

Deep Learning Introduction

WeGo

목 차

1. 인공지능 소개
2. 딥러닝
3. 객체 검출 모델
4. 출처
5. 부록

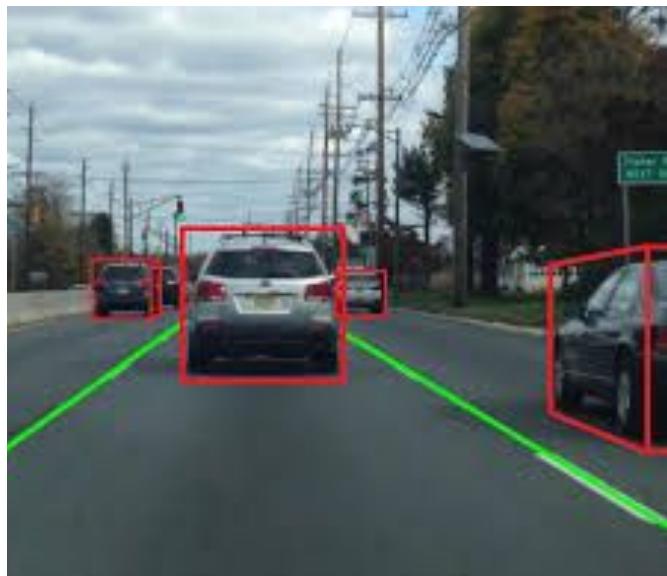
01

인공지능 소개

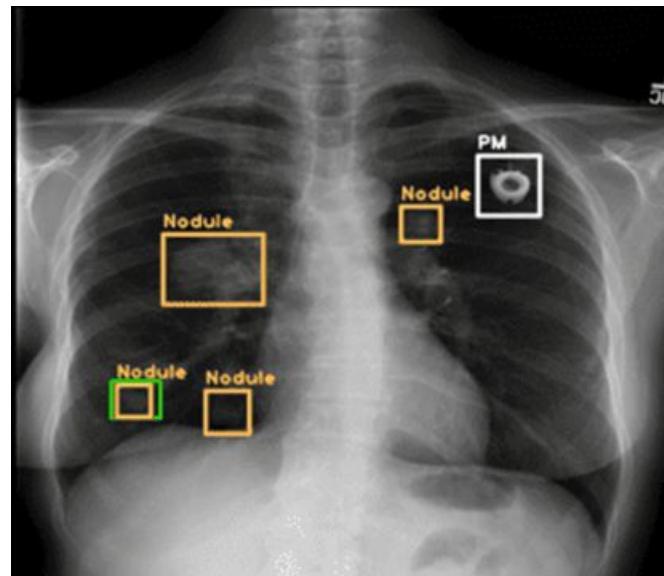
01 인공지능 소개

- 인공지능의 발전

- 인공지능은 많은 분야에 적용되어 우리의 생활에 많은 변화를 주고 있습니다.



자율주행 예시) 객체 및 차선 검출



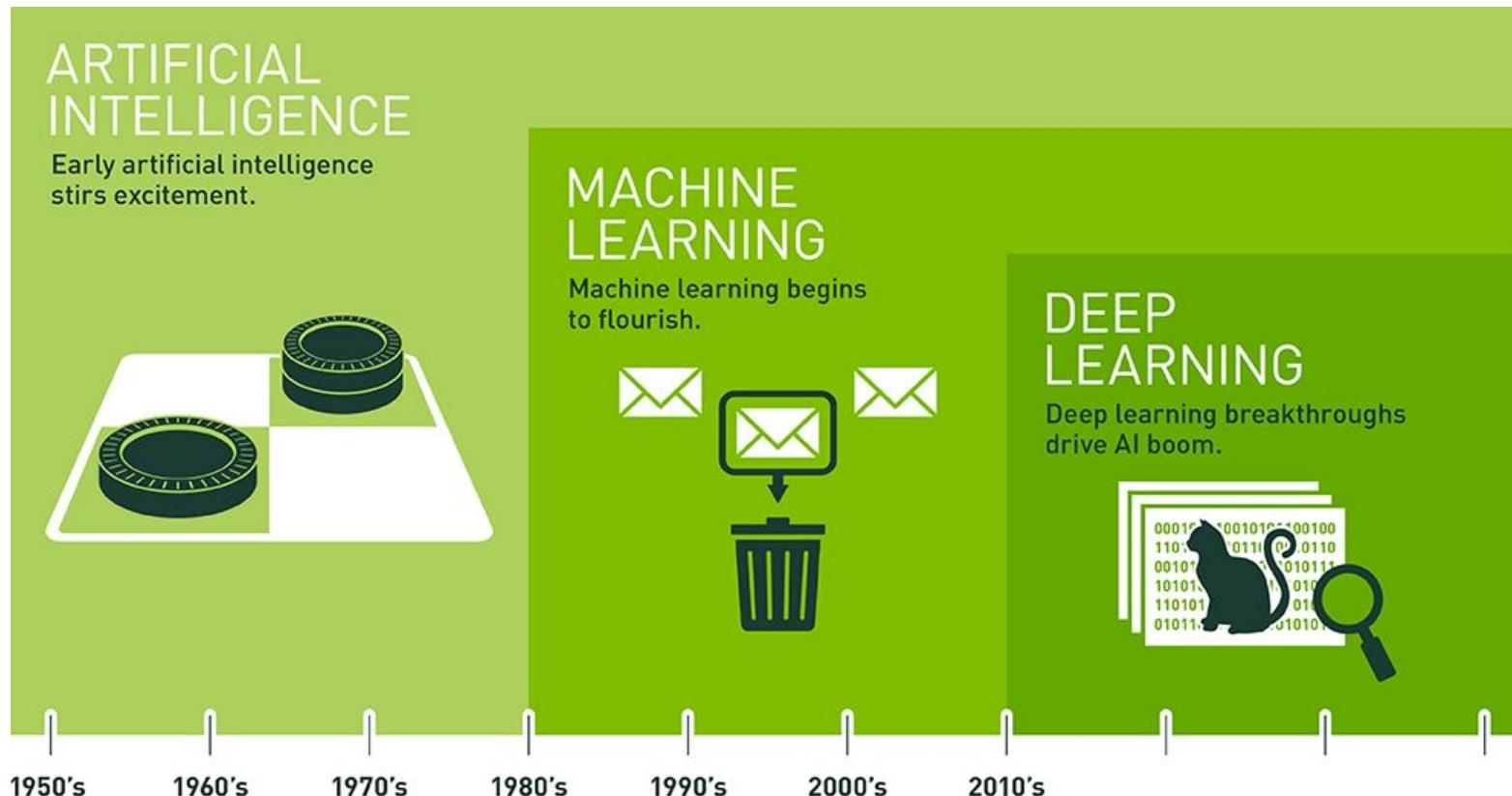
의료 예시) 종양 검출



농업 예시) 식물 질병 검출

01 인공지능 소개

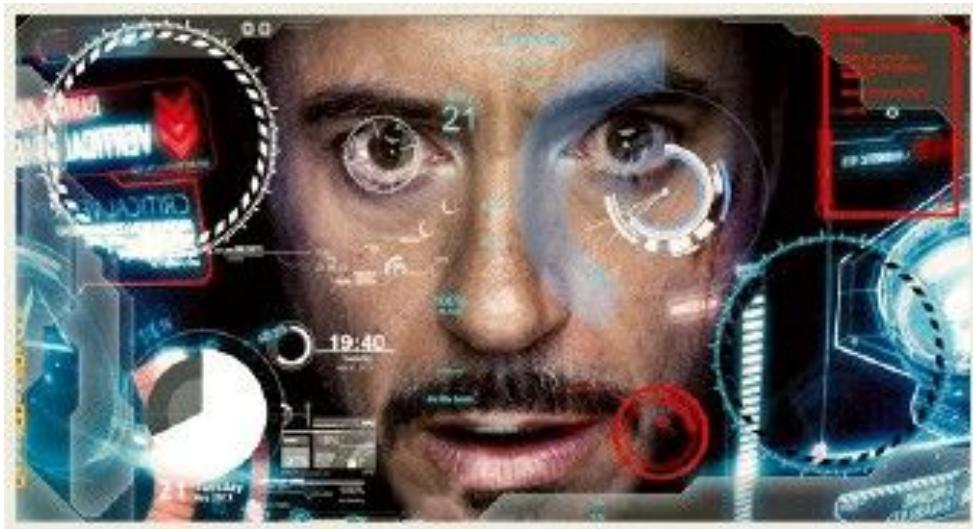
- 인공지능과 머신러닝, 딥러닝의 포함관계
 - 머신러닝은 인공지능의 하위 개념, 딥러닝은 머신러닝의 하위 개념입니다.



(이미지 출처 : <https://blogs.nvidia.co.kr/>)

01 인공지능 소개

- 인공지능 (Artificial Intelligence)
 - 인간의 학습능력, 추론능력, 지각능력을 인공적으로 구현하려는 컴퓨터과학의 한 세부분야입니다.
 - 강인공지능 (Strong A.I.) : 인간처럼 사고하고 행동하는 인공지능
 - 약인공지능 (Weak A.I.) : 인간은 쉽게 해결할 수 있으나 컴퓨터는 어려워하는 문제를 수행하는 인공지능



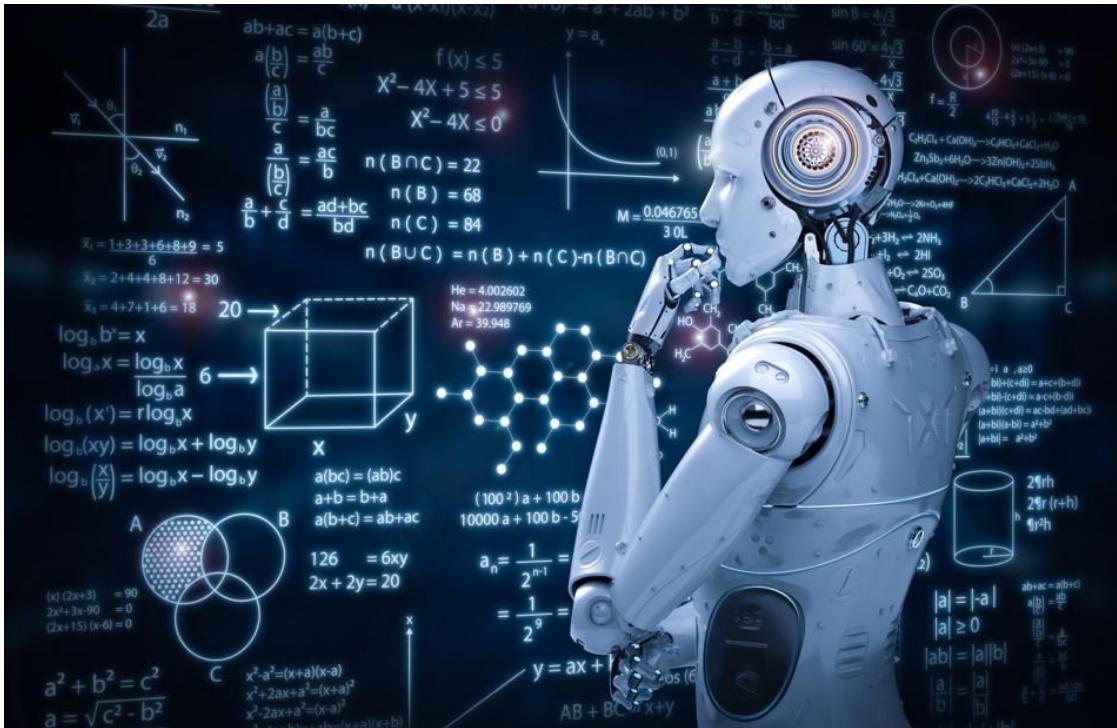
강인공지능 예시) 영화 '아이언맨' 속 인공지능 비서 '자비스'



약인공지능 예시) 바둑 인공지능 '알파고'

01 인공지능 소개

- **머신러닝 (Machine Learning)**
 - 기계가 친절한 판단을 할 수 있도록 스스로 학습하게 하는 기술입니다.
 - 프로그래머가 고려하지 않은 상황에 보다 유연한 대처가 가능합니다.



