

Dong-A Univ. (ISPL)



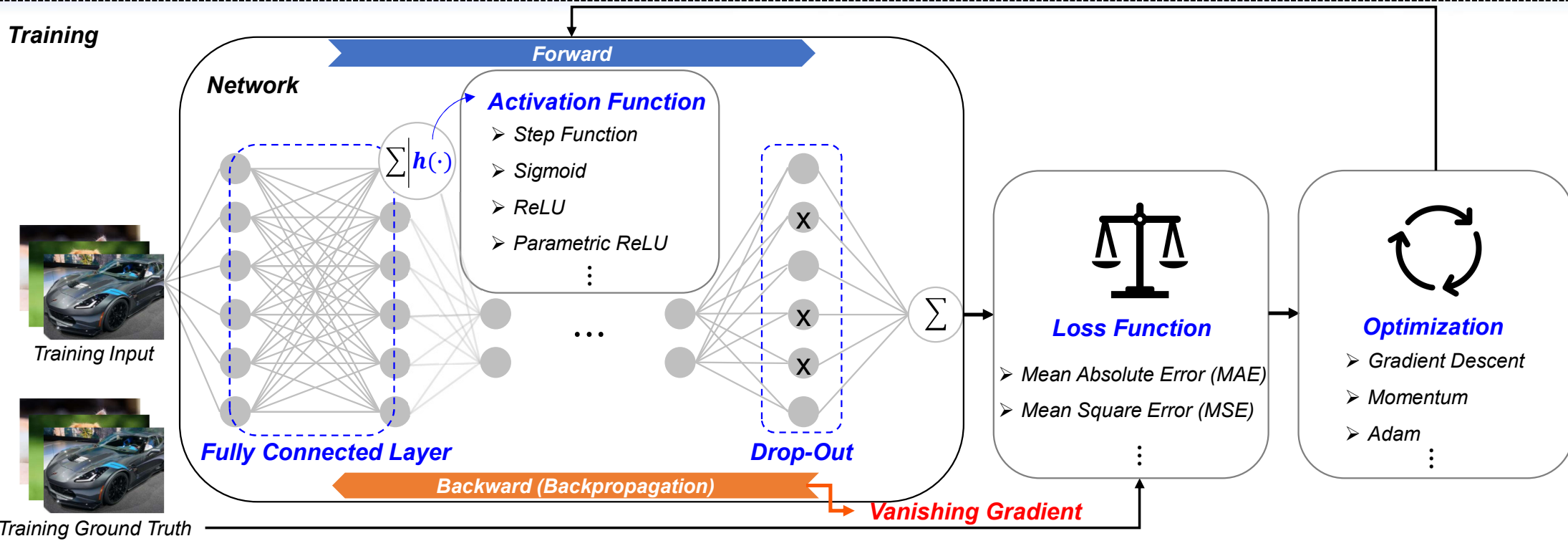
동아대학교
DONG-A UNIVERSITY

Perceptron

컴퓨터공학부 AI학과
2024년 1학기 인공지능

Overall Architecture of Deep Learning

Training

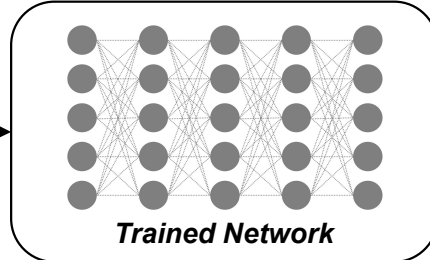


Overfitting

Test



Test Input



Trained Network

Evaluation Metric

- PSNR
- SSIM
- Total Memory
- ⋮

- ReLU: Rectified Linear Unit
- Adam: Adaptive Moment Estimation
- PSNR: Peak Signal-to-Noise Ratio
- SSIM: Structural Similarity Index Measure

Contents

1. Introduction

2. Applications

- ① Regression
- ② Classification
- ③ Logic gate

3. Multi-layer Perceptron (MLP)



Introduction

▪ Dartmouth Conference (1956년)

- 목적: 지능을 가진 기계 연구 → 인공지능 (Artificial Intelligence)
- 주요 성과: 초기 인공지능의 개념 및 분야 확립



John MacCarthy



Marvin Minsky



Claude Shannon



Ray Solomonoff



Alan Newell



Herbert Simon



Arthur Samuel



Oliver Selfridge



Nathaniel Rochester



Trenchard More

Dartmouth 학술회의 참여 연구자



Dartmouth 학술회의 개최지
(미국 뉴햄프셔 Dartmouth 대학)

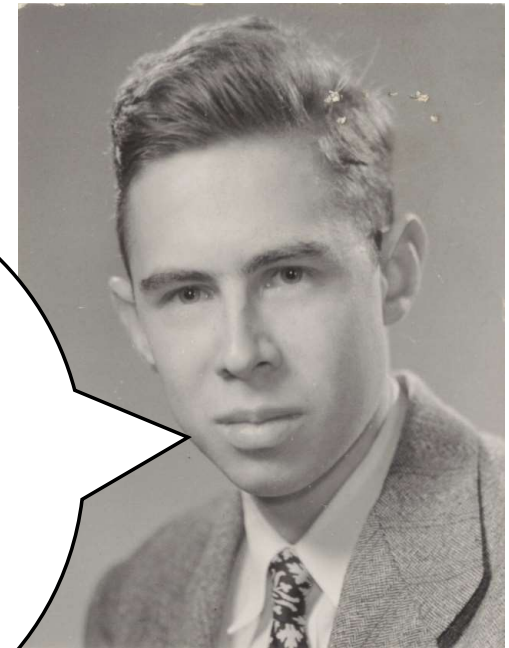
Introduction

- 기호주의 인공지능 vs. 연결주의 인공지능
 - 두 연구자를 중심으로 서로 다른 인공지능 학습 방법을 주장



인공지능은 컴퓨터
작동 방식에 가까운
기호 관계를 정의해
학습해야 한다.

<Marvin Minsky>



실제 뇌가 정보를
처리 하는 방식에
가까운 인공 신경망
으로 학습해야 한다.

<Frank Rosenblatt>

Introduction

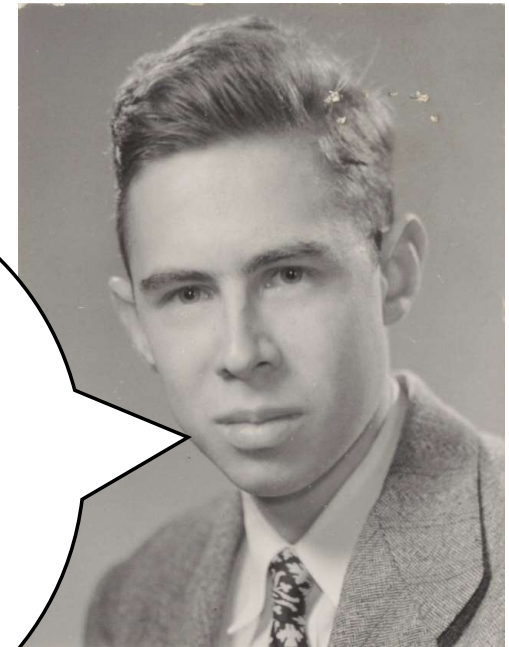
- 기호주의 인공지능 vs. 연결주의 인공지능
 - 두 연구자를 중심으로 서로 다른 인공지능 학습 방법을 주장



인공지능은 컴퓨터
작동 방식에 가까운
기호 관계를 정의해
학습해야 한다.

기호주의 인공지능

<Marvin Minsky>



실제 뇌가 정보를
처리 하는 방식에
가까운 인공 신경망
으로 학습해야 한다.

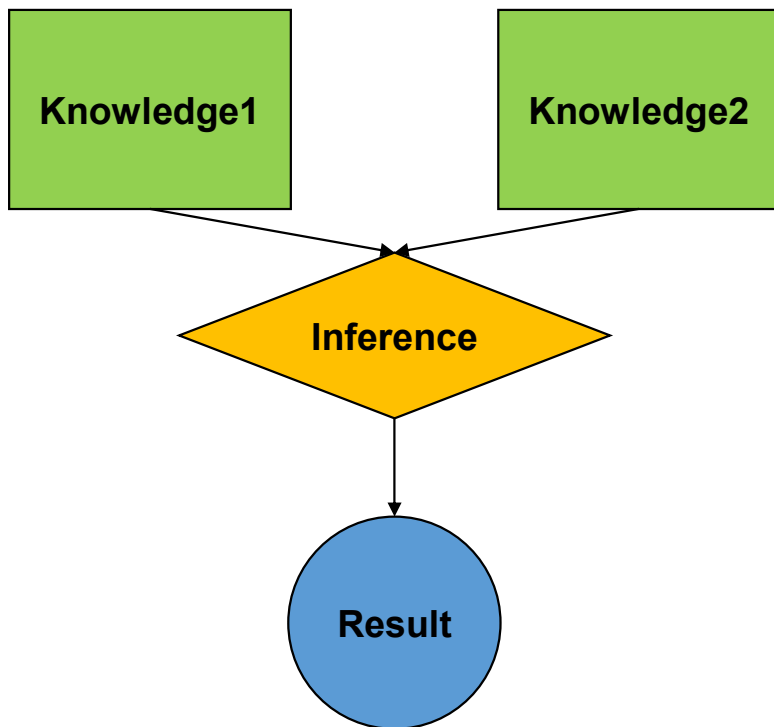
연결주의 인공지능

<Frank Rosenblatt>

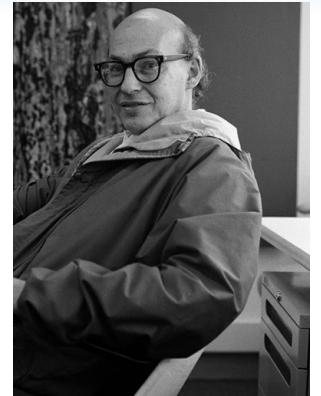
Introduction

■ 기호주의 인공지능 (Symbolism AI)

- 컴퓨터의 작동 방식에 맞추어 **규칙 기반 (Rule-based)**으로 인공지능을 학습
- (1) 인간의 지식을 기호화 → (2) 기호 간 관계 정의 → (3) 연역적 추론



- Knowledge1: 사람은 죽는다.
- Knowledge2: 소크라테스는 사람이다.
- Inference: 소크라테스는 죽는가?
- Result: 죽는다.



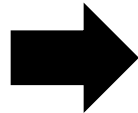
<Marvin Minsky>

Introduction

■ 기호주의 인공지능의 한계

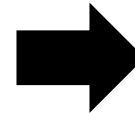
- 현실 세계의 모든 형상/규칙/개념을 기호화 하는 것은 불가능함
- Ex. 기호주의 기법으로 고양이를 판별하는 인공지능

[1] Training



20	100
52	173

픽셀 값 예시



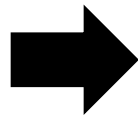
Cat = [20, 100, 52, 173]

Introduction

■ 기호주의 인공지능의 한계

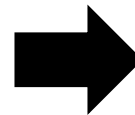
- 현실 세계의 모든 형상/규칙/개념을 기호화 하는 것은 불가능함
- Ex. 기호주의 기법으로 고양이를 판별하는 인공지능

[1] Training



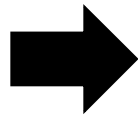
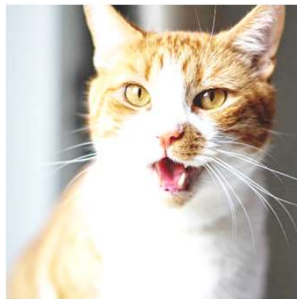
20	100
52	173

픽셀 값 예시



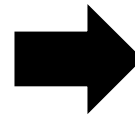
Cat = [20, 100, 52, 173]

[2] Inference



120	200
152	255

픽셀 값 예시



[120, 200, 152, 255] is Cat? **No**

Introduction

■ 기호주의 인공지능의 한계

- 현실 세계의 모든 형상/규칙/개념을 기호화 하는 것은 불가능함
- Ex. 기호주의 기법으로 고양이를 판별하는 인공지능
→ 밝기, 자세, 품종, 크기 등 모든 자연 형상을 고려하지 못함



Challenge(1): Illumination



Challenge(3): Background Clutter



Challenge(2): Deformation

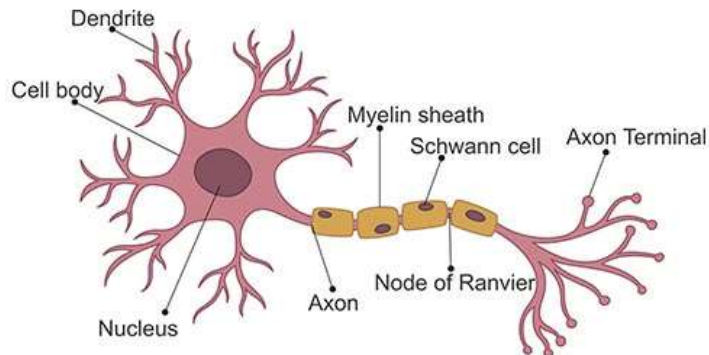


Challenge(4): Intraclass Variation

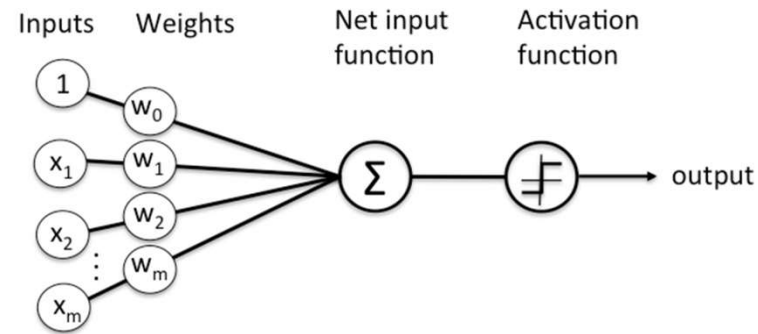
Introduction

■ 연결주의 인공지능 (Connection AI)

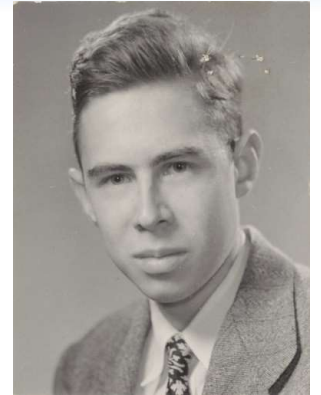
- 인간이 학습하는 방법을 구현/모방해 신경망 (Neural Network) 기반으로 인공지능을 학습
- 대표적인 알고리즘으로 perceptron이 존재함 (1958)



신경 세포 (Neuron)



인공 신경망 예시 (Perceptron)

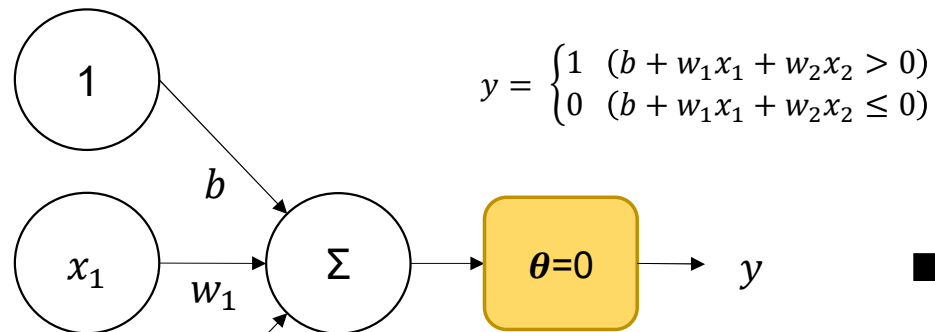


<Frank Rosenblatt>

Introduction

Perceptron 출력 과정

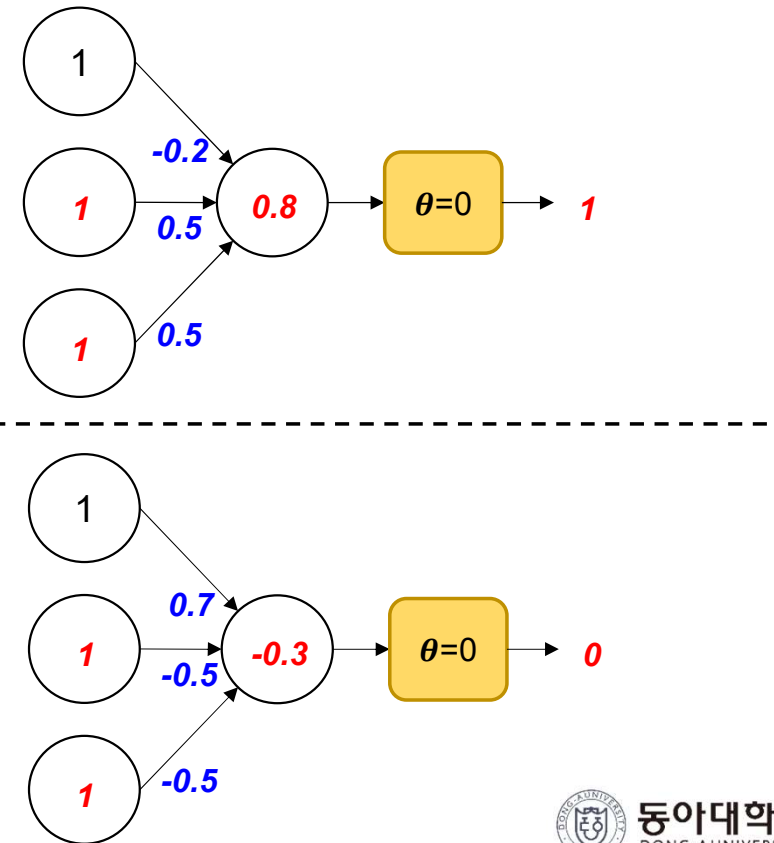
1. 입력신호와 가중치 값을 곱해준 뒤 편향 값을 더함
2. 더한 값이 임계 값을 넘으면 1 그렇지 않으면 0을 출력



