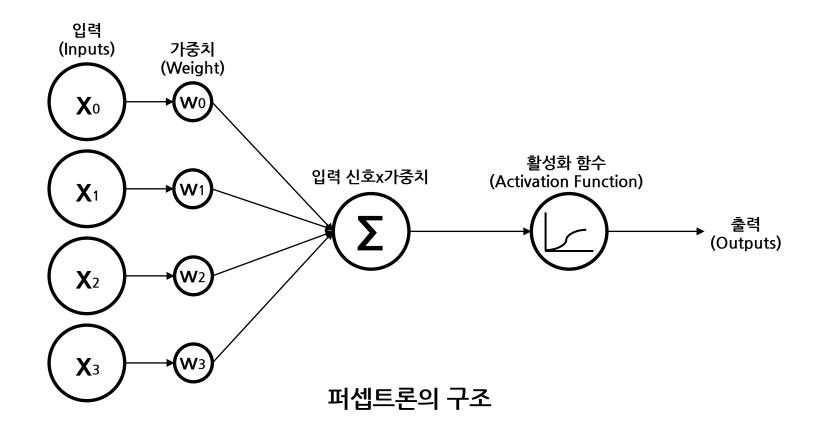






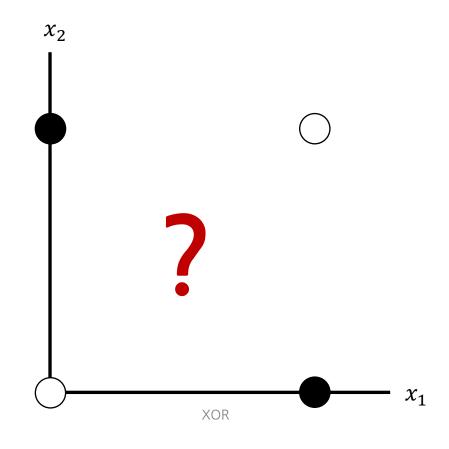
뉴런은 수상 돌기를 통해 주위 뉴런들로부터 신경 자극을 입력받아서 세포체에서 처리를 한 다음 임계치에 도달하면 발화하여 축색돌기를 통하여 다른 세포들로 출력을 내보냅니다. 이런 구조는 인공신경망 개발에 영감을 주었습니다.

프랑크 로젠블렛(Frank Rosenblatt) 1957년 뉴런에서 영감을 받아 '퍼셉트론'이라는 알고리즘을 고안했습니다. 퍼셉트론은 인공신경망의 기원이 되는 알고리즘입니다.





퍼셉트론이 발표되고, 모두가 스스로 생각하고 말하는 컴퓨터를 만드는 것은 시간문제로 생각했습니다. 하지만, 퍼셉트론의 한계점은 XOR(Exclusive OR) 문제를 통해 드러납니다.





위의 좌표에 하나의 직선을 그어 검은 구슬, 흰 구슬 같은 색끼리 나눌 수 있나요? 여러분도 한번 문제를 풀어보세요.

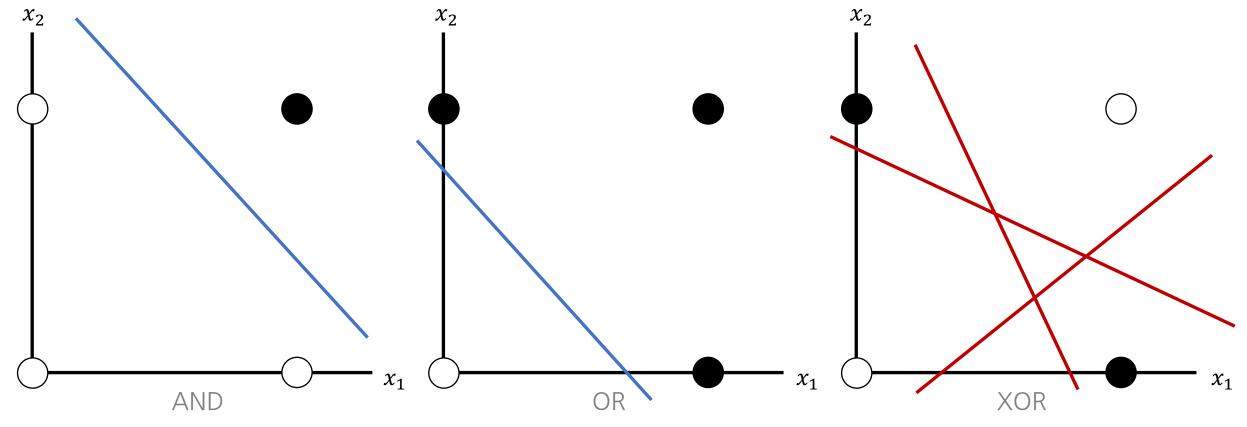
단층 퍼셉트론은 AND, OR 문제같이 **2차원 공간의 선형의 문제**는 풀 수 있었지만, XOR문제는 해결할 수 없었습니다.

