POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Advancing Loan Default Prediction with Interpretable TabNet Models

Tesi di Laurea Magistrale in
Mathematical Engineering - Ingegneria Matematica

Author: **Arnaldo Mollo**

Student ID: 953632
Advisor: Prof. Daniele Marazzina
Academic Year: 2022-2023

Abstract

This thesis provides an extensive analysis of the TabNet model, a deep learning architecture for tabular data, focusing on its application in loan default prediction. Building on the evolving landscape of machine learning models in credit risk assessment, we compare TabNet's performance and interpretability against prominent models such as LightGBM, XGBoost, Logit, and Random Forest. In our investigation, TabNet stands out as a powerful tool, offering a strong compromise between predictive performance and model interpretability, outpacing other models in key metrics such as Validation Accuracy, Test Accuracy, and Test Precision. One of the core contributions of this work lies in our detailed exploration of TabNet masks, a distinctive feature of the TabNet model that assigns importance to features at each decision step. This novel technique unravels the complex decision-making process in loan default prediction, providing granular insights into the interplay and influence of features at various stages. We also critique the traditional black-box interpretability techniques, such as SHAP values, in providing comprehensive insights, highlighting their shortcomings in capturing complex feature interactions, their assumption of feature independence, and their computational intensity. Our results underscore the potential of more transparent, interpretable models like TabNet in high-stress financial applications, emphasizing the importance of understanding the rationale behind predictions. In conclusion, we propose future research directions, including expanding the scope of TabNet to various financial tasks and geographical contexts, and enhancing computational efficiency in interpretability-focused models. This thesis seeks to drive advancements in the domain of interpretable machine learning for financial applications, reinforcing the balance between model performance and interpretability as a cornerstone of future research.

Keywords: Tabnet, loan prediction, loan default, lending club, interpretability, XAI, Deep learning

Abstract in lingua italiana

Questa tesi propone un'analisi accurata del modello Tabnet, un'architettura di deep learning per i dati tabulari, concentrandosi sul problema della predizione dei default sui prestiti. Utilizzando come punto di partenza il variegato assortimento di modelli per la valutazione del rischio di credito, mettiamo a paragone la prestazione e l'interpretabilità di Tabnet con modelli di spicco come LightGBM, XGBoost, Logit e Random Forest. Nella nostra indagine Tabnet emerge come strumento potente, offrendo un compromesso tra capacità predittiva e interpretabilità modellistica, superando gli altri modelli in metriche chiave come Validation Accuracy, Test Accuracy e Test Precision. Uno dei contributi centrali di questo lavoro consiste nella nostra dettagliata indagine delle maschere di Tabnet, una caratteristica distintiva del modello, che seleziona le caratteristiche più importanti ad ogni step decisionale. Questa metodologia rivela il complesso processo decisionale legato alla predizione dei default sui prestiti, fornendo una visione granulare sull'interazione e sull'influenza delle varie caratteristiche in differenti fasi. Oltretutto critichiamo le tecniche tradizionali black-box di interpretabilità, come gli SHAP values, mettendo in evidenza le loro limitazioni nel catturare interazioni complesse tra le caratteristiche, le loro assunzioni di indipendenza tra caratteristiche, e il loro carico computazionale. I nostri risultati esaltano la potenzialità di modelli più trasparenti e interpretabili, come Tabnet, in applicazioni finanziarie delicate, ponendo in rilievo l'importanza della comprensione del razionale dietro le decisioni. In conclusione, proponiamo future direzioni di ricerca, tra le quali: ampliare l'applicazione di tabnet a diversi task finanziari e contesti geografici e migliorare l'efficienza computazionale. Questa tesi si propone di guidare progressi nel dominio del machine learning interpretabile per applicazioni finanziarie, ponendo il bilanciamento tra prestazioni e interpretabilità come pietra miliare della futura ricerca.

Parole chiave: Tabnet, predizione prestiti, default prestiti, lending club, interpretabilità, XAI, Deep learning

Contents

Introduction

The application of machine learning models for loan default prediction has evolved significantly over the years, driven by the need to enhance the stability of financial institutions and the broader economy. As traditional credit scoring models, such as logistic regression and linear discriminant analysis, faced limitations in capturing non-linear relationships and processing high-dimensional data [9, 16], the focus shifted toward developing more advanced and effective techniques.

Initially, researchers relied primarily on statistical models, such as logistic regression, linear discriminant analysis, and the probit model. Despite their effectiveness in certain scenarios, these models encountered challenges in processing large and complex datasets and capturing the underlying non-linear relationships [9, 16]. As the limitations of early statistical methods became apparent, researchers turned their attention to more advanced machine learning techniques, such as decision trees, support vector machines (SVM), and artificial neural networks (ANN) [3].