- x_1, x_2 : 입력값
- w_1, w_2 : 가중치
- b : 편향
- θ : 임계값 (0으로 설정)
- y : 출력값

<Perceptron>

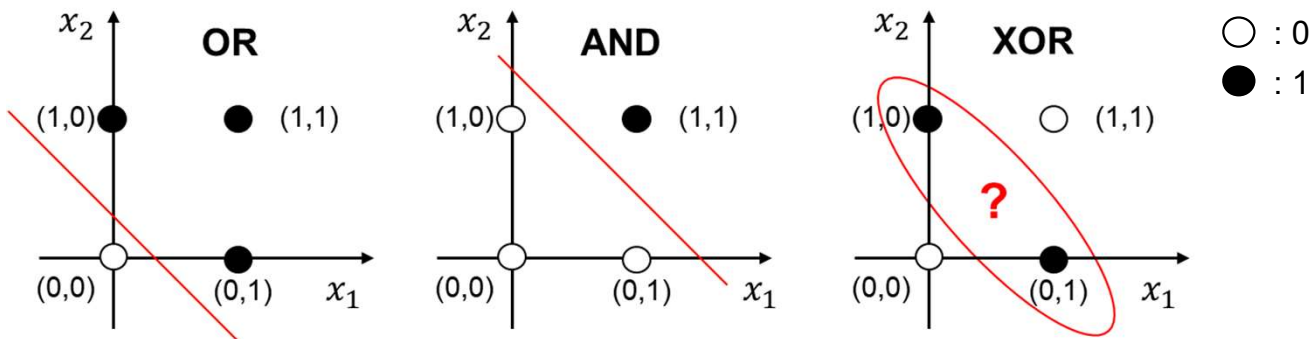
각 가중치 값과 편향 값을 통해 출력 값 조정



Introduction

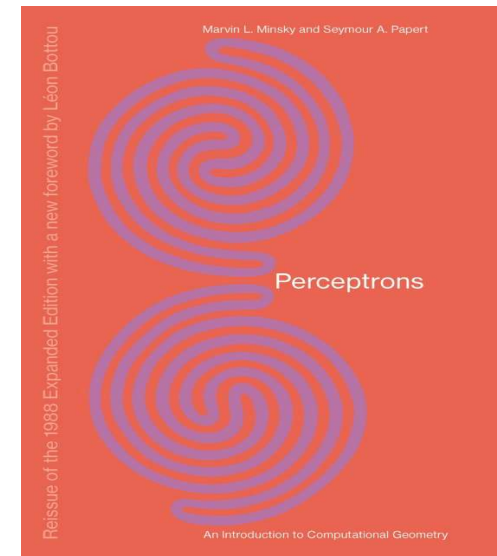
■ XOR 문제와 AI winter (1969 ~ 1985)

- 기호주의 인공지능을 지지하던 M. Minsky는 perceptron의 한계를 증명한 책을 출판
→ Perceptron은 어떤 방법으로도 XOR 문제를 풀 수 없음
- 이후 연결주의 인공지능 연구는 관심을 받지 못함 → AI winter



XOR 문제 예시

❖ 1개의 직선으로 0과 1을 구분할 수 없음



Minsky's perceptrons

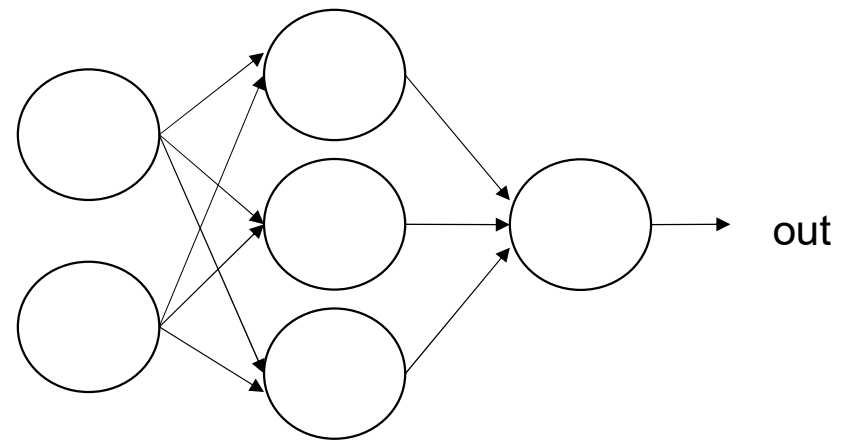
Introduction

▪ Multi-layer Perceptron (MLP, 1986)

- 여러 층의 perceptron으로 구성된 MLP와 오차역전파법이 개발됨
- XOR 문제를 해결함으로써 인공지능 연구가 다시 활발해지는 계기가 됨



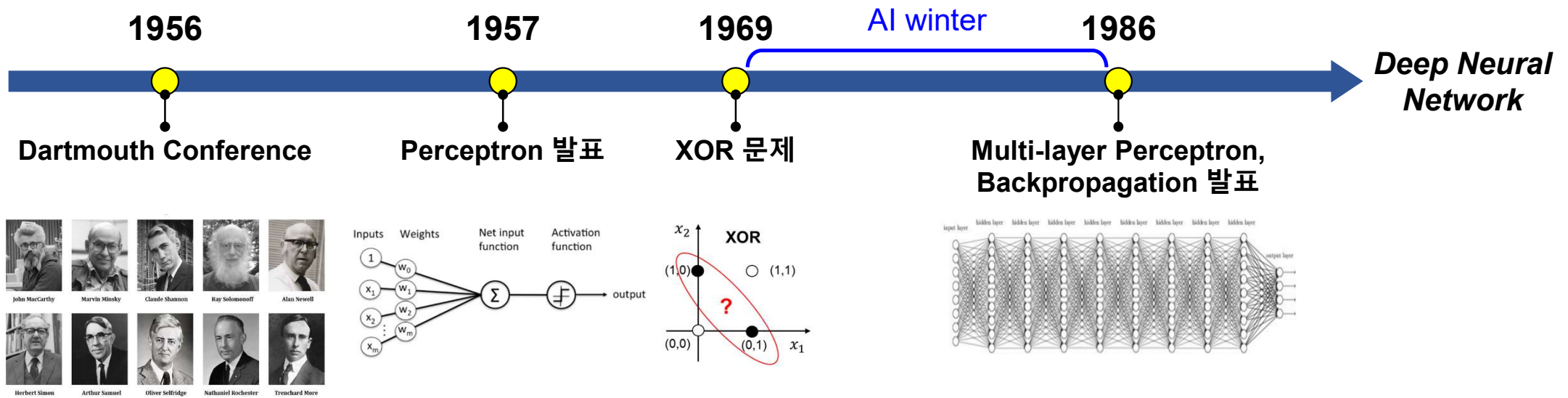
<D. Rumelhart, G. Hinton, and R. Williams>



<Multi-Layer Perceptron>

Introduction

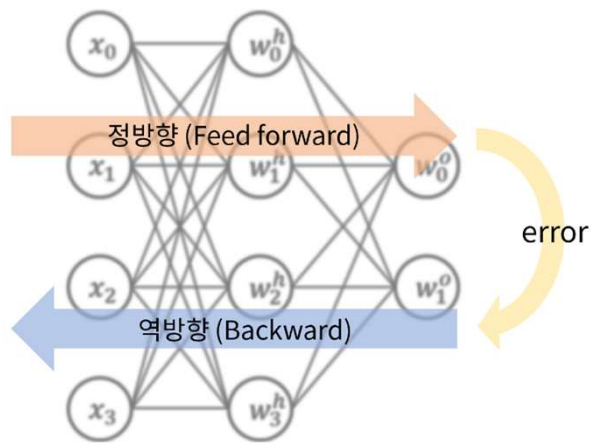
인공지능 역사의 흐름



Introduction

▪ Breakthrough for Artificial Intelligence

- (1) 오차 역전파법 개발 (Back-propagation)
- (2) 대규모 데이터셋 확보 (Big data)
- (3) 하드웨어 기술 발전



(1) 오차 역전파법



(2) 대규모 데이터셋

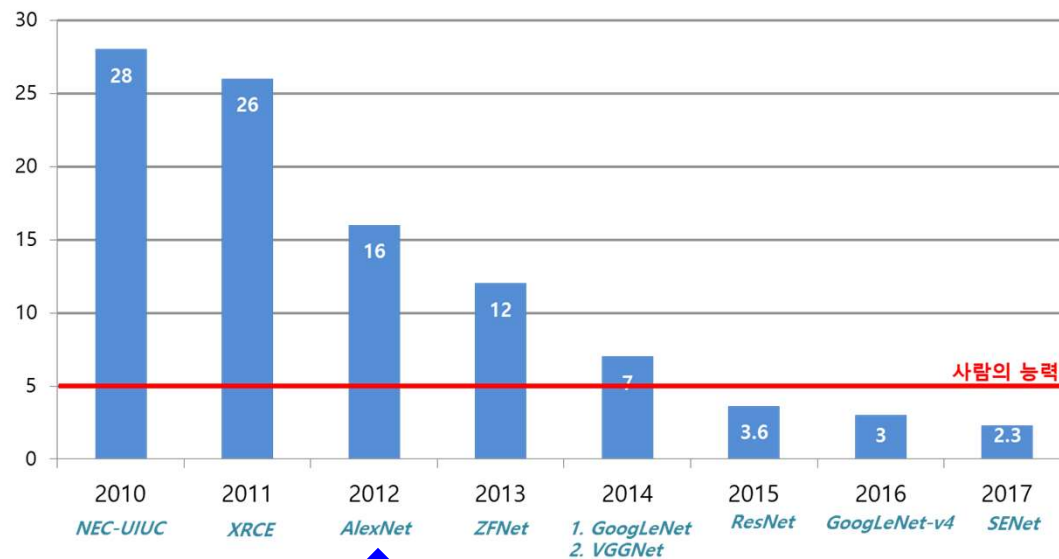


(3) GPU를 이용한 학습

Introduction

- 이미지넷 대규모 시각 인식 대회 (Image Large Scale Visual Recognition Challenge, ILSVR)
 - 대용량 이미지 데이터셋 (1,000개 클래스)에 대한 이미지 분류 알고리즘 성능 평가 대회
 - 딥러닝 기반 기법이 제안된 이후 분류 성능이 급격히 높아짐

우승 알고리즘의 분류 에러율(%)



딥러닝 기반 분류모델

Contents

1. Introduction

2. Applications

① Regression

② Classification

③ Logic gate

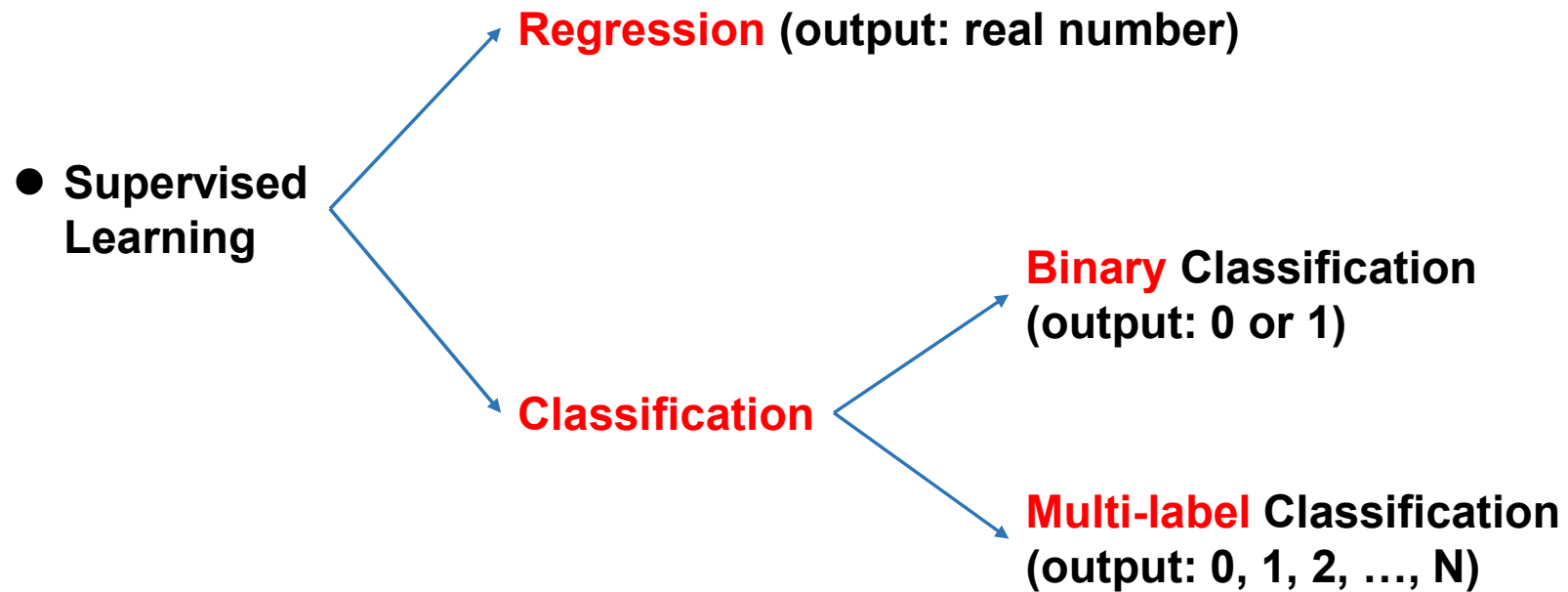
3. Multi-layer Perceptron (MLP)



Applications

▪ Supervised Learning (지도 학습)

- 입력 데이터와 대응되는 **정답 데이터**를 함께 이용해 학습하는 기법
- 대표적인 응용 분야로 **Regression (회귀)**와 **Classification (분류)**이 존재함

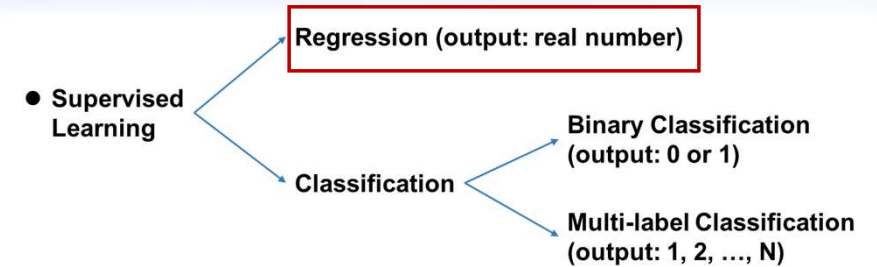


Applications

▪ Perceptron을 이용한 regression 예시

• Ex1. 공부한 시간으로 시험 점수 예측

- ✓ 입력 데이터: 공부한 시간 (x)
- ✓ 출력 데이터: 시험 점수 (y)



학습용 데이터셋 예시

Input x (hours)	Output y (score)
10	90
9	80
3	50
2	30

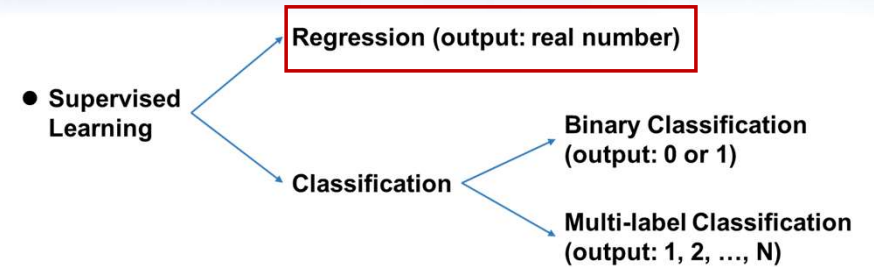
Applications

- Perceptron을 이용한 regression 예시

- Ex1. 공부한 시간으로 시험 점수 예측

- ✓ 입력 데이터: 공부한 시간 (x)

- ✓ 출력 데이터: 시험 점수 (y)



Input x (hours)	Output y (score)
10	90
9	80
3	50
2	30
5	?

