Predicting Credit Card Defaulters

Team Name: Doraemon

Arjun Verma (230102125) Subhashree Sahoo (230102122) Neha Sellapan Devdas (230123040)

INTRODUCTION

Effective credit risk management is crucial for ensuring profitability and minimizing losses in the financial sector. To achieve this, it is essential to establish a robust risk management framework for existing credit card customers. This includes implementing portfolio risk management strategies such as monitoring creditworthiness, adjusting credit limits, preventing defaults, and detecting fraud.

By segmenting customers based on risk, the bank can apply tailored strategies to reduce exposure and safeguard against potential losses.The primary objective of this project is to assess the probability of a customer defaulting on their credit card payments, enabling proactive and data-driven decision-making. By leveraging historical account details, transactional patterns, and credit bureau records, this initiative aims to create a comprehensive risk assessment tool.

The model's outcomes will guide targeted interventions, improve portfolio management, and minimize financial losses. This report outlines the systematic methodology adopted to develop and validate the model, detailing the technical steps and the rationale behind each decision. First we identify all the details of a particular credit card that has been given.

DATASET OVERVIEW:

The initial step in the process involved a thorough examination of the two datasets provided. The development dataset comprised 96,806 credit card accounts, each associated with a target variable ("bad_flag") indicating whether a customer had defaulted. The validation dataset included 41,792 accounts, where the task was to predict the probability of default. To streamline the analysis and tailor the preprocessing steps, the features were categorized into three distinct groups:

- **Onus attributes**: These refer to characteristics intrinsic to the credit card account, such as credit limit, account type, and account tenure. These variables provide insights into the structural aspects of the credit card portfolio.
- **Transaction attributes**: This category includes metrics that capture customer spending behavior such as the frequency and value of transactions across various merchant categories. These attributes help identify patterns in spending that may correlate with credit risk.
- **Bureau data**: These attributes reflect external credit behaviors, including historical delinquencies and recent inquiries from other lenders, offering a broader view of the customer's financial health and creditworthiness.

By organizing the data into these categories, the analysis could prioritize preprocessing steps that address the unique characteristics of each group. This structured approach facilitated the identification of meaningful patterns and relationships within the data, laying a solid foundation for subsequent modeling efforts.

DATA PREPROCESSING:

The data preprocessing technique employed focuses on handling missing values across various feature categories, ensuring the dataset is ready for predictive modeling. Missing data within the various attributes is imputed using a multi-step approach to preserve the temporal and structural integrity of the data.

Linear interpolation is applied first to estimate missing values based on adjacent entries, followed by backward and forward filling to handle edge cases. Any remaining missing values are replaced with zero, ensuring no feature contains null values. This systematic imputation strategy minimizes information loss and ensures that the processed data is consistent, which is critical for the robustness and accuracy of the behavior score model.

Visualizations such as histograms and boxplots are employed to assess the distribution and variability of numerical features, providing insights into their range, skewness, and presence of outliers. By comparing these distributions across the target variable (bad_flag), we can identify potential predictors of credit card default. Correlation heatmaps highlight the relationships between numerical features, helping to detect multicollinearity and guiding feature selection for modeling.

Categorical features are analyzed through bar charts to understand their frequency distributions and their relevance to default risk. Lastly, pair plots provide a comprehensive view of pairwise relationships among key features, enabling the identification of clusters or trends that may signify distinct customer behaviors. This step not only enhances our understanding of the data but also ensures that the processed dataset is optimized for building robust and accurate predictive models.

FEATURE ENGINEERING:

The first step involves aggregating **“bureau_enquiry”** features to capture the total number of enquiries, their recency trends using weighted averages and their statistical properties (mean, standard deviation). These metrics provide insight into the frequency and nature of credit enquiries, highlighting customer tendencies to seek new credit lines—a potential risk factor.

Similarly, features derived from **onus attributes** (e.g., credit limits) focus on calculating averages, trends over time and deviations, enabling an understanding of a customer’s credit utilization behavior. By assessing dynamic thresholds based on average credit limits, the process identifies high-value transaction ratios, offering insights into spending patterns relative to available credit.

