AML Final Report

Team 4: William Gu, Angad Nandwani, Jiaxi Zhou, Yue Wang, Yuxin Qian

I. Project Goal

The lending club data we use is a complete dataset from a peer-to-peer money lending institution. The goal of the project is to use this dataset to build models that can predict if an applicant will default. Furthermore, from the model output, we wish to understand the driving features behind the result.

II. Data Preparation

Based on our exploration of data, we decided to conduct several approaches to trim the dataset. First we would focus on the variables that wouldn't cause data leakage. Secondly, we removed features that have more than 10 percent missing data. We also removed features below a variance threshold and without significant relationship with response variable based on chi-square test. Highly correlated features were also partially removed. The final data set includes 27 features with 5 categorical variables and 22 numerical covariates.

Categorical variables would be encoded, and numericals would be standardized based on the requirements of our models. To reflect real-world scenarios, several features, such as the last payment amount, are not used for modeling. Although these features are useful in classification, they leak information about the label and are typically not available when the decision on whether to give an applicant a loan is made.

III. Models & Evaluation

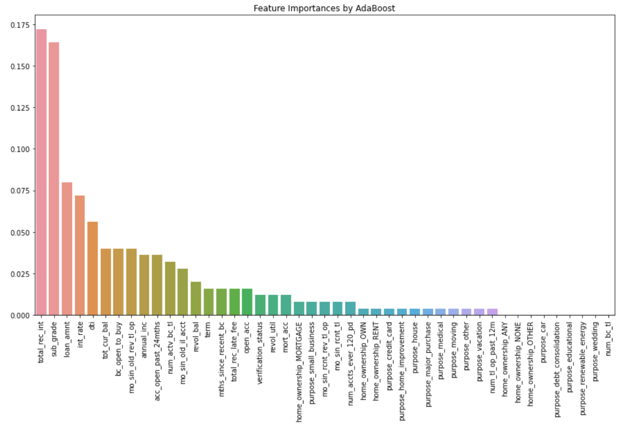
A. Table Summary

Models		Val Accuracy	Test Accuracy	Test Recall
Baseline	**Logistic Regression**	0.8131	0.8127	0.9736
Tree Models	**Random Forest**	0.8813	0.8220	0.9776
	CatBoost	0.8275	0.8259	0.9716
	XGBoost	0.8273	0.8282	0.9648
	AdaBoost	0.8174	0.8165	0.9690

B. Model Explanation and Detailed Evaluation of Selected Models

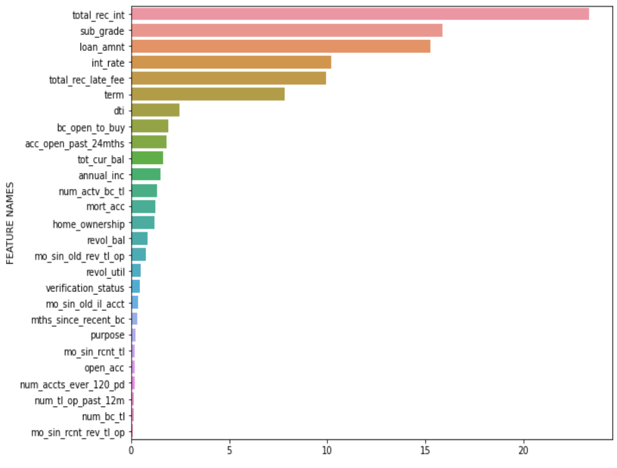
Logistic Regression: The first model we tried is logistic regression. As presented above, the recall of our best logistic regression model is around 97% percent which is preferable in our case, suggesting that our false negative is low. The best model is using Elasticnet with 0.02 L1 ratio, 1/0.6 regularization and max_iter to be 500. The top three significant features are term, purpose, and loan amount. A longer term loan and purpose to start a small business would reduce the default rate, while the larger loan amount would induce higher risk. Though it gives ideal performance on recall, this model has the drawback of not converging and an inability to efficiently handle categorical data. This motivates us to implement various tree models.

Tree Models:



AdaBoost: For the AdaBoost model, we tuned the learning_rate and n_estimators using three-way holdout. The top 3 most important features in the model were total_rec_int, sub_grade and loan_amnt, which were the same as that of the CatBoost model.

XGBoost: Keeping things simple, categorical features were label encoded and hyperparameter tuning was conducted using random search with 10-fold cross validation. Hyperparameters tuned include max depth, min child weight, number of estimators, and learning rate. For XGBoost, the top 3 important features were total_rec_late_fee, sub_grade, and term.



CatBoost: Despite the small number of categorical variables, we decided to try the CatBoost Model to try out the overfitting detector functionality. For the training, we had explicitly set the *od_pval* (threshold) to 0.001 and used the *IncToDec* default overfitting detector method. With the validation dataset passed and the parameter *use_best_model* set to true, we avoided the overfitting in each iteration. The hyper-parameters tuned through random search were *numTrees, Depth, borderCount, learningRate, bootstrapType, and L2_leaf_reg*. However, this didn't result in any better score when compared with other tree-based models. Using this model, we were also able to explore the Target Encoding, explicitly offered for this function and even execute the training of the Model over GPU. The top 3 features obtained from the model were total_rec_int, sub_grade and loan_amnt.

Random Forest : For the random forest model, we used grid search and 10-fold cross validation to find the best model. Hyperparameters tuned include the n_estimators and max_depth. We monitored the accuracy score of both the train set and test set to prevent it from overfitting. The best model gives us about 82% test accuracy and 97% recall score.

While the tree-based models perform slightly better than logistic regression, the improvement is not as high as we might have initially expected. This may be due to the types of categorical variables present in the dataset: of the remaining categorical variables, all but two are ordinal. This gives the majority of the dataset a fairly reasonable numerical representation that can be easily utilized by logistic regression. Looking at their feature importances, we observe that neither of the non-ordinal categorical features are heavily utilized by the models. When reincluding the categorical features previously removed due to fears of label leakage, tree based models score upwards of 5% higher than logistic regression models on test set classification accuracy while retaining similar recall, which showcases their potential to outperform logistic regression should impactful categorical variables be available in the future.

IV. Conclusion and Reflection

One significant feature captured by the top three list by all models is the grade level assigned to loans by 'Lending Club'. While the ranks of features are not exactly the same for different models, the top five most important features are quick common: Loan amount, subgrade, loan amount, interest rate, and total recovery late fee.

The tuned random forest model induces a bit of an overfitting issue, but it gives the highest recall score. Therefore, future improvement could be the introduction of an early stopping method or pruning method. The rest of the tree models we tried achieved similar levels of performance on all metrics. It was observed that varying the hyperparameters typically did not lead to a very large difference in performance regardless of search methodology (grid vs. random search). Of the model selection strategies (three-way holdout, k-fold CV) we tried, neither produced significantly better results than the other on the test set. There appears to be a tradeoff in classification accuracy and recall: models with higher accuracy tend to have lower recall.

With all the models' performance being so similar, there is likely no clear "best model". If we do not expect the problem or data to change, it may make sense to pick the lightest models to reduce training and prediction times while keeping costs low. However, should we expect additional features to be made available in the future, it may be wise to choose a model type that is more flexible and compatible with varying data types. If a categorical feature with high predictive power were added in the future, we may expect a greater difference in performance between tree-based models and logistic regression. Overall, in this application, a tree-based model would be ideal as the models are fairly lightweight and easily capable of handling new categorical features.