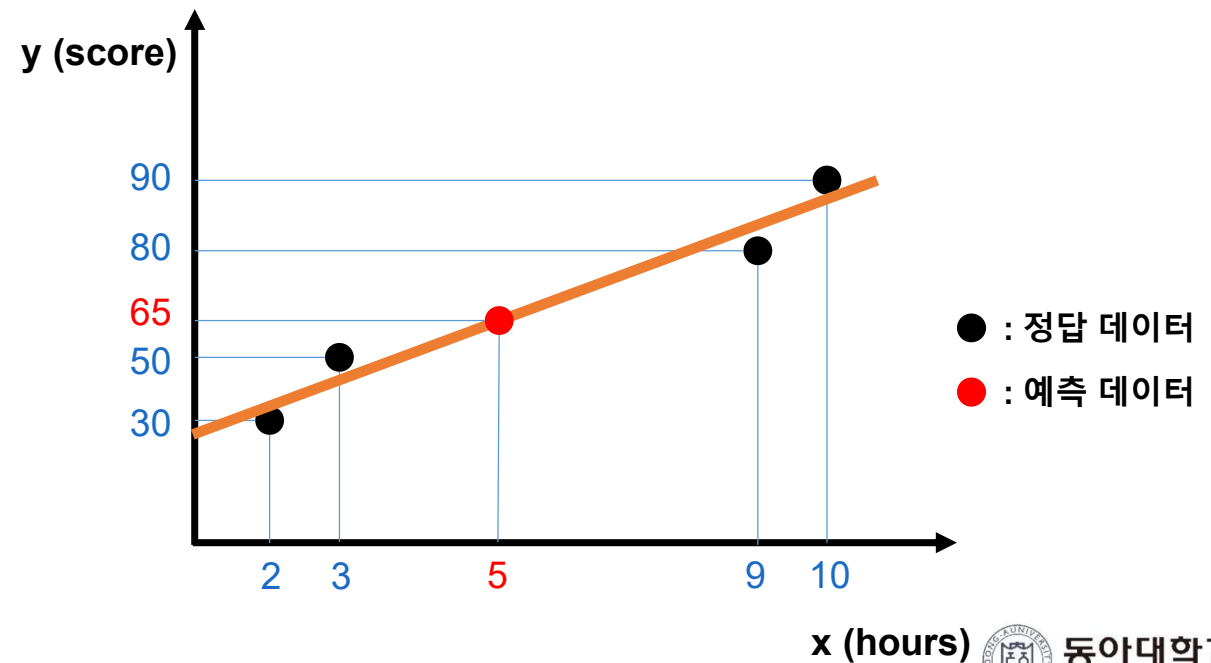
Applications

▪ Perceptron을 이용한 regression 예시

• Ex1. 공부한 시간으로 시험 점수 예측

- ✓ 입력 데이터: 공부한 시간 (x)
- ✓ 출력 데이터: 시험 점수 (y)

Input x (hours)	Output y (score)
10	90
9	80
3	50
2	30
5	65

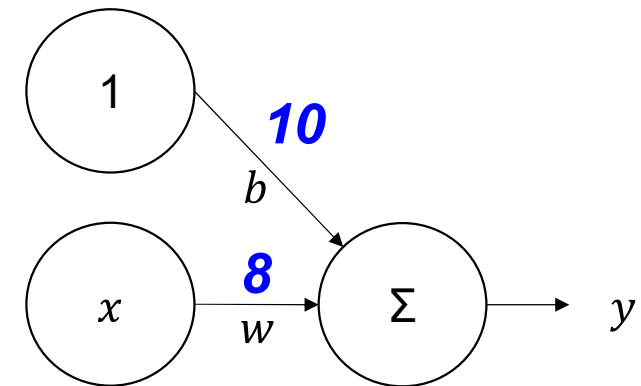
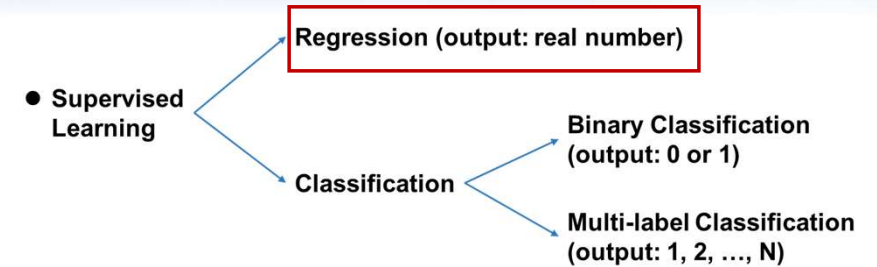
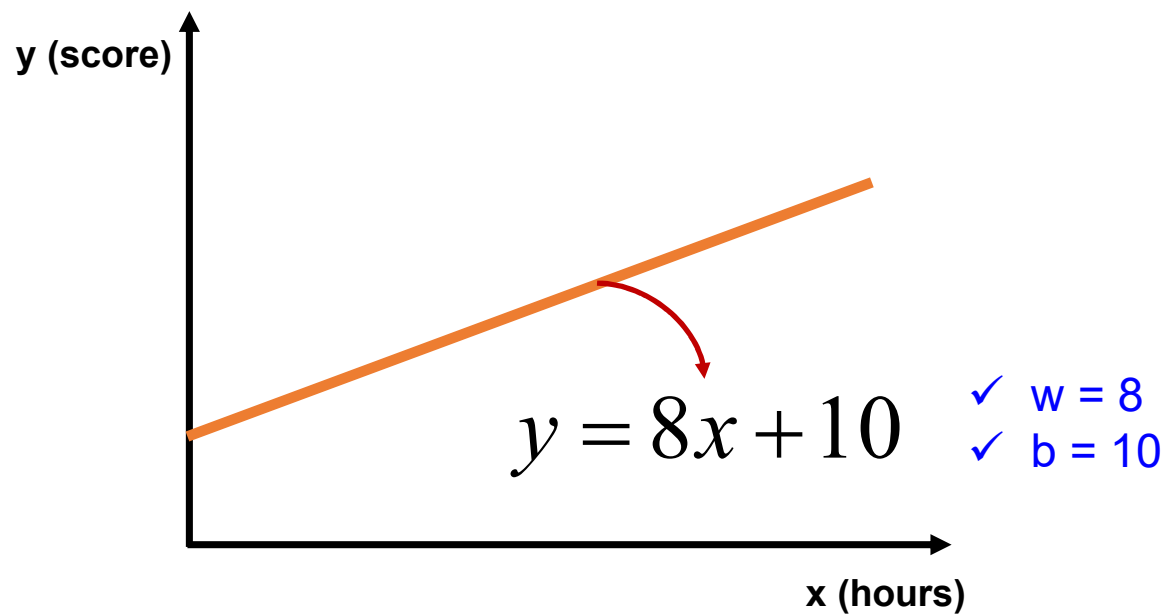


Applications

▪ Perceptron을 이용한 regression 예시

• Ex1. 공부한 시간으로 시험 점수 예측

- ✓ 입력 데이터: 공부한 시간 (x)
- ✓ 출력 데이터: 시험 점수 (y)



시험 점수 예측을 위한 perceptron 구조

Applications

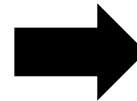
▪ Perceptron을 이용한 regression 예시

• Ex2. 주택 가격 예측

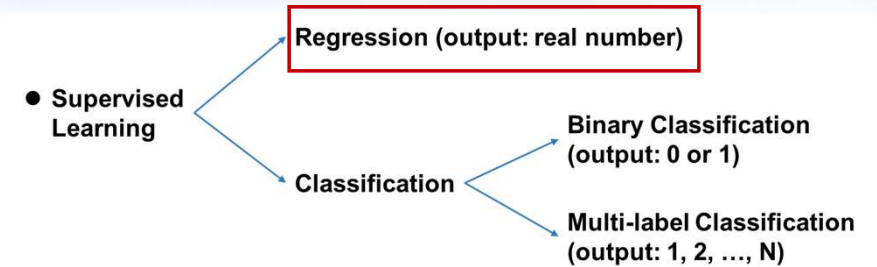
- ✓ 입력 데이터: 주택 가격에 영향을 미치는 정보 (x)
- ✓ 출력 데이터: 주택 가격 (y)

❖ 주택 가격에 영향을 미치는 정보

- ① 넓이 (x_1)
- ② 교통 (x_2)
- ③ 세대수 (x_3)
- ④ 층수 (x_4)



예상 주택 가격 (y)



Applications

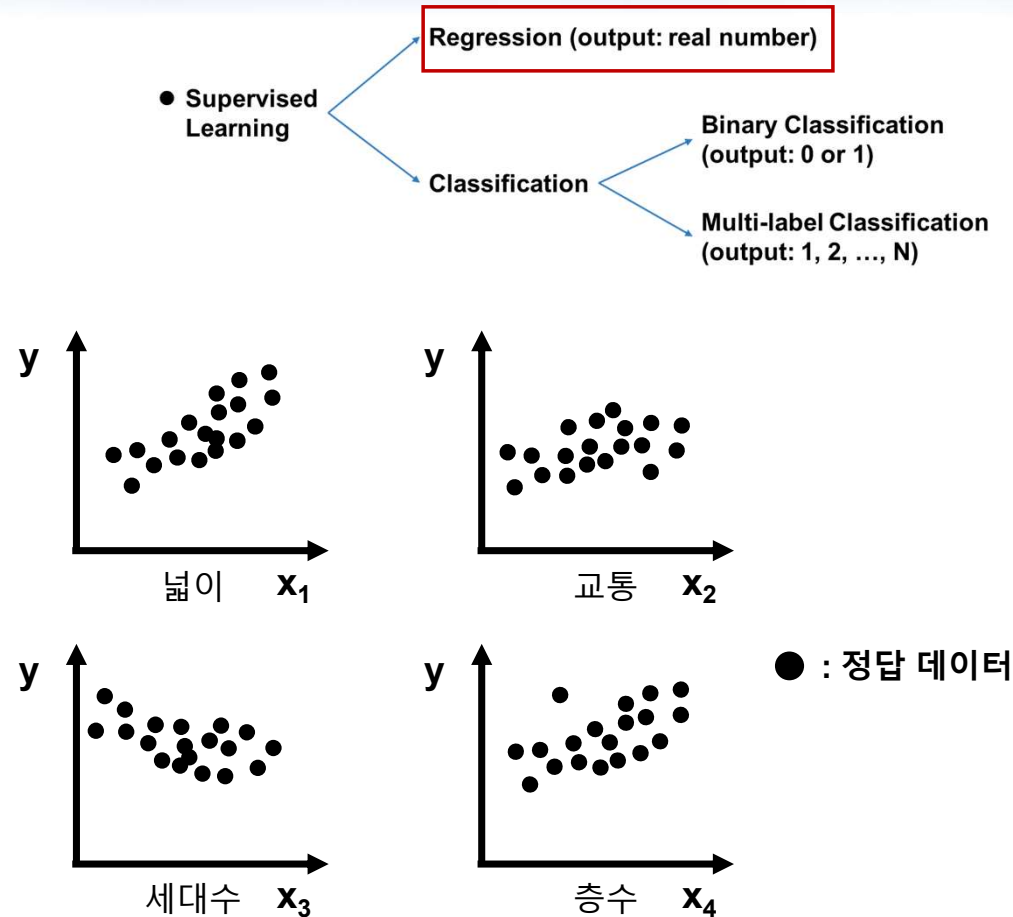
▪ Perceptron을 이용한 regression 예시

• Ex2. 주택 가격 예측

- ✓ 입력 데이터: 주택 가격에 영향을 미치는 정보 (x)
- ✓ 출력 데이터: 주택 가격 (y)

❖ 주택 가격에 영향을 미치는 정보

- ① 넓이 (x_1)
- ② 교통 (x_2)
- ③ 세대수 (x_3)
- ④ 층수 (x_4)



학습용 데이터셋 예시

Applications

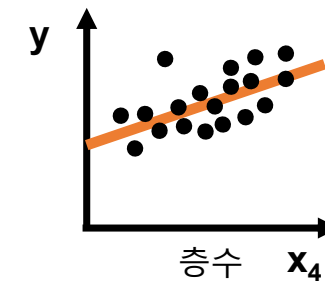
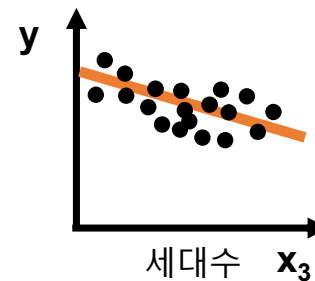
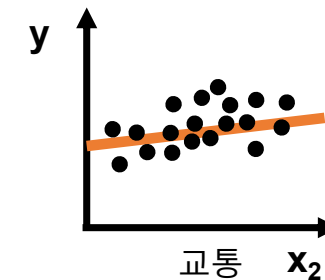
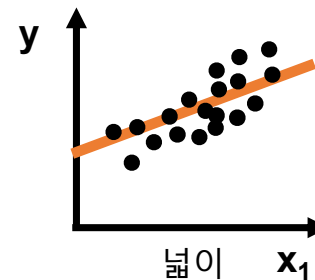
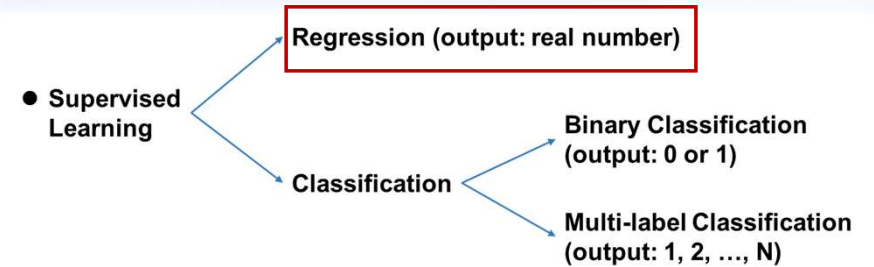
▪ Perceptron을 이용한 regression 예시

• Ex2. 주택 가격 예측

- ✓ 입력 데이터: 주택 가격에 영향을 미치는 정보 (x)
- ✓ 출력 데이터: 주택 가격 (y)

❖ 주택 가격에 영향을 미치는 정보

- ① 넓이 (x_1)
- ② 교통 (x_2)
- ③ 세대수 (x_3)
- ④ 층수 (x_4)

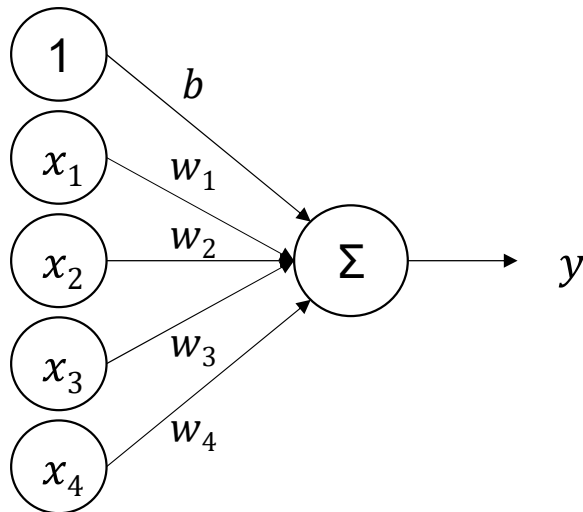


Applications

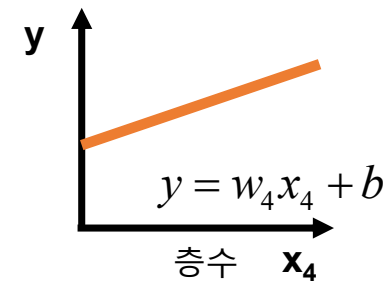
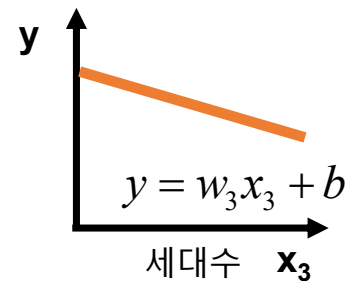
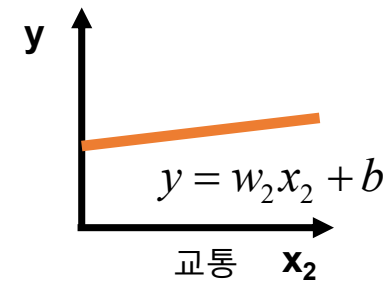
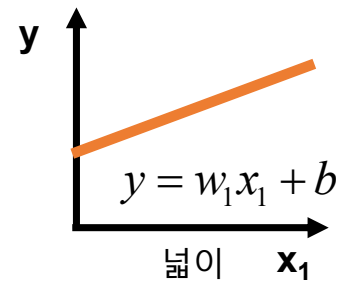
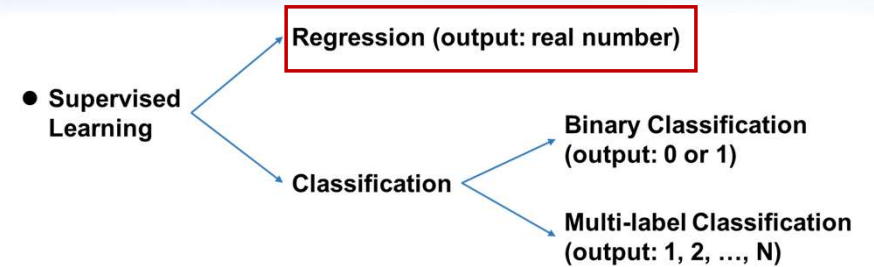
Perceptron을 이용한 regression 예시

Ex2. 주택 가격 예측

- ✓ 입력 데이터: 주택 가격에 영향을 미치는 정보 (x)
- ✓ 출력 데이터: 주택 가격 (y)



