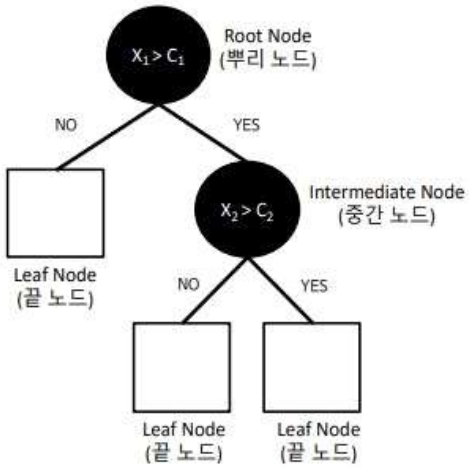
의사결정나무(Decision Tree)

: 학습 데이터에 내재한 패턴을 분석하여 새로운 데이터를 예측 및 분류하는 모델

자료(X)와 목표(Y)에 따라 분류 기준과 정지 규칙을 지정하여 나무 생성

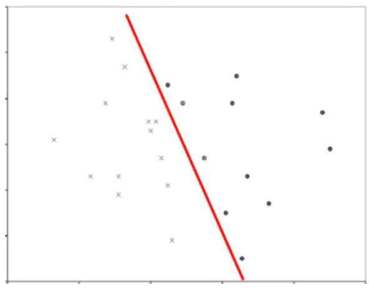
장점



1. 이해하고 적용하기 쉽다: 나무 구조(If-then 규칙)
2. 의사결정과정을 설명(해석)할 수 있다
3. 중요한 변수 선택에 유용하다
   : 상단 변수일수록 중요하다
4. 데이터의 통계적 가정이 필요 없다

단점

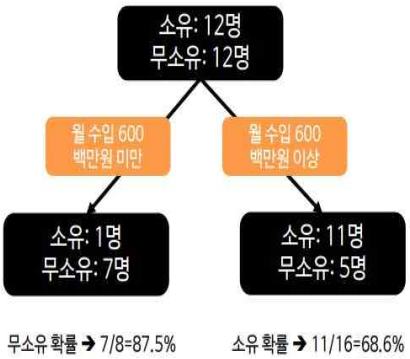
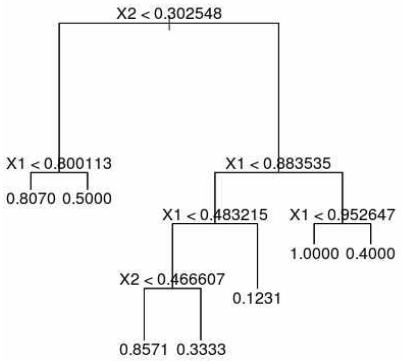
1. 좋은 모형을 만들려면 많은 데이터가 필요하다
2. 모형을 만드는데 상대적으로 많은 시간이 필요하다(Tree building)
3. 데이터 변화에 민감하다
   데이터에 따라 모델이 변화하기 때문이다
   학습과 테스트 데이터의 도메인이 유사해야 한다(smalll domain gap)
4. 선형 구조 데이터를 예측할 때는 더 복잡하다

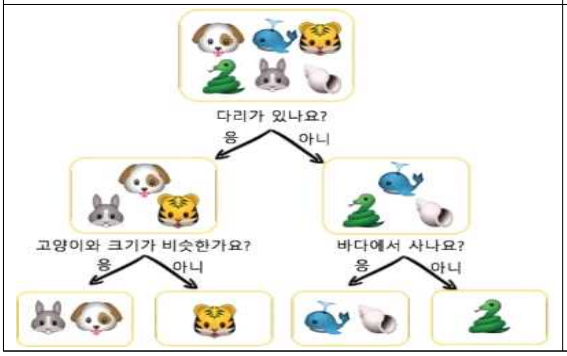
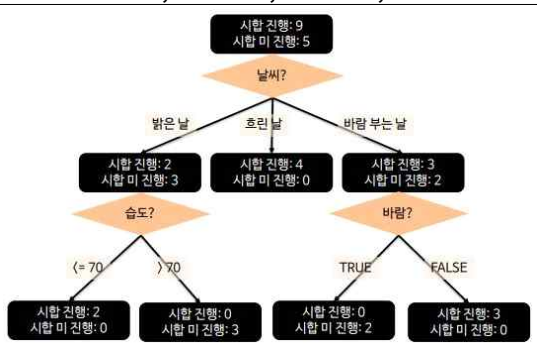


