

**13주차 과제**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



|  |  |
| --- | --- |
| 학번 | 2020136087 |
| 이름 | 윤아현 |
| 수강 과목 | 머신러닝및실습 |
| 담당 교수 | 이해윤 교수님 |
| 제출일 | 2024.06.02 |

**머신러닝 및 실습 과제 4**

**1번. 데이터 설명**

데이터셋은, 레드 와인과 화이트 와인의 화학적 성분을 기반으로 한 와인의 품질이다.

독립 변수: 와인의 화학적 성분

종속 변수: 와인의 품질 점수 (0~10)

데이터는 11개의 독립 변수와 1개의 종속 변수로 구성되어 있다.

이 때, 종속 변수인 와인의 품질 점수는 3 ~ 9점 사이에 있다.

전체 데이터 개수는 6497개이며, 품질 점수의 개수는 아래의 표와 같다.

|  |  |
| --- | --- |
| **quality** | **values** |
| 3 | 30 |
| 4 | 216 |
| 5 | 2,138 |
| 6 | 2,836 |
| 7 | 1,079 |
| 8 | 193 |
| 9 | 5 |

**2번. 데이터 전처리 과정**

**1) 기본 통계량 확인하기**

종속 변수(와인의 품질)에 대한 기본 통계량을 확인하였을 시에, 평균이 5.8이고 표준편차가 0.8이라는 것을 확인할 수 있다.

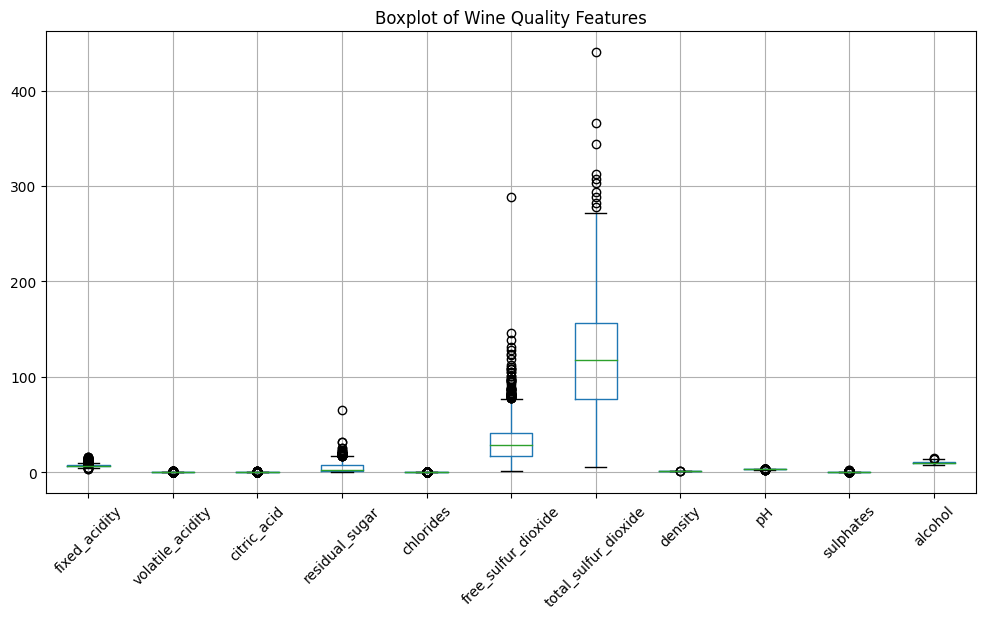
이는 데이터 값들이 평균 주변에 근접하게 분포하고 있음을 보여준다.

**2) 결측치 확인하기**

독립 변수 및 종속 변수 모두 결측치가 존재하지 않았다.

**3) 이상치 확인하기**

이상치를 쉽게 확인하고, 비교하기 위해 독립 변수들에 대한 Box-Plot을 그려 확인하였다.



대부분의 독립 변수에서 이상치로 판단될 값들이 보이지 않았지만,

독립변수인 residual\_sugar, free\_sulfur\_dioxide, 및 total\_sulfur\_dioxide에서는 튀는 값이 조금씩 존재한다는 것을 확인하였다.

