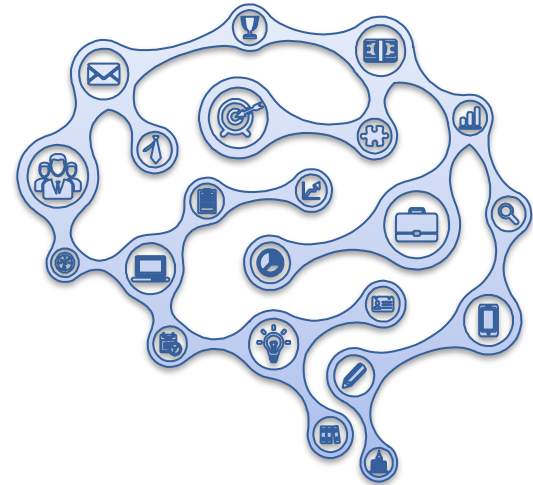


의료 Artificial Intelligence

강화 학습과 인공지능경망 (chap9,10)

2022.06.02



오늘 배울 내용 ...

1. 강화 학습, 인공신경망
2. 인공지능 실습
3. mblock 실습

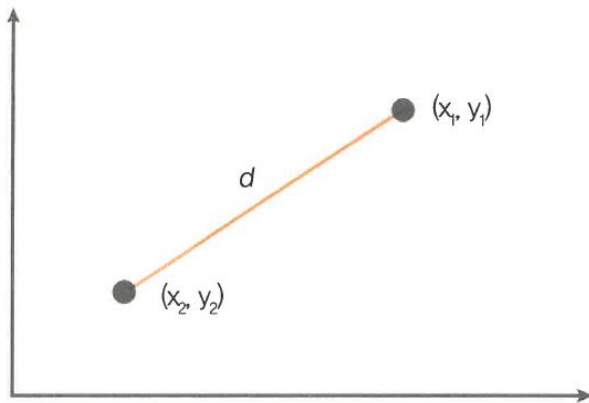
어렵지 않다
쉬운 것도 아니다



인공지능 이론

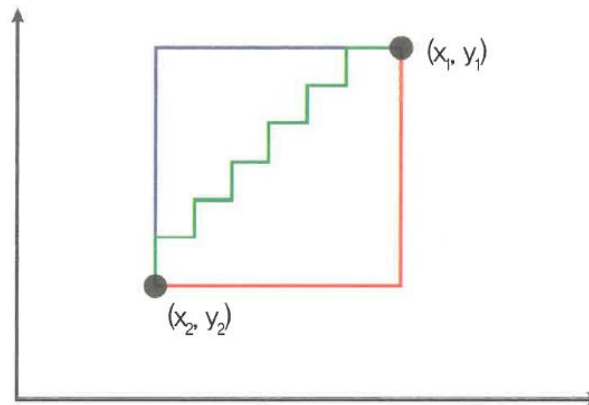
거리의 종류

유클리드 거리



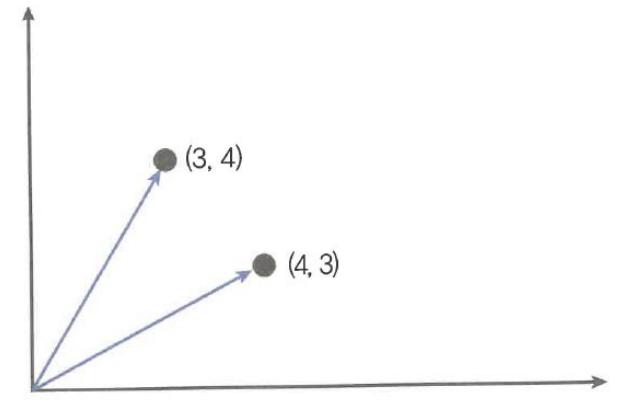
$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

맨해튼 거리



$$d = |x_1 - x_2| + |y_1 - y_2|$$

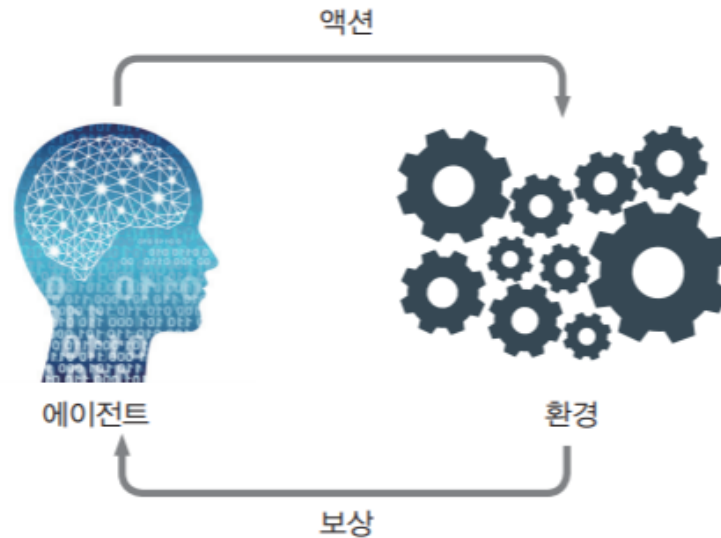
코사인 유사도



$$\cos\theta = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

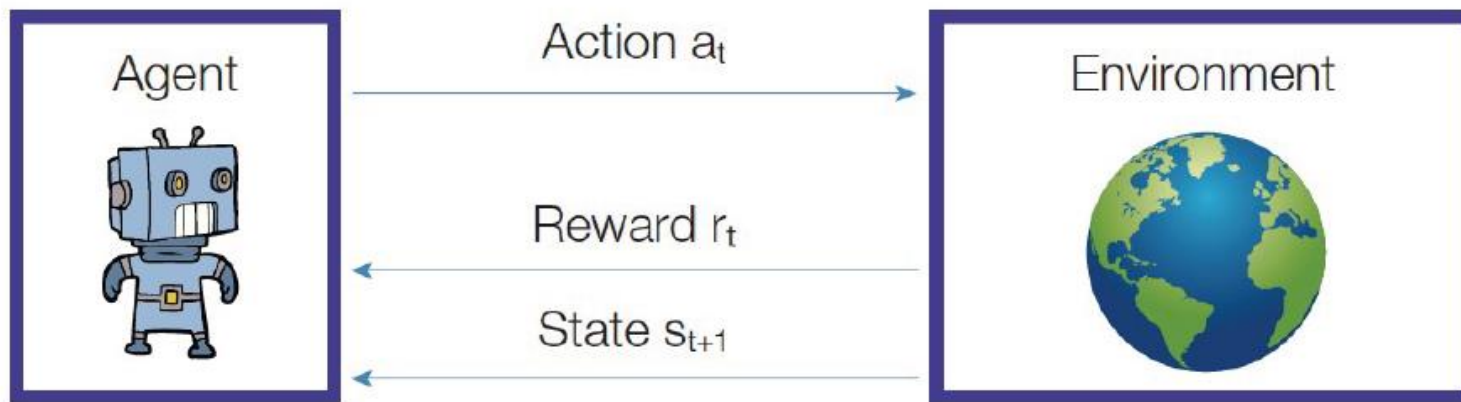
강화학습

- 강화학습은 준지도학습으로 행동한 결과(처리 결과)에 대해 보상(평가) 알고리즘을 제공하여 스스로의 행동을 개선하는 방법
- 시간이 지남에 따라 보상을 최대화하기 위해 에이전트가 특정 환경의 작업 공간에서 해결방안을 스스로 선택하도록 가르치는 머신러닝 방법



강화학습 - 주요 개념

- 에이전트Agent가 주어진 환경Environment에서 어떤 행동Action을 취하고 이에 대한 보상Reward을 얻고 지속적으로 상태State를 구축하면서 학습 진행 (보상을 최대화 하도록 학습 진행)



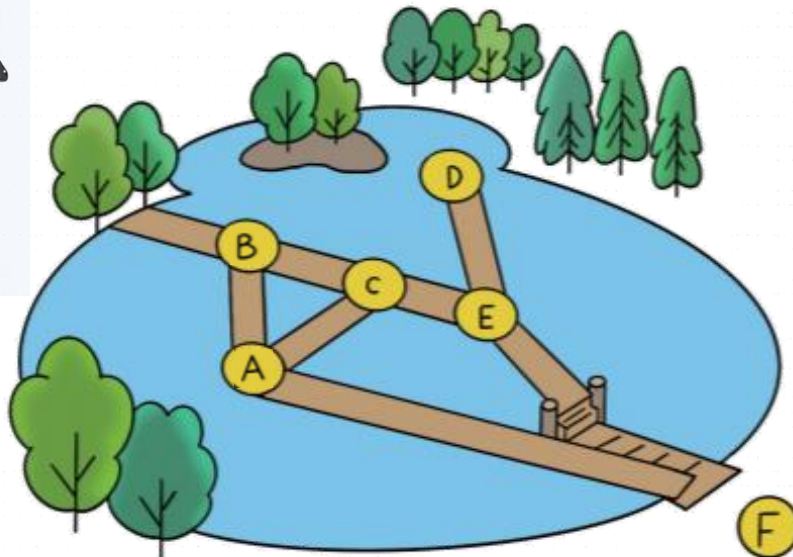
- 에이전트(Agent): 지정한 작업을 수행하기 위해 훈련하는 프로그램
- 환경(Environment): 에이전트가 조치를 수행하는 실제 또는 가상 세계
- 동작(Action): 에이전트에 의한 조치로 환경의 상태가 변경
- 보상(Reward): 긍정적이거나 부정적일 수 있는 행동의 평가 값

