1. 산탄데르 고객 만족 예측

(1) 대회 소개
산탄데르 은행에서 고객의 만족도를 높이기 위해 1:1 맞춤 금융 상품을 추천하는 머신러닝 알고리즘을 사용하고자 함.

[프로젝트 목적]
*자사의 금융 서비스를 이용하는 고객들의 특성 분석
*자사의 고객들을 대상으로 고객 맞춤형 상품 추천을 제공
=> 고객의 과거 이력과 유사한 고객군들의 데이터를 기반으로 다음달에 해당 고객이 무슨 상품을 사용할지 예측
=> 고객의 만족도를 높임과 동시에 은행 매출에 기여

-> 만족/ 불만족한 고객의 데이터를 분류하는 이진 분류 문제

(2) XGBoost 학습 모델
XGBoost : GBM에 기반해서 느린 수행 시간 및 과적합 규제 부재 등의 단점을 보완한 분류에 있어서 뛰어난 예측 성능을 나타내는 알고리즘.

(3) LightGBM 학습 모델
LightGBM : XGBoost와 함께 부스팅 계열 알고리즘에 속하며 XGBoost보다 빠른 학습과 예측 수행 시간, 더 작은 메모리 사용량을 자랑하는 알고리즘

cf. elarly_stopping_rounds
사이킷런 래퍼 XGBoost에서 조기 중단 관련 파라미터 중 하나.
평가 지표가 향상될 수 있는 반복 횟수를 정의.
주의! -> 조기 중단값을 너무 급격하게 줄이면 성능이 향상될 가능성이 있음에도 반복이 멈춰버려서 충분한 학습이 되지 않아 예측 성능이 저하될 수 있음.

(4) ROC 곡선과 AUC
ROC 곡선 : FPR이 변할 때 TPR이 어떻게 변하는지를 나타내는 곡선
AUC : ROC 곡선의 넓이

- ROC 곡선이 가운데 직선에서 멀어질수록 성능이 뛰어남.
- AUC는 1에 가까울수록 좋은 수치
- AUC가 커지려면 FPR이 작은 상태에서 얼마나 큰 TPR을 얻을 수 있는지가 관건.

2. 심장병 발병 예측

(1) 대회 소개

- 주제 : 심장병 사례분석을 위한 분류 모델 생성

- 분류 문제

- 진행 단계

1. 데이터에 대한 상세한 탐색적 분석 수행
2. 어떤 metric을 사용할지 결정
3. target data와 feature data들을 모두 분석(데이터 탐색)
4. 모형에 적용하기위해 범주형 변수를 숫자로 변환(Scaling)
5. data leakage를 방지하기위해 파이프라인 사용(make_column_transformer)
6. 각 모델의 결과를 보고 가장 적합한(정확도가 높은) 모델 선택
   => 개선 사항을 확인하기 위해 Optuna를 사용하여 Catboost의 하이퍼 파라미터를 튜닝
7. 각 feature의 중요도와 각 모델의 정확도 확인

(데이터 요약)

traget data는 균형 있는 데이터에 가깝다고 볼 수 있음.
수치형 변수들은 대상 변수와 약한 상관 관계를 가짐.
Oldpeak는 심장 질환과 양의 상관관계가 있음.
MaxHR은 심장 질환과 음의 상관관계를 가짐.
Cholesterol은 심장병과 음의 상관관계를 가지고 있음.
By Sex : 남자가 여자보다 심장병에 걸릴 확률이 거의 2.44배 높음.
ChestPainType별로 뚜렷한 차이를 관찰할 수 있음.
=> ASY인 사람 : 무증상성 흉통은 ATA 흉통을 가진 사람보다 심장 질환을 가질 가능성이 6배 정도 더 높음.
RestingECG : 휴식 심전도 결과는 크게 다르지 않음.
=> ST인 사람 : ST-T 파동 이상을 갖는 사람은 다름 사람들보다 심장병을 가질 가능성이 더 높음.
ExercisingAngina이 있는 사람이 없는 사람에 비해 심장 질환이 있을 확률이 거의 2.4배 높음.
ST_Slope에 따른 심장병 발병 비율의 차이가 있음.
=> ST_Slope가 Up인 사람은 다른 두 유형의 사람들에 비해 심장병에 걸릴 확률이 현저히 낮음.

