기계학습실습

2025.09.08.

SEJONG UNIVERSITY

주간 계획 (수업계획서)

- 주간 계획에 따른 예습과 복습을 철저히!

주차	이론 (월)	실습 (수)	수업 일
1	수업 소개: 평가기준, 주간계획 등	Kaggle 소개	9/1, 9/3
2	인공지능과 머신러닝, 딥러닝	Kaggle Notebook, Google Colab, Pycharm 소개	9/8, 9/10
3	Pandas 라이브러리 소개	테스트 문제 풀이, 실습과제0: Kaggle 테스트코드 제출	9/15, 9/17
4	마켓과 머신러닝, 훈련 세트와 테스트 세트	실습과제1: ML 문제풀이 (평가)	9/22, 9/24
5	데이터 전처리1	실습과제2: ML 문제풀이 (평가)	9/29, 10/1
6	추석연휴	추석연휴	10/6, 10/8
7	회귀 (KNN회귀, 선형회귀)	실습과제3: ML 문제풀이 (평가)	10/13, 10/15
8	중간고사: 10월 20일(월)	(중간고사기간)	10/20
9	로지스틱 회귀	실습과제4: ML 문제풀이 (평가)	10/27, 10/29
10	확률적 경사하강법, 특성공학과 규제	실습과제5: ML 문제풀이 (평가)	11/3, 11/5
11	데이터 전처리2	실습과제6: ML 문제풀이 (평가)	11/10, 11/12
12	결정트리	실습과제7: ML 문제풀이 (평가)	11/17, 11/19
13	앙상블1	실습과제8: ML 문제풀이 (평가)	11/24, 11/26
14	앙상블2	실습과제9: ML 문제풀이 (평가)	12/1, 12/3
15	교차검증, 그리드서치, 시계열데이터처리	실습과제10: ML 문제풀이 (평가)	12/8, 12/10
16	기말고사: 12월 15일 (월)	(기말고사기간)	12/15

인공지능과 머신러닝, 딥러닝

- 인공지능의 분류
- 인공지능(AI), 머신러닝(ML), 딥러닝(DL) 의 관계
  - Classical ML (주요 알고리즘)

인공지능의 분류

Source: 처음만나는 인공지능, 김대수, 2020

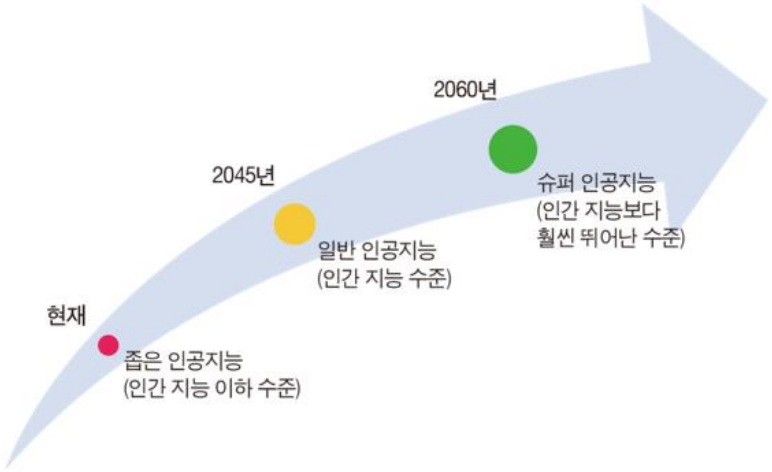
- **좁은 인공지능 (Narrow AI)**
  - 한 가지 또는 특정한 영역에 국한된 인공지능
  - 알파고, 자율주행자동차, 애플의 시리 등
- **일반 인공지능 (General AI)**
  - 인간 수준의 능력을 가진 인공지능
  - 모든 분야에 적용될 수 있는 인공지능 (일반화)
- **슈퍼 인공지능 (Super AI)**
  - 모든 면에서 인간보다 훨씬 뛰어난 지능을 가진 인공지능
  - 과학적 창의력, 일반적인 지혜, 사회적 능력을 가짐

약한 인공지능 (Weak AI)

강한 인공지능 (Strong AI)

인공지능의 분류

- Narrow / General / Super AI



	좁은 인공지능	일반 인공지능	슈퍼 인공지능
다른 이름	전용 인공지능	범용 인공지능	초인공지능
주요 특징	한 가지 또는 특정한 영역에 국한된 인공지능	인간 두뇌와 대등한 수준의 인공지능	모든 면에서 인간보다 뛰어난 인공지능
구현 시기	현재	2045년 무렵	2060년 이후
응용 분야	체스, 바둑 등	다방면에 적용 가능	현재의 SF 영화 수준
지능 수준	인간 지능의 흉내 수준	인간과 유사한 지능 수준	인간을 뛰어넘는 수준
대응 분류	약한 인공지능에 대응	강한 인공지능에 대응	강한 인공지능에 대응

Source: 처음만나는 인공지능, 김대수, 2020

인공지능(AI) / 머신러닝(ML) / 딥러닝(DL) 의 관계

- AI / ML / DL 개념 분류의 목적
  - AI 요소기술들을 더 잘 이해하기 위함
  - 분류 그 자체가 목적이 아님
  - Examples)

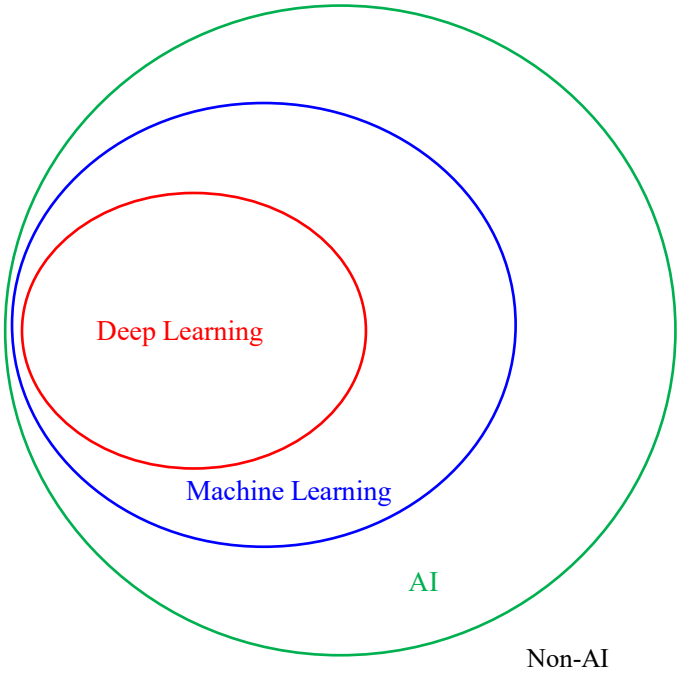
VS.

- 치타를 이해하기 위해서는,
  - 방법1) 치타 자체를 분석
    - 털이 있고, 귀가 쫑긋하며, 육식을 하고 …
  - 방법2) 비슷한/유사한 다른 동물과의 비교를 통한 이해
    - 사자와의 차이점은 점박무늬,

Source: https://ko.wikipedia.org/wiki/치타, https://ko.wikipedia.org/wiki/사자

인공지능(AI), 머신러닝(ML), 딥러닝(DL) 의 관계

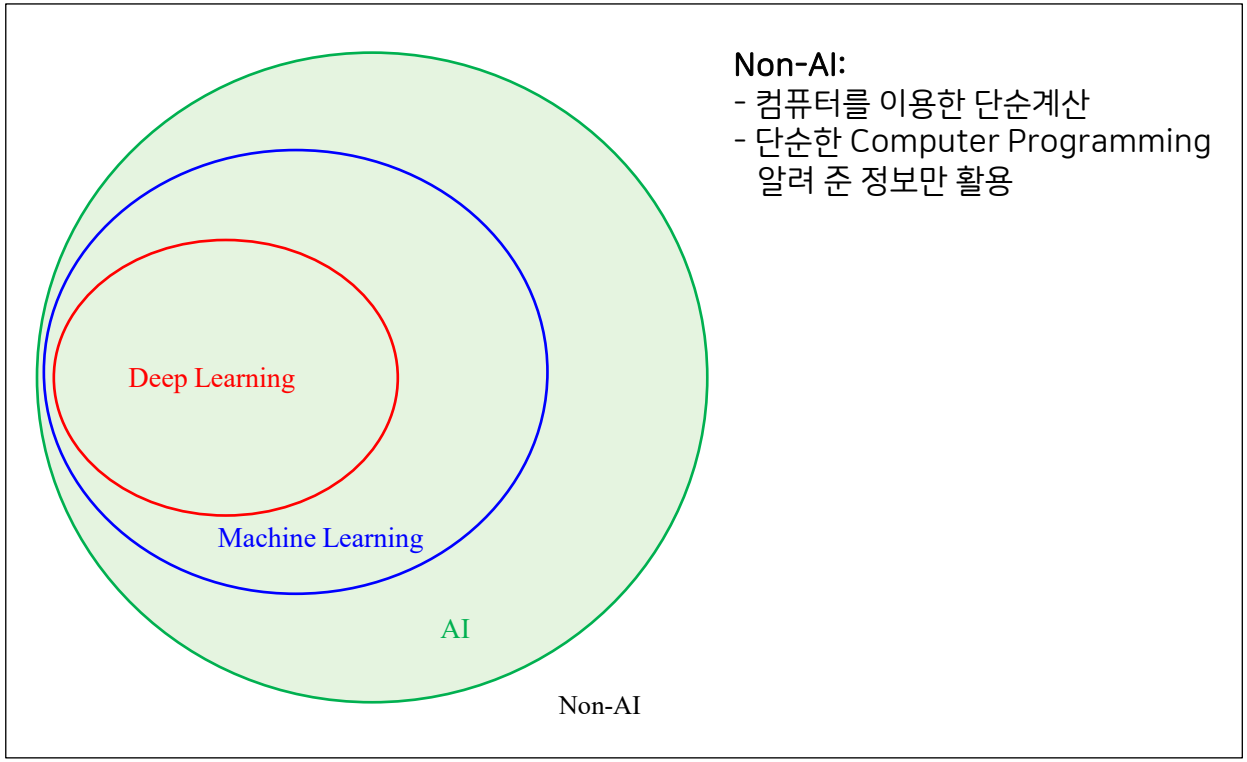
- 인공지능(AI) ⊃ 머신러닝(ML) ⊃ 딥러닝(DL)
  - AI but not ML
  - ML but not DL (= Classical ML)
  - DL



