



AI 프로그래밍

융합학과 권오영

oykwon@koreatech.ac.kr

과정소개

교과목소개

- ❖ 강사: 권오영 (구내전화 041-560-1354; email: oykwon@koreatech.ac.kr)
- ❖ AI프로그래밍:이 교과목은 인공지능 서비스를 개발하기 위한 기본적인 프로그래밍 방법을 학습한다. 코딩없이 알고리즘적 사고만으로 인공지능 서비스를 개발하는 방법을 학습하고, 실제 프로그래밍을 하기위하여 파이썬 언어와 프로그래밍 개발도구들을 학습한다. 기계학습에 필요한 기본적인 수학기개념을 학습한다.
- ❖ 평가: 출석 20%, 과제 25%, 중간고사 25%, 기말고사 30%
평가방법은 변경될 수 있음

강의 계획

| | | | |
|-----|------------------------------------|------|---|
| 1주차 | 강의소개 (알고리즘적 사고) | 9주차 | 파이썬 turtle 코딩 |
| 2주차 | 파이썬 개발 환경(Thonny, Jupyter, colab) | 10주차 | 파이썬 (가시화, matplotlib, seaborn, folium 등) |
| 3주차 | 파이썬 (기본문법, 데이터타입, 변수, 제어문) | 11주차 | 소프트웨어개발사이클 및 버전관리(git, github사용) |
| 4주차 | 파이썬 (기본자료구조,리스트,튜플,사전 등, 함수) | 12주차 | 매트릭스 calculus, 벡터와 선형방정식 활용 이해 |
| 5주차 | 파이썬 (예외처리, 디버깅, 테스트) | 13주차 | AI 프로그래밍1 (선형회귀) |
| 6주차 | 파이썬 프로그램 작성 | 14주차 | AI 프로그래밍2 (스팸메일 필터 구축) |
| 7주차 | 파이썬 (객체지향프로그램, 모듈 및 파일처리) | 15주차 | 기말고사 |
| 8주차 | 중간고사 | | |

프로그래밍 적합도 설문조사

인공지능

인공지능

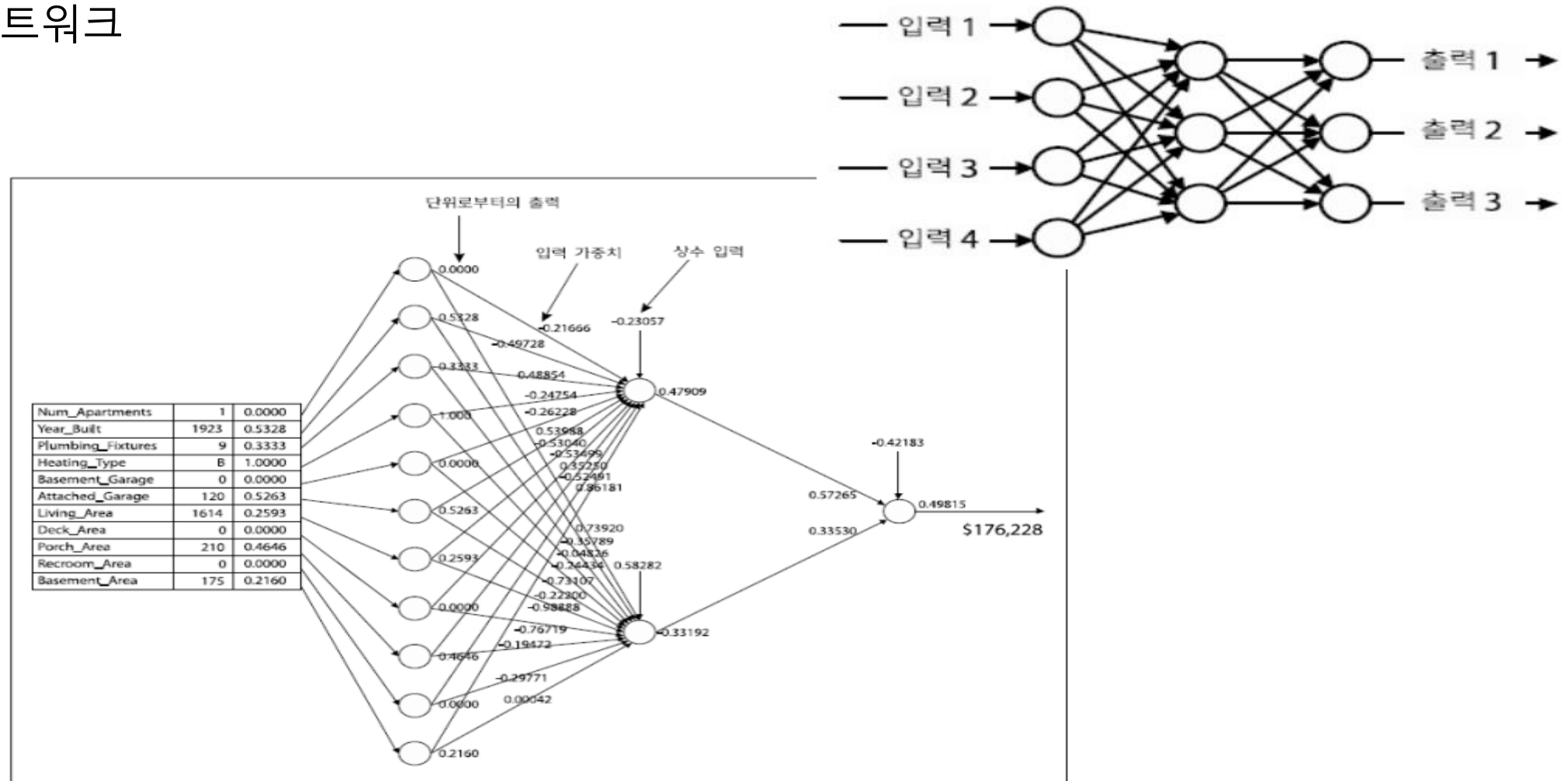
- ❖ 인공지능 : 인간의 학습능력, 추론능력, 지각능력, 언어이해능력등을 컴퓨터 프로그램으로 구현한 기술
- ❖ 위키의 정의(<https://ko.wikipedia.org/wiki/인공지능>)
 - 초기 인공지능 연구에 대한 대표적인 정의는 다트머스 회의에서 존 매카시가 제안한 것으로 "**기계를 인간 행동의 지식에서와 같이 행동하게 만드는 것**"이다.
 - **강인공지능(범용인공지능)** : 어떤 문제를 실제로 사고하고 해결할 수 있는 컴퓨터 기반의 인공적인 지능을 만들어 내는 것에 관한 연구다.
 - ✓ 인간의 사고와 같이 컴퓨터 프로그램이 행동하고 사고하는 인간형 인공지능.
 - ✓ 인간과 다른 형태의 지각과 사고 추론을 발전시키는 컴퓨터 프로그램인 비인간형 인공지능.
 - **약인공지능(weak AI)** : 어떤 문제를 실제로 사고하거나 해결할 수는 어떤 면에서 보면 지능적인 행동으로 볼 수 있는 인공지능
 - ✓ 주로 미리 정의된 규칙의 모음을 이용해서 지능을 흉내내는 컴퓨터 프로그램을 개발

