

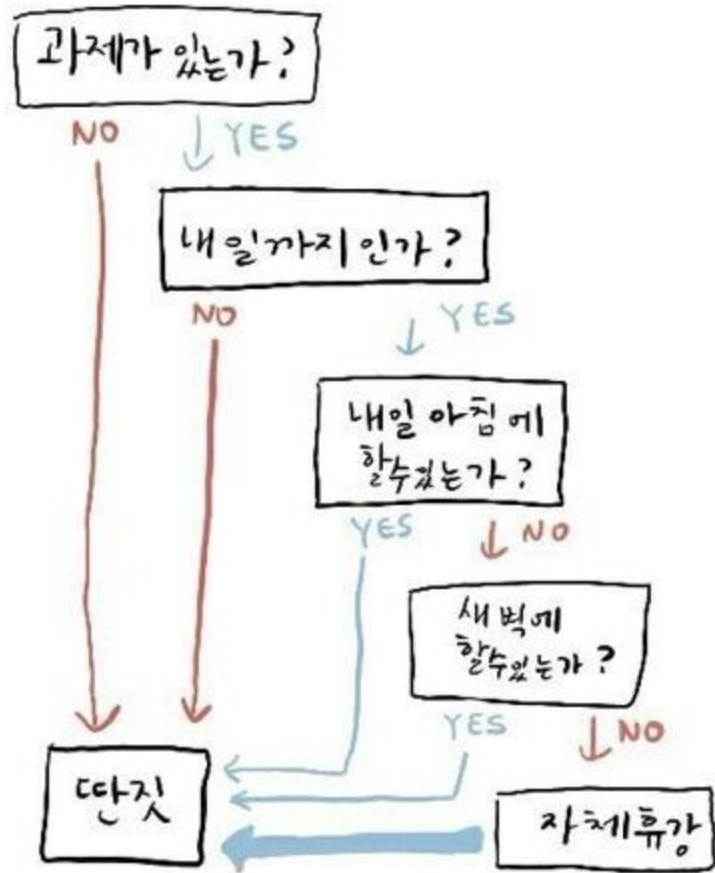
Decision Tree & Ensemble methods

Decision Tree

· 질문 하나에
필수적으로 질문이 필요하다



과제 알고리즘



Decision Tree 장점

- 이해하기 쉽다
- 전처리가 단순
- 빠르다
- 다양한 종류의 변수를 다룰 수 있음
- 모형의 시각화
- 통계적 가정이 적음

Decision Tree 단점

- 과적합(overfitting)

- 결과의 불안정

- 최적화가 어려움

- 학습시키기 어려운 문제들이 있음(예: XOR)

- Imbalanced data에 취약

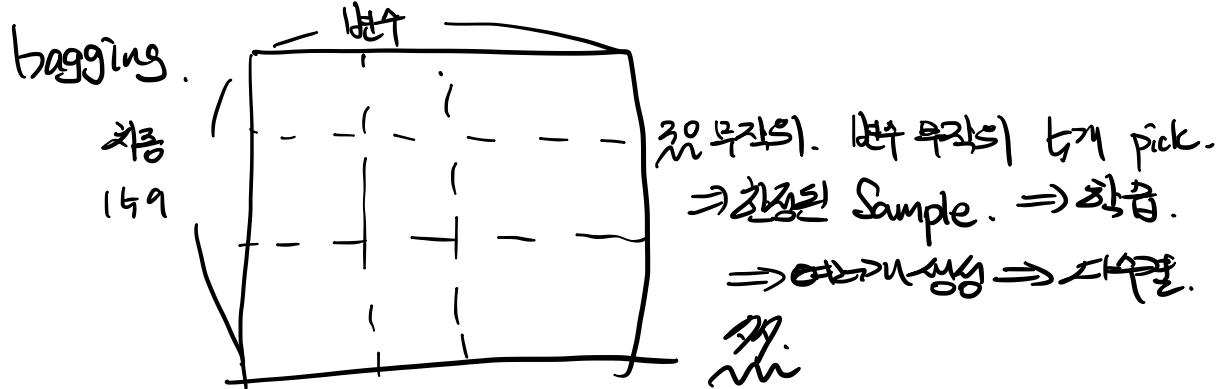
학습데이터가 평형한 경우
변수가 조금만 변해도
결과는 달라진다.

Heural Network가 유리.

극단적으로 불균형한 데이터에 취약하다.

앙상블(Ensemble)