주택 가격 예측을 위한 perceptron 구조



Contents

1. Introduction

2. Applications

① Regression

② Classification

③ Logic gate

3. Multi-layer Perceptron (MLP)

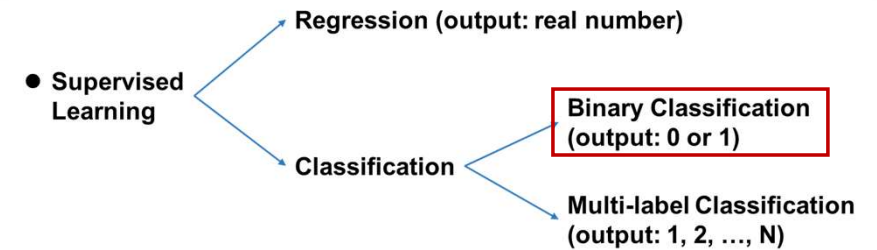


Applications of Perceptron

■ Perceptron을 이용한 classification 예시

• Ex1. 중간/기말고사 점수로 Pass/Non-pass (P/N) 분류

- ✓ 입력 데이터: 중간점수 (x_1), 기말점수 (x_2)
- ✓ 출력 데이터: Pass or Non-pass (y)



학습용 데이터셋 예시

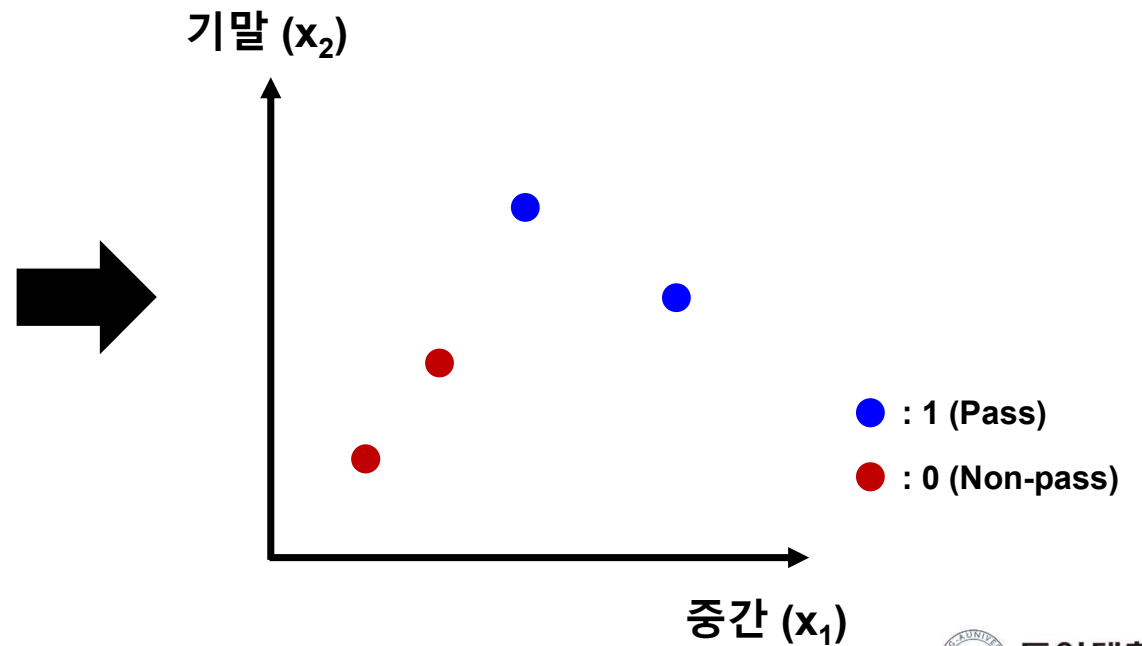
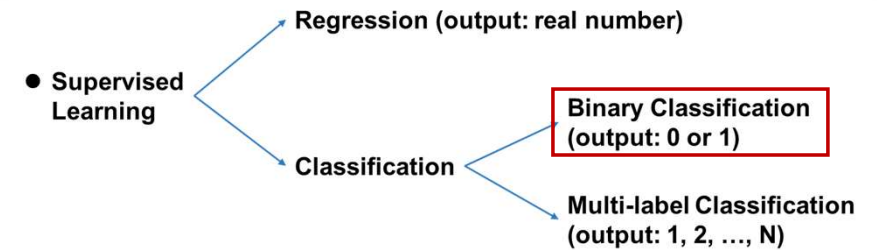
Input		Output
중간 (x_1)	기말 (x_2)	P/N (y)
90	80	1 (P)
75	60	1 (P)
40	50	0 (N)
25	25	0 (N)

Applications of Perceptron

- Perceptron을 이용한 classification 예시
 - Ex1. 중간/기말고사 점수로 Pass/Non-pass (P/N) 분류
 - ✓ 입력 데이터: 중간점수 (x_1), 기말점수 (x_2)
 - ✓ 출력 데이터: Pass or Non-pass (y)

학습용 데이터셋 예시

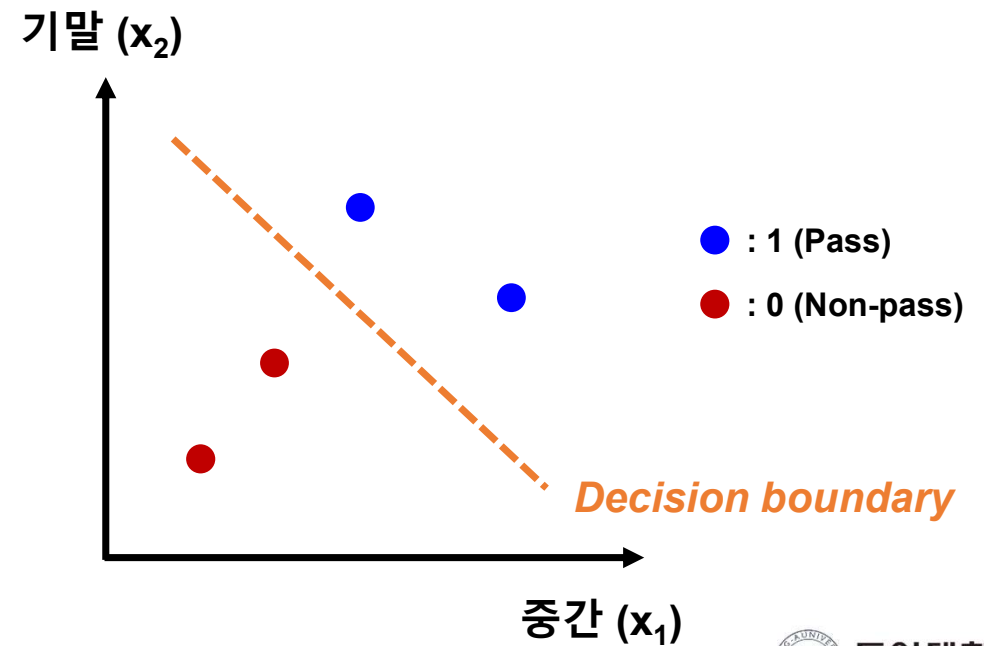
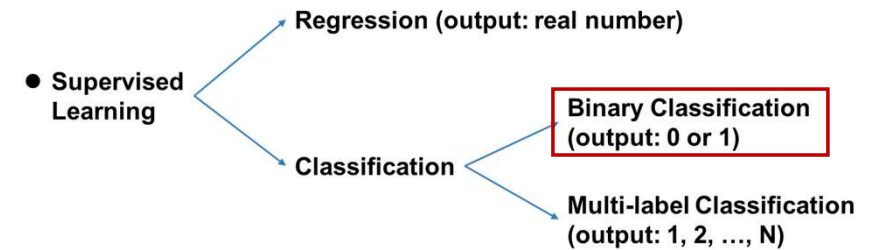
Input		Output
중간 (x_1)	기말 (x_2)	P/N (y)
75	90	1 (P)
80	60	1 (P)
40	50	0 (N)
25	25	0 (N)



Applications of Perceptron

- Perceptron을 이용한 classification 예시
 - Ex1. 중간/기말고사 점수로 Pass/Non-pass (P/N) 분류
 - ✓ 입력 데이터: 중간점수 (x_1), 기말점수 (x_2)
 - ✓ 출력 데이터: Pass or Non-pass (y)

❖ Perceptron은 weight, bias를 이용해
decision boundary를 구현함

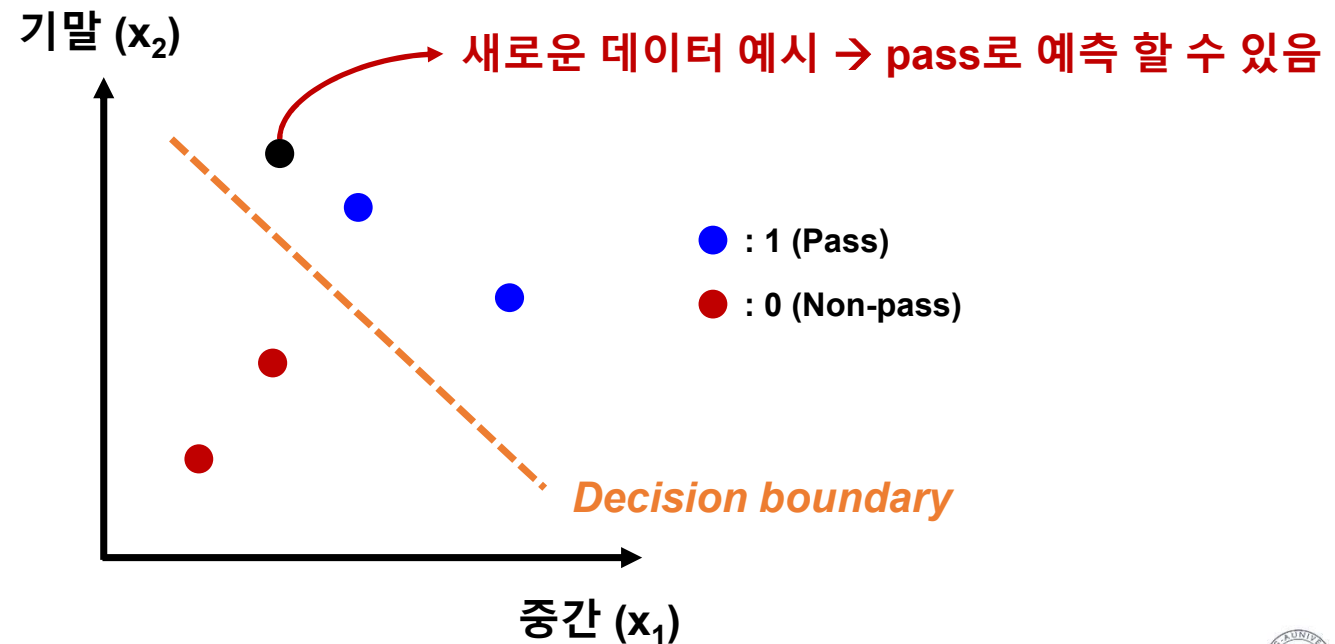
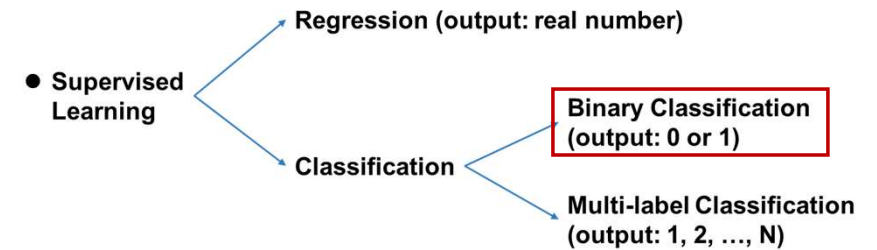


Applications of Perceptron

■ Perceptron을 이용한 classification 예시

• Ex1. 중간/기말고사 점수로 Pass/Non-pass (P/N) 분류

- ✓ 입력 데이터: 중간점수 (x_1), 기말점수 (x_2)
- ✓ 출력 데이터: Pass or Non-pass (y)

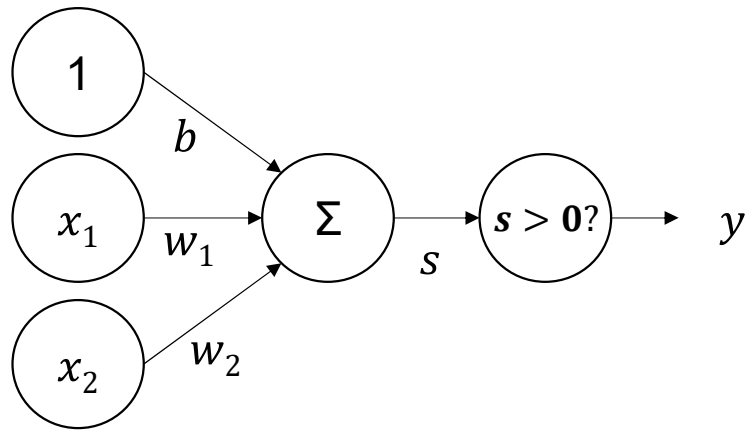


Applications of Perceptron

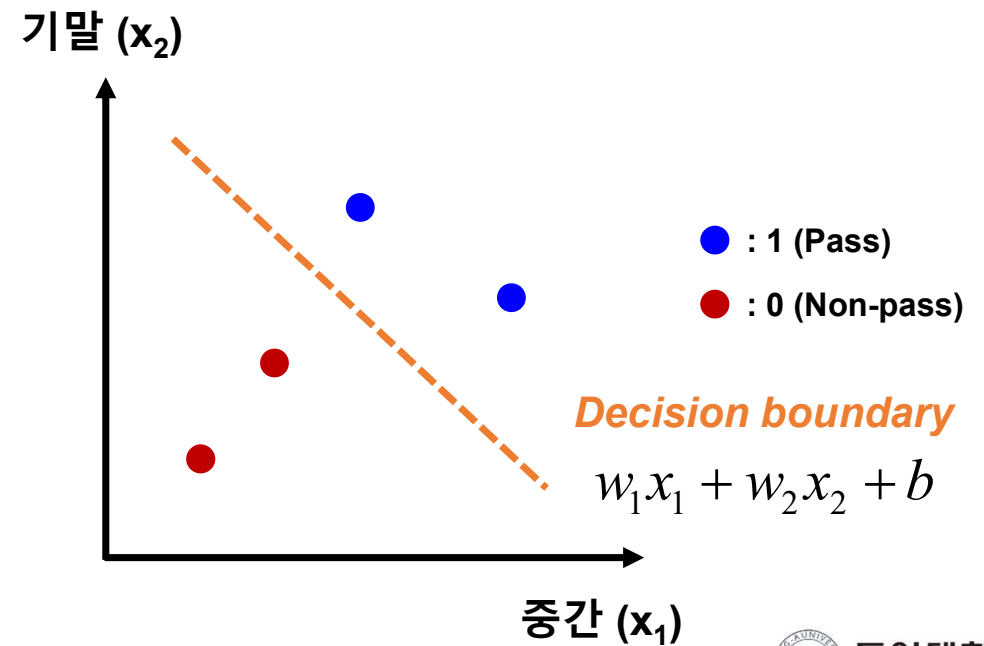
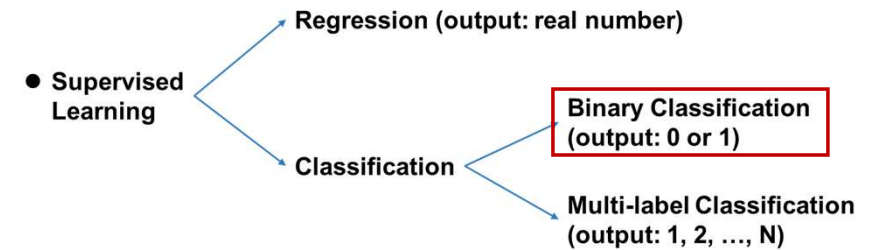
■ Perceptron을 이용한 classification 예시

• Ex1. 중간/기말고사 점수로 Pass/Non-pass (P/N) 분류

- ✓ 입력 데이터: 중간점수 (x_1), 기말점수 (x_2)
- ✓ 출력 데이터: Pass or Non-pass (y)

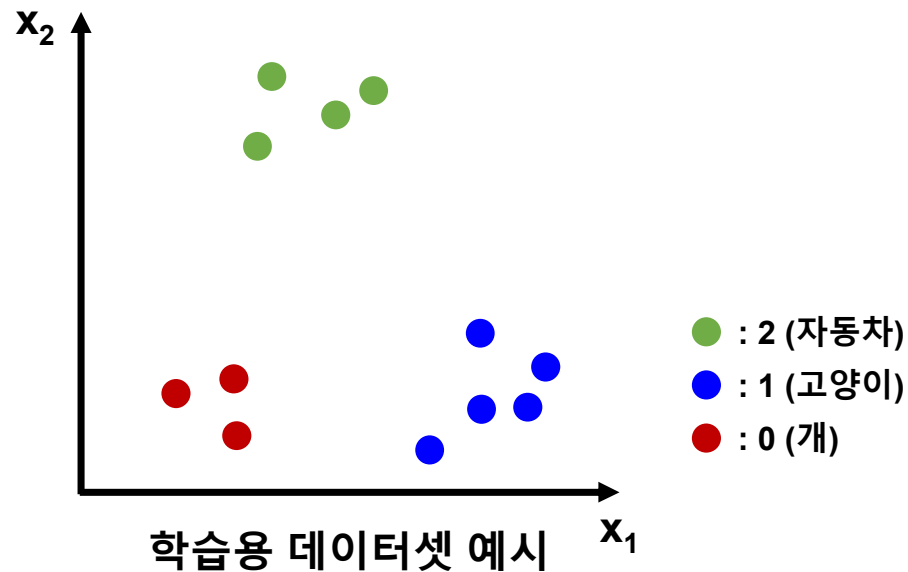
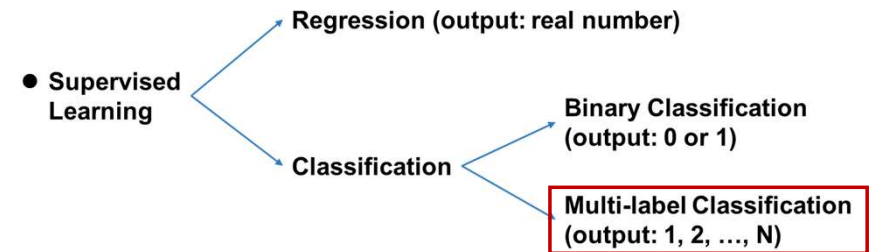