인공신경망

- ❖ 인간의 신경세포를 흉내낸 시스템
- ❖ 1943년 Warren McCulloch와 Walter Pitts
 - 예일 대학교 신경심리학자, 논리학자
 - 생물학적인 뉴런의 작동에 설명하기 위한 간단한 모형을 제안
- ❖ 1950년대, 공학자들은 맥컬록과 피츠의 연구에 기초한 퍼셉트론(perceptrons) 모형을 사용
- ❖ 1968년 MIT 교수 Papert와 Minsky
 - 단순한 신경망의 이론적 한계 지적
 - 1970년대까지 침체기
- ❖ 1982년 캘리포리아공대 John Hopfield
 - 역전파 알고리즘(Back propagation)
 - 이전 방법의 이론적 한계들을 극복
 - 인공신경망의 르네상스

인공신경망

❖ 다층네트워크



출처: 김종우, 김선태, 경영을 위한 데이터마이닝, 한경사, 2009

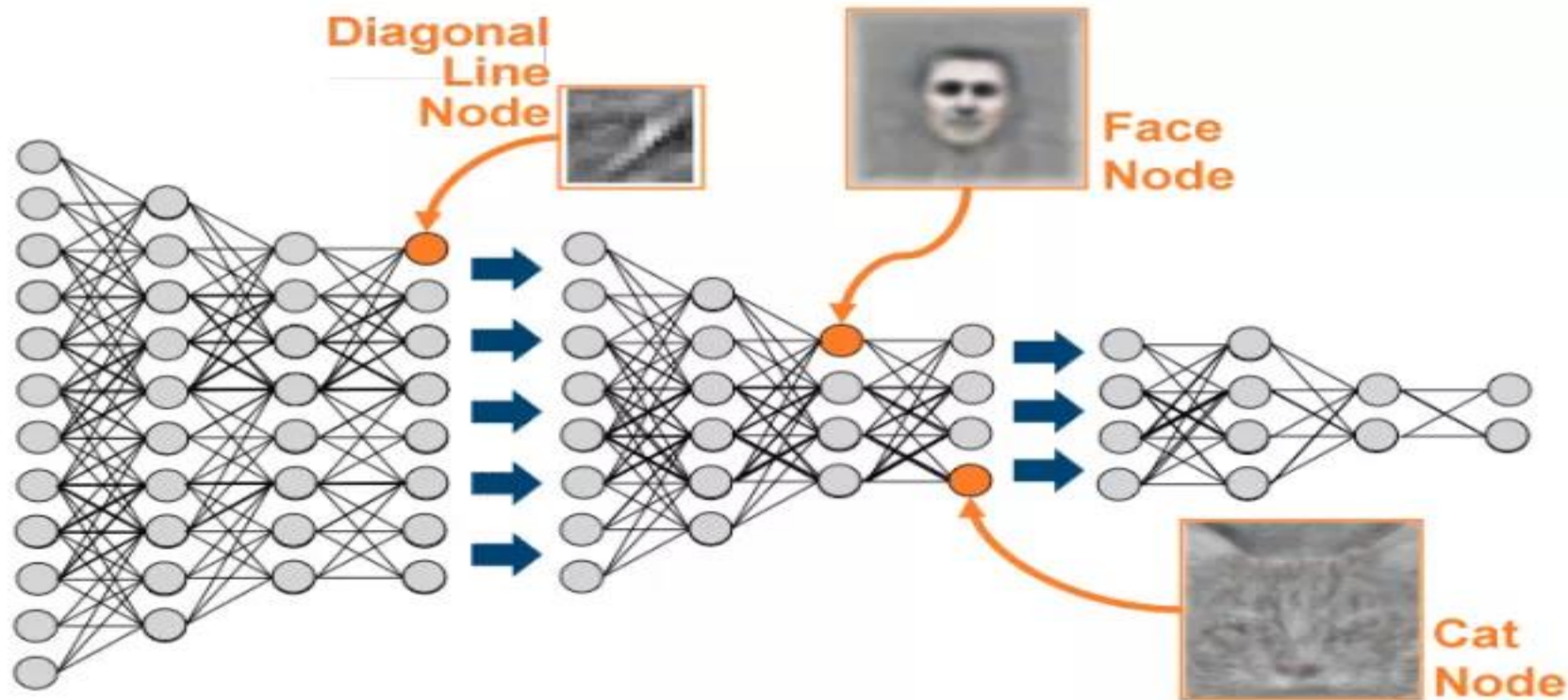
인공신경망

- ❖ 1980년대 인공신경망에 대한 연구는 연구실에서 상업계로 전이
 - 사기성 신용카드 거래 인식
 - 수표 금액 인식
 - 영상처리, 인공지능, 제어 등 다양한 분야에 적용
- ❖ 1990년대에 이르러 그 연구가 포화 상태에 이르고, 이내 한계가 보이기 시작하더니 곧 암흑기를 만남
- ❖ 2006년 Geoffrey Hinton
 - 데이터의 전처리과정(pre-training)을 통해 local minima에 빠지는 문제를 해결
 - “A fast learning algorithm for deep belief nets” 라는 논문
 - 인공신경망의 각 층을 먼저 비지도 학습방법(unsupervised learning)을 통해 잘 손질해주고, 그렇게 전처리한 데이터를 여러 층 쌓아올려 인공신경망 최적화를 수행하면 ‘이 산이 아닌가 봐?’ 없이 훌륭한 결과를 만들어 낼 수 있다는 것을 보임
 - Deep Learning

인공신경망

❖ 구글의 고양이 인식 (deep learning)

- 구글은 2012년 1,000대의 컴퓨터로 1,000만 개의 유튜브 이미지를 딥 러닝으로 분석해 사람과 고양이를 구분해 냈다.



