[Deep Learning] Homework 1

김수진

1. 데이터 조작 과정및 결과

1) 기본 전처리

-Time과 Amount를 스케일링한 후, 컬럼 전체를 사용해서 일차적으로 모델 구성

2) 차원 축소

-RFE 사용하여 모델을 만들 때 사용할 feature수 선택

-차원 축소 과정에서 n_features_to_select를 변화시켜가며 가장 예측력이 높은 변수들과 그 수를 확인

2. 사용한 모델

모델 학습

:train.csv 데이터를 통해 모델을 학습.

-Logistic Regression : 입력 값에 대한 선형 결합을 통해 0과 1의 데이터를 두 개의 범주로 나누어 결과로 출력하는 분류 기법으로 사용

-Decision Tree: 입력 값에 존재하는 패턴을 feature들의 조합을 통해 나타냄으로써 데이터를 분류하는 이진 분류 기법으로 사용

-Random Forest : 입력 값에 대해 여러 개의 decision tree를 이용해 데이터를 분류하는 기법으로 사용

3. 인자 탐색 과정 및 결과

1)모델 선택

: validation.csv 셋을 input으로 하여 만들어진 모델들의 f1 score을 확인함.

-Logistic Regression: RFE에서 선택하는 feature 수를 조정한 결과, 5개의 변수로 예측했을 때 최대 약 0.42의 f1 score를 보임

-Decision Tree: RFE에서 선택하는 feature 수를 조정한 결과, 8개 변수로 예측했을 때 최대 약 0.83 정도의 f1 score를 보임

-Random Forest: class_weight만 'balanced'로 두고 다른 파라미터를 조정하지 않은 상태로 원 데이터를 그대로 학습하고 예측했을 때 약 0.84정도의 f1_score를 보임.

→ 세 가지 모델의 성능을 f1 score로 평가한 결과, 가장 높은 점수를 보인 Random forest를 예측에 사용할 모델로 선택함

2) Random Forest에 대한 인자 탐색

:선택한 random forest모델에 대해 몇 가지 parameter를 변화시켜가며 예측 성능을 높이고자 함.

이 때 Random forest는 random_state를 고정하지 않은 상태에서 파라미터를 변화시킬 때마다 같은 파라미터에 대해 10번 반복적으로 예측을 수행하고 그 결과로 나온 f1 score의 평균값을 비교하여 가장 높은 점수를 보이는 파라미터를 선택하였다.

-class_weight='balanced'. Class가 매우 불균형하므로 class_weight 파라미터로 이를 고려

-max_features : (max_featurees=25) 1~29 range에 대해 탐색 후 최적값 선택

-max_depth=9 : auto, 1~30 range에 대해 탐색 후 최적값 선택

-n_estimators=20: 10, 20, 30, 50, 100, 120의 range에 대해 탐색 후 최적값 선택

3)결과

: RandomForestClassifier(class_weight='balanced', max_features=25, max_depth=9, n_estimators=20)

:약 0.86 정도의 f1_score점수가 나오는 모델로 구성.