의사결정 트리

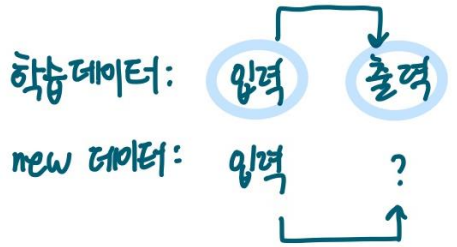
Intro

의사결정 나무는 어떤 종류의 알고리즘일까?

0. 지도학습 vs 비지도학습

지도학습

- 학습 데이터의 입력과 출력 간의 관계를 학습하여 규칙이나 함수로 표현되는 모델을 찾음.



비지도학습

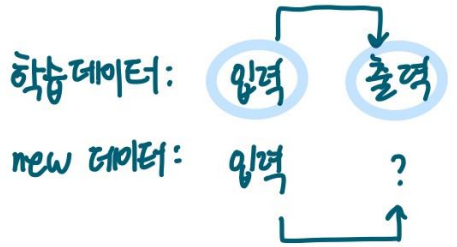
- 출력 정보가 없는 학습 데이터에 대해서 데이터의 패턴을 발견하는 학습

** 출력 = 종속 변수 , 입력 = 독립 변수

0. 의사결정 나무는 지도학습

지도학습

- 학습 데이터의 입력과 출력 간의 관계를 학습하여 규칙이나 함수로 표현되는 모델을 찾음.



데이터 유형

분류	회귀
출력 값이 범주형 데이터인 경우	출력 값이 연속형 데이터인 경우

의사결정나무는
분류문제 회귀 문제 둘 다 학습

* 입력값은 범주형이든 연속형이든 상관 없다.

** 출력 = 종속변수 , 입력 = 독립변수

4.1.1. 의사결정 나무의 개념

◎ root node 루트 노드 : 최상위 계층에 존재하는 노드 (=모든 샘플의 집합)

- 트리의 가장 위에 위치, 가장 중요한 속성이 배치 → 상위의 노드일수록 중요한 속성을 배치

◎ leaf node or terminal node 단말 노드 : 하위에 다른 노드가 연결되어 있지 않은 노드, 트리의 맨 아래에 위치

◎ branch node 분기 노드 : 트리 구조에서 최소한 한 개의 자식 노드를 갖는 노드

◎ parent node 부모 노드 : 임의의 노드 바로 위에 있는 노드

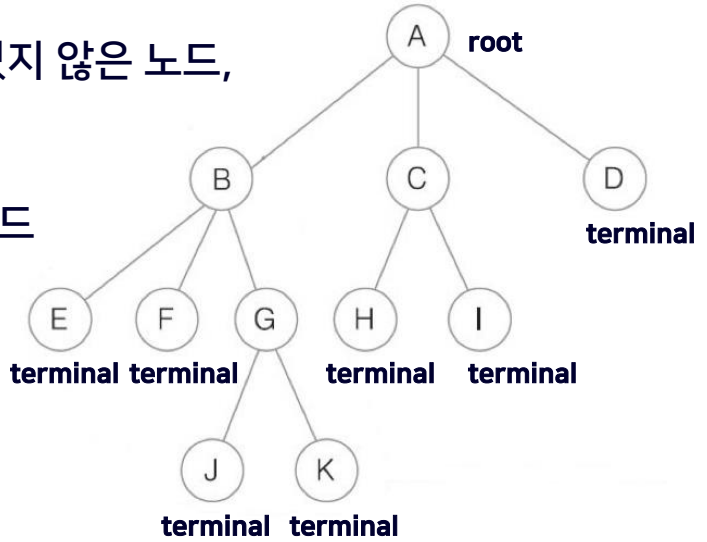
◎ child node 자식 노드 : 임의의 노드 바로 아래에 있는 노드