Ensemble learning methods emerged as a promising alternative for loan default prediction, with techniques such as random forests and AdaBoost gaining popularity due to their ability to reduce overfitting and improve prediction accuracy [1]. The introduction of gradient boosting models, specifically XGBoost and LightGBM, provided another leap forward in loan default prediction. These tree-based ensemble learning methods leverage gradient boosting to minimize prediction errors and have demonstrated superior performance across various applications, including loan default prediction [6, 11].

With the advent of deep learning, researchers began exploring its potential in loan default prediction. Deep learning models, such as feedforward neural networks, recurrent neural networks, and convolutional neural networks, can learn complex patterns and representations in large-scale data, making them well-suited for credit scoring tasks [12, 18].

However, the quest for more complex and advanced models led to a shift from interpretable white-box models to unintelligible black-box models, with ensemble learning methods and deep neural networks being prime examples. To counteract this effect, techniques such as SHAP values have been applied to these black-box models for interpretability [13].

The deep learning landscape changed profoundly with the introduction of the Transformer architecture in 2017 [17]. In the following years, previously heterogeneous research areas of machine learning were unified under the Transformer architecture, achieving state-of-the-art performance in their respective fields, such as natural language processing, computer vision and audio. The only field where gradient boosting techniques remained the top performers was tabular data, until the TabNet architecture was introduced [2]. TabNet aimed to seamlessly rival gradient methods while offering direct interpretability, a significant advantage over SHAP values, as the latter are linear approximations of non-linear input-output relationships and are computationally expensive, whereas TabNet provides a direct view of its inner workings at virtually zero computational cost.

In this thesis, we delve into the complex issue of loan default prediction, focusing specifically on the interpretability of the TabNet model. We demonstrate its competitive performance against other prevalent models while ensuring a level of interpretability often missing in complex machine learning models. Our work builds upon the findings presented in "Explainable Artificial Intelligence: interpreting default forecasting models based on Machine Learning" by Giuseppe Cascarino, Mirko Moscatelli, and Fabio Parlapiano [5]. In their research, the authors utilized and interpreted the Random Forest model for corporate default prediction.

Our analysis involves a comparison of several machine learning models for loan default prediction, including LightGBM, XGBoost, Logit, and Random Forest. The inclusion of Logit and Random Forest is influenced by insights from "Predicting Default Risk on Peer-to-Peer Lending Imbalanced Datasets" by Yen-Ru Chen, Jenq-Shiou Leu, Sheng-An Huang, Jui-Tang Wang, and Jun-Ichi Takada [7]. This paper highlights the significance of handling imbalanced datasets in default risk prediction, a common challenge in peer-to-peer lending scenarios.

However, the cornerstone of our thesis lies in our in-depth exploration of the TabNet model's interpretability. Based on a range of performance metrics, such as AUC, accuracy, recall, precision, and F1 score, we select TabNet as the model offering the best compromise between performance and interpretability. We enrich our exploration of interpretability through a detailed analysis of two interpretability techniques: SHAP values and TabNet masks.

The novelty and value of our work primarily reside in our comprehensive analysis of TabNet masks. This technique assigns feature importance at each decision step within the TabNet model, offering granular insights into how features interact and influence the prediction at various stages of the process. This level of detail allows for a richer under-

standing of the complex interplay of features, which is particularly beneficial in intricate domains like credit risk. Comprehending the sequence and interaction of feature importance can yield crucial insights, making the TabNet masks methodology an invaluable tool for credit risk prediction.

Through this work, we aim to make a significant contribution to the existing body of knowledge in the field of loan default prediction. Our goal is to provide practical insights that could guide the financial industry in making more informed decisions when assessing and managing loan default risk.

The chapters are structured in the following manner:

Chapter 1 serves as the theoretical groundwork for the thesis, presenting the Background and Related Work. This section provides a review of Traditional Statistical Models, Ensemble Learning Methods, and Deep Learning Models, and also discusses various Interpretability Techniques and Performance Metrics.

Chapter 2 delves into the TabNet model. It introduces the model, explains its architecture, and explores its key components such as Feature Embeddings, Sequential Attention, and Feature Selection mechanisms among others.

Chapter 3 presents the Dataset and Feature Preprocessing methodologies used in this study. The Lending Club Dataset is introduced, and various preprocessing steps, including missing values treatment, scaling and encoding, and label cleaning, are described.

In Chapter 4, the actual Unsupervised and Supervised Training of the models is demonstrated. Various models are trained, TabNet, LightGBM, XGBoost, Random Forest, and Logistic Regression. The results of these trainings are also presented.

Chapter 5 conducts a TabNet Interpretability Analysis, examining global and local interpretability features. It includes an analysis of feature importance and SHAP values.

Chapter 6 focuses on the LightGBM Interpretability Analysis. It provides an analysis of SHAP values, Feature Importances, and their comparison. A comparative analysis between LightGBM and TabNet interpretability is also included.