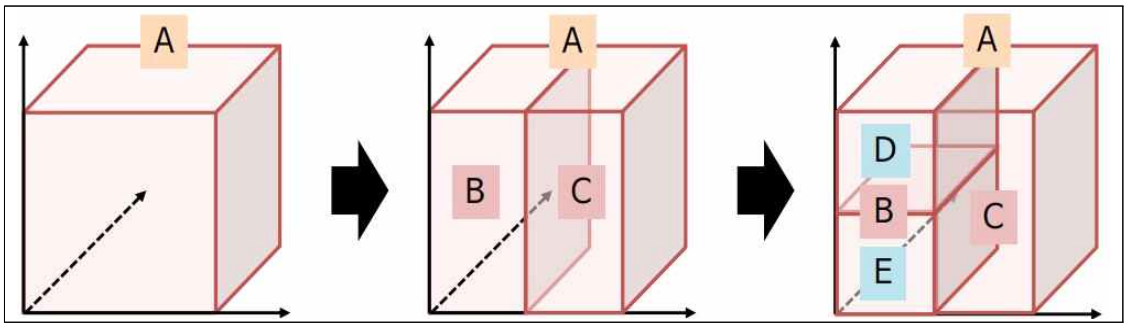
의사결정나무의 데이터 분석 단계

1. 다변량 변수 데이터
2. 트리 구조 모델 학습
   Tree and Rule 구조: 나무 모델 → 규칙 → 결과
   1) 한번에 설명변수 하나씩
   2) 데이터를 2개 혹은 그 이상의 부분 집합으로 나누어
   3) 데이터 순도가 균일해지도록 재귀적 분할(Recursive Partitioning)
   종료 조건: 데이터 순도 균일
      분류: 끝 노드에 비슷한 범주(클래스)를 갖는 관측 데이터끼리
      회귀: 끝 노드에 비슷한 수치(연속된 값)를 갖는 관측 데이터끼리
3. 추론(판별)
   분류: 끝 노드에서 가장 빈도가 높은 종속변수(y)를 새로운 데이터에 부여
   회귀: 끝 노드의 종속변수(y)의 평균을 예측값으로 반환

재귀적 분할 알고리즘	분순도 알고리즘(불순도 지표, 분할 기준)
CART (Classification And Regression Tree)	지니 지수(Gini index)
C4.5, C5.0	엔트로피 지수(Entropy index), 정보 이익(Information Gain)
CHAID(Chi-square Automatic Interaction Detection)	카이제곱 통계량(Chi_square Statistic)

	분류 나무 (Classification Tree)	회귀 나무 (Regression Tree)
목표 변수	범주형 변수 → 분류	수치형 변수 → 예측
분류 알고리즘과 불순도 지표	1. CART : 지니 지수 2. C4.5 : 엔트로피 지수, 정보 이익, 정보이익비율 3. CHAID : 카이 제곱 통계량	CART : F 통계량과 분산 감소량 (실제값과 예측값의 평균 차이가 작도록 $argmin \left\| Y - \hat{Y} \right\|_2$)
결과	분류(판별, 추론) 소속 집단 판단, 경향성도 확률로 표현 가능	회귀 끝 마디 집단의 평균 (예측에서는 회귀나무보다 신경망과 회귀분석이 더 좋다)
		

이진 분할(binary split)	다중 분할(multi-way splt)
CART	C4.5, C5.0, CAID, etc
	



재귀적 분할 알고리즘	CART	C4.5	CHAID
불순도 알고리즘	지니 지수	엔트로피 지수, 정보 이익, 정보 이익 비율	카이제곱 통계량
분류	O	O	O
회귀	O	O	X
목표 변수	범주, 수치	범주, 수치	범주
분리	Binary	Multi-way	Multi-way
나무 성장	완전 모형 생성(full tree) 후 가지치기		최적 모형 개발 (완전 모형 생성X)
가지치기(교차 검증)	학습데이터로 학습 검증데이터로 검증	학습데이터만 사용	X

