Loan Interest Rate Prediction

Introduction

Similar dataset, different problem

Credit risk analysis is relatively new thing to me, so before I started my data exploration and building models, I did small research on that topic including relevant machine learning models. I found that my dataset I was going to work on have similarity with **Lending Club Loan Data** that available online.

There are many researches on Lending Club's data, but the goal is totally different with my goal. While the existing research focused on predicting potential default borrowers, mine is to classify appropriate interest rate for the borrowers to minimize risk. I find that difference is super interesting.

With all those finding, the model I will build is based on previous research unless I will be focusing on ensemble models since the best model from previous research is ensemble algorithms and the problem I am trying to solve is multiclass classification so linear model will perform poor in running time let alone performance.

Project Details

Machine

Thinkpad T430 (Ubuntu 20.10 LTS, Intel Core i5-3320 M, 16GB RAM)

Workflow

1. Research
2. Recreate best model based on previous researches
3. Train and evaluate
4. Make improvement by feature engineering and building more advanced model

Packages

Python packages I used in this project are:

1. pandas
2. plotly
3. numpy
4. scipy
5. scikit-learn
6. pyjanitor
7. autosklearn
8. xgboost
9. lightGBM
10. catboost
11. imblearn

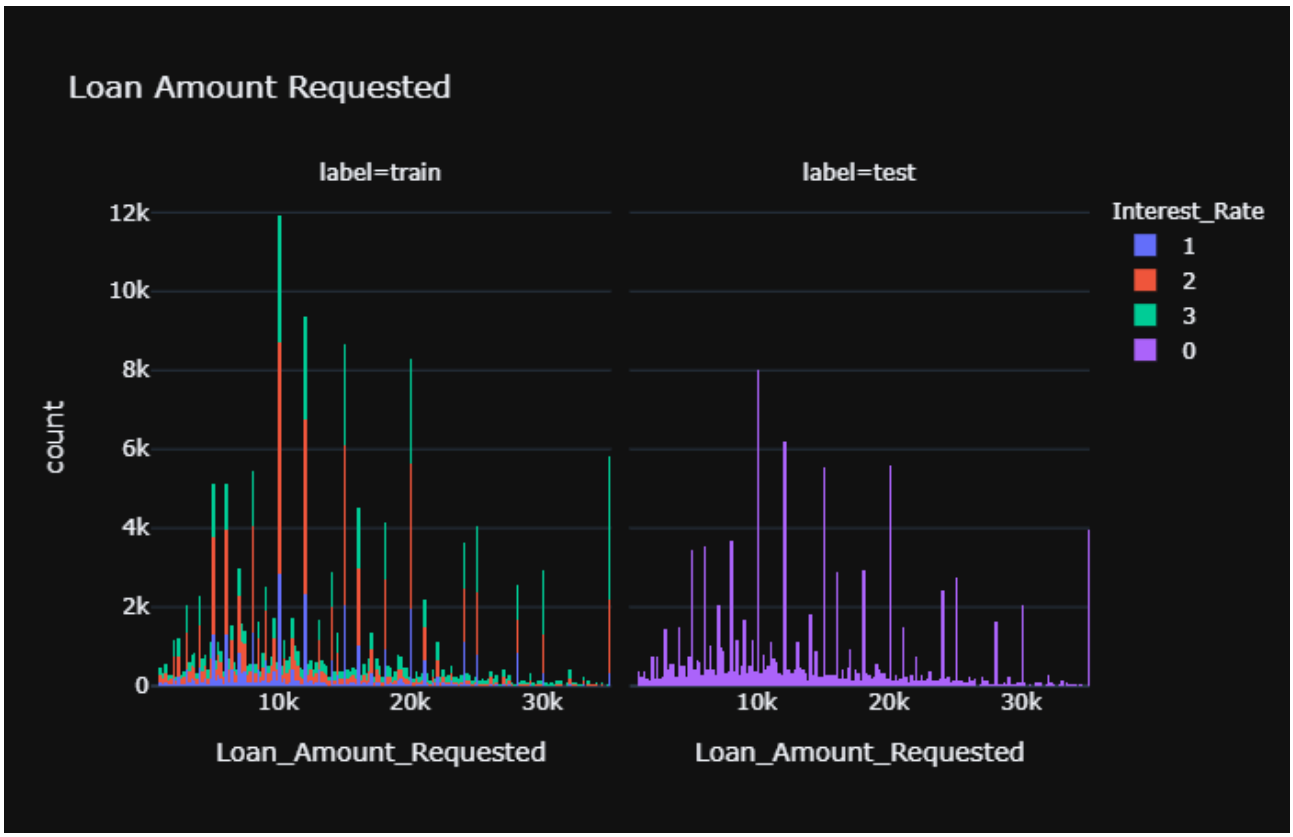
Data Exploration

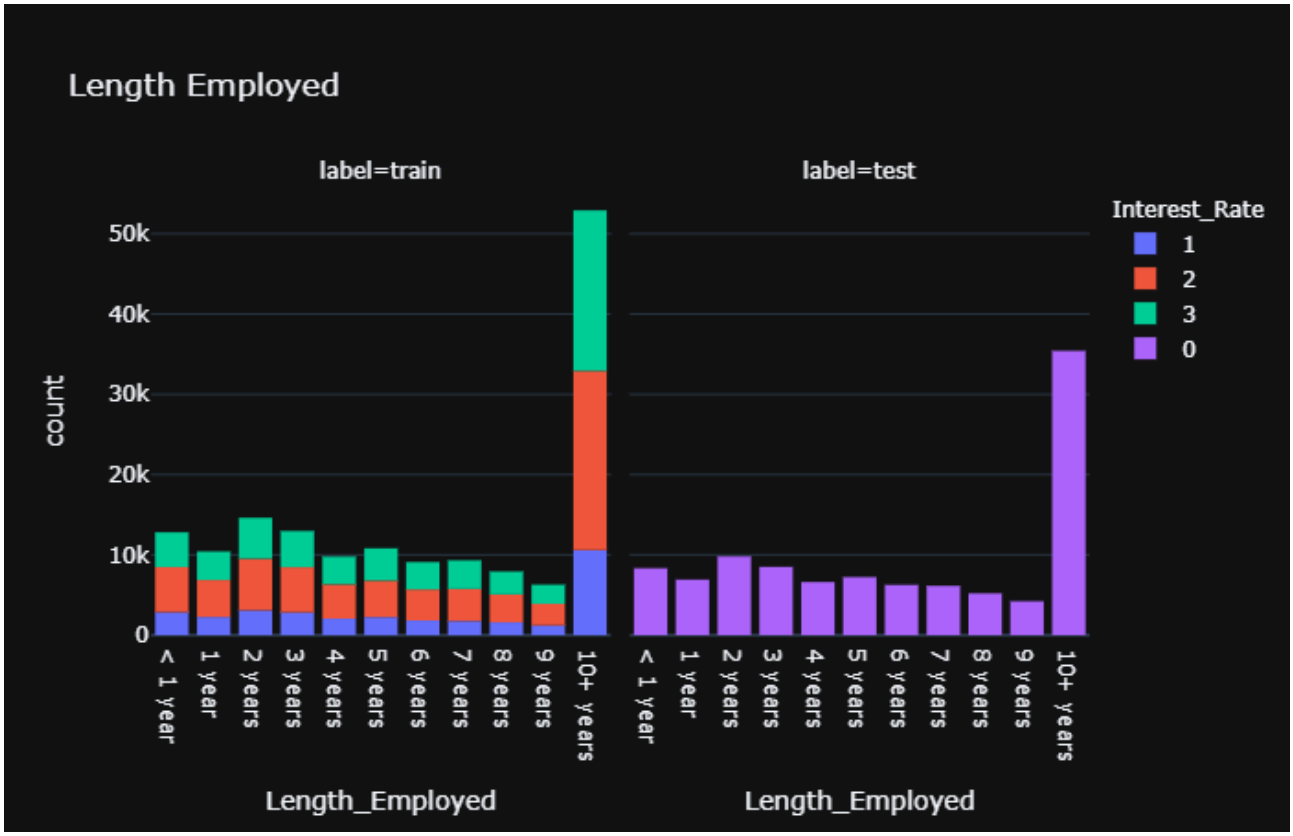
Discoveries

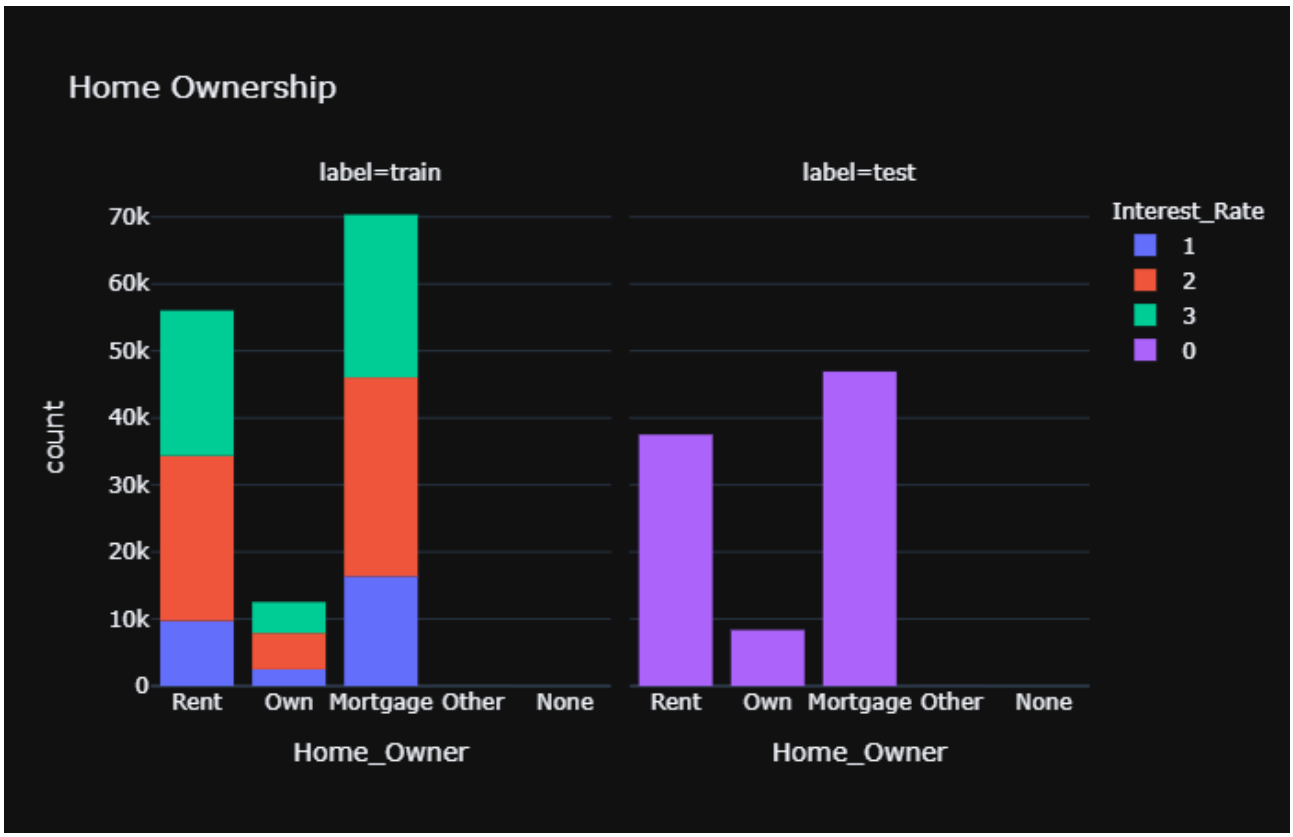
1. Training set and test set have similar distribution. Visualizations are presented in the following section.
