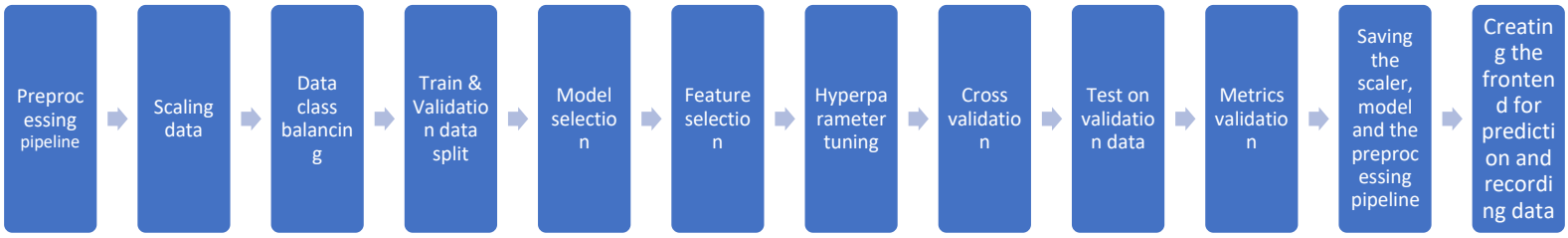
Loan risk prediction and recommendation

A. Libraries used for the project:

1. Pandas
2. Numpy
3. Math
4. Scikit-learn
5. Matplotlib
6. Seaborn
7. Xgboost
8. Imblearn



B. Preprocessing steps:

1. Encoding the categorical columns
2. Deriving the loan type risk percentage field:
    a. Group by each loan type
    b. Find each of their value counts for their respective loan status.
    c. Divide each status counts by the total population.
    d. Now we have estimated risks for each loan type which can be joined with the main data frame:

	fully_paid	charged_off	count	count_pct
Business Loan	68.114818	31.885182	1289	1.841429
Personal Loan	77.152775	22.847225	56462	80.660000
other	78.503359	21.496641	6401	9.144286
shopping loan	78.542510	21.457490	247	0.352857
Education Loan	79.710145	20.289855	69	0.098571
Home Loan	80.017528	19.982472	4564	6.520000
wedding loan	80.459770	19.540230	87	0.124286
Vehicle Loan	83.654938	16.345062	881	1.258571

3. Loan-to-income ratio is derived:

$$Loan\ to\ income\ ratio = \frac{Current\ loan\ amount}{Household\ income}$$

4. Create field called 'excessive loan':

$$Excessive\ loan = \begin{cases} 1, & loan\ to\ income\ ratio\ >=\ 0.4 \\ 0, & loan\ to\ income\ ratio\ <\ 0.4 \end{cases}$$

5. Fill the null excessive loan field values with 0.
6. Calculate monthly debt to income ratio:

$$Monthly\ debt\ to\ income\ ratio = \frac{Current\ loan\ amount}{Household\ income\ /\ 12}$$