The Conclusions and Future Developments chapter summarizes the findings of the research and outlines potential directions for future work.

1 | Background and Related Work

This chapter presents an overview of the prevalent methods and models for loan default prediction, focusing on their strengths, limitations, and interpretability. The methods covered include traditional statistical models like logistic regression, ensemble learning models such as random forests, gradient boosting models like XGBoost and LightGBM, and the TabNet architecture. We also provide a brief overview of some key studies that used these models for loan default prediction.

1.1. Traditional Statistical Models

1.1.1. Logistic Regression

Logistic regression is a linear model for binary classification problems, making it apt for credit scoring and loan default prediction [9]. In logistic regression, the log odds of the outcome is modeled as a linear combination of the predictor variables. The model is represented as follows:

Given predictor variables $X = (X_1, X_2, ..., X_n)$ and a binary outcome Y, the logistic regression model is:

$$P(Y = 1|X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \cdots + \beta_n X_n)}}. \tag{1.1}$$

Here, $\beta_0, \beta_1, ..., \beta_n$ are the parameters of the model. These parameters are typically estimated via the method of maximum likelihood.

While logistic regression is intuitive and easy to interpret, it assumes a linear relationship between the log odds of the outcome and the predictor variables. This linearity assumption makes it difficult for logistic regression to capture complex non-linear relationships. Furthermore, logistic regression struggles with high-dimensional data where the number of predictor variables is large, often leading to overfitting [9, 16].

1.2. Ensemble Learning Methods

1.2.1. Random Forests

Random forests are a powerful ensemble learning technique that aggregates the predictions of multiple decision trees [1, 4]. By constructing each tree from a different bootstrap sample of the training data and using a random subset of the predictor variables at each split, random forests achieve robust performance while avoiding overfitting.

Random forests are particularly well-suited for datasets with complex interactions and non-linear relationships. They can handle both categorical and numerical features, and are robust to outliers and missing data. However, due to the large number of trees and splits, random forests may be difficult to interpret, and the randomness in tree construction can sometimes lead to variability in predictions.

1.2.2. Gradient Boosting Models

XGBoost

XGBoost, short for "Extreme Gradient Boosting", is a powerful and efficient implementation of gradient boosting [6]. It enhances the base gradient boosting algorithm by adding a regularization term to the loss function, which helps control the complexity of the model and prevent overfitting.

In XGBoost, each tree is built to correct the mistakes of its predecessor, leading to a gradual improvement in prediction error. The final prediction is obtained by summing the predictions of all individual trees. XGBoost's ability to handle a variety of data types, its robustness to missing values and outliers, and its scalability make it a popular choice for many predictive tasks, including loan default prediction.

LightGBM

LightGBM is a gradient boosting framework that uses tree-based learning algorithms [11]. It improves on other gradient boosting models by employing a leaf-wise tree growth strategy, as opposed to the traditional level-wise strategy. This leaf-wise strategy can result in better model performance, as it allows the model to focus more on the misclassified instances.

LightGBM also incorporates several techniques to speed up training and reduce memory usage, such as gradient-based one-side sampling (GOSS) and exclusive feature bundling

(EFB). Its ability to handle large-scale data and its support for parallel and GPU learning make LightGBM a powerful tool for a wide variety of applications, including loan default prediction.

1.3. Deep Learning Models

Tabnet

TabNet is a novel deep learning model designed for tabular data [2]. It was developed by Google Cloud AI and addresses some of the weaknesses of traditional deep learning methods when applied to structured data. The architecture of TabNet leverages the strengths of both tree-based models and neural networks. It utilizes sequential decision-making like decision trees, combined with the representation learning ability of neural networks.

The TabNet model uses an attention mechanism to select features at each decision step, effectively learning to reason in a similar way to decision trees. It sequentially updates a hidden state based on the selected features at each step, similar to a recurrent neural network, but in the context of tabular data.

TabNet's unique ability to provide feature importance scores at the instance level gives it a degree of interpretability typically not associated with neural network models. This makes it easier to understand the model's decision-making process, providing valuable insights that can be crucial in fields such as finance, where interpretability is important for regulatory and trust reasons.

However, as a relatively new model, TabNet is less mature compared to established models like XGBoost and LightGBM. This can potentially lead to challenges in implementation and optimization.

In the context of loan default prediction, TabNet's feature selection and instance-level interpretability can be beneficial in identifying key risk factors and explaining individual predictions. These capabilities make TabNet an interesting and promising approach for improving loan default prediction.

