ML Day1

Decision Tree

- 특정 기준(descriptive feature)에 따라 데이터를 구분하는 모델
- Discriptive feature : node에 들어가는 조건 또는 질문
- Classification : 예측하고자 하는 target값이 categorical(범주형) variable인 경우 (Binary classification-이산형, Multi-class classification-다산형)
- Regression : 예측하고자 하는 타겟값이 continuous(연속형) variable인 경우
- 확률이 높을수록 얻을 수 있는 정보량이 적다.
- 구성 : Root node(시작이 되는, 어떤 root node로 시작되느냐에 따라 depth가 달라짐), Intermidate node, Leaf node

Impurity 지표

1. **Entropy** : 무질서도 또는 impurity(불순도), heterogeneity(불균질성)의 측정 —>root node에 집어 넣을 discriptive feature를 더 나은 것으로 정하기 위해 entropy를 구함

   1) 디지털 정보 → degree of surprisal = 놀람의 정도

2) Decision tree의 분기는 impurity(불순도)가 작은 방향으로 진행된다.

3) 데이터 정보 분류에서 class가 2개인 경우 두 class의 갯수가 같으면 entropy값이 1이다.

⇒ entropy = 1 → Low Knowledge - 얻어지는 지식이 낮다고 표현

⇒ 다수를 기준으로 classification을 한다. entropy = 1에서는 다수인 class가 존재하지

않으므로(class 숫자가 같기 때문) 분류할 기준이 없어 classification이 힘듦.

4) Entropy는 항상 0~1 사이에서 존재하는 것이 아닌 확률변수 class의 수에 따라 달라지게 된다.

⇒ 분류하고자 하는 종류가 2개인 경우만 entropy가 0~1값을 가지고 class가 많아지면

entropy max값도 커진다.

5) 정보이론관점에서 Entropy가 높다 = 복잡도 상 // Entropy가 낮다 = 복잡도 하

Entropy = 0, 정보량이 적음 // Entropy = 1, 정보량이 많음

$$H = \sum(\text{사건 발생확률}) \cdot \log_2\left(\frac{1}{\text{사건 발생확률}}\right)$$

$$= \sum_i p_i \ \log_2\left(\frac{1}{p_i}\right)$$

$$= -\sum_i p_i \ \log_2(p_i)$$

6) Weighting(가중치) : 하위 생성된 node의 dataset 크기에 따라 가중치를 부여한다

→ 가중치를 주는 이유 - 데이터 수가 많을 수록 더 큰 가중치를 줘 noise의 비중을 줄인다.

⇒ 가중치를 주는 식

$$rem(d, \mathcal{D}) = \sum_{l \in levels(d)} \underbrace{\frac{|\mathcal{D}_{d=l}|}{|\mathcal{D}|}}_{\text{weighting}} \times \underbrace{H(t, \mathcal{D}_{d=l})}_{\text{entropy of partition } \mathcal{D}_{d=l}}$$

2. Information Gain : 얼마만큼 정보를 획득했나(정보획득량) → 분할 전과 후 Entropy의 차이

⇒ 분기 이전의 불순도(전체 dataset의 entropy)와 분기 후의 불순도(하위 dataset의 entropy)차이

⇒ Information Gain의 수치가 클수록 더 변별력이 좋다고 판단할 수 있다.

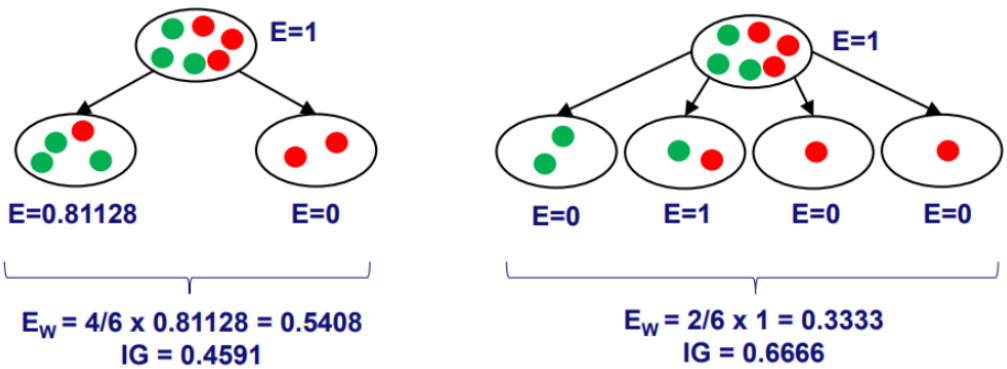
1) Original dataset의 전체 entropy값 계산

2) 각각의 descriptive feature로 나누었을 때의 각각의 subset들의 entropy를 계산하여 그 descriptive feature로 분기 했을 때의 total entropy계산(각각의 subset은 가중치를 곱한 값)

3) 1)의 값 - 2)의 값 = Information Gain

3. Information Gain Ratio : split(분기)을 더 잘게 많이 할수록 information gain값은 커질 수밖에 없다. 더 많이 split을 한 decision tree가 각 node의 impurity가 낮기 때문에 IG에서는 더 좋은 성능을 보이지만 아래 그림과 같이 왼쪽 그림보다 오른쪽 그림이 split이 더 잘게 일어났지만 반드시 split이 많이 일어나는 decision tree가 좋다고 보긴 힘들다.

⇒ IG의 한계점



⇒ IGR의 계산식

$$GR(d, \mathcal{D}) = \frac{IG(d, \mathcal{D})}{-\sum_{l \in levels(d)} (P(d = l) \times log_2(P(d = l)))}$$

4. Gini index(always 0~1)

1) diversity(다양성)을 판단하는 metrics = 한 쌍씩 뽑는다고 가정했을 때 서로 다른 class가 뽑힐 확률의 면

2) cart 알고리즘에서 지니계수(Gini Index)로 계산

3) 어떤 특정 dataset에 어떤 metric을 적용했을 때 더 좋은 decision tree를 완성할 수 있는지는 직접 시도해 보아야 한다.

4) Gini Index가 높을 수록 데이터가 분산되어 있음을 의미한다.

⇒ Gini index 식

$$Gini(t, \mathcal{D}) = 1 - \sum_{l \in levels(t)} P(t = l)^2$$