인공지능
- 강인공지능 (일반 인공지능) — 사람과 구분하기 어려운 지능을 가진 컴퓨터 시스템
- 약인공지능 — 아직까지는 제한된 특정 분야에서 사람의 일을 도와주는 보조 역할
  ex) 자율주행, 추천시스템, ...

머신러닝?

규칙을 일일이 프로그래밍 하지 않아도 자동으로 데이터 규칙을 학습하는 알고리즘.

최근 머신러닝 발전은 통계나 수학 이론보다 경험을 바탕으로 발전하는 경우도 많다.

#대표적인 머신러닝 라이브러리

파이썬의 사이킷런

구글의 텐서플로우

페이스북의 파이토치

딥러닝?

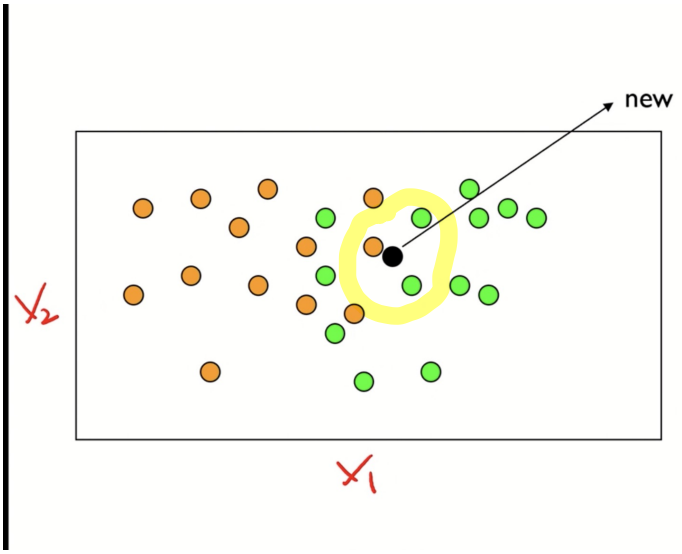
머신러닝 알고리즘 중 인공신경망을 기반으로 한 방법.

1998 - 얀 루큰	손글씨, 숫자 인식
2012 - 제프리 힌턴	이미지 분류
2016 - 딥마인드	바둑

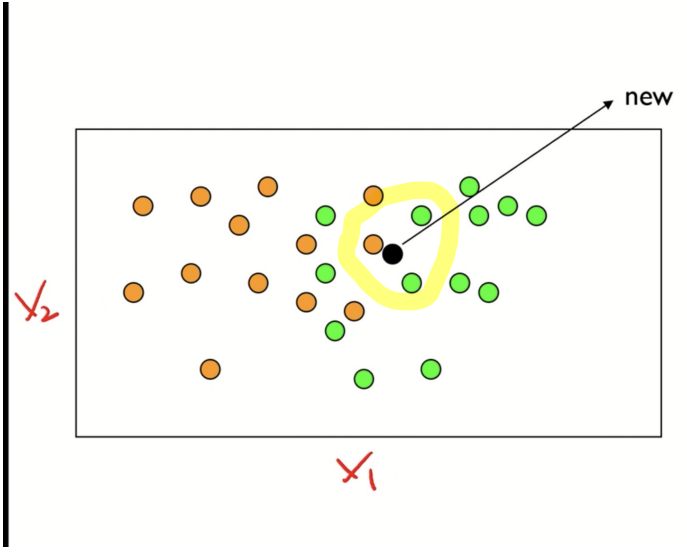
딥러닝은 머신러닝의 한 분야이지만 인공지능 붐을 일으켰다.