7. Create field called 'excessive debt':

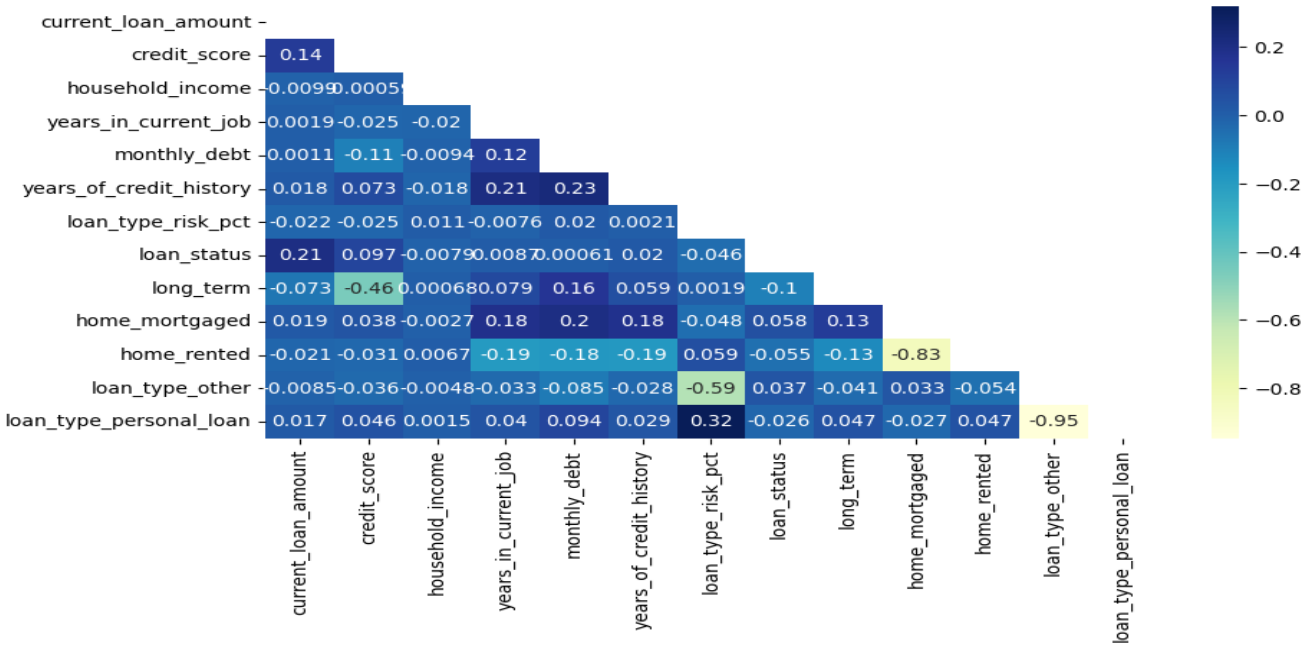
$$Excessive\ debt = \begin{cases} 1, & monthly\ debt\ to\ income\ ratio\ >=\ 0.36 \\ 0, & monthly\ debt\ to\ income\ ratio\ <\ 0.36 \end{cases}$$

8. Fill the null excessive debt field values with 0.
9. The threshlod value of debt to income ratio discussed above is taken from here.
10. The threshold of loan to income ratio has been choosen by rounding off and tuning percentage values around the standard debt to income ratio value from above.
11. Derive bad financial condition field:

$$Bad\ financial\ condition\ =\ Excessive\ loan\ +\ Excessive\ debt$$

12. Perform log10 and exponential transformation of the non-binary fields.

Now we have a better correlation matrix with decent collinearity for all the non-transformed fields:

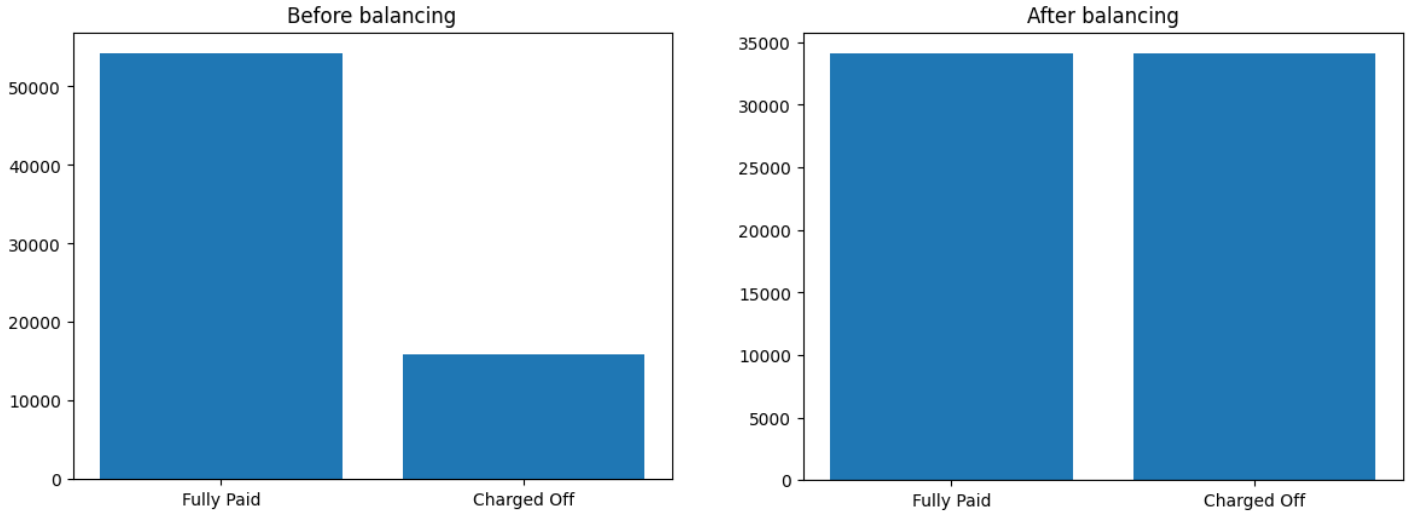


13. Performed Recursive Feature Elimination (RFE) performed to get to the most important fields and obtained the following fields:

```
'current_loan_amount', 'credit_score', 'years_in_current_job', 'years_of_credit_history',
'loan_type_risk_pct', 'loan_to_income_ratio', 'bad_financial_condition', 'long_term',
'home_mortgaged', 'loan_type_other', 'social_class_middle_class', 'current_loan_amount_log',
'current_loan_amount_exp', 'credit_score_log', 'credit_score_exp', 'household_income_exp',
'years_in_current_job_log', 'years_in_current_job_exp', 'monthly_debt_log',
'monthly_debt_exp', 'years_of_credit_history_log', 'years_of_credit_history_exp',
'loan_type_risk_pct_log', 'loan_type_risk_pct_exp', 'loan_to_income_ratio_log',
'loan_to_income_ratio_exp']
```

C. Training steps and discussions:

1. Min-Max scaling is performed.
2. SMOTE oversampling is performed.
3. KFold 5 folds cross validation performed.
4. We have data augmented to have complete balanced data. Imbalanced data visualization:



5. Train dataset split at 95%
6. Model performance, cross validation scores:

accuracy	f1	recall	specificity	precision	roc_auc
0.859905	0.863943	0.889565	0.830295	0.839796	0.934516

Validation:

1. Both sensitivity and specificity are considered since organization can neither afford to sanction loan to a risky customer nor lose a potentially good and profit worthy customer.
2. Therefore, the validation dataset must be balanced one too.
3. Validation scores:

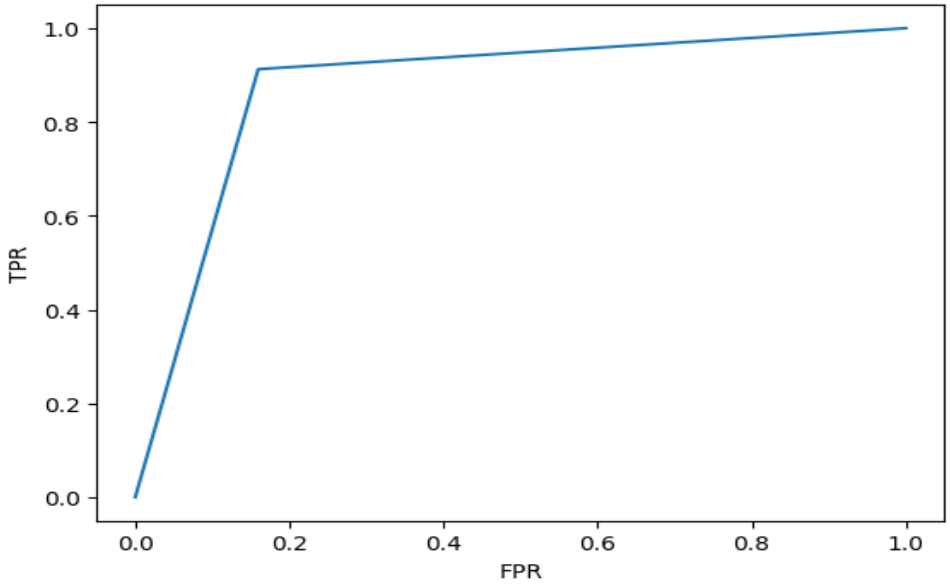
accuracy	f1	recall	specificity	precision	roc_auc
0.877383	0.842136	0.91261	0.842136	0.852603	0.877373

4. Overfitting check:

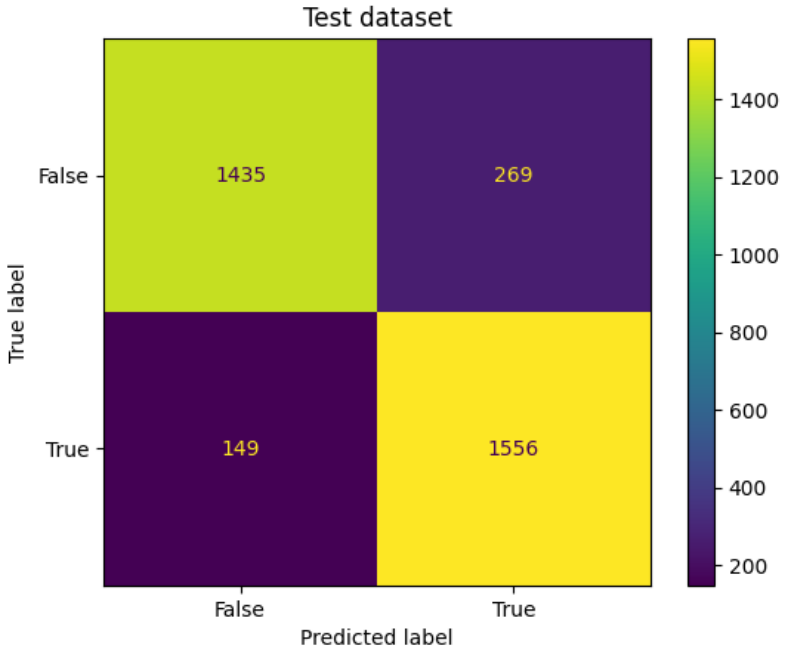
accuracy	f1	recall	specificity	precision	roc_auc	Average difference
0.017479	0.021806	0.023045	0.011841	0.012807	0.057143	0.024020063

