Decision Tree
(Information Theory)

Jeonghun Yoon

Terms

정보량

Entropy

Information gain정보 이득

Gini Impurity

Decision Tree

CART 알고리즘

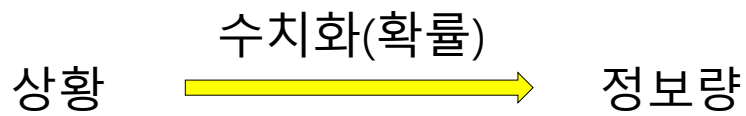
Information Theory

상황 1)
오늘 하루 종일 맑다. 뉴스에서 "내일은 맑다"라는 일기 예보를 듣는다.

상황 2)
오늘 하루 종일 맑다. 뉴스에서 "내일은 큰 비가 온다"라는 일기 예보를 듣는다.

두 가지 상황을 비교해보자. 어느 상황에서 더 많은 정보를 얻었을까?

Information Theory(정보 이론)는 주어진 상황에서 우리가 얻을 수 있는 정보량을 수치화해주는 기능을 제공한다.



Information Theory

상황 1)
오늘 하루 종일 맑다. 뉴스에서 "내일은 맑다"라는 일기 예보를 듣는다.

직관적으로, 오늘 날씨가 맑기 때문에 내일의 날씨도 맑을 확률이 높다.
이 상황에서 얻을 수 있는 정보(놀라운 정도)는 적다고 볼 수 있다.

상황 2)
오늘 하루 종일 맑다. 뉴스에서 "내일은 큰 비가 온다"라는 일기 예보를 듣는다.

직관적으로, 오늘 날씨가 맑기 때문에 내일 비가 올 확률은 상황 1에 비해 상대적으로 낮다. 이 상황에서 얻을 수 있는 정보(놀라운 정도)는 상황 1에 비해 상대적으로 높다.

Information Theory

간단한 예이지만,
어떤 사건의 확률과 그것이 전달하는 정보량은 반비례 관계임을 알 수 있다.

정보량은 놀라운 정도라고 할 수 있다.

일반적으로 확률이 낮은 사건일수록 더욱 놀랍고 정보량은 크다.

이 관계는 정보 이론(Information Theory)이 토대로 삼는 가장 중요한 원리이다.

그러면, 정보량을 어떻게 **수치화** 할 수 있을까?

$$p(\text{내일} = \text{큰비} \mid \text{오늘} = \text{맑음})$$
$$p(\text{내일} = \text{맑음} \mid \text{오늘} = \text{맑음})$$

정보량

x : random variable(랜덤 변수)

- 확률 변수는 샘플공간에서(발생 가능한 이벤트, 사건) 실수로의 함수이다.

  ex1) input : 주사위를 던졌을 때 3의 눈이 나오는 사건, output : 3

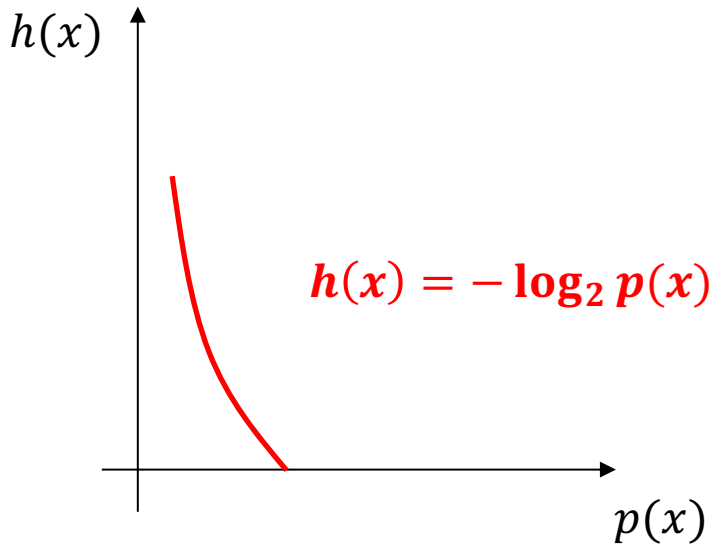
  ex2) input : 동전을 던졌을 때 앞면이 나오는 사건, output : 1(T)