K - nearest neighbor (k 근접 이웃) 모델

분류 (Classification)



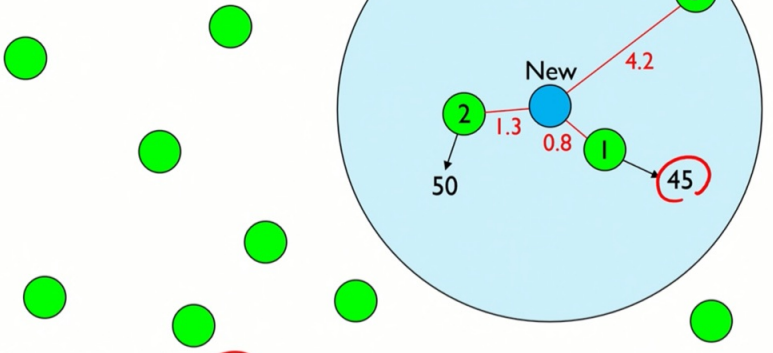
회귀 (Regression)



Weighted KNN - 예시

- Weighted 3NN 예측모델

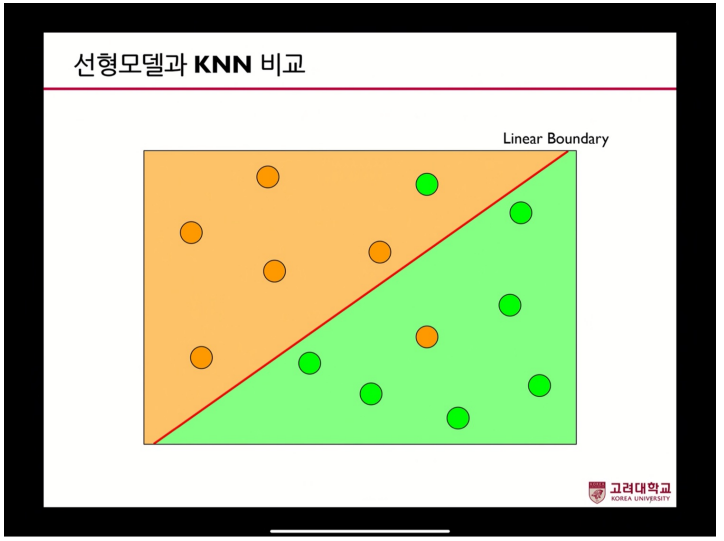
거리에 반비례하게 가중치 적용.

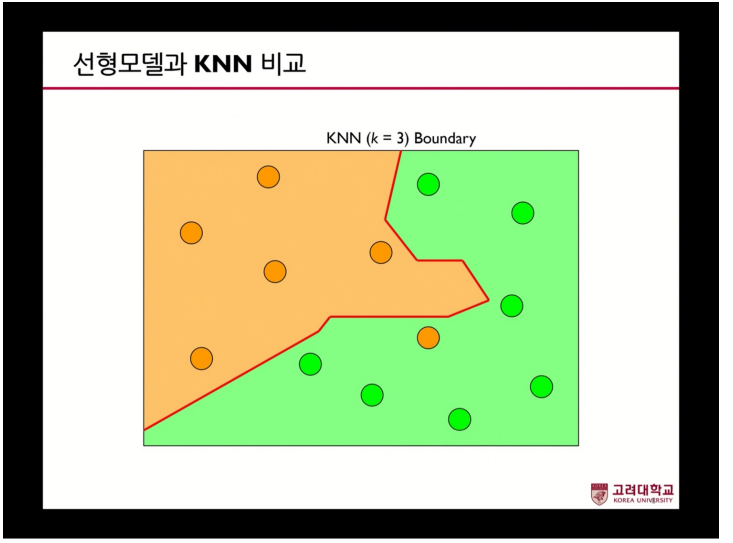


$$\text{New} = \frac{(45 + 50 + 10)}{3} = 35$$

$$\text{New}_{\text{weighted}} = \left(\frac{1}{0.8^2} \cdot 45 + \frac{1}{1.3^2} \cdot 50 + \frac{1}{4.2^2} \cdot 10\right) \Big/ \left(\frac{1}{0.8^2} + \frac{1}{1.3^2} + \frac{1}{4.2^2}\right) = 45.4$$