Volatility measures, such as standard deviation of **transactional attributes,** are introduced to capture fluctuations in spending behavior, while thresholds are defined to flag high-risk customers based on excessive volatility and near-limit credit utilization. Trend analysis across recent versus older credit limits reveals shifts in credit behavior over time, which can signal improving or deteriorating financial discipline.

Finally, ratios like credit utilization and enquiries-to-credit-limit are computed to link spending and credit-seeking behaviors with available limits. High utilization and enquiry ratios often correlate with financial strain or risk, making them valuable predictors in risk assessment.

This comprehensive approach enriches the dataset with nuanced features, improving the predictive power of models. By uncovering behavioral trends, deviations, and risk indicators, the engineered features enable a deeper understanding of customer financial behavior and allow for targeted portfolio risk management.

FEATURE SCALING

Z-Score scaling is used which uses a standard library named **‘StandardScaler’** that helps in standardising numerical features by centering them around a mean of zero and scaling them to have a standard deviation of one. This process ensures that all features are on a comparable scale, which is particularly important when dealing with models sensitive to the magnitude of input variables.

In the context of our project, scaling eliminates the dominance of features with larger numeric ranges, such as credit utilization ratios, over smaller-ranged features like enquiry counts. This allows the model to assess the relative importance of each feature fairly, enhancing its predictive accuracy. Moreover, consistent scaling improves the convergence of optimization algorithms during model training, leading to faster and more stable learning. By preprocessing the data with Z-Score scaling, we ensure a standardized and robust input that facilitates the development of an effective behavior score model.

Model Design and Justification

The model aims to address a binary classification task focused on predicting whether a customer will default on a credit card or engage in fraudulent activity. To effectively handle this, the model uses a hybrid deep neural network (DNN) architecture that integrates both temporal and static features. This combination allows the model to capture dynamic patterns over time, such as payment history, while also leveraging static attributes like demographic data. The model is designed to handle class imbalance, which is common in fraud detection and default prediction tasks, by applying class weighting during training. By assigning higher weights to the minority class (fraud/default), the model places greater emphasis on these critical yet underrepresented cases.

The architecture itself is a feedforward neural network with LSTM (Long Short-Term Memory) layers designed to capture temporal dependencies in the data. This is particularly important for credit card fraud detection, where fraudulent activity may exhibit patterns over time, such as irregular spending behavior. The static input features, such as age, income, or previous credit history, are processed using dense layers to extract meaningful information. This hybrid approach—combining temporal sequences with static features—aims to maximize the predictive power of the model by accounting for both time-dependent patterns and static factors.

Data Preprocessing and Class Imbalance Handling

In fraud detection and credit card default prediction tasks, one of the most significant challenges is class imbalance. The majority of customers do not engage in fraudulent activity or default, making these cases rare but highly important. To mitigate this, the model applies **class weighting**. Class weights are dynamically adjusted so that the minority class is penalized more heavily during training, forcing the model to focus on correctly predicting these instances. Additionally, the model includes a custom resampling function—`custom_balanced_sample`—which combines **undersampling** of the majority class and **synthetic data generation** for the minority class. This ensures a more balanced dataset, reducing bias toward the majority class and improving the model's ability to learn from the minority class.

The resampling process works by first undersampling the majority class to match a desired ratio with the minority class. If the number of majority samples is insufficient, synthetic data points are generated by averaging samples from the minority class. This synthetic generation, akin to **SMOTE (Synthetic Minority Over-sampling Technique)**, ensures the model has enough varied examples of the minority class to learn meaningful patterns without introducing noise.

Model Architecture and Training Process

The hybrid model architecture begins with LSTM layers for temporal data processing. The LSTM layers are designed to capture sequential dependencies in the data, which is crucial when analyzing time-series patterns, such as transaction frequency, amount, or changes in spending behavior. The LSTM layers are followed by **Dropout** and **Batch Normalization** layers, which prevent overfitting and help with faster convergence. **Dropout** randomly deactivates a percentage of neurons during training to force the model to generalize better, while **Batch Normalization** standardizes the activations of neurons, which helps avoid the vanishing/exploding gradient problem and accelerates training.