앙상블

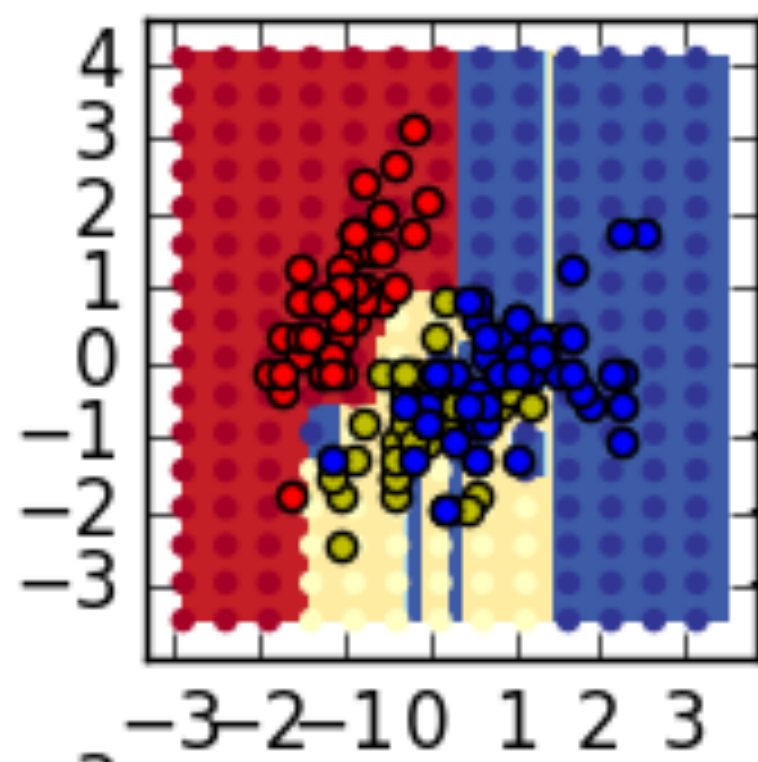


- 하나의 모형은 under-/over-fitting 될 수 있음
- 앙상블: 여러 개의 모형을 만들어 다수결/평균을 사용
Voting. Classifier.
- 배깅(bagging 또는 bootstrap aggregation):
 1. 데이터에서 일부 변수의 샘플을 무작위로 뽑는다
 2. 샘플에 모형을 학습시킨다
 3. 1-2를 반복하여 여러 개의 모형을 만든다
 4. 위의 모형들의 예측의 다수결/평균으로 예측한다
- Random Forest: DT + bagging

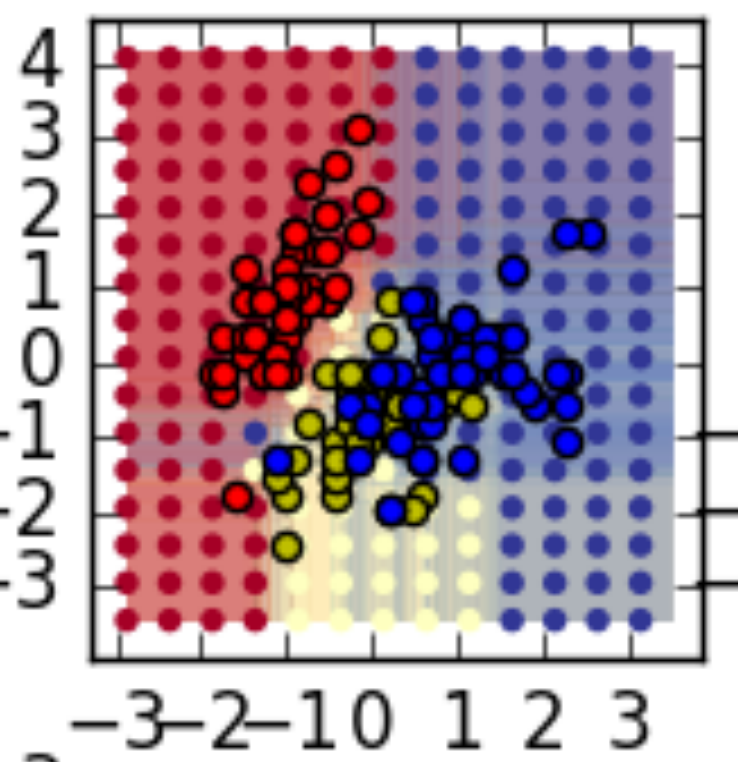
Decision tree
bagging

=> Random
forest

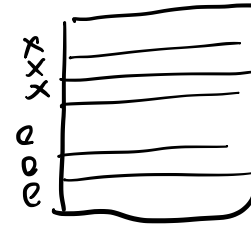
DecisionTree



RandomForest



부스팅(boosting) 이런 것 X: 부정확 O: 정확 상호보완적.



X에 가중치를 준다.
⇒ 새로운 모델 학습.
모델1 + 모델2. + 모델3.
↓
플래닛이 가장. $\uparrow \oplus$

⇒ 지속적으로 추가.

1. 모든 데이터에 동일한 가중치
2. 데이터로 모형1을 학습
3. 모형1이 틀린 데이터의 가중치 높임
4. 데이터로 모형2를 학습
5. 3-4의 과정을 반복

경사 부스팅(Gradient Boosting)

1. 데이터로 모형1을 학습
2. 모형1의 예측과 실제의 오차
3. 위의 오차를 모형2를 학습
4. 3-4의 과정을 반복

오차 + 모형1 \rightarrow 실제 데이터
 \rightarrow 오차만 예측하는 모형2. + 오차2
 \Rightarrow 모형2 + 오차2 \Rightarrow 오차1.
 \Rightarrow 오차2 \rightarrow 모형3 + 오차3
 \Rightarrow 점점 오차↓.

경사 부스팅(Gradient Boosting)

- 실제값 = 모형1의 예측 + 모형1의 오차
- 모형1의 오차 = 모형2의 예측 + 모형2의 오차
- 모형2의 오차 = 모형3의 예측 + 모형3의 오차
- 실제값 = 모형1의 예측 + 모형2의 예측 + ... + 아주 작은 오차
- 다변량 회귀 모형의 dummy coding (one-hot encoding)
 $\Rightarrow a \ b \ c \Rightarrow a \begin{bmatrix} 1.0.0 \end{bmatrix} \ b \begin{bmatrix} 0.1.0 \end{bmatrix}$
 $c \begin{bmatrix} 0.0.1 \end{bmatrix} \Rightarrow \text{숫자형으로}$