2. Surprisingly, training set and test set have similar proportion of missing values and outliers too.
3. There exist super-rich persons (>7 million/year) while the majority have annual income of ~60K/year. That make Annual_Income chart heavily imbalanced and the descriptive statistic is not representative
4. Strangely, Loan_ID has the largest absolute correlation coefficient to the Interest_Rate, which is 0.746107. While the second largest, Inquiries_Last_6Mo has absolute relative coefficient of 0.066536. Loan_ID should not be a predictor since it is only identifier feature but it has largest absolute correlation coefficient.
5. Month_Since_Deliquency is feature with the most missing value (>50%). My assumption is the entity with missing Month_Since_Deliquency have no deliquency record or always paid on time.
6. There are many questions I wanted to ask for missing values in Length_Employed, Home_Owner, and Annual_Income. For example, are missing values in Length_Employed represents “Unemployed” or “I don’t know”. What is exactly case considered as “Other” and “None” values in Home_Owner? Are homeless people allowed to borrow money? And is missing value in Annual_Income represents “Have no income”? Those questions might be best pointed to front liner where the data is collected.
7. Loan_Amount_Requested is misclassified as string data type where it should be numeric. That is because the thousands separator ‘,’ is included in the data.
8. Loan_Amount_Requested is distributed from 1000 to 40000 and that’s similar with Lending Club’s terms.
9. Two most frequent purpose people apply for loan is to consolidate current debt and credit card debt.