(이미지 출처 : <https://www.forbes.com/>)

01 인공지능 소개

- **머신러닝 (Machine Learning)**
 - 머신러닝은 크게 두 가지로 분류됩니다.
 - **지도학습 (Supervised Learning)** : 훈련 데이터들과 그에 대응하는 정답을 함께 학습시켜, 컴퓨터가 데이터와 정답 사이의 규칙을 학습하는 방법
 - **비지도학습 (Unsupervised Learning)** : 입력 데이터는 있지만 정해진 답은 없으며, 데이터들의 특징을 분석하여 비슷한 데이터들끼리 군집화하는 방법

01 인공지능 소개

- **지도학습 (Supervised Learning)**

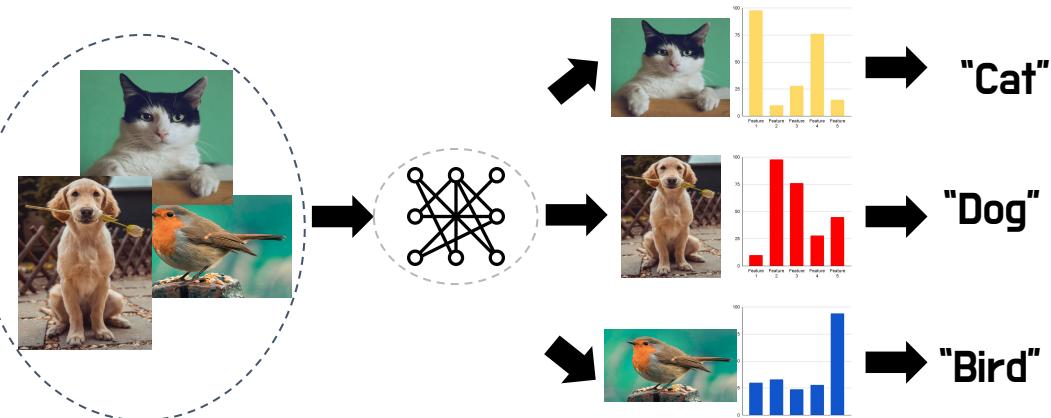
- 지도학습은 훈련 데이터의 특징을 분석하고 라벨 (정답) 사이의 규칙을 학습하는 방법입니다.
- 규칙이란 주어진 데이터로부터 정답을 유추하는 함수를 뜻합니다.
- 지도학습은 크게 두 가지로 분류됩니다.
 - **분류 (Classification)** : 주어진 데이터들을 분류하는 문제
 - **회귀 (Regression)** : 주어진 데이터들의 패턴이나 경향을 예측하는 문제

01 인공지능 소개

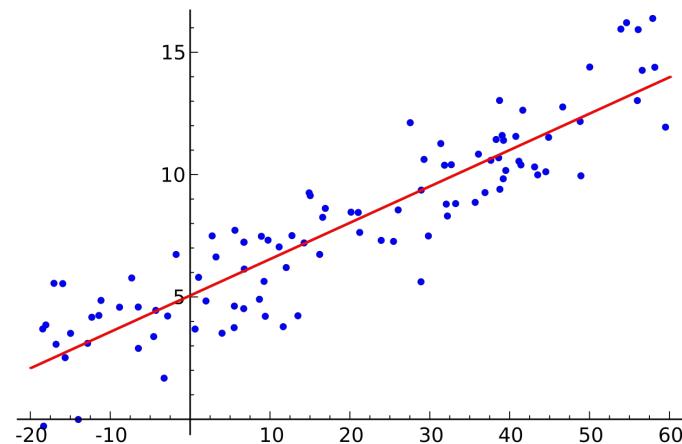
● 지도학습 (Supervised Learning)

○ 예시)

- 분류 : 개와 고양이의 특징을 분석하여 사진을 분류하는 문제
- 회귀 : 주어진 데이터들의 경향성을 예측하는 문제



분류) 동물 사진을 분류하는 문제



회귀) 데이터(파란점)들의 경향성을 분석하여 예측모델(빨간선)을 찾는 문제

(이미지 출처 : 우) <https://en.wikipedia.org/wiki/Regression/>

01 인공지능 소개

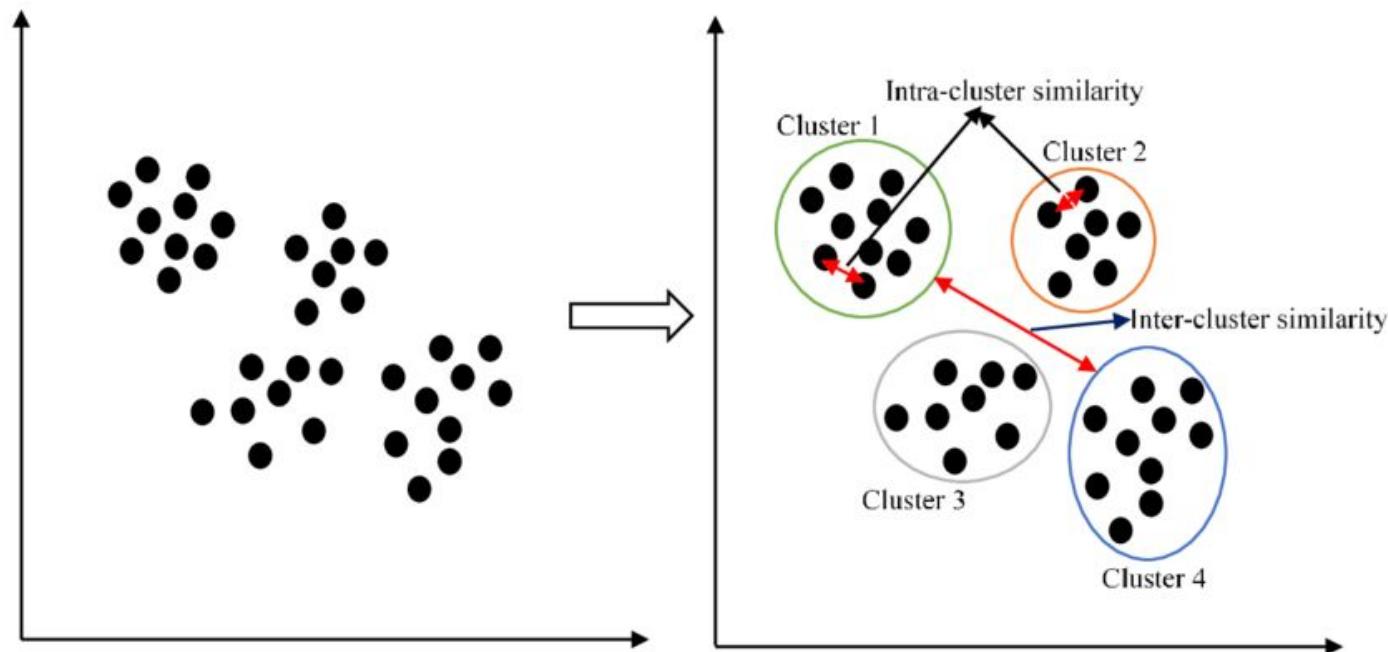
- **비지도학습 (Unsupervised Learning)**
 - 데이터들의 특징을 분석하고 군집화하는 방법입니다.
 - 새로운 데이터의 특징을 분석한 뒤 어느 군집에 포함되는지 알 수 있습니다.
 - 비지도학습은 지도학습과 달리 라벨을 제공하지 않습니다.

01 인공지능 소개

- 비지도학습 (Unsupervised Learning)

- 예시)

- 주어진 데이터들의 특징을 분석하여 비슷한 데이터들끼리 군집화한 모습

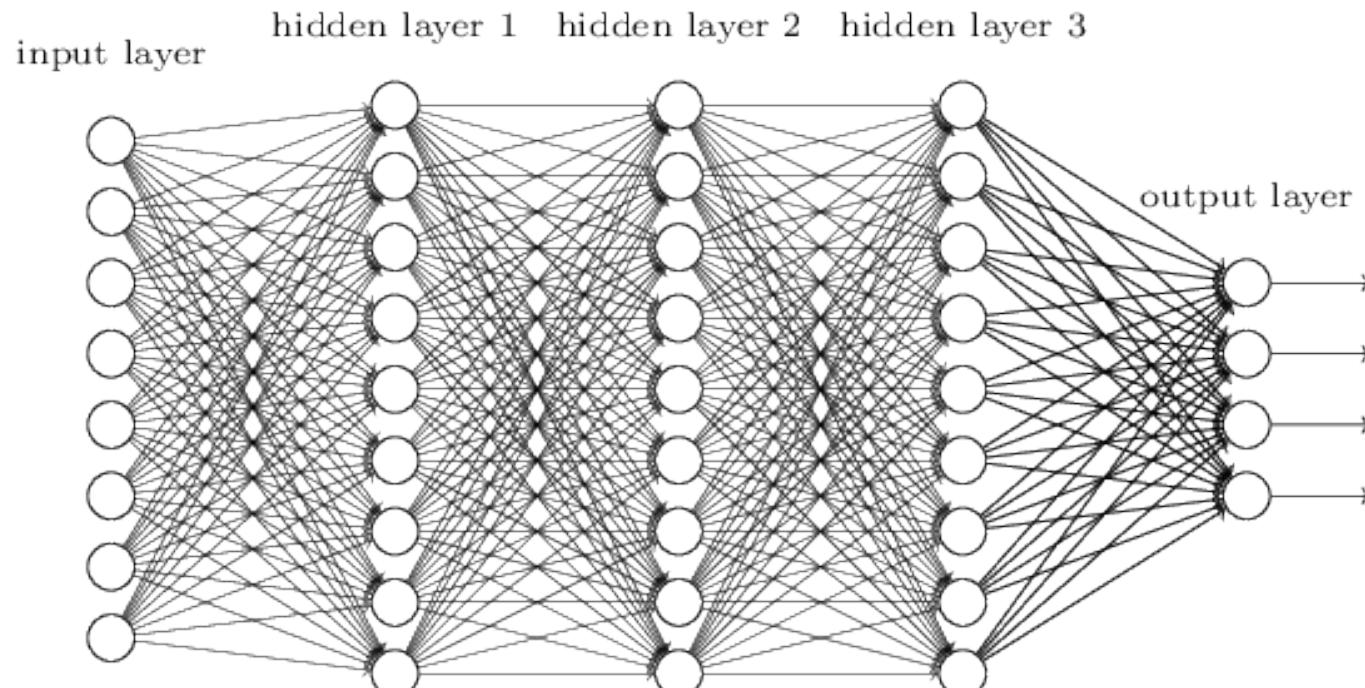


(이미지 출처 : <https://www.researchgate.net/>)

