빅데이터와 금융자료분석 프로젝트 (Team 4)

XGboost 알고리즘을 활용한 은행 대출의 부도 여부 예측 모델 구축

강상묵(20259013) 김형환(20249132) 유석호(20249264) 이현준(20249349) 최영서(20249430) 최재필(20249433)

1. 프로젝트 개요

본 프로젝트는 **여러 데이터 전처리 기법(결측치, 이상치, 특성공학 등)과 머신러닝 알고리즘(이상치 분류, 차원축소, XGBoost 등)을 실제 금융데이터에 적용**해보고 시사점을 도출하기 위해 작성되었습니다.

이를 위해 미국 Lending Club의 P2P 대출 데이터를 사용하였으며, 전반적인 워크플로우는 아래와 같습니다.

1. 데이터의 구조, 특성 파악 (EDA)
2. 데이터의 전처리 (특성에 따른 칼럼 가공, 문자형 변수 처리, 결측치 및 이상치 처리, 변수 선택)
3. 여러 Grediant Boosting 계열의 알고리즘을 이용한 대출 연체여부 예측 모델 구축 및 평가

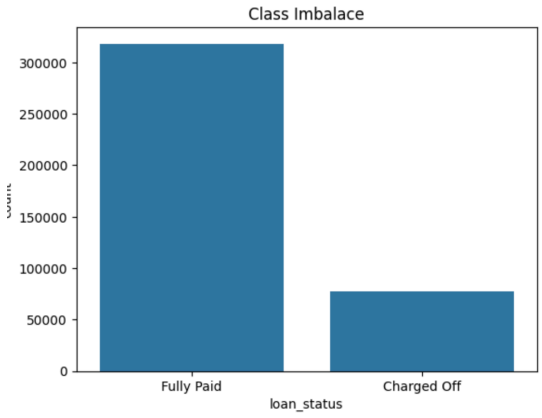
2. 데이터의 구조, 특성 (EDA)

데이터의 수집, 기본구조

미국 소재의 P2P 대출 전문은행인 Lending Club의 '07~'20년 대출 데이터를 사용하였습니다. (출처 : Kaggle)

