**Tell us how you validate your model, which, and why you chose such evaluation technique(s).**

1. Since the Kaggle dataset is the test dataset, I split the training dataset into a train set and a validation set to evaluate the performance of the model after we have obtained the best training model.
2. I used a 5-fold cross validation on the training set to prevent overfitting since the results are averaged across the folds. I tried this on a few different types of baseline models to see which ones were likely to perform better, and selected those for further tuning.
3. To tune the model, I used GridSearchCV to iterate through a few possible values for key parameters to obtain the best possible model, then applied the best model to the training set.
4. To evaluate the model, I fit the best model from step 3 onto the validation dataset and obtained the ROC AUC score to evaluate the performance. This metric was used as it was the evaluation metric for the Kaggle competition.

**What is AUC? Why do you think AUC was used as the evaluation metric for such a problem? What are other metrics that you think would also be suitable for this competition?**

AUC refers to the Area under the Curve, usually used as the AUC ROC (Receiving Operator Characteristics) curve. The AUC measures how well the model can distinguish between the classes - the higher the AUC, the better the model is. The ROC is plotted with the true positive rate against the false positive rate of the model at all classification thresholds, which is suitable for this problem where the probabilities are calculated and used for assessment.

Another AUC method using the Precision-Recall AUC (PRAUC) curve might actually be more suitable since it is effective for unbalanced classes, which was seen in the Kaggle dataset.

F1-Score might also be appropriate for this competition since it balances the precision and recall, which would be a good middle ground for banks that want to deny loans to those that are likely to default but do not want to accidentally exclude some individuals who might also be capable of paying the loans.

**What insight(s) do you have from your model? What is your preliminary analysis of the given dataset?**

Based on the final model and the rest of the models and their respective feature importances, the target variable, SeriousDlqin2yrs (i.e. Person experienced 90 days past due delinquency or worse), is highly related to:

1. The age of the individual and their past records of making late mayments
2. The RevolvingUtilizationOfUnsecuredLines, which is the total balance on credit cards and personal lines of credit divided by the sum of credit limits

These indicators are generally indicative of the individual’s spending, debt and payment habits

At the same time, the variables that are less related to the target variable are:

1. No. of dependents the individual has
2. No. of real estate loans or lines. This is might be because it was specific to real estate and the median and 75th percentile were 1 and 2 loans respectively, thus it might not be a very distinguishable feature

Summarising from the exploratory analysis, there were also insights that the target variable was related to:

1. Age. The 25th to 75th percentile of positive cases were from individuals that were slightly younger, around the ages of 35-55, as compared to negative cases at around 40-65 years old. This is similar to the insight obtained above from the model.
2. Track record of multiple late payments, with positive cases more likely to have made multiple late payments
3. MonthlyIncome. Positive cases had slightly lower monthly income. This was actually the 3rd ranked important feature from the model as well.
4. Revolving utilisation of unsecured loans, similar to the insights obtained above from the model

**Can you get into the top 100 of the private leaderboard, or even higher?**

I did not manage to get into the top 100 but it was close and would likely be possible if I had more time (and daily tries on kaggle) to test out other models such as XGBoost or to further finetune the LightGBM model with RandomizedSearchCV. I would also have liked to try ensembling different models together, however, that might affect the interpretability.

Scores

Private: 0.86619 (Rank 166 if competition was still open), just 0.00104 less than 100th place

Public: 0.86034

Graphical user interface, application

Description automatically generated