Capstone Project: Loan Default Classification

1. Executive Summary

This project proposes the Tuned Random Forest model for the prediction of home equity loan defaulters for a bank based on various features of the potential borrowers. The suggested model gives a good and balanced prediction performance on accuracy, recall, and precision, and thus, is reasonably good to be deployed as an additional reference tool in the loan approval decision making process.

Proposed model specification:
RandomForestClassifier(class_weight='balanced', criterion='entropy', max_depth=7, max_features=0.9, min_samples_leaf=20, n_estimators=500)

The model gives a balanced scores (accuracy > 85%, recall ~75%, precision ~65%) and is not overfitting. The model is also interpretable in aggregate as shown in the model feature importance analysis. It is suggested that important factors that increase the chance of default include borrowers not providing any income-to-debt ratio information, high income-to-debt ratio, high numbers of delinquency credit lines, high numbers of major derogatory reports, etc. However, the model is subjected to a number of limitations, including the lack of complete interpretable decision rules for each individual borrower that can be clearly presented to customers and the regulators, the lack of consideration of other features that may also impact the defaulting chances, and the lack of consideration of changes of the features' impact on default over time.

The suggested general guidance based on the model result is that the Bank should not approve loans to those customers who check multiple boxes in the below list of characteristics as they have high chances of defaulting:

- If the potential customers do not provide debt-to-income ratio
- If customer's debt-to-income ratio is greater than 50%
- If customer has three or more delinquent credit lines
- If customer has two or more major derogatory reports

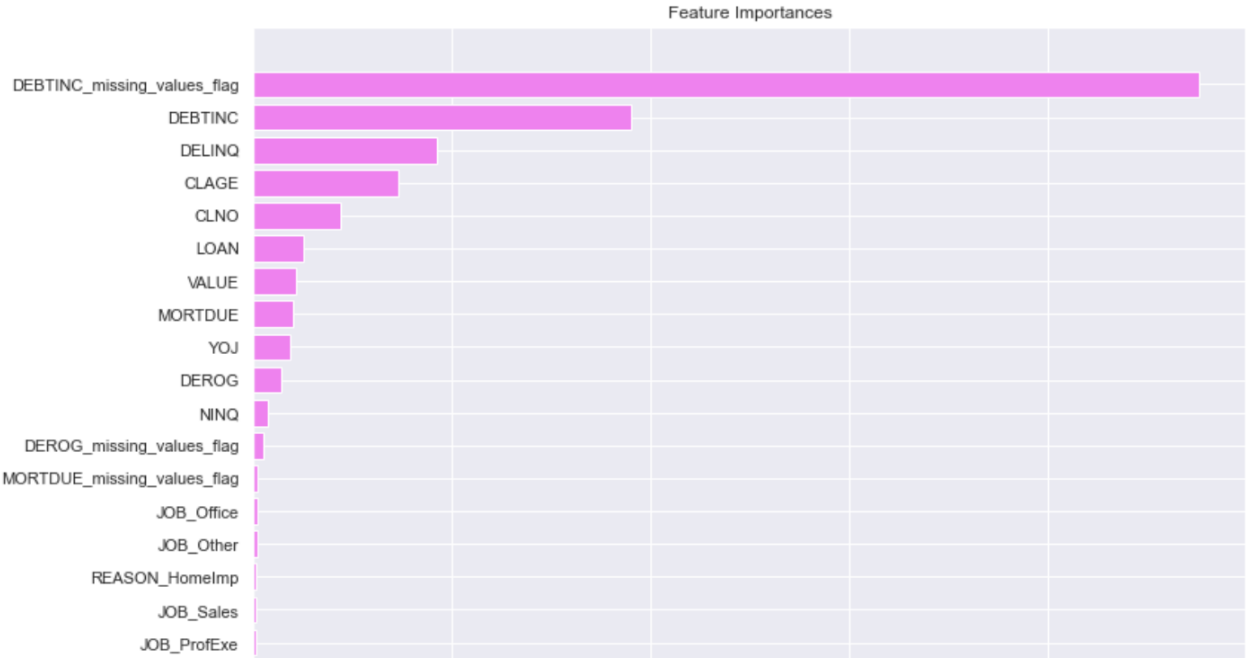

Figure 1: Tuned Random Forest Feature Importance

By using the model as an assisting tool, the bank will enjoy many benefits including higher loan approval accuracy by reducing human errors and biases, higher loan approval efficiencies, lower default losses, bigger market shares and higher interest revenues from serving more good customers, higher customer satisfaction levels, and potentially lower labor costs from reducing junior loan review officer headcounts.

The management is suggested to deploy the model and the Data Science team and the consumer credit department should work together to come up with policies and procedure manuals for using the model in the loan application review, approval, appeal, and communication processes. The Data Science team should also continuously monitor the model performance and enhance the model over time.