01 인공지능 소개

- 딥러닝 (Deep Learning)

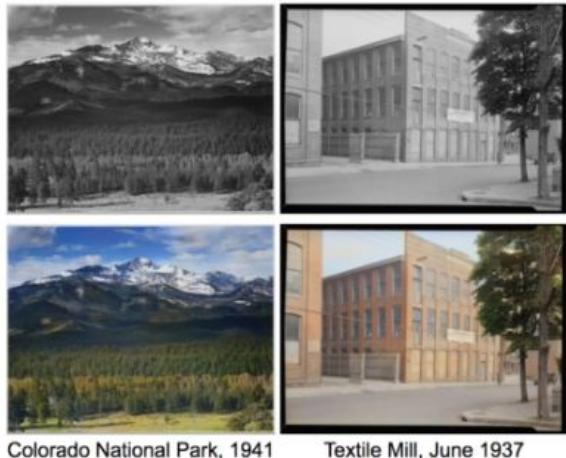
- 딥러닝은 머신러닝의 한 방법으로 뇌의 작동 방식에서 영감을 얻어 문제를 해결하는 방법입니다.
- 딥러닝은 컴퓨터 비전, 자연어 처리 등 다양한 분야에서 성과를 내고 있습니다.



01 인공지능 소개

● 딥러닝 (Deep Learning)

○ 예시)



흑백 사진 색 복원



새로운 이미지 생성하기

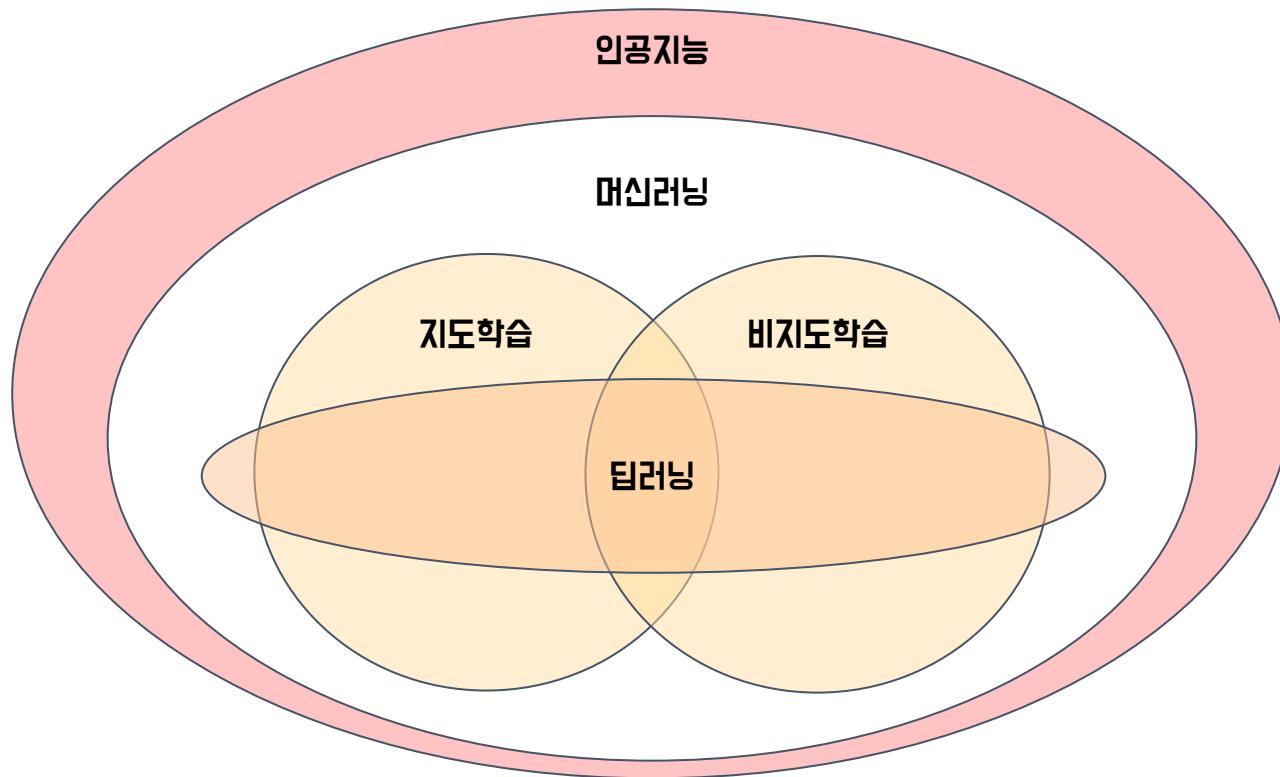


사진 속 언어 번역

01 인공지능 소개

- 딥러닝 (Deep Learning)

- 딥러닝 기법은 성능 향상을 위해 지도학습, 비지도학습 가리지 않고 다양한 알고리즘을 활용합니다.
- 따라서, 앞에서 소개된 개념들의 관계를 정리하자면 아래와 같습니다.



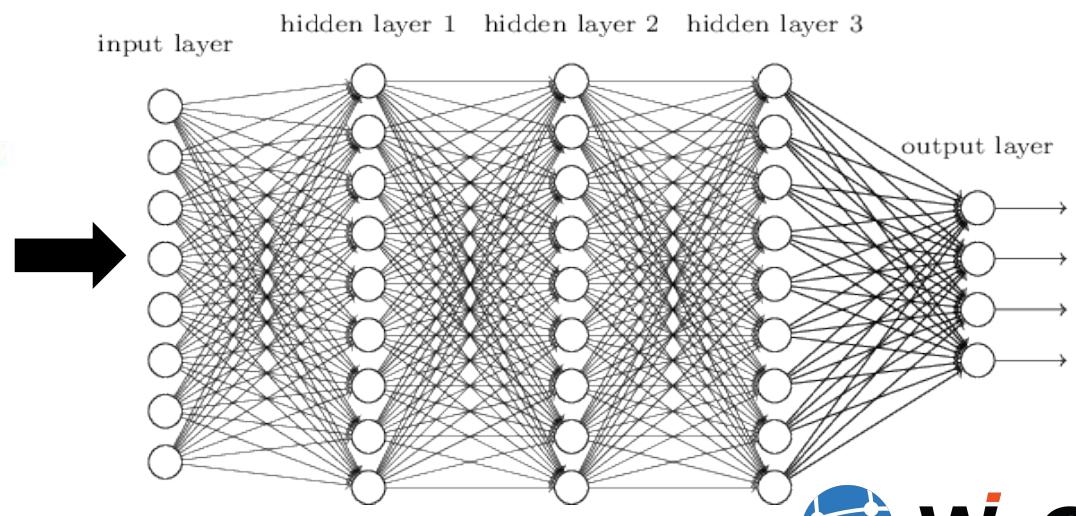
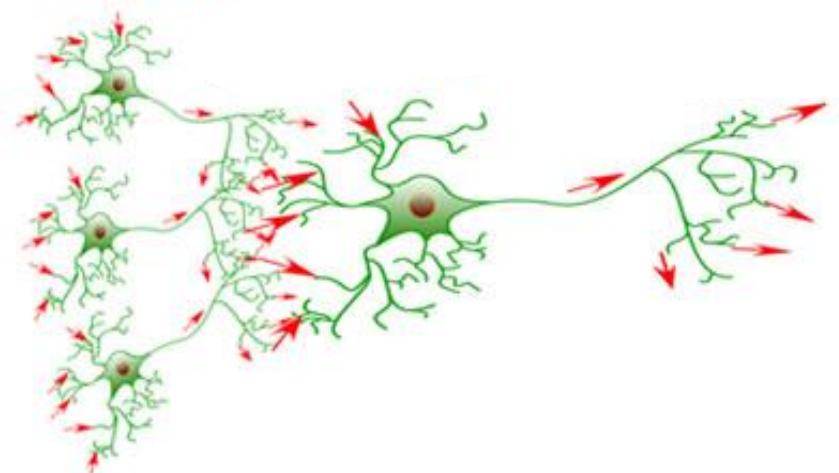
02

팀리닝

02 딥러닝

● 심층 신경망

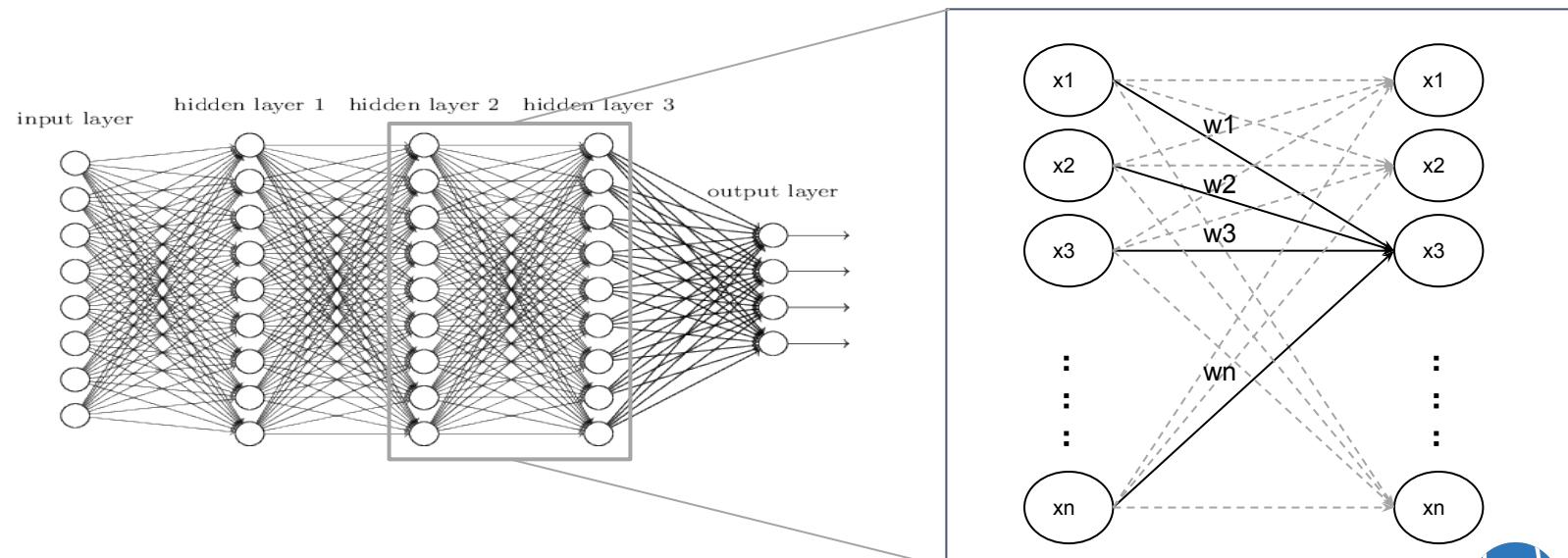
- 머신러닝 알고리즘 중 '인공 신경망'은 뇌의 신경망을 표방하여 만든 알고리즘입니다.
- 인공 신경망은 입력 데이터를 받은 입력층 (input layer)과 출력값을 주는 출력층 (output layer)을 가지고 있으며, 그 사이 은닉층 (hidden layer)을 가질 수도 있습니다.
- 은닉층의 수가 2개 이상인 인공 신경망을 '심층 신경망 (Deep Neural Network)'라고 합니다.



