LG Aimers 2기 – 오프라인 해커톤

스마트 공장의 제어 시스템 구축을 위한 제품 품질 분류 AI 모델 개발

[노아의 방주]

정재윤 윤한나 함지율

주최 LG AI Research 주관 DACON

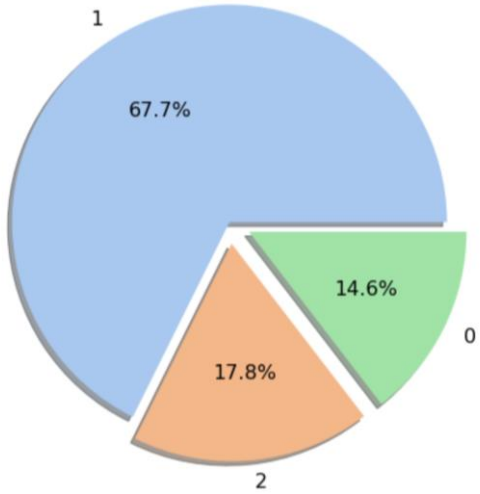
CONTENTS

01 데이터 분석

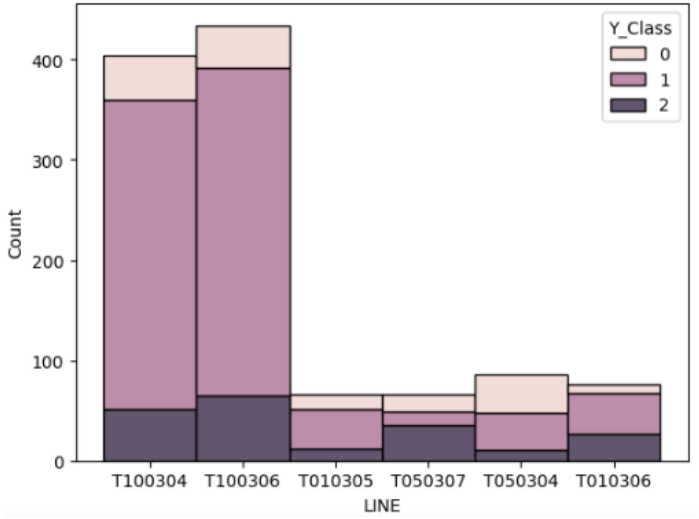
1. **Class Label 분포 분석**

2. **LINE, PRODUCT_CODE → 식별화된 변수들에 따른 Class 분포 분석**

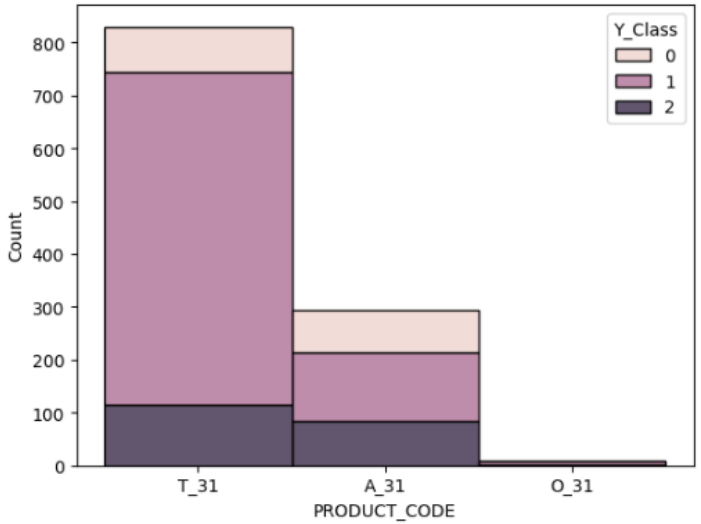
3. **Y_Quality의 분포 확인**



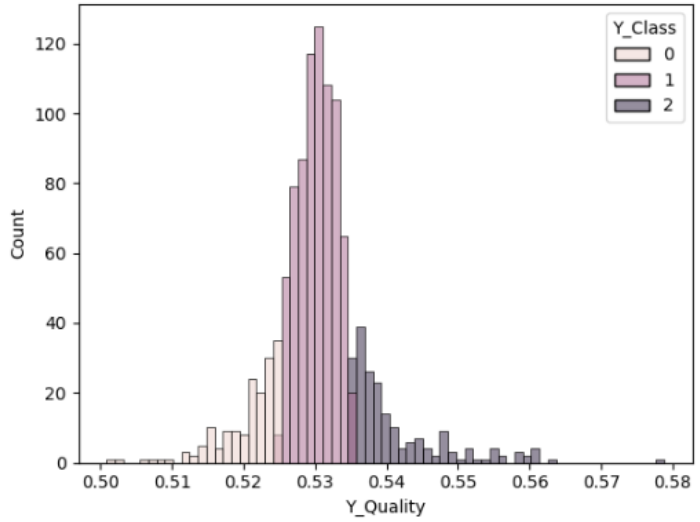
Class Label의 분포



LINE별 Y_Class 분포