Pass/Non-pass 예측을 위한 perceptron 구조



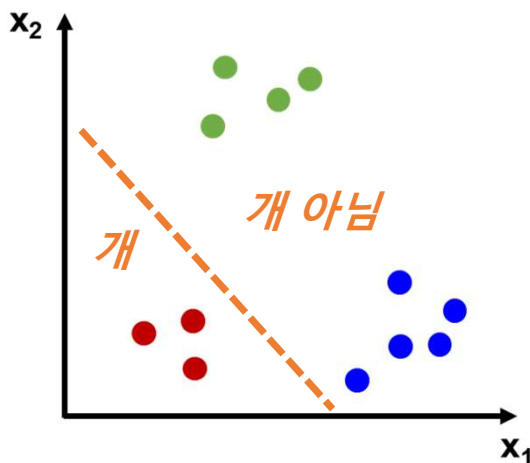
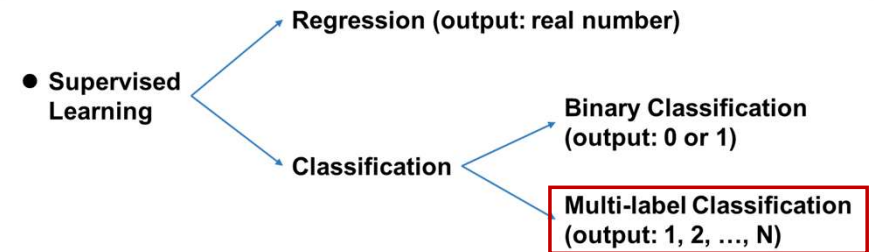
Applications of Perceptron

- Perceptron을 이용한 classification 예시
 - Ex2. 입력 데이터 x_1, x_2 로 개/고양이/자동차 분류
 - ✓ 입력 데이터: x_1, x_2
 - ✓ 출력 데이터: 개 or 고양이 or 자동차 (y)

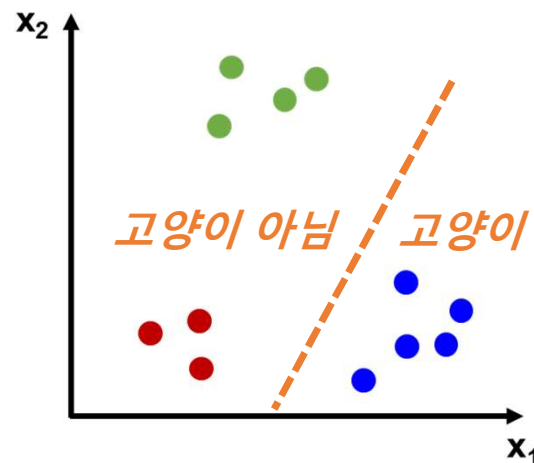


Applications of Perceptron

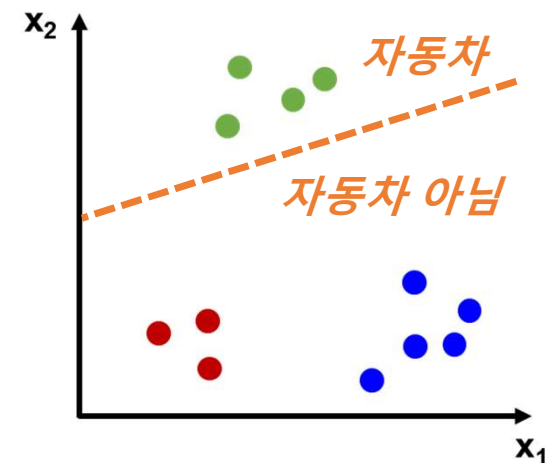
- Perceptron을 이용한 classification 예시
 - Ex2. 입력 데이터 x_1, x_2 로 개/고양이/자동차 분류
 - ✓ 입력 데이터: x_1, x_2
 - ✓ 출력 데이터: 개 or 고양이 or 자동차 (y)



개를 판단하는 perceptron



고양이를 판단하는 perceptron



자동차를 판단하는 perceptron

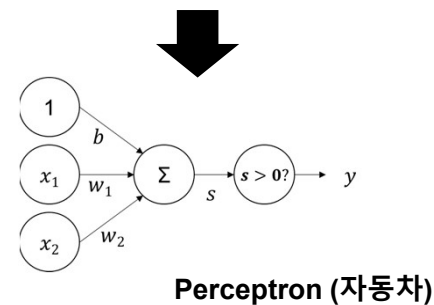
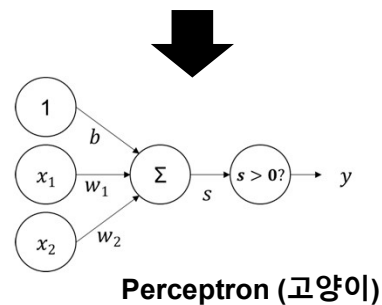
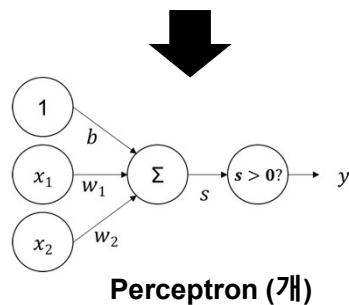
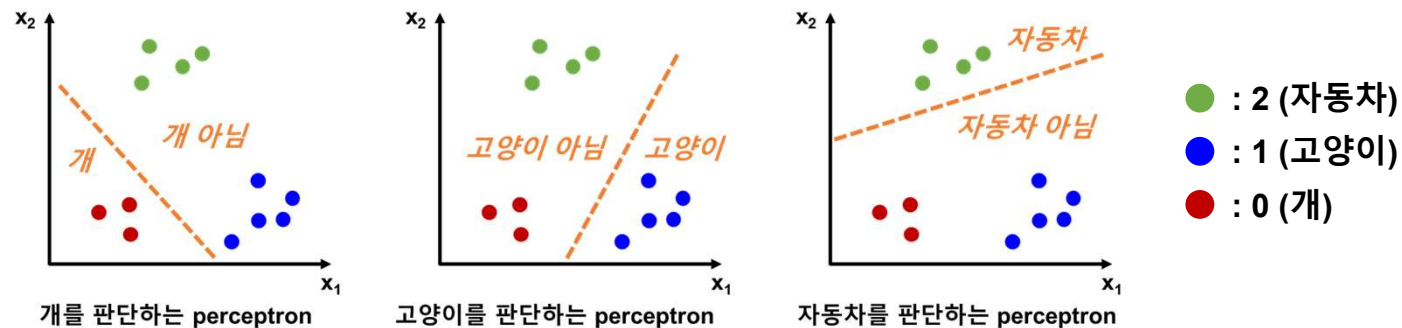
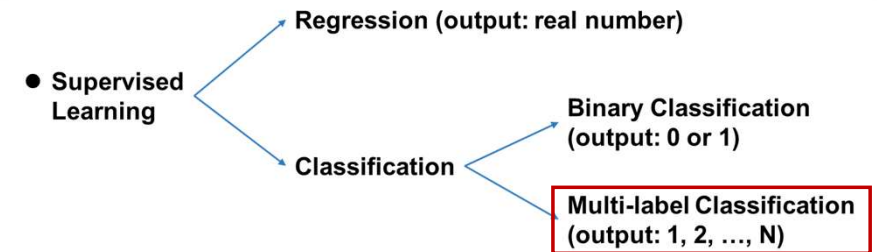
- : 2 (자동차)
- : 1 (고양이)
- : 0 (개)

Applications of Perceptron

Perceptron을 이용한 classification 예시

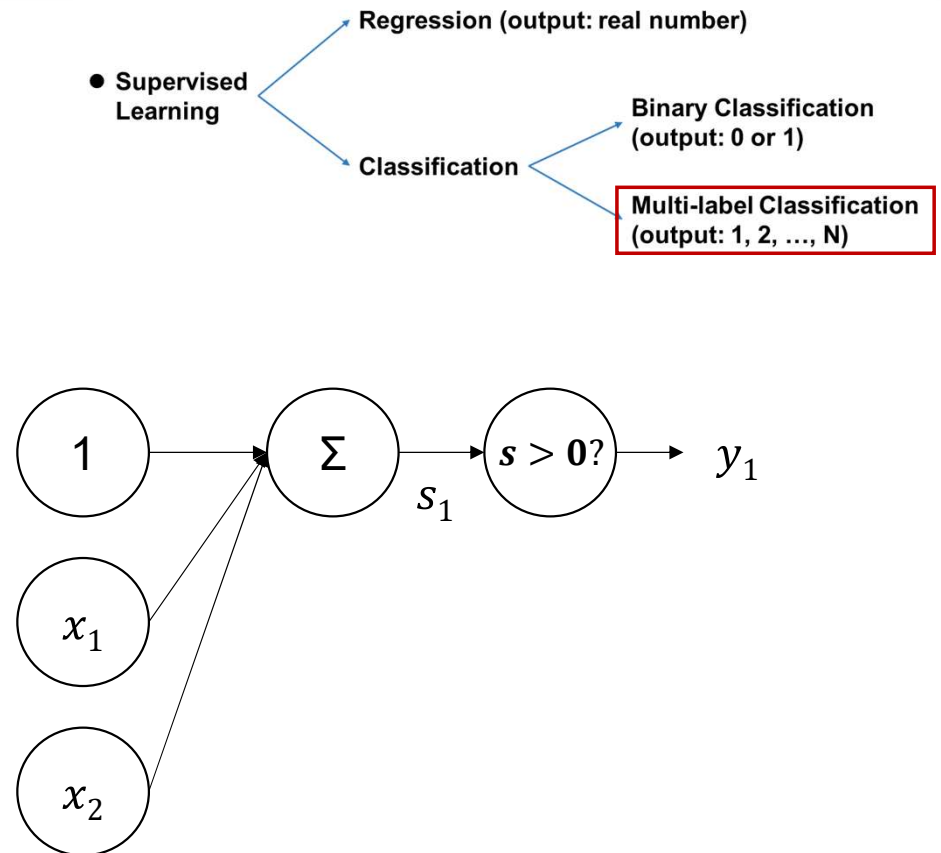
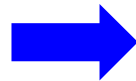
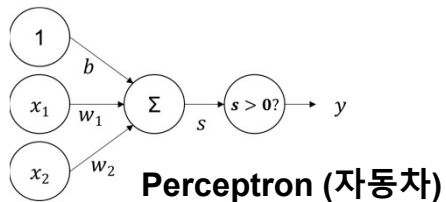
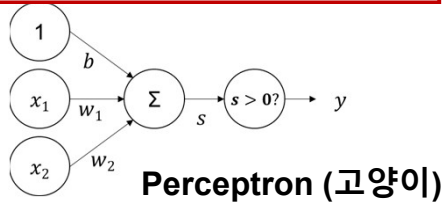
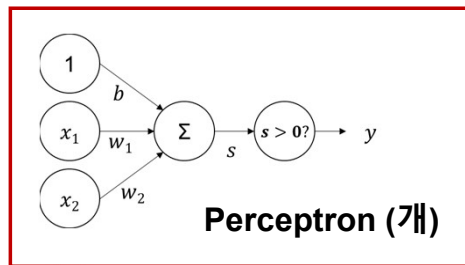
Ex2. 입력 데이터 x_1, x_2 로 개/고양이/자동차 분류

- ✓ 입력 데이터: x_1, x_2
- ✓ 출력 데이터: 개 or 고양이 or 자동차 (y)



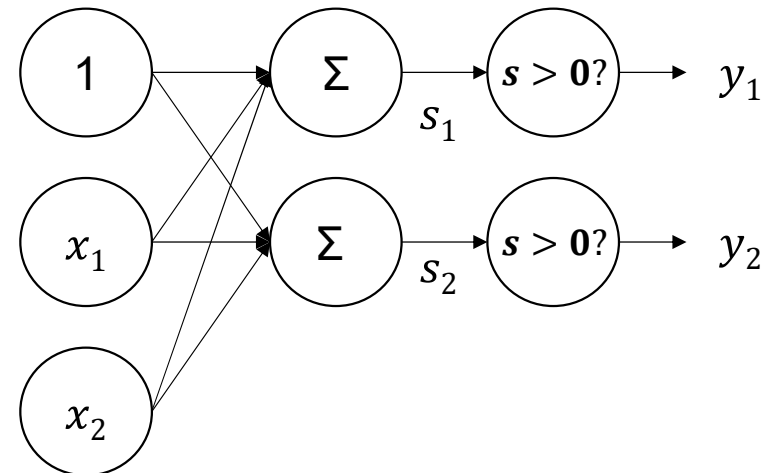
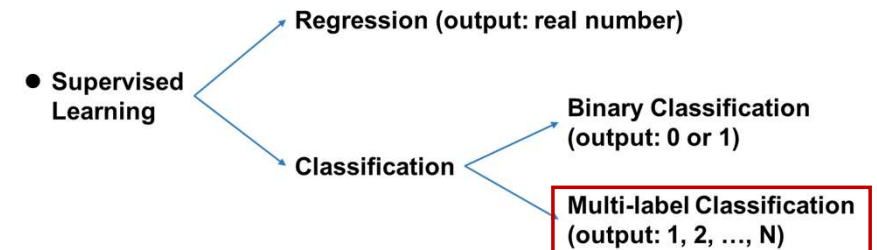
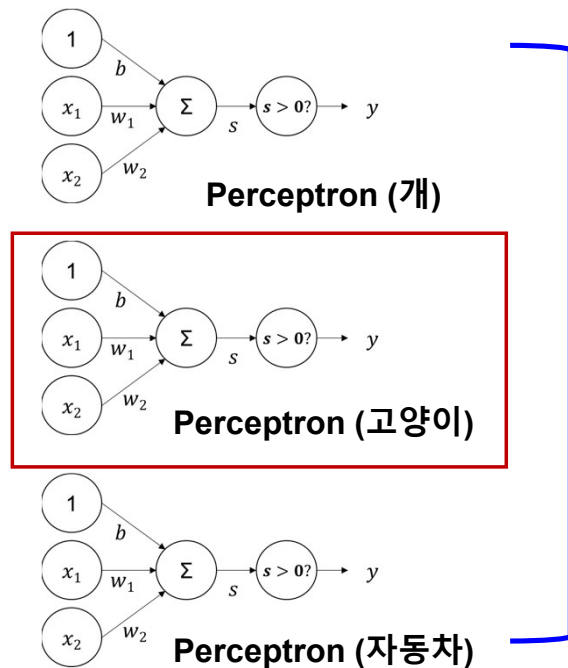
Applications of Perceptron

- Perceptron을 이용한 classification 예시
 - Ex2. 입력 데이터 x_1, x_2 로 개/고양이/자동차 분류
 - ✓ 입력 데이터: x_1, x_2
 - ✓ 출력 데이터: 개 or 고양이 or 자동차 (y)



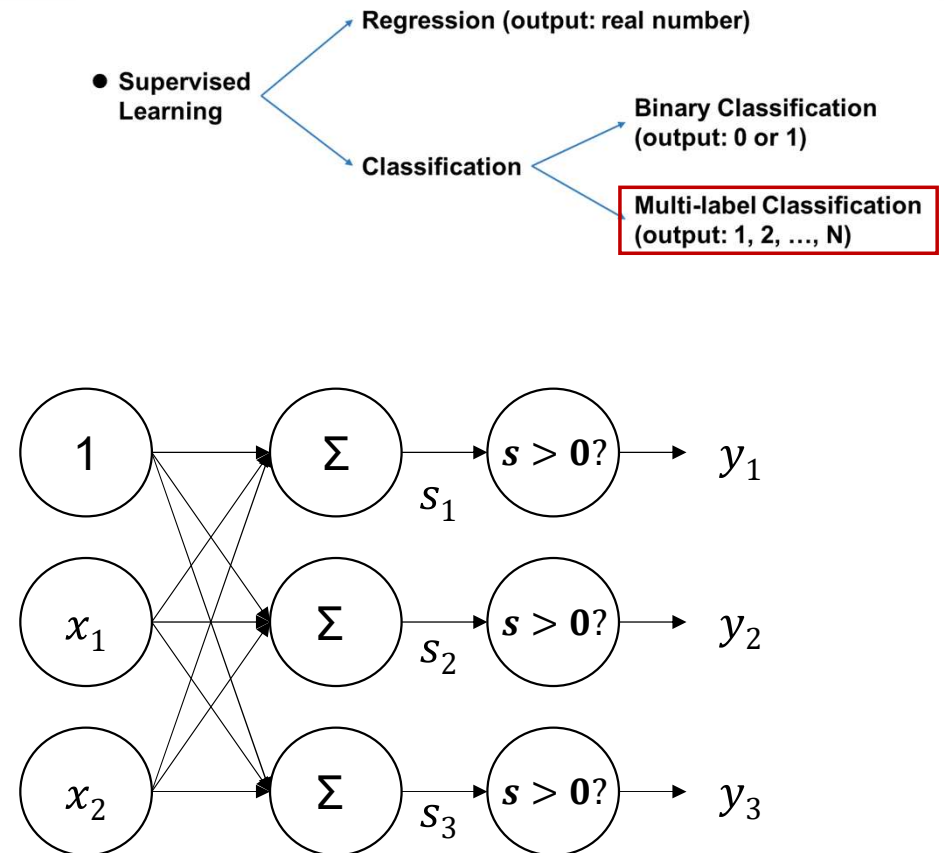
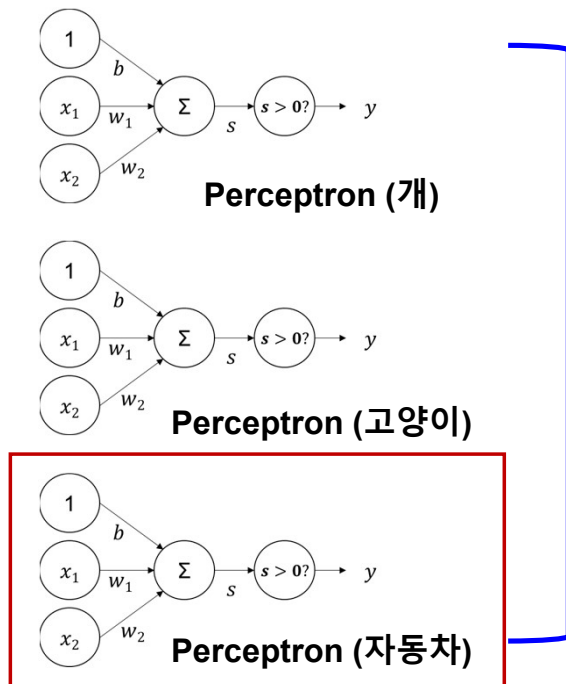
Applications of Perceptron