02 딥러닝

- 딥러닝

- 인공 신경망 속 인공 뉴런들은 서로 연결되어 있으며, 각 연결은 고유의 가중치를 가지고 있습니다.
- 뉴런이 다음 뉴런에게 출력값을 전달할 때, 고유의 가중치를 곱하여 전달합니다.
- 입력 데이터에 대해 원하는 출력값을 받을 수 있도록 인공 신경망의 가중치를 조정하는 것을 '학습' 혹은 '훈련'이라고 하며, 심층 신경망을 훈련시키는 것이 '심층 학습 (Deep Learning)'입니다.



02 딥러닝

- 딥러닝에 필요한 개념
 - 손실 함수 (Loss function)
 - 옵티마이저 (Optimizer)
 - 오차역전파 (BackPropagation)
 - 활성화 함수 (Activation function)
- * 설명 도중 언급되는 '모델'은 딥러닝 모델을 뜻합니다.

02 딥러닝

- 손실 함수 (Loss function)

- 심층 신경망의 출력값을 '예측값'이라고 부릅니다.
- 예측값과 정답 사이 오차를 수치화하는 함수를 '손실 함수'라고 부릅니다.
- 딥러닝은 손실 함수의 최솟값을 찾도록 진행됩니다.

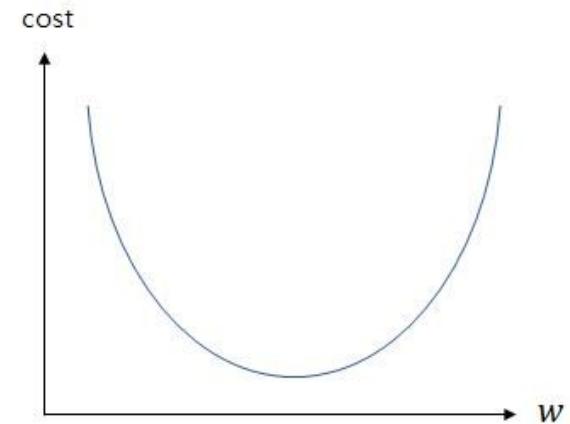
02 딥러닝

● 손실 함수 (Loss function)

- 예시)

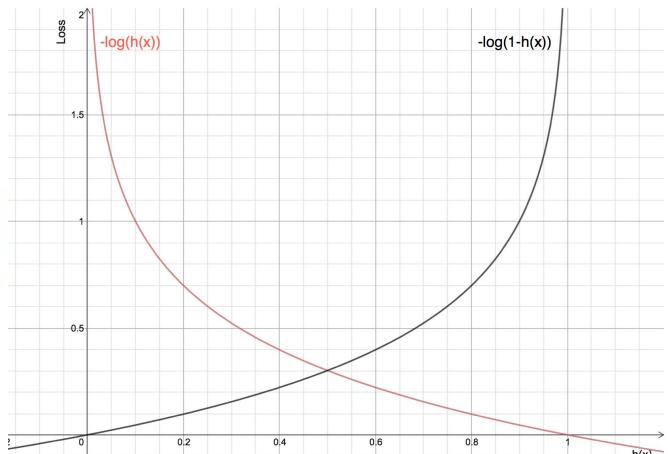
- 평균 제곱 오차

$$cost = \frac{1}{m} \sum_{i=1}^m (H(x^{(i)}) - y^{(i)})^2$$



- 이진 크로스 엔트로피

$$cost = -\frac{1}{m} \sum_{i=1}^m y^{(i)} \log(H(x^{(i)})) + (1 - y^{(i)}) \log(1 - H(x^{(i)}))$$



- * $H(x)$ 는 입력 데이터 x 에 대한 모델 예측값
- * y 는 입력 데이터 x 에 대한 라벨

(이미지 출처 : <https://www.analyticsvidhya.com/>)

02 딥러닝

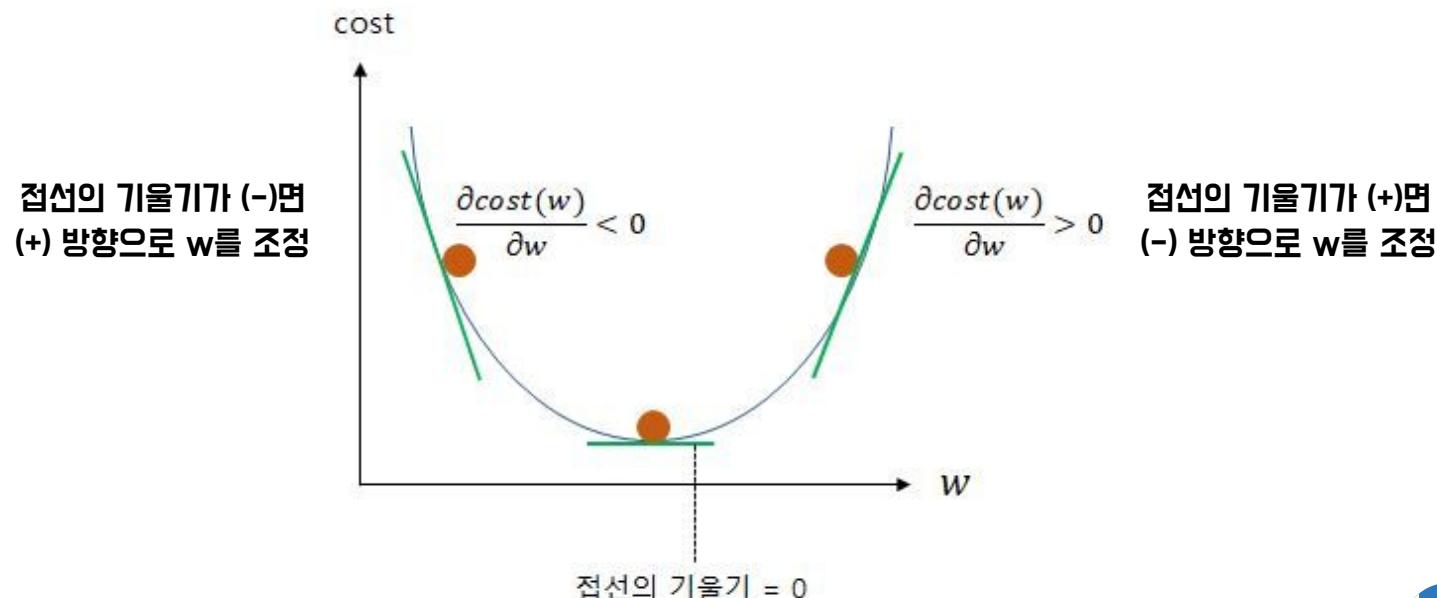
- 옵티마이저 (Optimizer)

- 오차를 최소화하는 문제를 최적화 (Optimization) 문제라고 합니다.
- 최적화 알고리즘을 옵티마이저 (Optimizer)라고 부릅니다.
- 딥러닝은 옵티마이저로 모델의 적절한 가중치를 찾아내는 과정입니다.
- 옵티마이저는 경사 하강법 (Gradient Descent)을 기본으로 합니다.

02 딥러닝

- 옵티마이저 (Optimizer)

- 경사 하강법은 오차의 최소화를 위해 손실 함수의 접선의 기울기를 사용하는 방법입니다.
- 손실 함수의 한 점에서 접선의 기울기 반대 방향으로 가중치 w 를 조정합니다.
- 이를 통해 최적의 가중치를 찾을 수 있습니다.



02 딥러닝

- 옵티마이저 (Optimizer)

- 경사 하강법의 수식

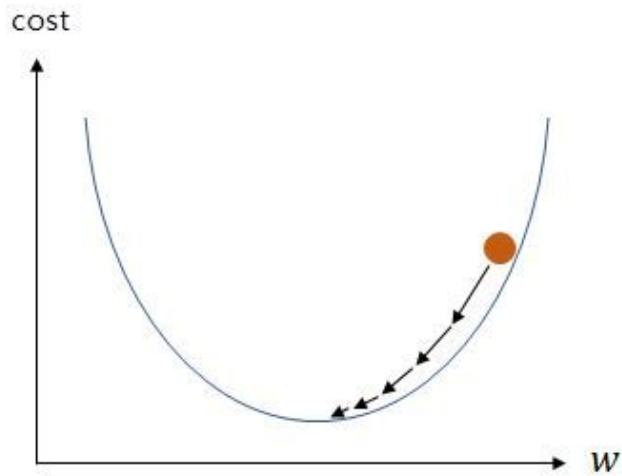
$$w := w - \alpha \frac{\partial}{\partial w} cost(w)$$

- α 는 학습률 (Learning rate)입니다. w 가 업데이트될 때, 얼마나 크게 업데이트될 지 결정합니다.
 - α 는 양수이기 때문에 기울기가 양수이면 w 가 감소하고, 기울기가 음수이면 w 가 증가합니다.

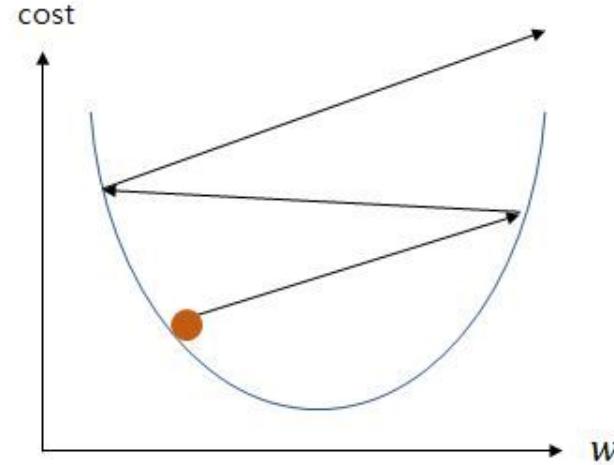
02 딥러닝

- 옵티마이저 (Optimizer)

- 학습률이 너무 높으면 w 가 큰 폭으로 조정되어 손실 함수의 최솟값에 수렴하지 않을 수 있습니다.
- 학습률이 너무 낮으면 모델 학습 속도가 너무 느려질 수 있습니다.