Q-Learning 알고리즘

Q-Learning : 보상값(R)을 알고 있을 때, 목표까지 도달하는 효용의 기대값(Q)를 계산하는 함수를 통해 최적의 정책을 찾을 수 있는 Q'지도'를 만드는 알고리즘

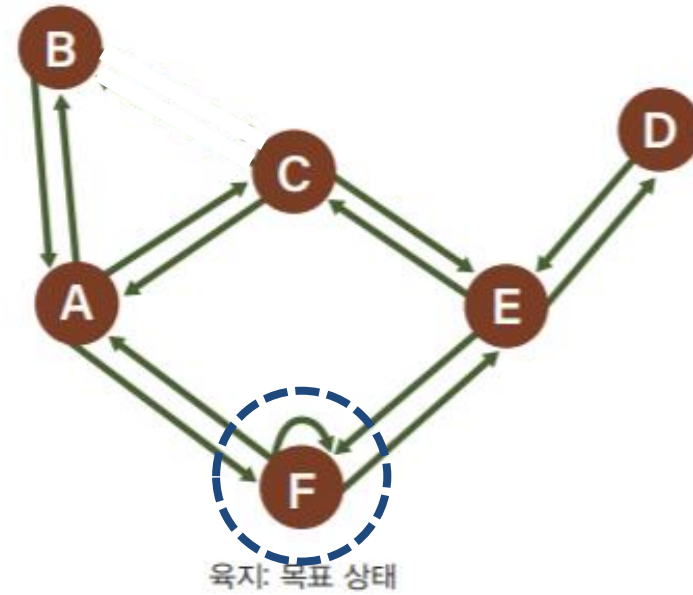
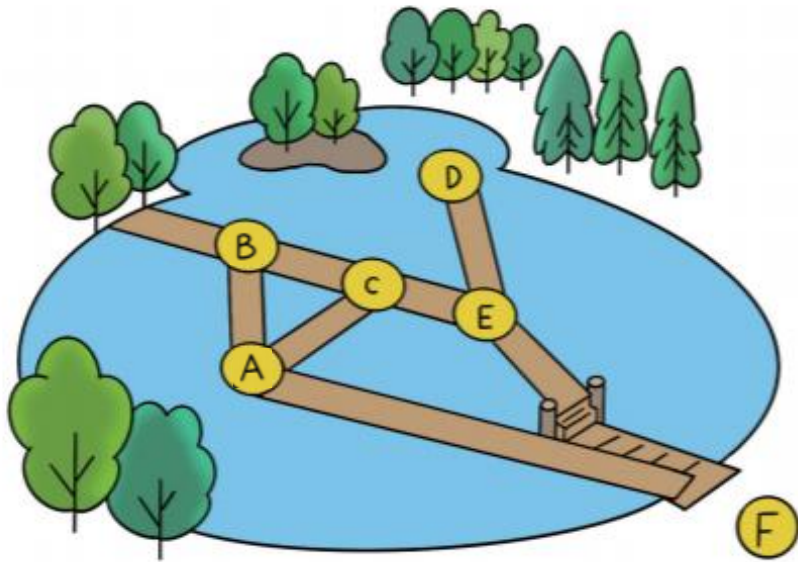
로봇이 호수 가운데서 육지로 이동하려고 한다.

→ A 또는 E로 나가야 F인 육지로 이동할 수 있다.



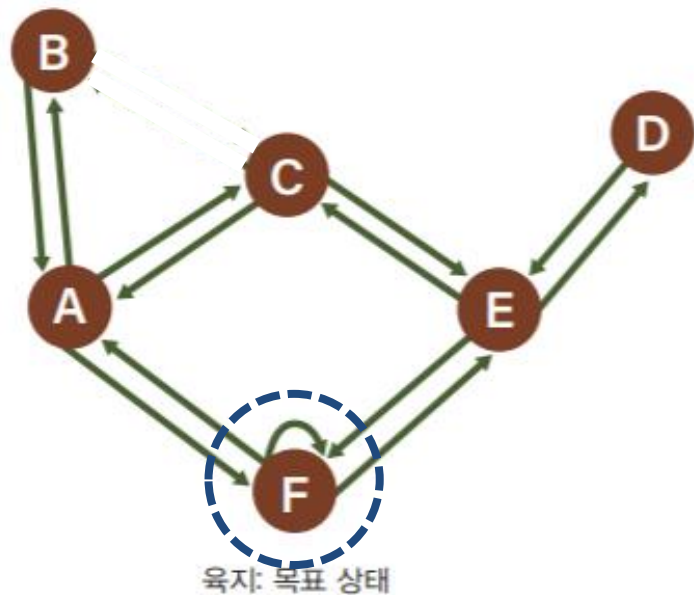
Q-Learning 알고리즘

문제를 추상화



Q-Learning 알고리즘

- 목표 : B에서 F로 이동
- 합리적인 경로(B-A-F)가 눈에 바로 보이지만, 계산만 가능한 로봇이라면 시행착오를 통해 길을 찾아야 한다



B	A	F	
G	C	E	D

Q-Learning 알고리즘

Q보드에 정보를 기록하는 규칙

① 초기 상태에서 Q보드는 모두 0으로 합니다.

② Q를 구하는 계산식을 설정합니다.

$$Q(\text{상태}, \text{액션}) = R(\text{상태}, \text{액션}) + \gamma \times \text{Max} (Q (\text{다음 상태}, \text{모든 액션}))$$

③ γ (감마) 매개변수와 보상을 결정합니다.

γ 는 할인율이라고 하며 범위는 0 ~ 1입니다. 0에 가까울수록 즉각적인 보상을 중시하는 경향이 있습니다.

④ 에피소드(학습)를 여러 번 반복하여 에피소드 종료시마다 Q보드에 값을 기록합니다.

⑤ 1개의 에피소드는 무작위로 지정된 상태(방)에서 시작하며, 최대의 보상(F)에 도착한 시점에서 종료합니다.

⑥ 1개의 에피소드는 다음과 같은 처리로 진행합니다.

1. 시작 위치를 정한다

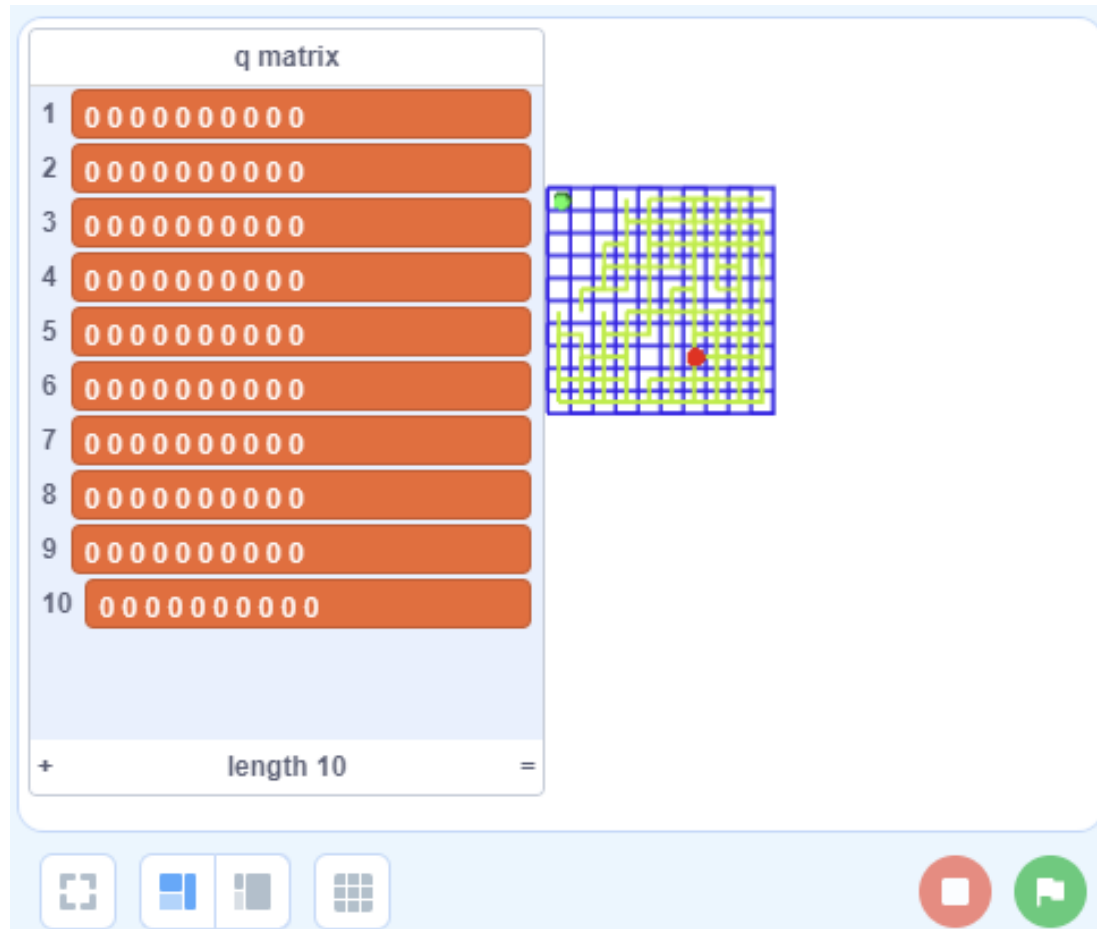
2. 현재 방에서 이동할 수 있는 액션 중에서 하나를 선택 (랜덤 or 높은 보상)