◎ sibling node 형제 노드 : 같은 노드를 가지는 노드

◎ level 레벨 : 루트 노드에서 임의의 노드까지 방문한 노드의 수 ex) 레벨 1, 1세대

◎ depth 깊이 : 트리의 최대 레벨

◎ degree 차수 : 하위 트리의 개수 = 간선 수



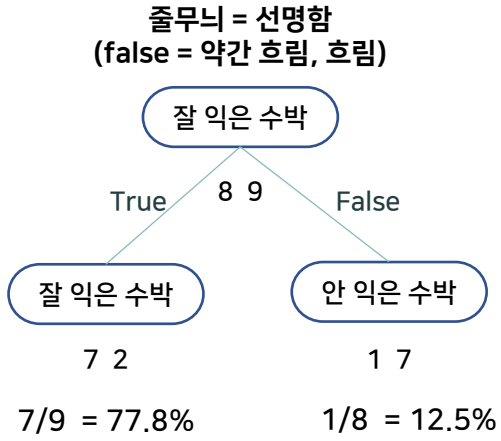
4.1.2. 기본 프로세스

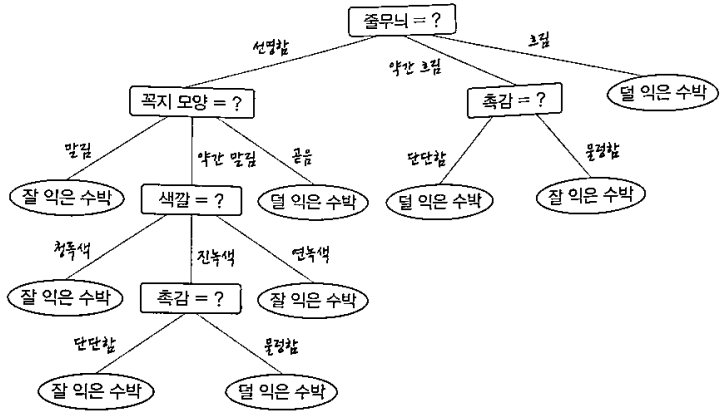
- 입력 값(판정질문)을 기반으로 결과를 분류하거나 예측
- 뒤집힌 나무 모양 → 위-뿌리, 아래-잎
- if – then 구조: If 판정질문, then 최종결론
- 직관적, 판정 질문(조건문)을 알 수 있다.

  → 분류된 이유, 근거가 필요한 분야가 적용됨.

  → 상위의 노드일수록 중요한 속성.

  예) 의료, 은행 대출, 카드 발급 대상 등





4.1.3. 의사나무 나무의 궁극적인 목표

- **분기된 노드가 최대한 같은 클래스에 속하도록 !**

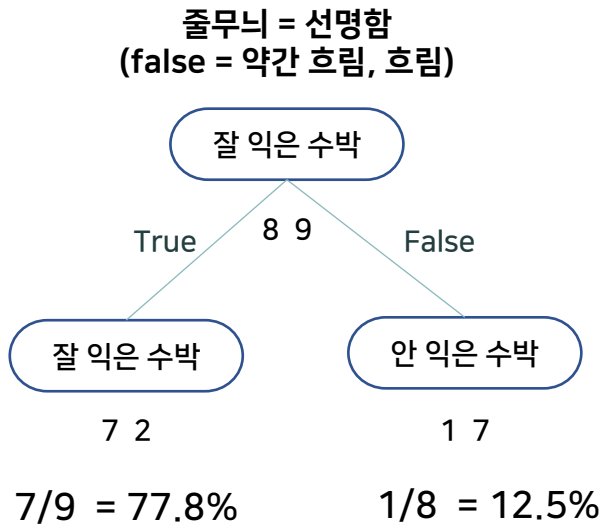
  <첨부된 그림>에서 이상적인 목표: 잘 익은 수박의 표본의 수가 9:0

  But 현실적으로 불가능

- **순도 purity**

  순도↑ → 성능 ↑

  ⇔ 불순도, 불확실성(entrophy)



- **한 단계 위의 결정 결과는 정답 범위를 한정시키는 역할을 한다. (p. 90, 5 line)**

  '어떤 속성이 상위에 있냐', '속성값을 어떤 기준으로 구분하느냐' 에 따라 모델의 성능이 달라진다.

