Introduction:

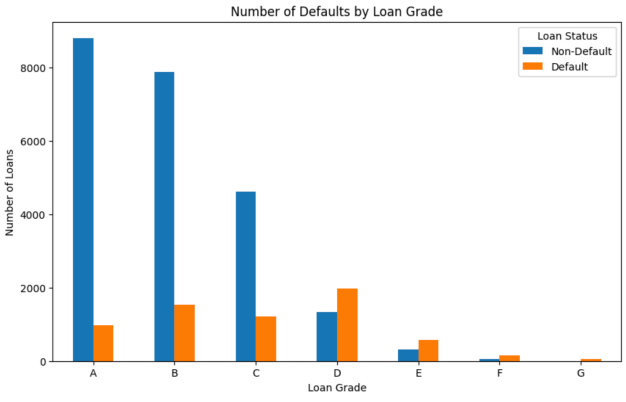
Here at InclusiFinance, we aim to make credit more accessible, particularly for individuals who currently face challenges in obtaining it. As a relatively new business, we want to build a sustainable business model that makes this possible while also making loans that are profitable to the company, meaning they are not likely to default. An essential part of developing this business model is identifying which characteristics of a loan or loanee may contribute to an eventual default. In this project, I will be using interest rates, loan-income percentage, length of credit history, grades, and whether the loanee has defaulted in the past to predict the probability of each loan defaulting. I will be choosing the best-performing model while also considering interpretability out of Random Forest, Support Vector Machines (SVM), Logistic Regression, and Simple Multi-Layer Perceptron (MLP).

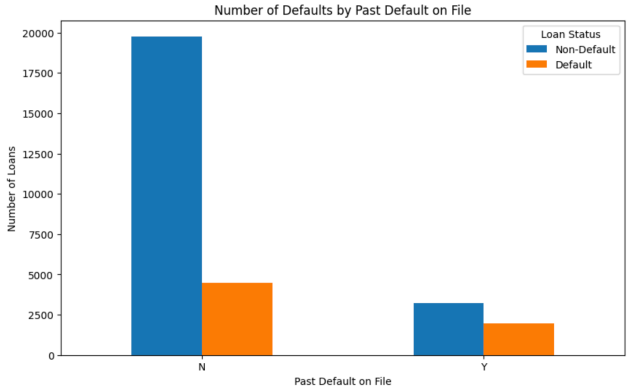
Data:

Data used in the project was obtained from a public dataset found on Kaggle: (https://www.kaggle.com/datasets/laotse/credit-risk-dataset). This dataset consists of simulated credit bureau data, though the means and methods of simulation were not explained. I selected this data as I felt that it provided strong features to determine the potential risk of a loan. The dataset also contained over 32,000 loans, which I felt would be more than enough to run models on.

One issue found with the data was a high amount (3116 observations, or 9.6% of total observations) of loans with no interest rate recorded. This needed to be addressed as I believed that the interest rate could be a very important feature for our models, especially because some models (such as SVM) cannot take null values. Upon considering multiple methods, I decided to drop these observations from our dataset as I felt that there was no appropriate way to accurately impute these missing values. While some loans may not be issued with an interest rate, this is rare and only in some specific cases. Dropping these loans could introduce potential bias towards loans with complete credit histories, so future iterations of this project could attempt to find this missing data or tune these loans to have an interest rate of zero.

Two variables explored in my exploratory data analysis were loan grades and past defaults. An interesting trend observed in loan grades was that loans did not default more than not until they reached a grade of D, as shown in the first plot below. Loanees with a previous default tended to default on their current loans nearly twice as often in our dataset, as shown in the second plot.





Methodology:

Another key consideration was how to deal with class imbalance, as roughly 25,000, or 78.2% of all loans, did not default, meaning that a model predicting loans not to default would have an accuracy of 78.2%. This means we should likely disregard accuracy metrics in evaluating our models, instead focusing on F1 Score, Precision, Recall, and Area Under the ROC Curve (AUC). We will also be stratifying our samples, which will guarantee that each of our training, validation, and test sets will have loans that defaulted. Once this has been completed, we split our data into a training set (60% of loans), a validation set (20% of loans), and a test set (20% of loans).

We now need to make a selection of models to test on our validation set before making a final decision and running the chosen model on the test set. I chose the following models:

- Random Forest Classifier: Random Forest uses many decision trees trained on various subsets of data and features. Each tree makes a prediction, and the model outputs a majority vote. It offers improved accuracy and avoids overfitting better than a traditional decision tree.
- Support Vector Machine (SVM) Classifier: SVM finds the optimal boundary (called a hyperplane) that best separates classes in the feature space. It focuses on the points closest to the boundary (support vectors) and works well for both linear and non-linear classification using kernel functions.
- Logistic Regression: Logistic Regression models the probability of an observation belonging to a class using a logistic function. This is the simplest and most interpretable model used.
- Neural Network: A Neural Network consists of layers of interconnected nodes that transform input data through weighted connections. They can learn complex, non-linear relationships, but greatly sacrifice interpretability.

Results and Analysis:

Models	Accuracy	Precision	Recall	F1 Score	ROC AUC
Random Forest	0.806041	0.559618	0.544470	0.551940	0.811910
SVM	0.811471	0.571654	0.561485	0.566524	0.820769
Logistic Reg.	0.793314	0.520129	0.749420	0.614068	0.830568
Neural Network	0.809096	0.548110	0.740139	0.629812	0.835505

Ignoring accuracy due to the class imbalance in our data, the best performing model was found to be the Neural Network, due to its high recall, F1 score, and ROC AUC. However, the tradeoff in using a Neural Network is that you lose model interpretability. It is also much more intensive to train and run this model, which must also be considered in model selection.

The feature that had the greatest impact in predicting loan defaults was the loan-income percentage, as it had the highest feature importance in the Random Forest Model and the highest coefficient in Logistic Regression. This makes sense as loans of high balance are much

easier to pay back for those with a high income, so InclusiFinance should focus on flagging loans that are a higher proportion of the loanee's income.

Model Recommendation:

In considering model results, interpretability, and runtime, I would recommend InclusiFinance use Logistic Regression. It outperformed the Neural Network in recall and was very similar in F1 score and ROC AUC. This decrease in some performance metrics is made up for in the increase in interpretability, along with much lower runtime. Using Logistic Regression, we can evaluate the features' coefficients, which gives us a much clearer idea of the effect each feature has on the model predictions.

Upon selecting Logistic Regression, this model was run on the test set data. The following results were given:

Accuracy	Precision	Recall	F1 Score	ROC AUC
0.80010	0.53165	0.74710	0.62122	0.83292

Conclusion:

In summary, I recommend that InclusiFinance move forward with the Logistic Regression model as it performed fairly well while offering strong interpretability and feasibility. According to multiple of the models tested, the feature with the strongest impact on model performance was the loan-income percentage. Loans that were a higher proportion of the loanee's income were more likely to default.

As a company, our goal is to provide credit for individuals who may have a hard time otherwise building credit. Low-income individuals greatly fall into this category, so we want to ensure loans are not denied simply due to lower income. InclusiFinance could explore flagging loans with concerning loan-income percentages. This will guarantee that loans will be considered for all people, while also providing a sustainable business model for InclusiFinance moving forward.