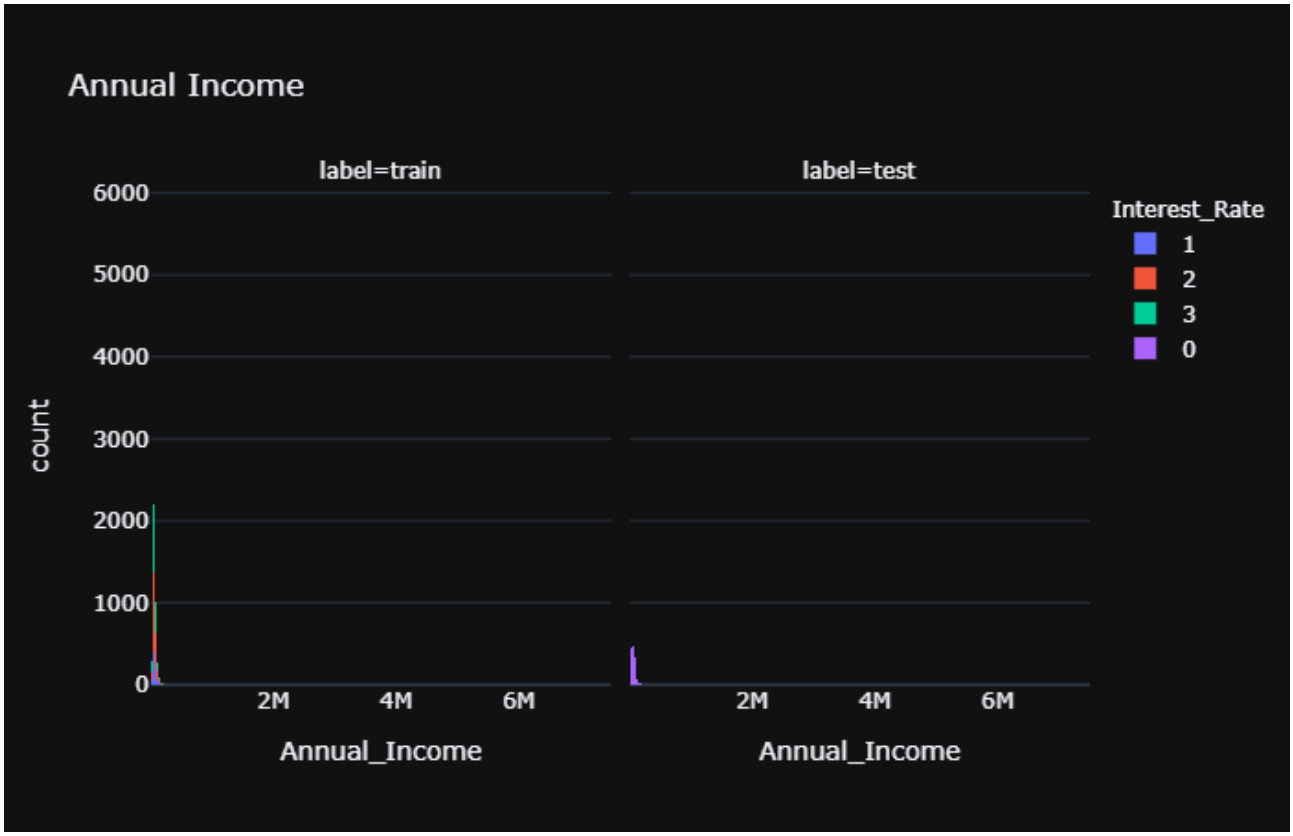
Visualizations

Please note that Interest_Rate = 0 indicates the entities’ Interest_Rate category is not yet known and will be predicted by the model