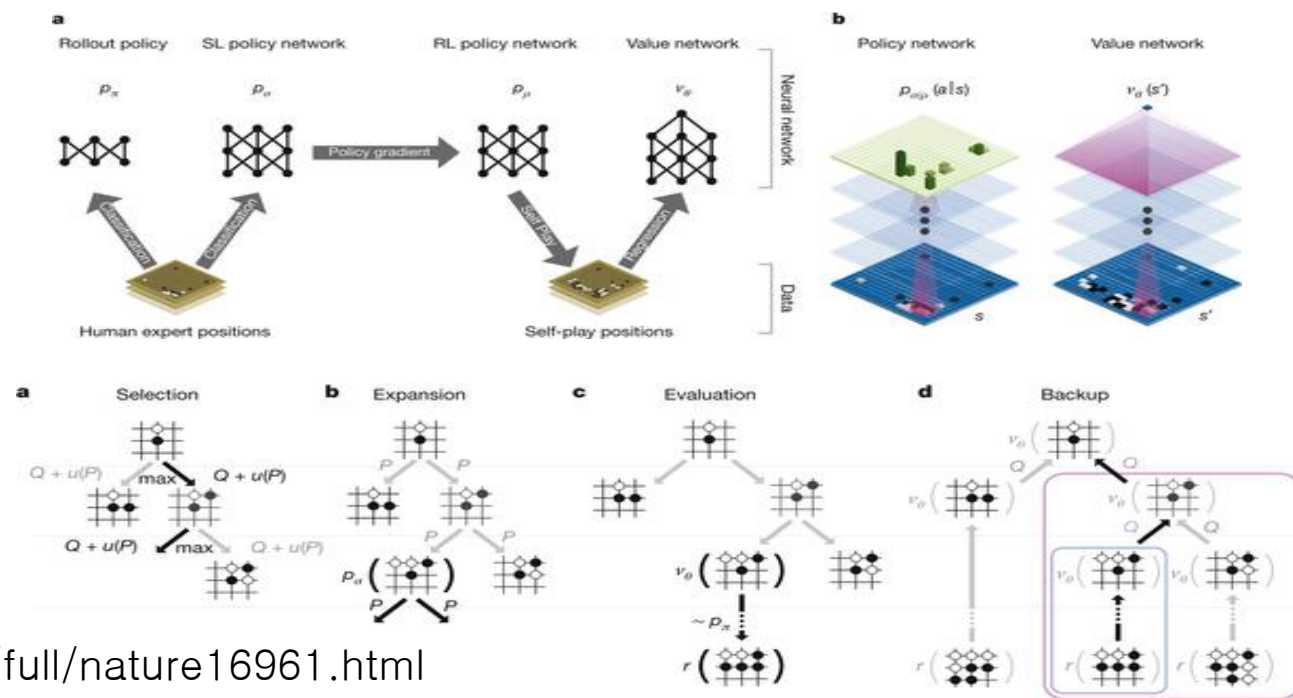
인공지능

- ❖ 1997년 IBM의 딥 블루 컴퓨터
 - 인간 체스 챔피언을 3.5-2.5로 승리한 인공지능을 채택한 컴퓨터
 - 가능한 모든 경우를 조사하여 다음 수를 결정하기 때문에 엄청난 계산처리 능력이 필요
 - ✓ 딥 블루는 30개의 노드로 구성된 컴퓨터이고 특별히 설계한 480개의 VLSI 체스 칩 장착 (1초당 200,000,000 개의 위치를 계산)
- ❖ 2011년 미국 퀴즈쇼 '제퍼디'에서 우승한 인공지능 컴퓨터 '왓슨'
 - 전 세계 백과사전과 위키피디아, 뉴욕타임스 아카이브, 성경 등 2억 페이지에 달하는 정보를 스스로 자기학습하며 수십만 가지의 질문에 답을 내놓음
 - '왓슨'은 2013년 미국 뉴욕의 MSKCC병원에서 암 환자를 위한 맞춤형 치료에 활용
 - ✓ 의학문헌 검토, 진료기록분석등을 수행 암 진단 보조
 - ✓ 유방암진단 정확도 91%이상 (전문의 초기 오진비율 20%~44%)
- ❖ 수많은 판례를 정보로 처리해 개별 사례에 맞는 법률적 조언을 제공하는 인공지능이 등장하기 시작

인공지능

❖ 알파고 (구글)

- 1202개의 CPU와 176개의 GPU 활용 (1378:1의 게임)
- Deep Neural Network, tree search 사용
- 프로바둑기사의 기보 16만건 확보해 3000만개 이상의 착점 학습
- 100만번(1000년)
대국을 4주에 학습
- 타 인공지능바둑프로그램
대비 99.8%의 승리
- 2016년 1월
유럽 챔피언 판후이
2단에게 5:0 승리



<http://www.nature.com/nature/journal/v529/n7587/full/nature16961.html>

인공지능

- ❖ 인공지능의 발전 단계 (미국경제학자 타일러 코웬 교수의 예측)
인간보다열등 → 인간과 동등 → 인간을 보조 → 인간을 대체(?)
 - 체스가 좋은 예임
 - ✓ 1997년 인간이 패배
 - ✓ 2000년대 프리스타일 가능 (인간고수, 인공지능, 인간+인공지능)
 - ✓ 2010년대 인공지능들간의 대전
- ❖ MIT의 브린올프슨과 맥아피 교수 ‘2차 기계 시대’ 선언
 - 1차 기계 시대: 대량 생산 기계가 단순 육체 노동을 대체한 시대
 - 2차 기계 시대: 로봇과 인공지능이 복잡한 육체노동, 나아가 지식 노동마저 대체하는 시대

*<https://www.youtube.com/watch?v=EK2iSPjryRM>
(Humans need not apply.)*

기계학습

The Anatomy of Machine Learning

**Computer
Science**



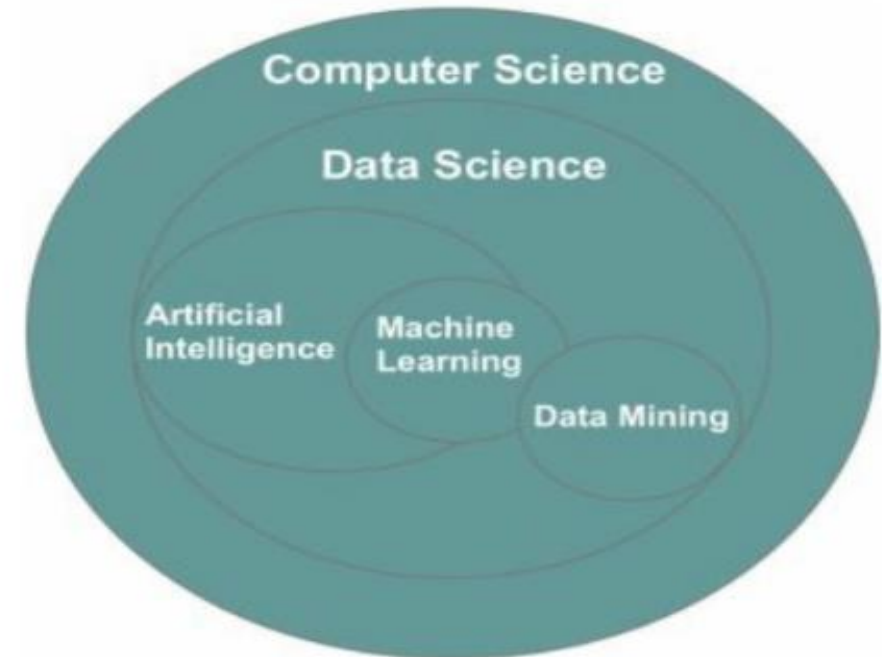
**Data
Science**



AI



**Machine
Learning**



인공지능-머신러닝-딥러닝

인공지능(AI)

