Loan Approval Classification

Data 622 Assignment 1

Shaya Engelman

# Introduction

In this analysis, I explored two datasets to predict loan approval using several algorithms, including Random Forest (RF), Neural Networks, Logistic Regression, and AdaBoost. The goal was to evaluate how these models performed based on the size and characteristics of the datasets, the presence of class imbalance, and how feature relationships influenced the results. Additionally, I considered model interpretability, which is particularly important in regulatory contexts like loan approvals, where decisions must often be explained to consumers.

## Dataset 1: Exploratory Data Analysis (EDA) and Key Insights

The first dataset contained 32,500 rows and 13 columns, providing a manageable starting point for detailed exploratory data analysis (EDA). I generated correlation heatmaps, histograms, and boxplots to understand the distribution of the data and its relationships with the target variable, loan status.

One of the immediate findings was that many numerical variables—such as age, employment length, and credit history length—were heavily right-skewed. These variables logically correlated with one another, particularly age and credit history length, as older applicants tend to have longer credit histories and more years of employment. While this relationship was expected, it did not appear to have a strong direct correlation with the target variable.

On the other hand, the income variable was notably right-skewed, with most applicants earning under a million dollars but a few outliers earning far more. Since income is a key factor in loan approval decisions, I had to consider how to handle this skewness, as the presence of extreme high-income values could distort model performance. I considered winsorizing the income variable to reduce the influence of these outliers.

The target variable, loan status, was imbalanced, with a higher proportion of approved loans. This imbalance needed to be addressed in the modeling phase, as imbalanced data can skew model results, particularly for algorithms that struggle with unequal class distributions. I also identified outliers in variables such as age, employment length, and income—one extreme outlier in income, in particular, skewed the distribution significantly. These outliers were taken into account during the data cleaning process to ensure the models were not unduly influenced by them.

### Categorical Variables

Among the categorical variables, housing status, loan intent, loan grade, and default on file stood out as important predictors:

* Housing status showed that most applicants were either renters or mortgage holders, while very few fell into other categories. This concentration could limit the model's ability to generalize for applicants outside these two groups.
* Loan intent and loan grade were fairly evenly distributed, which was beneficial for the model since it ensured there was enough data to learn from for each category. The loan grade variable, being ordinal, was later converted into a numeric format to facilitate model training.
* Default on file was highly imbalanced, with only a small percentage of applicants having a default on record. I addressed this by converting the feature into a binary variable for modeling.

The pairplots showed that most numerical variables were not strongly correlated with the target variable. However, loan interest rate and loan percent income showed the clearest delineation between approved and non-approved loans, suggesting that these features were significant predictors. Grouped boxplots further supported this observation, as these two variables exhibited the most distinct separation between loan approval classes.

### Model Selection and Performance on Dataset 1

Given the manageable size of the first dataset and its clear feature relationships, I explored several algorithms, including logistic regression, Random Forest, and AdaBoost, to handle the imbalanced target variable. I also considered that the dataset's size and dimensionality were low enough to allow for one-hot encoding of categorical variables without worrying about computational overhead.

I initially experimented with logistic regression, given its simplicity and interpretability. However, logistic regression struggled with the non-linear relationships in the data and the imbalance in the target variable. While it offered insights into how individual features contributed to loan approval, its overall performance lagged behind more sophisticated algorithms.

Next, I applied a Random Forest (RF) model, expecting it to perform better given its robustness against class imbalance and its ability to capture non-linear relationships. As expected, the RF model outperformed logistic regression significantly:

* **Accuracy**: 93.66%
* **Precision**: 96.90%
* **Recall**: 71.75%
* **F1 Score**: 82.45%
* **ROC AUC**: 85.57%

The Random Forest model's performance was strong across all metrics, making it the clear winner for this dataset. Its ability to handle the imbalanced target variable and capture complex feature interactions made it ideal for predicting loan approvals. Importantly, RF's feature importance scores also provided insights into which features drove the predictions, making it a useful tool despite its more complex nature compared to logistic regression.

Surprisingly, AdaBoost, which I had assumed would perform well on imbalanced data, performed poorly in this case. Given AdaBoost’s reputation for handling imbalanced datasets effectively by focusing on misclassified instances, I expected it to be a strong competitor to Random Forest. However, its performance fell short, leading me to wonder if I might have misconfigured the algorithm or if the nature of this particular dataset did not suit AdaBoost’s strengths. This unexpected result highlighted the importance of thoroughly testing different models, as theoretical advantages do not always translate into real-world performance.

## Dataset 2: EDA and Model Performance

The second dataset was much larger, containing 252,000 rows and 13 columns. While the features were similar to the first dataset, the larger scale presented additional challenges, especially in managing outliers and imbalanced categories. I performed a similar EDA, but due to the size of the dataset, I could not manually inspect individual outliers or explore feature relationships as deeply.

Many of the categorical columns had a much larger number of distinct values than in the first dataset, making one-hot encoding impractical. As a result, I used target encoding to handle these features, which allowed me to maintain a reasonable number of dimensions without overwhelming the model. The target variable was even more imbalanced in this dataset, which posed an additional challenge for model performance.

I tested both a Neural Network and a Random Forest model on this dataset. The Neural Network model underperformed, with low precision and recall:

* **Accuracy**: 80.97%
* **Precision**: 24.60%
* **Recall**: 27.21%
* **F1 Score**: 25.84%
* **ROC AUC**: 57.82%

In contrast, the Random Forest model once again delivered the best results:

* **Accuracy**: 90.62%
* **Precision**: 66.10%
* **Recall**: 47.26%
* **F1 Score**: 55.11%
* **ROC AUC**: 71.95%

Although the Random Forest model performed well on this larger dataset, the increased size presented potential challenges. I was initially concerned that the computational cost of running RF on such a large dataset would exceed my device’s limitations, especially if I had used one-hot encoding, which would have greatly increased the number of dimensions. However, with target encoding, the model remained feasible.

## Interpretability and Regulatory Implications

Interpretability is a key consideration in predictive models for loan approval, particularly in regulated industries like finance. In both the U.S. and EU, regulations often require that consumers receive explanations for loan denials. While models like Random Forest provide high accuracy, their interpretability is limited compared to simpler models like logistic regression, which directly show how each feature contributes to the prediction.

However, Random Forest’s feature importance scores offer some level of interpretability, providing insights into which features influence the model’s decisions. This transparency is crucial for meeting regulatory requirements, and if more interpretability is needed, linear regression or other interpretable models can complement the predictive power of more complex models.

## Conclusion

In both datasets, Random Forest emerged as the most effective model for predicting loan approval, particularly in handling class imbalances and capturing complex relationships between features. Despite the challenges posed by larger datasets, feature engineering—particularly the use of target encoding—helped manage the computational complexity and maintain model feasibility.

While Neural Networks showed promise, they struggled with the imbalanced target variable and offered limited interpretability. AdaBoost, surprisingly, did not perform as well as expected, even though it is theoretically suited for imbalanced data. This result raised questions about potential misconfigurations or dataset-specific limitations.

For making business decisions, I would trust the Random Forest model for its consistent performance across both datasets. Additionally, for regulatory contexts where transparency is critical, the combination of Random Forest’s feature importance and simpler, interpretable models like logistic regression can offer a balanced approach to predictive accuracy and explainability that might justify their trade-off in accuracy.