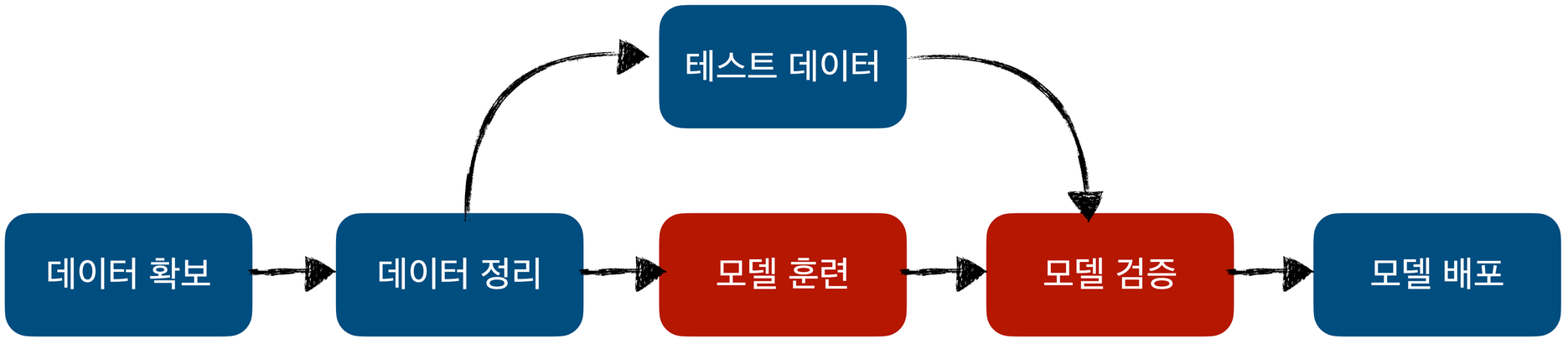
빅데이터 분석 프로그래밍
Application

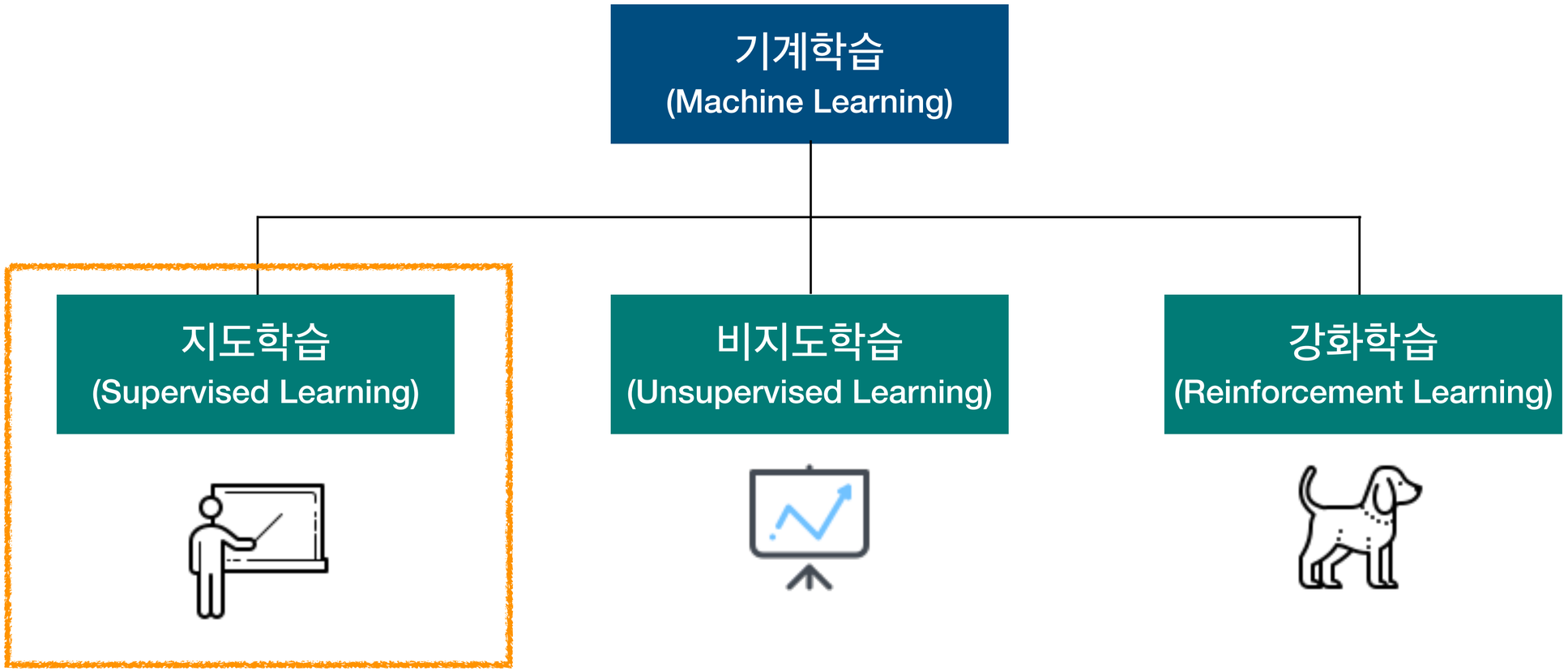
Week-02. 지도학습 Part-2

Jungwon Seo, 2021-Spring

Machine Learning Process



기계학습의 종류


기계학습
(Machine Learning)
지도학습
(Supervised Learning)
비지도학습
(Unsupervised Learning)
강화학습
(Reinforcement Learning)

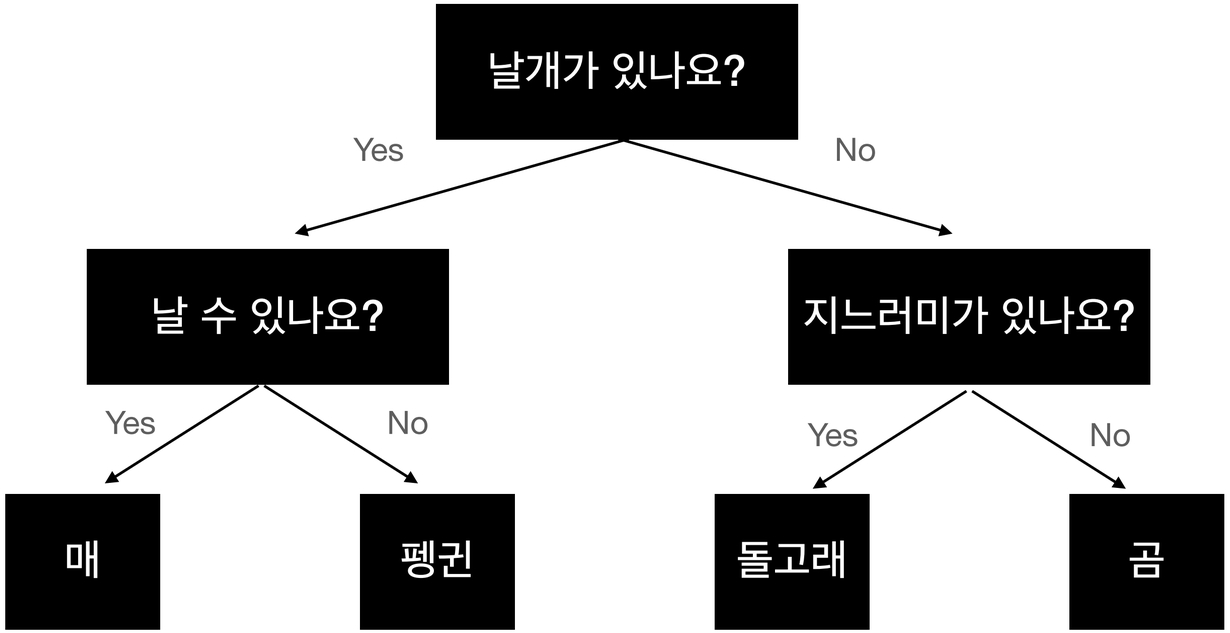
최근 알고리즘 추세

Tree vs Deep Neural Network

- Deep Neural Network(DNN) 이전에 다양한 알고리즘들이 경쟁
  - Support Vector Machine, Tree, Naïve Bayes, K-Nearest Neighbors
- 하지만 DNN이 발전된 이후로 대부분 경쟁에서 이탈
- 단, Tree 계열의 알고리즘은 생존
  - 더 정확히 말하면, Tree를 기반으로 한 앙상블 모델

Decision Tree (의사 결정 트리)

스무고개하기



Decision Tree (의사 결정 트리)

왜 Tree 인가?