정상적인 수렴



발산 (학습률이 너무 높은 경우)

02 딥러닝

- 옵티마이저 (Optimizer)
 - 옵티마이저와 관련된 딥러닝 용어
 - 에포크 (epoch) : 전체 데이터에 대한 1 번의 훈련
 - 10 에포크 학습시켰다는 뜻은 모델을 전체 데이터로 10번 학습시켰다는 뜻
 - 이터레이션, 스텝 (iteration, step) : 가중치의 업데이트 횟수
 - 10 이터레이션 학습시켰다는 뜻은 모델의 가중치가 10번 업데이트 되었다는 뜻

02 딥러닝

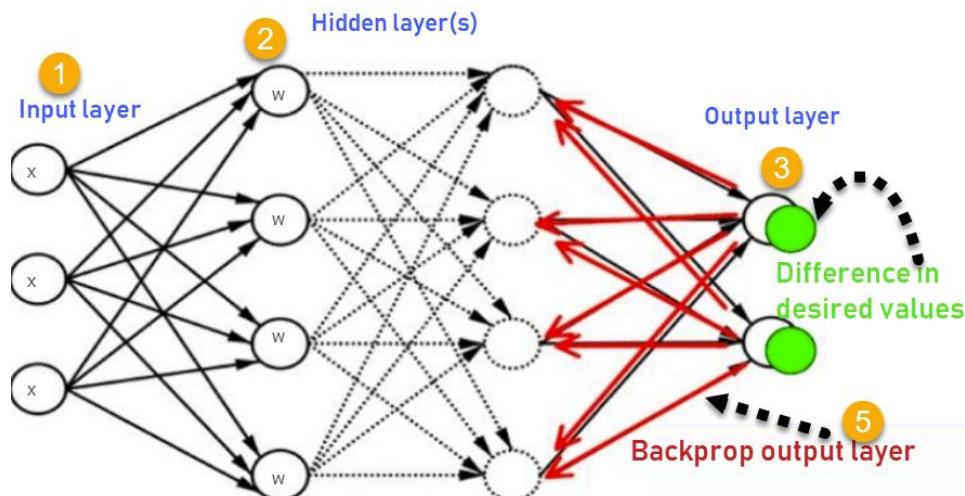
- 오차역전파 (Back Propagation)

- 심층 신경망은 보통 수만개가 넘는 가중치를 가지고 있습니다.
- 가중치 조정을 위해 효율적으로 경사 하강법을 계산해야 합니다.
- 이를 위해 딥러닝에서는 오차역전파를 사용합니다.

02 딥러닝

● 오차역전파 (Back Propagation)

- 순전파 (Forward Propagation)는 입력 데이터가 신경망을 따라 신호를 전파하면서 출력을 만들어가는 과정을 말합니다.
- 반대로 오차역전파를 통한 경사하강법은 순전파의 반대 방향으로 이루어집니다.
- 오차역전파의 진행 과정 예시는 부록을 참고해주시기 바랍니다.



(이미지 출처 : <https://www.guru99.com/>)

02 딥러닝

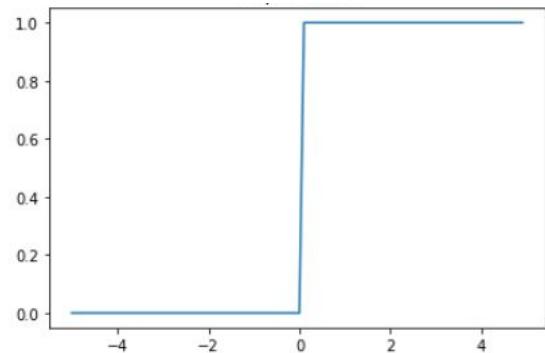
- 활성화 함수 (Activation function)

- 활성화 함수는 인공 신경망에 비선형성을 추가해주는 함수입니다.
- 비선형성을 통해 인공 신경망은 더 복잡한 문제를 풀 수 있습니다.
- 활성화 함수 역시 오차역전파 과정에서 미분됩니다. 따라서, 활성화 함수의 수식과 미분 형태는 학습 성능에 큰 영향을 끼칩니다.

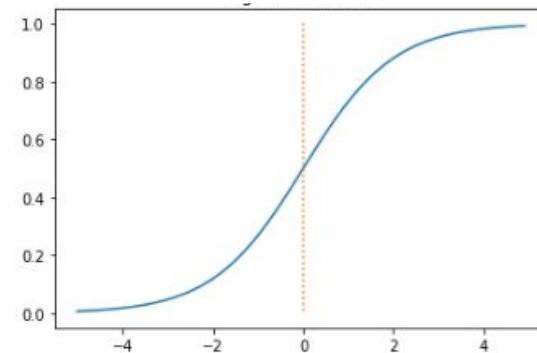
02 딥러닝

● 활성화 함수 (Activation function)

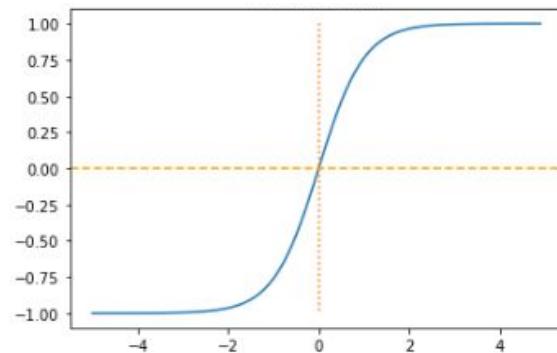
○ 예시)



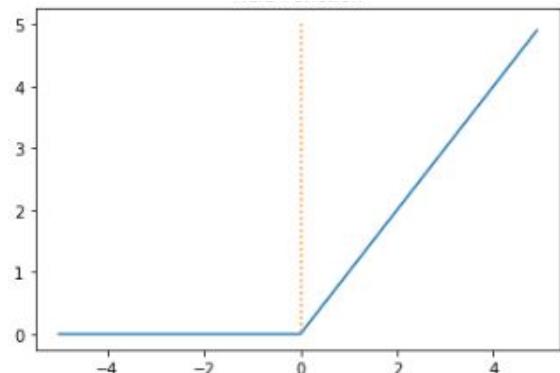
step function



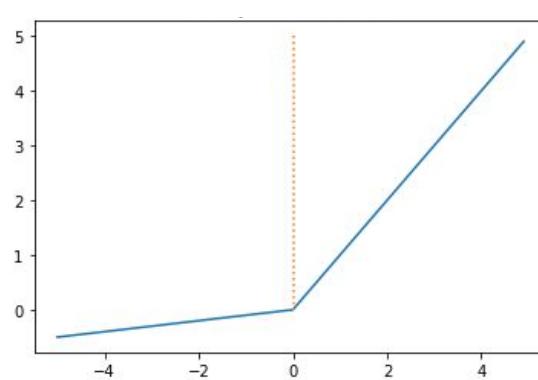
sigmoid function



hypertangent function



ReLU function



Leaky ReLU function

03

액체 검출 모델

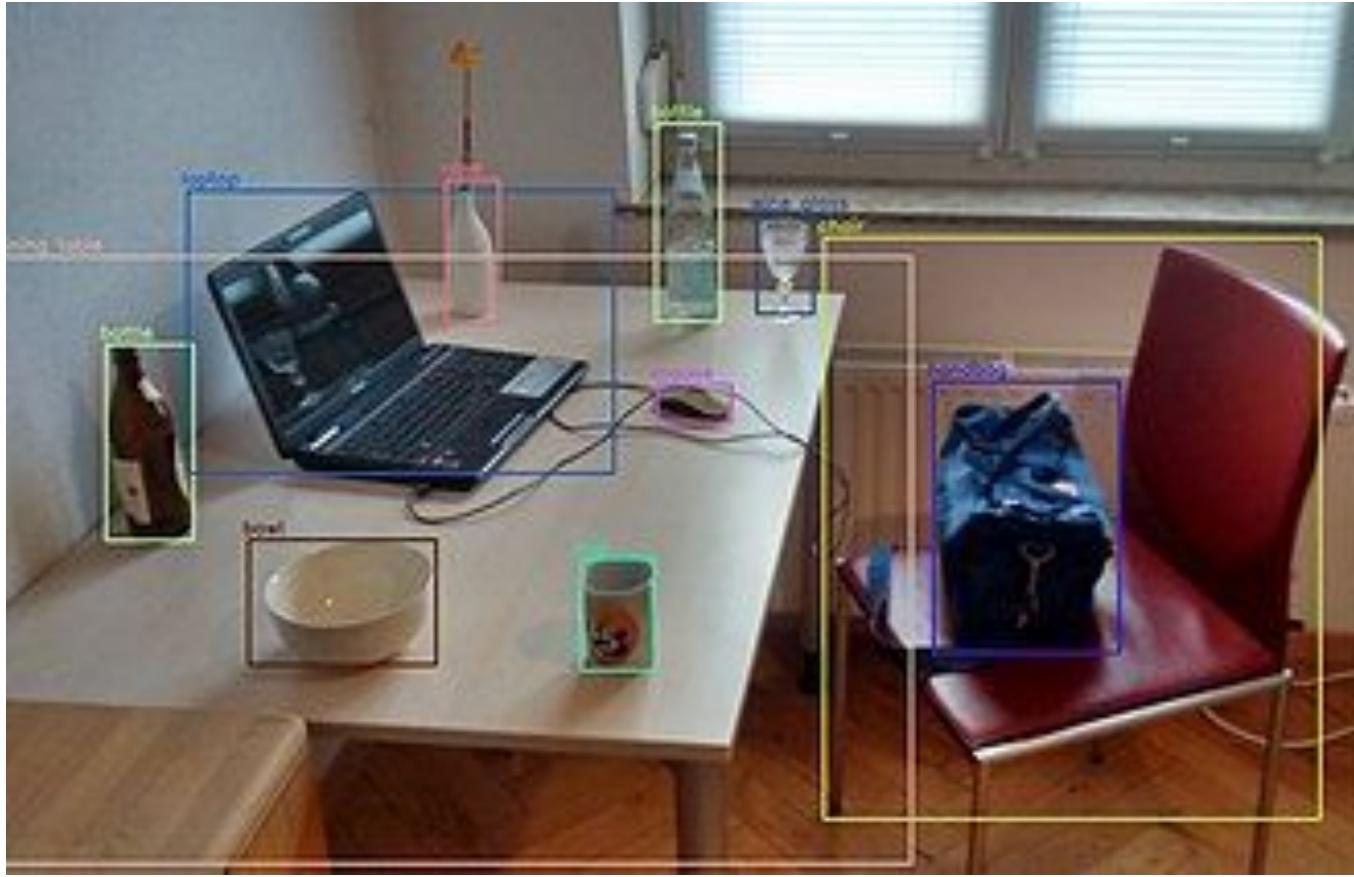
03 객체 검출 모델

- **객체 검출 (Object Detection)**
 - 컴퓨터 비전 분야의 딥러닝은 크게 두 가지의 목표를 가지고 있습니다.
 - **분류 (Image Classification)** : 이미지 속 물체의 종류 (Class)를 예측하는 문제
 - **위치 추정 (Image Localization)** : 이미지 속 물체의 위치를 Bounding Box라고 불리는 사각형을 통해 예측하는 문제
 - **객체 검출 (Object Detection)**이란 이미지 내 존재하는 다양한 객체의 위치와 종류를 동시에 예측하는 문제를 말합니다.