5. No overfitting, with descent bias, precision and generalization obtained.

6. ROC Curve:

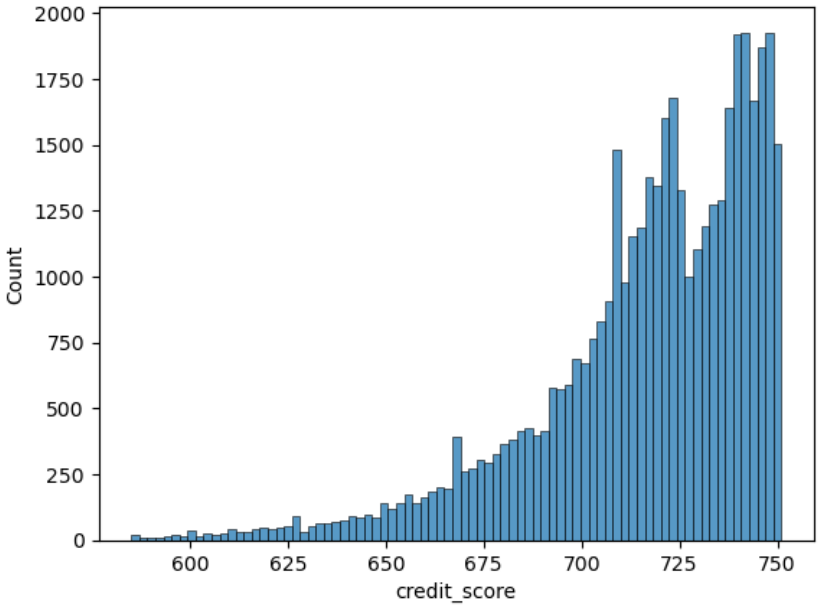


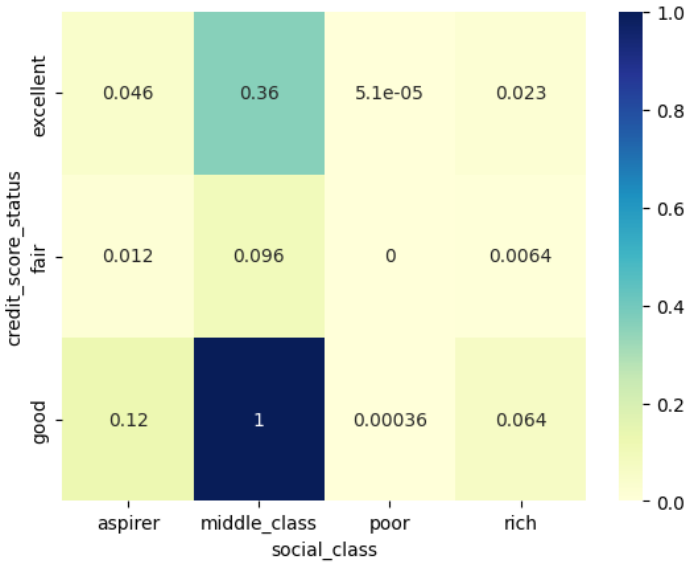
7. Validation dataset confusion matrix:

