**HW2\_34\_A #2 (4조)**

**1. 데이터 요약**

- 데이터 형태: 284,807 row와 31개의 column으로 이루어진 테이블

- Column은 시간 정보를 포함하고 있으며, 28개의 이름을 알 수 없는 column과 amount column, 그리고 class column으로 구성

- 테이블에는 null value를 가지고 있지 않음

- Class가 1인 경우는 0.17% 수준으로 매우 적은 비율을 가지고 있는 불균형 데이터

- 우리가 예측하고자 하는 것은 Time과 Amount를 포함한 모든 column 정보를 이용하여 Class를 분류하는 것임

**2. 데이터 전처리**

- 위 데이터는 Class 1인 경우가 전체의 0.17% 수준으로 매우 불균형함

- 현재 데이터로 랜덤하게 추출하여 training 후 예측할 때에서 여전히 데이터 분포는 불균형한 상태이기 때문에 예측 결과에 왜곡 발생 가능성 있음

- 따라서 test data는 class 0과 class 1의 개수를 각각 100씩 동일하게 구성하여 머신러닝 알고리즘에 따른 성능을 평가함

- 총 6개의 머신러닝 알고리즘 적용함

(Logistic regression, support vector regression, k-nearest neighborhood, neural network, decision tree, Bayesian classification)

**3. 알고리즘별 분석 결과**

- X = [‘Time’ : ‘Amount’], y = [‘Class’]

(1) Logistic regression

- sklearn.linear\_model module의 LogisticRegression() 이용하여 default 조건에서 분석

- 정확도: 0.72

(2) support vector classification

- sklearn.svm module의 SVC() 이용하여 gamma=’scale’ 및 나머지는 default 조건에서 분석

- 정확도: 0.5

(3) K-nearest neighborhood

- sklearn.neighbors module의 KNeighborsClassifier() 이용하여 default 조건에서 분석

- 정확도: 0.51

(4) Neural network

- sklearn.neural\_network module의 MLPClassifier() 이용하여 learning\_rate\_init=0.01 조건에서 분석

- 정확도 0.5

(5) Decision tree

- sklearn.tree module의 DecisionTreeClassifier() 이용하여 분석

- 정확도: 0.85

(6) Bayesian classification

- sklearn.naive\_bayes module의 BernoulliNB() 이용하여 분석

- 정확도: 0.77

**4. 불필요한 컬럼 제거**

- 시각화 과제에서 분석한 결과를 바탕으로 class에 영향을 주지 않는 몇몇 컬럼을 제거하고 분석시 결과에 어떤 영향을 주는지 분석함

- 예를들어 V20 column의 경우 class와 상관없이 유사한 histogram을 보였음

- Logistic regression의 경우 모든 컬럼을 사용한 경우 정확도 0.72, V20을 제거 후 분석시에도 0.735로 다소 향상된 정확도를 보였음

- 다른 알고리즘도 V20 제거 전후 유사하거나 다소 개선된 결과를 보였음

**5. 정리**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 알고리즘 | Logistic | SVC | KNN | NN | DT | Bayesian |
| 정확도 | 0.72 | 0.5 | 0.51 | 0.5 | 0.85 | 0.77 |
| V20제거 | 0.735 | 0.5 | 0.51 | 0.5 | 0.86 | 0.77 |

* 총 6가지의 머신러닝 알고리즘을 30의 변수로부터 class를 0과 1로 분류하는 모델을 평가함
* SVC, KNN, NN의 경우 50% 수준의 확률로 예측
* Logistic regression, decision tree, Bayesian classification 등의 알고리즘은 상대적으로 높은 정확도를 보임
* 특히 decision tree의 경우 0.926의 높은 정확도를 보임
* 하지만 이런 결과들은 training set에 따라, 알고리즘별 parameter들에 따라 성능이 달라질 수 있음
* 또한 class에 영향을 주지 않는 불필요한 컬럼을 제거한 경우에 예측 정확도에 영향을 주지 않거나 다소 개선된 결과를 보였음
* 랜덤하게 샘플링을 하여 계산한 결과라 다른 샘플을 통해 분석하면 성능은 바뀔 수가 있어 충분한 반복이 필요함