- Perceptron을 이용한 classification 예시
 - Ex2. 입력 데이터 x_1, x_2 로 개/고양이/자동차 분류
 - ✓ 입력 데이터: x_1, x_2
 - ✓ 출력 데이터: 개 or 고양이 or 자동차 (y)



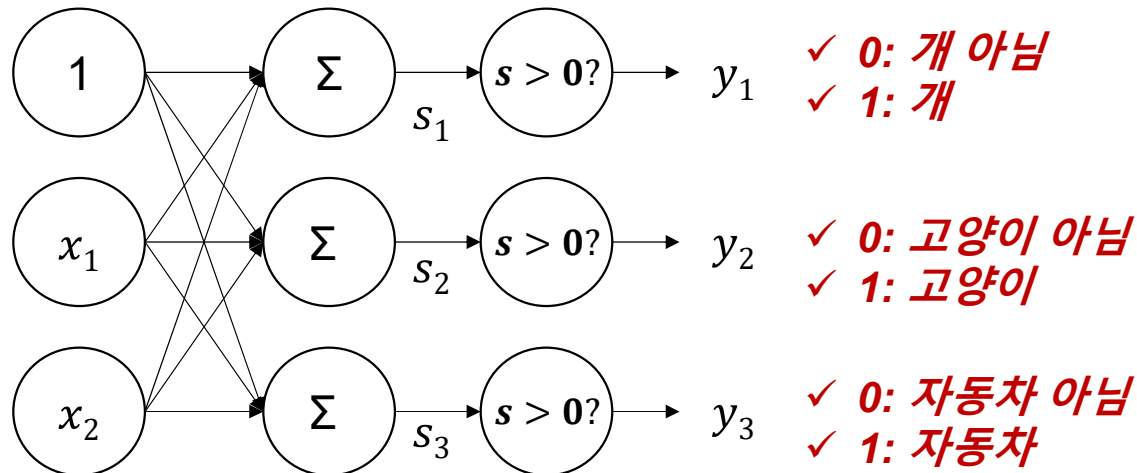
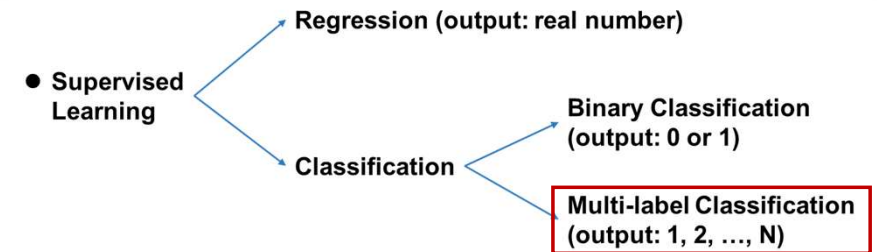
Applications of Perceptron

- Perceptron을 이용한 classification 예시
 - Ex2. 입력 데이터 x_1, x_2 로 개/고양이/자동차 분류
 - ✓ 입력 데이터: x_1, x_2
 - ✓ 출력 데이터: 개 or 고양이 or 자동차 (y)



Applications of Perceptron

- Perceptron을 이용한 classification 예시
 - Ex2. 입력 데이터 x_1, x_2 로 개/고양이/자동차 분류
 - ✓ 입력 데이터: x_1, x_2
 - ✓ 출력 데이터: 개 or 고양이 or 자동차 (y)



Contents

1. Introduction

2. Applications

① Regression

② Classification

③ Logic gate

3. Multi-layer Perceptron (MLP)



[Preliminary Study] Logic Gate

■ 논리 회로 (Logic gate)

- 입력 값에 대해 **논리 연산을 수행**하여 하나의 출력 값을 얻는 전자회로
- 대표적인 논리 회로로 AND, OR, NOT gate가 존재함

❖ AND gate



$$x_1 \cdot x_2 = y$$

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

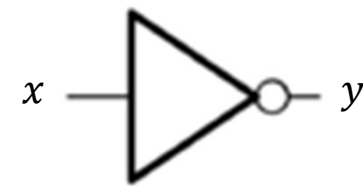
❖ OR gate



$$x_1 + x_2 = y$$

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	1

❖ NOT gate



$$\bar{x} = y$$

x	y
0	1
1	0

[Preliminary Study] Logic Gate

논리 회로 (Logic gate)

- 입력 값에 대해 논리 연산을 수행하여 하나의 출력 값을 얻는 전자회로
- 대표적인 논리 회로로 AND, OR, NOT gate가 존재함

❖ NAND gate



$$\overline{x_1 \cdot x_2} = y$$

x_1	x_2	y
0	0	1
0	1	1
1	0	1
1	1	0

❖ NOR gate



$$\overline{x_1 + x_2} = y$$

x_1	x_2	y
0	0	1
0	1	0
1	0	0
1	1	0

[Preliminary Study] Logic Gate

- 논리 회로 (Logic gate)

- 입력 값에 대해 논리 연산을 수행하여 하나의 출력 값을 얻는 전자회로

❖ XOR gate

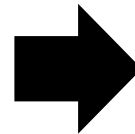
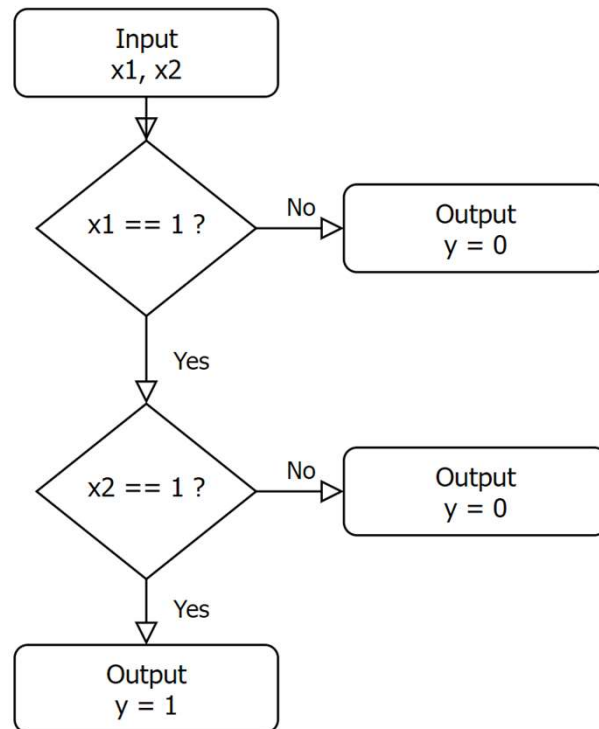


$$x_1 \oplus x_2 = y$$

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

Logic Gate using Perceptron

- 일반적인 방법을 통한 logic gate 구현 예시 - AND gate



❖ AND gate



x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

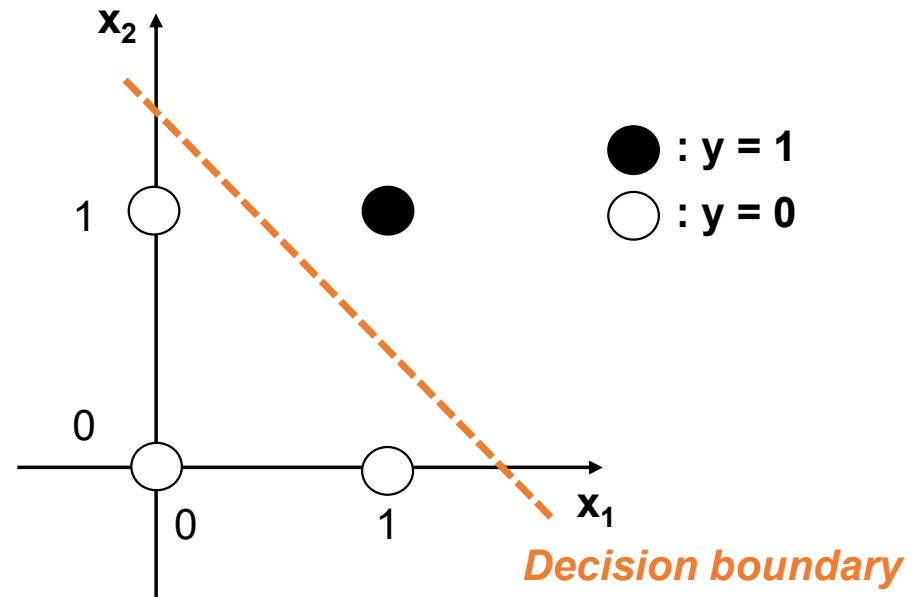
Logic Gate using Perceptron

- Perceptron을 이용한 논리 회로 구현
 - Binary classification (이진 분류)을 통해 논리 회로를 구현할 수 있음

❖ AND gate



x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

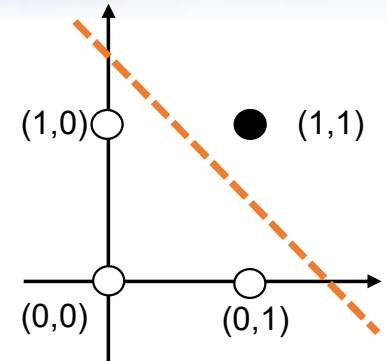
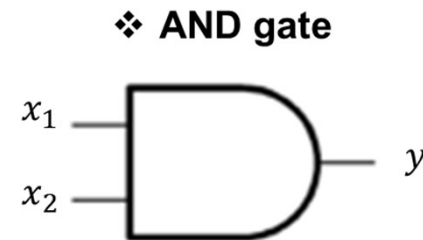
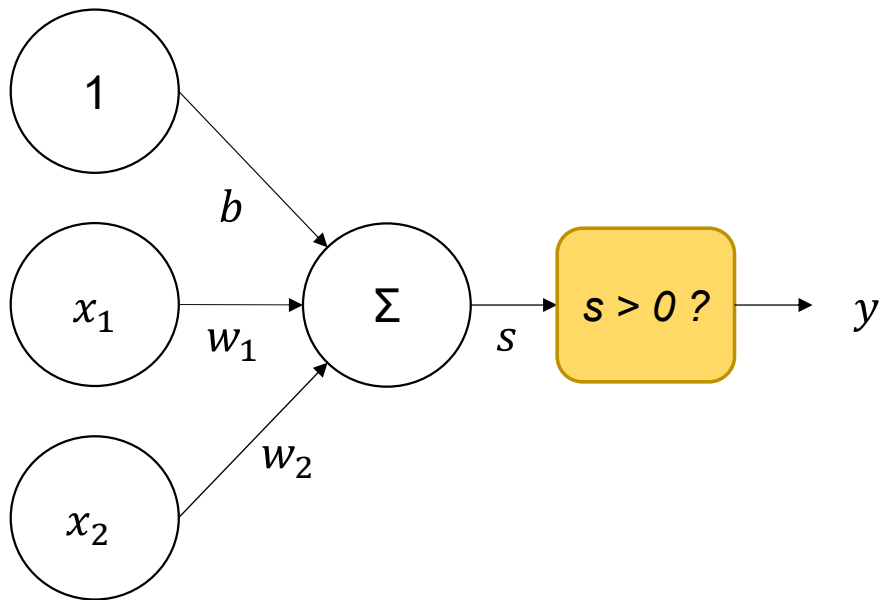


AND gate에 대한 이진 분류 예시

Logic Gate using Perceptron

▪ Perceptron을 이용한 논리 회로 구현 - AND gate

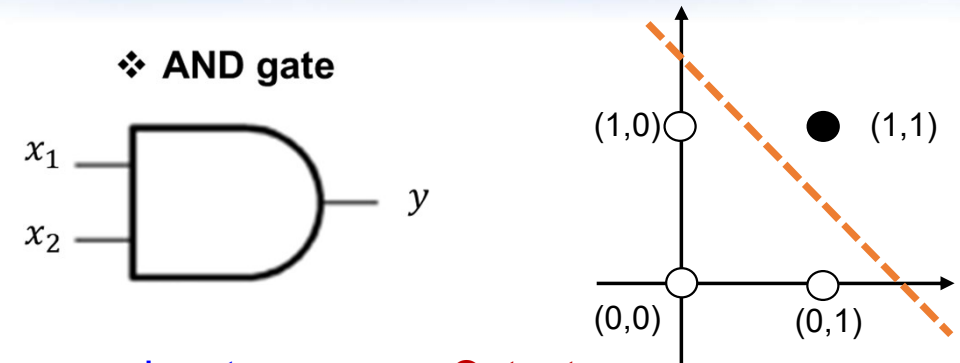
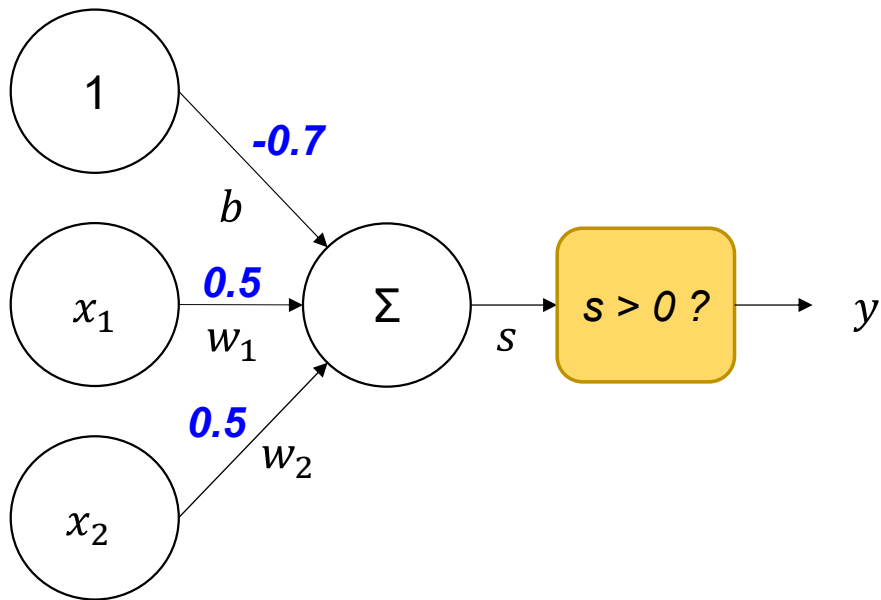
- 적절한 weight (w_1, w_2), bias (b)를 지정 해야함
- Ex. $w_1 = 0.5, w_2 = 0.5, b = -0.7$