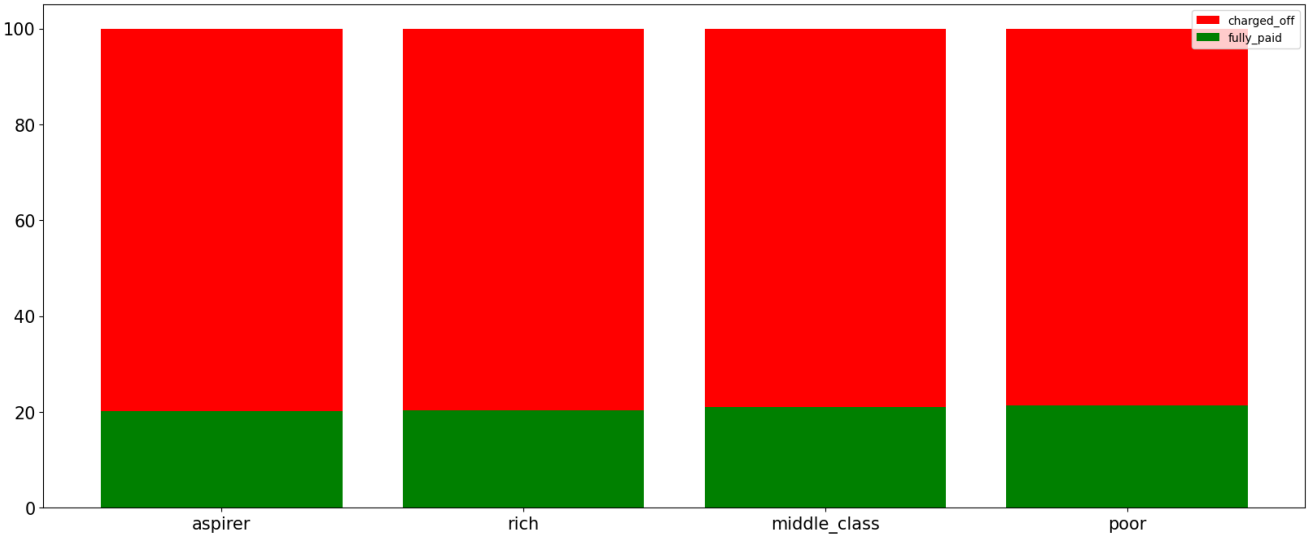


8. Model used: XGBoost
9. Model parameters: **{'objective': 'binary:logistic', 'base_score': None, 'booster': None, 'callbacks': None, 'colsample_bylevel': None, 'colsample_bynode': None, 'colsample_bytree': None, 'device': None, 'early_stopping_rounds': None, 'enable_categorical': False, 'eval_metric': None, 'feature_types': None, 'gamma': 0.01, 'grow_policy': None, 'importance_type': None, 'interaction_constraints': None, 'learning_rate': None, 'max_bin': None, 'max_cat_threshold': None, 'max_cat_to_onehot': None, 'max_delta_step': None, 'max_depth': 30, 'max_leaves': None, 'min_child_weight': None, 'missing': nan, 'monotone_constraints': None, 'multi_strategy': None, 'n_estimators': 300, 'n_jobs': None, 'num_parallel_tree': None, 'random_state': None, 'reg_alpha': None, 'reg_lambda': None, 'sampling_method': None, 'scale_pos_weight': None, 'subsample': 0.8, 'tree_method': None, 'validate_parameters': None, 'verbosity': None, 'eta': 0.08}**

D. A few points on EDA:







We can deduce the following from the above graphs:

1. Around 75% of people have either a good or excellent credit score.
2. Middle class customers maintain their credit scores at the best.
3. Business loans are most risky and have more chance to get charged off than other loan types.
4. No significant difference in the chances of repaying the loan based on social status.

E. Notes regarding approach:

1. Banks never recommend any loan products apart from personal loans without any considerable information.
2. If a customer seems to be of low risk and of potential profit, the bank will recommend a personal loan until the bank has relevant data to offer any other loan.
3. There is no data in the recommender dataset that specifies the eligibility of a particular loan.
4. Loan products are not recommendable. Customers take loans based on requirements but not choices or desires.
5. The data set gives some information about the customers who opted for a particular loan. It does not have any information that says why they took that loan from a business and technical point of view.
6. The dataset belongs to Spain. The financial lifestyle and standard may not match with the loan risk dataset which does not have any country information (possibly global).
7. Recommender dataset is possible not an actual recommendation dataset. Since for every customer code, there has been no more than 1 loan type recommendation. And it is very unusual not to recommend multiple potential options to a customer. Therefore, it's mostly just the customers and their details who opted for a particular loan and not recommendations.
8. Therefore, the approach taken is to detect the customer's risk based on the query done and validate the customer's risk with the same loan risk model for all the other different available loan types and recommend all the loan types for which the customer will be able to pay, i.e., low risk is predicted.