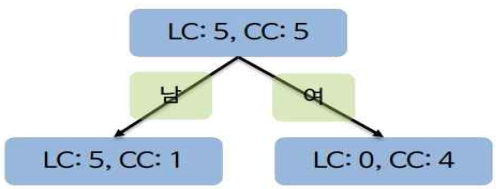
분류 나무

CART

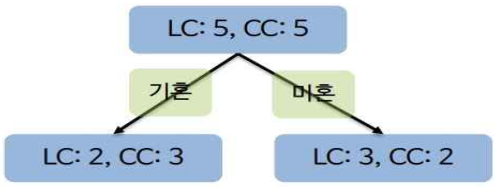
학습 데이터로 나무 생성, 검증 데이터로 가지치기

- 지니 지수(불확실성)는 낮을수록 좋다

$$G.I(A) = \sum_{i=1}^{d}(R_i(1-\sum_{k=1}^{m}\hat{p}_{ik}^2))$$



성별에 의한 분류



결혼유무에 의한 분류

$G(root) = 1-\left(\frac{5}{10}\right)^2-\left(\frac{5}{10}\right)^2 = 0.5$

$G(남) = 1-\left(\frac{5}{6}\right)^2-\left(\frac{1}{6}\right)^2 = 0.278$

$G(여) = 1-\left(\frac{0}{4}\right)^2-\left(\frac{4}{4}\right)^2 = 0$

$G(성별) = 가중치 * G(남) + 가중치 * G(여)$
$= \left(\frac{6}{10}\right)(0.278) + \left(\frac{4}{10}\right)(0) = 0.167$

$G(root) = 1-\left(\frac{5}{10}\right)^2-\left(\frac{5}{10}\right)^2 = 0.5$

$G(기혼) = 1-\left(\frac{2}{5}\right)^2-\left(\frac{3}{5}\right)^2 = 0.48$

$G(미혼) = 1-\left(\frac{3}{5}\right)^2-\left(\frac{2}{5}\right)^2 = 0.48$

$G(결혼) = 가중치 * G(기혼) + 가중치 * G(미혼)$
$= \left(\frac{5}{10}\right)(0.48) + \left(\frac{5}{10}\right)(0.48) = 0.48$

C4.5, C5.0

학습 데이터만 이용해서 나무 성장과 가지치기

- 정보 이론: 엔트로피

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

$\log_2$로 계산하는 이유: bit 단위 계산

$-\log_2$로 계산하는 이유: $\log_2(\frac{1}{2}) = -1$이므로 +로 전환 필요

- 정보 이익(IG. Information Gain)는 클수록 좋다

$$IG = E(before) - E(after)$$

성별에 의한 분류

E(root) $= -\left(\frac{5}{10}\right) log_2\left(\frac{5}{10}\right) - \left(\frac{5}{10}\right) log_2\left(\frac{5}{10}\right) = 1$

E(남) $= -\left(\frac{5}{6}\right) log_2\left(\frac{5}{6}\right) - \left(\frac{1}{6}\right) log_2\left(\frac{1}{6}\right) = 0.65$

E(여) $= -\left(\frac{0}{4}\right) log_2\left(\frac{0}{4}\right) - \left(\frac{4}{4}\right) log_2\left(\frac{4}{4}\right) = 0$

E(성별) = 가중치 $*$ E(남) + 가중치 $*$ E(여)
$= \left(\frac{6}{10}\right)(0.65) + \left(\frac{4}{10}\right)(0) = 0.39$

IG (성별) = E(Root)-E(성별)=0.61

← 불 확실성 감소량(클수록 좋음)

결혼유무에 의한 분류

E(root) $= -\left(\frac{5}{10}\right) log_2\left(\frac{5}{10}\right) - \left(\frac{5}{10}\right) log_2\left(\frac{5}{10}\right) = 1$

E(기혼) $= -\left(\frac{2}{5}\right) log_2\left(\frac{2}{5}\right) - \left(\frac{3}{5}\right) log_2\left(\frac{3}{5}\right) = 0.971$

E미혼) $= -\left(\frac{3}{5}\right) log_2\left(\frac{3}{5}\right) - \left(\frac{2}{5}\right) log_2\left(\frac{2}{5}\right) = 0.971$

E(결혼) = 가중치 $*$ E(남) + 가중치 $*$ E(여)
$= \left(\frac{5}{10}\right)(0.971) + \left(\frac{5}{10}\right)(0.971) = 0.971$

IG (결혼) = E(Root)-E(결혼) = 0.029

← 거의 변화가 없음. 작, 결혼 여부는 큰 영향을 주지 못함

- 정보 이익 비율(Information Gain Ratio)