AND 진리표			OR 진리표			XOR 진리표		
x_1	x_2	결과값(y)	x_1	x_2	결과값(y)	x_1	x_2	결과값(y)
0	0	0	0	0	0	0	0	0
0	1	0	0	1	1	0	1	1
1	0	0	1	0	1	1	0	1
1	1	1	1	1	1	1	1	0



AND 문제는 두 입력값 (x_1, x_2) 이 모두 1일 때만 결괏값을 1로 출력합니다. 한편 OR 문제는 두 입력값 중하나라도 1이면 결과값을 1로 출력합니다. XOR 문제는 입력값 중하나만 1일 때 1을 출력합니다.

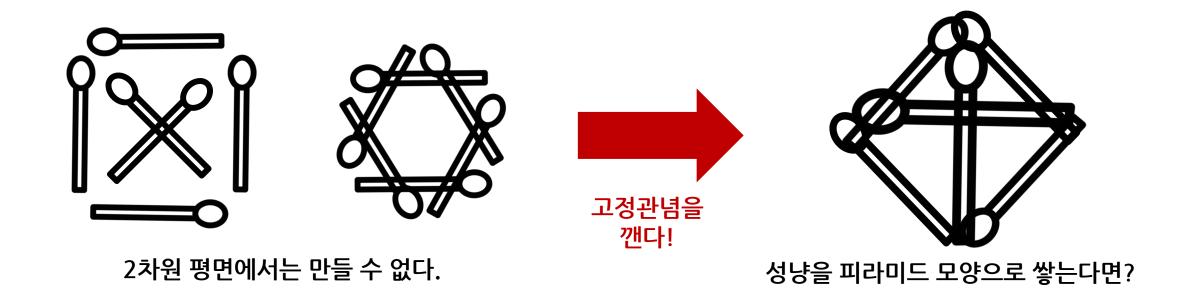
AND, OR, XOR 문제를 좌표 평면에 옮기고, 결괏값을 구분할 수 있는 하나의 직선을 그어봅시다. AND와 OR 문제는 직선 하나로 문제 해결이 가능하지만, XOR 문제는 어떤 방법을 써도 결괏값을 직선 하나로 구분할 수 없습니다.





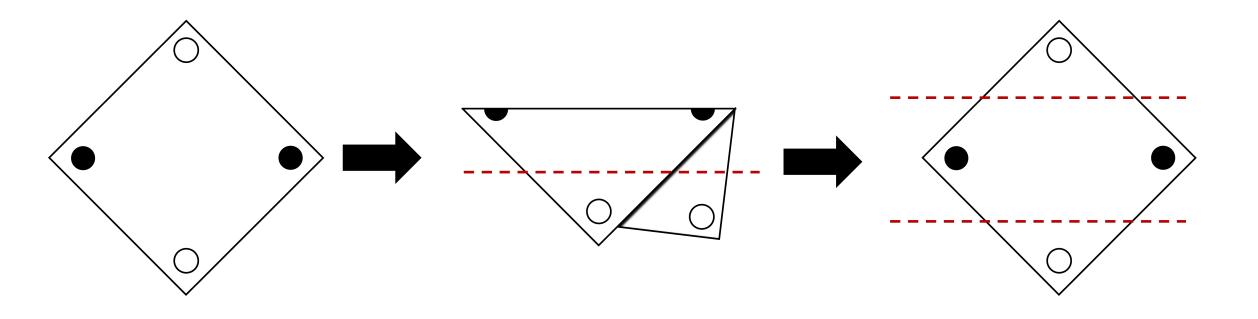
이런 XOR문제의 등장으로 인해 인공지능 역사의 딥러닝 또한 긴 암흑기를 맞이하게 됩니다. 하지만, XOR 문제를 해결하는 방법은 생각보다 간단했습니다.

2차원 평면에서는 XOR 문제를 해결할 수 없습니다. 우리는 이와 비슷한 문제를 알고 있는데요. 바로 '성냥개비 여섯 개로 정삼각형 네 개를 만들 수 있는가?' 문제입니다.