2. Problem and Solution Summary

2.1 Problem Summary

To supplement the previous purely manual loan approval process, it is important for the bank to simplify and partially automate the decision-making process for home equity lines of credit to 1) achieve better business results including maximizing revenues and minimizing default losses, 2) reducing the loan approval lead time, 3) increasing the efficiencies and reducing consumer credit department FTEs.

To achieve the above business objective, we need to come up with an explainable Machine Learning classification model to predict potential clients who will and will not default on their loans based on application data provided, and provide recommendations to the bank on the important features to consider while approving a loan.

2.2 Solution Summary

2.2.1 Solution Design

The overall process of analysis is as below:

- EDA, treating outliers and imputing missing values: we performed this to understand data distribution as well as bi-variate relationships between the target feature and the rest. Also, by treating the outliers and imputing missing values, the data is more clean and ready to be used in modeling.
- Feature engineering, mainly creating dummy variables for categorical data and adding binary missing value flags as new features. We performed this step to make sure the models can process the categorical data, and also we believe the fact of certain data is missing can provides certain predictive value.
- Separate the data into target and features, then, further split the data into train and test sets. We use the 20% test dataset to evaluate the model performances.
- Build the prediction models using the various classification algorithms and fit the models with the training data.
- Evaluate each model’s performance on both train and test sets using the recall, precision, and accuracy scores.
    - Model can make wrong predictions in the following ways: 1) Predicting a customer will default and the customer doesn't default in reality (False Positive); 2) Predicting a customer will not default and the customer defaults in reality (False Negative).

    - Considering the business costs of False Negative errors are the highest, the Class 1 recall score is the most important metric. However, the precision is also considered in the overall evaluation.
- GridSearchCV algorithm is used in hyperparameter tuning to locate the best hyperparameter combinations for Decision Tree and Random Forest models to achieve the best performance for each algorithm.
- Finally, compare all the above models' performances and pick the best model balancing various evaluation factors.

The classification algorithms used include Logistic Regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Decision Tree, Tuned Decision Tree, Random Forest, and Tuned Random Forest.

2.2.2 Analysis and Key Insights

EDA – Univariate Analysis Observations:

- Many numerical variables have right-skewed distributions with outliers on the right-hand side. Those include "LOAN", "MORTDUE", "VALUE", "YOJ", "DEROG", "CLNO", etc.
- For "DEROG' and "DELINQ", the vast majority of observations has a zero value, while a small amount has very large numbers, indicating their low credit levels.

Based on the combined results from EDA and feature importance analysis of various models, these below factors appear to be important in identifying loan defaulters.

- DEBTINC_missing_values_flag: If a borrower did not provide the debt-to-income ratio information at all, it is a strong indicator that the borrower is more likely to default. It makes sense since a borrower with low DEBTINC would have the incentive to provide the information; thus, on the other hand, if this information is not provided, it is likely the borrower has a very high debt-to-income ratio.
- 'DEBTINC' appears to be a significant factor that impacts the default chance: From EDA, for those borrowers who have the ratio over 50%, they are indeed all defaulted borrowers. In the Decision Tree model, the cut-off value for DEBTINC is about 43%, indicating a borrower with a debt-to-income ratio of over 43% is more likely to default.
- DELINQ' appears to be a significant factor that impacts the default chance: In general, the borrowers who defaulted have higher numbers of delinquent credit lines – for people with 3 or more delinquent lines, it is over 50% chance they will default. On the other hand, the vast majority of the borrowers who did not default have zero major derogatory report. Based on the decision tree model, if the DELINQ is larger than 2.5, it is very likely the borrower will default.
- 'DEROG' appears to be another factor that impacts the default chance: In general, the borrowers who defaulted have higher numbers of major derogatory reports – for people with 2 or more derogatory reports, it is over 50% chance they will default. On the other hand, the vast majority of the borrowers who did not default have zero major derogatory report.
- Job type has some impact on the default: Borrowers with Sales job type has the highest default rate among all job groups, followed by "Self"; Borrowers with Office and ProfExe jobs have lowest default rates.