4.1.3. 의사나무 나무의 궁극적인 목표

- 분기된 노드가 최대한 같은 클래스에 속하도록 !

  <첨부된

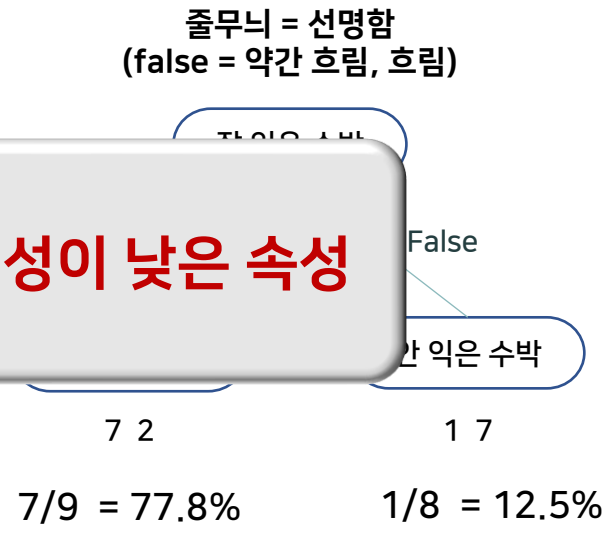
  But 현

- 순도 p

  순도↑ → 성능 ↑

  ⇔ 불순도, 불확실성(entrophy)

최적의 분할 속성 선택 = 순도가 높은 속성 = 불확실성이 낮은 속성



- 한 단계 위의 결정 결

  ‘어떤 속성이 상위에 있냐’, ‘

  , 5 line)

  라진다.

정보 이득(entrophy), 정보 이득율, 지니계수

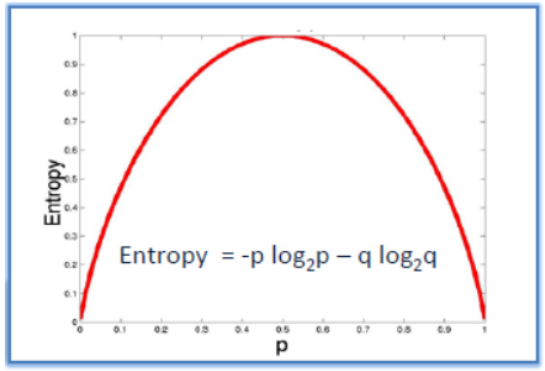
4.2.1 정보이득

- **정보 이득 Information Gain: 정보량. 엔트로피를 이용하여 계산.**

  정보이득 ↑ → 정보의 가치 ↑

  엔트로피 ↓ → 정보 이득 ↑

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



Entropy = -0.5 $\log_2$0.5 – 0.5 $\log_2$0.5 = 1

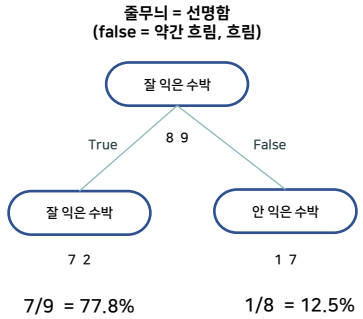
*** log의 밑이 2인 이유:** IG는 컴퓨터의 정보량을 계산하는 것 → 컴퓨터의 정보량은 bit수로 계산

2진수 : 0과1

*** '-'를 붙이는 이유:** log의 진수에 확률이 들어감 -> 무조건 값이 음수임

- **정보이득: E(전) – E(내가 보고자 하는 노드)**

  정보이득 = A (A는 값) → 불확실성이 A만큼 감소했다.