위 3가지 변수에 대해 이상치 제거를 수행하였다.

1. Residual\_sugar

값이 50 이상인 데이터를 삭제하였다.

1. free\_sulfur\_dioxide

값이 200 이상인 데이터를 삭제하였다.

1. total\_sulfur\_dioxide

값이 300 이상인 데이터를 삭제하였다.

전체 6,497개의 데이터에서 총 7개의 데이터가 삭제되었다.

**4) 상관관계 확인하기**

독립 변수와 종속 변수 사이에 강한 상관관계를 나타내는 변수는 없었으며, 가장 큰 상관관계를 나타낸 독립 변수는 alcohol이었다. (0.446)

**5) 데이터 불균형 해소**

위의 데이터 설명 표에 나타나 있듯이, 각 데이터 간의 불균형이 심한 것을 확인 할 수 있다.

처음에는 오버샘플링 기법을 모든 class에 대해 수행하려고 했지만, class 9같은 경우에는 데이터가 10개 미만이기에, 사용이 불가능하였고 과적합이 발생할 우려가 있었다.

먼저, 클래스를 3개로 재조정을 수행하였다.

|  |  |
| --- | --- |
| **quality** | **values** |
| 0 (3, 4) | 243 |
| 1 (5, 6, 7) | 6,049 |
| 2 (8, 9) | 198 |

범주를 조정하여도 각 클래스 간의 불균형은 해소되지 않았다.

이를 해결하기 위해, 데이터 정규화를 수행한 뒤 훈련 데이터에 대해 오버샘플링을 수행한다.

**6) 데이터 정규화**

훈련 데이터와 테스트 데이터로 분할한 뒤, 정규화를 수행하였다. (test\_size=0.2)

모든 독립 변수의 단위를 동일하게 맞추기 위해 “StandardScaler(표준정규화)”를 적용하였다.

추가적으로, 클래스 간의 불균형을 해결하기 위해 class 1은 UnderSampling을, class 0과 2는 OverSampling을 수행한다.

**이 경우의 문제점은,**

class 0과 2에 대한 class를 1000개 이상으로 늘리게 되면, 테스트 데이터에서 전체적인 정확도는 높을지라도 class 0과 2로 분류하는 성능이 현저히 낮음을 알 수 있다.

이는 합성된 데이터가 데이터의 패턴을 제대로 인식하지 못해 생긴 문제로, **일반화 능력이 아주 낮다고** 해석할 수 있다.

그러므로 전체적인 정확도가 높지 않더라도, class 1에 대한 데이터 수를 줄여 일반화 능력을 키우도록 한다.

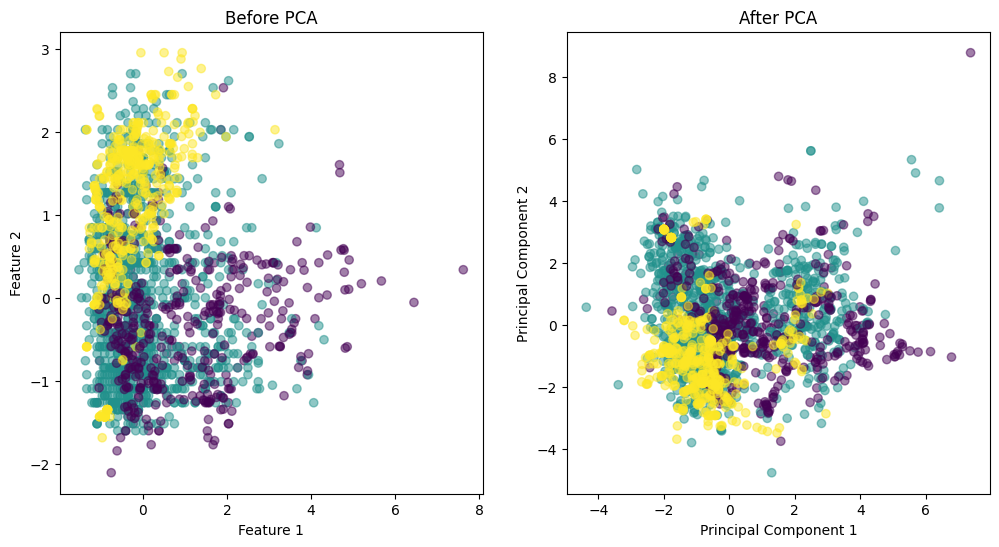
**7) Label Encoder**

종속 변수에 대해 Encoding을 수행하여 범주형 데이터를 수치형 데이터로 변환한다.