Model Performance Comparison:

	Model	Train_Accuracy	Test_Accuracy	Train_Recall	Test_Recall	Train_Precision	Test_Precision
0	Logistic Regression	0.889022	0.892617	0.615385	0.607843	0.781679	0.806691
1	LDA	0.885427	0.885347	0.669471	0.649860	0.732895	0.743590
2	QDA	0.853068	0.850112	0.561298	0.551821	0.653147	0.645902
3	Decision Tree	1.000000	0.880313	1.000000	0.655462	1.000000	0.720000
4	Tuned Decision Tree	0.823826	0.820470	0.843750	0.792717	0.537108	0.533962
5	Random Forest	1.000000	0.911633	1.000000	0.703081	1.000000	0.828383
6	Random Forest with Class Weight Specified	0.999760	0.912192	0.998798	0.691877	1.000000	0.840136
7	Tuned Random Forest	0.883030	0.865772	0.818510	0.742297	0.668959	0.641646

Figure 2: Model Performance Metrics

- Overall, the best performing model that is recommended by this paper is the ***Tuned Random Forest model***, balancing all evaluation factors: accuracy > 85%, recall ~75%, precision ~65%. The Test precision score is close to 65% which means the bank would lose less legitimate customers compared to the Tuned Decision Tree model. Also, the ***Tuned Random Forest model*** does not have significant overfitting issues as the difference between performance metrics on training data and test data is not significant.
    - Based on the factor importance chart, these are the key factors that impact the predicted decision: DEBTINC_missing_values_flag, DEBTINC, DELINQ, CLAGE. Again, this is consistent with the EDA findings and it provides a certain level of interpretability of the model.
- The ***Logistic Regression*** and ***LDA*** models have similar levels of performance, with recall scores lower than 70%. The ***QDA*** model is not performing well as the recall score is below 60%. They should not be chosen due to the unsatisfactory recalls.
- The problem of the ***untuned Decision Tree model*** and the ***two untuned Random Forest models*** is that they are overfitting. They all perform perfectly on the training dataset, but the performances on the test dataset are much worse. Comparing these two types of untuned models, the ***Random Forest models*** perform better on the test set than the ***Decision Tree model***. It makes sense as a Random Forest model is an ensembled learning model and should improve the performance of a single Tree.
- The ***Tuned Decision Tree*** is not overfitting anymore as expected, and the test recall score is over 80% which is quite high. However, its test precision score is only over 50%, meaning we would have turned down too many good customers incorrectly. Although the ***Tuned Decision Tree*** can be visualized and explained to stakeholders easily, it is still not a satisfactory model at this stage due to low precision score.

Impacts on business:

- By correctly predicting the loan defaulters, the bank can reduce its Non-Performing Assets. Thus, the bank can reduce default losses and maximize profits.
- By correctly identifying the good customers who will not default, the bank can extend more loans to good customers (and turn down fewer good customers incorrectly) and as a result, obtains bigger market share, loan size and interest revenues.
- By using the ML model in assisting the loan approval process, the average loan process time should decrease.

    - That will increase customer satisfactions and give the bank a competitive advantage in the market with its speed, further enhancing the banks' reputation and market shares.
    - The increased efficiency may also decrease the number of employees (especially at junior level) needed in the loan review/approval department. This can help the bank save labor costs.
- The ML model provides certain interpretability with the important factors identified.
    - Thus, this information can assist internal communication between loan approval teams, management, and sales teams so that everyone can be at the same page on why a certain customers are approved or rejected.
    - In addition, such information can be leveraged by the loan sales team to explain the banks' decisions to a certain extent to their potential customers, or to advise their potential customers on actions that can increase their chances of obtaining the loans in the future. That will increase customer satisfaction and future revenues.
    - Furthermore, such information can be used by the bank's marketing and sales team in coming up with better marketing initiatives or outbound lead generation activities targeting the right segments of potential customers who are more likely to be approved. That will increase the conversion rate and eventually interest revenues.
- However, the implementation of the model may face certain resistance from the bank employees due to reasons such as they do not fully understand the model or they do not want to change the way of doing business. Especially, if the headcounts of loan review department were to be reduced as a result, it could trigger negative sentiment and needs to be handled delicately by the management.

2.2.3 Limitations and Recommendations for Further Analysis

Key limitations include the fact that even the best model is not 100% accurate. This may be due to the existence of latent variables or we are not yet using the most suitable model or hyperparameters. The Tuned Random Forest Model will generate both False Negative and False Positive errors, meaning the bank will make loans to actual defaulters and also turn down some good customers, respectively. Another limitation is that the Tuned Random Forest Model is not fully interpretable at individual customer level. Thus, although certain factors can be identified and communicated to customers in general, for every individual potential client that get rejected by the model, there is not an exact set of rules that can rigorously explain the rejection fully.

Recommended areas for further analysis:

- As there could be latent variables that impact the defaulting chance, we may consider to add more variables by obtaining more data such as total income, co-applicant, education, property area, etc.; Also, we can use feature engineering to come up with new features and test if those features are significant.
- Further hyperparameter tuning on the existing models. We also try other hyperparameter tuning algorithms than GridSearchCV such as Optuna.
- Increase the training data size
- Try other algorithms such as KNN and neural network and see if those algorithms can provide better performances. If Neural Network if used, we can use SHAP framework to obtain certain level of interpretability of the model and explain to stakeholders.