PRODUCT_CODE별 Y_Class 분포

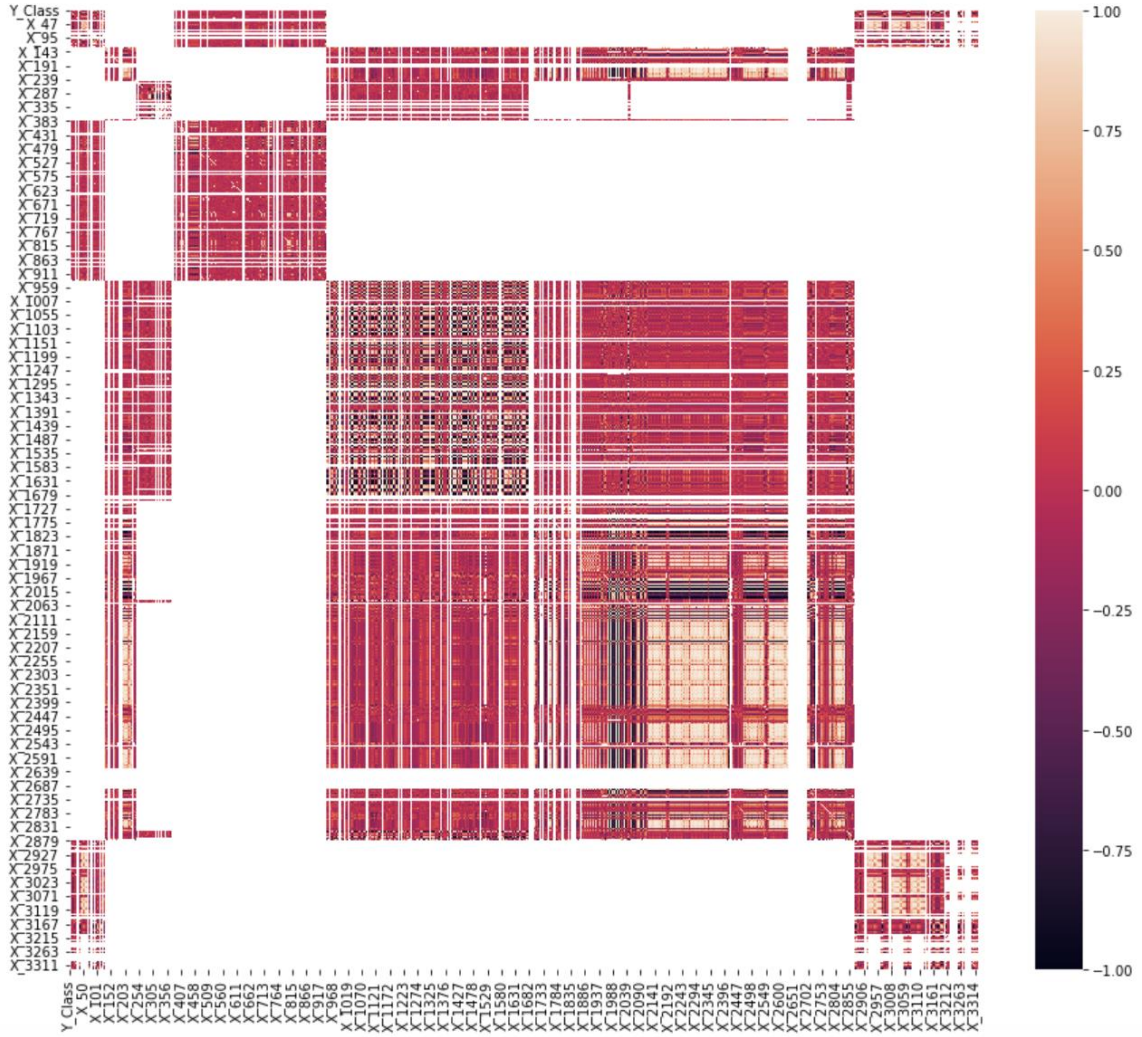


Y_Quality 분포

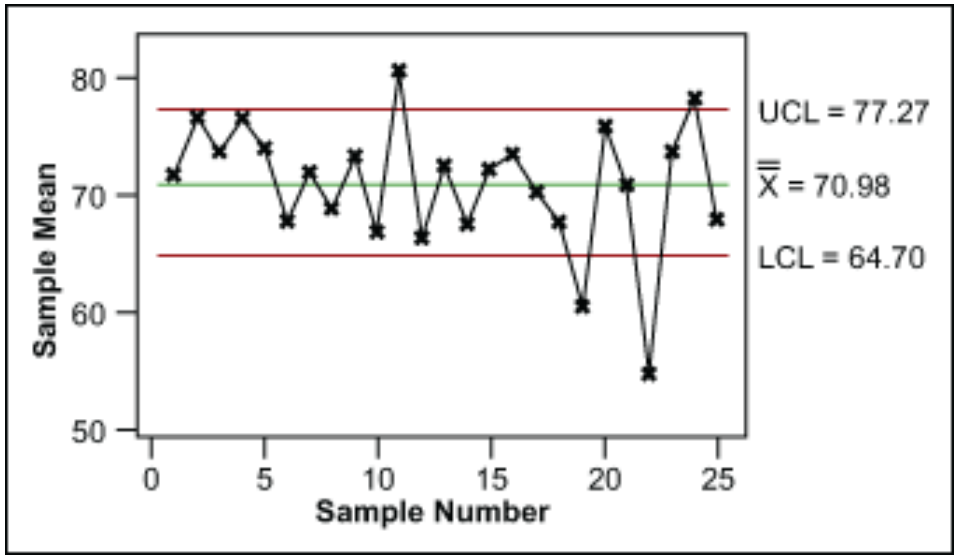
4. Feature 상관관계 분석

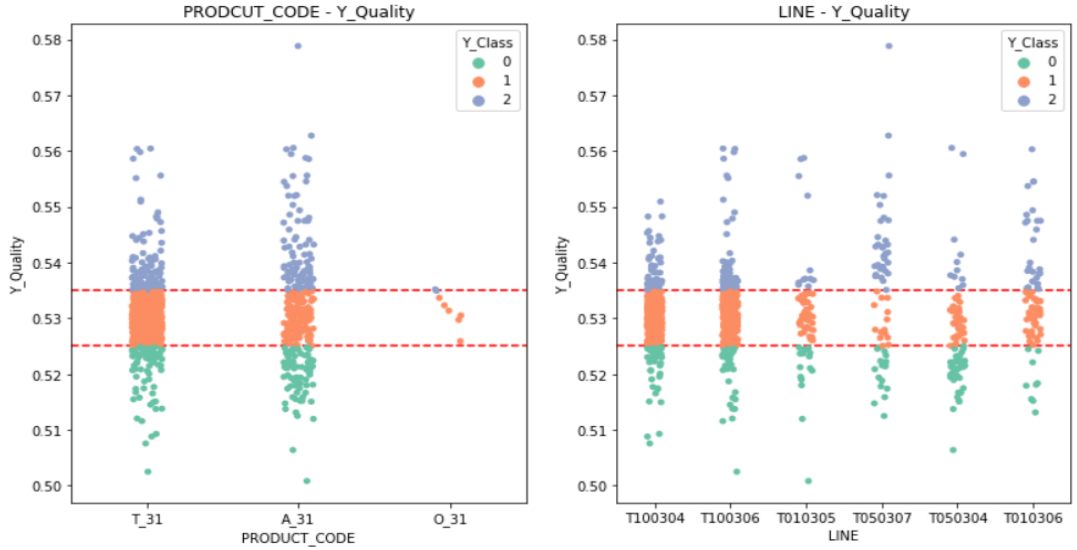
- Catboost의 Feature Importance, X_value Heatmap, Shapley value 등의 분석을 사용해 전반적인 데이터 분석

```
model = CatBoostClassifier(iterations=1000,
                           random_seed=42,
                           learning_rate=0.1,
                           max_depth=5,
                           grow_policy="Depthwise",
                           verbose=0,
                           task_type="GPU")

summary = model.select_features(
    X = X_train,
    y = y_train,
    features_for_select = '0-2876',
    eval_set=(X_valid, y_valid),
    num_features_to_select=500,
    steps=50,
    train_final_model=False,
    logging_level='Silent',
    plot=False
)

result.append(summary['selected_features'])
```