**3번. 차원 축소 과정 및 결과 분석**

차원 축소의 경우, 4차원 이상은 시각화를 수행할 수 없기 때문에 2차원과 3차원으로 대폭 축소를 해본 뒤 시각화를 수행한다.

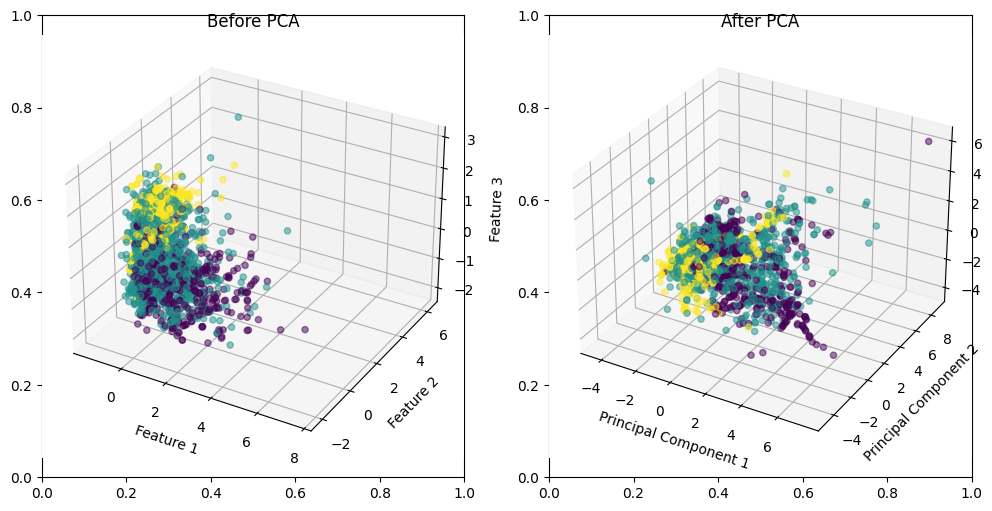
**1) 2차원 PCA 수행**



PCA 전 그래프는 종속 변수와 가장 상관관계가 높은 alcohol과 volatile\_acidity를 기준으로 시각화하였다.

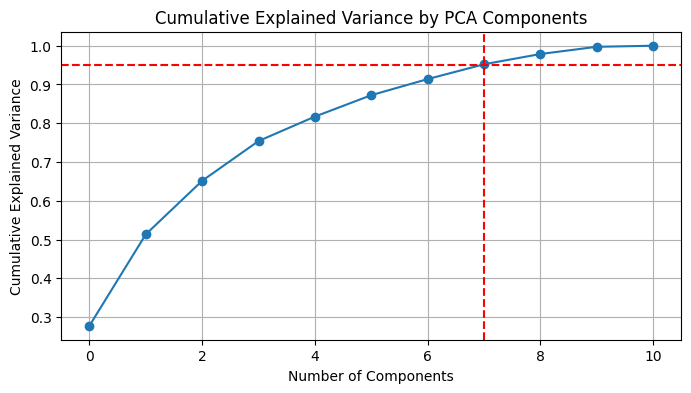
PCA 전과 후의 차이를 보면, PCA 후에 데이터 분포가 비교적 명확하게 보이는 것을 확인할 수 있다.

**2) 3차원 PCA 수행**



2차원으로 PCA를 수행했을 때와 동일하게 PCA 후에 데이터 분포가 비교적 명확하게 보이는 것을 확인할 수 있다.