Source: https://www.deeplearningbook.org/

AI

- Artificial Intelligence?
  - 스스로 생각하고 판단할 수 있는 컴퓨터
  - 알려준 것 이상을 처리할 수 있어야 한다



Source: https://www.deeplearningbook.org/

AI

- Ex) 똑똑한 에어컨
  - 에어컨 1

기능
- 사람이 방에 있는지 감지 후 (센서) 동작(사람이 있는 경우) 및 종료(사람이 없는 경우) 를 실행 - 가동 중 24도가 되면 스스로 꺼짐, 동작 중 25도가 되면 스스로 켜짐 - 집주인이 차로 정문을 통과하고, 25도 이상이면 스스로 미리 켜짐

➔ 알려준 것 (사람이 있는경우 동작, 24도에서 꺼지고 25도에서 켜짐, 정문에 도착 시 미리 켜짐) 에 대해서만 동작 함. 알려주지 않은 것에 대한 동작을 하는 기능이 없음

➔ 겉으로 보기엔 매우 스마트하고 똑똑한 에어컨으로 보임. 그러나 인공지능이 아님 (Non-AI)

  - 에어컨 2

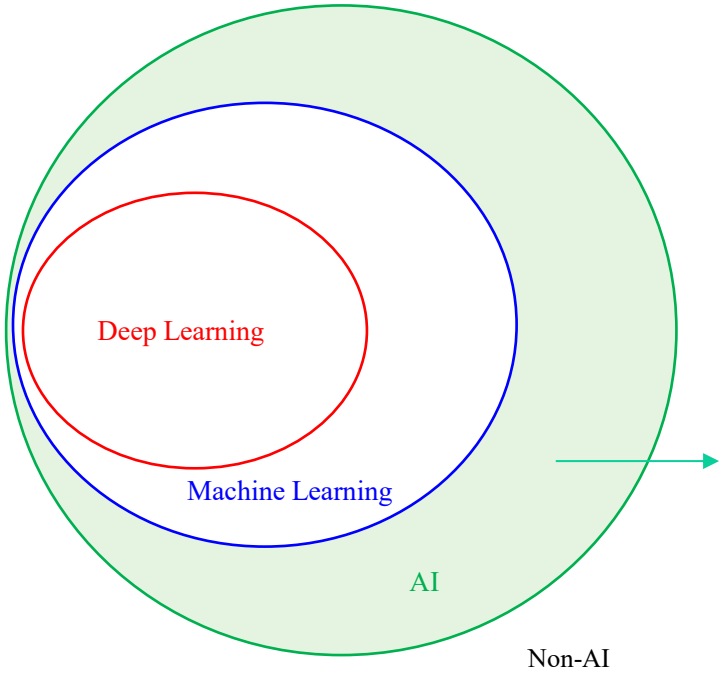
기능
- (초기 세팅) 가동 중 24도가 되면 스스로 꺼짐, 동작 중 25도가 되면 스스로 켜짐 - 수동조작을 한 경우(데이터)를 바탕으로, 사용자에 따른 최적온도를 설정 함

➔ 임의의 사용자에 따른 최적온도 값을 미리 알려주지 않음

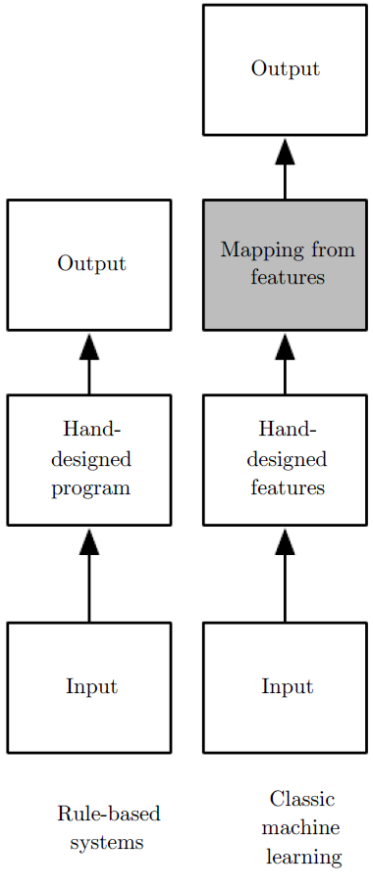
➔ 스스로 판단하여 최적온도를 설정 함 (알려준 것 이상을 처리): AI

AI but not ML

- AI but not ML
  - Ex) 규칙기반 시스템 (Rule-based System)



AI but not ML
- 규칙기반 시스템
  (Rule-Based Systems)
- 전문가 시스템 (Expert System)



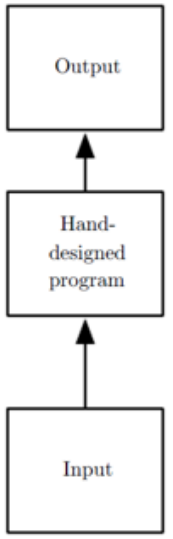
Source: https://www.deeplearningbook.org/

AI but not ML

- **규칙기반 시스템**
  - 정교하게 설계 된 Rule (hand-designed program) 에 따라 Output을 출력 함
  - Hand-designed program
    - 입력에 따른 원하는 출력(답)을 낼 수 있도록 사람이 직접 정교한 설계를 함

[주의] 모든 경우를 다 미리 설계하여 자칫 non-AI 로 오해할 수 있으나, 미리 알려준 것이상을 스스로 생각하고 판단해서 처리할 수 있음

  - Ex) Expert System



Rule-based systems

- **EX) 동물 판별 시스템**
  - 입력: 치타는 포유류 인가?
  - 출력: YES
    - Hand-designed program 이라면 치타가 포유류라는 정보가 기존에 없더라도, 다양한 추론방법을 사용하여 기존에 알고 있는 정보들로부터 결론을 추론 한다

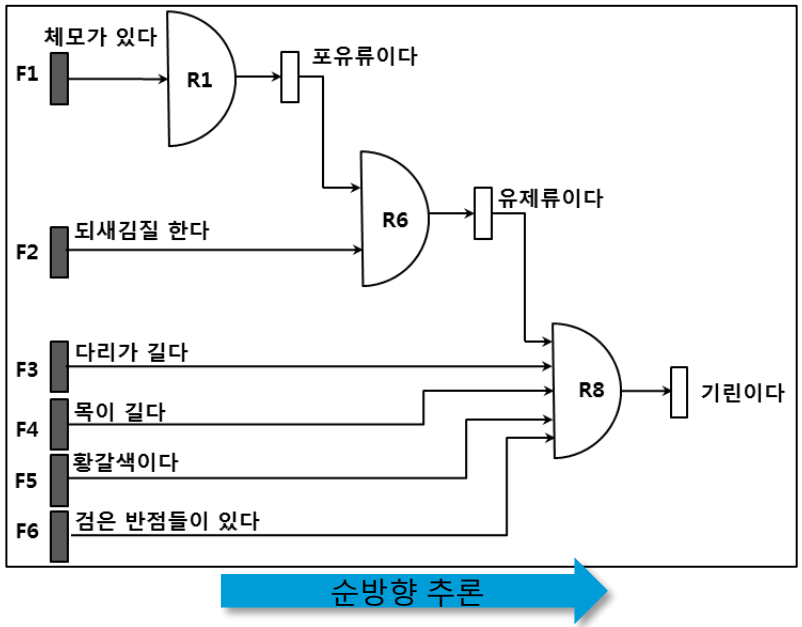
AI but not ML

- EX) 순방향 추론 (동물판별 시스템)
  - 동물 분류 규칙 (사실관계를 통해 확보됨)

R1: IF x는 체모가 있다 THEN x는 포유류이다.
R2: IF x는 수유를 한다 THEN x는 포유류이다.
R3: IF x는 깃털이 있다 THEN x는 조류이다.
R4: IF x는 난다 AND x는 알을 낳는다 THEN x는 조류이다.
R5: IF x는 포유류이다 AND x는 고기를 먹는다 THEN x는 육식동물이다.
R6: IF x는 포유류이다 AND x는 되새김질한다 THEN x는 유제류이다.
R7: IF x는 육식동물이다 AND x는 황갈색이다 AND x는 검은 반점들이 있다 THEN x는 치타이다.
R8: IF x는 유제류이다 AND x는 다리가 길다 AND x는 목이 길다 AND x는 검은 반점들이 있다 AND x는 황갈색이다 THEN x는 기린이다.
R9: IF x는 포유류이다 AND x는 눈이 앞을 향해있다 AND x는 발톱이 있다 AND x는 이빨이 뾰족하다
THEN x는 육식동물이다.

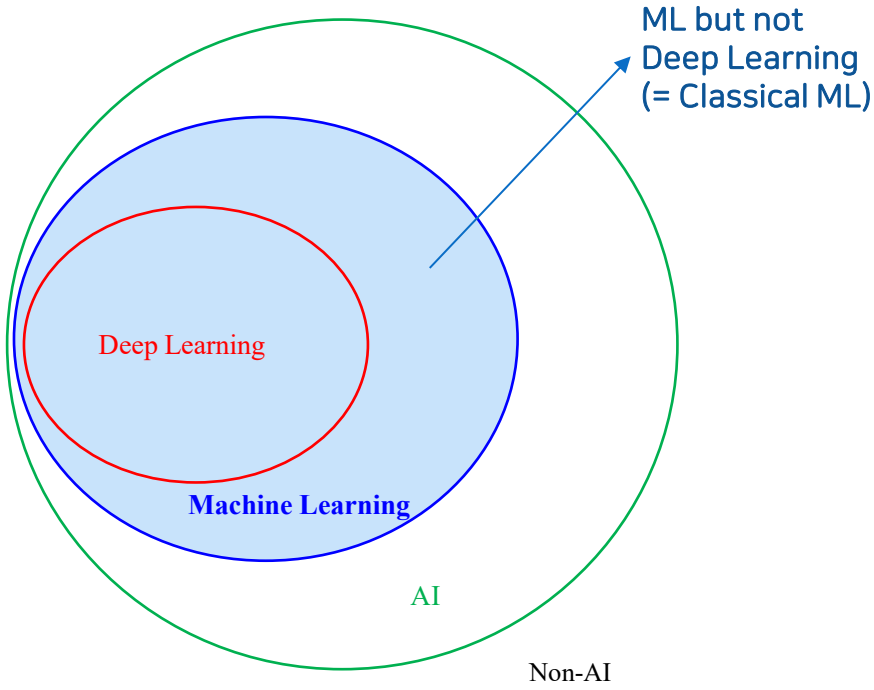
- 질문) A는 무엇인가?