$p(x)$: x가 특정한 값을 가질 때, 랜덤 변수 x의 확률

$h(x)$: x의 자기 정보량(self-information)

$$\boldsymbol{h(x) = -\log_2 p(x)}$$

정보량



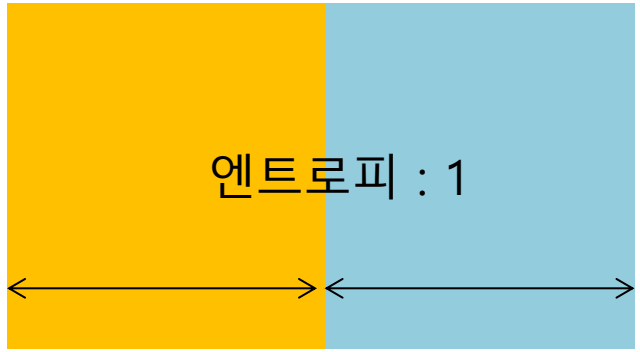
- $h(x)$의 단위는 bit이다. 밑이 e인 자연 로그를 사용한다면 단위는 nat 가 된다.

- 랜덤 변수 x가 a의 값을 갖는다고 가정하고, $p(a) = \frac{1}{1024}$ 라고 하자. 즉 1024번에 한 번 정도 발생하는 시간이다. 이 때 a의 정보량은 10bits가 된다. $p(a) = 1$이라고 하자. a의 정보량은 0bit이다.

엔트로피 (Entropy)

랜덤 변수 x가 가질 수 모든 값(사건)에 대한 정보량(자기 정보량)의 평균으로 데이터의 불순도$_{\text{impurity}}$에 관한 척도이다.

표본 전체가 완전히 균질(균일)하면 엔트로피는 0이 되고 표본 전체가 완전히 균등하게 분할 되어 있으면 이 값은 1이 된다.



엔트로피 (Entropy)

엔트로피의 수학적인 정의

$$H(x) = -\sum_{x} p(x) \log_2 p(x)$$

$$H(x) = -\int_{-\infty}^{\infty} p(x) \log_2 p(x)\, dx$$

위의 경우는 x가 이산 값을 가지는 경우이고, 아래의 경우는 연속적인 값을 갖는 경우이다.

즉, 랜덤 변수 x의 엔트로피를 통하여 x에서 얻을 수 있는 정보를 계산할 수 있다.

- 엔트로피↑ → 정보량↑ → 발생 가능성↓ → 혼란도, 불순함↑
- 엔트로피↓ → 정보량↓ → 발생 가능성↑ → 혼란도, 불순함↓

※ 랜덤 변수 x의 함수 $h(x)$의 평균 : $\sum_x h(x)p(x)$

Entropy 예제 1

랜덤 변수 x : 주사위를 던지는 사건에서 나올 수 있는 값이다. 즉 1, 2, 3, 4, 5, 6 이다.

x의 엔트로피를 구하여라.

모든 사건이 같은 확률을 가질 때 엔트로피는 최대가 된다.

무질서 정도(불순도)가 최대가 된다. 즉, 불확실성(uncertainty)이 최대이다.

- 여섯 개의 사건이 모두 같은 가능성을 가지므로 어떤 눈이 나올 지 가늠하기 매우 어렵다.

한 사건이 1의 확률을 가지고, 나머지 사건이 0의 확률을 가지면 엔트로피는 0이다.

- 어떤 눈이 나올 지 가늠하기 매우 쉽다.

데이터의 카테고리

계량 데이터 (metric data, numerical data)

- 어떤 사물 또는 개념을 양적으로(quantitatively) 표시한 데이터
- 데이터를 좌표평면상으로 수치화 시킬 수 있다. ⇒ 단순한 계산으로 거리 측정 가능

ex) 점수(score), 매출액, 국내 총생산 등

비 계량 데이터 (non-metric data, categorical data)

- 어떤 사물 또는 개념을 질적으로 (qualitatively) 구분한 데이터
- 양(quantity)의 개념이 없다. ⇒ 단순한 계산으로 데이터간의 거리 측정 불가능

ex) 지역번호, 행정 구역, 혈액형 등

: 경북(054) 제주(064) 전북(063) ⇒ 전북이 제주도와 더 가깝나?