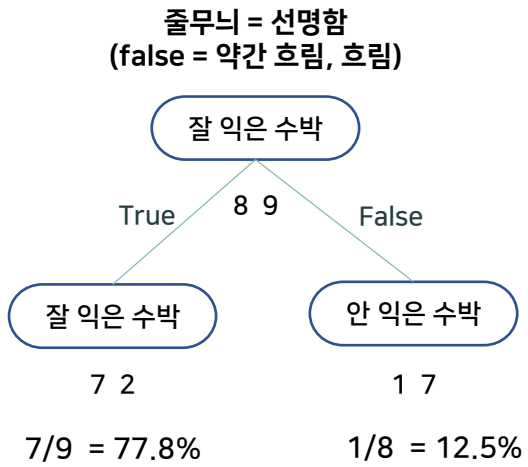
4.2.1 정보이득

- **가중치G: 분기노드의 샘플 수 ↑ → 영향력↑**

$$\text{가중치 } G = \frac{\text{구분된 노드의 표본의 수}}{\text{분기노드의 표본의 수}}$$

<첨부된 그림>에서 왼쪽 노드의 엔트로피를 계산할 때, 가중치는 9/17이고,
오른쪽 노드의 가중치는 8/17이다.

Information Gain 계산 필기 첨부



4.2.2 정보이득율

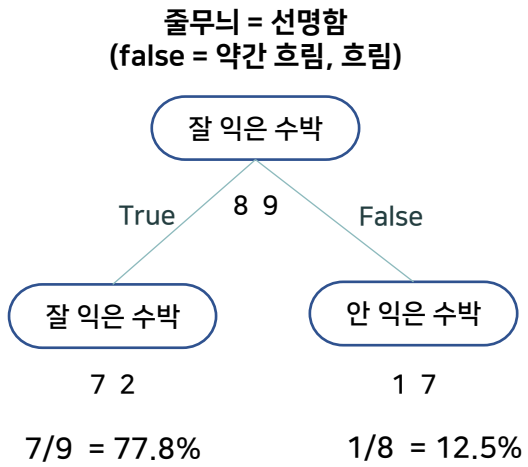
- **정보 이득율: 정보 이득의 확장된 개념.**

  등장 배경: -

  내재 값 intrinsic value

$$\text{Gain ratio}(\text{D}, \text{a}) = \frac{Gain(D,a)}{IV}$$

Information Gain ratio 계산 필기 첨부

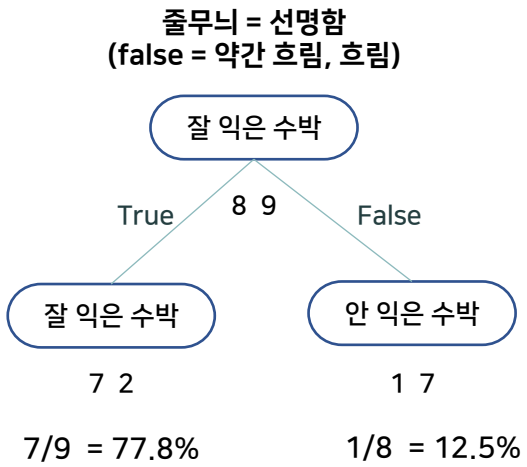


4.2.3 지니계수

- **지니계수: entropy와 같은 개념. 불확실성 의미함.**

  Entropy와 계산 방법이 다름

$$G.I(A) = \sum_{i=1}^{d} \left(R_i \left(1 - \sum_{k=1}^{m} p_{ik}^2 \right) \right)$$



*복원 추출 개념을 씀 → 두 번 측정하면 더 정확해서

*P확률은 얼마나 맞는지, 확실한지를 구하는 것 반대로 지니계수는 불확실성을 구하는 것이기 때문에 1에서 확률의 합을 뺀다.

- **가중치 G 구하기**

Gini 계수 계산 필기 첨부

Kaggle 문제 설명

https://github.com/dannylisa/ml-study/blob/main/%EB%A8%B8%EC%8B%A0%EB%9F%AC%EB%8B%9D/04.%20%EC%9D%98%EC%82%AC%EA%B2%B0%EC%A0%95%20%ED%8A%B8%EB%A6%AC/kaggle/4w_yeonjulee_DecisionTree.ipynb