F1: A는 체모가 있다.
F2: A 는 되새김질을 한다.
F3: A 는 다리가 길다.
F4: A 는 목이 길다.
F5: A 는 황갈색이다.
F6: A 는 검은 반점들이 있다



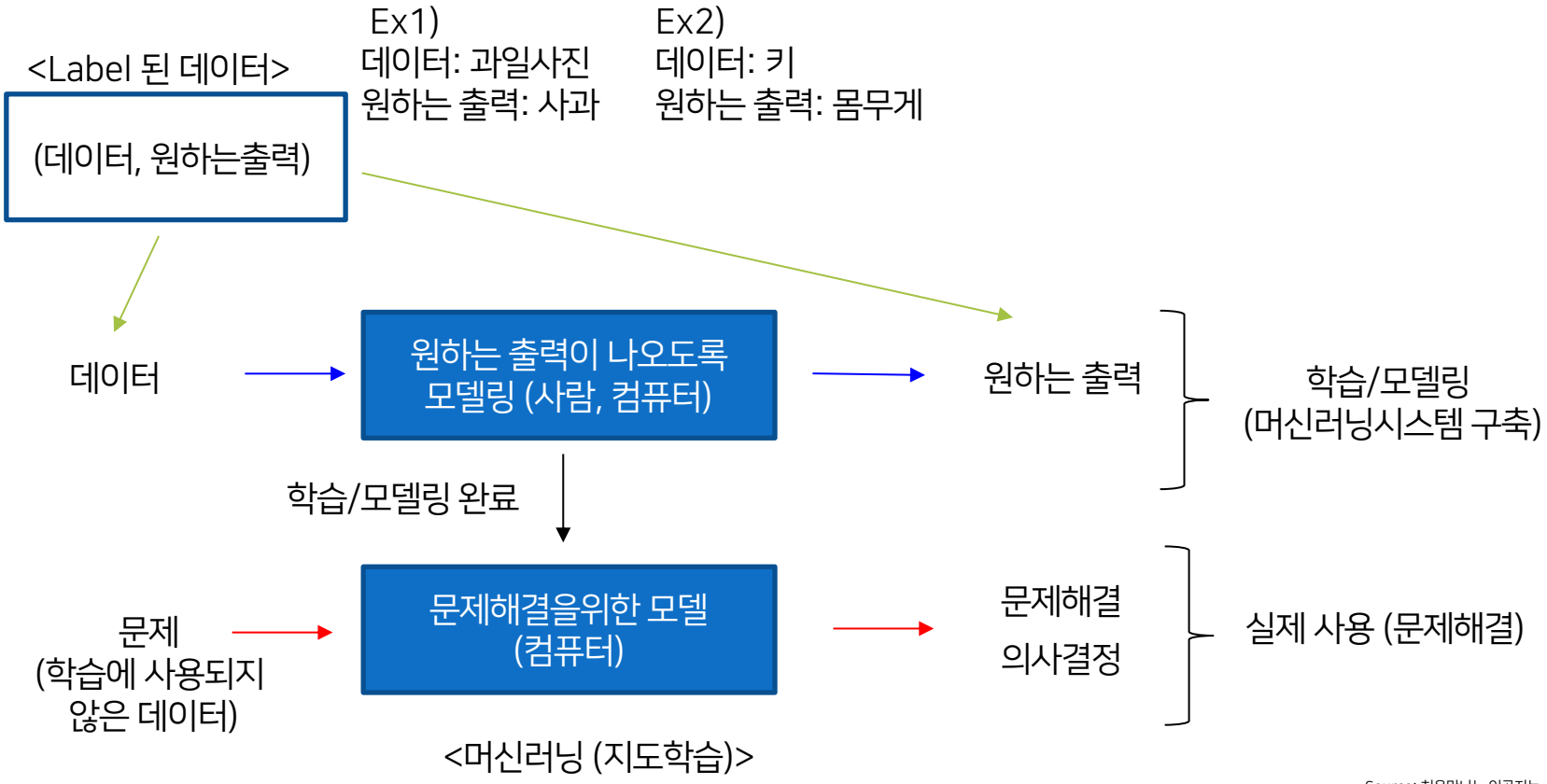
ML

- Machine Learning?
  - 컴퓨터 프로그램이 데이터로부터의 학습하는 과정
  - 경험으로부터 지식을 학습
  - 학습방법에 따른 구분
    - 지도학습 (Supervised Learning)
    - 비지도학습 (Unsupervised Learning)
    - 강화학습 (Reinforcement Learning)
  - 문제에 따른 구분
    - 분류 (Classification)
    - 회귀 (Regression)



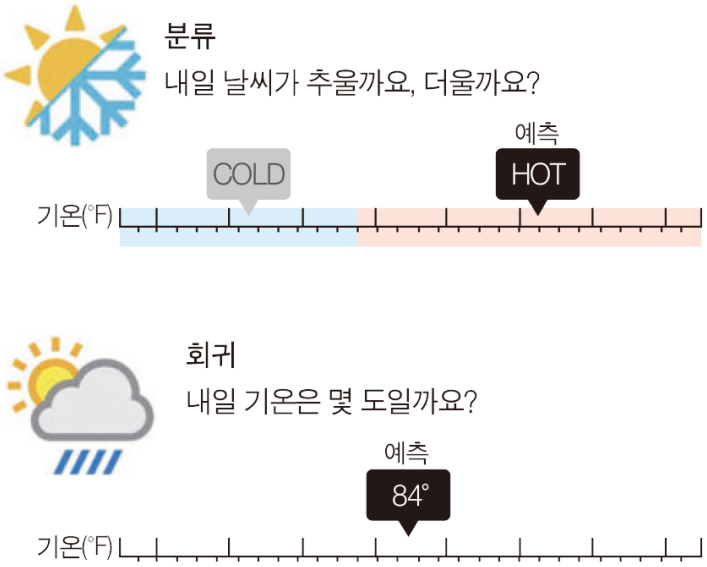
ML

- **지도학습 (Supervised Learning) 이란?**
  - 입력과 미리 알려진 출력을 연관시키는 관계를 학습
  - 주어진 입력과 출력 쌍 사이의 대응 관계를 학습
  - 각 데이터에 레이블(label) 또는 태그(tag) 표시 붙임

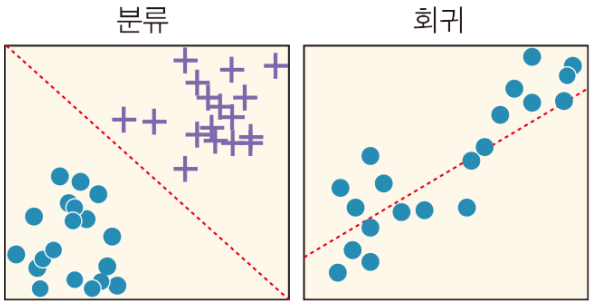


Source: 처음만나는 인공지능, 김대수, 생능출판사

- **대표적인 ML 문제: 분류 vs. 회귀**
  - 분류는 일정한 기준에 따라 명백하게 구분 짓는 것
  - 분류의 출력은 남자/여자 등과 같은 선택식 출력
    - Ex) "내일 날씨는 더울 것이다."와 같은 이분법적 선택
  - 회귀는 오차 제곱의 합을 최소화하는 직선을 긋는 작업 따라서 명확히 직선으로 구별되는 것이 아님
  - 회귀의 출력은 연속값으로 나타냄
    - Ex) "내일 기온?"에 대해 "18.3도로 추정된다." 등의 형태



[그림 8.19] 분류와 회귀의 차이점



[그림 8.18] 분류와 회귀의 비교

Source: 처음만나는 인공지능, 김대수, 생능출판사

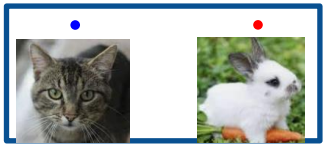
ML

- 분류의 예
  - 고양이, 토끼 분류

<Label 된 데이터>


• 고양이 •토끼

사람: Hand-designed features, 특징결정 (꼬리의길이, 귀의크기)
컴퓨터: Mapping from features, 데이터를 기반으로 결정경계 결정



원하는 출력이 나오도록 모델링 (사람, 컴퓨터)

고양이 or 토끼

학습/모델링 (머신러닝 시스템 구축)

학습/모델링 완료

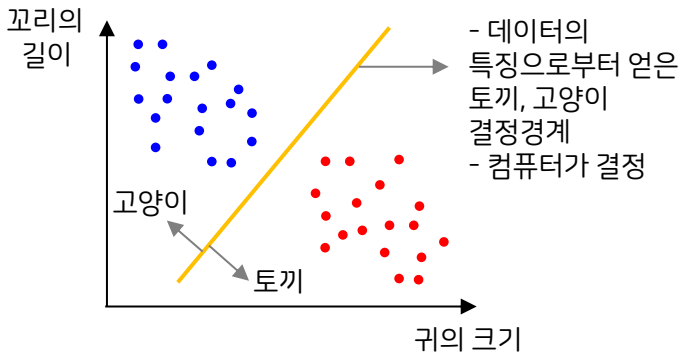
문제 (사진 •) (학습에 사용되지 않은 데이터)

문제해결을위한 모델 (컴퓨터)