3. Q값을 ③의 식으로 계산하여 Q보드를 업데이트

4. 이동의 도착지가 'F'면 에피소드는 종료

5. 위 2번에서 다시 반복

Q-Learning 알고리즘 과정

mblock에서 day13_glearning.mblock 프로젝트 실행

지도학습과 강화학습 비교

· 지도학습과 강화학습의 차이점

구분	지도학습	강화학습
정적/동적	준비된 훈련 데이터에서 패턴을 탐색하고 배우는 것이 목표	시간의 흐름에 따라 데이터를 수집하는 과정을 수행함
정답	정답이 포함된 데이터를 활용해 학습과정에서 사용	정답이 명시적으로 제공되지 않음 (에이전트가 시행착오를 통해 학습함)
탐색	다른 답변을 탐색하지 않고도 훈련데이터에서 직접 답 적용	실시간으로 탐색해야 함 (환경 탐색, 보상을 얻는 새로운 방법 찾기, 이미 발견된 보상의 정책 활용)
결정 프로세스	하나의 예를 통한 하나의 학습 (단일 결정 프로세스)	특정 작업을 마치는 데 필요한 과정에서 의사결정 체인을 형성 (다중 결정 프로세스)

강화학습 사례

웹 페이지에 광고 배치

- 에이전트(프로그램): 페이지에 적합한 광고 수를 결정하는 프로그램
- 환경(공간): 웹 페이지와 사이버 공간
- 동작(결정): 세 가지 중 하나를 결정
 - ① 다른 광고를 페이지에 추가하기
 - ② 페이지에서 광고를 삭제하기
 - ③ 추가하거나 제거하지 않기
- 보상(수익): 매출이 증가하면 긍정적(양수), 매출이 떨어지면 부정적(음수)



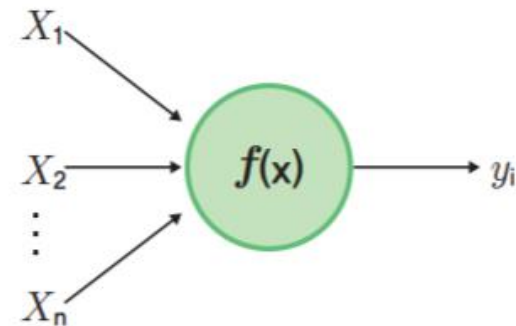
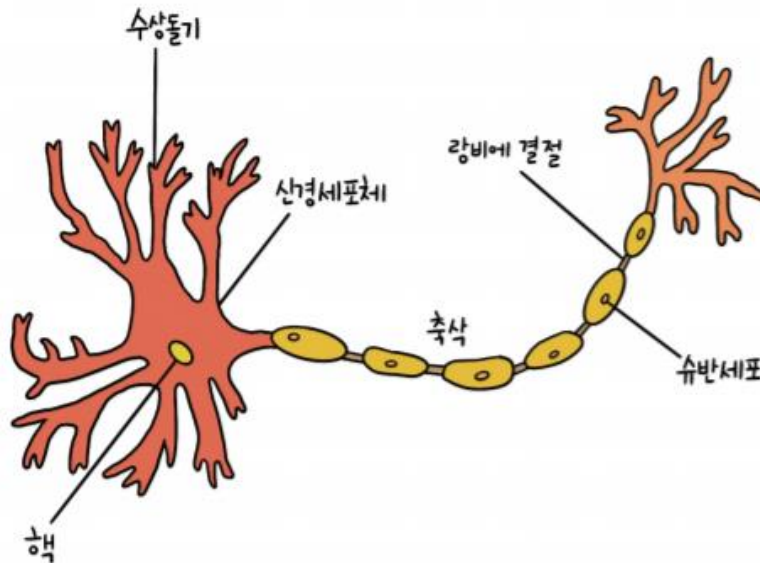
인공 신경망 - 최초 아이디어

- 인간 뇌 신경세포를 복잡한 스위치들이 연결된 네트워크로 표현

: 'McCulloch-Pitts 뉴런' 모델

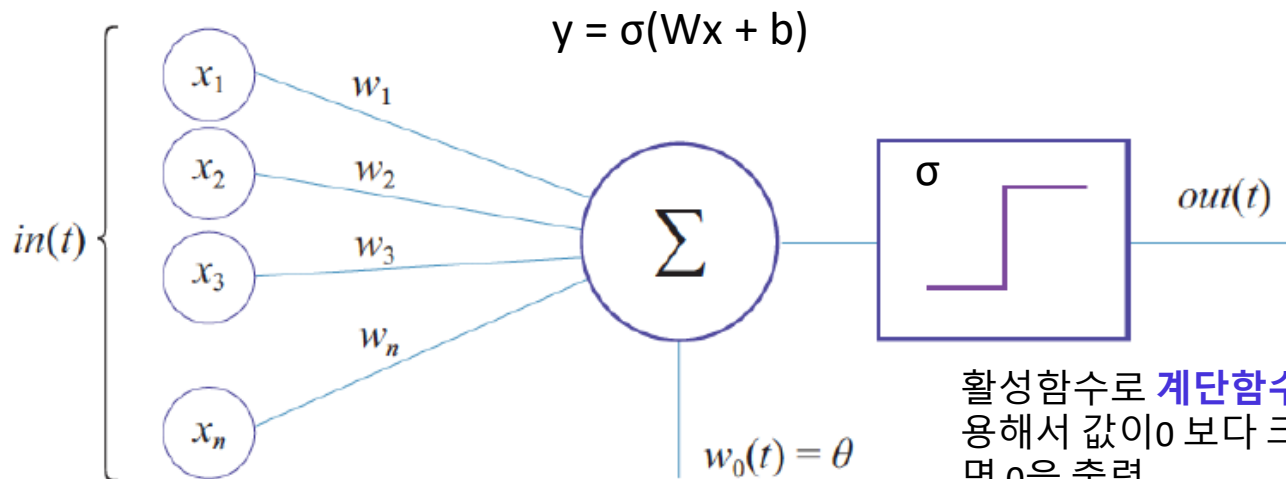
“A logical calculus of the ideas immanent in nervous activity” 논문 (1943)

- “컴퓨터도 인간의 뇌처럼 대량의 병렬처리 연산을 수행하도록 만들면 컴퓨터도 인간이 쉽게 할 수 있는 인지행동을 할 수 있지 않을까?”