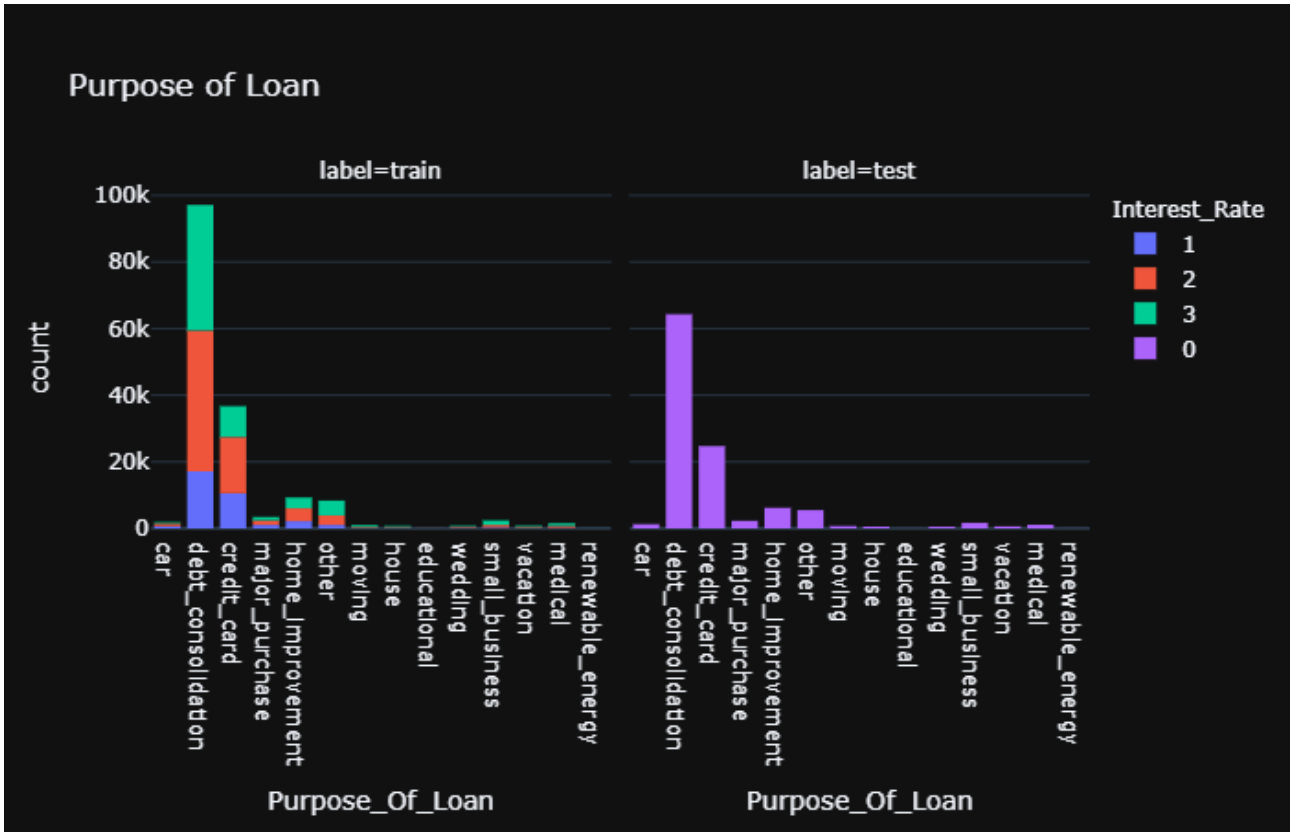

Loan Amount Requested
label=train
label=test
Interest_Rate
1
2
3
0
count
12k
10k
8k
6k
4k
2k
0
10k
20k
30k
Loan_Amount_Requested
Loan_Amount_Requested

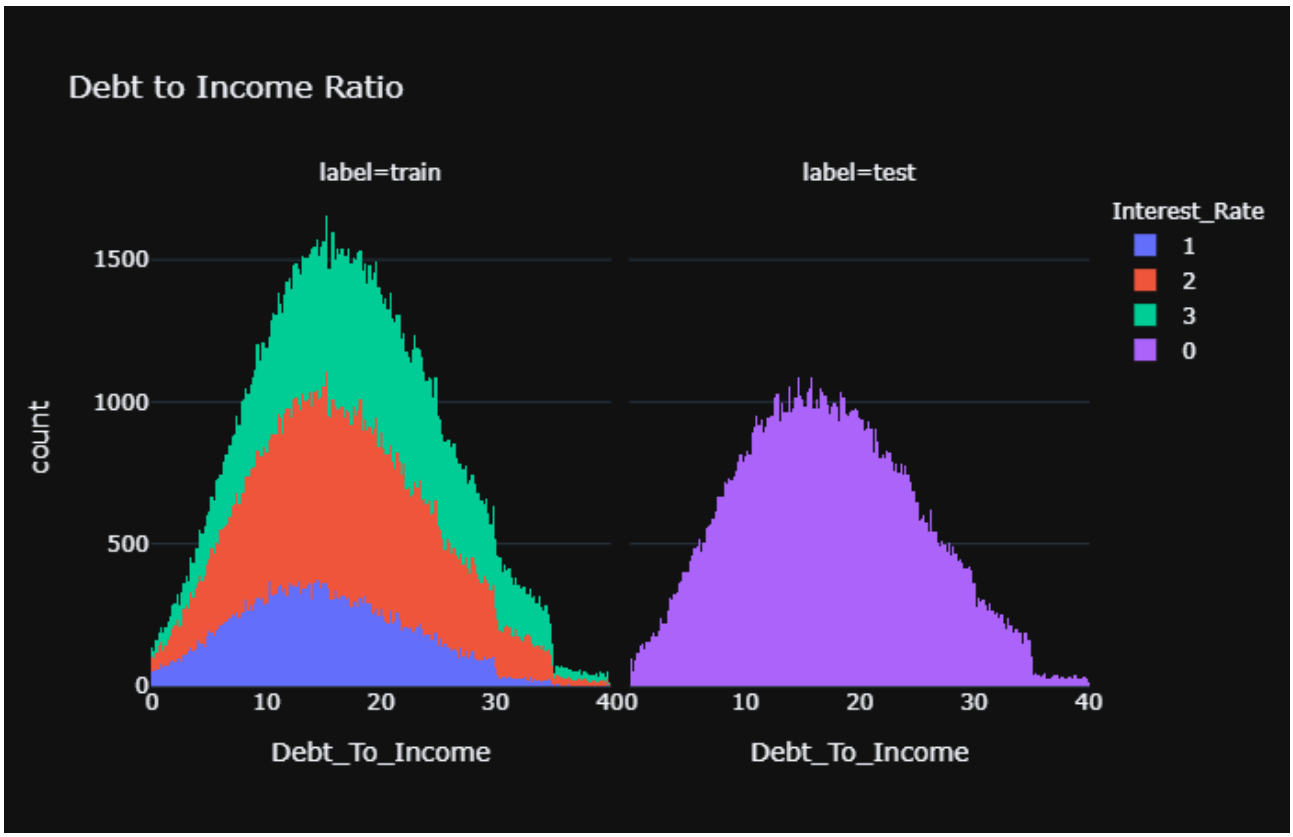

Length Employed
label=train
label=test
Interest_Rate
1
2
3
0
count
50k
40k
30k
20k
10k
0
< 1 year
1 year
2 years
3 years
4 years
5 years
6 years
7 years
8 years
9 years
10+ years
Length_Employed
Length_Employed