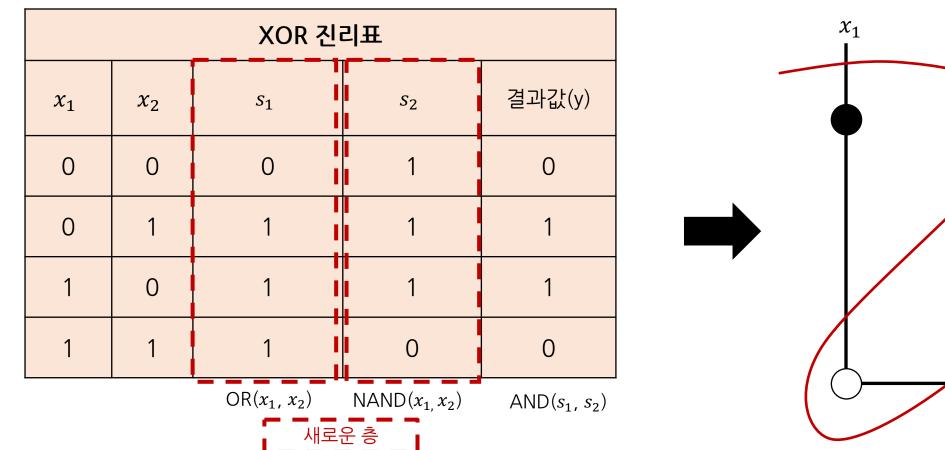


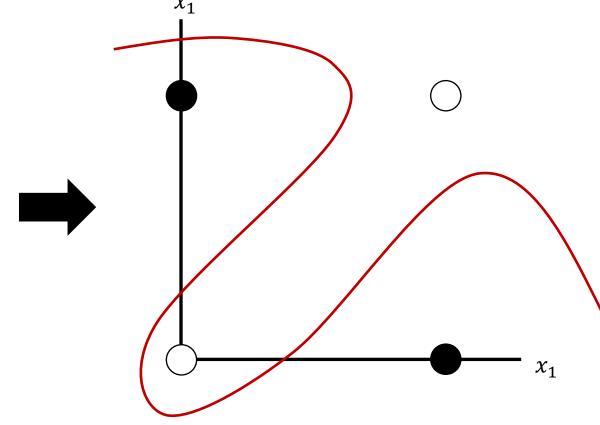
성냥개비 문제의 방식을 XOR 문제에 적용한다면 종이를 **휘어 주어 선을 그어볼 수 있을 것입니다**.





두 개의 퍼셉트론을 계산하는 것이란 입력과 출력 사이에 새로운 층(게이트)를 추가하는 것입니다. XOR 문제는 OR, AND, NAND 게이트를 조합하여 새로운 층을 만들면 해결할 수 있습니다.

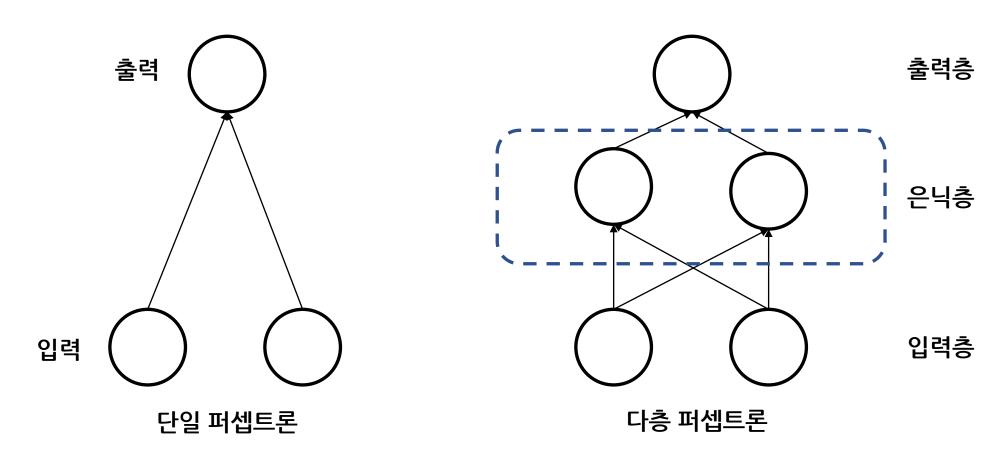






입력값 (x_1, x_2) 은 각각 새로운 층 (s_1, s_2) OR, NAND 게이트에 따라 계산되고, 새로운 층에서 계산된 값은 AND 게이트에 따라 한 번 더 계산되어 최종 결괏값을 도출합니다.

다층 퍼셉트론은 단일 퍼셉트론에 숨어있는 층, <mark>은닉층(Hidden layer)</mark>을 만들어줍니다. 마치 2차원 평면이던 성냥개비를 세워 3차원 피라미드로 만들어 층을 추가한 것처럼 말이죠.

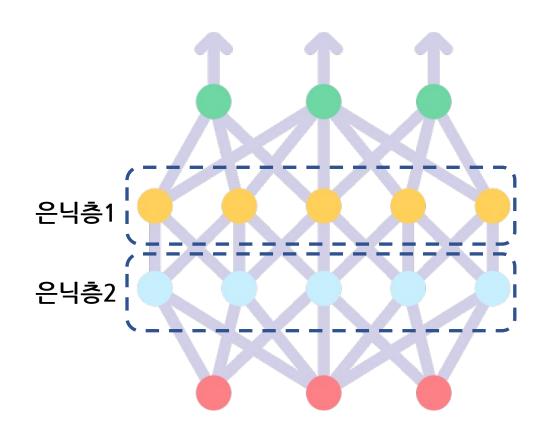




퍼셉트론』 신경망

퍼셉트론에 은닉층을 추가함으로써 더 좋은 학습 결과를 얻을 수 있게 되었습니다.