약 40만개의 데이터로, 목적변수인 대출상태를 포함해 전체 27개의 칼럼(수치형 12 + 문자형 15)으로 이루어져있으며, 목적변수는 정상(상환, Fully paid) 및 부도(연체, Charged off)로 이진분류 문제입니다.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 396030 entries, 0 to 396029
Data columns (total 27 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   loan_amnt             396030 non-null  float64
 1   term                  396030 non-null  object
 2   int_rate              396030 non-null  float64
 3   installment           396030 non-null  float64
 4   grade                 396030 non-null  object
 5   sub_grade             396030 non-null  object
 6   emp_title             373103 non-null  object
 7   emp_length            377729 non-null  object
 8   home_ownership        396030 non-null  object
 9   annual_inc            396030 non-null  float64
 10  verification_status   396030 non-null  object
 11  issue_d               396030 non-null  object
 12  loan_status           396030 non-null  object
 13  purpose               396030 non-null  object
 14  title                 394274 non-null  object
 15  dti                   396030 non-null  float64
 16  earliest_cr_line      396030 non-null  object
 17  open_acc              396030 non-null  float64
 18  pub_rec               396030 non-null  float64
 19  revol_bal             396030 non-null  float64
 20  revol_util            395754 non-null  float64
 21  total_acc             396030 non-null  float64
 22  initial_list_status   396030 non-null  object
 23  application_type      396030 non-null  object
 24  mort_acc              358235 non-null  float64
 25  pub_rec_bankruptcies  395495 non-null  float64
 26  address               396030 non-null  object
dtypes: float64(12), object(15)
memory usage: 81.6+ MB
```



데이터의 특성

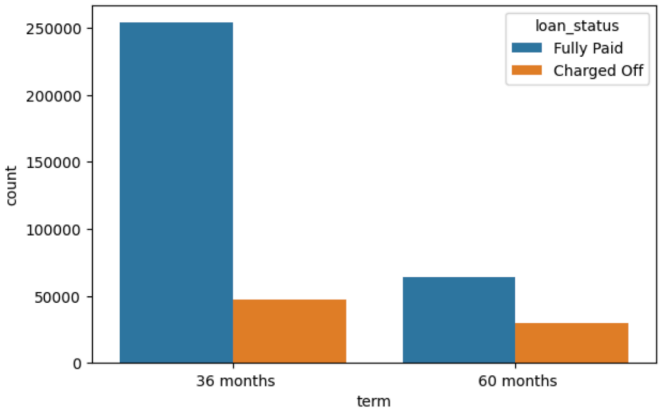
데이터의 각 칼럼별 특징을 알아보고, 적절한 전처리 방법을 탐색해보았습니다.

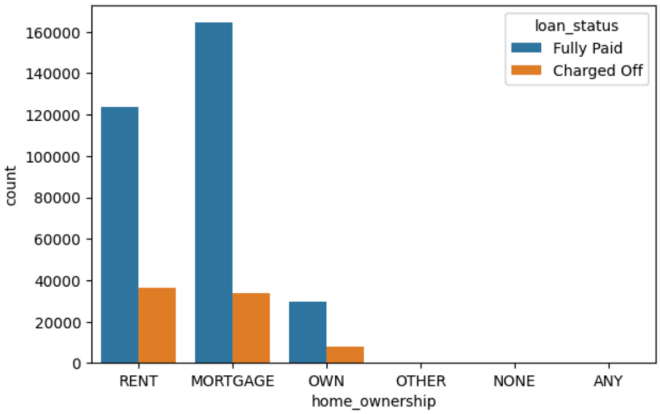
먼저 **수치형 변수**입니다. 결측치 및 이상치 처리는 별도 진행 예정으로 따로 다루지 않겠습니다.

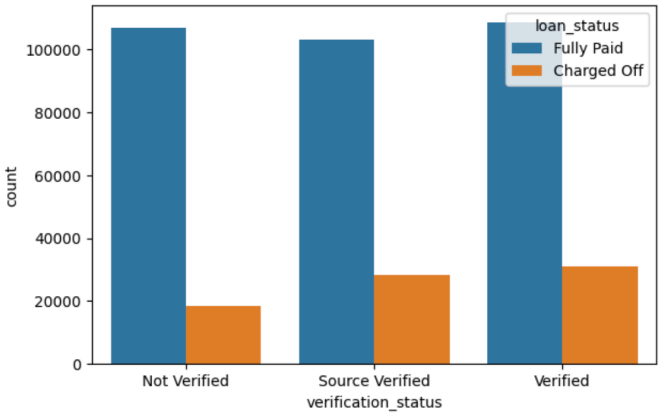
상관관계 행렬을 Heatmap으로 살펴보았습니다. 대체적으로 변수들 간 **상관관계가 미미**하였으며, **일부 상관계수가 높은 변수**들은 변수선택 과정에서 제외하는 등 **별도의 전처리 과정을 통해 다중공선성 문제를 해결**할 계획입니다.

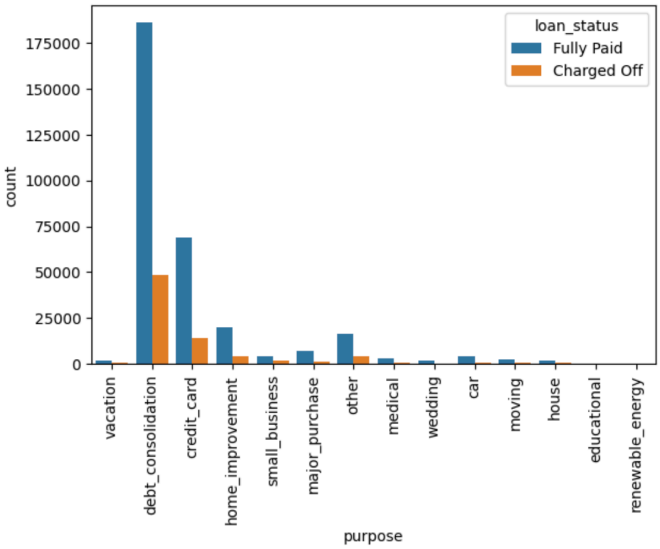
다음으로 **문자형 변수**입니다. 목적변수와 관련이 있는 것으로 보이는 주요 예시만 살펴보겠습니다.

먼저, **대출기간(term), 집보유형태(home_ownership), 대출목적(purpose)이 영향**을 미치는 것으로 추정됩니다.

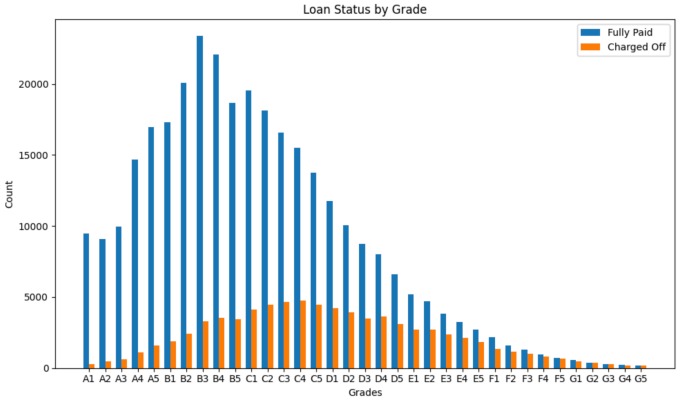








다음으로, **신용등급(A1~G5)에 따라 부도율이 높아지는 추이**를 보였으며, 문자형 변수들 중 **일부는 고유값이 너무 많아 분석에서 제외**하는 것이 효과적일 것으로 보입니다.