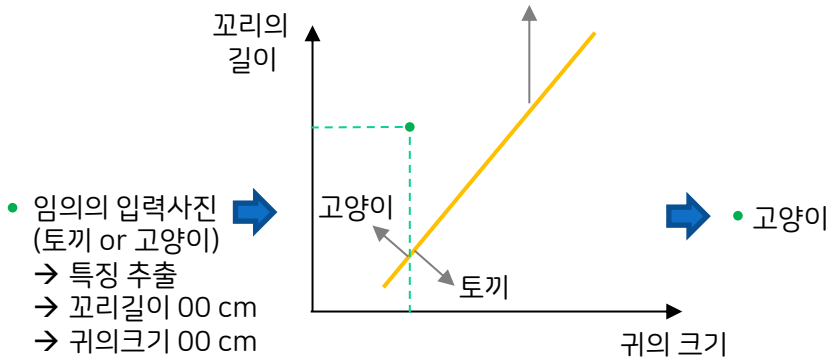
문제해결 (고양이)

학습완료된 모델

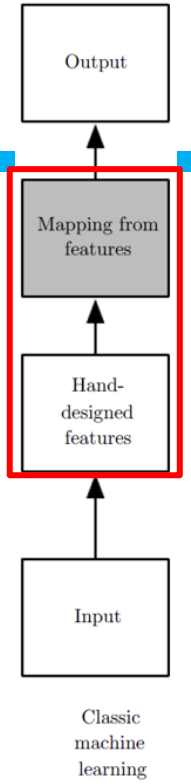
실제 사용 (문제해결)



<학습 (모델링)>



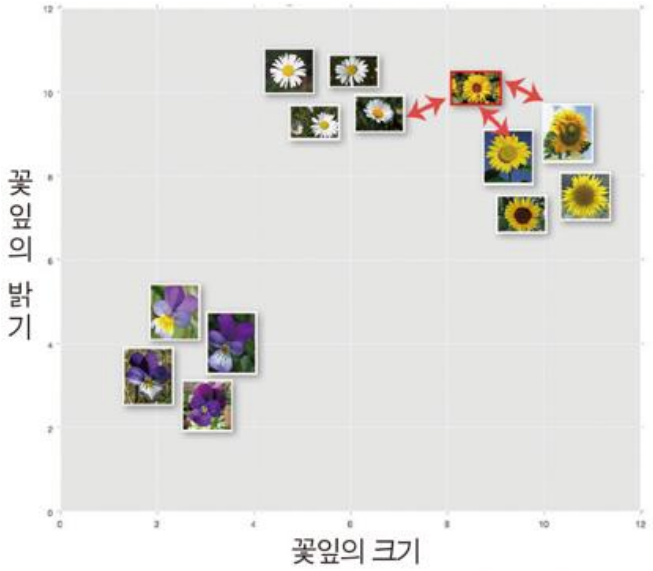
<실제 사용>

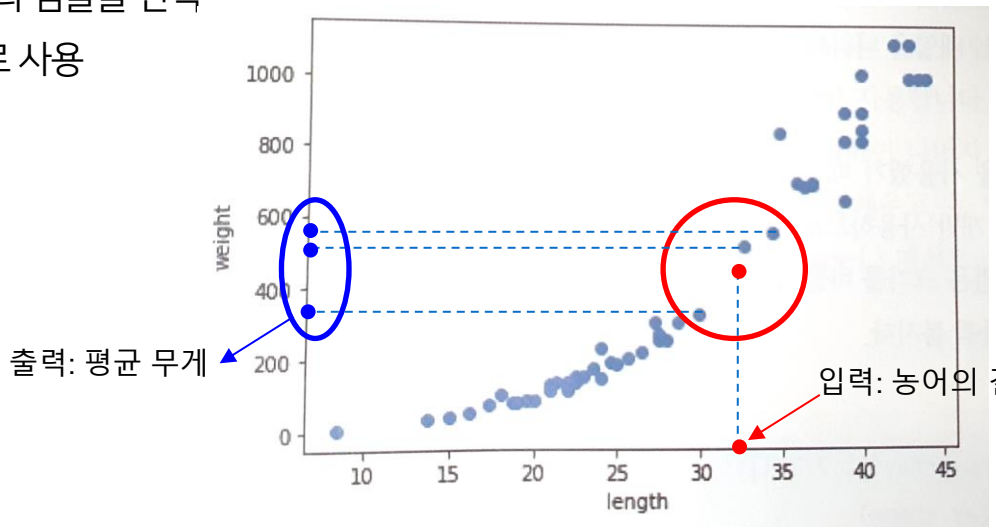


Classical ML

- K-Nearest Neighbor (KNN)
  - KNN Classifier
    - 가장 가까운 것 들과의 거리 계산으로 클래스를 분류
    - 새로운 입력 데이터와 가장 가까운 k개의 이웃 데이터 선택
    - 이웃 데이터들의 클래스 중 다수결로 데이터의 클래스 결정
    - 다수결에서 결과가 나오기 위해 k는 반드시 홀수여야 함
  - KNN Regressor
    - Ex) 농어의 길이 데이터로 무게를 예측
    - 테스트 샘플과 가장 가까운 K개의 샘플을 선택

      → 무게를 평균 내어 예측 값으로 사용



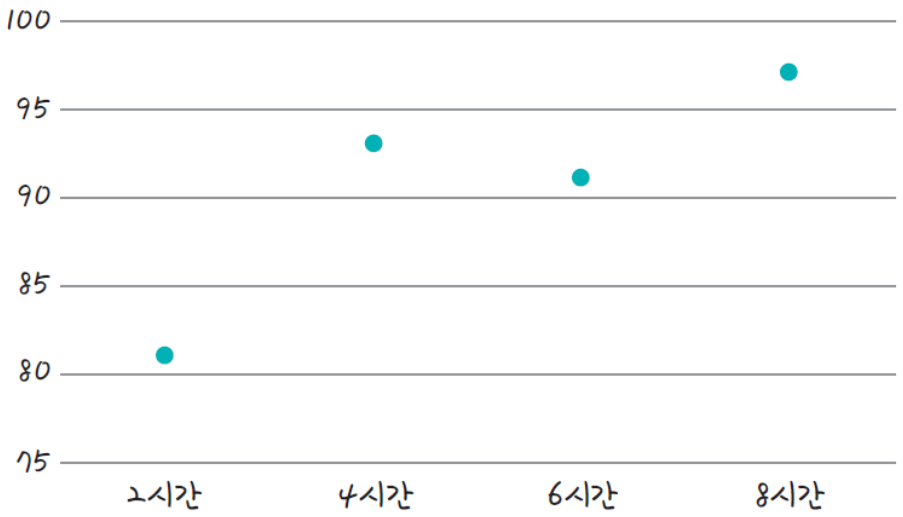


Classical ML

- **선형 모델**
  - Linear Regression (회귀)
    - Ex) 공부한 시간에 대한 성적 예측

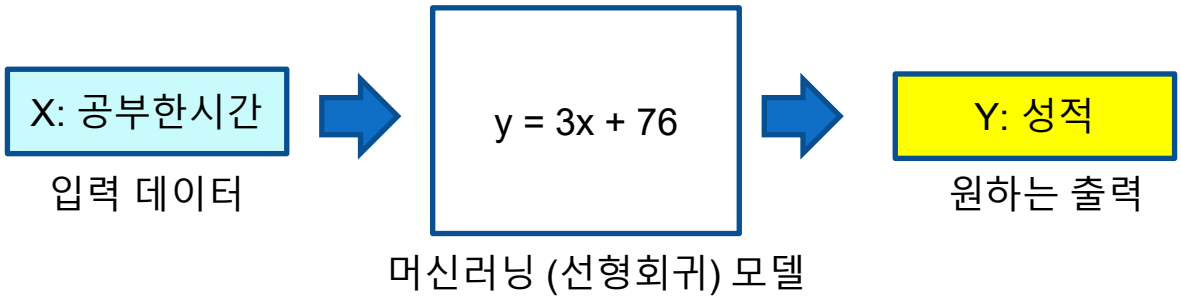
공부한 시간	2시간	4시간	6시간	8시간
성적	81점	93점	91점	97점

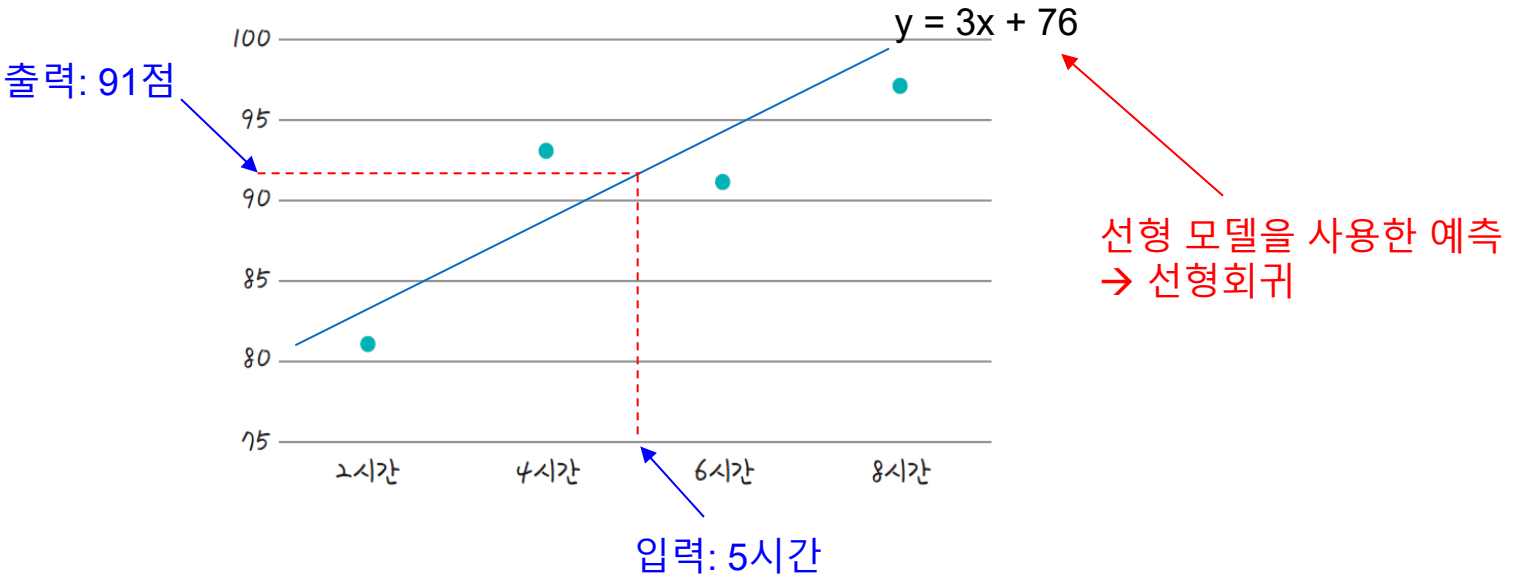
    - 여기서 공부한 시간을 x라 하고 성적을 y라 할 때 집합 X와 집합 Y를 다음과 같이 표현할 수 있음
      - x = {2, 4, 6, 8}
      - y = {81, 93, 91, 97}