(2) 모델링

- 기준선 모델로 dummy Classifier를 사용
- 이후 scaler유무를 다르게 하여 각각 Logistic, Linear Difcriminat, KNeighbors, Supprot Vector Machine 모델을 적용
- 이후 앙상블 모델링 기법인 Adaboost, Randomforest, Gradient Boosting and Extra Trees를 활용

- 유명한 부스팅 모델들인 XGBoost, LightGBM, Catboost를 다룰 예정
- 마지막으로 Catboost를 위한 hyper parameter tuning을 자세하게 알아볼 예정.

cf. sklearn.compose.make_column_transformer
: 데이터에 수치형 변수와 범주형 변수가 섞여 있는 경우, 이들을 각각 따로 Encoding 해주기 위해 사용.
처리할 데이터를 분리한 후, 각각에 맞는 Encoding 기법을 넣어주면 됨.

● CatBoost
-범주형 변수를 더 잘 처리하고, 과적합 문제를 개선한 알고리즘
- 다른 모델들의 경우 범주형 변수를 사용하기 위해서는 One-Hot Encoding 등 데이터 전처리가 선행되어야 하지만, Catboost에서는 자동으로 이를 변환하여 사용할 수 있다는 장점이 있음.
- 내부적 알고리즘을 통한 과적합, sampling 다양성 등의 문제가 개선됨.
  => hyper parameter에 따른 영향이 적음.
[사용 의도]
- 분류 문제에 대한 교육 및 모델 적용 => scikit-learn 도구와의 호환성을 제공

기본 최적화 목표는 다양한 조건에 따라 달라질 수 있음!
Logloss : 대상에 두 개의 다른 값만 있거나 target_border 매개 변수가 None이 아닌 경우
MultiClass :대상에 두 개 이상의 다른 값이 있으며 border_count 매개변수가 None인 경우

● Optuna를 활용한 Catboost의 hyper parameter tuning
Optuna
- 하이퍼 파라미터 최적 프레임워크
- 파라미터의 범위를 지정해주거나 파라미터가 될 수 있는 목록 설정 시 매 trial마다 파라미터를 변경하면서 최적의 파라미터 탐색

[Tuning을 진행할 Catboost의 Parameters]
1. Objective : 과적합 감지 및 최상의 모델을 위해 지원되는 측정 기준
- Logloss : 정답을 더 높은 확률로 예측할수록 좋은 모델이라고 평가
- CrossEntropy : 훈련 데이터를 사용한 예측 모형에서 실제 값과 예측값의 차이를 계산

2. colsample_bylevel : 학습 속도를 높임, 일반적으로 품질에 영향을 미치지 않음.

3. depth : 트리의 깊이

4. boosting_type : 기본적으로 부스팅 유형은 작은 데이터 세트의 경우로 설정됨 -> 과적합을 방지하지만 계산 측면에서 많은 비용이 소모됨.
=> 교육 속도를 높이기 위해 해당 parameter를 활용할 수 있음.

Ordered : 일반적으로 작은 데이터 세트에서 더 나은 품질을 제공하지만 Plain 방식보다 느릴 수 있음.
Plain : 전형적인 GradientBoosting 방식

5. bootstrap_type : 객체의 가중치를 샘플링하는 방법을 정의
Baysian : 앞서 나온 결과의 “경험”을 계속해서 반영하면서 최적 하이퍼파라미터 값에 빠르게 도달할 수 있게 함
Bernoulli : 확률분포를 베르누이 분포로 가정
MVS : 최소 분산 sampling