```
term                   2
grade                  7
sub_grade             35
emp_title         173105
emp_length            11
home_ownership         6
verification_status    3
issue_d              115
loan_status            2
purpose               14
title              48816
earliest_cr_line     684
initial_list_status    2
application_type       3
address           393700
dtype: int64
```

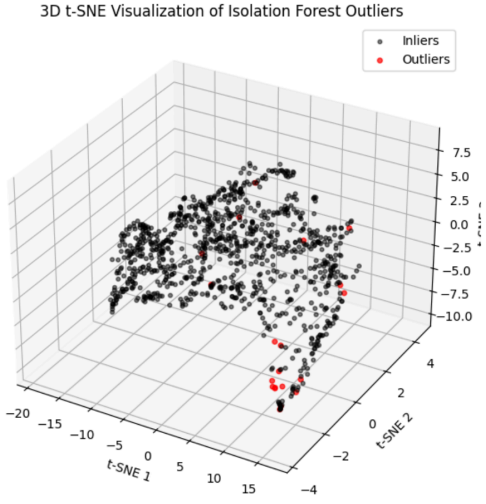
3. 데이터의 전처리

데이터의 전처리는 아래의 과정으로 실시하였습니다.

1. 분석에 적합하도록 칼럼 변환 및 통합, 제거
    - 변환/통합 : 주소(address)는 우편번호(zip_code)만 추출하고 제거, 대출기간(term, 36month 등)은 수치형으로 변환, 집 소유여부(home_ownership)의 극소수값들은 Other로 통합
    - 불필요한 noise 방지를 위해 100개 이상의 고유값을 가진 칼럼 제거 : 직업(title), 직업글자수(emp_title), 발행일(issue_d), 최초연도(earliest_cr_line)
    - 다른 변수와 중복되거나 추론 가능한 칼럼 제거 : 신용점수-대분류(grade), 근속연수(emp_length)
2. 문자형 변수 처리 : 순서가 있거나 이진변수인 경우 라벨인코딩, 단순 점주인 경우 원핫인코딩 적용
    - 라벨인코딩 : 목적변수(이진), 신용점수(순서 존재) / 원핫인코딩 : 이외의 문자형 변수
3. 수치형 변수의 결측치 및 이상치 처리 : 중간값 처리 및 1% 이상치 제거
    - 결측치 : 변수간 상관관계가 미미하고, 이후 Boruta를 적용 예정이므로 예측형 모델보다는 중간값을 채택
    - 이상치 : 고차원, 많은 샘플(약 40만)을 고려, 분포에 대한 가정이 불필요한 Isolation forest 기법 채택
4. 변수 선택을 통해 분석에 적합한 최종 데이터 가공 : Boruta 알고리즘 적용
    - 일부 변수간 상관관계가 존재하는 점을 고려, 최적의 변수 조합을 찾고자 Boruta 알고리즘 채택
    - 원핫인코딩 대상 변수를 제외한 13개(수치형+라벨)에 알고리즘을 적용한 결과 11개의 변수를 선택하였고, 원핫인코딩 대상 변수와 결합하여 최종 데이터 구성

i Isolation Forest 검증(T-SNE 적용) 및 최종 데이터 구성

수치형 변수에 **T-SNE를 적용하여 3차원으로 축소**한 결과, **이상치 제거(Isolation Forest)가 적절히 작동**하였으며, **Boruta 알고리즘으로 변수 선택**까지 마친 후 **최종 데이터**는 **7개의 문자형 변수**(원핫인코딩 6 + 라벨인코딩 1) 및 **9개의 수치형 변수, 1개의 목적변수(이진분류)**로 구성되어 있습니다.