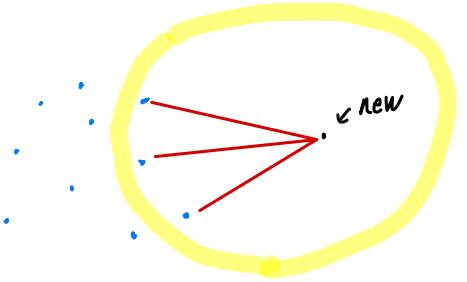
K 근접 이웃 모델의 장점, 단점



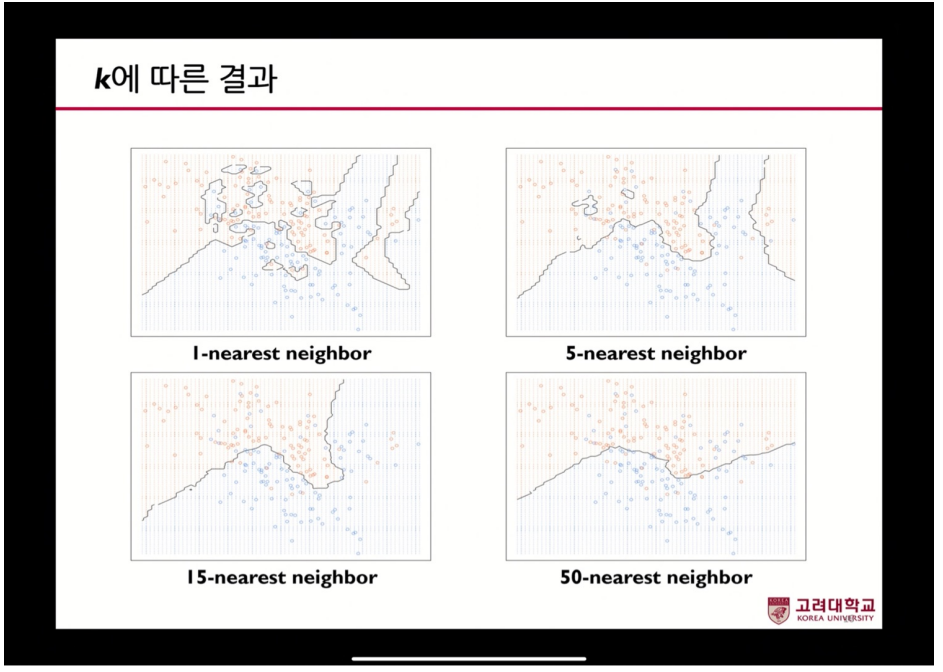


장점 - 학습 데이터의 수가 많은 경우 굉장히 정밀한 모델을 만들 수 있다.
(노이즈의 영향을 크게 받지 않는다.)

단점 - 새로운 데이터가 들어왔을 때 비효율적이다.
- 이웃의 갯수(k)를 최적화 해야 한다.
- 아웃라이어에 취약하다.



최적의 K?



K↓ : 데이터의 지역적인 특성을 지나치게 반영함.
(over-fitting)

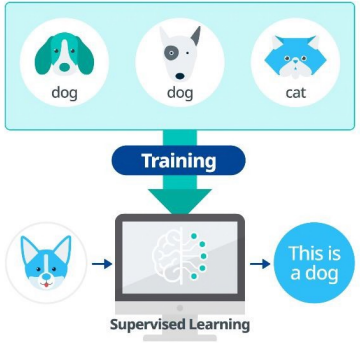
← 최적의 K.

K↑ : 다른 개체의 범주를 너무 많이 포함.
(under-fitting)

최적의 K 구하는 방법? → chapter 3 규제

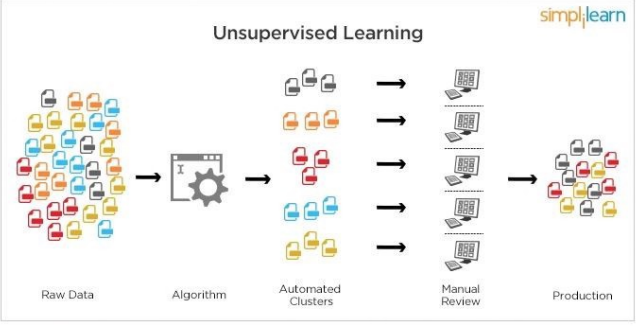
머신 러닝

지도 학습



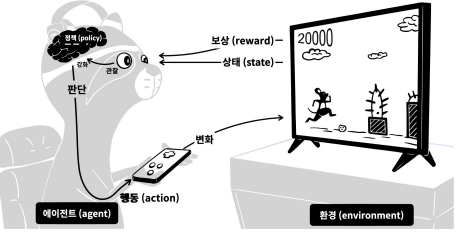
학습 데이터를 통해 컴퓨터를 훈련.
(분류 , 회귀)