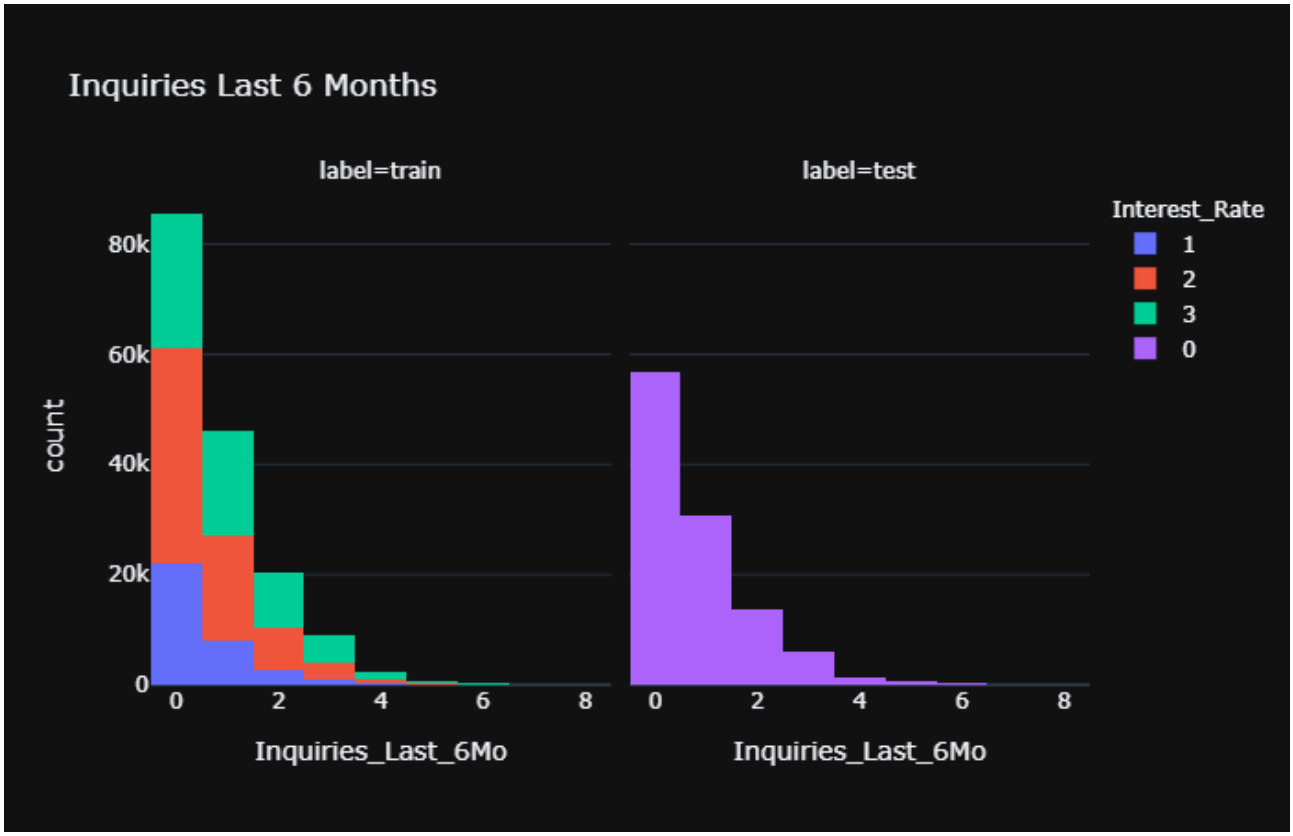

Home Ownership
label=train
label=test
70k
60k
50k
40k
30k
20k
10k
0
count
Rent
Own
Mortgage
Other
None
Home_Owner
Interest_Rate
1
2
3
0

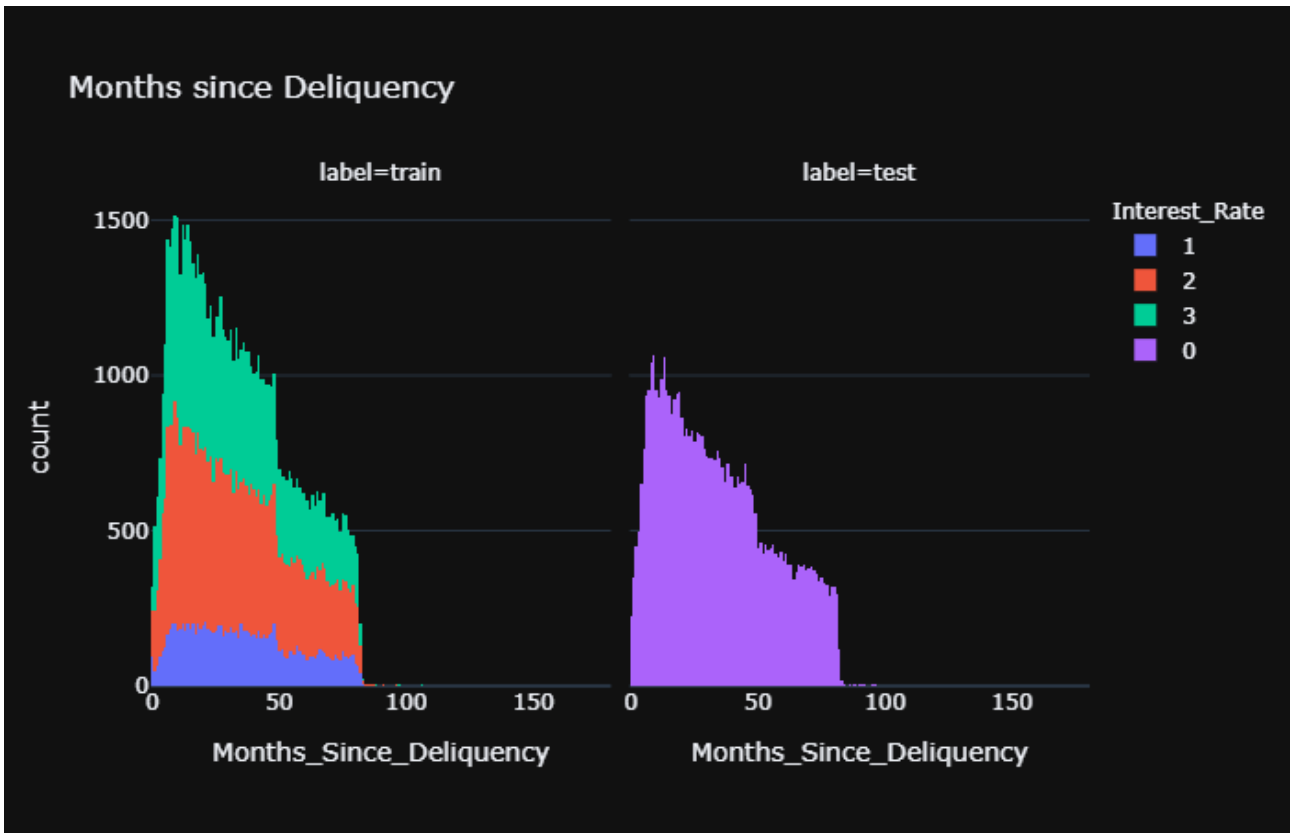

Annual Income
label=train
label=test
6000
5000
4000
3000
2000
1000
0
count
2M
4M
6M
Annual_Income
Interest_Rate
1
2
3
0