```
<class 'pandas.core.frame.DataFrame'>
Index: 274448 entries, 3412 to 121958
Data columns (total 41 columns):
 #   Column                              Non-Null Count   Dtype
---  ------                              --------------   -----
 0   loan_amnt                           274448 non-null  float64
 1   term                                274448 non-null  int64
 2   int_rate                            274448 non-null  float64
 3   installment                         274448 non-null  float64
 4   sub_grade                           274448 non-null  int64
 5   annual_inc                          274448 non-null  float64
 6   dti                                 274448 non-null  float64
 7   revol_bal                           274448 non-null  float64
 8   revol_util                          274448 non-null  float64
 9   total_acc                           274448 non-null  float64
 10  mort_acc                            274448 non-null  float64
 11  home_ownership_OTHER                274448 non-null  bool
 12  home_ownership_OWN                  274448 non-null  bool
 13  home_ownership_RENT                 274448 non-null  bool
 14  verification_status_Source Verified 274448 non-null  bool
 15  verification_status_Verified        274448 non-null  bool
 16  purpose_credit_card                 274448 non-null  bool
 17  purpose_debt_consolidation          274448 non-null  bool
 18  purpose_educational                 274448 non-null  bool
 19  purpose_home_improvement            274448 non-null  bool
 20  purpose_house                       274448 non-null  bool
 21  purpose_major_purchase              274448 non-null  bool
 22  purpose_medical                     274448 non-null  bool
 23  purpose_moving                      274448 non-null  bool
 24  purpose_other                       274448 non-null  bool
 25  purpose_renewable_energy            274448 non-null  bool
 26  purpose_small_business              274448 non-null  bool
 27  purpose_vacation                    274448 non-null  bool
 28  purpose_wedding                     274448 non-null  bool
 29  initial_list_status_w               274448 non-null  bool
 30  application_type_INDIVIDUAL         274448 non-null  bool
 31  application_type_JOINT              274448 non-null  bool
 32  zip_code_05113                      274448 non-null  bool
 33  zip_code_11650                      274448 non-null  bool
 34  zip_code_22690                      274448 non-null  bool
 35  zip_code_29597                      274448 non-null  bool
 36  zip_code_30723                      274448 non-null  bool
 37  zip_code_48052                      274448 non-null  bool
 38  zip_code_70466                      274448 non-null  bool
 39  zip_code_86630                      274448 non-null  bool
 40  zip_code_93700                      274448 non-null  bool
dtypes: bool(30), float64(9), int64(2)
memory usage: 33.0 MB
```

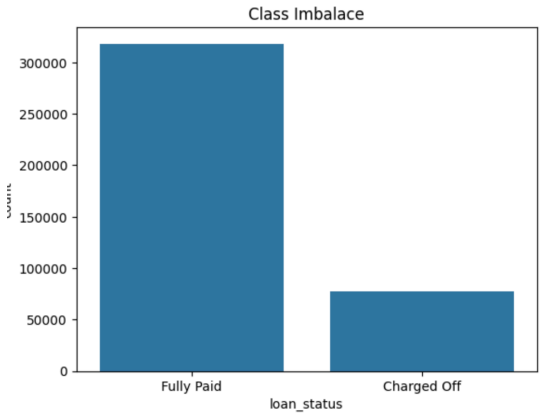
4. 대출 연체여부 예측 모델 구축 및 평가

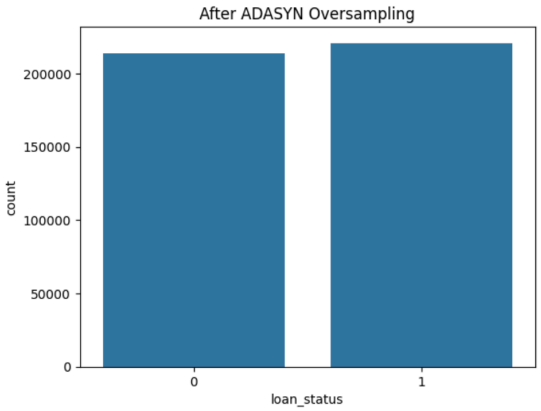
알고리즘 소개 및 오버샘플링

앞서 구성한 40개 변수로 **"대출 연체 여부"를 예측하는 모델을 여러 Grediant Boosting 계열의 알고리즘**을 통해 구축할 예정입니다. **XGBoost 알고리즘을 중심**으로, 다양한 알고리즘과 비교하여 분석하도록 하겠습니다.

- **XGBoost**: Regularization과 트리 구조 최적화에 강점을 가진 Gradient Boosting 모델 |
- **CatBoost**: 범주형 변수 자동 인식 기능이 있는 Gradient Boosting 기반 모델 |
- **LightGBM**: 빠른 학습 속도와 낮은 메모리 사용의 Gradient Boosting 기반 모델 |
- **Soft Voting**: CatBoost, LightGBM의 예측 확률 평균을 통한 결합(앙상블) 모델 |
- **Stacking**: CatBoost, LightGBM의 예측 결과를 Logistic Regression에 전달하는 메타 모델 기반 앙상블 |

본격적인 모델 구축에 앞서, 클래스 불균형 해소를 위해 **ADASYN을 이용하여 훈련데이터를 오버샘플링** 하였습니다.





하이퍼파라미터 튜닝

오버샘플링된 훈련데이터를 이용하여 **RandomizeCV** 방식으로 튜닝하였습니다. **과적합 문제**를 피하기위해 적정 파라미터 그룹을 구성하고 L1/L2/최소가중치 등 **여러 규제를 적용**하였습니다. 또한, **클래스 불균형을 고려하여 F1-score를 기준**으로 진행하였습니다. **XGBoost 모델의 하이퍼파라미터 튜닝 결과**는 아래와 같습니다.