**3) 축소할 차원 지정하기**



PCA 결과에 따르면, 7차원으로 줄였을 때 데이터의 95%까지 설명이 가능하다는 것을 확인할 수 있다.

이를 통해, 전체 독립 변수를 사용했을 때와 7개의 주성분을 사용했을 때 모델 성능을 비교해본다.

**4번. 분류 모델 구축 및 성능 평가**

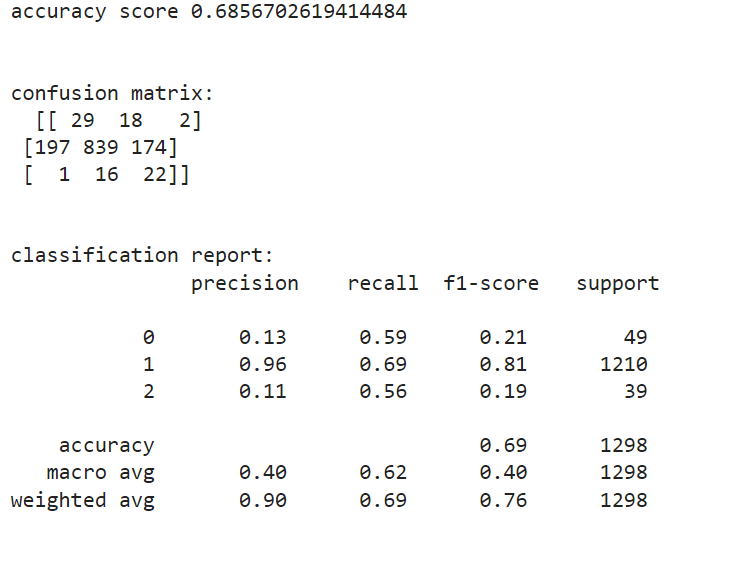
PCA를 수행하기 전과, 수행한 후의 성능을 비교해본다.

대표적인 성능 지표로 “accuracy\_score”를 사용한다.

|  |  |  |
| --- | --- | --- |
| **모델** | **PCA 수행 전** | **PCA 수행 후** |
| 1. 로지스틱 회귀 | 61.24% | - |
| 1. 의사결정 나무 | 66.95% | 68.57% |
| 1. K-NN | 59.63% | - |
| 1. SVM | 63.56% | - |

\*가장 성능이 좋은 모델에만 PCA후 데이터셋에 대한 모델 생성을 추가적으로 수행하였음

**의사 결정 나무 결과 (PCA 후):**



결과를 보면, class 1에 대한 예측 성능이 좋지만 class 0과 class 2에 대한 정밀도(precision)이 현저히 낮은 것을 볼 수 있다.

이는 class 0과 2에 대한 합성 데이터 개수가 많아 데이터가 왜곡되었을 가능성이 있다고 추측된다.

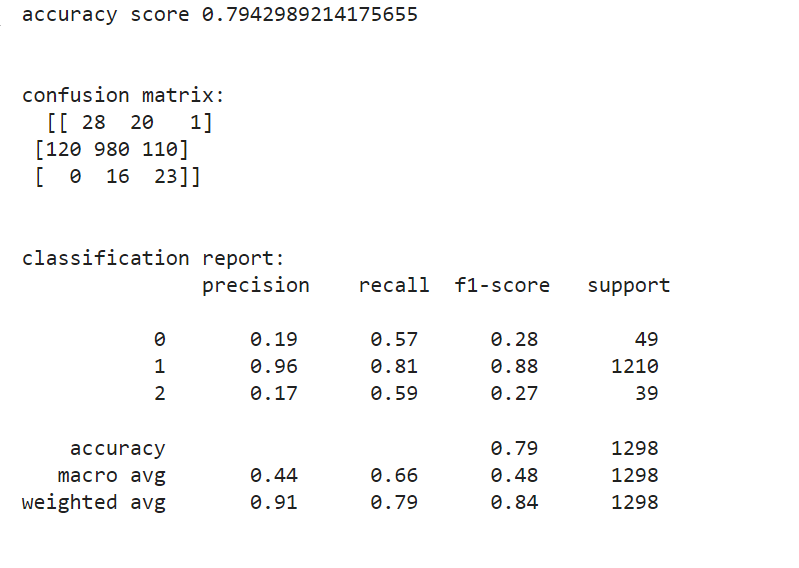
PCA 후의 모델 성능이 더 높은 것을 보아, 주요 특성을 잘 찾아내고 불필요한 변동성을 제거된 것으로 추측된다.