Classical ML

- a = 3, b = 76 이라고 가정 (구하는 방법은 추후 강의에서 자세히 다룰 예정)
- 5시간 공부하면 성적이 어떨까? → 주어진 선형 모델에 입력 → 3 x 5 + 76 = 91 : 91점 으로 예측

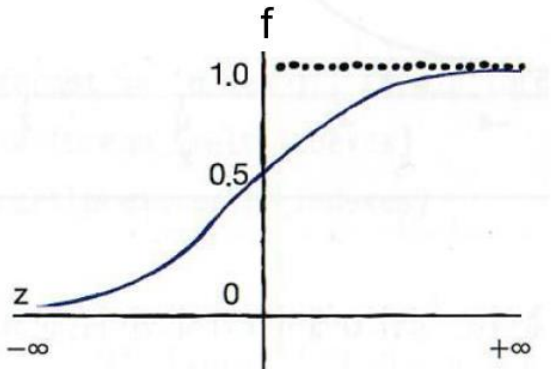


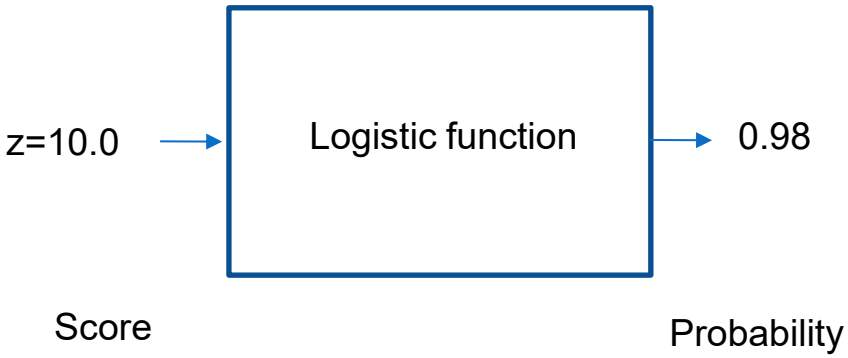


Classical ML

- 선형 모델
  - Logistic Regression (분류)
    - 로지스틱 함수를 활용한 분류

$$f = \frac{1}{1+e^{-z}}$$





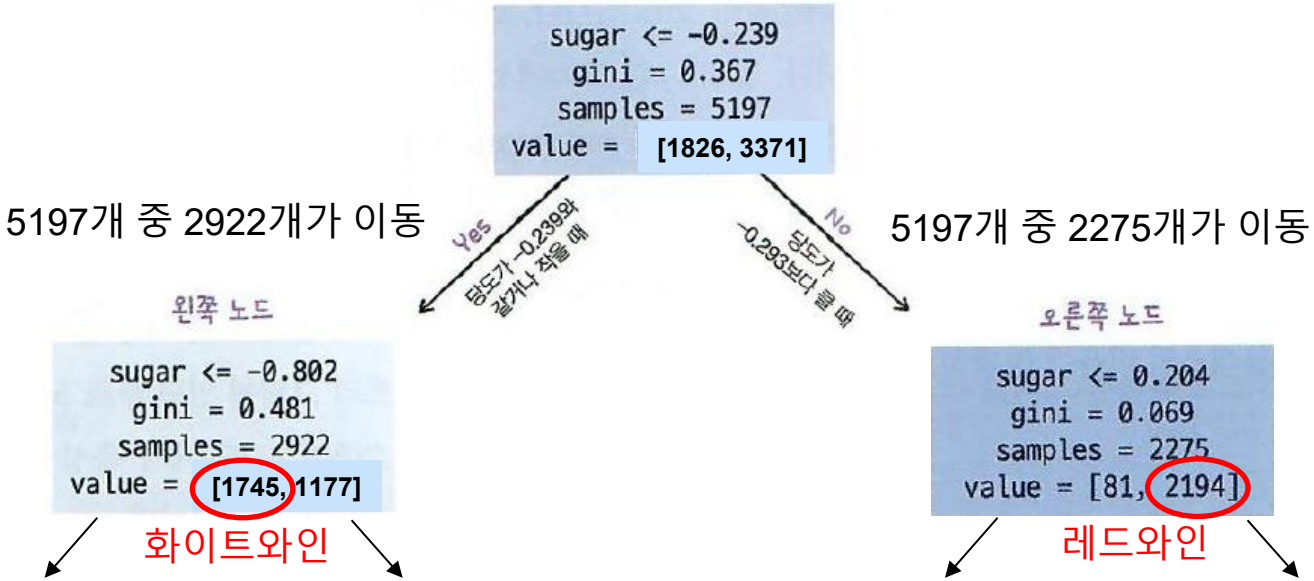
Ex) $z = ax_1 + bx_2 + cx_3 + d$

Ex) Sigmoid function: S자형 곡선을 갖는 수학함수

(Logistic function, Hyperbolic tangent, Arctangent function, Gudermannian function, …)

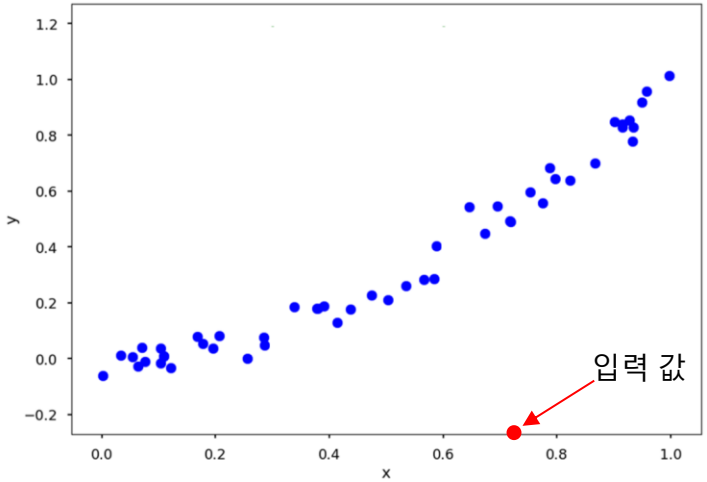
Classical ML

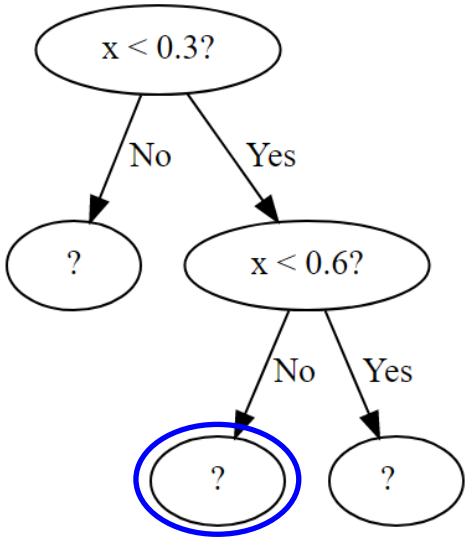
- Decision Tree
  - 특정 기준에따른 노드 분할을 통한 분류, 회귀
  - Classification Ex)
    - 어떻게 노드를 분할하는지가 중요 (decision tree의 학습)
    - Root node
      - 당도가 -0.239 이하 → 왼쪽 가지로 이동
      - 총 샘플의 수: 5197개
      - 음성(화이트와인) 샘플의 수: 1826, 양성(레드와인) 샘플의 수: 3371

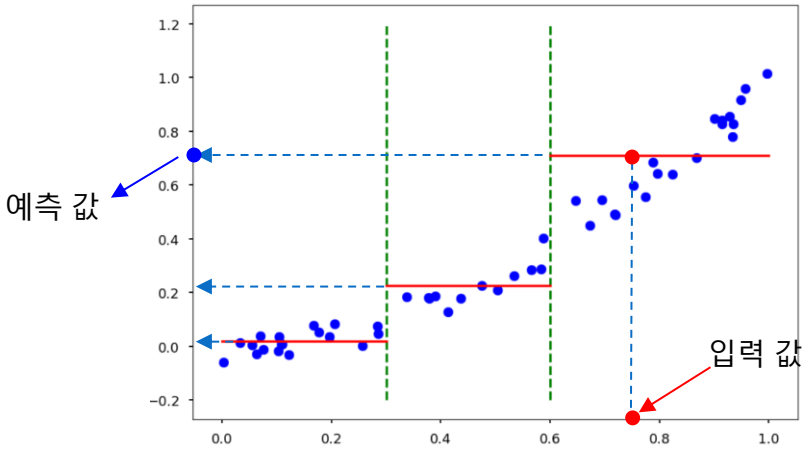


Classical ML

- Regression Ex)
  - 결정 (예측) 방법: Regression (원리)
    - Ex) x 로 부터 y 값을 예측하는 회귀 문제