컴퓨터가 사람처럼
생각하고, 판단하게
만드는 기술

머신러닝(ML)

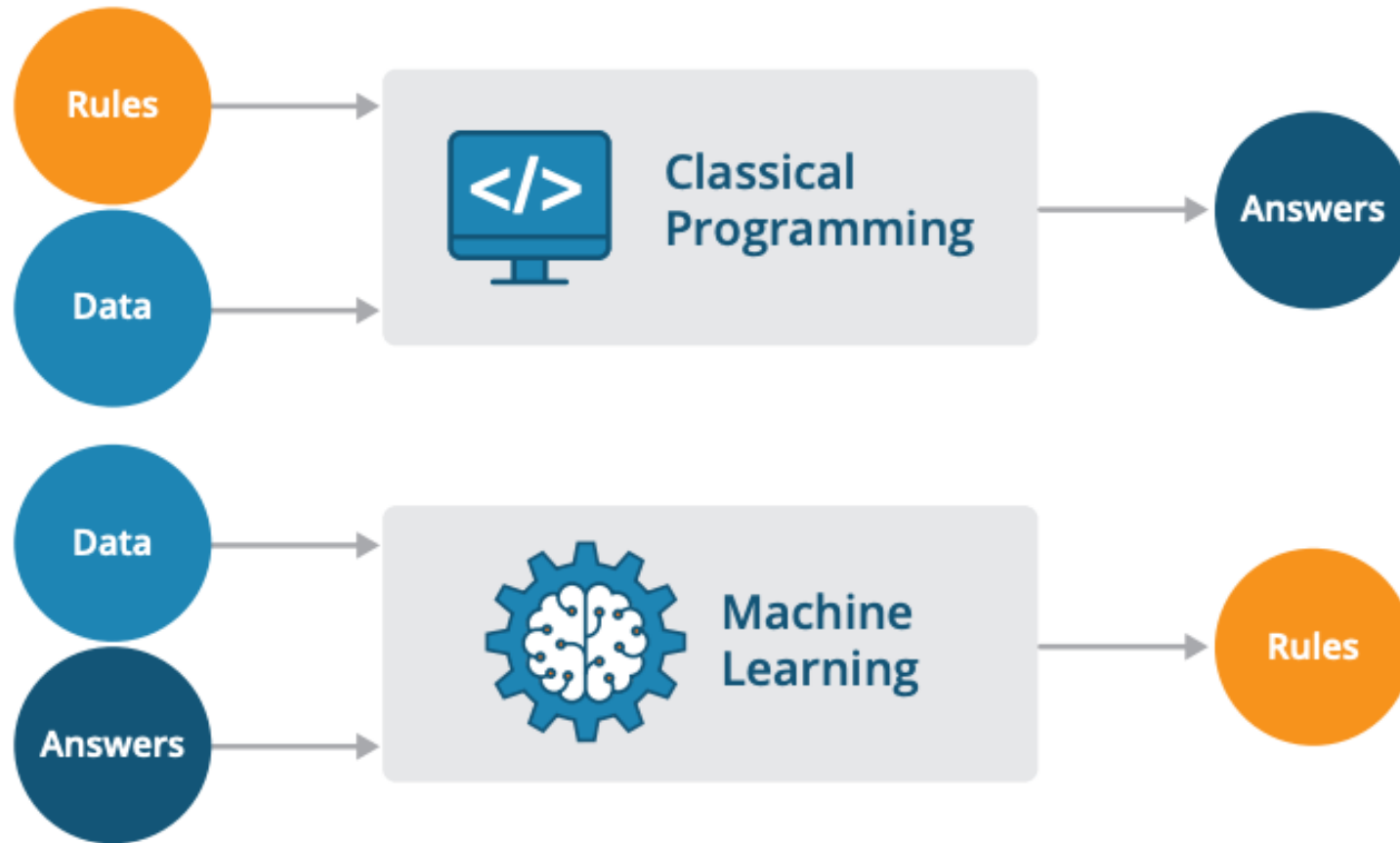
인간의 학습능력과
같은 기능을 컴퓨터에
부여하기 위한 기술

딥러닝(DL)

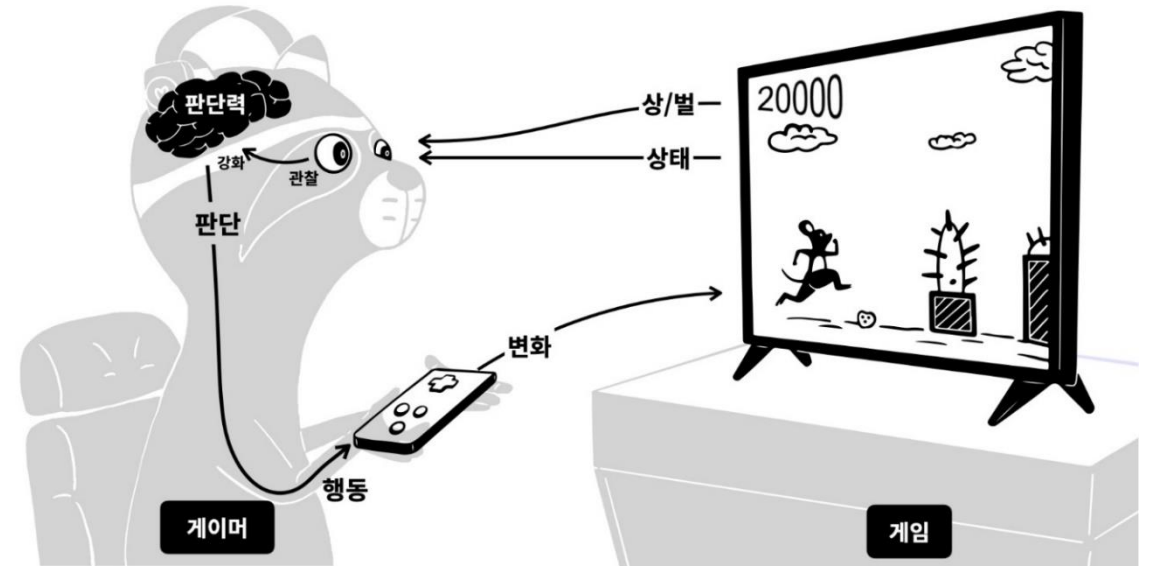
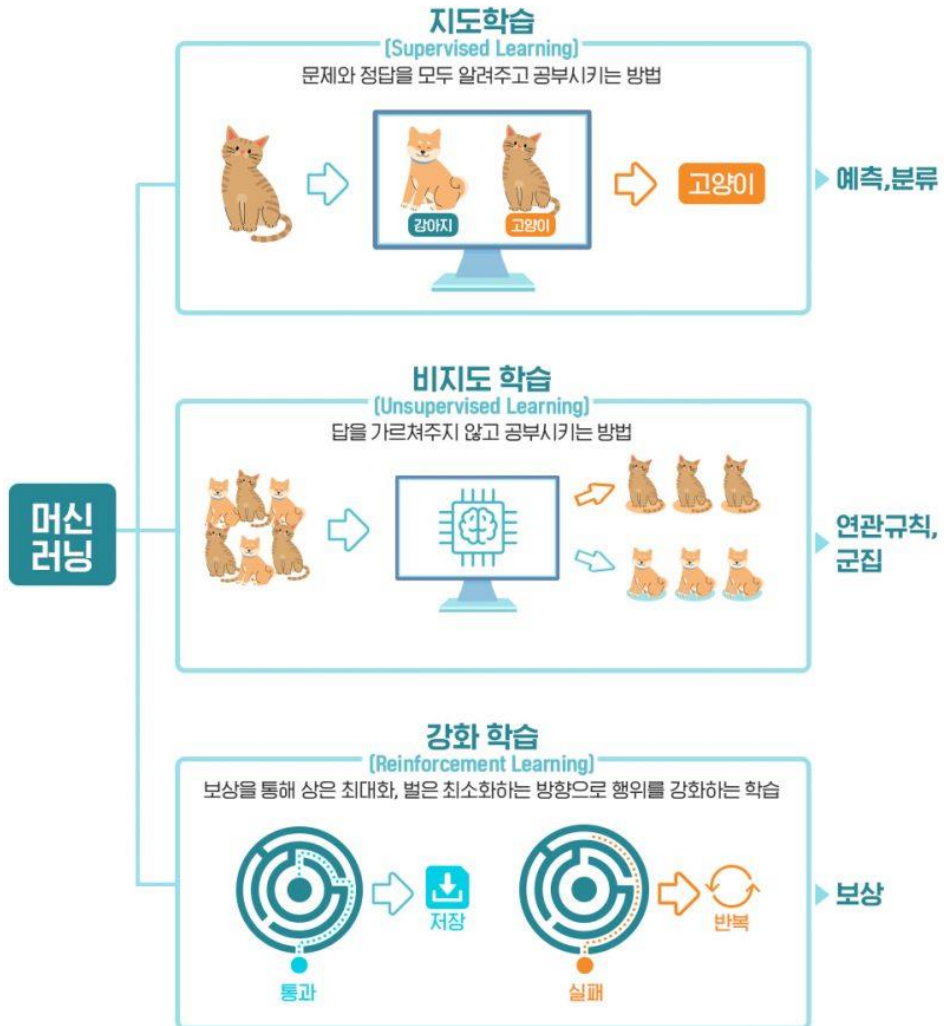
- ✓ 인공 신경망을 기반으로 한 머신러닝 방법론 중 하나
- ✓ 빅데이터를 기반으로 스스로 학습/판단하는 기술

