AI+X:머신러닝 2차과제

ICT융합학부2021093581 임동희

결정 나무 기법이란 의사 결정트리(Decision Tree) 머신러닝으로, 분류와 회귀가 모두 가능한 지도학습모델입니다. 결정나무 기법은 계속해서 “예” 혹은 “아니오” 질문을 거듭하여 학습힙니다. 즉, 특정 기준에 따라 데이터를 구분하는 모델입니다. 그런데, 만약 의사결정 나무의 깊이(max_depth)가 깊어지게 되면 더 많은 기준을 사용하여 정교하게 데이터를 분류할 수 있으나 사용중인 데이터에 대한 의존성이 너무 높아지게 되어 overfitting(과적합) 문제를 야기할 수 있습니다. 여기서 max_depth는 트리의 최대 깊이를 규정하는 변수이며, 과적합은 모델이 학습 데이터의 impurity를 높이기 위한 과정에서 학습데이터에만 지나치게 잘 맞게 학습하여 테스트 데이터에서 오류가 증가하는 문제입니다. 이 과적합문제는 머신러닝 알고리즘의 성능문제와도 이어지므로 이를 잘 해결 하는 것이 매우 중요합니다. 따라서 결정 나무 기법에서 최대 깊이 값을 너무 높은 값을 적용시키지 않도록 적절한 값을 할당해야합니다.

다음으로, random_state를 고정시키는 이유에 대해 이야기해보려고합니다. random_state 인자는 수행 시마다 동일한 결과를 얻기 위하여 고정하는 것입니다. 숫자를 임의로 생성할 때 재현가능하도록 난수의 초기값을 입력(난수 생성기) 지정하는 정수값에 따라 분석결과가 다르게 나옴. 생략할 경우, 실행할 때마다 다른결과가 나오기 때문에(test/train 데이터가 계속달라짐) 정확한 분석결과를 얻거나 성능 분석을 하기 어려워집니다. 예를 들어 가위바위보 승률 예측 그래프를 그리는 머신러닝 알고리즘을 만들었다고 했을 때 랜덤 값이 계속해서 바뀐다면 정말 운이 안좋은 경우 모든 실행 결과가 같은 경우가 발생할 수 도 있습니다. 따라서 random_state를 고정시켜 여러번 수행하더라도 같은 레코드 값을 추출할 수 있도록, 즉 재현 가능 하도록 고정되는 숫자를 지정하는 것입니다.

feature가 여러 개인 상황에서 heatmap을 이용하면 특징간의 상관관계를 연결지어 데이터를 보다 효율적으로 분류하는데 도움을 주게 됩니다. Feature가 여러 개인 예씨로, 수업중 판교역 지하철탑승객의 예시를 들 수 있습니다. 즉, Feature가 여러 개인 상황에서는 heatmap을 통해서 데이터 간 상관관계를 알 수 있습니다. 앞서 말했던 예시에서의 내리는 역과 하고있는 행동간의 상관관계, 내리는 역과 탑승객의 복장특징의 관계 와 같이 다른 특징들 간 상관관계를 통해 DataSet을 보다 더욱 효과적이고 쉽게 분류 가능합니다.

다음으로, train과 test로 데이터 분리를 하는 이유에 대해 말해보고자 합니다. 이렇게 데이터를 분리하는 이유는 과적합을 피하기 위함입니다. 머신러닝 모델에 train 데이터를 100% 학습시킨 후 test 데이터에 모델을 적용하면 성능이 생각보다 나오지않습니다. 앞서 말한 과적합 문제 때문입니다. 따라서 overfitting을 방지하는 것은 전체적인 모델 성능 향상시키기 때문에 데이터분리는 매우 중요한 프로세스 중 하나입니다.

다음으로 특성간 격차를 줄이는 방법 두가지에 대해 이야기하고자 합니다. 특성간 격차가 너무 크면 변수로 이용하기 어려워 지기 때문에 격차를 줄이는 과정이 필요합니다.첫번째로는 MINMAXSCALER(MMS) : 최솟 값 0, 최댓값 1이 되도록 스케일링 하여 특성간 격차를 줄입니다. 두번 째로는 STANDARDSCALER(SS): 평균과 표준편차가 같아지도록 스케일링하는 방법으로 특성간 격차를 줄입니다.

마지막으로 회귀 기법을 평가하기 위해 어떤 방법을 사용해야하는지에 대해 이야기하고자합니다. 보스톤 집값예측의 경우, ‘주택 가격’ 즉 연속된 값을 예측 하는 알고리즘입니다. 따라서, 지도학습의 한 종류인 회귀모델에 속합니다. 회귀 모델의 평가는 실제값과 에측 값의 오차를 기준으로 합니다. 따라서 예측 집값과 실제 집값의 차를 구하면 됩니다.

예시로 제시한 주택가격의 경우, 머신러닝 회귀 알고리즘을 통해 그린 예측 그래프와 실제값이 다음과 같습니다. 이 경우, 빨간선이 예측값이므로 예측 값에서 실제 값(파란색 점)을 뺀 값들의 평균이 이 머신러닝 알고리즘의 성능을 평가하는 방법이 됩니다.