데이터 종류에 따른 대표적인 분류기

양적 분류기 (quantitative classifier)

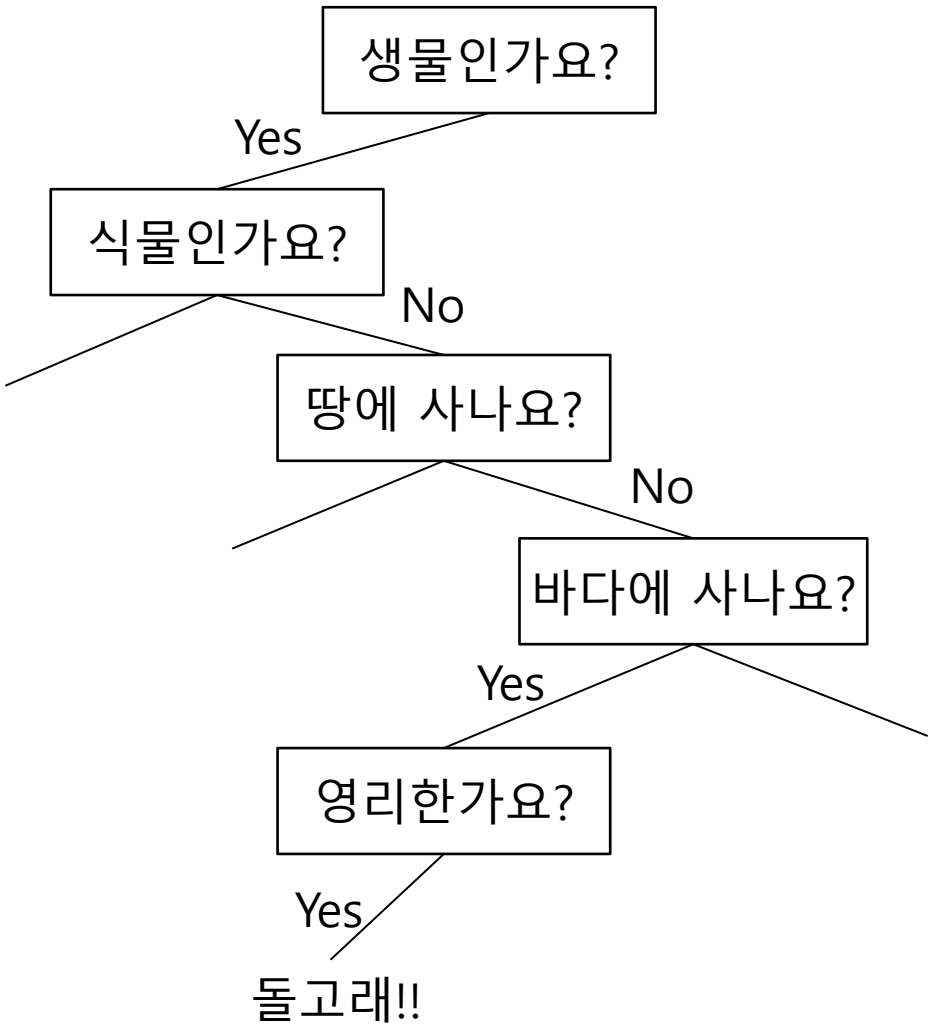
- 계량 데이터를 분류할 때 사용
- 데이터가 d차원 공간의 점들로 표시되고, 점들 간의 거리 개념을 바탕으로 분류
  - Bayesian classifier
  - $k-NN$
  - MLP
  - SVM

질적 분류기 (qualitative classifier)

- 비 계량 데이터를 분류할 때 사용
- 데이터를 특징 벡터 $\mathbb{x} = (x_1, \dots, x_d)$로 표현할 수 있다. 단, x_i는 비 계량의 값을 갖는다.
- 단순한 계산으로 데이터간의 거리 측정 불가능
  - Decision tree
  - String 인식기

Decision Tree

Think about 스무고개!



Decision tree의 원리

결정 트리는 계층구조를 가진 트리를 이용하여 데이터를 분류하는 분류기이다.

결정 트리는 스무고개와 유사한 원리를 사용한다.

- 스무 고개는 사람(진행자)이 그 때마다 문제를 만들어낸다.
- 결정 트리는 컴퓨터가 자동으로 질문을 생성해내야 한다.