인공 신경망 - 퍼셉트론

- 공학적 구현을 최초로 제안한 것은 1958년에 Frank Rosenblatt이 발표한
“The perceptron : A probabilistic model for information storage and organization in the brain” 논문
- 퍼셉트론Perceptron은 생물학적 뉴런을 공학적인 구조로 변형한 그림
입력층Input Layer $in(t)$ 과 출력층Output Layer $out(t)$ 을 가지고 있음
- 퍼셉트론은 입력층에서 인풋데이터 x 를 받고, 이를 가중치 W 와 곱한 후, 이 값에 바이어스Bias b 를 더함.
이 값을 활성화함수 σ 의 입력값으로 대입해서 출력층은 최종적으로 0 또는 1의 값을 출력



활성함수로 **계단함수** Step Function를 사용해서 값이 0 보다 크면 1, 0보다 작으면 0을 출력

인공 신경망 - 가중치

· Perceptron의 가중치(W) : Input의 중요도를 나타낸다

· 입력 데이터 (고려사항 데이터) 수 만큼 가중치(W) 필요

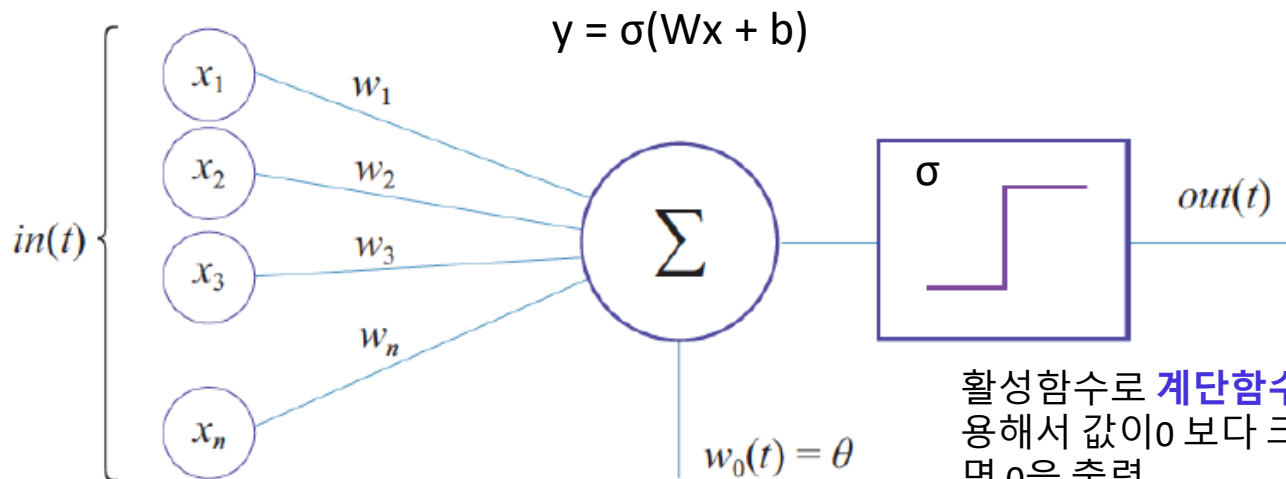
· 예를 들어, "주말에 집에서 나가 데이트를 할 것인가?"에 대한 의사결정모델을 Perceptron을 이용해서 만든다

이때 의사결정 고려사항은 다음과 같이 가정

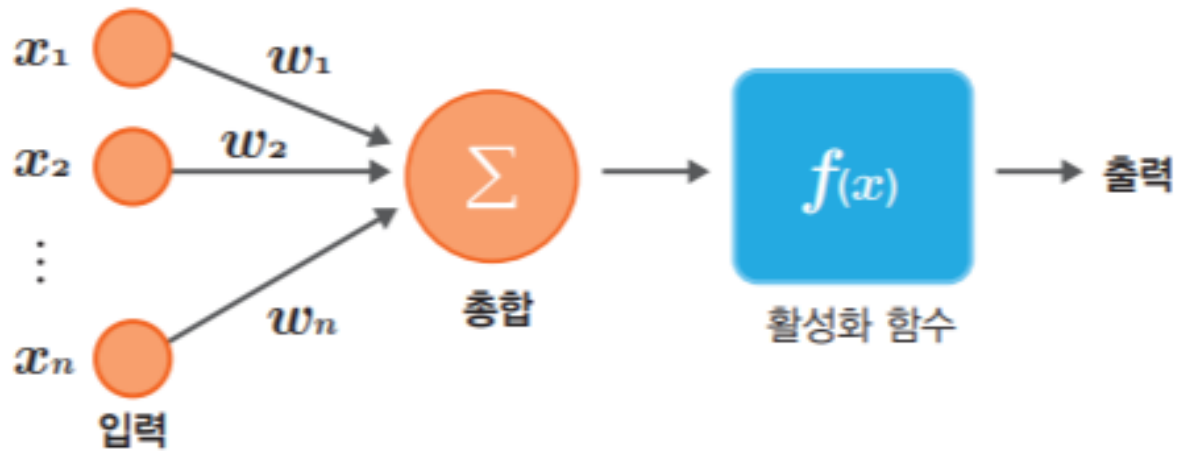
1. 날씨가 좋은가? (W1)
2. 이성 친구가 바쁜 일이 없는가? (W2)
3. 데이트 장소가 집에서 가까운가? (W3)

세 가지 고려사항 데이터를 Input(x)으로 넣음. 날씨를 가장 중요하게 고려한다면 W1=6, W2=2, W3=2 가중치 부여

데이트 장소가 집에서 가까운 것을 가장 중요하게 고려한다면 W1=2, W2=2, W3=6 가중치 부여

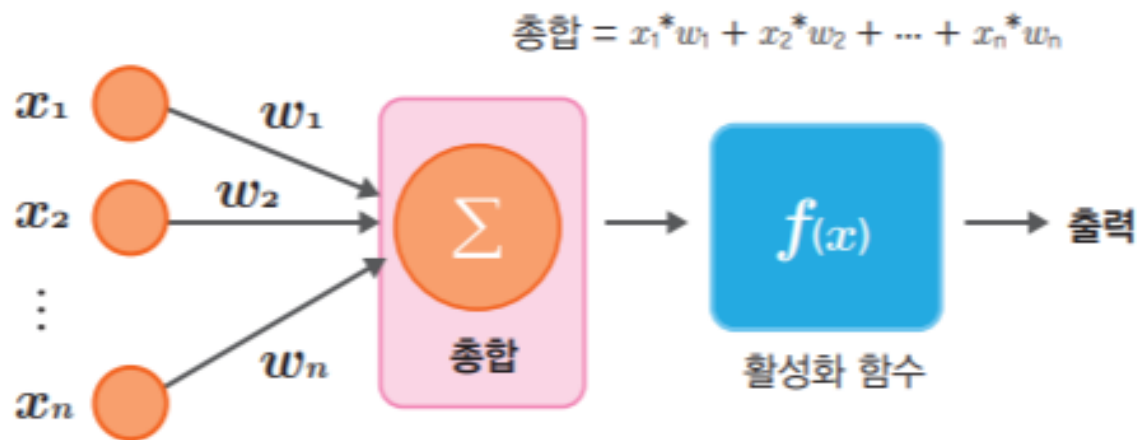


인공 신경망 계산 방법



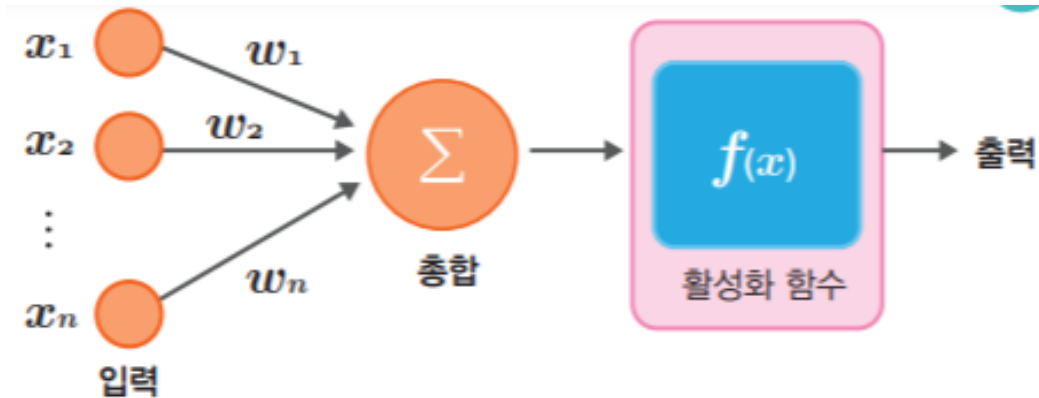
1. 입력 층에서 데이터 값을 입력 받음
2. 입력의 중요도에 따라 가중 값으로 크기를 변경