03 객체 검출 모델

- 객체 검출 (Object Detection)

- 예시)



03 객체 검출 모델

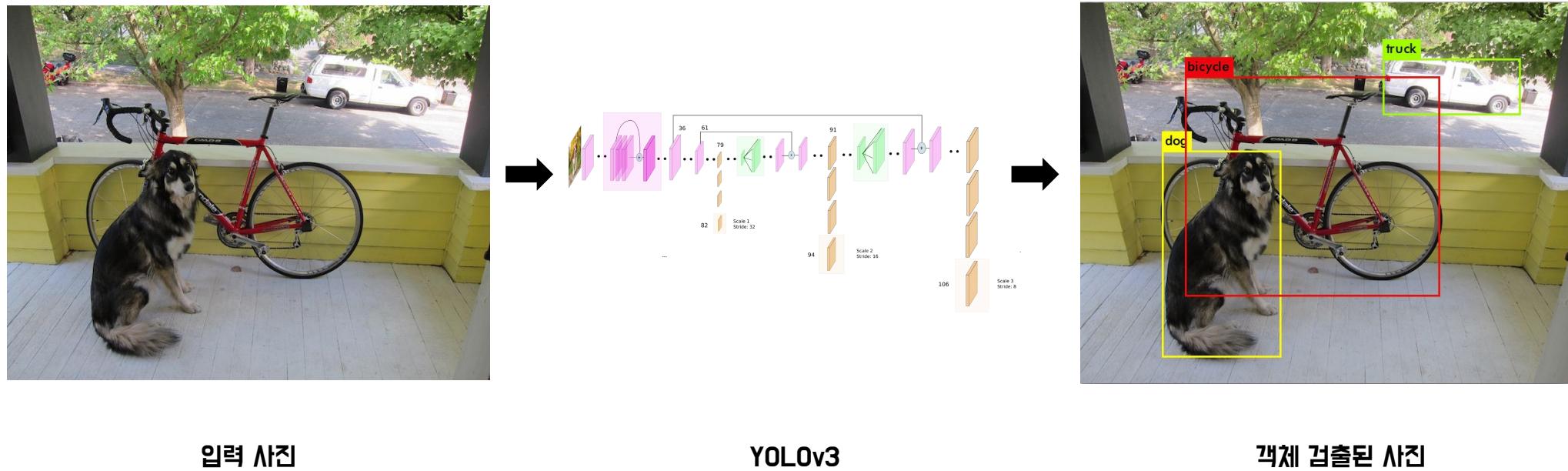
- YOLOv3 모델

- YOLOv3는 대표적인 객체 검출 딥러닝 모델입니다.
- YOLOv3는 객체의 위치 추정과 분류를 한 번에 계산합니다.
- YOLOv3는 다음 정보들을 예측합니다.
 - Bounding Box 정보 (중심좌표 x, y, 높이와 너비 w, h)
 - 객체 존재 확률
 - 객체의 Class 확률 (객체가 어느 class에 속할지)

03 객체 검출 모델

- YOLOv3 모델의 객체 검출

- YOLOv3로 예측한 정보 중 객체 존재 확률이 미리 설정한 기준치를 넘어가면 객체를 검출합니다.
- 예측한 Bounding Box로 객체의 위치를 추정하고, Class 확률이 가장 높은 Class로 분류합니다.



입력 사진

YOLOv3

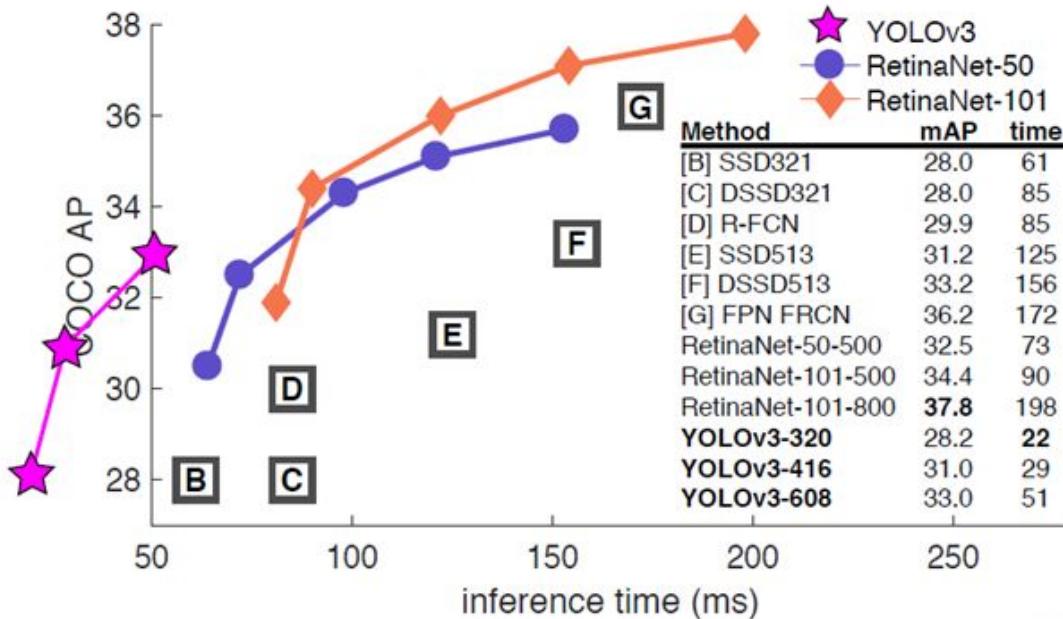
객체 검출된 사진

(이미지 출처 : <https://blog.paperspace.com/>)

03 객체 검출 모델

- YOLOv3 모델의 성능

- YOLOv3는 당시 최신 모델에 비해 준수한 정확도에 빠른 속도를 보였습니다.
- 이 빠른 속도로 인해 YOLOv3는 객체 검출 분야의 대표적인 모델이 되었습니다.



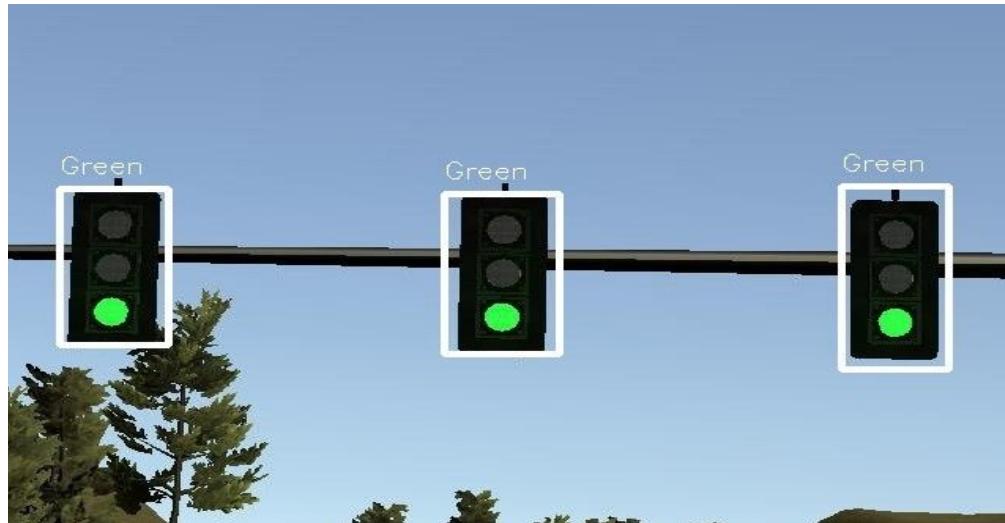
YOLOv3와 타 모델간 성능 비교

(이미지 출처 : <https://pjreddie.com/darknet/yolo/>)

03 객체 검출 모델

- YOLOv3 커스텀 데이터셋 훈련

- 커스텀 데이터셋으로 YOLOv3를 훈련하면 각자 원하는 객체를 사진 상에서 검출할 수 있습니다.
- 예시)



신호등 불빛 검출



마스크 착용 여부
검출

04 출처

1. Andrew Ng. Machine Learning.

<https://www.coursera.org/learn/machine-learning>

2. 딥 러닝을 이용한 자연어 처리 입문. <https://wikidocs.net/book/2155>

3. Darknet 공식 홈페이지. <https://pjreddie.com/darknet/yolo/>

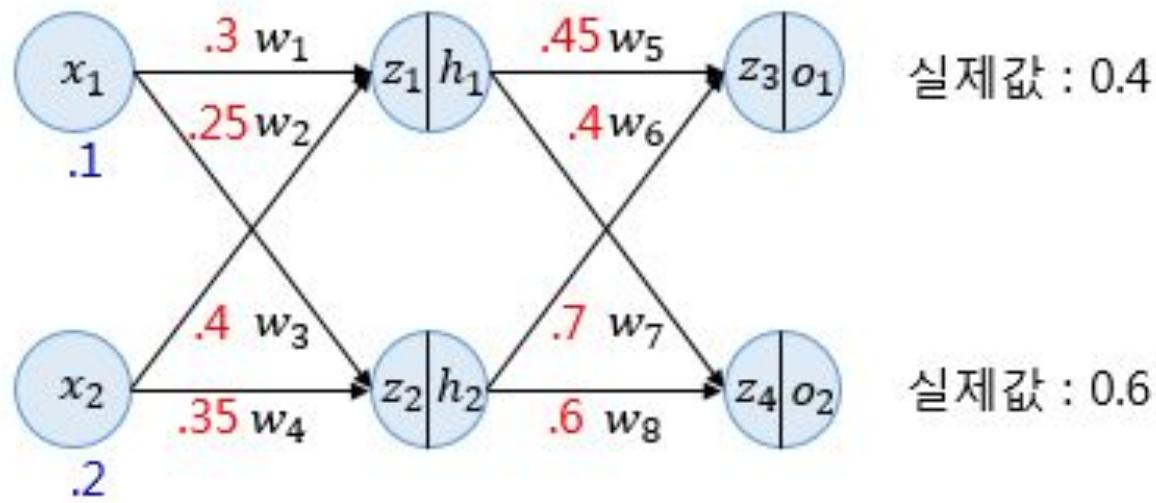
4. NVIDIA 공식 블로그. <https://blogs.nvidia.co.kr/>

5. YOLOv3 : <https://arxiv.org/pdf/1804.02767.pdf>

05 부록

부록 오차 역전파 (Back Propagation)

- 오차역전파 과정을 통해 가중치가 어떻게 업데이트 되는지 살펴보겠습니다.
- 아래는 예시로 사용할 인공 신경망입니다. 입력층, 은닉층, 출력층, 총 3개의 층으로 이루어져 있으며, 각 층은 두 개의 뉴런으로 이루어져 있습니다.
- 변수 z 는 이전 층의 출력값과 가중치의 가중합을 의미합니다. 이 가중합이 활성화 함수를 거친 값이 h 와 o 입니다.