Question

- 각 노드의 질문을 어떻게 만들 것인가?
- 노드에서 몇 개의 가지로 나눌 것인가? (자식노드의 개수)
- 언제 멈출 것인가? (leaf node)
- leaf node를 어느 부류로 할당할(assign) 것인가?

Decision tree의 구조

결정 트리의 동작 원리 및 구조

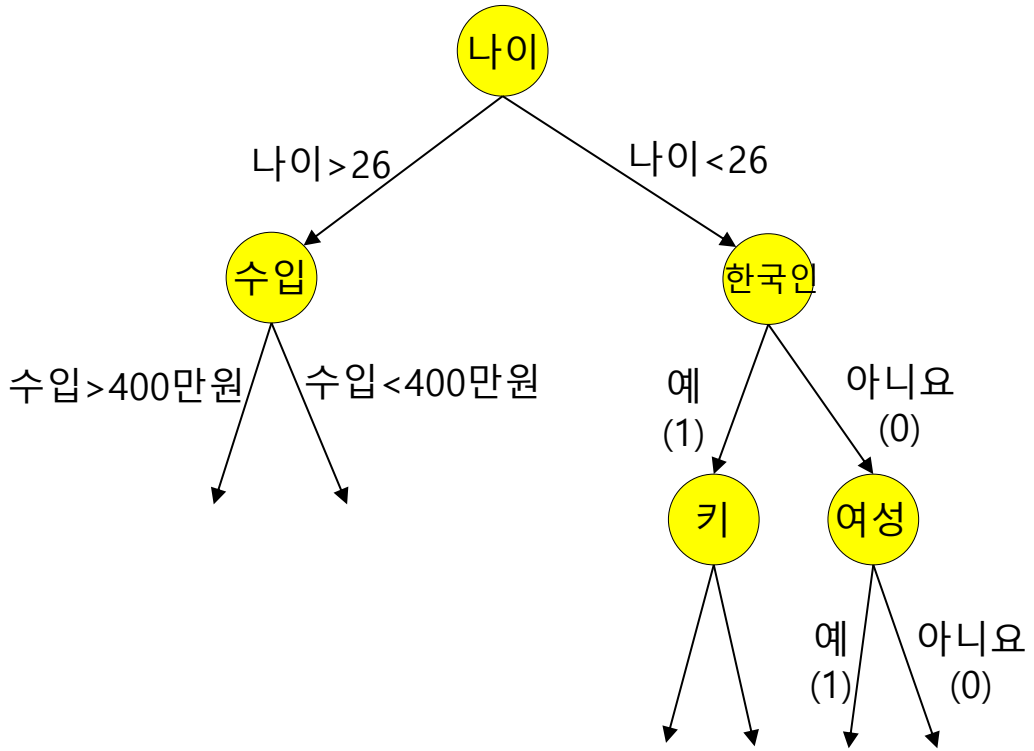
- $X \in \mathbb{R}^d$: 분류하고자 하는 비 계량 데이터의 집합
- 샘플 $\mathbb{x} = (x_1, \ldots, x_d) \in \mathrm{X}$: x_i는 비 계량 값
  - 샘플 $\mathbb{x}$가 주어지면 루트 노드에서 출발하여 질문에 대한 답을 구하고, 그 결과에 따라 자식 노드로 이동한다.
  - 이 과정을 leaf 노드에 도달할 때까지 순환적으로(recursively) 반복한다.
  - 잎 노드에 도달하면 $\mathbb{x}$를 그 leaf 노드에 해당하는 분류로 분류하고 끝낸다.

Decision tree의 구조

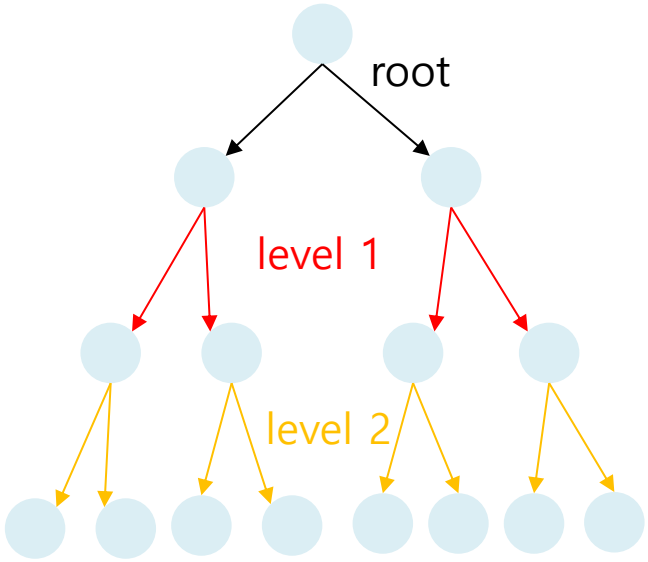
① internal node : attribute or feature
(속성 or 특징)