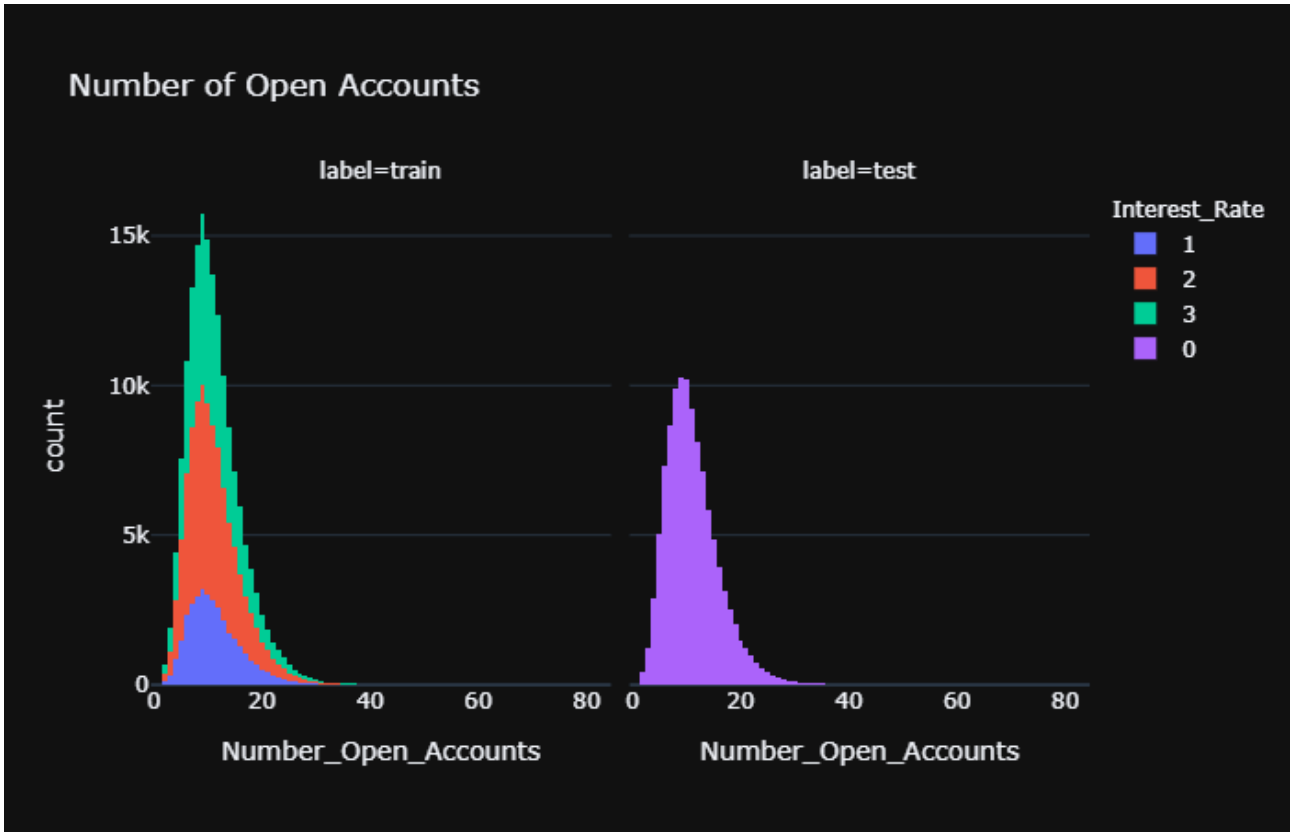

Income Verified
label=train
label=test
60k
50k
40k
30k
20k
10k
0
count
Interest_Rate
1
2
3
0
not verified
VERIFIED - income source
VERIFIED - income
Income_Verified
Income_Verified

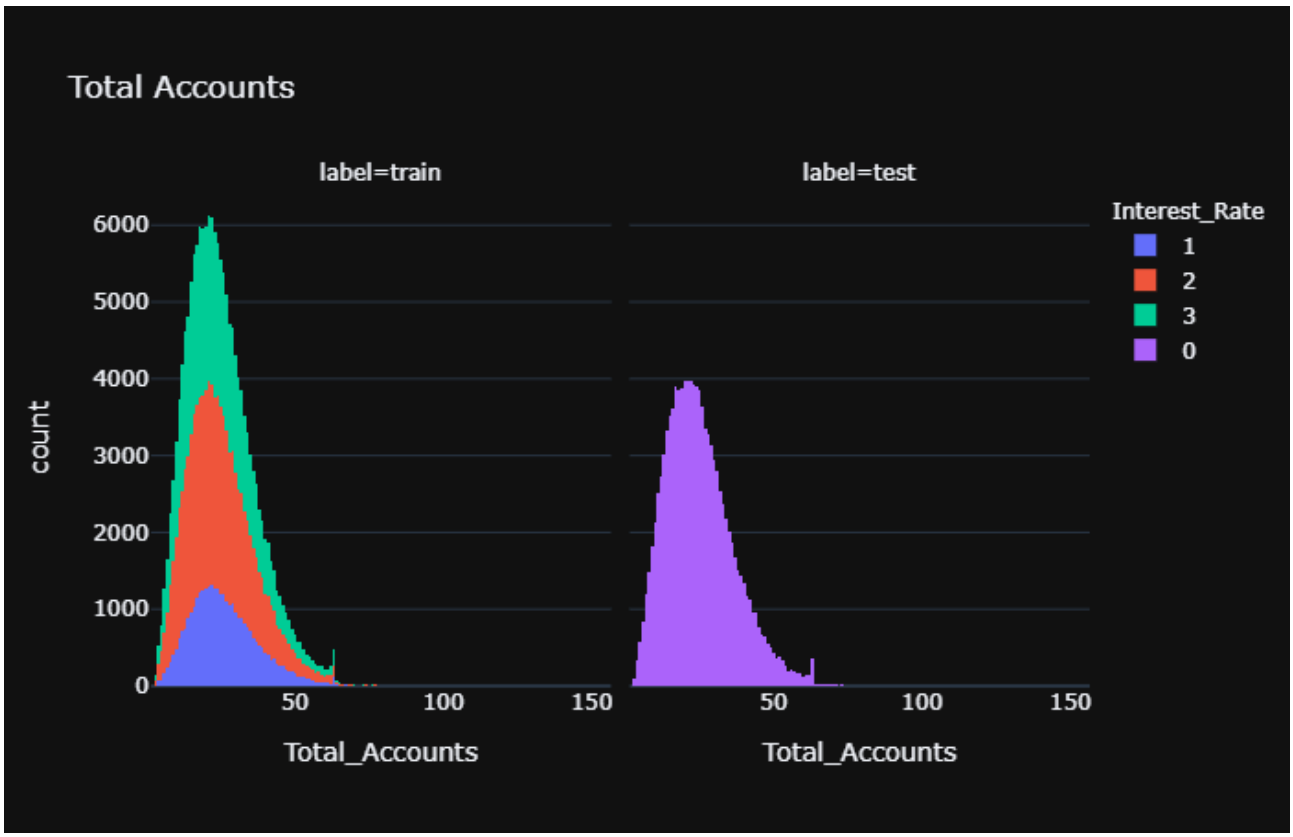

Purpose of Loan
label=train
label=test
100k
80k
60k
40k
20k
0
count
Interest_Rate
1
2
3
0
car
debt_consolidation
credit_card
major_purchase
home_improvement
other
moving
house
educational
wedding
small_business
vacation
medical
renewable_energy
Purpose_Of_Loan
Purpose_Of_Loan