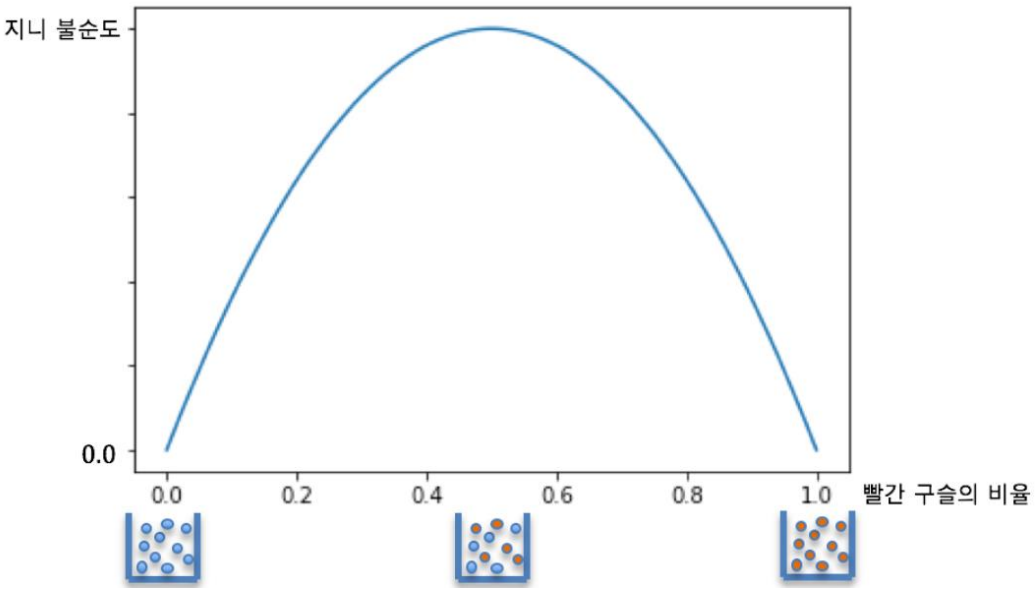


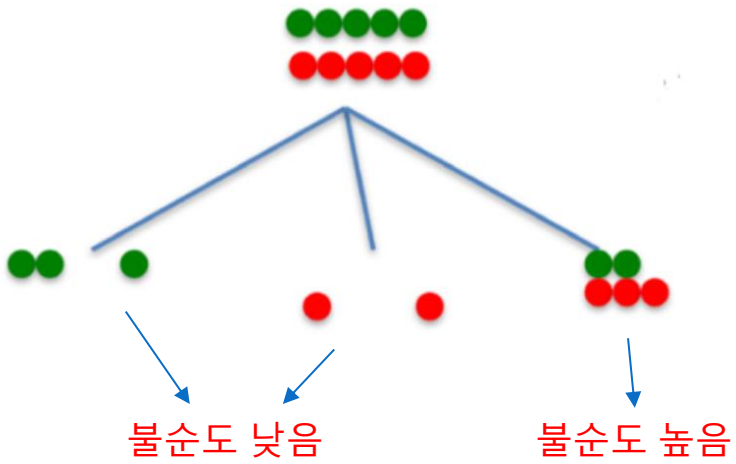




Classical ML

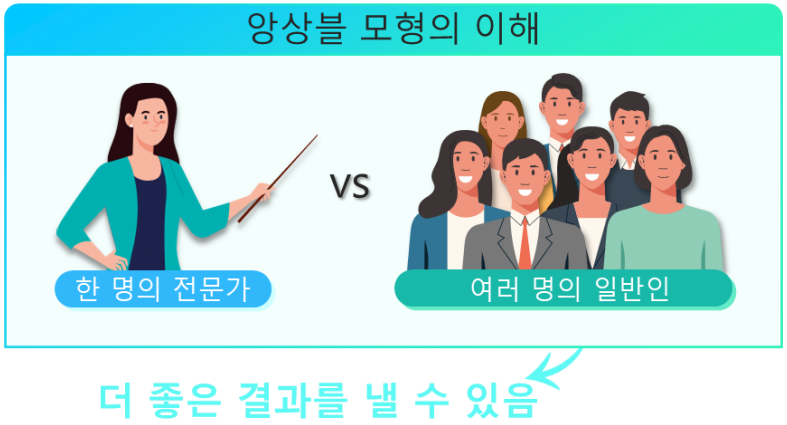
- 노드 분할 규칙
  - 분할 후 각 영역의 순도 (homogeneity) 증가
  - = 불순도 (impurity) 혹은 불확실성 (uncertainty)
  - = 정보 이득 (information gain) 증가
- 지니불순도





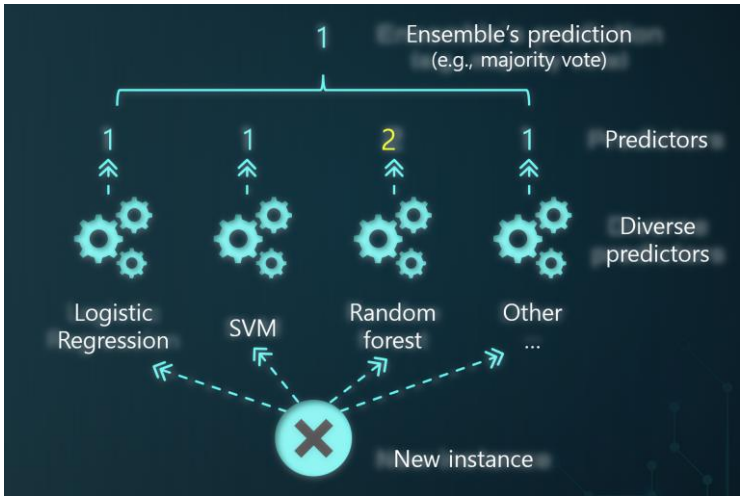
Classical ML

- Ensemble
  - 주어진 학습 데이터 집합에 대해서 여러 개의 서로 다른 모델을 만들고, 이들 모델의 판정 결과 참고하여 최종 결과를 내는 방법
    - 분류
      - 판정결과를 투표
    - 회귀
      - 출력 값을 평균
  - 주요 Ensemble 알고리즘: Voting, Bagging, Boosting, Stacking
  - 장점
    - 단일 모델을 사용하는 것에 비해 결과의 variance를 줄이는 효과가 있음
    - 즉, 보다 일정한 결과를 내도록 해 줌

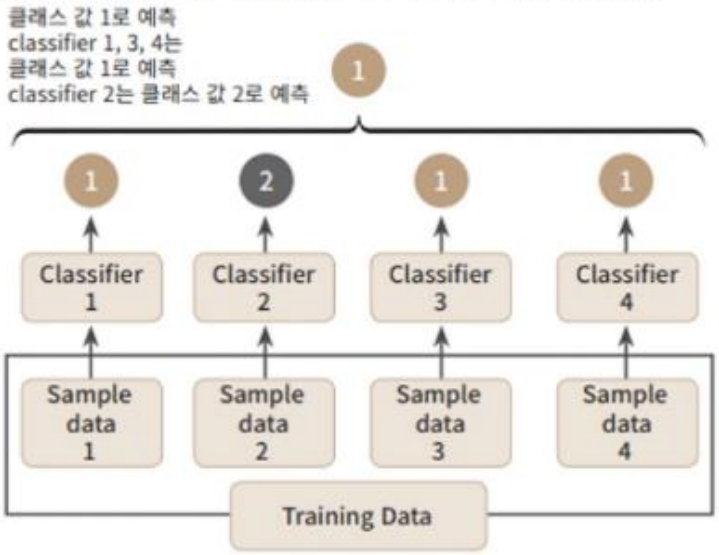


Classical ML

- Voting

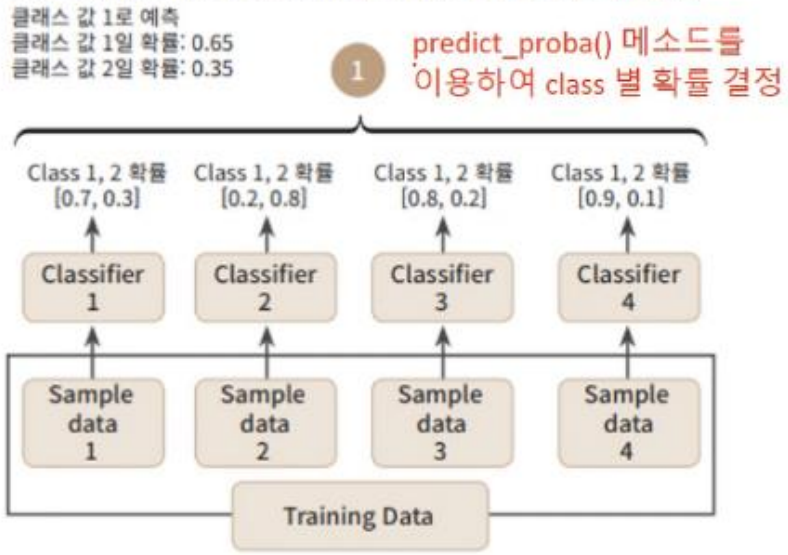


Hard Voting은 다수의 classifier 간 다수결로 최종 class 결정



<하드 보팅>

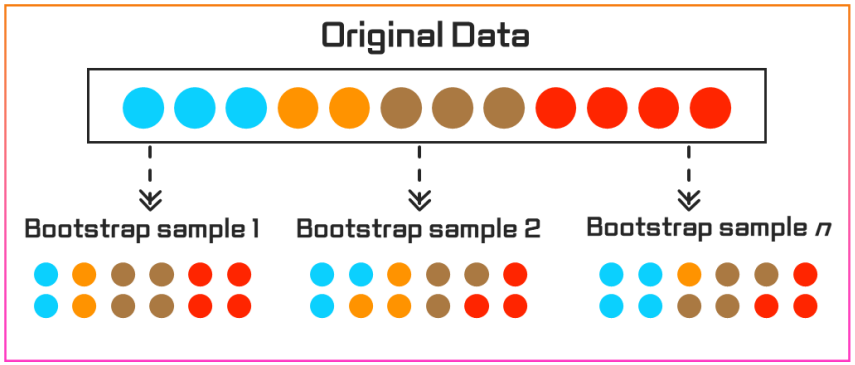
Soft Voting은 다수의 classifier 들의 class 확률을 평균하여 결정