**5번. 앙상블 기법 적용 결과**

|  |  |  |
| --- | --- | --- |
| **모델** | **PCA 수행 전** | **PCA 수행 후** |
| 1. Bagging(DT) | 77.5% | - |
| 2. AdaBoosting | 59.1% | - |
| 3. RandomForest | 66.1% | - |
| 4. XGBoost | 79.43% | 76.89% |
| 1. LGBM | 74.5% |  |

\*가장 성능이 좋은 모델에만 PCA후 데이터셋에 대한 모델 생성을 추가적으로 수행하였음

**XGBoost 결과 (PCA 전):**



결과를 보면, 의사결정나무 모델과 동일하게 class 1에 대한 예측 성능이 좋지만 class 0과 class 2에 대한 정밀도(precision)이 현저히 낮은 것을 볼 수 있다.

또한, 의사결정나무와 비교해보았을 때에도 class 1에 대한 예측 성능은 더욱이 높아졌지만 다른 클래스에 대한 예측 성능은 동일하다.

이는 합성 데이터로 인한 데이터 왜곡이 발생했을 가능성이 높다는 것을 말해준다.

XGBoost 모델 같은 경우, PCA 전이 PCA 후보다 성능이 더 좋다.

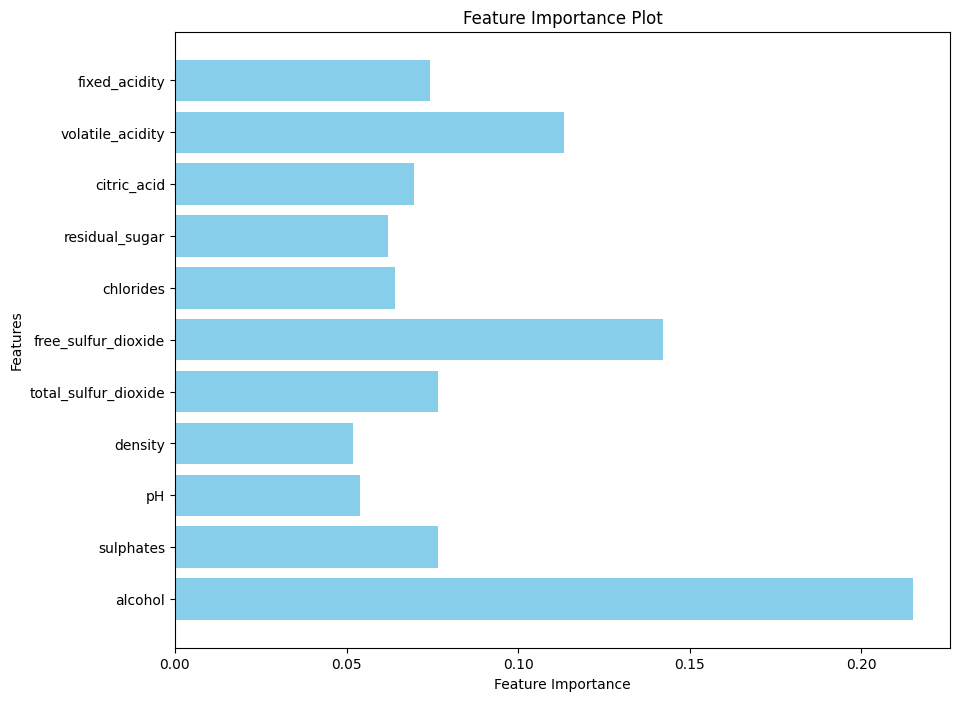
이유를 추측해보았을 때, 앙상블 모델인 XGBoost는 복잡한 알고리즘을 가지고 있어 주요 특징들만을 추출했을 때보다 전체 독립 변수를 모두 가지고 있을 때 정보 손실이 덜 일어나 성능이 비교적 좋은 것이라고 판단된다.