인공 신경망 계산 방법

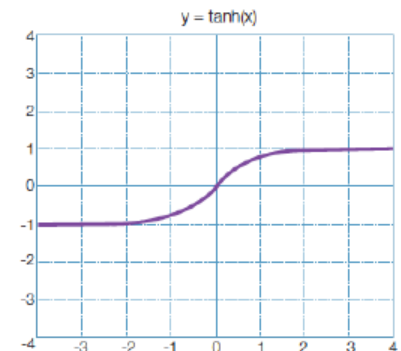
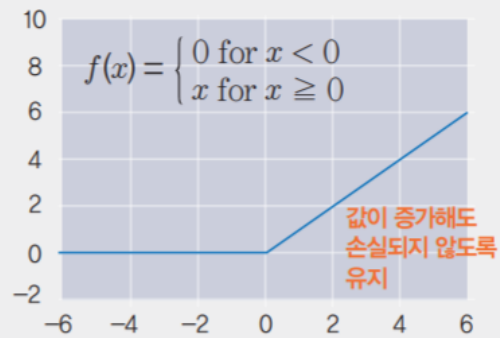
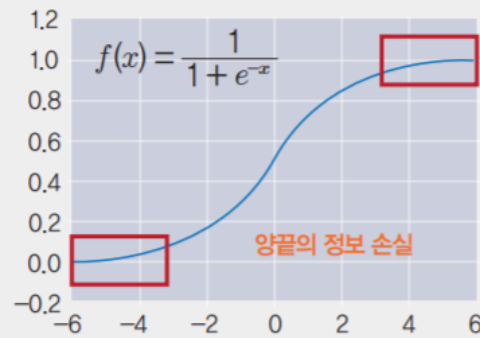
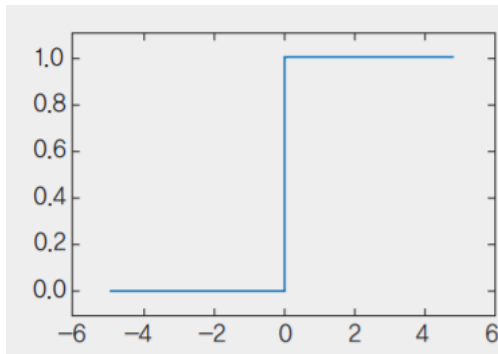


3. 입력 x 와 가중 값 w 를 곱하고 모두 더함

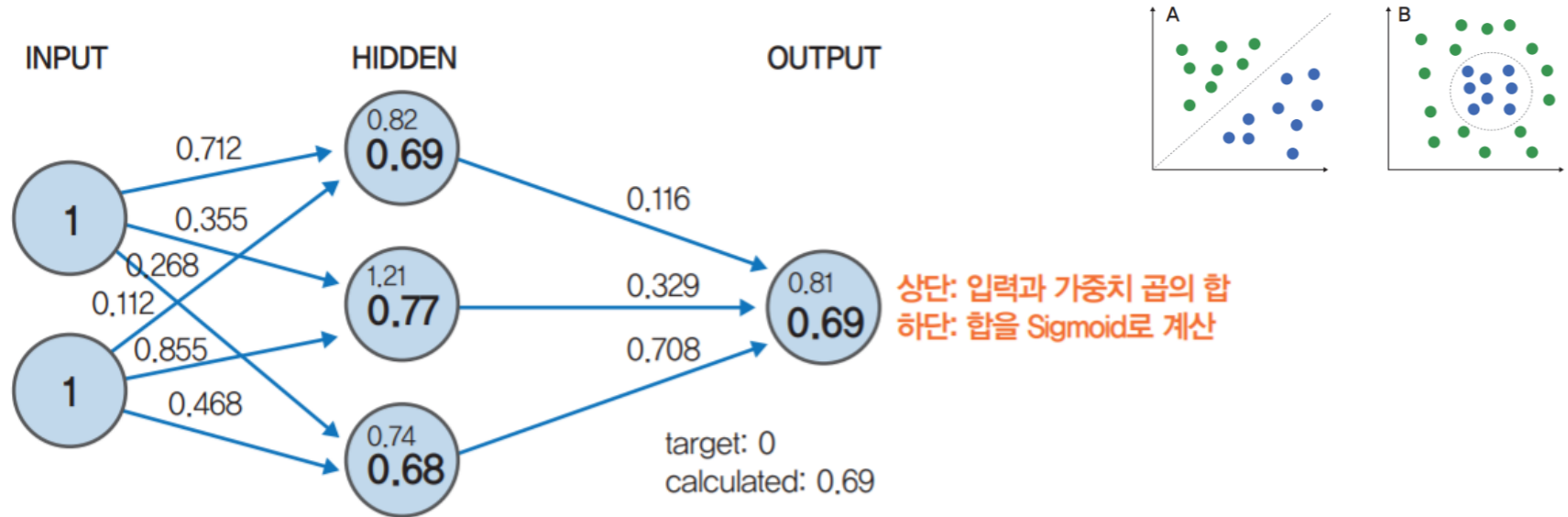
인공 신경망 계산 방법



4. 합한 값을 활성화 함수로 출력 여부를 계산
(계단함수, 시그모이드, ReLU, 탄젠트 함수 등을 활용)



인공 신경망 계산 방법



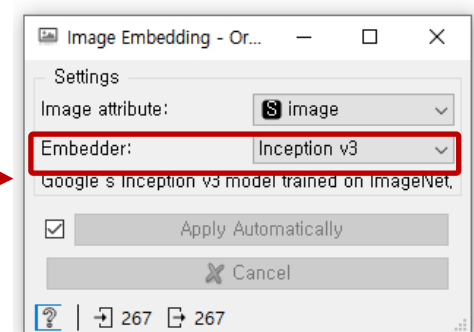
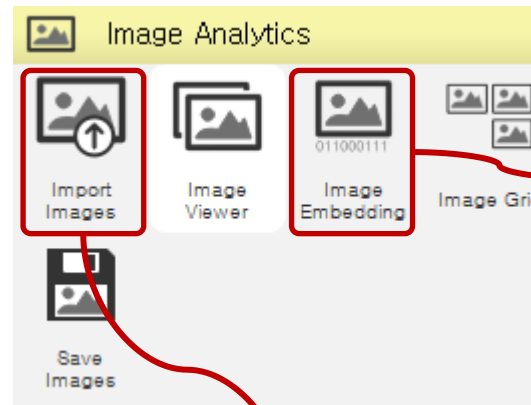
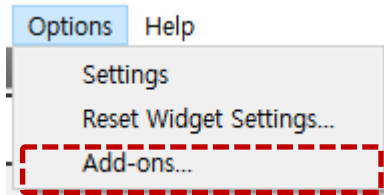
5. 여러 개의 입력값과 가중치 곱의 합을 활성화 함수로 계산해 여러 개 뉴런에서 출력

인공지능 실습

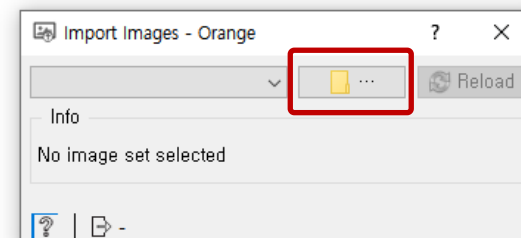
인공지능 실습

실습 1 : 신경망을 이용한 이미지 분류

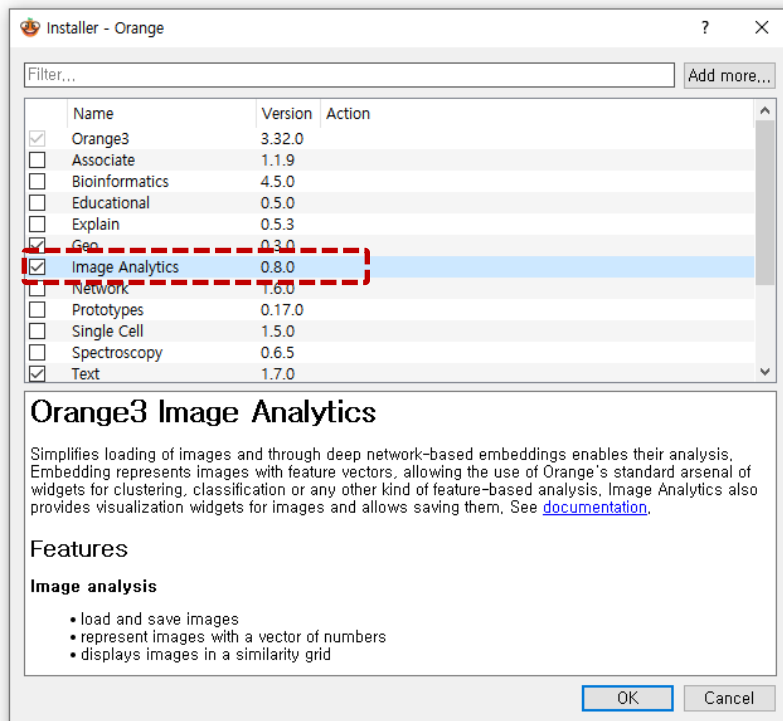
- 데이터 모델 : Image Analytics



이미지 전처리 작업
: 이미지 데이터를 분석에 적합한
형태로 가공 (벡터화)
- inception v3 선택

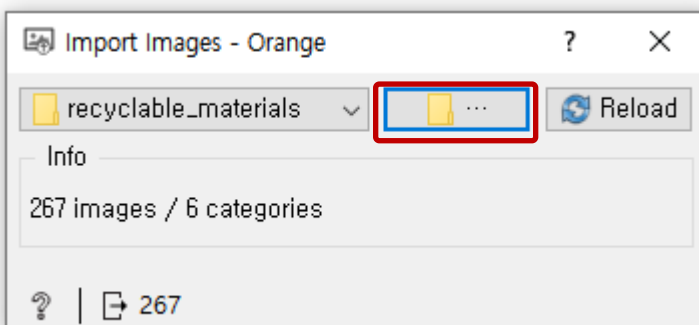
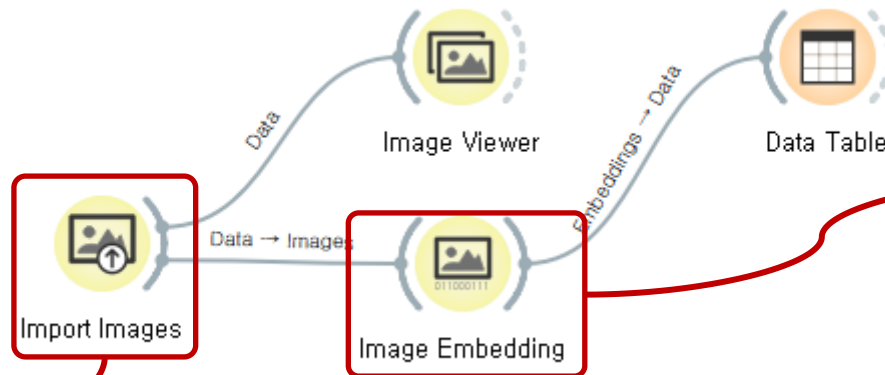