Classical Programming vs ML

❖ Classical Programming vs ML



기계학습방법



- 게임 → 환경(environment)
- 게이머 → 에이전트(agent)
- 게임화면 → 상태(state)
- 게이머의 조작 → 행동(action)
- 상과 벌 → 보상(reward)
- 게이머의 판단력 → 정책(policy)

[출처] <https://opentutorials.org/course/4548/28949>

[출처] https://live.lge.co.kr/live_with_ai_01/

Types of machine learning

❖ Supervised(or predictive) learning

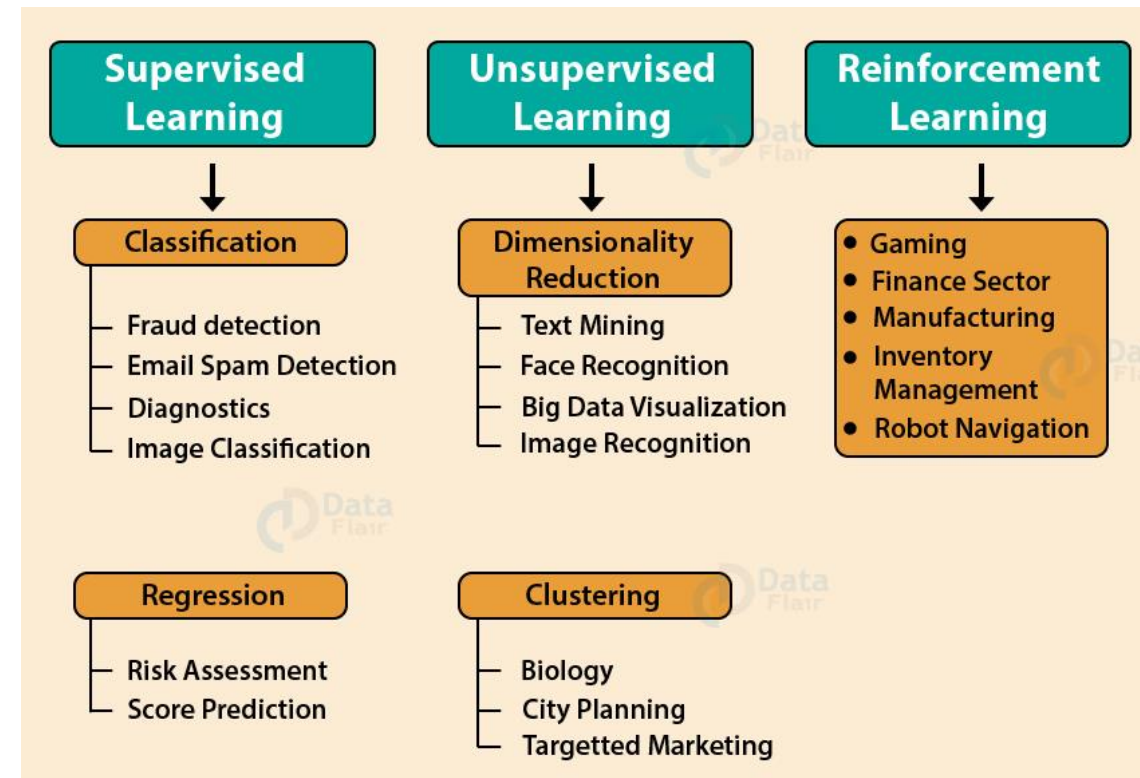
- learn a mapping from inputs x to outputs y
- training set (input-output pairs) $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$

