2주차 예비보고서

 전공: 컴퓨터공학과 학년: 2학년 학번: 20221519 이름: 구현모

1.

Feature engineering은 머신러닝 알고리즘이 작동할 수 있도록 하는 feature(데이터 분석에 사용되는 독립 변수)를 만드는 과정을 의미한다. 기존 데이터를 수정하여 더 나은 변수(feature)를 만드는 기법이다. 이때 그 분야에 대한 사전 지식을 바탕으로 feature를 만든다. 스팸 이메일 필터의 경우 '이메일 내 특정 단어의 수' 또는 '이메일이 전송된 시간', ‘IP 주소’ 등이 feature가 될 수 있다. 이러한 feature를 얼마나 잘 만드느냐에 따라서 같은 데이터를 바탕으로 만들었어도 머신러닝의 성능이 달라질 수 있다.

Feature engineering의 기법에는 정규, 표준화, one-hot encoding, Feature Interaction 등이 있다.

먼저, 정규, 표준화는 데이터의 크기를 조절하는 방법이다. 그중 정규화는 데이터의 전체 범위를 변경한다. 데이터의 범위를 [0, 1]로 정규화한다면 [10, 20, 30] -> [0, 0.5, 1]로 바뀌게 된다. 이 방법은 데이터들 간의 비율이 유지된다는 특징이 있다. 따라서 분포가 바뀌지 않아 거리기반 알고리즘을 사용할 때 유용하다. 표준화는 데이터를 평균 0, 표준편차 1인 표준 정규 분포로 변환하는 방법이다. 이 기법을 사용하면 정규분포에 가까워지기 때문에 데이터가 정규분포를 따를 것으로 예상될 경우 사용하면 좋다. 선형 회귀와 같이 분포 가정이 있는 모델에서 사용하기 좋다. 이상치에 상대적으로 민감하지 않다는 특징이 있다. 정규화는 특성 값의 범위를 통일해야 할 때, 표준화는 데이터의 통계적 성질(평균, 분산)을 활용하거나 이상치가 많을 때 사용하면 적절하다.

One-hot encoding은 범주형 데이터를 수치형 데이터로 변환하는 기법이다. 특히 순서가 없이 단순 분류된 자료에 적용할 때 유용하다. 각각의 범주마다 하나의 column을 만들어 해당 범주에 속하면 1, 아니면 0을 채워서 변환한다. 범주간의 순서나 거리 개념이 없어 범주간의 거리 왜곡을 방지할 수 있다. 선형 회귀, 분류 등 다양한 모델에 사용할 수 있다는 장점이 있다. 단점으로는 다양한 범주가 있다면 열의 개수가 그만큼 늘어나기 때문에 메모리와 계산 비용이 커지게 된다는 것이 있다. 예시는 성별이라는 범주형 데이터를 (남자, 여자)로 할 때 [1, 0], [0, 1]이라는 수치형 데이터로 변환하는 것이다.

Feature Interaction은 기존 특성 간의 관계를 반영하여 새로운 특성을 만드는 기법이다. 각각으로 있을 때보다 합쳐졌을 때 더욱 가치 있는 정보가 될 때 사용할 수 있다. 위도와 경도라는 데이터가 있다면 지역라는 데이터로 바꿀 수 있다. Feature Interaction을 진행하면 비선형적 패턴을 보다 명확하게 나타낼 수 있다. 또한 특성 간의 관계가 중요한 경우 이를 직접적으로 모델에 학습시킬 수 있기 때문에 모델의 성능을 향상시킬 수 있다. 또한 Feature Interaction을 통하여 복잡한 모델을 사용하지 않고 비교적 간단한 모델로 상호작용 효과를 학습시킬 수 있다는 장점이 있다. 전문적 지식을 데이터에 반영시킬 수 있다는 점도 장점이다. 단점으로는 어떤 특성을 어떻게 결합하는 지에 때라 결과가 달라짐으로 이를 결정하기 위해 많은 사전지식이나 시도가 있어야 한다는 점이다. 또한 불필요한 특성 간의 관계를 반영한다면 모델이 잘못된 학습을 하게 되어 과적합될 수 있다는 점도 단점으로 꼽을 수 있다. 특성 간 관계가 예측 변수에 영향을 미칠 가능성이 클 때나 선형 모델을 사용하는데 데이터가 비선형 관계를 가질 때 사용하면 좋은 기법이다.

2.

결측치가 데이터에 포함되어 있다면 여러 문제가 생길 수 있다. 결측치를 무시하거나 단순히 제거해버린다면 모델이 학습에 필요한 유용한 정보가 사라져 다른 데이터들이 과대평가되거나 과소평가 될 수 있다. 이것을 데이터의 양이 적을수록 큰 영향을 미칠 수 있다. 또한 결측치가 많이 있다면 모델이 데이터를 제대로 학습하지 못하게 되어 성능이 떨어질 수 있다. 선형 회귀 모델의 경우는 작동이 안되기도 한다는 문제가 있다. 결측치가 일부 특성에만 집중되있는 경우에는 일반화를 실패하고 모델의 정확성이 떨어질 수 있다.

결측치를 처리하는 방법에는 여러가지가 있다.

먼저 결측치가 있는 특성을 제거해버리는 것이다. 장점은 빠르고 간단하다는 것이다. 하지만 데이터 손실이 발생하고 전체 데이터의 크기가 작거나 결측치의 비율이 높을 때는 부적합하다.