5. 도메인 지식 및 아이디어 수집





- 도메인 지식을 통한 데이터 이해를 위해 제조 공정 공정관리도를 착안함
- 공정 관리도에서 품질 값의 상한선(UCL)과 하한선(LCL) 사이에 위치할 경우 양품으로 판단함

→ Train 데이터 셋의 Y_Quality 값을 사용해 regression 모델도 예측에 사용

02 데이터 전처리

1. Categorical 데이터 Label Encoding 방식으로 변환

```
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()
le.fit(train['LINE'])
train['LINE'] = le.transform(train['LINE'])
test['LINE'] = le.transform(test['LINE'])

le.fit(train['PRODUCT_CODE']) # only train data
train['PRODUCT_CODE'] = le.transform(train['PRODUCT_CODE'])
test['PRODUCT_CODE'] = le.transform(test['PRODUCT_CODE'])
```

2. 다양한 결측지 보간법을 사용하였으나, LINE 별로 결측치가 있는 feature가 다르고 결측치가 너무 많아 기존 데이터를 사용한 결측 방법이 무의미했음
   → train, test 모두 결측치 대체 X

3. Y_Quality 예측 모델
   → 특정 변수 선택법을 사용하지 않고 모든 변수 사용함

4. Y_Class 분류 모델
   → CatBoostClassifier의 Shapley value값을 Stratified KFold로 300개씩 5회 선택하여 사용

03 모델 선정

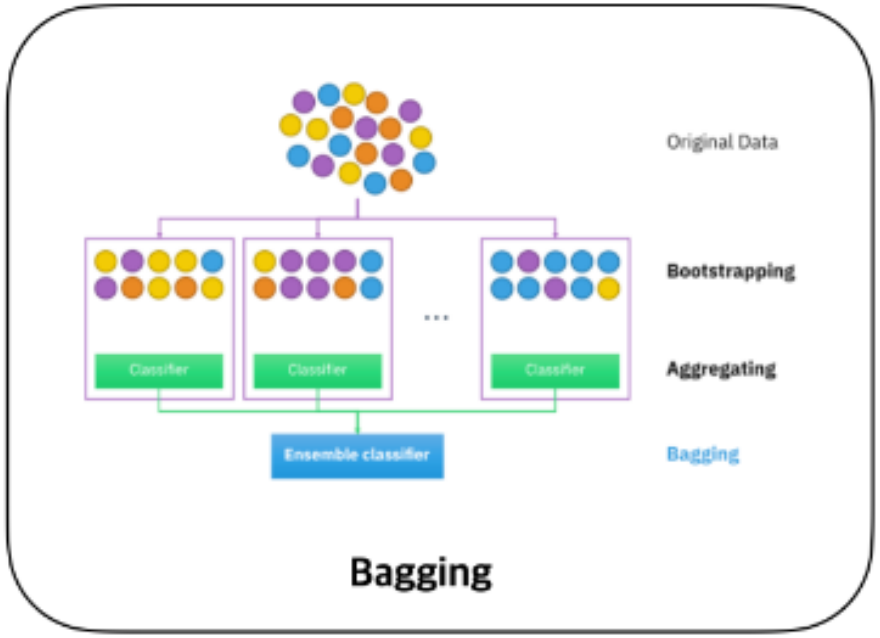
CatBoostRegressor 사용
CatBoostClassifier, XGBClassifier, LightGBMClassifier 사용