recyclable_materials 폴더 선택

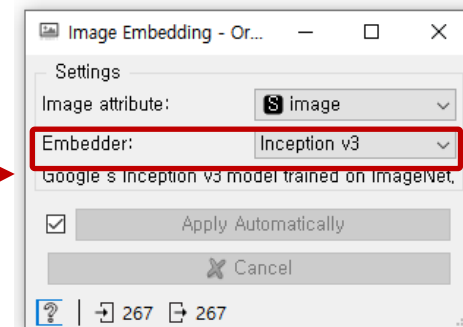


인공지능 실습

- 이미지 데이터 읽기 : 'recyclable_materials'



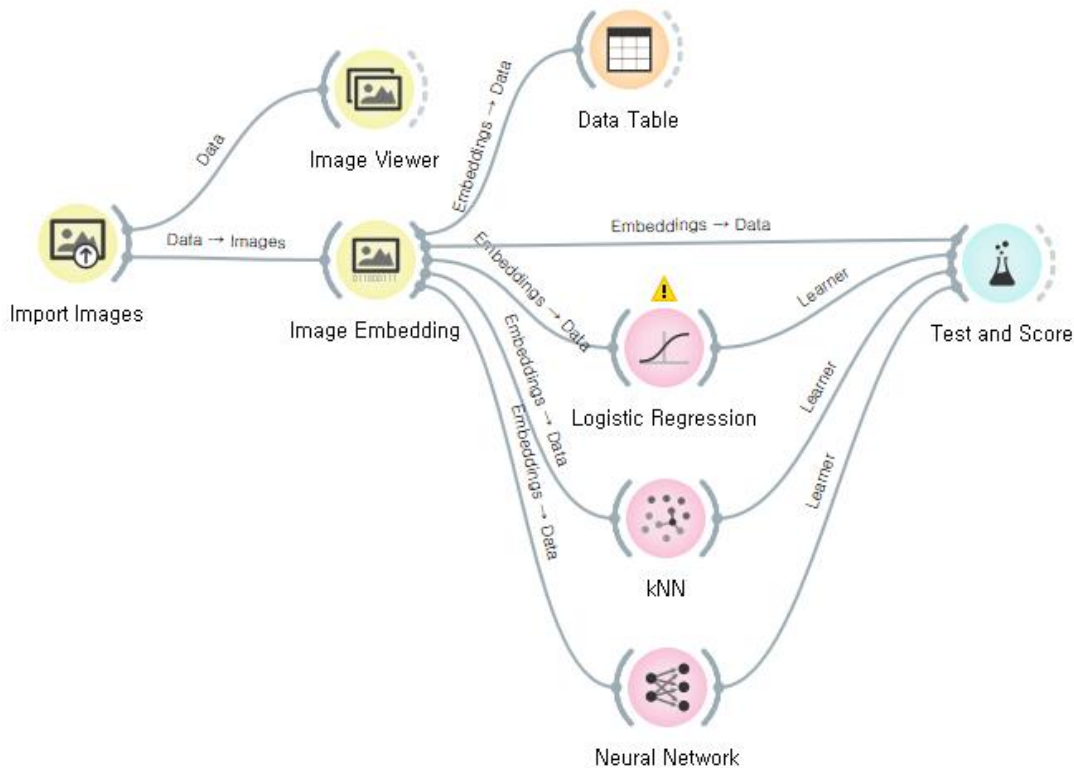
recyclable_materials 폴더 선택



inception v3 선택

인공지능 실습

- 이미지 분류 학습 : Logistic Regression, kNN, Neural Network



타겟 선택

Filter...
(None, show average over classes)
can
glass
paper
plastic
plastic_bag
styrofoam

Test and Score - Orange

☒ Cross validation
Number of folds: 5
☒ Stratified
☐ Cross validation by feature
☐ Random sampling
Repeat train/test: 10
Training set size: 80 %
☒ Stratified
☐ Leave one out
☐ Test on train data
☐ Test on test data

Evaluation results for target (None, show average over classes)

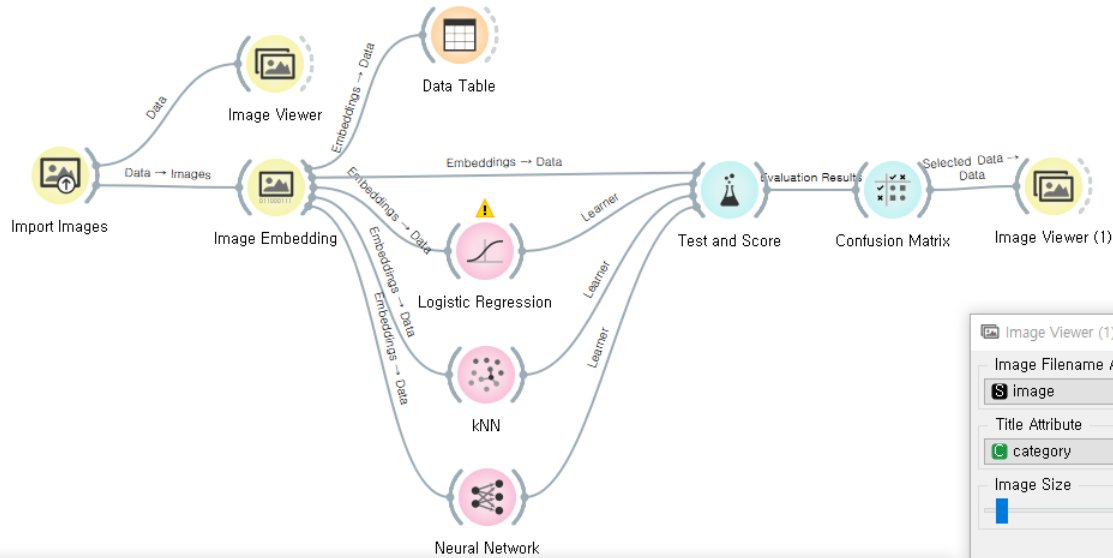
Model	AUC	CA	F1	Precision	Recall
kNN	0.964	0.798	0.786	0.791	0.798
Neural Network	0.981	0.876	0.872	0.875	0.876
Logistic Regression	0.990	0.918	0.916	0.920	0.918

Compare models by: Area under ROC curve ☐ Negligible diff.: 0.1

	kNN	Neural Network	Logistic Regression
kNN		0.090	0.046
Neural Network	0.910		0.066
Logistic Regression	0.954	0.934	

Table shows probabilities that the score for the model in the row is higher than that of the model in the column. Small numbers show the probability that the difference is negligible.