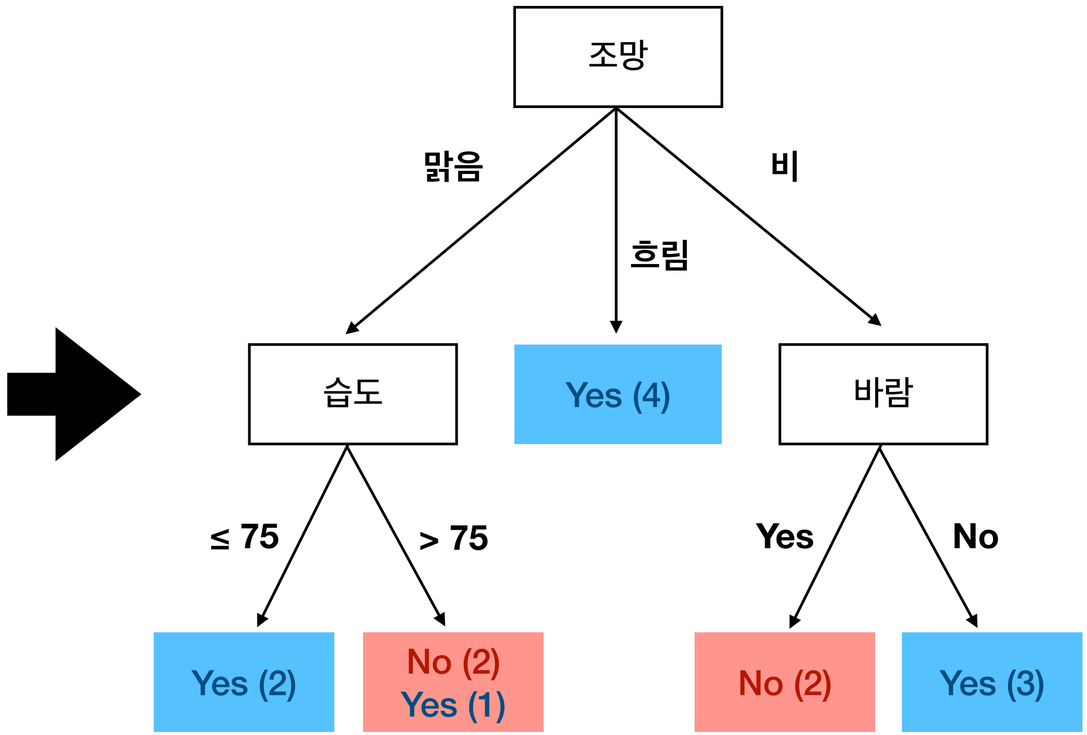
개발자의 나무

그외 모든 사람들의 나무

Decision Tree

의사 결정트리 예제

온도	조망	습도	바람	테니스
보통	맑음	80	No	Yes
더움	맑음	75	Yes	No
더움	흐림	77	No	Yes
시원함	비	70	No	Yes
시원함	흐림	72	Yes	Yes
보통	맑음	77	No	No
시원함	맑음	70	No	Yes
보통	비	69	No	Yes
보통	맑음	65	Yes	Yes
보통	흐림	77	Yes	Yes
더움	흐림	74	No	Yes
보통	비	77	Yes	No
시원함	비	73	Yes	No



Decision Tree

어떻게 노드를 정할 수 있을까?

- 3개의 feature와 2개의 Class로 이루어진 데이터

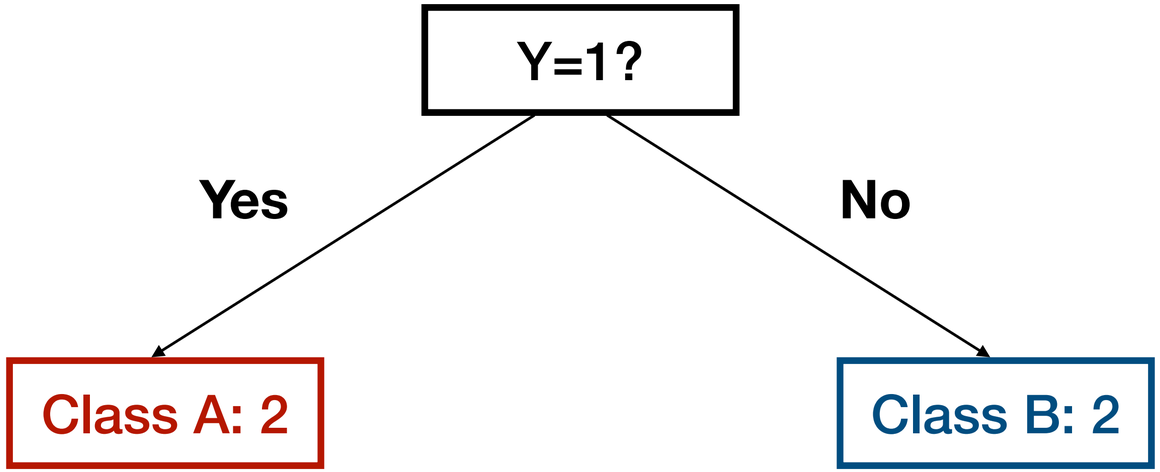
X	Y	Z	Class
1	1	1	A
1	1	0	A
0	0	1	B
1	0	0	B