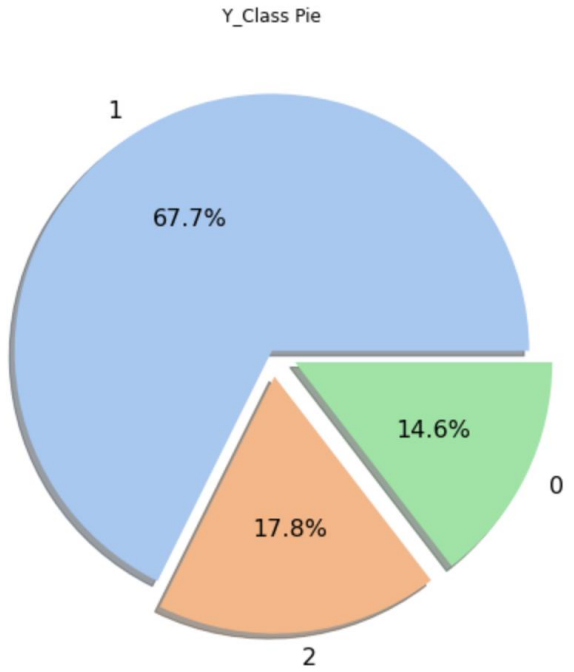
모델 설명

- Gradient Boosting 알고리즘을 기반으로 하는 모델
- Decision tree 구조
- CatBoost, XGBoost 는 level-wise, LGBM은 leaf-wise로 서로 상호보완이 가능함

04 Validation Set 구축 전략

- 전체 train data셋의 개수가 1132개이며, 피처의 개수는 3328개 정도로 아주 많기 때문에 학습을 위한 데이터셋이 터무니없이 적다는 것을 파악
- 따라서 Stratify 10-fold 기법을 이용하여 전체 train 데이터셋을 학습에 사용할 수 있도록 하였으며, 학습과 검증에 사용되는 개별 샘플이 동일한 class 분포를 가지도록 했고, 3번의 시드 앙상블을 추가하여 총 30번의 개별 샘플을 학습하고, 검증할 수 있도록 구축