결측치를 다른 데이터들을 바탕으로 대체하는 방법도 있다. 먼저 평균, 중앙값, 최빈값으로 대체하는 방법은 간단하고 데이터의 분포를 크게 변화시키지 않는 장점이 있지만 데이터의 변동성을 낮게 평가할 수 있다는 단점이 있다. K-최근접 이웃(KNN) 대체라는 방식은 결측치와 유사한 다른 데이터를 결측치에 채우는 방법으로 보다 정교하게 대체할 수 있다. 하지만 계산비용이 높다는 단점이 존재한다.

결측치를 변수로 사용하는 방법도 있다. 결측 여부를 나타내는 새로운 더미 변수(0 또는 1)를 만들고 결측치를 0이나 특정 값으로 채워서 사용하는 것이다. 결측치 자체가 의미를 가질 때 사용하기 적합한 방법이다. 설문조사나 여론 조사 등에서 응답 거부한 경우와 같은 상황에서 결측치 자체를 변수로 사용할 수 있는 것이다. 장점으로는 결측치의 패턴을 분석하여 모델에 반영할 수 있다는 것이 있다. 단점으로는 처리해야 할 사항이 많아지므로 모델이 복잡해질 수 있다는 것이 있다.

각각의 상황과 모델에 맞는 결측치 처리 방법을 이용하는 것이 모델의 성능을 더욱 좋게 만들 수 있다. 결측치가 적고 무작위로 위치한 경우에는 결측치 제거를, 데이터 손실을 줄이고 연속성을 유지하고자 한다면 결측치 대체를, 결측치가 특정의미를 가진다면 결측치를 변수로 활용하는 방법을 사용하는 것이 좋다.

3.

Random Forest는 의사결정나무(Decision Tree)를 기반으로 하는 앙상블 학습 알고리즘이다. 앙상블 학습이란 여러 개의 개별 모델을 만들고 각각의 각각 모델의의 예측을 종합해서 최적의 결과를 만들어 내는 모델로 일반화하는 학습방식이다.

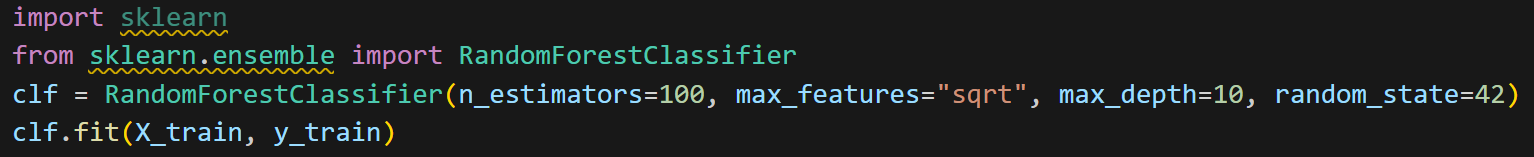
Random Forest의 특징은 앙상블 학습, 무작위성, 병렬처리 등이 있다.

여러 개의 의사결정나무를 생성하고 각각의 의사결정나무의 예측을 종합하여 최종 결과를 도출하는 의사결정나무 기반의 앙상블 학습을 한다. 이는 과적합 되기 쉬운 단일 의사결정나무 알고리즘과 다르게 여러 의사결정나무를 사용하여 결과를 평균화하거나 많은 모델이 예측한 값을 예측값으로 나오도록 최종모델을 만드는 투표 등의 방법으로 과적합을 방지할 수 있다.

학습에 사용하는 데이터에 대해서 각각의 의사결정 나무는 서로 다른 데이터 집합을 가지고 학습한다. 또한 의사결정 나무의 각 노드에서 분할 시 모든 특성을 고려하는 것이 아니라 랜덤하게 선택한 특성을 고려하여 분할한다. 이 특성은 모델의 다양성을 높이고 과적합을 줄이는데 도움이 된다.

Random Forest는 병렬 처리가 가능하다. 학습에 사용되는 각각의 의사결정나무는 독립적으로 학습하기 때문에 병렬 연산이 가능하다. 그렇기 때문에 모델의 학습 속도를 높이는데 도움이 된다.

scikit-learn에서 Random Forest를 사용하는 방법은 다음과 같다. 먼저 분류 알고리즘을 사용할 것인지 회귀 알고리즘을 사용할 것인지 정한 후 각각 RandomForestClassifier와 RandomForestRegressor 클래스를 import해준다. 그후 하이퍼파리미터들을 정해준 뒤 학습시키는 것이다. 하이퍼파라미터에는, n\_estimators, max\_depth, min\_samples\_split, max\_features이 있다. n\_estimators는 생성할 의사결정나무 개수를 정하는 파라미터로 100을 기본값으로 가진다. max\_depth는 각 나무의 최대 깊이로 따로 지정하지 않으면 최대로 성장한다. min\_samples\_split는 노드를 분할하기 위한 최소 샘플 수를 지정하는 파라미터로 기본값은 2이다. max\_features는 각 분할에서 고려할 최대 특성 수를 정하는 파라미터이고 기본값은 전체 특성의 제곱근만큼이다.



다음 그림은 Random Forest에서 RandomForestClassifier를 사용한 예시 코드이다. 이 예시 코드에서는 max\_depth를 10으로 정하고 결과의 일관성을 위하여 random\_state를 42로 고정하였다. 이외의 하이퍼파라미터들은 기본값과 같다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

이후 다음 코드와 같이 accuracy\_score과 predict로 정확도를 평가할 수 있다.