Decision Tree

어떻게 노드를 정할 수 있을까?

- Feature Y로 나눌 경우, 두 클래스를 분류

X	Y	Z	Class
1	1	1	A
1	1	0	A
0	0	1	B
1	0	0	B



Decision Tree

어떻게 노드를 정할 수 있을까?

- 다른 feature로 나눠 볼시

X	Y	Z	Class
1	1	1	A
1	1	0	A
0	0	1	B
1	0	0	B

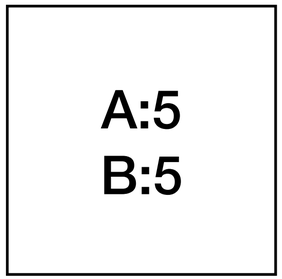
첫번째 split

X		Y		Z	
X=0	X=1	Y=0	Y=1	Z=0	Z=1
	A		A		A
	A		A	A	
B		B			B
	B	B		B	

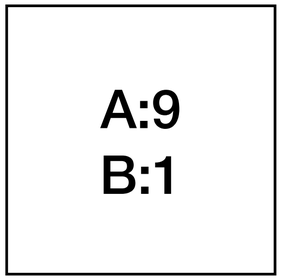
Decision Tree

어떻게 노드를 정할 수 있을까?

- 좋은 트리를 만들기 위해서는, 어떤 feature를 어떻게 나누면 될까?
  - 범주형(categorical)은 각각의 값별로
  - 연속형(continuous)는 가장 분류를 잘하는 값을 기준으로
  - 최대한 같은 Class를 묶어주는 기준으로



Non-homogenous
불순도가 높음



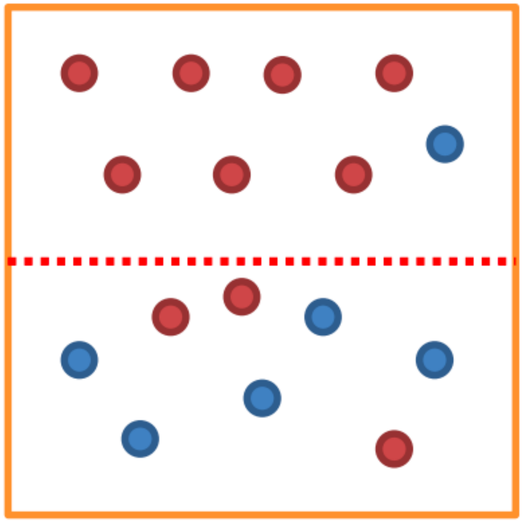
Homogenous
불순도가 낮음

Decision Tree

어떻게 노드를 정할 수 있을까?

$$\text{Entropy} = -\sum_j p_j \log_2 p_j$$

초기 엔트로피 : $$Entropy(A) = -\frac{10}{16}\log_2\left(\frac{10}{16}\right) - \frac{6}{16}\log_2\left(\frac{6}{16}\right) \approx 0.95$$