❖ Unsupervised(or descriptive) learning

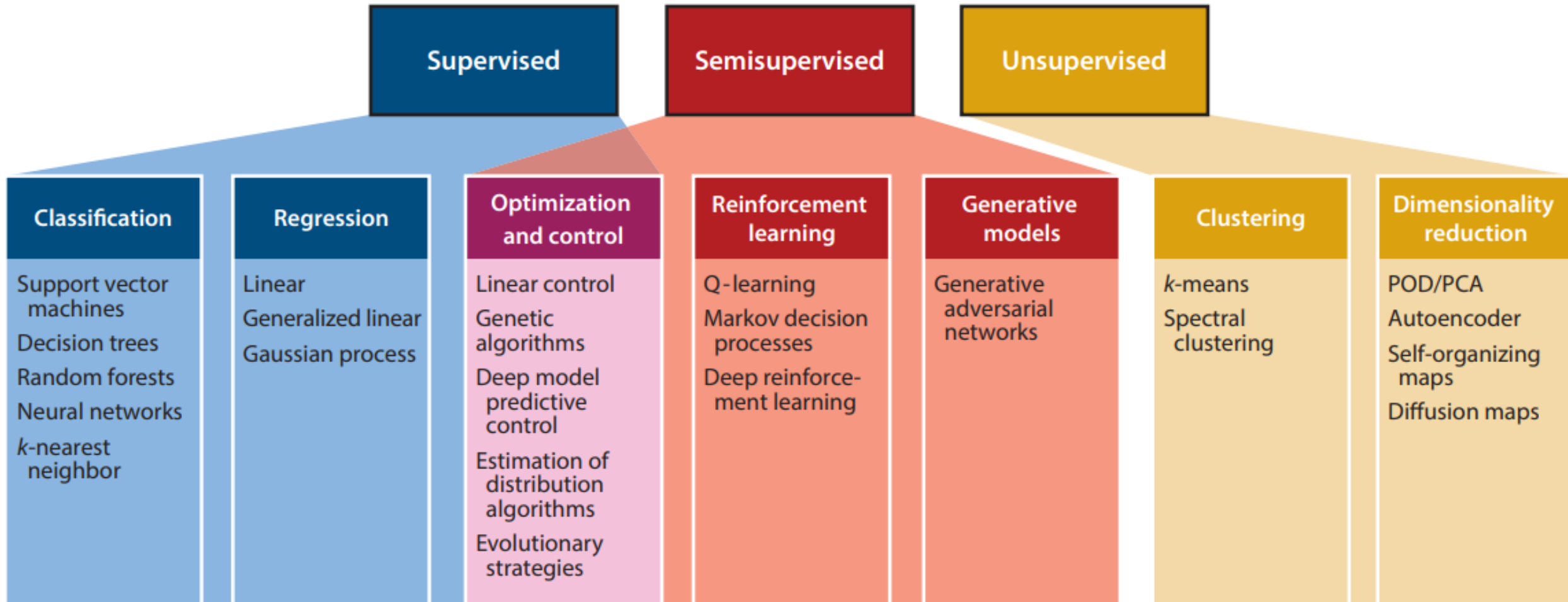
- only given inputs, $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^N$
- goal : find “interesting patterns” in the data.

❖ Reinforcement learning

- reward or punishment signals.



기계학습 분류



알고리즘사고

알고리즘의 유래

❖ 알고리즘

- 9세기경 페르시아 수학자인 알콰리즈미(al-Khwarizmi)의 이름으로부터 유래

❖ 최초의 알고리즘

- BC 300년 경 유클리드의 최대공약수 알고리즘
 1. 임의의 자연수 a , b 가 주어져 있고, $a > b$ 라 가정
 2. 자연수 a 를 b 로 나누어 나머지 n 을 구한다.
 3. 나머지 n 이 0이면 b 가 최대공약수
 4. 나머지 n 이 0이 아니면, $a \leftarrow b$, $b \leftarrow n$ 을 대입하고, step 2부터 다시 반복
 - ✓ 예) (4, 2); (8, 3); (30, 25)

❖ 문제를 해결해 가는 절차

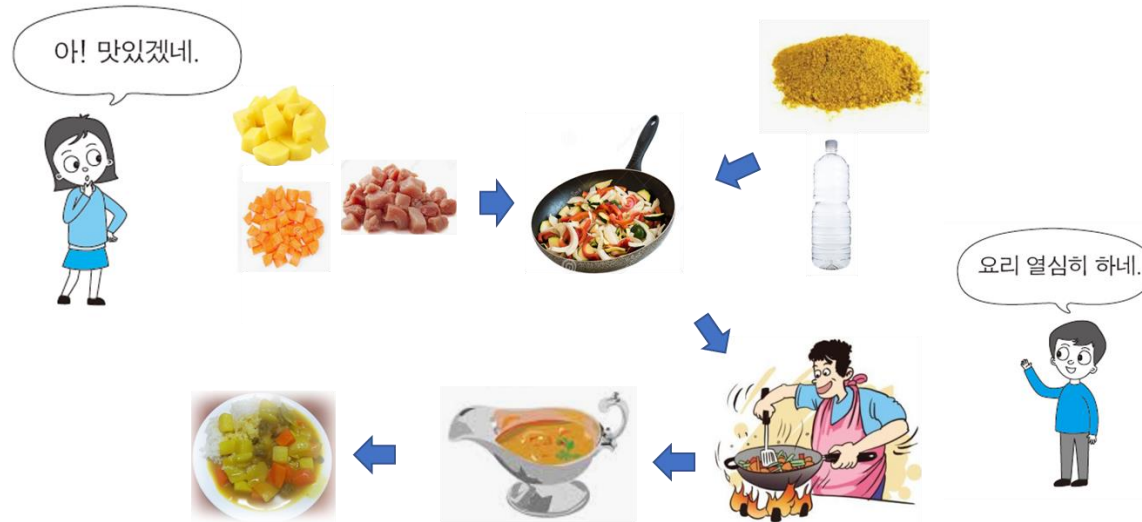
알고리즘이란?

- 알고리즘

- 문제를 해결하기 위한 단계적인 절차를 의미

- 알고리즘은 요리법과 유사

- 단계적인 절차를 따라하면 요리가 만들어지듯이, 알고리즘도 단계적인 절차를 따라 하면 주어진 문제의 답을 준다.



알고리즘이란?

- 마당에 있는 강아지를 한마리씩 세어보자.
 1. $N = 0$ // N 은 강아지의 수
 2. 마당에 아직 카운트하지 않은 강아지가 있으면 (즉, 카운트 표식이 없으면)
 3. 카운트 안 된 강아지에 카운트했다는 표식을 하고
 4. $N = N + 1$
 5. 단계 2로 가서 반복
- 강아지 11마리를 세는 경우 11번 반복

알고리즘이란?

- 마당에 있는 강아지를 두마리 (한쌍) 씩 세어보자.
 1. $N = 0$ // N 은 강아지의 수
 2. 마당에 아직 카운트하지 않은 강아지 쌍이 있으면 (즉, 카운트 표식이 없으면)
 3. 카운트 안 된 강아지 한 쌍에 카운트했다는 표식을 하고
 4. $N = N + 2$
 5. 마당에 아직 카운트 안된 강아지가 있다면, (즉, 카운트 표식이 없으면)
 6. 카운트 안 된 강아지에 카운트했다는 표식을 하고
 7. $N = N + 1$
 8. 스텝 2로 가서 반복
- 강아지 11마리를 세는 경우 6번 수행 (코드는 길어짐)
- 효율적인 알고리즘 고안이 중요
 - 주어진 문제에 대해 여러 종류의 알고리즘이 있을 수 있으나, 항상 보다 효율적인 알고리즘을 고안하는 것이 매우 중요

알고리즘이란?