목적: 가지 수가 많을수록 IG가 높아지는 경향을 보인다

이진 분할 vs. 다중 분할

해결방법: IV(Intrinsic Value) 도입하여 정보 이득률 정규화

가지가 많으면 감점

$$IV(A) = -\sum_{k=0}^{n} \frac{1}{n} \log_2(\frac{1}{n}) \rightarrow IGR(A) = \frac{IG(A)}{IV(A)}$$

성별에 의한 분류

IG (성별) = E(Root)-E(성별)=0.61

IV(성별) $= -\left(\frac{6}{10}\right) log_2\left(\frac{6}{10}\right) - \left(\frac{4}{10}\right) log_2\left(\frac{4}{10}\right) = 0.97$

IGR(성별) = IG(성별)/IV(성별) = 0.61/0.97 = 0.63

결혼유무에 의한 분류

IG (결혼) = E(Root)-E(결혼)= 0.029

IV(결혼) $= -\left(\frac{5}{10}\right) log_2\left(\frac{5}{10}\right) - \left(\frac{5}{10}\right) log_2\left(\frac{5}{10}\right) = 1$

IGR(결혼) = IG(결혼)/IV(결혼) = 0.029/1 =0.029

* 다중 분할이면 $-\log_2$이 계속 붙게 되고 IV가 1를 넘기도 한다

끝없는 분할 → 과적합(Overfitting)

해결방법: 나무 성장 중단

1. 성장 멈추기(Stop condition)
    - 깊이를 나타내는 파라미터
        트리의 길이(depth) n까지만 성장한다
    - 나무 성장을 특정 조건에서 중단
        불순도 지표(t)가 얼마 이상/이하이면 중단한다
    - 노드 내의 최소 관측치의 수
        노드의 관측치 수가 n개보다 작으면 중단한다
    - 불순도 최소 감소량
        불순도 변화량(Δt)이 n보다 작으면 중단한다

2. 가지치기(Pruning)
    완전 모형 생성(각 영역에 동일 클래스만 존재)하고
    가지치기(필요 없는 가지 제거)
    데이터를 합치는 개념(버리는 것X)
    성장 멈추기 보다 성능이 우수하다
    비용함수를 최소로 하는 분기를 찾는다

회귀 나무

입력 데이터의 결과 예측

: 데이터가 도달한 끝 노드 데이터들의 평균으로 결정

불순도 측정 방법

: 제곱 오차 합(the sum of the squared errors)

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y_i})^2$$

성능 평가 방법

: 예측 모델 평가 방법(RMSE. Root Mean Square Error)

앙상블(Ensemble)

여러 모델을 함께 사용한다(의사결정나무, kNN, LDA, 로지스틱, 등)

1. 같은 종류의 모델을 여러 개
2. 다양한 종류의 모델을 여러 개

설명보다 예측이 중요할 경우에 사용

의사결정나무의 앙상블

1. Random Forest
    - Bootstrap: 데이터에서 복원 추출하여 여러 샘플 추출
    - 무작위로 예측 변수를 선택하여 모델 구축

        의사결정나무는 기준 지표를 사용하여 예측 변수를 선택했으나 random forest는 무작위로 예측 변수를 선택한다

    앙상블 결과 결합: 분류 → 투표, 회귀 → 평균화

    BUT. 나무에서 숲이 되면서 의사결정과정을 해석하지 못하게 되었으나 결과 분석을 통해 설명 변수 중 중요한 변수는 판별할 수 있다
2. Boosted Trees

의사결정나무 api

분류나무 클래스

sklearn.tree.DecisionTreeClassifier(criterion='gini', max_depth=None, random_state=None)

criterion: 불순도 알고리즘{"gini", "entropy"}, default = "gini"

max_depth: 성장 멈추기의 길이 파라미터(int, default = None)

회귀나무 클래스

sklearn.tree.DecisionTreeRegressor(criterion='mse', max_depth=None, random_state=None)

criterion: 불순도 알고리즘

{"mse", "friedman_mse", "mae", "poisson"}, default = "mse“

max_depth: 성장 멈추기의 길이 파라미터(int, default = None)

트리 출력

matplotlib.pyplot

sklearn.tree.plot_tree()

MSE/RSME 성능평가

sklearn.metrics.mean_squared_error(y, y_pred, squared=True)

squared: (bool, default = True)

True → MSE(Mean Squared Error)

False → RMSE(Root Mean Square Error)