**6번. 결과 분석**

가장 좋은 성능을 가진 모델은 정확도가 79.43%인 PCA를 수행하지 않은 XGBoost Model이다.

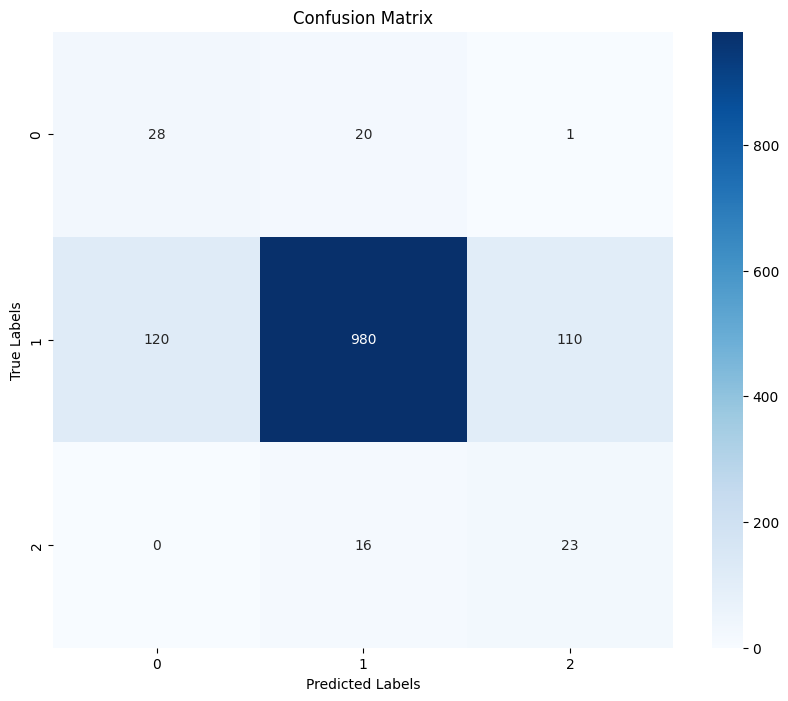
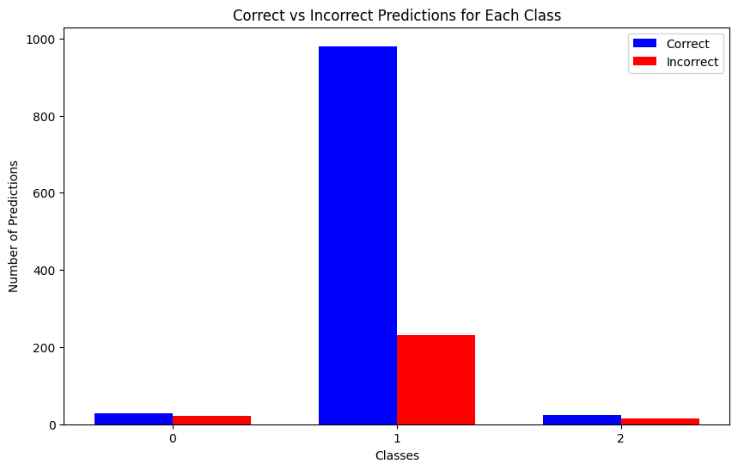
**1) Feature Importance 추출**

Feature Importance를 추출하였을 때, 결과는 아래의 표와 같다.



종속 변수와 상관관계가 가장 높았던 alchol이 가장 변수 중요도가 높다는 것을 확인할 수 있다.

**2) 예측 결과 시각화**

** **

위의 그림을 보았을 때에도 class 1에 대한 예측 정확도는 높지만, 다른 class는 정확도가 현저히 낮고 제대로 예측을 수행하지 못한다는 것을 볼 수 있다.

Quality 점수가 5, 6, 7에 몰려 있지 않고 분산되어 있거나 더 다양한 데이터가 수집되었다면, 더 좋은 성능이 나왔을 것이라고 생각된다.