```
Fitting 3 folds for each of 15 candidates, totalling 45 fits
Best Params: {'subsample': 0.8, 'n_estimators': 200, 'max_depth': 3, 'learning_rate': 0.05, 'colsample_bytree': 0.8}
Best F1 Score (CV): 0.923741343521181
```

> 모델별 하이퍼파라미터 튜닝 요약
>
> - **XGBoost**: `RandomizedSearchCV`로 `n_estimators`, `max_depth`, `eta`, `gamma` 등 튜닝
> - **CatBoost**: `RandomizedSearchCV`로 `depth`, `iterations`, `learning_rate` 등 튜닝
> - **LightGBM**: `RandomizedSearchCV`로 `n_estimators`, `learning_rate`, `max_depth` 등 튜닝
> - **Soft Voting**: `CatBoost`, `LightGBM`을 사용하여 예측 확률 평균 산출
> - **Stacking**: base 모델로 `CatBoost`, `LightGBM` 사용, 메타 모델로 `LogisticRegression`

모델 일반화성능 평가

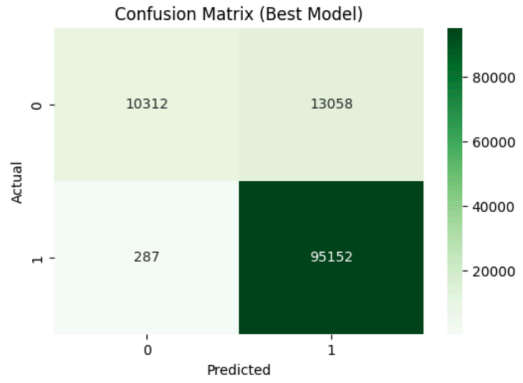
최종 모델과 평가데이터를 통해 일반화 성능을 비교해보겠습니다.

모든 모델에 있어서 **F1-score는 0.93 이상, ROC-AUC는 약 0.9로 실제 연체 여부를 잘 예측**하는 것으로 보입니다. 또한, XGBoost 모델의 **F1-score가 튜닝 과정 보다 개**선된 것은 **과적합 방지 기법이 성과**가 있었음을 시사합니다.

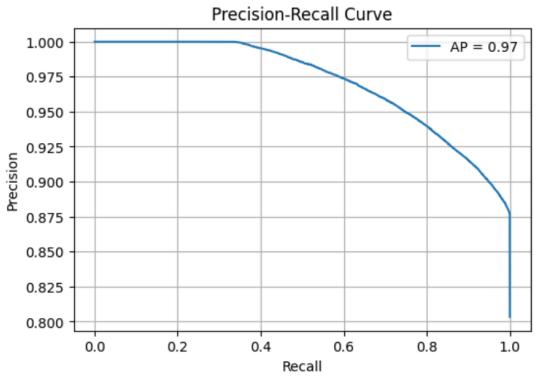
모델	F1 Score	ROC AUC
XGBoost	0.9345	0.8967
CatBoost	0.9350	0.9032
LightGBM	0.9348	0.9068
Soft Voting	0.9352	0.9061
Stacking	0.9316	0.9068

그러나, **샘플이 적은 "부도"인 경우, 예측 성능이 다소 떨어지는 모습**이 관측되었습니다.

부도의 절반 이상이 정상으로 분류되었으며 모든 모델에 동일한 문제가 있는 것으로 볼 때, **데이터의 한계**인 것으로 보입니다. 또는 **신경망 계열을 적용해보는 것도 개선방법**이 될 수 있습니다.



(a) XGBoost Confusion Matrix



(a) XGBoost PRCurve

마지막으로, **변수 중요도(feature importance)**를 살펴보겠습니다.

부도 여부에는 예상 외로 소득이나 대출금액이 아닌 **대출목적과 주거지가 큰 영향**을 미치는 것을 확인할 수 있었으며, 이외에도 **이자율(int_rate) 및 대출기간 순으로 분류에 영향**을 미치는 것을 알 수 있었습니다.

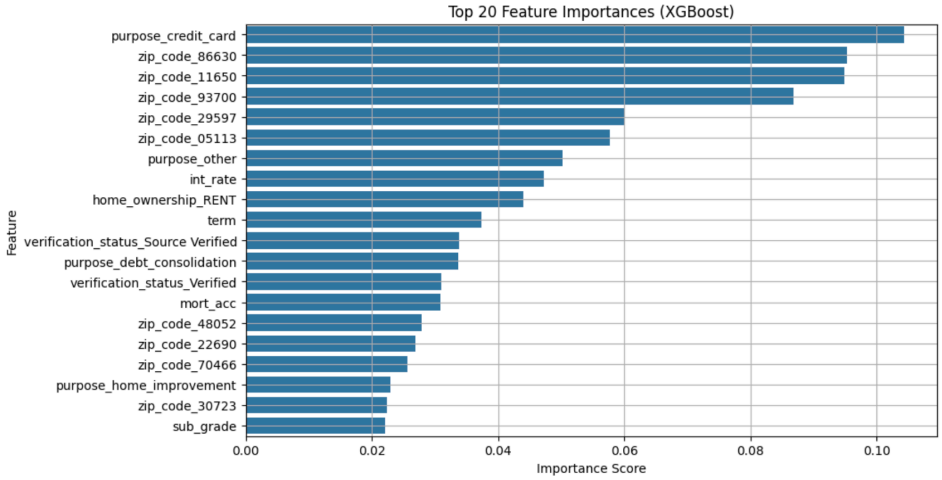


Figure 1.3: XGBoost Feature Importance

시사점

이번 프로젝트를 통해 실제 은행의 데이터를 살펴보고, 대출의 부도 확률 예측 모델을 구축해보았습니다.

먼저 **실제 데이터를 전처리**하는 과정에서 발생하는 결측치, 이상치, 적합하지 않은 변수 분류 등의 **문제점을 실제로 경험**할 수 있었고 Isolation Forest 및 T-SNE, Boruta 알고리즘을 적용해보면서 **각 알고리즘이 어떻게 작동하는지, 어떤 방식으로 문제를 해결하고 활용**되는지 알 수 있었습니다.

또한, **XGBoost 알고리즘을 모델 구축에 활용**하면서 앙상블 계열의 grediant boosting 알고리즘이 **금융데이터 예측에 강력한 성능**을 가진 것을 확인하였고, 모델 성능에는 알고리즘의 선택 및 튜닝 뿐만아니라 **목적변수의 불균형을 해소(oversampling)하고 변수를 적절히 선별(boruta)하는 것이 매우 중요**하다는 것을 느꼈습니다.

전반적으로 수업시간에 다룬 여러 알고리즘을 통해 이론이 실제 세상에 적용되는 과정을 이해하게 되었고, 무엇보다 **적합한 데이터를 구하고 적절히 전처리하는 것이 매우 중요**하다는 것을 알게 된 프로젝트였습니다.