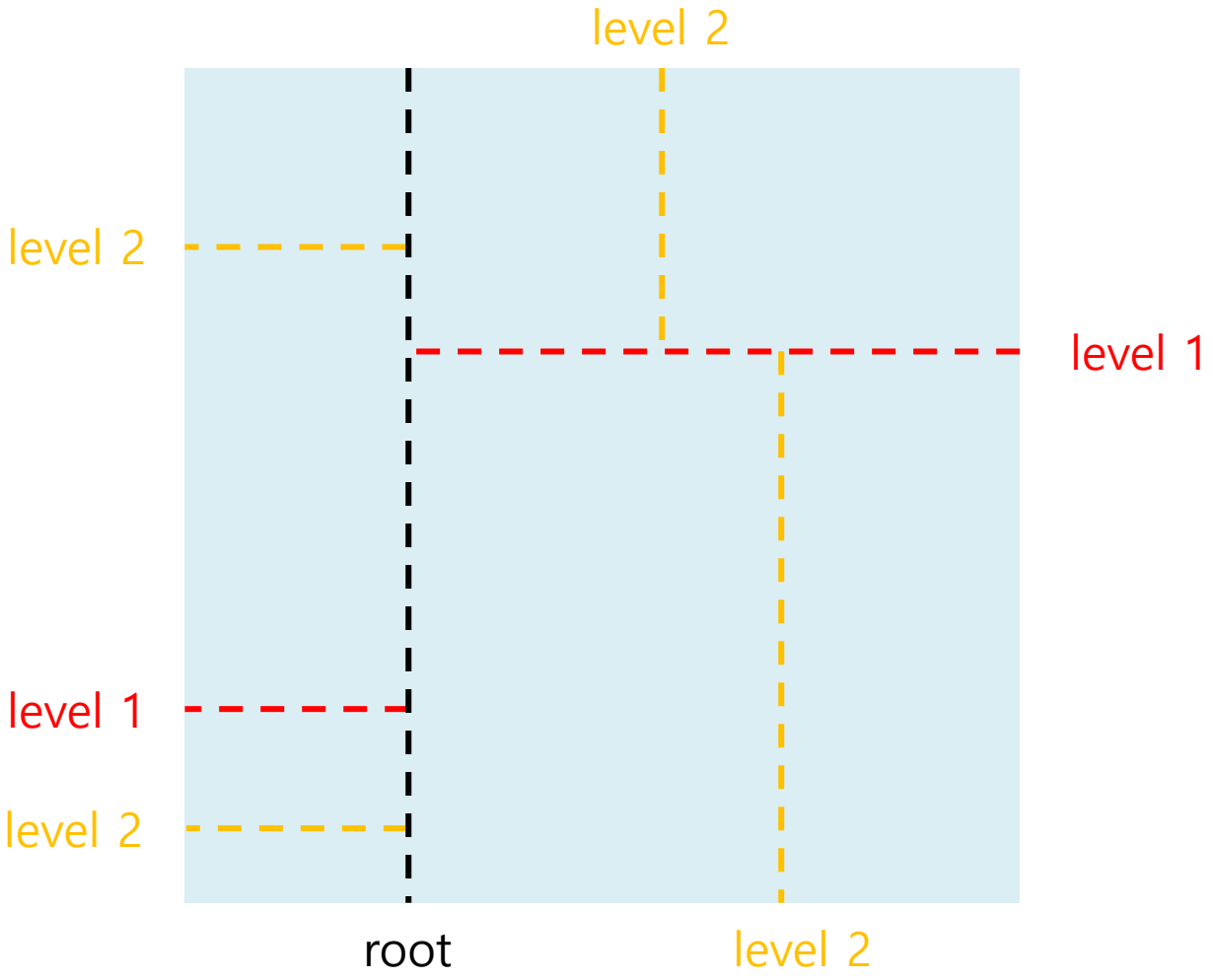
② leaf node : classification 결과

③ edge : assginment(or 조건)



Decision tree의 구조





트리의 분기는,
특징 공간 (feature space) 을
다수의 단순 사각 영역으로
나누는 것!