Logic Gate using Perceptron

▪ Perceptron을 이용한 논리 회로 구현 - AND gate

- 적절한 weight (w_1, w_2), bias (b)를 지정 해야함
- Ex. $w_1 = 0.5, w_2 = 0.5, b = -0.7$



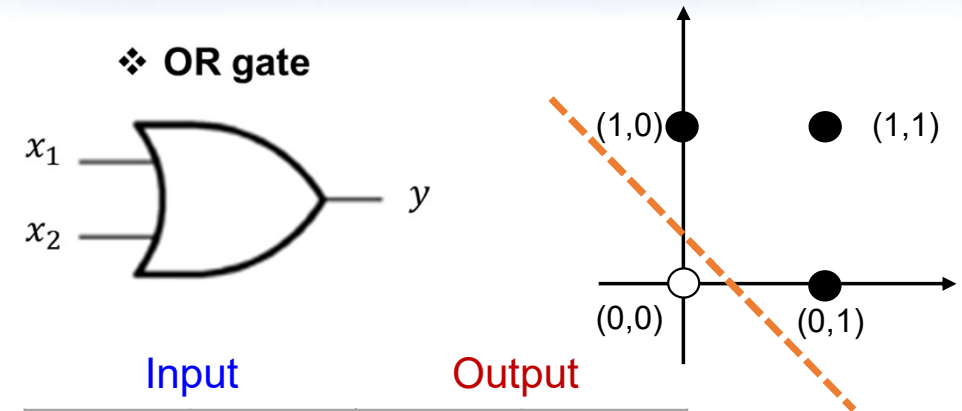
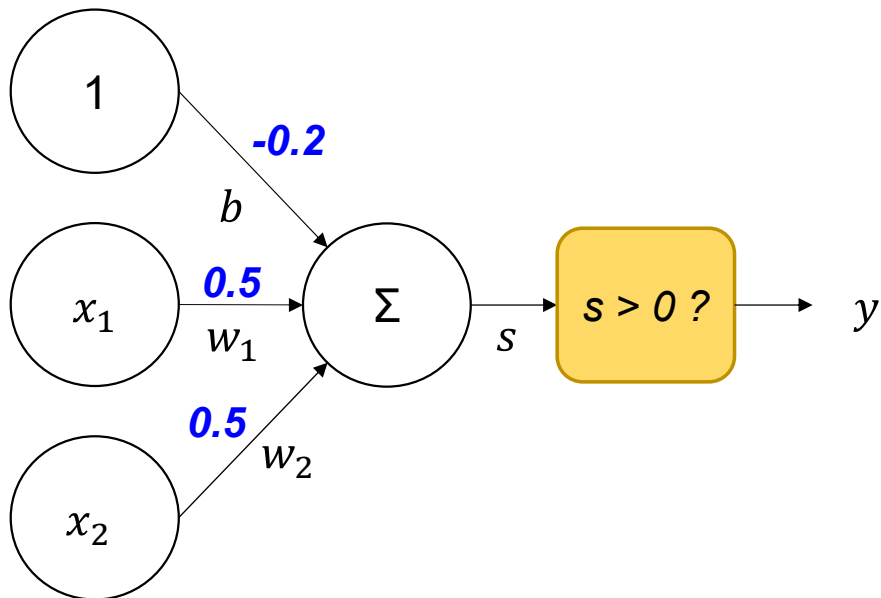
Input		Output	
x_1	x_2	s	y
0	0	-0.7	0
0	1	-0.2	0
1	0	-0.2	0
1	1	0.3	1

$$s = w_1 * x_1 + w_2 * x_2 + b$$

Logic Gate using Perceptron

Perceptron을 이용한 논리 회로 구현 - OR gate

- 적절한 weight (w_1, w_2), bias (b)를 지정 해야함
- Ex. $w_1 = 0.5, w_2 = 0.5, b = -0.2$



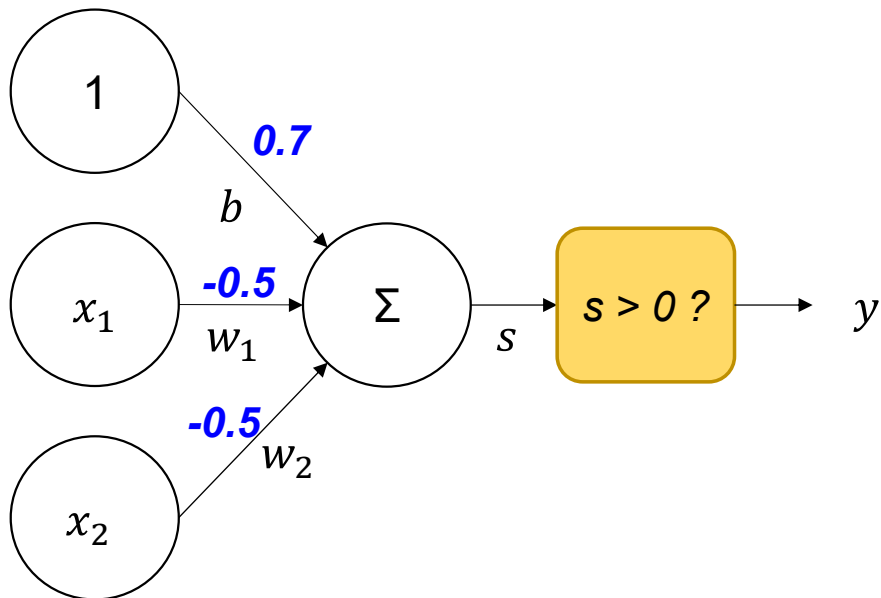
Input		Output	
x_1	x_2	s	y
0	0	-0.2	0
0	1	0.3	1
1	0	0.3	1
1	1	0.8	1

$$s = w_1 * x_1 + w_2 * x_2 + b$$

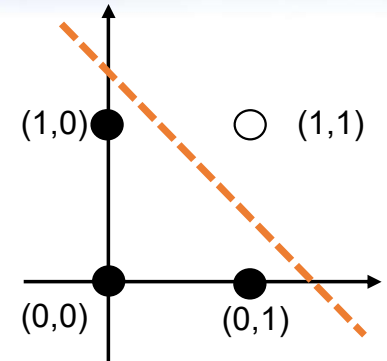
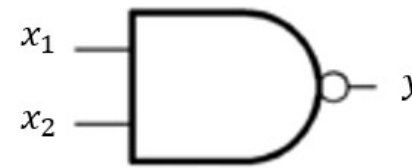
Logic Gate using Perceptron

Perceptron을 이용한 논리 회로 구현 - NAND gate

- 적정한 weight (w_1, w_2), bias (b)를 지정 해야함
- Ex. $w_1 = -0.5, w_2 = -0.5, b = 0.7$



❖ NAND gate



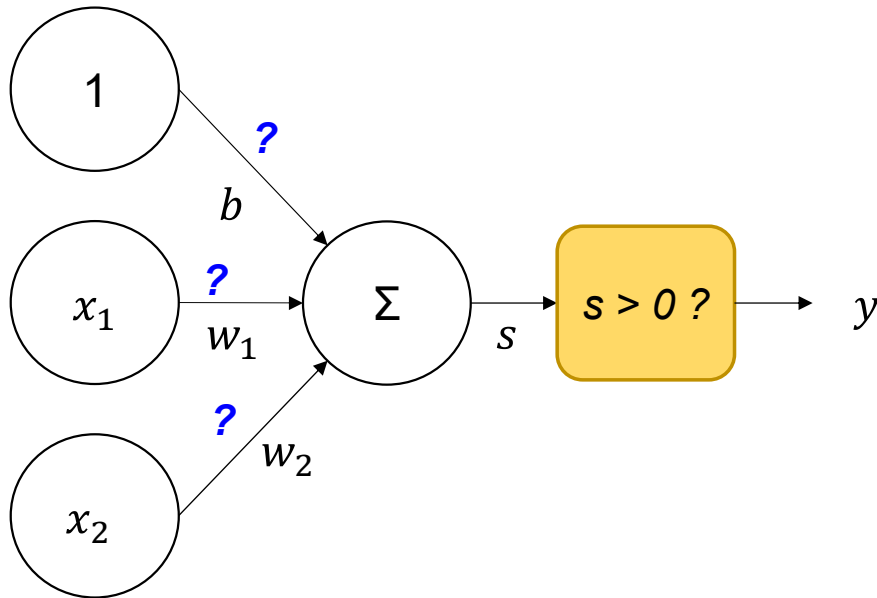
Input		Output	
x_1	x_2	s	y
0	0	0.7	1
0	1	0.2	1
1	0	0.2	1
1	1	-0.3	0

$$s = w_1 * x_1 + w_2 * x_2 + b$$

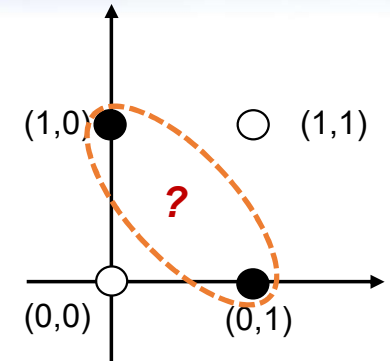
Logic Gate using Perceptron

Perceptron을 이용한 논리 회로 구현 - XOR gate

- 적절한 weight (w_1 , w_2), bias (b)를 지정 해야함
- XOR gate는 단층 perceptron을 이용해 구현할 수 없음



❖ XOR gate



x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

Contents

1. Introduction

2. Applications

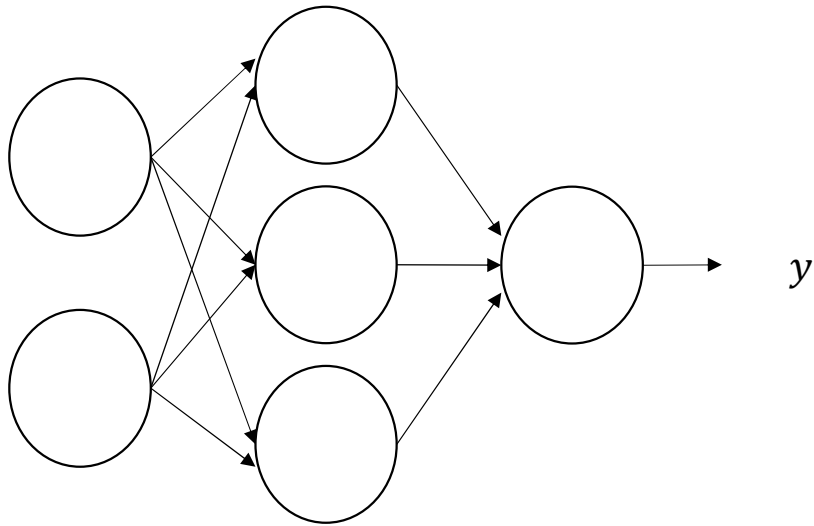
- ① Regression
- ② Classification
- ③ Logic gate

3. Multi-layer Perceptron (MLP)

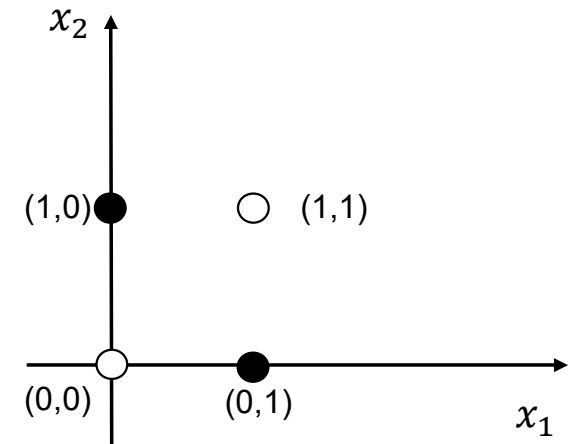


Multi-Layer Perceptron

- Multi-Layer Perceptron : 여러 개의 층을 가진 Perceptron → XOR Gate 구현 가능



x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

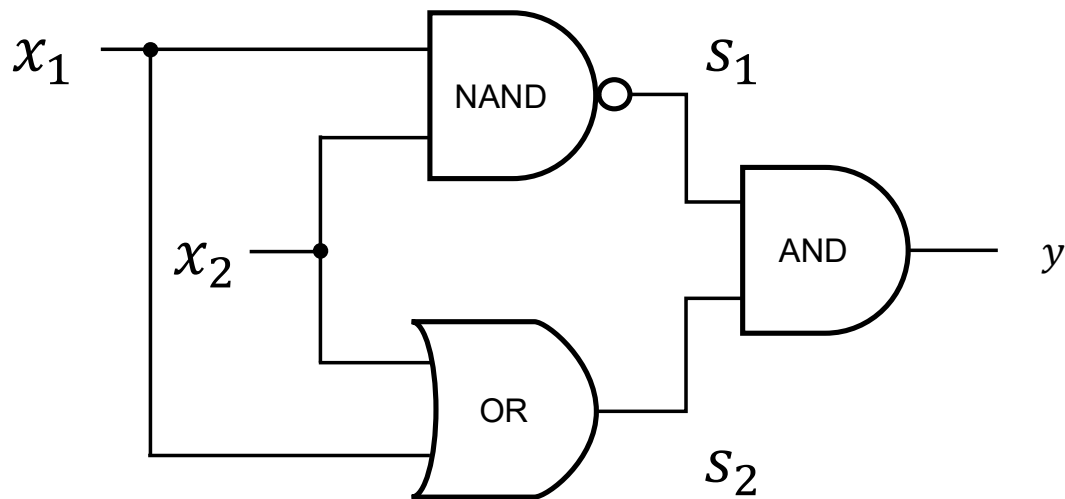


<Multi-Layer Perceptron>

Multi-Layer Perceptron

■ XOR Gate

- NAND Gate, OR Gate, AND Gate 서로 연결 하여 XOR Gate 구현가능



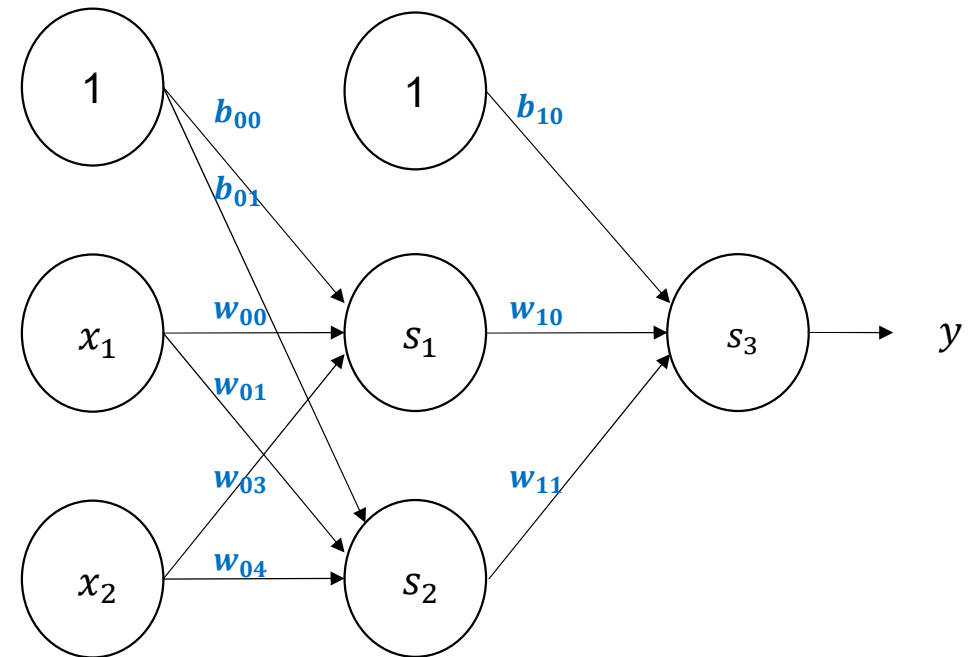
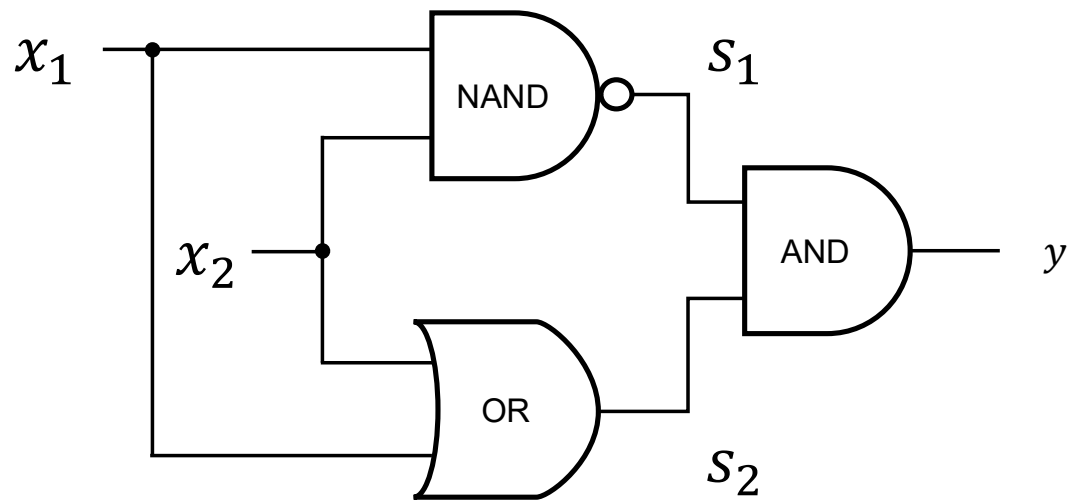
x_1	x_2	s_1	s_2	y
0	0	1	0	0
0	1	1	1	1
1	0	1	1	1
1	1	0	1	0

- $s_1 : x_1 \text{ NAND } x_2$
- $s_2 : x_1 \text{ OR } x_2$
- $y : s_1 \text{ AND } s_2$

Multi-Layer Perceptron

■ XOR Gate

- NAND Gate, OR Gate, AND Gate 서로 연결 하여 XOR Gate 구현가능
- Perceptron 에서 층을 쌓아 서로 연결하여 XOR Gate 구현가능