비 지도 학습



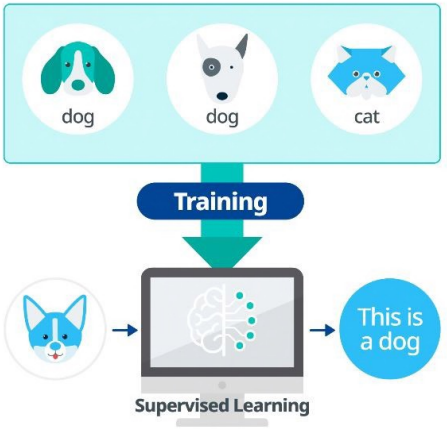
학습 데이터 없이 오직 입력 데이터만으로 컴퓨터를 훈련.
예측이 목적이 아닌 데이터의 구성, 특징을 발견하는 목적으로 사용되는 그루핑 알고리즘.
(뉴스 기사 분류 , 군집 분류, ···)

강화 학습



주어진 환경에서 선택 가능한 행동들 중 보상을 최대화하는 행동 또는 행동 순서를 학습.
(알파고, 게임, ···)

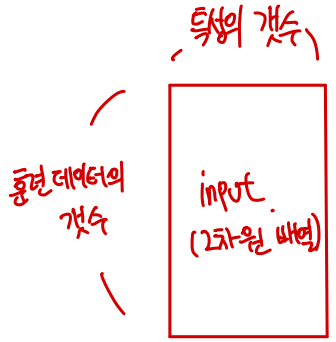
K 근접 이웃 모델은 지도학습



→ 데이터와 정답으로 이루어진 (input) (target) 훈련 데이터가 필요함.

target.

	길이 특성	무게 특성 (feature)	
하나의 데이터 →	[25.4 ,	240.0] ,	[1 ,
" →	[26.3 ,	290.0] ,	1 ,
	⋮		⋮
	[15.0 ,	19.9]]	0]



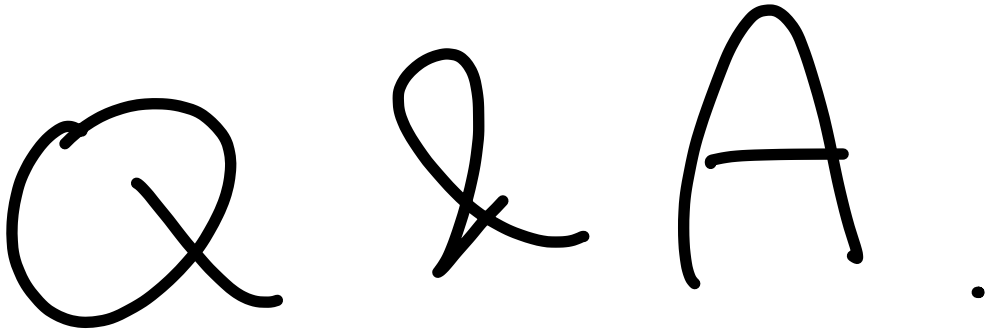
학습에 필요한 훈련 데이터 뿐 아니라, 알고리즘의 성능을 평가하는 테스트 데이터도 필요함.

⇒ 샘플링 편향이 일어나지 않게 샘플 데이터를 잘 섞어서 테스트 세트와 훈련 세트로 나눔.

(교재 chapter2 에서는 넘파이를 이용해 셔플)

(샘플 데이터의 비율을 유지하면서 테스트 세트를 만드는 함수 존재)

데이터 전처리 -> colab 코드로 진행.



출처

https://youtu.be/W-DNu8nardo

https://m.blog.naver.com/k0sm0s1/221863569856