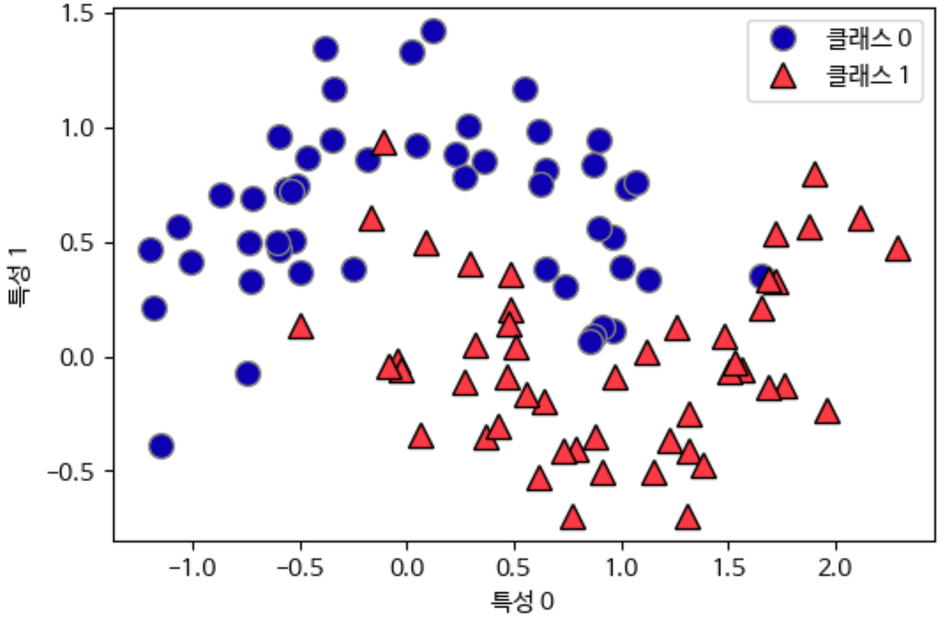
$$Entropy(A) = \sum_{i=1}^{d} R_i \left(-\sum_{k=1}^{m} p_k \log_2 (p_k) \right)$$

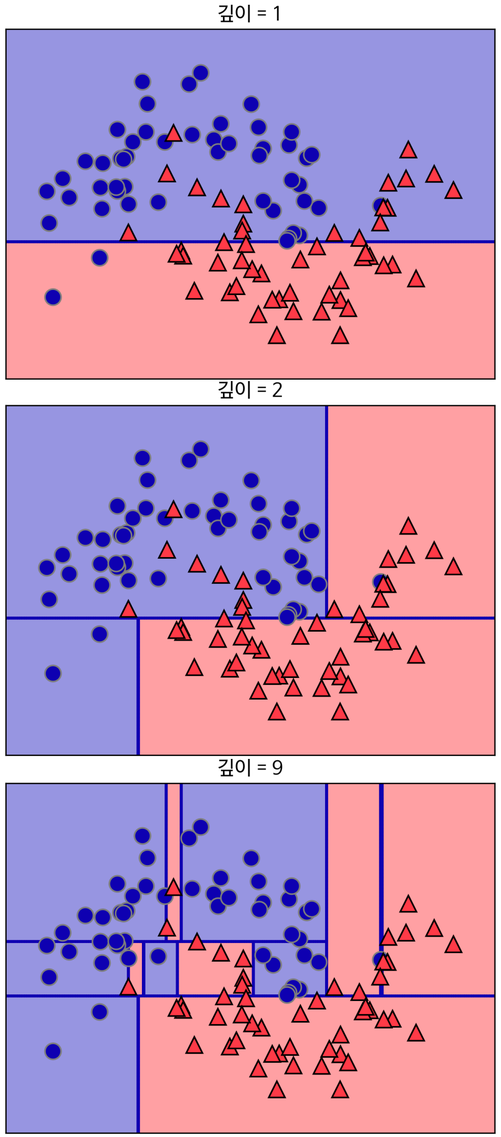
분기 후 엔트로피: $$Entropy(A) = 0.5 \times \left(-\frac{7}{8}\log_2\left(\frac{7}{8}\right) - \frac{1}{8}\log_2\left(\frac{1}{8}\right)\right) + 0.5 \times \left(-\frac{3}{8}\log_2\left(\frac{3}{8}\right) - \frac{5}{8}\log_2\left(\frac{5}{8}\right)\right) \approx 0.75$$

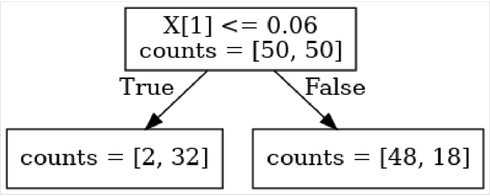
https://ratsgo.github.io/machine%20learning/2017/03/26/tree/

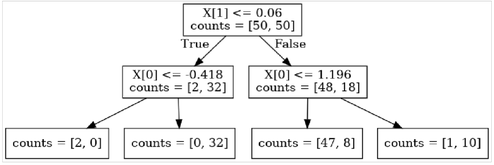
Decision Tree

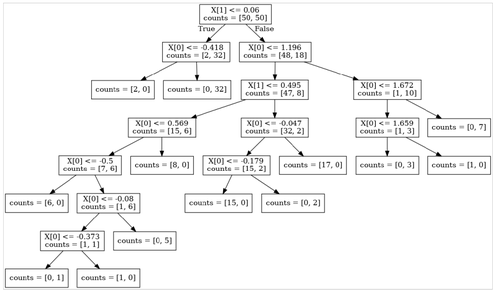
깊이가 깊을 수록 좋을까?











