Decision Tree : 분류를 위한 방법론(협의)
모델의 의사결정 규칙을 나무 형태로 표현
의사결정 규칙(나무의 경로)에 따라 입력변수 데이터에 비추어 출력변수(범주형) 예측

분할된 영역에 동일한 클래스 데이터가 최대한 많이 존재하도록 형성
=> 변별력 좋은 질문 요구

불순도 : 특정 범주 안에 다른 class의 데이터가 섞인 정도
결정 트리는 불순도를 최소화하여야 한다. == 변별력 좋은 질문

Entropy == 사건이 갖는 정보량. ID3, C4.5, C5.0 알고리즘에서 사용되는 불순도 지표
정보이득(IG) : 분기 전후 엔트로피의 차이값.

지니불순도 : CART에서 클래스 동질성을 측정하는 지표
1 - 정상 데이터 비율^2 - 불량 데이터 비율^2
=> 오분류 최소화 기준. 추가적인 데이터 구분 여부 지표

CART 알고리즘에서는 모든 조합에 대해 Gibi Impurity를 계산후 가장 낮은 지ㅛ표에섭 분기

정보이득 : 어떤 질문을 기준으로 나눠야 하는가에 대한 지표
추가 분할 시의 혼잡도 감소량

결정 트리 알고리즘은 정보 이득 최대화하는 방향으로 학습 결정

나무구조 생성
이진 분할 -> IG가 가장 큰 입력변수 선택 후 입력변수 공간 분할
-> 다시 IG가 가장 큰 입력변수를 선택하여 입력변수 공간 분할
-> 반복
Overfitting 문제를 해결하기 위해 끝 노드 일부를 제거(가지치기)

가지치기 종류 : 사전 / 사후
사전 가지치기 : 특정 조건 만족시 알고리즘 중단
사후 가지치기 : 나무 완성 후 하단 노드부터 무의미한 Subtree 제거
비용 복잡도 가지치기 수행 : 나무 모델에 대한 비용 복잡도 지표 최소하

분기 지표를 선택 기준을 불순도가 아닌 오차(=분산)를 사용

연속형 입력변수 => 구간 분할로 범주화(단일 경계값 방법 사용)... 정보 이득 고려

앙상블 기법; 일종의 집단 지성
다수 데이터 집합에 대해 각각 머신 러닝 모델을 학습하고 학습된 모델의 예측을 결합하여 최종 예측 수행... 다양한 학습 모델 사용
Strong Learner by Weak Leaners' Voting

Hard Voting -> 다수결 투표
Soft Voting(Avg) -> 예측값 단순 평균
Soft Voting(Weighted Sum) => 가중치 합

알고리즘
Bagging : Bootstrap Aggregating 의 약자.Bootstrap이용
bootstrap : 복원추출. 부트스트랩으로 만들어진 여러 데이터셋으로 Weak Learner 생성
-Random forest
부트스트랩 샘플링 기반 여러 개의 의사결정나무를 생성하여 다수결 또는 평균에 따라 예측
무작위 변수 선택기법 사용; 입력변수 일부 랜덤 선택 -> 선택 변수 중 가장 큰 불순도 감소량을 보이는 입력변수 선택
Randmo Forest는 변수 중요도를 산출하여 제공

Boosting : 반복적으로 모델 업데이트. 이전 iteration의 결과에 따라 데이터셋 샘플 가중치 부여
반복할 때마다 각 샘플의 중요도에 따라 다른 분류기 생성. 최종적으로 모든 iteration 모델 결과 반영
- 베이스 모델을 각 학습 라운드마다 순차 학습. 결합
- 이전 모델의 오예측 부분을 차후 모델이 중점 학습
- Ada Boost 계열과 GBM 계열이 많이 사용
- GBM 원리 : 잔차 지속 학습
이전 결합 모델의 예측 오류는 아직까지 학습 못한 특징이라고 가정하고 잔차를 학습하는 알고리즘
회귀문제에서의 학습 : 손실함수를 최소화하기 위해 일반적으로 함수의 음의 기울기가 0인 되는 방향으로 베이스 모델의 학습 파라미터를 조절
음의 기울기 = 잔차. 잔차의 순차적 최소화 과정 == 손실함수 최소화 방법
- XGBoost GBM과 대전제는 동일. but 학습 목적식에 규제항 추가
~ 규제항은 트리 복잡성에 패털니 부여 => 과적합 방지
~ 규제항을 표상하는 하이퍼 파라미터를 통해 가지치기 횟수 결정
- light GBM : 성능이 좋지만 학습시간이 긴 XGBoost 대체. 다른 tree와 구별되는 수직확장 구조
- CatBoost : 기존 GBM 모델들의 범주형 입력변수 처리 문제 해결

Stacking : weak learner 예측 결과를 바탕으로 meta learner 학습시켜 최종 예측값 결정
meta learner 또한 학습이 필요하며 사용되는 데이터는 학습용 데이터에 따른 각 weak learner 예측 확률값 모음