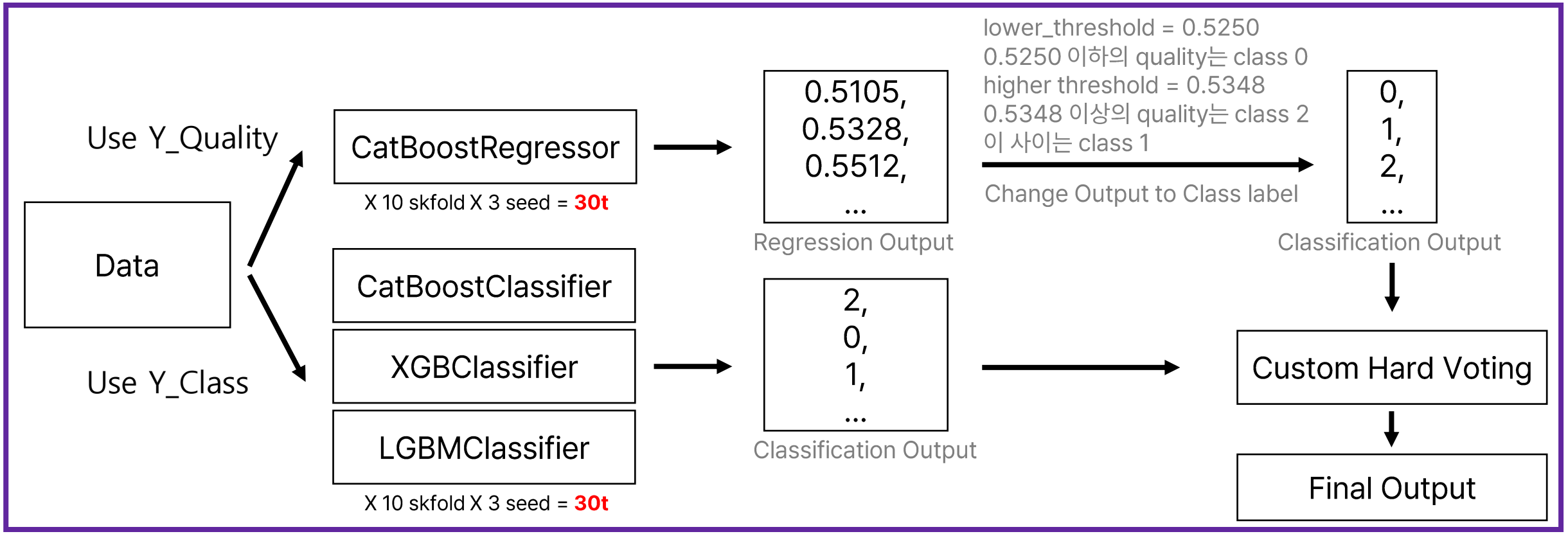




05 전체 과정

1. Data의 개수가 적기 때문에 seed 앙상블 및 stratify kfold 기법을 이용해서 전체 train 데이터에서 최대한 다양한 샘플들을 사용해 모델을 학습시킬 수 있는 방법으로 데이터를 사용함

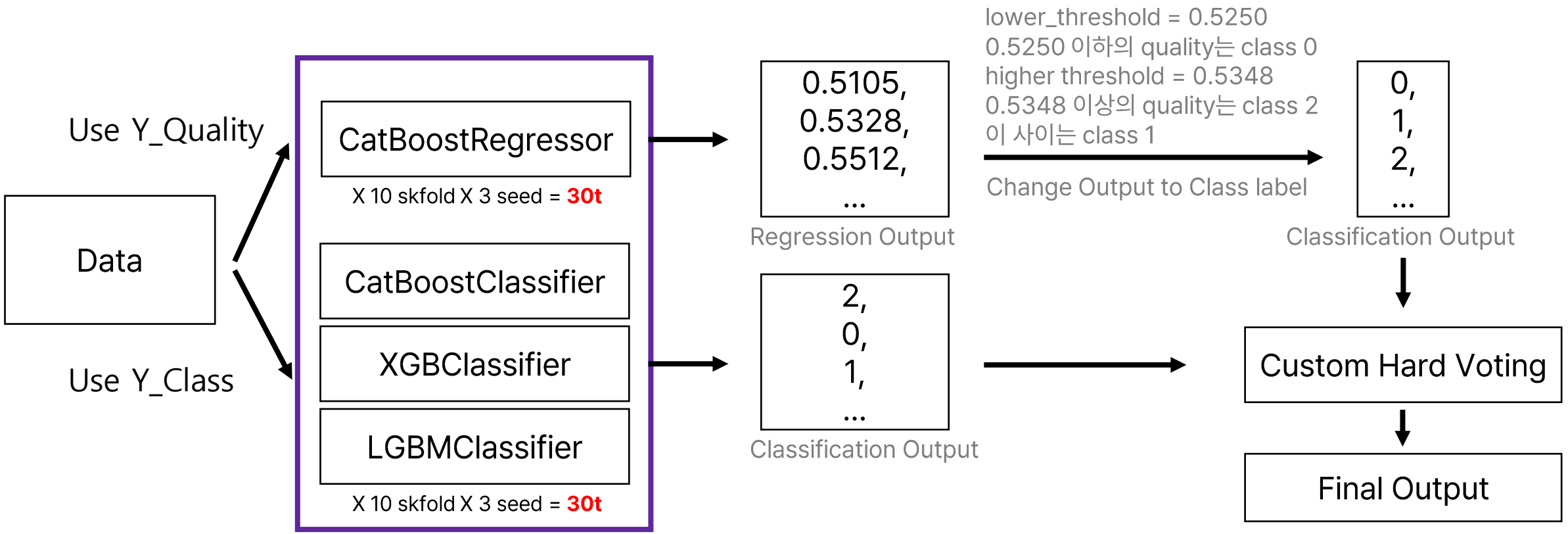
2. 최종 예측을 위해 단일 Regressor 모델과, 3개의 Classifier 모델을 통해 서로 다른 성향의 예측을 진행하여 Custom Hard Voting을 통해 좀 더 robust한 결과를 도출하도록 구축함



05 전체 과정

- **Regressor의 경우에는 하이퍼파라미터 튜닝을 최소화 하고 feature를 하나도 드랍하지 않은 원본 학습 데이터셋을 사용하여 generalize한 능력을 갖추도록 학습함**