For the static data (e.g., demographic information), the model uses fully connected dense layers, also with **Dropout** and **Batch Normalization**, ensuring a similar regularization approach. The combination of these two types of data—temporal and static—takes advantage of the sequential dependencies and the more static attributes, ultimately enabling the model to capture a broader set of patterns and behaviors that lead to defaults or fraud.

The model is compiled with the **Adam optimizer**, which is widely used for its adaptive learning rate properties, making it efficient for deep learning tasks. The **focal loss function** is chosen instead of traditional binary cross-entropy loss, as it gives more weight to the misclassified minority class samples. This is especially useful in highly imbalanced datasets, where the model needs to focus more on correctly predicting fraud or default cases.

Evaluation and Performance Metrics

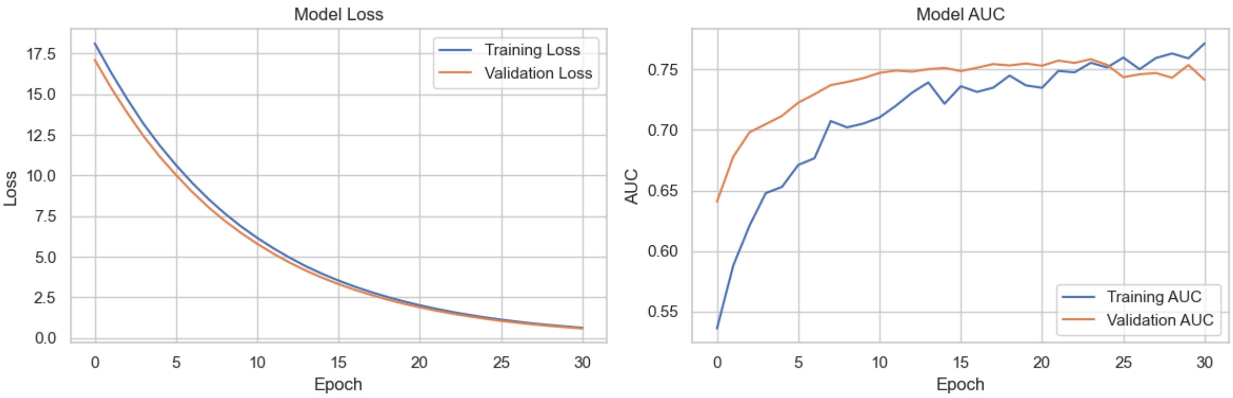
Once the model is trained, its performance is evaluated using a set of metrics that provide insight into both its overall accuracy and its ability to distinguish between the two classes. **Precision**, **Recall**, and **F1 Score** are the key metrics, as they are particularly important when dealing with imbalanced datasets. Precision indicates how many of the predicted fraud/default cases are actually true positives, while recall shows how many actual fraud/default cases were correctly identified. The F1 score is the harmonic mean of precision and recall, providing a balance between these two metrics.

Additionally, the **AUC-ROC score** is used to evaluate the model's ability to distinguish between the two classes across different thresholds, offering a more nuanced measure of performance. The **classification report** provides a breakdown of these metrics for each class, giving a detailed view of how well the model is handling both the majority and minority classes. The **confusion matrix** complements this by showing the raw counts of true positives, true negatives, false positives, and false negatives, which is essential for understanding the exact nature of errors made by the model.

Limitations and Future Work

Despite the strengths of this approach, several limitations remain. The risk of **overfitting** persists, especially with small or highly imbalanced datasets. While **Dropout** and **L2 regularization** help mitigate this to some extent, overfitting can still occur if the model is excessively complex or if the dataset does not sufficiently represent all possible variations of fraudulent or default behavior. As a result, **cross-validation** and **hyperparameter tuning** will be necessary to fine-tune the model and ensure it generalizes well to unseen data.

Another limitation arises from the handling of **class imbalance**. Although class weighting and resampling methods are applied, these approaches may not fully address situations where the minority class is extremely underrepresented. In such cases, **advanced resampling techniques** like **generative adversarial networks (GANs)** for data augmentation or **anomaly detection** algorithms may be explored as alternative solutions.


Model Loss

Model AUC