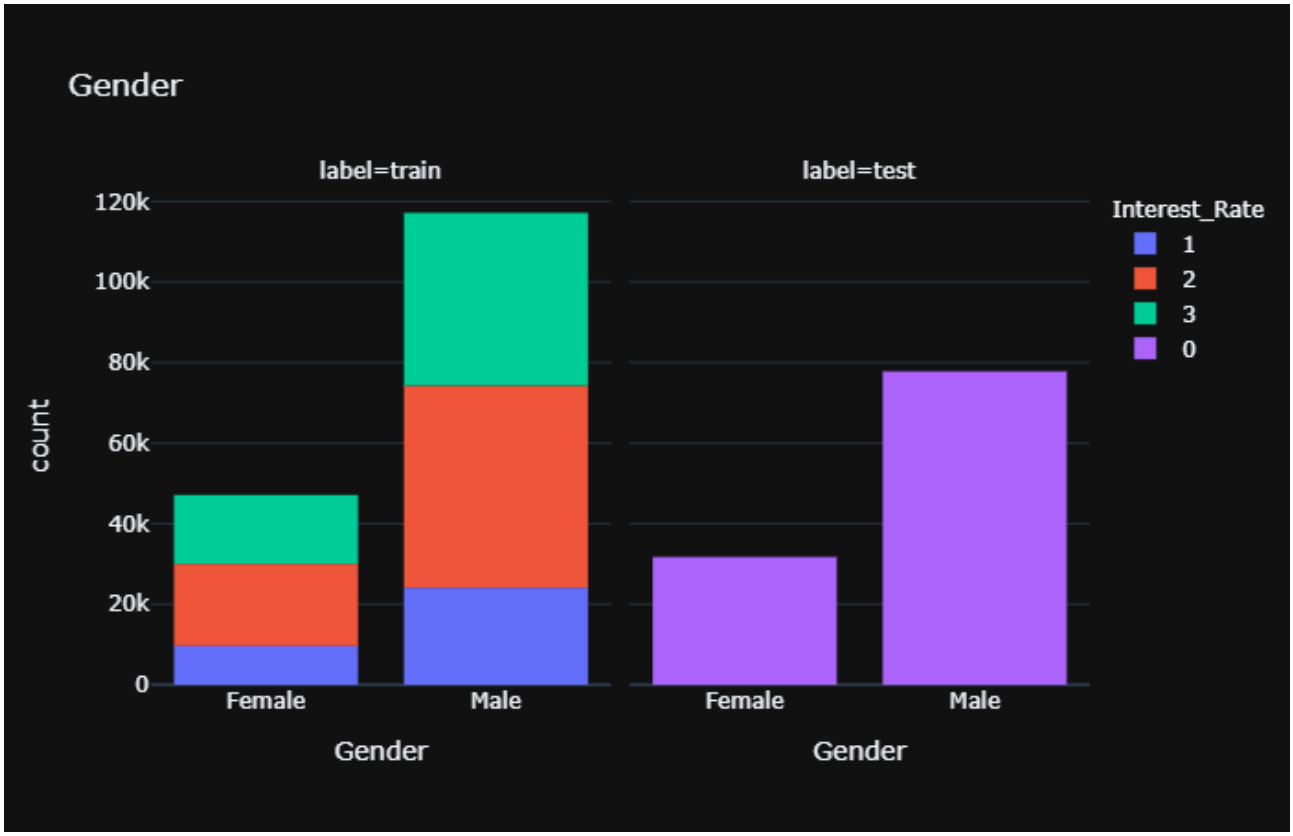

Debt to Income Ratio
label=train
label=test
Interest_Rate
1
2
3
0
count
1500
1000
500
0
0
10
20
30
400
10
20
30
40
Debt_To_Income
Debt_To_Income

