Decision Tree. 의사결정나무.

의사결정나무란 의사 결정 규칙을 결정하는 알고리즘을 말하며, 이 모양이 나무와 유사해 붙여진 이름이다. 이때 분기를 만드는 방식은 불확실성을 나타내는 entropy 또는 gini impurity 최소화하는 분류를 이용해 만든다. 이때 entropy 와 gini impurity 지수 각각 구하는 수식은 조금 다르지만 둘다 같은 교차해서 이용할 수 있으며 entropy 은 범위가 [0, 0.5] 이지만 gini impurity 은 범위가 [0, 1] 이다.

또한 의사결정나무를 만드는 데 크게 두 가지 방식이 있는데 그것은 각각 ID3(Iterative Dichotomiser 3) 와 CART(Classifying and Regression Tree) 이다. ID3 와 CART 둘다 entropy 또는 gini impurity 가 최소가 되게 나눈다. 하지만 CART 은 binary tree 를 만들지만 ID3 은 이진 말고도 여러 갈래로 나눌 수 있다는 차이가 있다.

이처럼 의사결정나무는 불확실성이나 무질서도를 최소한으로 만드는 알고리즘이기 때문에 classifying problem 에만 적용이 가능할 것 같은데 regression problem 에도 접목이 가능하다. 이때 numerical value 인 종속변수를 여러 기준점으로 나누거나, 중위수와 사분위수를 이용해 분류를 하는 등의 방법을 이용한다.

이러한 의사결정나무는 regression based algorithm 에 비해 여러 장점들을 가지고 있다. 우선 의사결정나무는 value scale 자체에 영향을 받지 않고 오로지 좋은 분류점을 찾으려고 하기 때문에 min-max scaling 이나 log scaling 의 과정이 필요가 없다. 또한 결정나무를 출력해보면 알 수 있듯이 결과를 도출하는 과정을 해석하고 이해하기가 매우 쉽다.

하지만 장점만 있는 것은 아니다. Decision Tree 의 가장 큰 단점 중 하나가 데이터가 특정 변수에 수직 또는 수평으로 잘 나뉘지가 않으면 성능이 크게 떨어진다. 또한 작은 값 차이로 인해 분류점이 완전히 달라질 수 있기 때문에 적은 개수의 노이즈에도 영향을 받을 수 있다는 점도 의사결정나무의 단점으로 꼽힌다.

이러한 단점들을 보완한 것이 Random Forest 알고리즘이다. Random Forest 알고리즘은 Ensemble Method 중 bagging 에 해당되는 알고리즘 중 하나로, 다수의 의사결정나무를 만들어 이들의 다수결에 따라 최종답을 내는 알고리즘이다. 이렇게 다수의 트리를 만들기 때문에 이상치에 영향을 받기가 어려우며, 또한 과적합도 피할 수 있다. 실제 산업에서도 단일 의사결정나무보다 random forest 알고리즘이 훨씬 많이 사용되며 아주 좋은 성능을 내고 있다. Random Forest 알고리즘의 단점으로는 수많은 의사결정나무들을 만들어내야 하기 때문에 연산과 시간적인 면에서 불리하다는 점을 꼽을 수 있다.