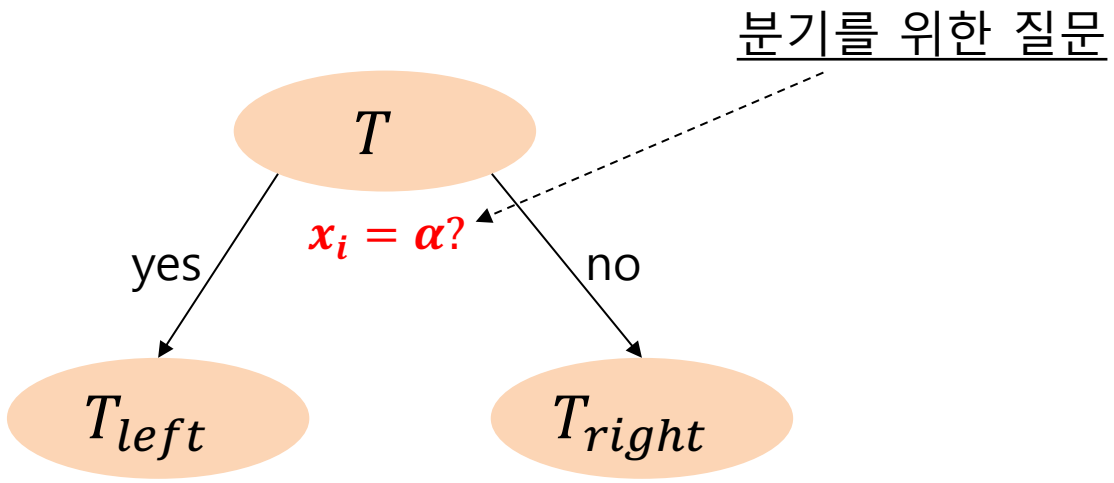
각 노드의 질문을 어떻게 만들 것인가?

학습에 사용할 샘플의 집합 : $X = \{\mathbb{x}_1, \dots, \mathbb{x}_n\}, \{(\mathbb{x}_1, y_1), \dots, (\mathbb{x}_n, y_n)\}$

- $\mathbb{x}_i = (x_1, \dots, x_d) \in \mathbb{R}^d$
  - x_i는 비 계량 값을 가지는 특징 or 계량 값을 가지는 특징
- y_i : sample $\mathbb{x}_i$가 속한 부류(class)
- $X_T = \{\mathbb{x}_i | \mathbb{x}_i \in T \text{ for } 1 \le i \le \text{n}\}$

※ 2개의 노드로 분기하는 상황으로 생각한다.

$$\begin{cases} X_{T_{left}} \cup X_{T_{right}} = X \\ X_{T_{left}} \cap X_{T_{right}} = \emptyset \end{cases}$$



각 노드의 질문을 어떻게 만들 것인가?

질문 $x_i = \alpha$?를 어떻게 만들 것인가?

- x_i : 특징 벡터 $\mathbb{x}$를 구성하는 i번째 특징
- α : x_i가 가질 수 있는 값 중의 하나

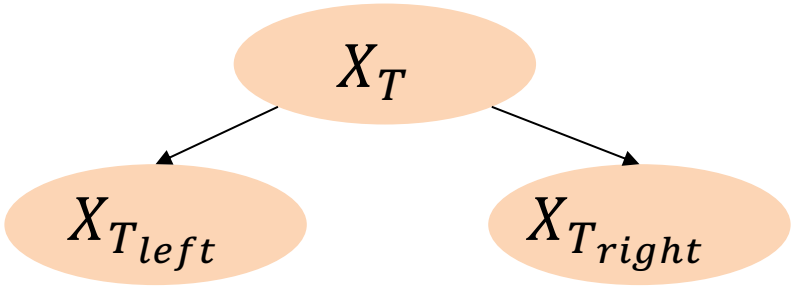
  ex) x_i가 혈액형이라면, α는 4개의 값을 가질 수 있다.