- Try Boosting algorithms to see if we can get better results. For example, we can try AdaBoosting, XGBoosting, GradientBoosting
- Evaluate the performance over time to ensure the model is still valid with changes in economic and social environments

3. Recommendations for Implementation

3.1 Key action steps for stakeholders

- It is suggested to use the model as a supplemental tool and to integrate the model prediction as one factor in the loan review process. Due to the limitations of the model, ***the consumer credit department and management*** should not fully rely on the model to make approval and rejection decisions, but to use it as an additional reference data point. The management should come up with a clear guidance and operating manual on the model-assisted loan review, approval, and escalation/appeal process.
- The bank's ***management and Legal & Compliance departments*** should review the proposed policies and new workflow in light of the legal and regulation requirements to ensure the use of the model is proper and compliant.
- The ***Data Science team*** should conduct presentations and training sessions on the model to broader stakeholders, including the consumer credit loan review teams, compliance, sales, etc. The goal is for loan review team to understand how the model can be used as a supplemental tool by loan review team in their day-to-day work.
- Based on the factors suggested by the model, the ***consumer credit department*** should create a general communication internal and external guidance on the decision rules.
- The ***management*** should work together to create a communication plan and guidance that can be passed on to the ***sales team,*** for them to properly communicate any necessary information post-decision with the clients. The data science team should also host some training sessions with the sales team to make sure they understand the benefits and clear any doubts they might have.
- The ***Data Science team*** of the bank should continuously work closely with the front line and loan approving department to evaluate the performance of the model, and to come up with model enhancements periodically.

3.2 Expected benefits:

Increased loan review and approval efficiencies, faster decisions which will lead to better customer experience and securing good customers due to the speed of approval; Reduce manual approval work load, reducing junior loan reviewer FTE.

Increase the loan decision quality, meaning reducing both False Negative and False Positive errors compared with previous pure human review approach.

- Reduced False Negative leads to less default loss: the model will identify 75% (i.e., class 1 recall) of all the true defaulters and reject their application. This should have been an increase compared to the previous pure human approval results (e.g., assuming 50%), and the difference of 25% would result in avoiding default losses of the loan amounts.
- Reduced False Positive leads to less lost revenue from good customers: among all the true good customers who would not default, the model will be able to identify over 90% (i.e., class 0 recall) of them as non-defaulters and thus results in doing

business with them and making interest incomes. Equivalently, the model has a class 1 precision of 70%, meaning the model is only making 30% mistakes in all rejected customers which are lost revenues. Pre-model, if the bank is trying to be prudent and minimize default, they might be turning down a lot more business from potential good customers.

Increased fairness among all applicants as the rule-based model reduce the risk of unequal treatment of different applicants by human approvers based on features that may introduce bias and discrimination (e.g., race, color, religion, sex, etc.).

3.3 Expected costs, risks and challenges

- Implementation, training and model maintenance costs:
  Those are explicit costs are mainly human resources costs in the data science department who will own the model; training time will be also required from loan review and sales department.
- Model performance related costs:
  The factors impacting the default may change overtime, the model is a static model if we do not update the training dataset with new features periodically. And as a result, the performance may determinate over time. Even the features remain the same, their impact on the defaulting prediction may change (model structural break) after certain event such as financial crisis and Covid. The two
  - Potential incremental default loss (False Negative) due to the model uncertainty: The model is generating False Negative as of now and those will lead to default loss; furthermore, if there is a structural change and the model performance determinate over time, there could be more than expected default loss realized. Also, the model performance by nature has certain randomness, meaning there could be losses due to the random errors produced by the model.
  - Lost revenues (False Positive) due to the model uncertainty: same as the above reason, the model is generating False Positive errors and may generate increased level of such error in the future and lead to lost revenues.
- Potential implicit HR related costs due to ineffective implementation of and/or communication around the new ML approach: Internal resistance and misunderstanding may rise if the implementation is not handled well, and that could lead to lower employee moral and higher turnovers.
- There are also potential regulation risks of using the model if not fully explainable.

3.5 Suggested further analysis to be done

- Explore other variables
- Post-transaction evaluation to study the error cases (False Negative) made by the model to gain insights: to study the cases that are approved by the model but defaulted later.
- Monitor the performance of the model overtime and enhance the model with feedback
- Try Boosting algorithms to see if we can get better results. For example, we can try AdaBoosting, XGBoosting, GradientBoosting
- Consider to use further ensembled models, for example, combining random forest with KNN and neural network and take the majority label from the three models. Compare the performance of this further ensembled approach with just the tuned random forest model.