Decision Tree

깊이가 너무 깊으면 Overfitting 발생

- 오버피팅이란?
  - 훈련데이터에만 최적화가 되고, 새로운 (unseen) 데이터에는 좋은 성능을 보이지 못하는 상태
  - 일반화된 모델이 아니라는 뜻!
- 가지치기
  - 사전 가지치기: 최대 깊이, 최대 leaf 노드를 정해 트리의 생성을 중단
  - 사후 가지치기: 데이터 포인트가 적은 노드 삭제
  - sklearn에서 사전 가지치기에 해당하는 값들을 조정 가능: 하이퍼파라미터 튜닝

Decision Tree
의사 결정 트리의 장단점

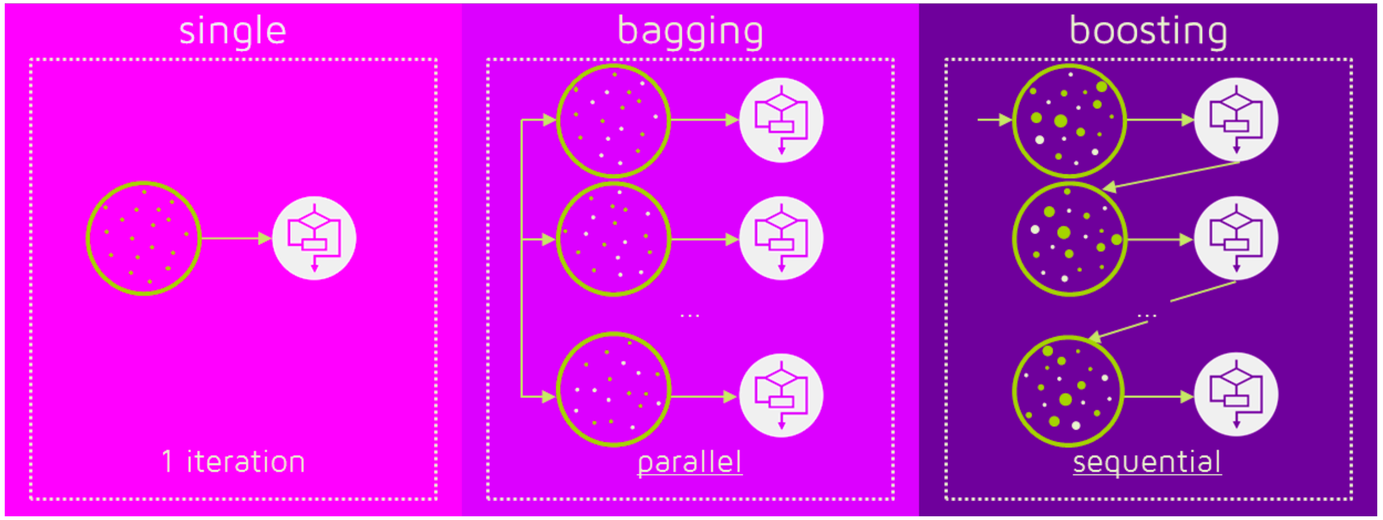
- 장점
  - 모델을 시각화하기가 쉽다 (설명 가능하다)
  - feature의 정규화나 전처리 과정이 필요없다

- 단점
  - 과대적합(overfitting) 경향이 크다
  - Regression 문제에서는 훈련데이터에 존재하는 값 이외의 값은 예측하지 못한다.