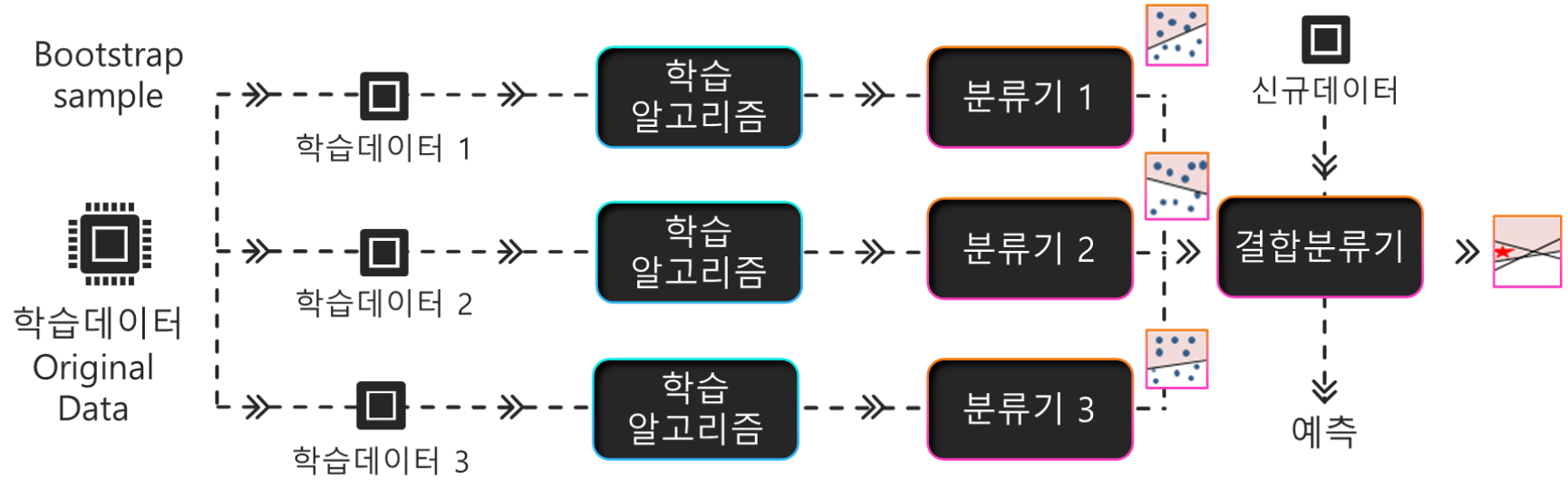


<소프트 보팅>

Classical ML

- Bagging (Bootstrap Aggregating)
  - Bootstrap을 통해 여러 개의 학습 데이터 집합을 만들고, 각 학습 데이터 집합별로 분류기를 만들어, 이들이 투표나 가중치 투표를 하여 최종 판정을 하는 기법





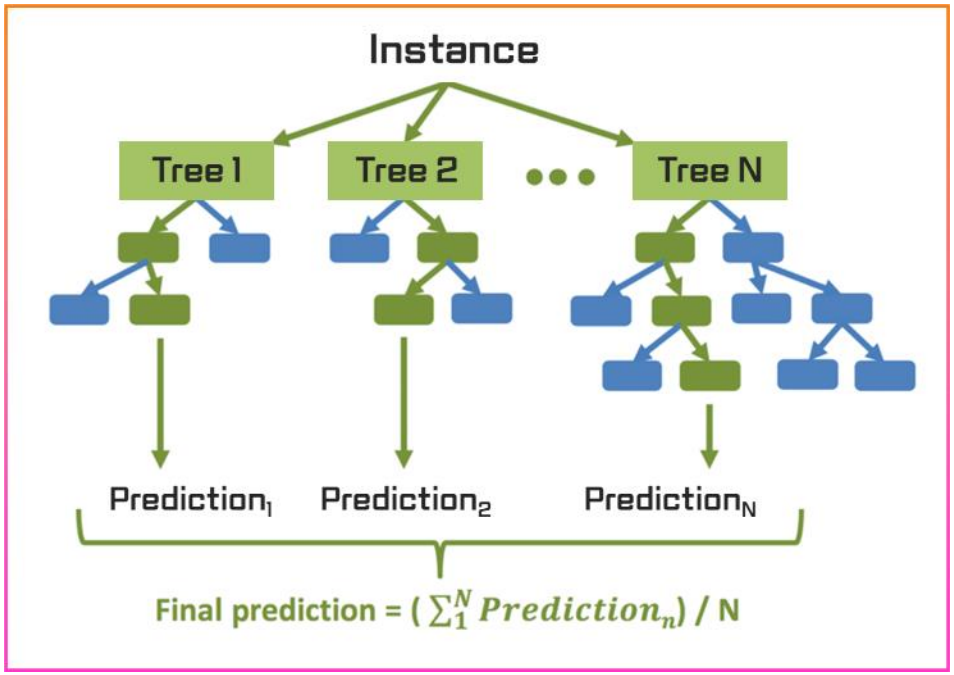
Classical ML

- Ex) Random Forest
  - 분류기로 decision tree 를 사용하는 bagging 기법

`sklearn.ensemble`.RandomForestClassifier

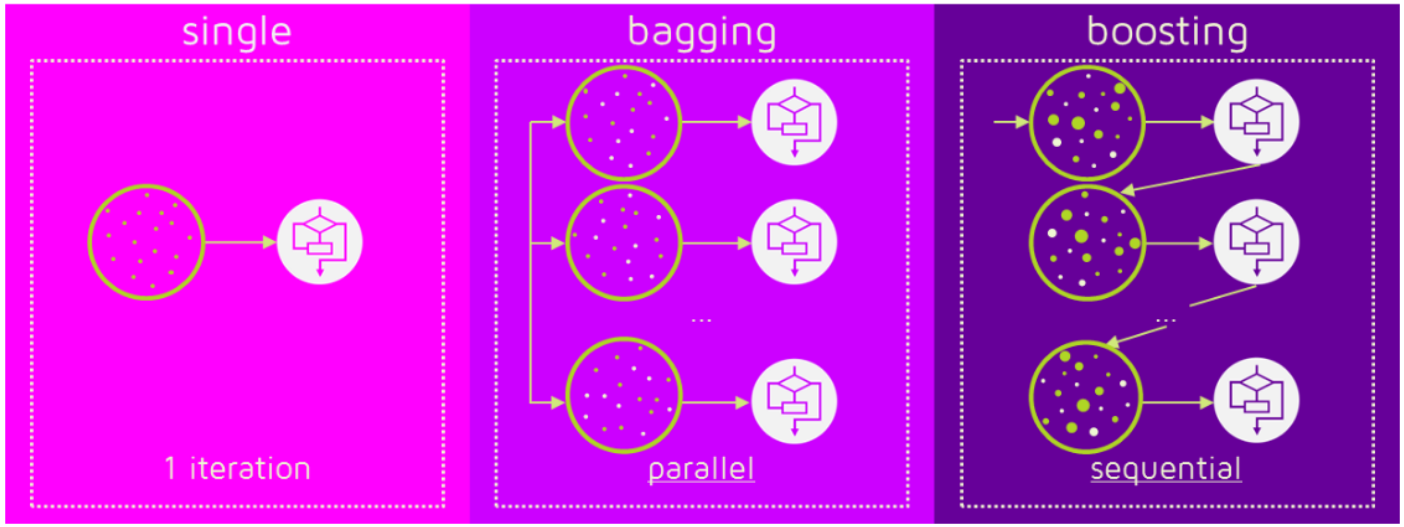
`sklearn.ensemble`.RandomForestRegressor

Parameters
n_estimators: int, default=100
: The number of trees in the forest
max_depth: int, default=None



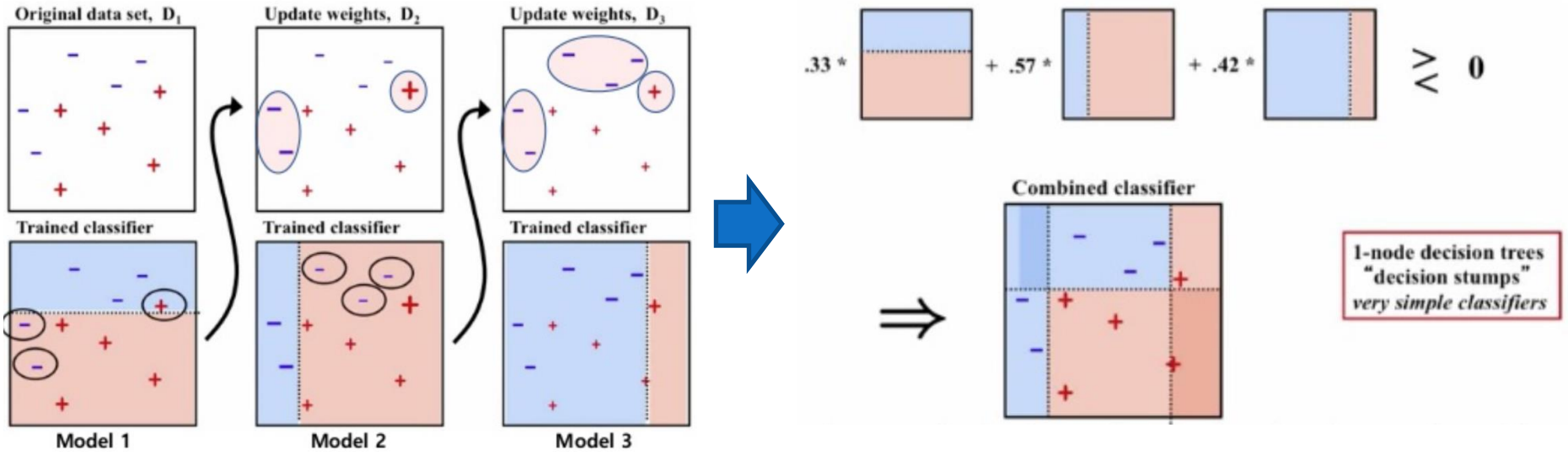
Classical ML

- Boosting
  - 순차적 학습
    - Ex) 약한 분류기를 결합하여 강한 분류기를 만듦
    - Ex) A 분류기를 만든 후, 그 정보를 바탕으로 B 분류기를 만들고, 다시 그 정보를 바탕으로 C 분류기를 만듦
      - → A, B, C 분류기를 결합하여 최종 모델 구축