- 알고리즘이란 문제를 이해하고 해결방안을 구상하는 것
 - 해결방안 구상이라는 것은 단순히 머리로만 생각하는 것이 아니라 논리적으로 명세화 시켜놓는 것
 - 즉, 알고리즘은 논리적이어야 하고 반드시 명세화하여야 함
 - 명세화 하는 방법으로는 단순히 글로 작성해도 되지만 순서도, pseudo code 등 도구를 사용할 수 도 있음
 - 중요한 것은 명세화, 즉 적어놔야 한다는 것 (문서화가 필요)

- 요리법 (Recipe)

- ① 빈 그릇을 준비한다.
- ② 이스트를 밀가루, 우유에 넣고 저어준다.
- ③ 버터, 설탕, 계란을 추가로 넣고 섞는다.
- ④ 따뜻한 곳에 놓아두어 발효시킨다
- ⑤ 170~180도의 오븐에서 굽는다

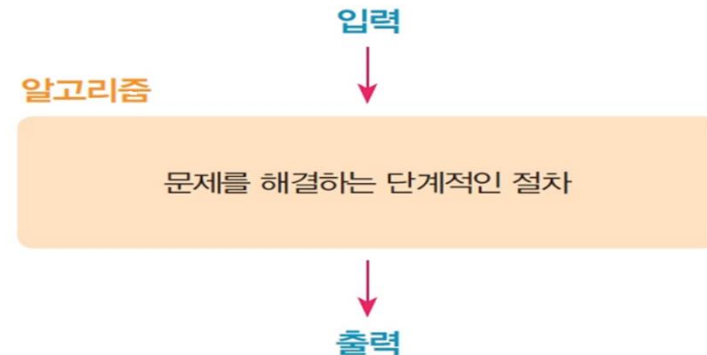


알고리즘이란?

- 빵을 만드는 요리법 (알고리즘) 주의사항
 - 빵을 만들 때 순서가 잘못되면 빵이 만들어지지 않는다. (순서 중요)
 - 빵을 만드는 방법도 여러 가지가 존재할 수 있듯이
하나의 문제에 대한 여러 가지 알고리즘이 존재할 수 있다.
 - 정확성과 효율성이 중요함 (컴퓨터 자원은 제한적이기 때문에)
 - 같은 문제를 해결할 때 가장 효율적인 알고리즘을 선택하는 것이 중요

알고리즘이란?

- 문제를 해결하도록 하는 독립된 동작들의 순서
 - 쿠키를 굽기, 넥타이 매기, 계획서 작성, 병 진단, 비행기 설계 등 알고리즘 방식의 생각을 필요로 하는 예들이 많음.
- 그대로 따라서 했을 때 목적을 달성하거나 문제를 해결하도록 독립된 동작들의 순서로 정의됨
- 컴퓨터 과학에서 알고리즘은 어떠한 입력 값을 받아서 어떤 종류의 만드는데 필요한 동작들의 순서를 정의한 것
- 알고리즘을 프로그래밍 언어로 구현하면 컴퓨터 프로그램이 됨.



알고리즘의 4대 요소

- 알고리즘의 모든 동작들은 의미(semantic)가 있어야 한다.
 - 각 단계별로 정확한 의미를 알아야 함.
- 알고리즘의 모든 동작들은 모호하게 기술되어서는 안 된다.
 - 순서 상으로 상충되지 않도록
- 알고리즘의 동작들은 수행되는 순서가 확실하게 정의되어야 한다.
- 알고리즘은 반드시 유한한 숫자의 동작들을 실행한 후 종료되어야 한다.

알고리즘사고

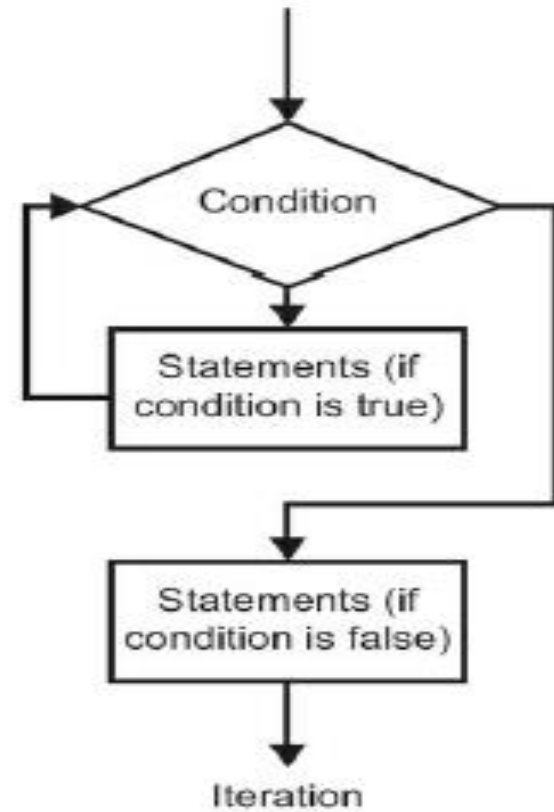
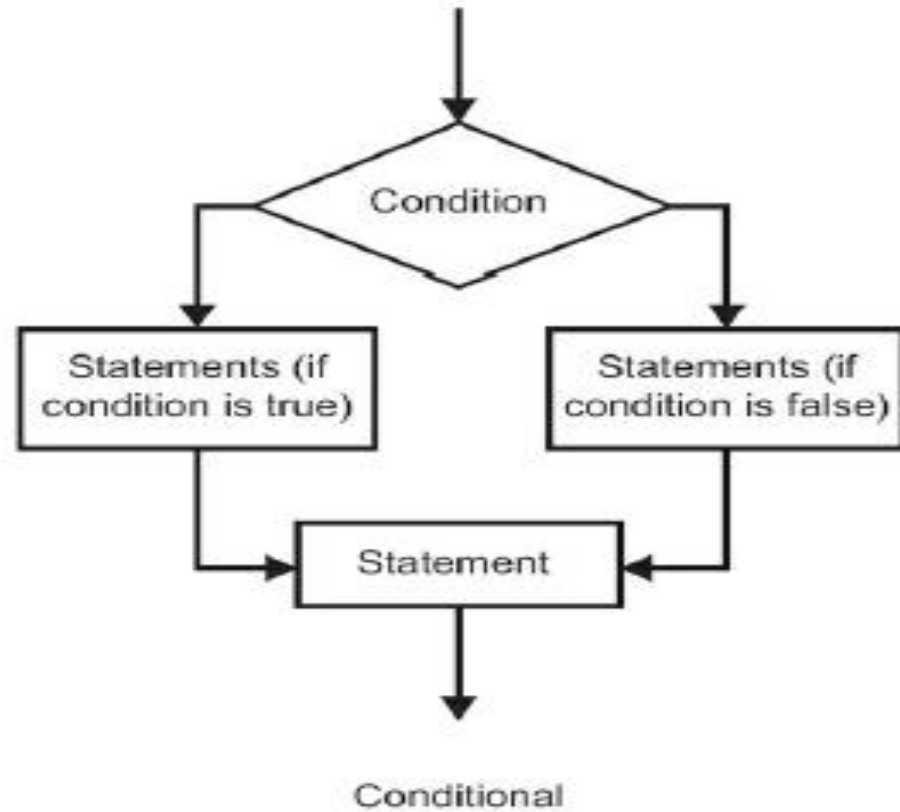
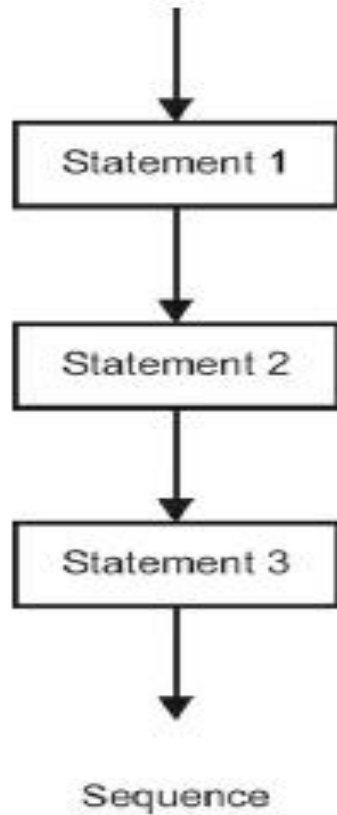
- ❖ 논리적인 결정들을 연결해나가는 과정을 알고리즘이라고 할 수 있다.
 - Algorithm: a sequence of clearly defined steps that describe a process to follow a finite set of unambiguous with clear start and end points.
 - Properties of algorithm
 - ✓ Collection of individual steps (eg. recipe)
 - ✓ Definiteness (Every step must be precisely defined.)
 - ✓ Sequential
 - ✓ Detour: State in algorithms (알고리즘은 state(상태)를 원하는 방향을 바꾸어 나가는 과정으로, 중간 과정들을 잘 추적해 갈 수 있어야 한다.)