Ensemble

여러 모델을 연결하여 더 강력한 모델을 만드는 기법

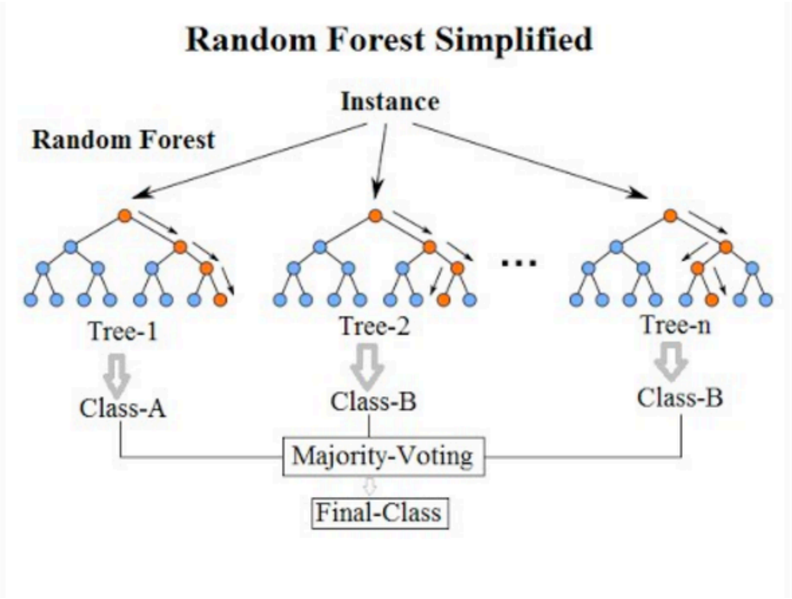
- Bagging
  - 데이터 포인트를 다르게 추출하여 여러개의 모델을 훈련시키는 방법
- Boosting
  - 간단한 모델(약한 학습기)를 많이 연결하여 복잡한 모델(강한 학습기)를 만드는 방법



https://quantdare.com/what-is-the-difference-between-bagging-and-boosting/

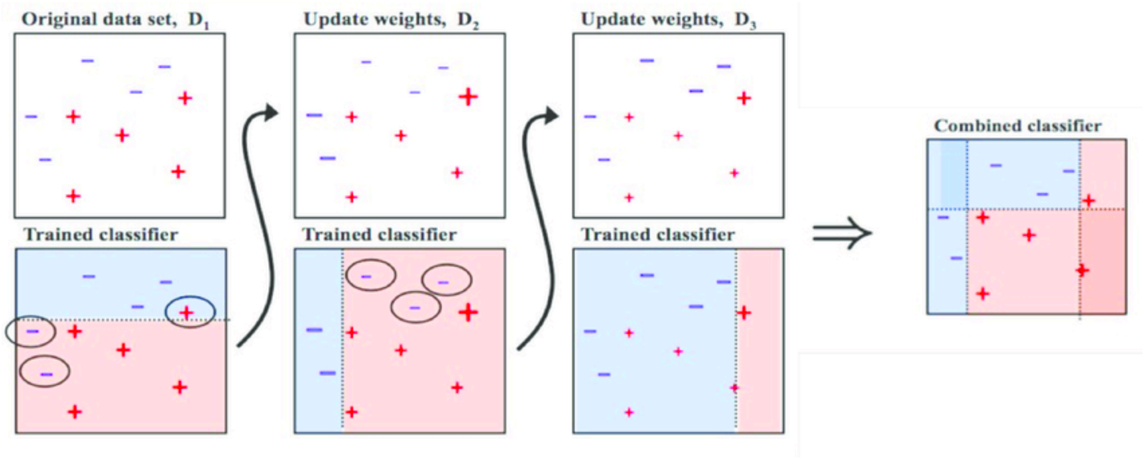
Bagging
집단지성

- 원리
  - n개의 데이터 셋에서 n개의 데이터 셋을 무작위로 추출: 중복 가능성이 있음
  - 다른 데이터셋에 의해 자연스럽게 다른 Decision Tree 가 생성됨
  - 최종적으로 이들의 예측값을 종합해서 결정: 분류는 다수결, 회귀는 평균



Boosting
오답노트

- 원리
  - 의도적으로 모델을 안 좋게 학습: 예)트리의 깊이를 얕게
  - 앞선 모델에서 잘못 분류한 값들에 대해 가중치를 증가 시킨후 그 데이터를 기반으로 훈련 또 반복
  - 최종적으로 결정을 내릴때는, 각각의 모델에 다른 가중치를 부여한 상태로 종합



Multivariate Analysis of the Vector Boson Fusion Higgs Boson - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Training-of-an-AdaBoost-classifier-The-first-classifier-trains-on-unweighted-data-then_fig3_306054843 [accessed 4 Jun, 2021]

E.O.D