Classical ML

- Ex) Boosting 과정



Classical ML

- Ex) AdaBoost
  - 약한 분류기의 가중치와 약한 분류기 (stump) 값의 곱을 다 더해서 최종 강한 분류기를 생성

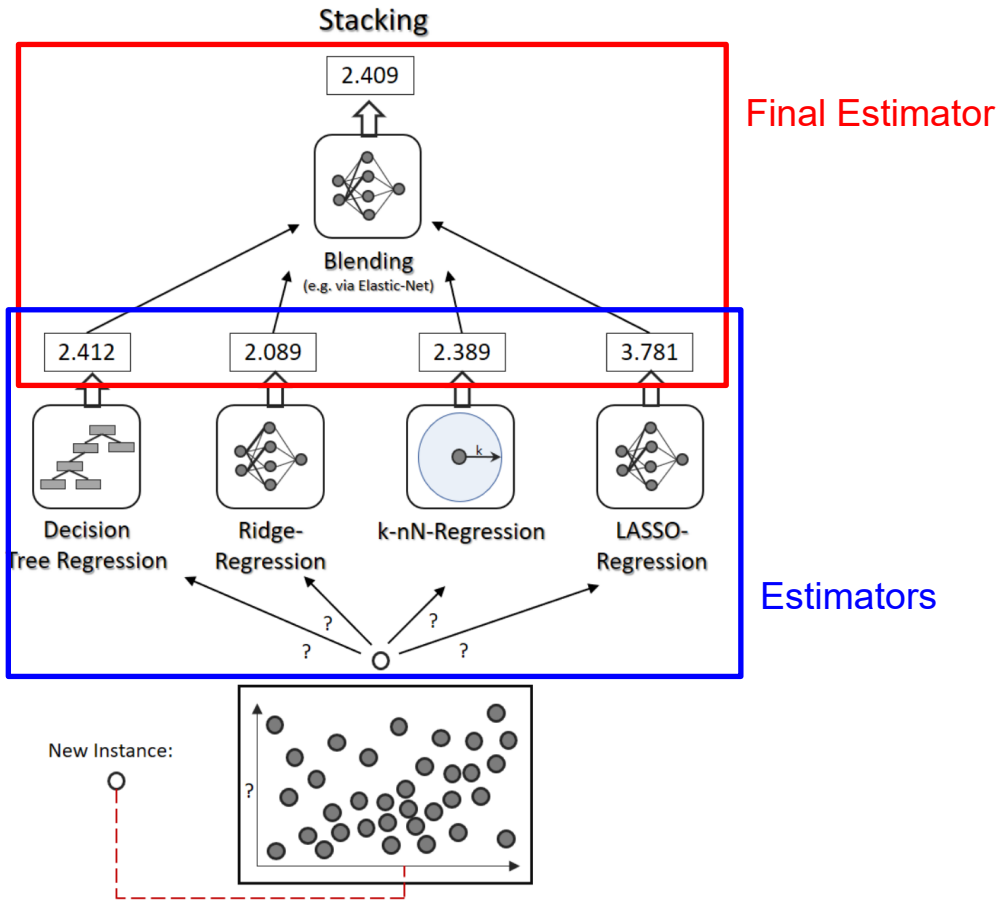
$$H(x) = \alpha_1 h_1(x) + \alpha_2 h_2(x) + ... + \alpha_t h_t(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$$

<stump>

- H(x) = 최종 강한 분류기 (Strong Classifier)
- h = 약한 분류기 (Weak Classifier)
- alpha = 약한 분류기의 가중치 (Weight)
- T = 반복 횟수 (iteration)

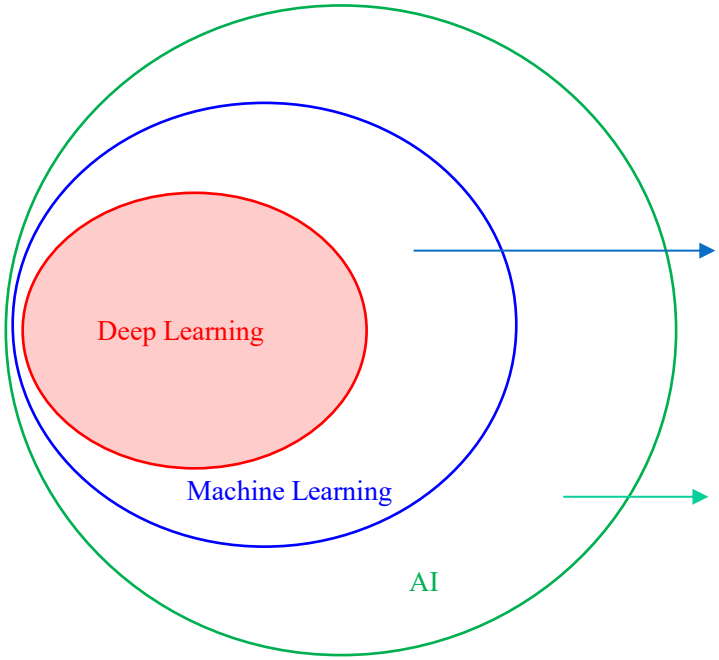
Classical ML

- Stacking
  - "Two heads are better than one"
  - 서로다른 모델들을 조합해서 최고의 성능을 내는 모델을 생성 함



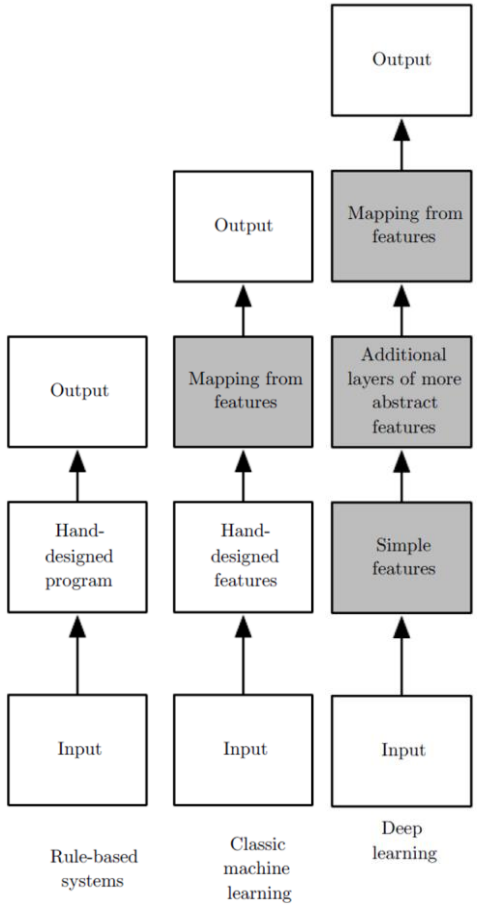
Deep Learning

- Deep Learning?
  - 고차원 특성 (high-level feature) 를 스스로 추출하여 학습



ML but not
Representation Learning
(= Classical ML)

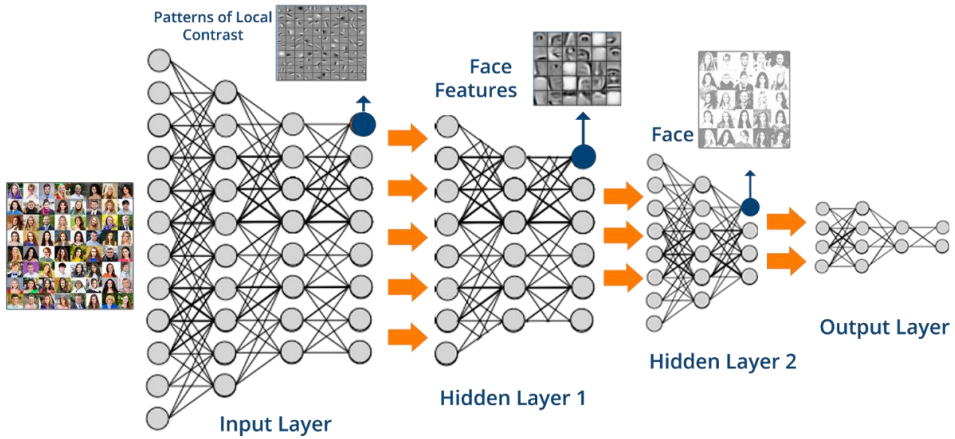
AI but not ML
- 규칙기반 시스템
  (Rule-Based Systems)
- 전문가 시스템 (Expert System)



Source: https://www.deeplearningbook.org/

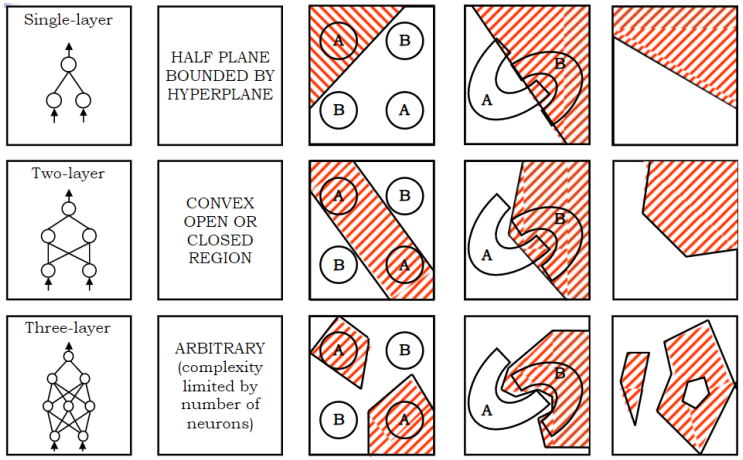
Deep Learning

- Ex 1.8) 얼굴인식
  - 고차원의 얼굴 특성을 자동으로 추출



Source: https://www.slideshare.net/activeeon/machine-learningfordummiesandrewssobralactiveeon-81373804

- 다층 구조의 장점
  - 결정경계 (decision boundary) 를 더 자유롭게 (비선형) 구축할 수 있음
  - 복잡한 Classification 문제의 정확도 향상



Source: https://slideplayer.com/slide/8218292/

정리

구분		내용	특징추출	판단
Non-AI		- 컴퓨터를 이용한 단순계산 - 알려 준 정보만 활용	알려 준 대로 실행	
AI		- 스스로 생각하고 판단한다 - 알려준 것 이상을 처리한다	알려 준 것 이상을 처리	
	AI but not ML	- 원하는 결과를 얻기 위해 Rule에 따라 사람이 정교하게 설계 한다	사람	사람
	Classical ML	- 사람이 사용된 데이터로부터 특징들(features) 을 선정 - 선정된 특징들을 활용하여 컴퓨터가 판단 함	사람	컴퓨터
	DL	- 데이터로부터 고차원 특성 (high-level feature) 를 스스로 추출하여 학습	컴퓨터	컴퓨터