그렇다면 이 은닉층을 여러 겹 쌓는다면 더 좋은 결과를 얻을 수 있지 않을까요?

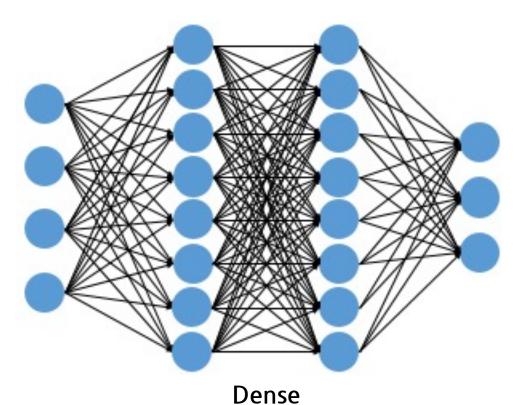


딥러닝(Deep Learning)은

연속된 층을 통해 학습하는 모델입니다. 이때, 딥(Deep)은 층이 많을수록 깊이가 깊어지기(deep) 때문에 붙은 이름입니다. 이런 방법을 층 기반 표현 학습, 계층적 표현 학습이라고 합니다.



딥러닝에서 가장 기본적인 형태는 입력과 출력, 그 사이에 있는 모든 층을 연결하는 밀집형 구조의 심층 신경망입니다.



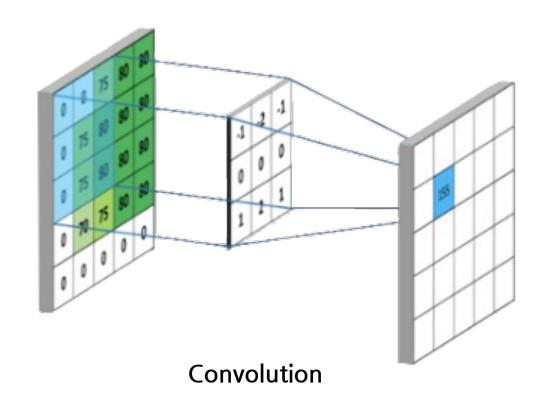
밀집(Dense) 레이어는 딥 러닝에서 가장 일반적인 형태의 레이어다.





합성곱(Convolution) 레이어는 이미지에 필터를 적용하여 특징을 추출합니다.

이를 활용한 모델을 합성곱 신경망(Convolutional Neural Network: CNN)이라 합니다.

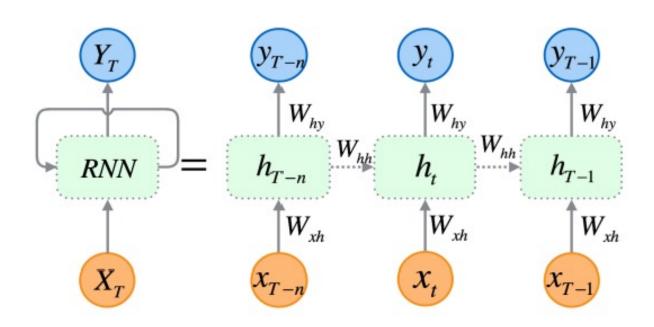


합성곱(Convolution) 레이어는 풀링(Pooling) 레이어와 주로 같이 사용되며 사람 얼굴 인식, 자율 주행차의 실시간 사물 인식 등 널리 사용 되고 있습니다.



합성곱 신경망의 핵심은 이미지 특징을 추출하는 합성곱 레이어와 이미지의 크기를 줄여주는 풀링 레이어를 서로 반복되는 형태로 만드는 것입니다.

순환(Recurrent) 레이어는 시계열 데이터와 같이 시간의 흐름에 따라 변화하는데이터를 학습하기 위한 레이어로, 과거 데이터가 현재 데이터에 영향을 주는 것이 특징입니다.이를 이용한 모델을 순환 신경망(Recurrent Neural Network: RNN)이라 합니다.



순환(Recurrent) 레이어는 과거의 출력 데이터를 참조합니다. 대표적인 활용 예시로는 문자 자동완성이 있습니다.

Recurrent



문자 데이터의 경우 앞뒤 글자 간 연관성이 있습니다. 만약 우리가 '동해물과'라고 입력한다던 그 다음은 높은 확률로 '백두산이' 이 나올 것을 예측 가능한 것과 같은 원리입니다.