인공지능 실습



Confusion Matrix - Orange

Learners: Logistic Regression, kNN, Neural Network

Show: Number of instances

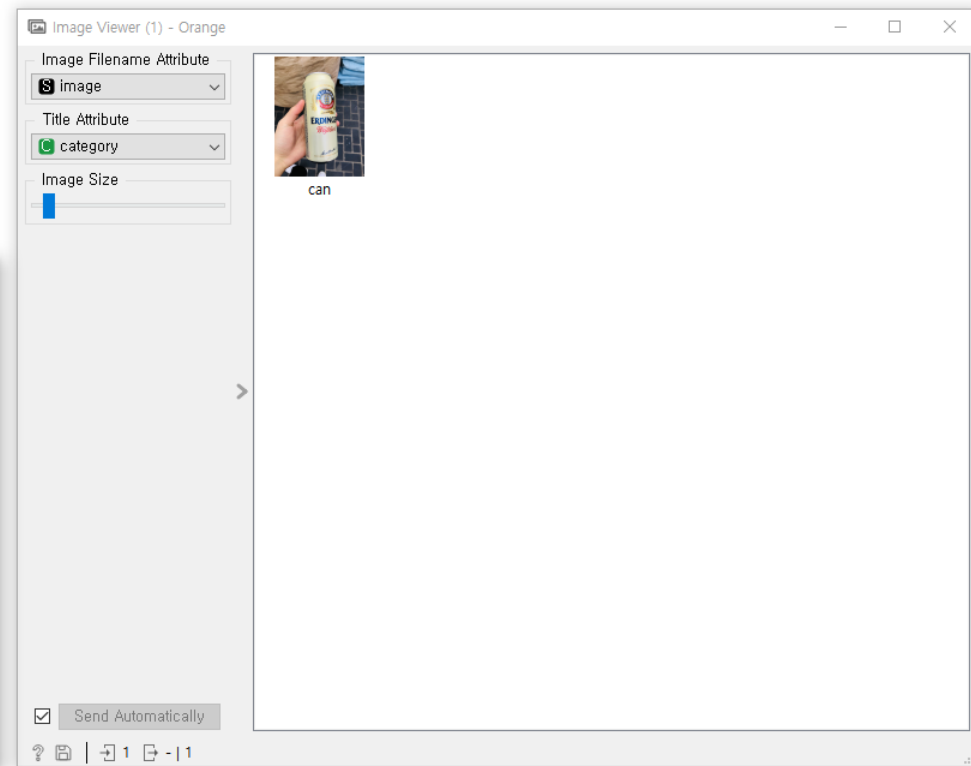
Actual \ Predicted							Σ
	can	glass	paper	plastic	plastic_bag	styrofoam	
can	17	1	1	5	0	0	24
glass	0	55	0	1	0	0	56
paper	2	0	43	3	3	1	52
plastic	1	1	0	77	1	3	83
plastic_bag	0	0	1	0	33	0	34
styrofoam	0	0	2	7	0	9	18
Σ	20	57	47	93	37	13	267

☒ Predictions ☐ Probabilities

☒ Apply Automatically

Select Correct Select Misclassified Clear Selection

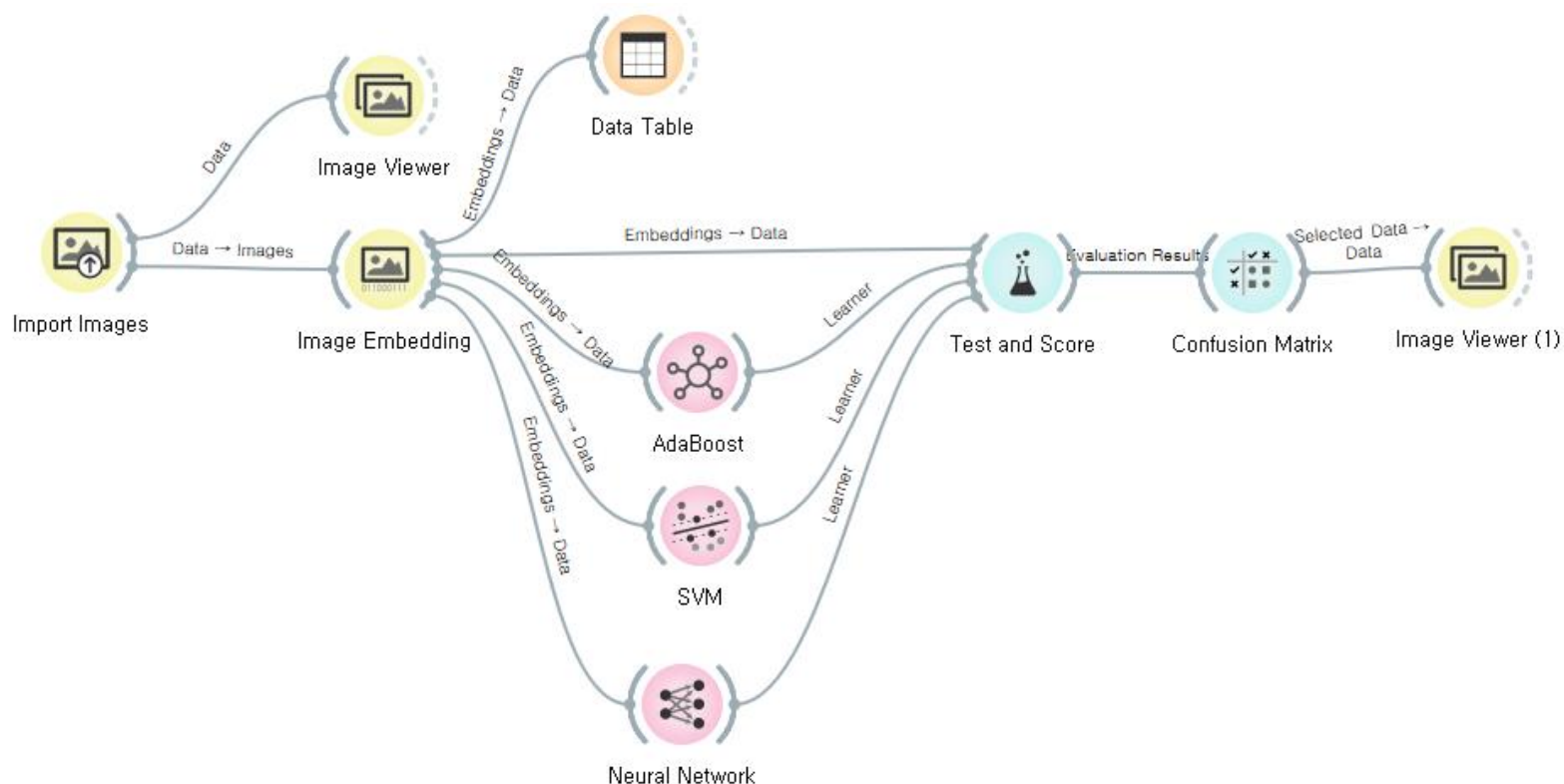
3×267 1 | 267



인공지능 실습

실습 2 : 신경망을 이용한 이미지 분류

- 'kawibawibo_Train' 데이터 셋을 읽어 SVM, AdaBoost, Neural Network 학습
- 'kawibawibo_Test' 데이터 셋으로 test 함