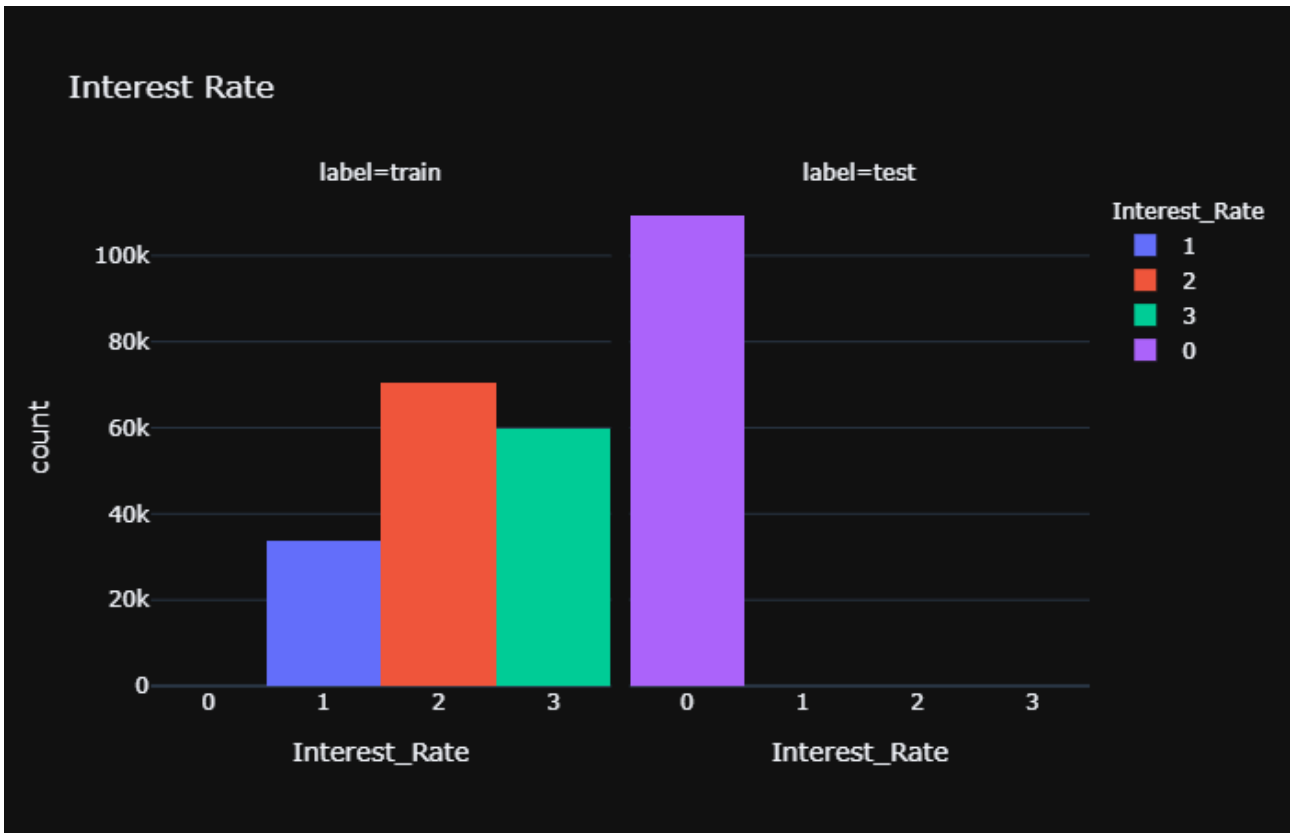

Inquiries Last 6 Months
label=train
label=test
Interest_Rate
1
2
3
0
count
80k
60k
40k
20k
0
0
2
4
6
8
Inquiries_Last_6Mo
Inquiries_Last_6Mo


Months since Deliquency
label=train
label=test
Interest_Rate
1
2
3
0
count
1500
1000
500
0
0
50
100
150
Months_Since_Deliquency
Months_Since_Deliquency


Number of Open Accounts
label=train
label=test
Interest_Rate
1
2
3
0
count
15k
10k
5k
0
0
20
40
60
80
Number_Open_Accounts
Number_Open_Accounts


Total Accounts
label=train
label=test
Interest_Rate
1
2
3
0
6000
5000
4000
3000
2000
1000
0
count
50
100
150
Total_Accounts
Total_Accounts


Gender
label=train
label=test
Interest_Rate
1
2
3
0
120k
100k
80k
60k
40k
20k
0
count
Female
Male
Gender
Female
Male
Gender



Data Preprocessing

Initial State

Number of features: 13
Length : 164309 (train) and 109541 (test)

Workflow

1. Drop Loan_ID since it is a identifier feature
2. Remove ‘,’ in Loan_Amount_Requested then transform the feature into numeric
3. Drop Months_Since_Deliquency because it contains too much missing values
4. Replace missing values in categorical features with “MISSING”. In this case, categorical features that contain missing values are Length_Employed and Home_Owner
5. Replace “< 1 year” in Length_Employed with “less”. It is useful to distinguish with “1 year” because when we use janitor to clean column names, special character like “<” will be omitted.
6. Fill missing values in numeric features with corresponding median
7. I didn’t filter out outliers since the test set has similar pattern for outliers and I don’t want to cancel out any single rows in test set
8. I didn’t rescale the numeric features too because it is not necessary when working with ensemble tree-based algorithms
9. Generate new features manually, here I defined new features

   9.1 Monthly_Income = Annual_Income / 12

9.2 Accounts_Ratio = Number_Open_Accounts / Total_Accounts

9.3 Loan_Income_Ratio = Loan_Amount_Requested / Annual_Income

9.4 Inv_Loan_per_Active_Account = Number_Open_Accounts / Loan_Amount_Requested

9.5 Loan_Per_Total_Account = Loan_Per_Amount_Requested / Total_Accounts

9.6 NMRA = Debt_To_Income x Monthly_Income

9.7 NDTI = NMRA / Monthly_Income

I still can expand the feature synthesis with categorical features combination but it consumes too much RAM in my machine. So I'll stick to this plan

10 One hot encoding to categorical features

11 Export dataset in csv form for future use

Final State

Number of features: 50
Length: 164309 (train), 109541 (test)

Modeling

In this project, I used following models:

1 Stochastic Gradient Descent

2 Ridge

3 Random Forest

4 Gradient Boosting

5 Histogram-Based Gradient Boosting

6 Auto-Sklearn

7 XGBoost

8 LightGBM

9 CatBoost

Since the the target variable consisted of 3 class with proportion approximately 3:7:6 for each interest rate (1,2,3) it is enough to use accuracy score as a metrics. The result is as follow

Model	**Accuracy**
Stochastic Gradient Descent	0.3679
Ridge	0.5120
Random Forest	0.5055
Gradient Boosting	0.5252
Hist-Based Gradient Boosting	0.5285
Auto-Sklearn	0.5266
XGBoost	0.5284

LightGBM	0.5295
CatBoost	0.5325

With the preprocessing technique and feature engineering I did before, the best score for accuracy I have for now is only 0.5325 achieved by CatBoost model. For the next step I'm going to be more focused on three model ie XGBoost, LightGBM, and CatBoost and do hyperparameter tuning. Or maybe I'll try to merge my data with additional features that available online and going back to the data exploration to gain more insight.

Summary

If I'm sure that my dataset is a subset of Lending Club Loan Data, why didn't I merge my dataset with Lending Club Loan Data to improve accuracy? That is because I think domain experts' advice is really needed here e.g. which additional features should I consider, which features to avoid to prevent data leakage that will lead to unrealistic high accuracy.

Reference

Chen, Shuhui, et al. "Credit Risk Prediction in Peer-to-Peer Lending with Ensemble Learning Framework." *2019 Chinese Control And Decision Conference (CCDC)*, 2019, doi:10.1109/ccdc.2019.8832412.

Cohen, Maxime C., et al. "Data-Driven Investment Strategies for Peer-to-Peer Lending: A Case Study for Teaching Data Science." *Big Data*, vol. 6, no. 3, 2018, pp. 191–213., doi:10.1089/big.2018.0092.

Namvar, Anahita, et al. "Credit Risk Prediction in an Imbalanced Social Lending Environment." *International Journal of Computational Intelligence Systems*, vol. 11, no. 1, 2018, p. 925., doi:10.2991/ijcis.11.1.70.

Turiel, J. D., and T. Aste. "Peer-to-Peer Loan Acceptance and Default Prediction with Artificial Intelligence." *Royal Society Open Science*, vol. 7, no. 6, 2020, p. 191649., doi:10.1098/rsos.191649.

"What Affects Your Credit Score and Interest Rate?" *LendingClub*, www.lendingclub.com/loans/resource-center/what-affects-my-credit-score-interest-rate.