- $\mathbb{x} = (x_1, \ldots, x_d)$, x_i가 평균 m의 값을 갖는다면 dm개의 후보 질문이 생김
- 후보의 개수가 지수적이지 않기 때문에 모든 후보를 평가하여 가장 좋은 것을 선택(exhaustive search)

가장 좋은 것다는 것의 판단 기준은?
후보 질문을 평가하기 위한 기준 함수?

기준이 되는 함수

노드 T에 속하는 샘플 X_T에는 여러 부류에 속하는 샘플들이 혼재되어 있다.



분기를 반복하면 결국 leaf node에 도달하게 될 것이고, leaf node에는 같은 부류에 속하는 샘플들이 대부분이어야 한다.

따라서 X_T의 분기의 결과인 $X_{T_{left}}$와 $X_{T_{right}}$에는 각각 최대한 동질의 샘플이 담기는 것이 좋다. 동질의 샘플은 같은 부류에 속하는 샘플이라는 것이다.

동질성을 측정하는 기준 함수를 만들어, $X_{T_{left}}$와 $X_{T_{right}}$의 동질성을 가장 높여줄 수 있는 분기를 생성해 내는 질문을 선택한다. *동질성을 측정하는 기준 함수는 어떻게 만드는가?*

Impurity (불순도)

Impurity (불순도)

- 동질성을 측정하는 기준.
- 같은 부류에 속하는 샘플이 많을수록 불순도는 낮고, 다른 부류에 속하는 샘플이 많이 섞여 있으면 불순도는 높다.
  - 불순도가 낮으면, 특정 부류에 속한다고 예측하기가 쉽다.
  - 불순도가 높으면, 특정 부류에 속한다고 예측하기가 어렵다.
- 불순도를 정의하는 방법
  - Entropy
  - Gini impurity (지니 불순도)

Entropy (노드 분기)

노드 분기를 위한 entropy의 정의

- y_i : M개의 부류 중 i번째 부류
- 랜덤 변수 : 노드 T의 샘플이 특정 부류에 속하는 사건
- $P(y_i|T)$: 노드 T의 샘플이 y_i에 속하는 확률 (노드 T에서 y_i가 발생할 확률)

$$im(T) = -\sum_{i=1}^{M} P(y_i|T) \log_2 P(y_i|T)$$

- 엔트로피는 M개의 부류가 같은 확률을 가질 때 가장 큰 값을 가짐
  - 같은 확률을 가진다는 의미는, 한 노드에 다른 부류에 속한 샘플들이 같은 빈도로 나타나는 것을 의미한다.
  - 특정 샘플이 어떤 부류에 속하는지 예측하기 어렵다. ⇒ 불순도가 높다.

Information gain (정보 이득)

정보 이득은 표본들을 주어진 속성에 따라 분류함으로써 얻을 수 있는 엔트로피의 감소량의 기댓값이다.

트리 분류기의 기본적인 아이디어는 각 노드가 여러 부류가 혼합된 상태에서 시작해 각 노드의 관측값이 최대한 순수 부류만 남을 때까지 greedy모드로 반복적으로 분할을 수행해 나간다는 것이다.

각 단계마다 **최대 정보 이득값을 갖는 변수**가 선택된다.

Cart 알고리즘

$$\textit{정보 이득} = \textit{부모 노드의 엔트로피} - \textit{sum}(\textit{가중값 \%} \times \textit{자식 노드의 엔트로피})$$

$$\textit{가중값\%} = \frac{\text{특정 자식 노드에서의 관측값 수}}{\text{모든 자식 노드에서의 관측값 수의 합}}$$

노드 분기를 위한 Entropy 예제

다음 표는 과거 몇 일 동안 테니스를 쳤던 날과 테니스를 치지 않은 날의 데이터를 기록한 것이다. 테니스를 칠 것인지 말 것인지를 결정하는 데 있어 가장 중요한 역할을 하는 변수가 무엇인지 찾는 것이다.