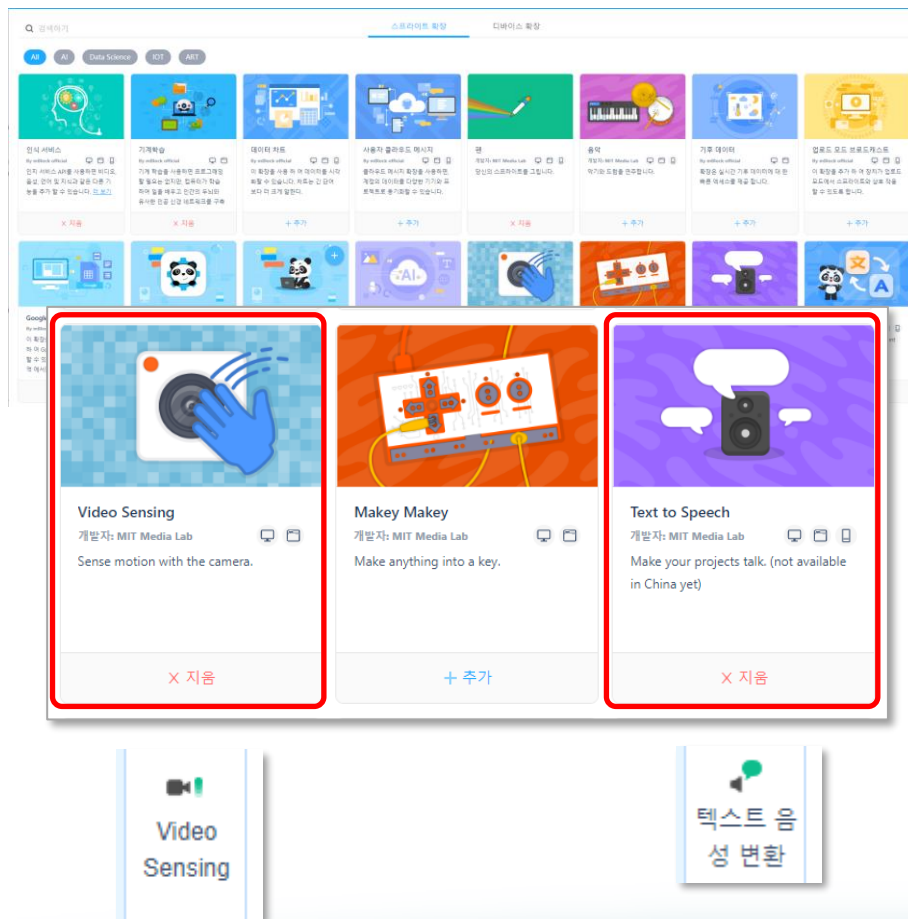


MBlock 실습

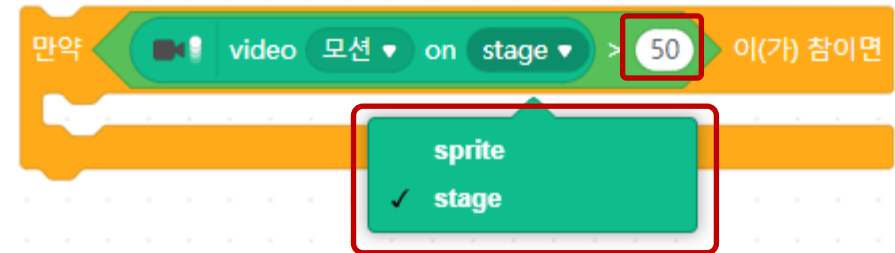
MBlock 인공지능 실습

실습 : 비디오 센싱을 이용한 AR

비디오센싱, TTS 확장 추가

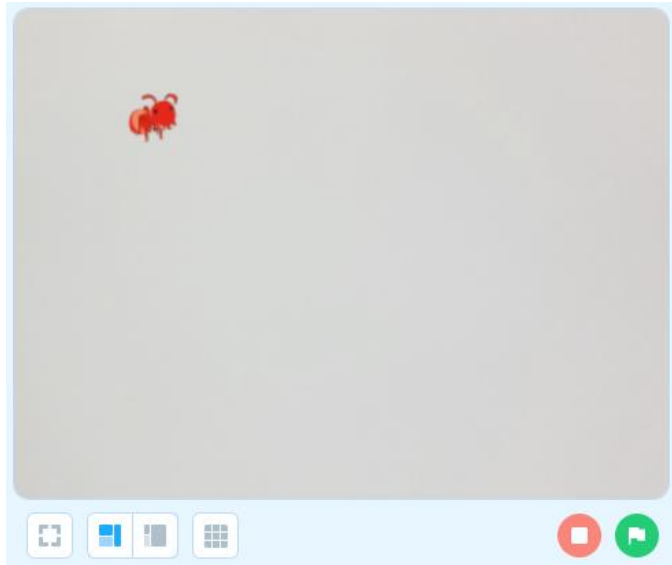


감지 민감도



동작 감지 대상

MBlock 인공지능 실습



스프라이트

Ant

X

Y

-138

100

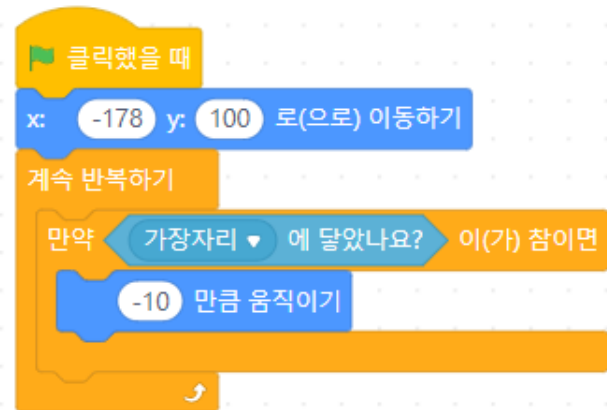
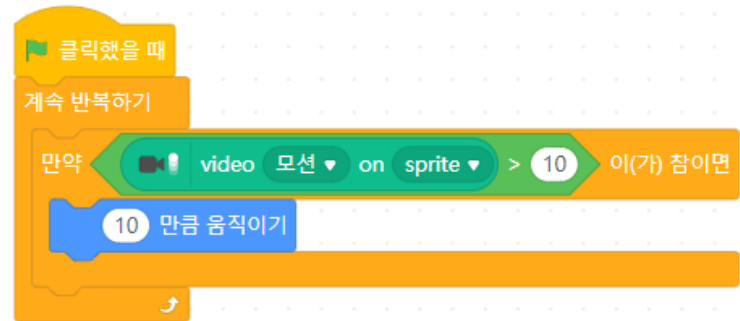
크기

방향

30

90

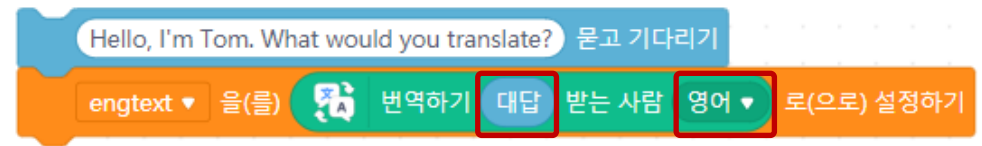
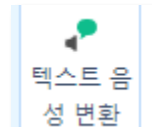
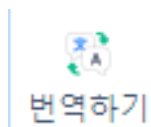
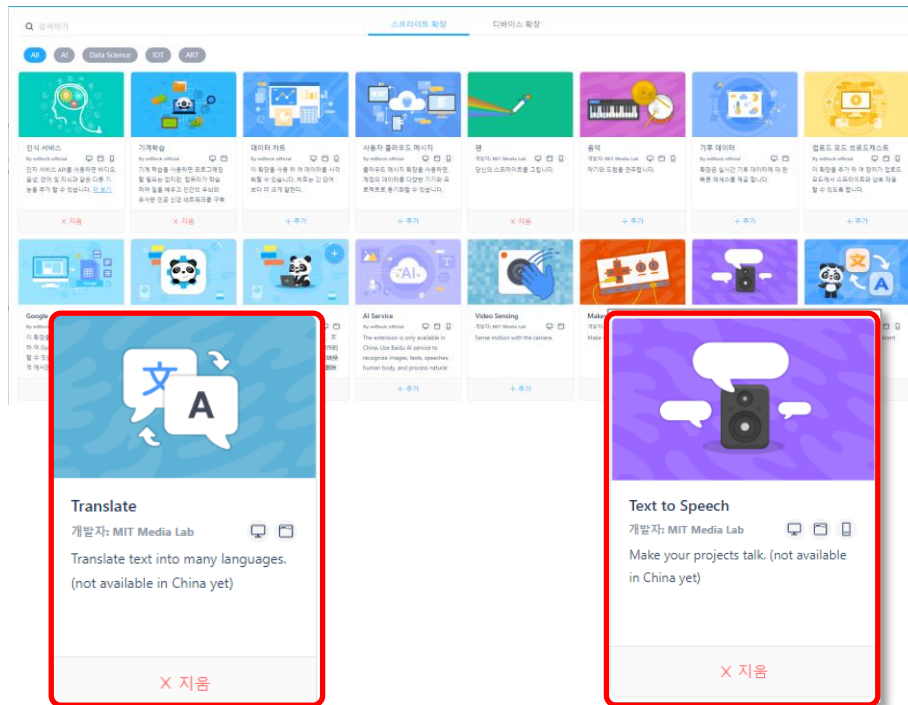
보이기



MBlock 인공지능 실습 2

실습 : 번역하기 앱

Translate, TTS 확장 추가

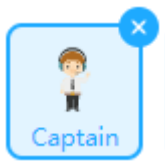
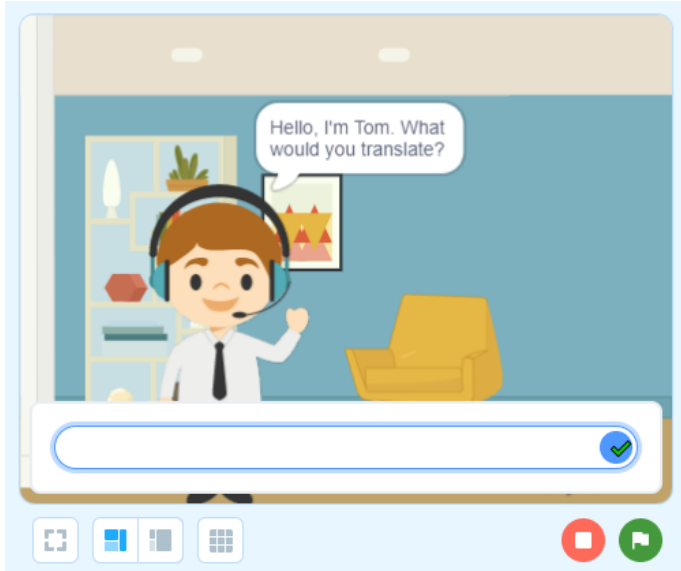


번역할 텍스트

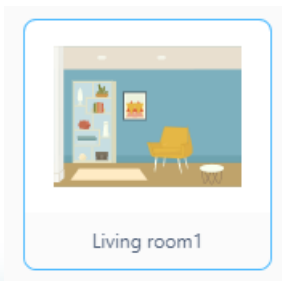
번역 언어



MBlock 인공지능 실습 2



코스틀 수: 2



팀 활동