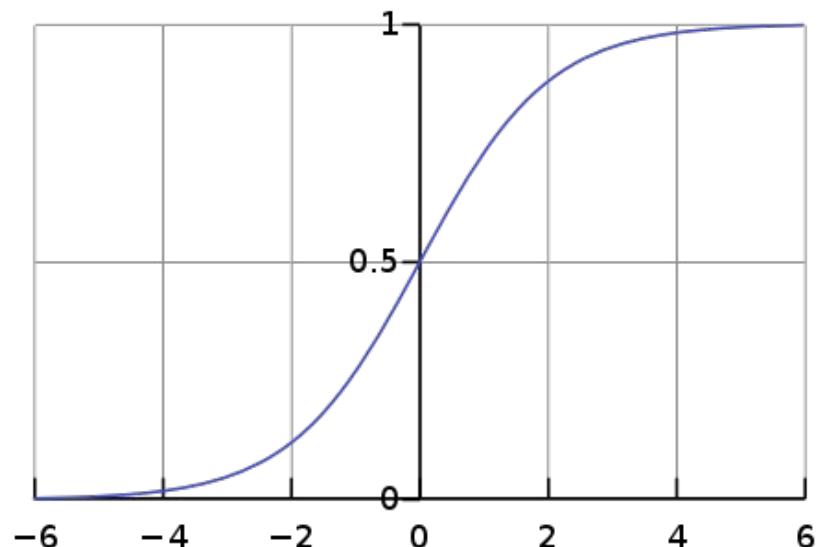


부록 오차 역전파 (Back Propagation)

- 활성화 함수는 시그모이드 함수입니다.

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

$$\frac{\partial f(x)}{\partial x} = f(x)(1 - f(x))$$



부록 오차 역전파 (Back Propagation)

1. 순전파

a. z_1, z_2 의 값 구하기

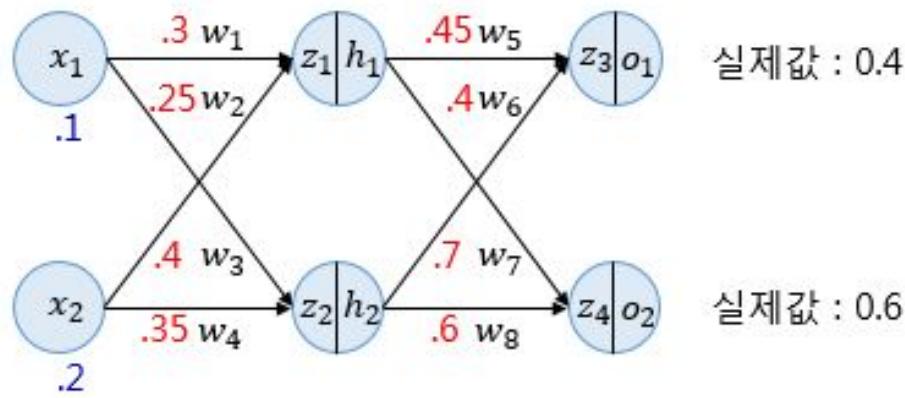
$$z_1 = w_1x_1 + w_2x_2 = 0.3 \times 0.1 + 0.25 \times 0.2 = 0.08$$

$$z_2 = w_3x_1 + w_4x_2 = 0.4 \times 0.1 + 0.35 \times 0.2 = 0.11$$

b. 시그모이드 함수를 적용하여 h_1, h_2 의 값 구하기

$$h_1 = \text{sigmoid}(z_1) = \frac{1}{1 + e^{-z_1}} = \frac{1}{1 + e^{-0.08}} = 0.51998934$$

$$h_2 = \text{sigmoid}(z_2) = \frac{1}{1 + e^{-z_2}} = \frac{1}{1 + e^{-0.11}} = 0.52747230$$



부록 오차 역전파 (Back Propagation)

c. z_3, z_4 의 값 구하기

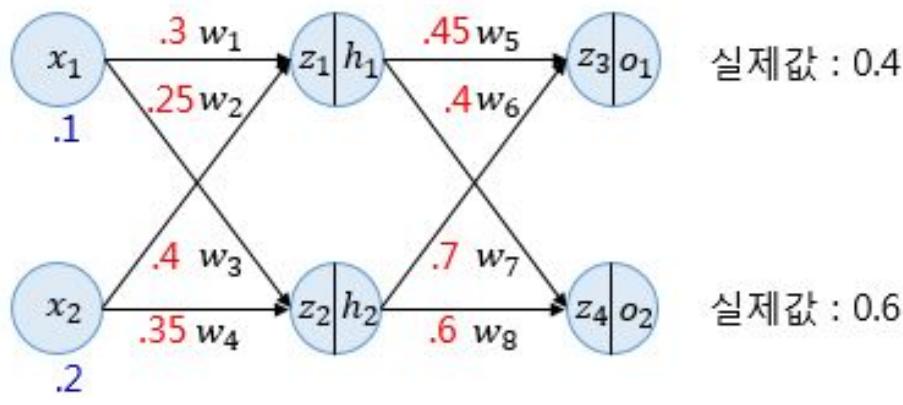
$$z_3 = w_5 h_1 + w_6 h_2 = 0.45 \times h_1 + 0.4 \times h_2 = 0.44498412$$

$$z_4 = w_7 h_1 + w_8 h_2 = 0.7 \times h_1 + 0.6 \times h_2 = 0.68047592$$

d. 시그모이드 함수를 적용하여 o_1, o_2 의 값 구하기

$$o_1 = \text{sigmoid}(z_3) = 0.60944600$$

$$o_2 = \text{sigmoid}(z_4) = 0.66384491$$



부록 오차 역전파 (Back Propagation)

- e. 예측값과 라벨의 오차를 구합니다. 오차는 평균 제곱 오차를 사용하겠습니다.
- f. o1, o2에 대한 라벨을 target o1, o2라 하고, o1, o2의 오차를 Eo1, Eo2라 하겠습니다.

$$E_{o1} = \frac{1}{2}(\text{target}_{o1} - \text{output}_{o1})^2 = 0.02193381$$

$$E_{o2} = \frac{1}{2}(\text{target}_{o2} - \text{output}_{o2})^2 = 0.00203809$$

$$E_{total} = E_{o1} + E_{o2} = 0.02397190$$

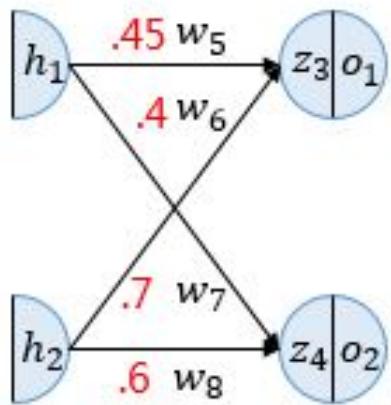
- g. 출력층 바로 이전의 은닉층이 N층이라고 할 때, 출력층과 N층 사이의 가중치를 업데이트하는 단계를 1단계, 그리고 N층과 N-1층 사이의 가중치를 업데이트하는 단계를 2단계라고 해봅시다.

부록 오차 역전파 (Back Propagation)

2. 역전파 1단계

- 먼저 업데이트해야 할 가중치는 w_5, w_6, w_7, w_8 입니다. w_5 를 예시로 들겠습니다.
- w_5 를 업데이트하기 위해서는 $\frac{\partial E_{total}}{\partial w_5}$ 를 계산해야 합니다.
- 미분의 연쇄법칙에 의해 $\frac{\partial E_{total}}{\partial w_5}$ 는 다음과 같이 계산합니다.

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial o_1} \times \frac{\partial o_1}{\partial z_3} \times \frac{\partial z_3}{\partial w_5}$$



부록 오차 역전파 (Back Propagation)

d. $\frac{\partial E_{total}}{\partial o_1}$ 의 계산은 다음과 같습니다.

$$\text{i. } E_{total} = \frac{1}{2}(target_{o1} - output_{o1})^2 + \frac{1}{2}(target_{o2} - output_{o2})^2$$

$$\text{ii. } \frac{\partial E_{total}}{\partial o_1} = 2 \times \frac{1}{2}(target_{o1} - output_{o1})^{2-1} \times (-1) + 0$$

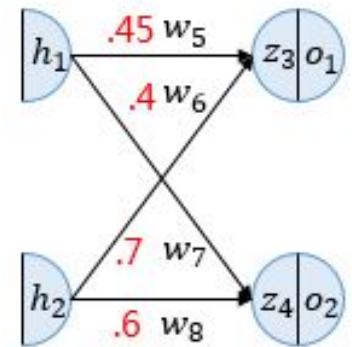
$$\text{iii. } \frac{\partial E_{total}}{\partial o_1} = -(target_{o1} - output_{o1}) = -(0.4 - 0.60944600) = 0.20944600$$

e. 다음 $\frac{\partial o_1}{\partial z_3}$ 의 계산입니다.

i. o1은 z3가 시그모이드 함수를 거친 뒤의 결과값입니다.

ii. 시그모이드 함수의 미분은 $f(x) \times (1 - f(x))$ 입니다.

$$\text{iii. } \frac{\partial o_1}{\partial z_3} = o_1 \times (1 - o_1) = 0.60944600(1 - 0.60944600) = 0.23802157$$



부록 오차 역전파 (Back Propagation)

f. 마지막 $\frac{\partial z_3}{\partial w_5}$ 의 계산입니다.

i. $z_3 = w_5 h_1 + w_6 h_2$

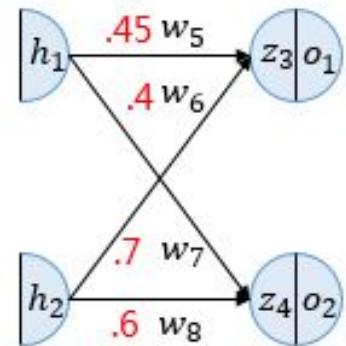
ii. $\frac{\partial z_3}{\partial w_5} = h_1 = 0.51998934$

g. 미분에 필요한 모든 값을 계산했습니다.