알고리즘 사고

- ❖ 명확성, 정확성, 세밀성 (clarity, precision, and meticulousness) 필요
 - 컴퓨터는 지시한대로 일을 한다.
만일 불가능한 것을 하라면, 컴퓨터는 고장날 것이다.
 - 컴퓨터는 내재된 지식이 없다.
지시를 내리지 않으면 어떤 일도 하지 않는다.
 - 컴퓨터는 상식이 없다.
명령을 다른 방향으로 해석하려 하지 않는다.
불완전한 알고리즘의 가정을 만들거나 누락된 부분을 채울 수 없다.

기본 알고리즘 구성물

❖ 순차(Sequence), 분기(condition/branch), 반복(iteration/loop)

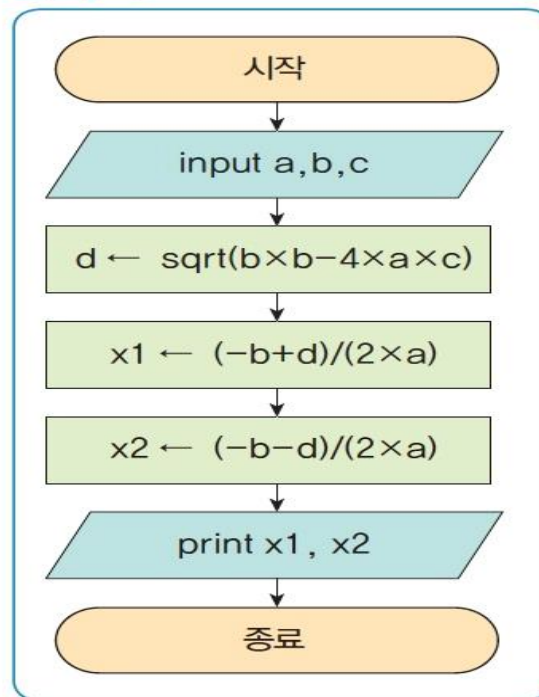


알고리즘의 표현

- 알고리즘을 작성할 때 컴퓨터로 수행이 가능한 형태로 작성
- 순서도(flowchart)나 의사코드(pseudo code)

$$ax^2 + bx + c = 0$$

순서도



의사코드

- Step 1: input a, b, c
- Step 2: $d \leftarrow \sqrt{b \times b - 4 \times a \times c}$
- Step 3: $x_1 \leftarrow \frac{-b+d}{2 \times a}$
- Step 4: $x_2 \leftarrow \frac{-b-d}{2 \times a}$
- Step 5: print x1, x2

알고리즘의 종류

- 문제해결 방식에 따른 분류
 - 확률 개념이 사용되는 랜덤(random) 알고리즘
 - 빠른 계산을 위한 병렬(parallel) 알고리즘
 - 다수의 컴퓨터들이 협력하여 문제를 해결하는 분산(distributed) 알고리즘
 - 양자 컴퓨팅을 고려한 양자(quantum) 알고리즘
 - 최적화 문제 해결에 사용되는 유전자(genetic) 알고리즘, 모의 담금질(simulated annealing) 알고리즘

알고리즘의 종류

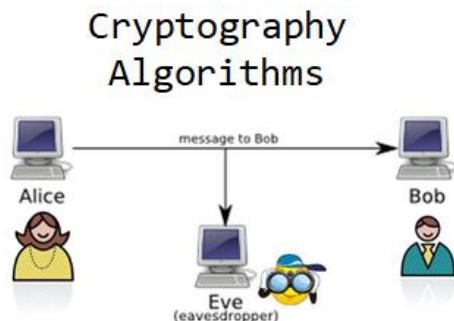
- 문제 중심 분류
 - 숫자를 순서대로 정렬하는 정렬(sorting) 알고리즘
 - 많은 데이터에서 주어진 것을 찾는 탐색(searching) 알고리즘
 - 교통, 전력망, 상하수도, 오일, 소셜 네트워크등 점과 간선들로 이루어지는 그래프 상의 문제들을 위한 그래프(graph) 알고리즘
 - 기하 문제를 해결하기 위한 기하(geometry) 알고리즘

알고리즘의 종류

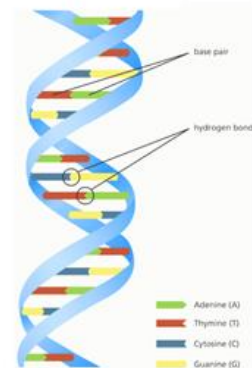
- 문제 중심 (특정분야) 분류
 - 암호학에서 다루는 암호(cryptology, cipher) 알고리즘
 - 인공지능을 위한 학습(learning) 알고리즘
 - 생물정보학의 유전자 관련 생물정보(bioinformatics) 알고리즘

알고리즘의 종류

● 문제 중심 (특정분야) 분류



Bioinformatics Algorithms



Deep Learning Algorithms