Day	날씨	기온	습도	바람	테니스 유무
1	맑음	높음	높음	약함	N
2	맑음	높음	높음	강함	N
3	흐림	높음	높음	약함	Y
4	비	보통	높음	약함	Y
5	비	낮음	보통	약함	Y
6	비	낮음	보통	강함	N
7	흐림	낮음	보통	강함	Y

노드 분기를 위한 Entropy 예제

Day	날씨	기온	습도	바람	테니스 유무
8	맑음	보통	높음	약함	N
9	맑음	낮음	보통	약함	Y
10	비	보통	보통	약함	Y
11	맑음	보통	보통	강함	Y
12	흐림	보통	높음	강함	Y
13	흐림	높음	보통	약함	Y
14	비	보통	높음	강함	N

Root에서 분기할 특성을 결정하기 위해 바람 , 습도 변수를 선택한 경우 엔트로피를 구하여라. (날씨, 기온, 습도, 바람을 다 구해보고 그 중에서 정보 이득이 가장 높은 것을 선택해야 함.)

Gini impurity (노드 분기)

불순도를 정의하는 또 다른 방법이다. **잘못된 분류를 측정하는 도구**로, 다부류 분류기에 적용된다. (부류가 2개 이상인)

엔트로피와 거의 동일하지만 훨씬 더 빨리 계산할 수 있다.

Sklearn에서 제공하는 DecisionTree 라이브러리에서 default 값으로 사용되는 기준이다. M은 class$_{부류}$의 수이다.

$$im(T) = 1 - \sum_{i=1}^{M} P(y_i|T)^2 = \sum_{i \neq j} P(y_i|T)P(y_j|T)$$

노드 분기를 위한 Gini 예제

Root에서 분기할 특성을 결정하기 위해 바람 , 습도 변수를 선택한 경우 Gini 불순도를 구하여라. (날씨, 기온, 습도, 바람을 다 구해보고 그 중에서 지니의 기대값이 가장 낮은 것을 선택해야 함.)

Cart 알고리즘

Classification And Regression Tree 의 약자이며, Binary decision tree 를 만들 때 사용하는 알고리즘이다.

앞에서 다루었던, 동질성을 체크하는 기준 함수를 이용하여 parent node 에서 child node 로 split분기 한다.

기준 함수로는 entropy의 정보 이득 또는 gini impurity 의 기대값을 사용한다.

따라서, 기준 함수는

$$J = \frac{m_{\text{left}}}{m} \times \text{Impurity}_{\text{left}} + \frac{m_{\text{right}}}{m} \times \text{impurity}_{\text{right}}$$

이며, 이것을 기준으로 child node 로 분기한다.

당연히 CART 알고리즘은 탐욕적 알고리즘Greedy algorithm 이다. 맨 위 root 노트에서 최적의 분할을 찾으며 각 단계에서 이 과정을 반복한다. 현재 단계의 분할이 몇 단계를 거쳐 가장 낮은 불순도로 이어질 수 있을 지 없을 지를 고민하지 않는다.

Decision tree의 성질

- 결정 트리 분류기는 단순하게 문장으로 규칙을 만들 수 있기 때문에, 상위 관리자에게 설명이 용이하다.
- 결정 트리는 Non-parametric 모델로, 미리 가정된 parameter 같은 것은 존재하지 않는다. 내부적으로 변수 선택과 특징 선택을 수행한다. (Non-parametric models assume that the data distribution cannot be defined in terms of such a finite set of parameters. 즉 훈련되기 전에 파라미터 수가 결정되지 않는다.)
- 기저 분포에 관한 어떠한 가정도 하지 않는다.
- 모델의 모양이 미리 정해지지 않고 모델은 데이터에 관한 최적 분류로 학습된다.
- 모든 변수가 범주형 변수(categorical variable)일 때 가장 잘 작동한다.
- Parameter 사이의 비선형 관계가 트리의 성능에 영향을 주지 않고 수치 데이터도 잘 처리한다.
- Outlier 또는 missing value 를 잘 처리한다.

Question

한 노드의 지니 불순도는 보통 그 부모 노드 보다 항상 작을까? 일반적으로 작을까? 아니면 클까?

- 한 노드의 지니 불순도는 일반적으로 부모의 불순도보다 낮다. 이것은 자식의 지니 불순도의 가중치 합이 최소화되는 방향으로 각 노드를 분할하는 CART 훈련 알고리즘의 비용 함수 때문이다.
- 하지만, 항상 작은 것은 아니다. 어느 경우일까? (예를 들어 생각해보라)