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial o_1} \frac{\partial o_1}{\partial z_3} \frac{\partial z_3}{\partial w_5} = 0.20944600 \times 0.23802157 \times 0.51998934 = 0.02592286$$

h. 경사 하강법을 이용하여 가중치를 업데이트합니다. 학습률은 0.5라고 가정합니다.

$$w_5^+ = w_5 - \alpha \frac{\partial E_{total}}{\partial w_5} = 0.45 - 0.5 \times 0.02592286 = 0.43703857$$



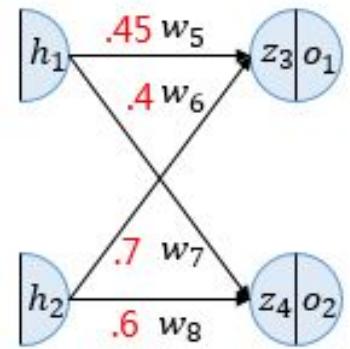
부록 오차 역전파 (Back Propagation)

- i. 같은 방식으로 w_6 , w_7 , w_8 을 업데이트합니다.

$$\frac{\partial E_{total}}{\partial w_6} = \frac{\partial E_{total}}{\partial o_1} \times \frac{\partial o_1}{\partial z_3} \times \frac{\partial z_3}{\partial w_6} \rightarrow w_6^+ = 0.38685205$$

$$\frac{\partial E_{total}}{\partial w_7} = \frac{\partial E_{total}}{\partial o_2} \times \frac{\partial o_2}{\partial z_4} \times \frac{\partial z_4}{\partial w_7} \rightarrow w_7^+ = 0.69629578$$

$$\frac{\partial E_{total}}{\partial w_8} = \frac{\partial E_{total}}{\partial o_2} \times \frac{\partial o_2}{\partial z_4} \times \frac{\partial z_4}{\partial w_8} \rightarrow w_8^+ = 0.59624247$$



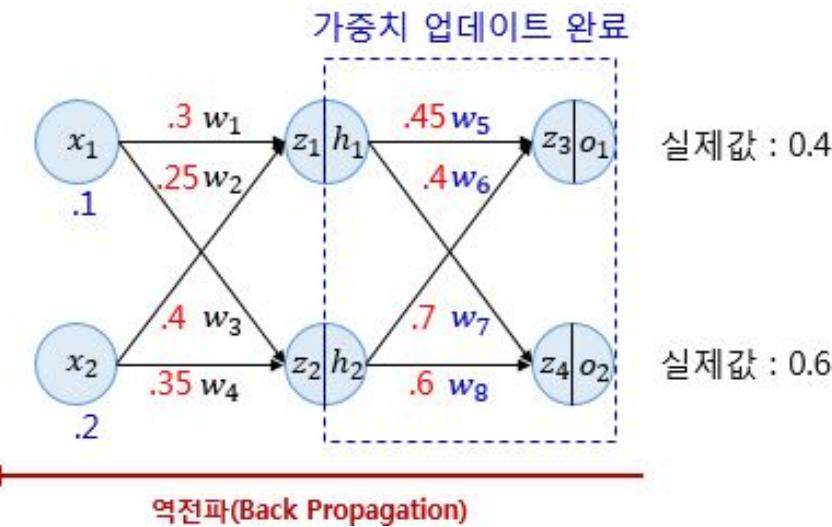
부록 오차 역전파 (Back Propagation)

3. 역전파 2단계

- a. 역전파 2단계에서 업데이트해야 할 가중치는 w_1, w_2, w_3, w_4 입니다. w_1 의 업데이트를 예시로 들겠습니다.

b. $\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial h_1} \times \frac{\partial h_1}{\partial z_1} \times \frac{\partial z_1}{\partial w_1}$

c. $\frac{\partial E_{total}}{\partial h_1} = \frac{\partial E_{o1}}{\partial h_1} + \frac{\partial E_{o2}}{\partial h_1}$



부록 오차 역전파 (Back Propagation)

d. $\frac{\partial E_{total}}{\partial h_1}$ 를 먼저 구해보겠습니다.

$$\begin{aligned} \text{i. } \frac{\partial E_{o1}}{\partial h_1} &= \frac{\partial E_{o1}}{\partial z_3} \times \frac{\partial z_3}{\partial h_1} = \frac{\partial E_{o1}}{\partial o_1} \times \frac{\partial o_1}{\partial z_3} \times \frac{\partial z_3}{\partial h_1} \\ &= -(target_{o1} - output_{o1}) \times o_1 \times (1 - o_1) \times w_5 \\ &= 0.20944600 \times 0.23802157 \times 0.45 = 0.02243370 \end{aligned}$$

$$\text{ii. } \frac{\partial E_{o2}}{\partial h_1} = \frac{\partial E_{o2}}{\partial z_4} \times \frac{\partial z_4}{\partial h_1} = \frac{\partial E_{o2}}{\partial o_2} \times \frac{\partial o_2}{\partial z_4} \times \frac{\partial z_4}{\partial h_1} = 0.00997311$$

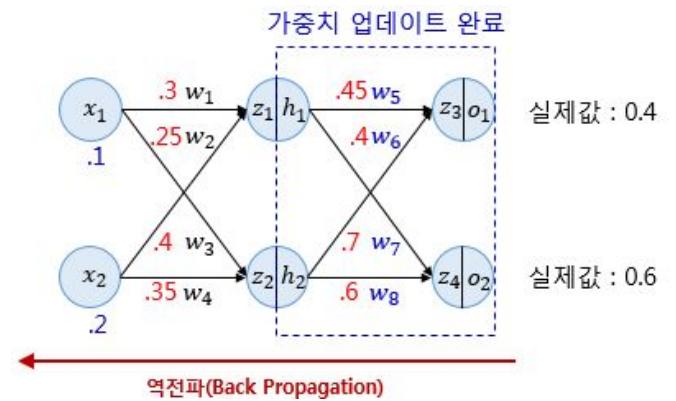
e. $\frac{\partial h_1}{\partial z_1}, \frac{\partial z_1}{\partial w_1}$ 를 계산합니다.

$$\text{i. } \frac{\partial h_1}{\partial z_1} = h_1 \times (1 - h_1) = 0.51998934(1 - 0.51998934) = 0.24960043$$

$$\text{ii. } \frac{\partial z_1}{\partial w_1} = x_1 = 0.1$$

f. $\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial h_1} \times \frac{\partial h_1}{\partial z_1} \times \frac{\partial z_1}{\partial w_1}$ 를 계산합니다.

$$\frac{\partial E_{total}}{\partial w_1} = 0.03240681 \times 0.24960043 \times 0.1 = 0.00080888$$



부록 오차 역전파 (Back Propagation)

g. 경사 하강법을 이용하여 가중치를 업데이트합니다.

$$w_1^+ = w_1 - \alpha \frac{\partial E_{total}}{\partial w_1} = 0.3 - 0.5 \times 0.00080888 = 0.29959556$$

h. 같은 방식으로 w_2, w_3, w_4 를 업데이트합니다.

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial E_{total}}{\partial h_1} \times \frac{\partial h_1}{\partial z_1} \times \frac{\partial z_1}{\partial w_2} \rightarrow w_2^+ = 0.24919112$$

$$\frac{\partial E_{total}}{\partial w_3} = \frac{\partial E_{total}}{\partial h_2} \times \frac{\partial h_2}{\partial z_2} \times \frac{\partial z_2}{\partial w_3} \rightarrow w_3^+ = 0.39964496$$

$$\frac{\partial E_{total}}{\partial w_4} = \frac{\partial E_{total}}{\partial h_2} \times \frac{\partial h_2}{\partial z_2} \times \frac{\partial z_2}{\partial w_4} \rightarrow w_4^+ = 0.34928991$$



부록 오차 역전파 (Back Propagation)

4. 업데이트된 가중치에 대해서 오차 확인

a. 순전파 진행

$$z_1 = w_1x_1 + w_2x_2 = 0.29959556 \times 0.1 + 0.24919112 \times 0.2 = 0.07979778$$

$$z_2 = w_3x_1 + w_4x_2 = 0.39964496 \times 0.1 + 0.34928991 \times 0.2 = 0.10982248$$

$$h_1 = \text{sigmoid}(z_1) = 0.51993887$$

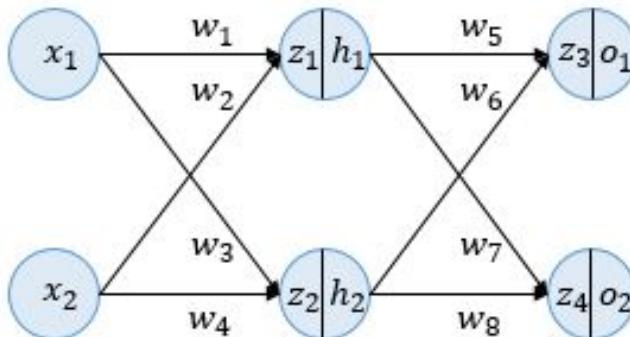
$$h_2 = \text{sigmoid}(z_2) = 0.52742806$$

$$z_3 = w_5h_1 + w_6h_2 = 0.43703857 \times h_1 + 0.38685205 \times h_2 = 0.43126996$$

$$z_4 = w_7h_1 + w_8h_2 = 0.69629578 \times h_1 + 0.59624247 \times h_2 = 0.67650625$$

$$o_1 = \text{sigmoid}(z_3) = 0.60617688$$

$$o_2 = \text{sigmoid}(z_4) = 0.66295848$$



부록 오차 역전파 (Back Propagation)

b. 오차 확인

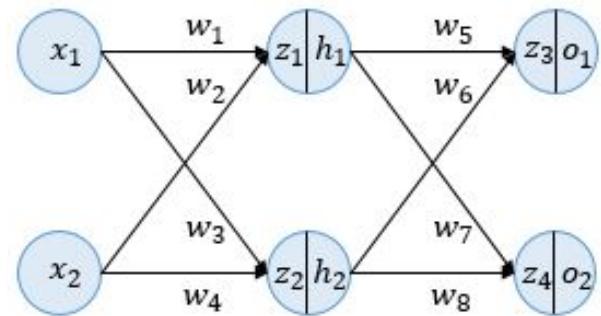
$$E_{o1} = \frac{1}{2} (\text{target}_{o1} - \text{output}_{o1})^2 = 0.02125445$$

$$E_{o2} = \frac{1}{2} (\text{target}_{o2} - \text{output}_{o2})^2 = 0.00198189$$

$$E_{total} = E_{o1} + E_{o2} = 0.02323634$$

c. 기준 오차 : 0.02397190. 현재 오차 : 0.02323634

d. 오차 역전파를 통해 오차가 감소한 것을 확인할 수 있습니다.





WeGo Robotics

Tel. 031 – 229 – 3553



Fax. 031 – 229 – 3554



제품 문의: go.sales@wego-robotics.com

기술 문의: go.support@wego-robotics.com