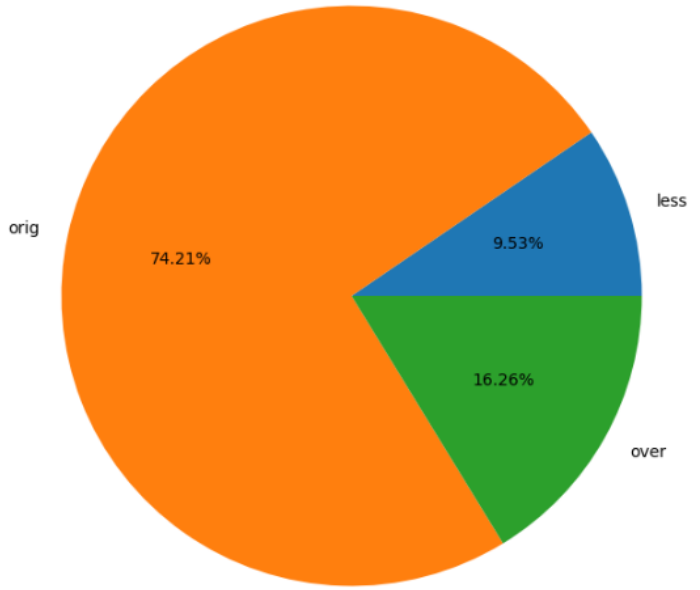
- **Classification들의 경우에는 하이퍼파라미터 튜닝을 더욱 하고, feature를 드랍시킨 학습 데이터셋을 사용하여 데이터에 어느정도 fit된 능력을 갖추도록 학습함**

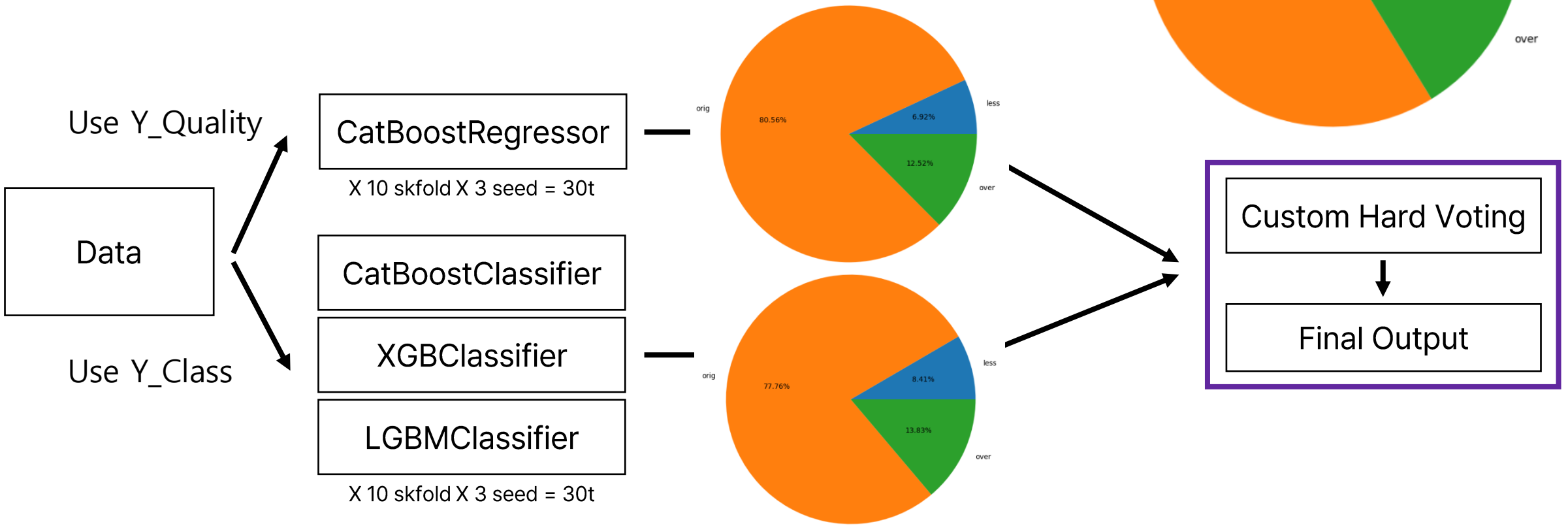


05 전체 과정

- 최종 regression 모델 및 classification 모델의 결과를 합치는 Custom Hard Voting은, 예측 결과가 서로 다를 경우에 0과 2에 가중치를 주어서 예측하도록 함

- 따라서 Custom Hard Voting을 통해 최종 모델이 **under 및 over 데이터를 더욱 잘 검출**할 수 있도록 구축함





Q&A

감사합니다.