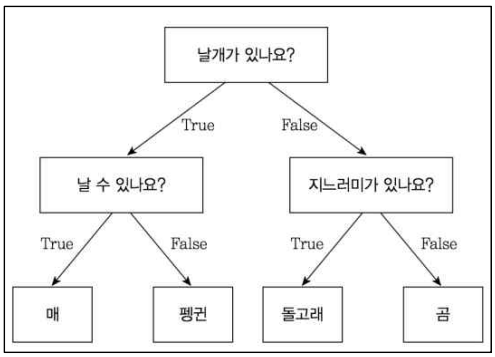
7주차 발제 - 지도학습 회귀

회귀 트리



- 이 그림은 분류 트리의 예시로, 각 노드에서 질문을 받고 다음 노드로 넘어가 최종 리프 노드에서 클래스를 결정하게 됨.
- 노드가 너무 많은 경우, 오퍼피팅 문제가 발생할 수도 있음.
- 지니 불순도를 최소화하거나 정보 이득을 최대화하는 분기법 중 하나를 선택하게 되는데,
  - 1. 지니 불순도: 클래스가 잘못 분류될 확률의 가중 평균임. 각 노드는 $1-\sum_{i=1}^{2} p_i^2$ 의 지니 불순도를 가지게 되고, 한 노드를 거치고 난 후의 지니 불순도는 $Gini\,impurity(k,t_k)=\frac{|B|}{|B|+|C|}G_B+\frac{|C|}{|C|+|B|}G_C$ 가 된다. B, C는 한 노드를 거친 이후의 자식 노드를 의미한다. tk(k는 피쳐-데이터 분리 규칙/질문-을 의미함 / 경계점)를 조절 해 지니 불순도의 최솟값을 찾아내야 한다.
  - 2. 정보 이득: 정보이론과 엔트로피를 기반하는 것으로, 부모 노드의 엔트로피에서 전체 자식 노드의 엔트로피를 뺀 값으로 정의함. 엔트로피는 $H(A)=-\sum_{i=1}^{2} p_i(y)\log p_i(y)$로 계산하고, 분할 이후의 전체 엔트로피는 분리된 자식 노드 B,C에서의 엔트로피의 가중합으로 나타나기에 $H(B,C)=\frac{|B|}{|B|+|C|}H(B)+\frac{|C|}{|C|+|B|}H(C)$ 결국 분할로 인한 정보 이득은 $IG(A,B,C)=H(A)-\frac{|B|}{|B|+|C|}H(B)-\frac{|C|}{|C|+|B|}H(C)$가 된다. 이 또한 경계점을 바꿔가며 최솟값을 찾아야 한다.

결정트리 회귀(회귀 트리)는 분류 트리와 비슷하지만, **분기에 따른 두 자식 노드에서의 평균 제공 오차, 평균 절대 오차 등 연속 변수에서의 흩어짐 정도를 계산하고, 그 가중합의 최솟값을 구하게 된다.**

- MSE(평균 제공 오차)는 $J_{MSE}=\frac{1}{n}\sum_{i=1}^{n}(y_i-\bar{y})^2$

- 푸아송 편차 절반 기준은 $J_{poisson} = \frac{1}{n}\sum_{i=1}^{n}(y_i \log\frac{y_i}{\bar{y}} - y_i + \bar{y})$이며(푸아송 편차 절반 기준은 목표 변수가 반드시 양수이며, 단위 시간 동안 발생한 사건의 수에 해당하는 카운트 변수인 경우 유용하다)

- MAE(평균 절대 오차)는 $J_{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\tilde{y} - y_i|$이 된다.

sklearn.tree.DecisionTreeRegressor 클래스에서 결정트리회귀모델 구현이 가능하고, DecisionTreeClassifier / DecisionTreeRegressor 클래스의 feature_importances_ 어트리뷰트로 피처 중요도를 구해, 피처 별로 학습 모델 구축에 영향을 미친 정도를 확인할 수 있다.

랜덤포레스트 모델

여러 개의 결정 트리를 학습하고 그 결과를 종합하는 앙상블 학습 모델이다. 회귀에서 사용하는 경우, 각 결정 트리의 예측 값의 평균을 예측 값으로 한다.

- 랜덤포레스트를 사용하면 결정 트리의 과적합 문제 해결이 가능하다. 기존 결정트리는 깊이가 깊어질수록 학습 데이터에만 적합한 패턴을 보이게 되지만, 랜덤포레스트에서는 학습 데이터를 다르게 샘플링하고 각각의 서브 데이터에서 깊이가 깊은 결정 트리를 학습하고 이 결과를 최종으로 종합하여 출력값을 계산하기에 해석력은 줄더라도 일반화 가능성이 높아진다.
- 앙상블 학습 기법을 사용하는데, 여러 개의 베이스 학습기를 준비 해 학습한 후, 학습 결과를 종합 해 최종 결과를 예측하는 기법이다. 일반화 가능성을 높이는데 기여하고, 배깅(bagging)과 부스팅(boosting)을 사용한다.
    - 배깅은 데이터셋에서 많은 부트스트랩(주어진 데이터셋을 모집단으로 간주하고 수 많은 시뮬레이션 샘플을 만들어내는 기법으로 복원추출 함) 샘플을 생성해 독립적인 다수의 학습기를 만들어 병렬로 학습하고 결과물을 집계하는 방식이다.
    - 그 과정에서 부스팅을 사용 해, 이전 모델의 학습 결과에 따라 오답에 대해서는 높은 가중치를 부여하고 정답에 대해서는 낮은 가중치를 부여 해 부여된 가중치가 다음 모델에 영향을 미치도록 한다. 잘못 분류된 데이터에 집중해서 새로운 분류 규칙을 만드는 과정을 반복한다.
- 하이퍼파라미터 튜닝을 통해 트리의 개수, 각 트리에서 사용되는 최대 깊이, 분할 시 고려할 특성의 수 등을 조정하여 모델의 성능을 향상 할 수 있다.