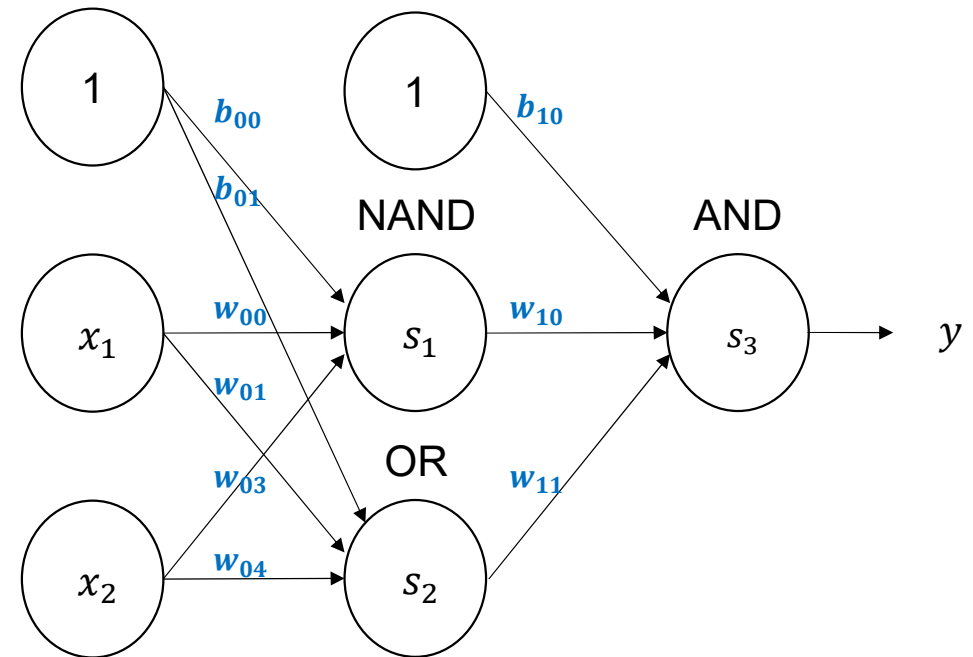
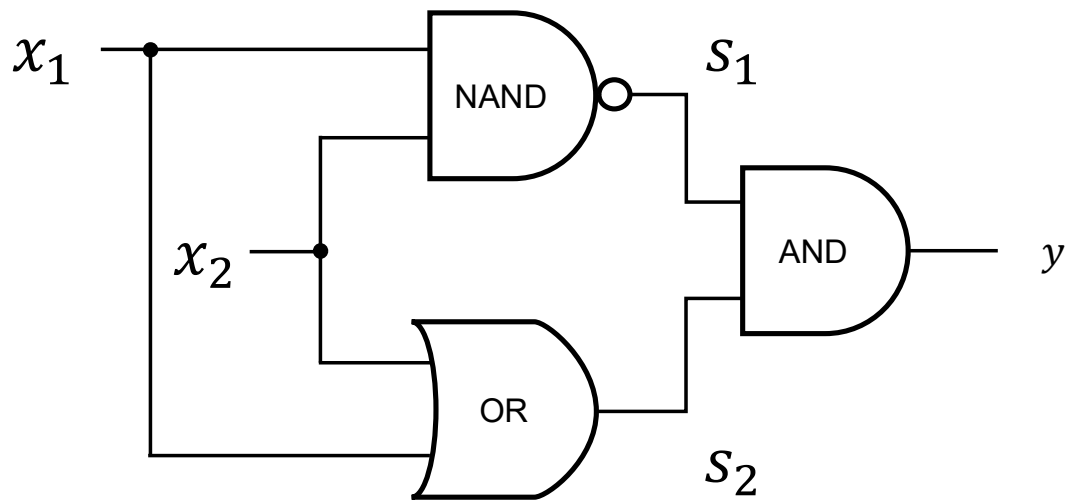


<Multi-Layer Perceptron>

Multi-Layer Perceptron

■ XOR Gate

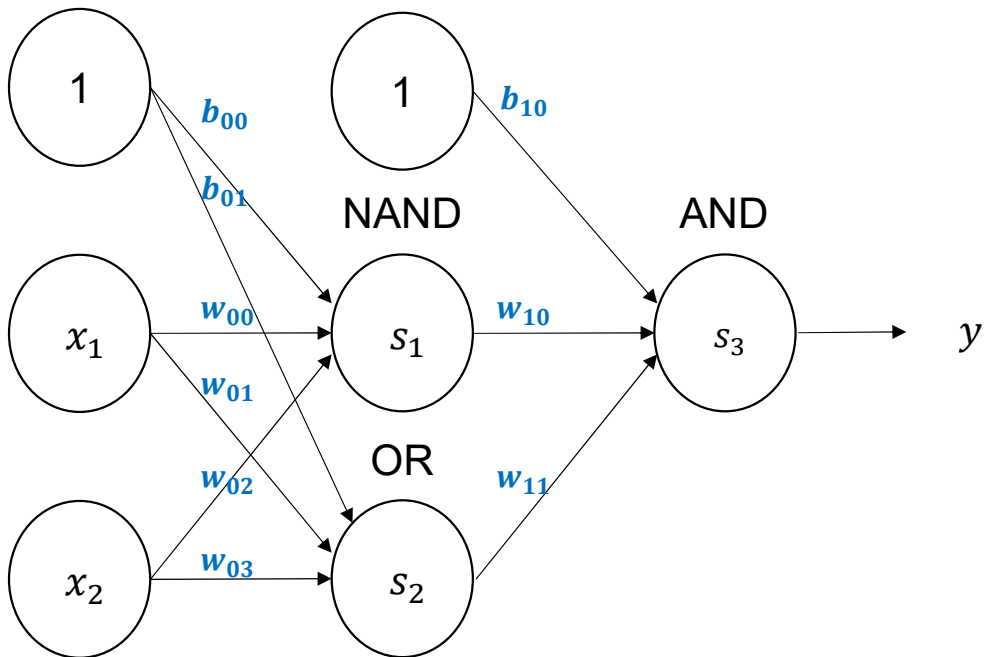
- NAND Gate, OR Gate, AND Gate 서로 연결 하여 XOR Gate 구현가능
- Perceptron 에서 층을 쌓아 서로 연결하여 XOR Gate 구현가능



<Multi-Layer Perceptron>

Multi-Layer Perceptron

- XOR Gate

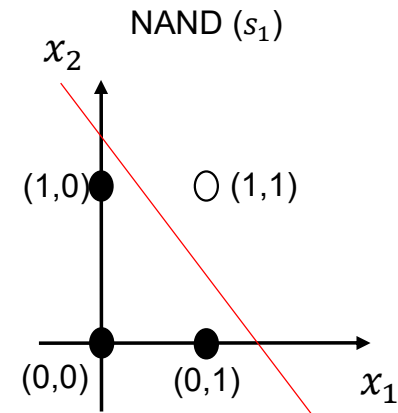
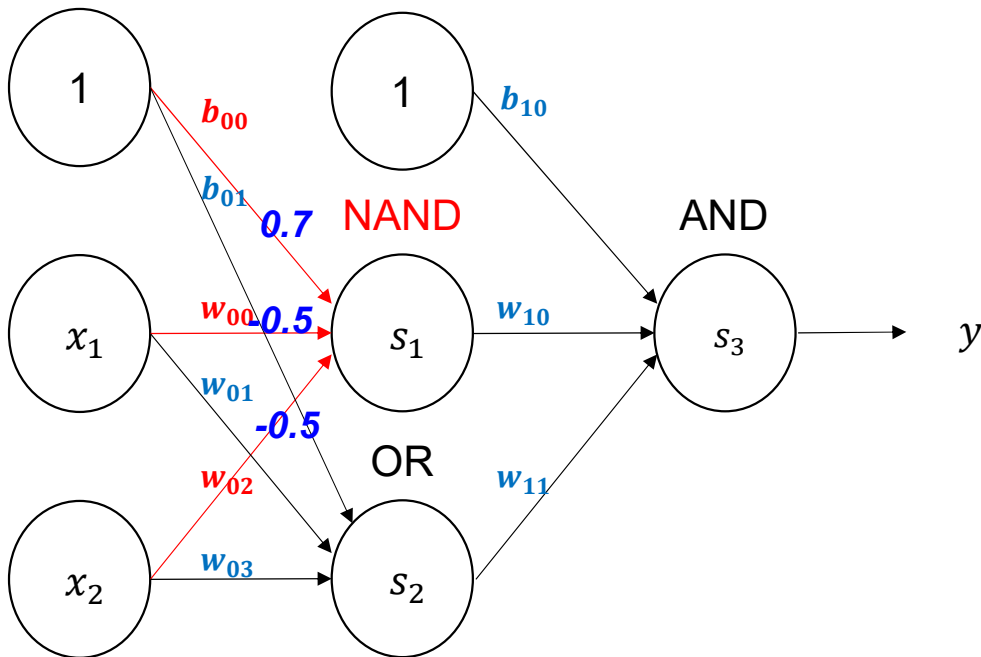


<Multi-Layer Perceptron>

Multi-Layer Perceptron

■ XOR Gate

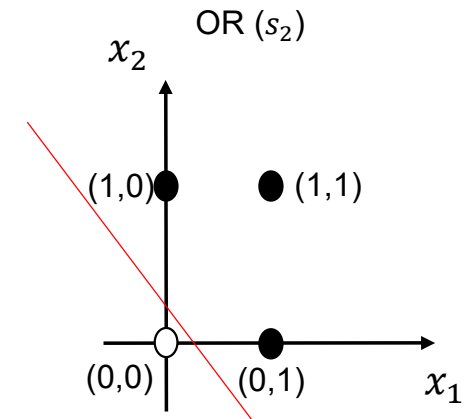
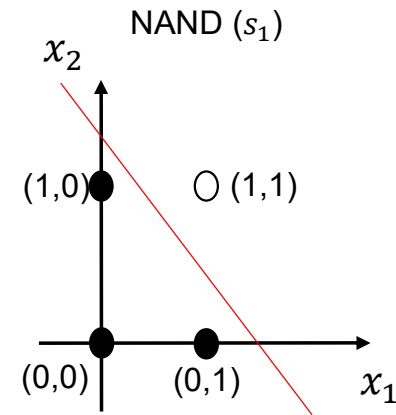
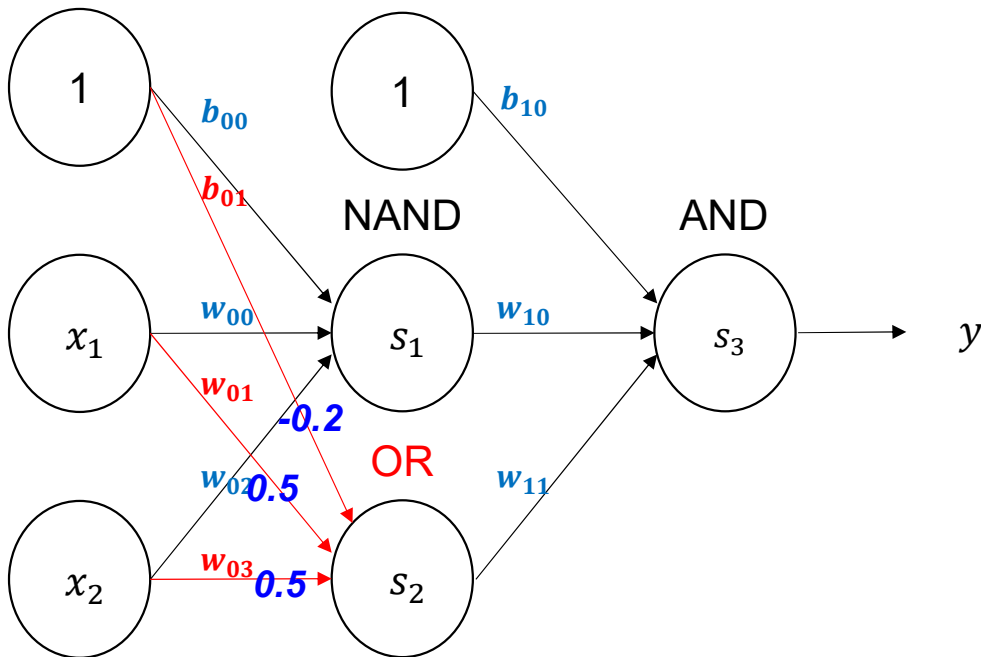
- $w_{00} = -0.5, w_{02} = -0.5, b_{00} = -0.7$ (NAND Gate 구현)
- S1 : NAND Gate



Multi-Layer Perceptron

■ XOR Gate

- $w_{01} = 0.5, w_{03} = 0.5, b_{01} = -0.2$ (OR Gate 구현)
- S2 : OR Gate

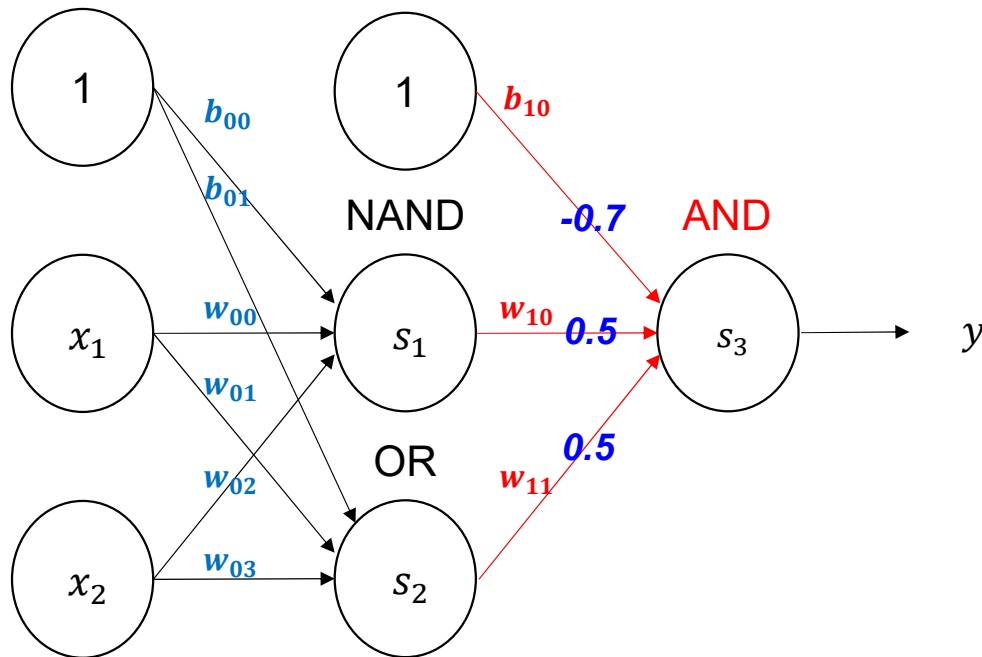


<Multi-Layer Perceptron>

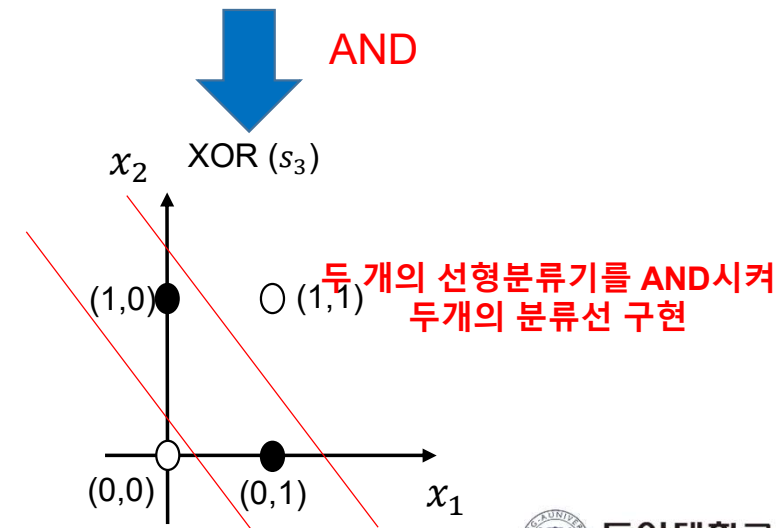
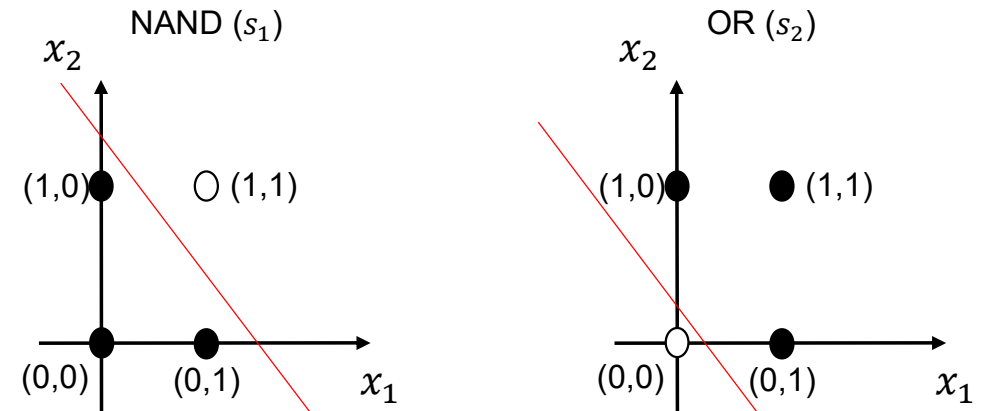
Multi-Layer Perceptron

■ XOR Gate

- $w_{10} = 0.5, w_{11} = 0.5, b_{10} = -0.7$ (AND Gate 구현)
- S1 AND S2 : S3 (XOR Gate)



<Multi-Layer Perceptron>





Questions & Answers

Dongsan Jun (dsjun@dau.ac.kr)

Image Signal Processing Laboratory (www.donga-ispl.kr)

Dept. of AI

Dong-A University, Busan, Rep. of Korea