1.4. Interpretability Techniques

1.4.1. SHAP values

SHAP values, derived from the concept of Shapley Values of cooperative game theory, provide a powerful tool for understanding the importance of features in a prediction model [13]. They assign a contribution value to each feature, showing how much each feature in the data contributed to the prediction for a particular instance. This ability to explain individual predictions can help to clarify the inner workings of complex models, and to identify potentially influential features or patterns in the data. Mathematically, the Shapley value for a feature i is defined as:

$$\phi_i(f) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!}(f(S \cup \{i\}) - f(S)), \tag{1.2}$$

Where, f represents the prediction function, N denotes the set of all features, and S is any subset of N excluding feature i. The cardinalities of S and N are represented by $|S|$ and $|N|$ respectively. The Shapley value $\phi_i(f)$ calculates the average marginal contribution of feature i over all possible subsets of features, summing up the differences in the prediction function's output when feature i is included and excluded from each subset.

The exact computation of SHAP values is a nontrivial task. The SHAP explanation model simplified input mapping is then given as:

$$f(h_x(z^0)) = E[f(z)|z_S], \tag{1.3}$$

Where, f is the function that the SHAP values aim to explain, h_x is a simplified input mapping function, z^0 represents the simplified version of the instance, $E[f(z)|z_S]$ represents the expected value of $f(z)$ given the features in the set S, z_S represents the values of features in the set S and S is a subset of all features.

Despite their utility, SHAP values are not without their shortcomings. As Molnar et al. [15] noted, SHAP values, like other model-agnostic interpretation techniques, may lead to incorrect conclusions if applied inappropriately. For example, they may not correctly interpret models that do not generalize well, leading to misleading feature importance assessments. This is particularly concerning because the computation of SHAP values fundamentally assumes each feature contributes independently to the prediction, an assumption which does not consider the potential interactions between features. Consequently,

this could result in over- or underestimation of the importance of certain features.

Moreover, SHAP values provide a contribution for each feature independently, meaning that feature dependencies and interactions could be overlooked, leading to an over- or underestimation of the importance of certain features. The authors further caution that SHAP values may struggle in high-dimensional settings, because of the complexity associated with summing over all possible subsets of features. In addition, SHAP values may be prone to promoting unjustified causal interpretations. In the context of your work, understanding these limitations can help ensure a cautious and informed application of this tool.

1.4.2. TabNet Masks

TabNet, a neural network architecture, offers a unique method for interpretability through the use of attention masks. These masks essentially reveal which features the model focuses on during each decision step, making the model's decision process more transparent.

The primary strength of TabNet masks is their ability to capture both feature importance and feature interaction directly within the model's architecture. This contrasts with many post-hoc interpretability methods, like SHAP values, which may overlook feature interactions or dependencies.

Furthermore, TabNet masks handle high-dimensional data well, making them effective even in complex datasets with many features. Also, as they are an integral part of the model's architecture, they provide insight into the model's behavior without the need for additional interpretation methods. It's worth mentioning, however, that while TabNet masks provide useful insights, the masks themselves may require interpretation, as they indicate feature importance across decision steps rather than providing an immediate measure of global feature importance [2].

1.5. Performance Metrics

Predictive model performance is critically important in the field of machine learning, with different metrics offering distinct ways to assess and compare the effectiveness of models. In the context of loan default prediction, where the goal is often to identify the likelihood of future default events, it is vital to select appropriate performance metrics that can effectively evaluate the model. The metrics selected for this study are accuracy, precision, recall, F1 score, and the area under the receiver operating characteristic curve (AUC-ROC). Each of these metrics provides different insights into the model's capabilities.

1.5.1. Accuracy

Accuracy is perhaps the most straightforward performance evaluation metric. It provides an intuitive snapshot of the overall performance of a model by calculating the proportion of correct predictions made out of all predictions. The mathematical representation of accuracy is as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \tag{1.4}$$

In the equation above, TP represents true positives, TN represents true negatives, FP represents false positives, and FN represents false negatives. However, it should be noted that while accuracy can provide a useful high-level view of model performance, it can also be misleading, particularly in the case of imbalanced classes where the number of instances in each class significantly differs.

1.5.2. Precision

Precision is the ratio of true positives to the sum of true positives and false positives. In other words, it measures the proportion of positive identifications that were actually correct. Precision is especially valuable in situations where the costs of false positives are high. For instance, in loan default prediction, incorrectly predicting that a customer will default when they will not could lead to missed opportunities and revenue loss. The mathematical formula for precision is:

$$Precision = \frac{TP}{TP + FP}. \tag{1.5}$$

1.5.3. Recall

Recall, also known as sensitivity, hit rate, or true positive rate (TPR), is another essential metric. It measures the proportion of actual positives that are correctly identified. In terms of loan default prediction, recall would signify the model's ability to correctly identify those loans that will indeed default. This metric is particularly important when the cost of false negatives is high. Mathematically, recall is defined as:

$$Recall = \frac{TP}{TP + FN}. \tag{1.6}$$

1.5.4. F1 Score

The F1 score, also known as the harmonic mean of precision and recall, provides a balance between these two metrics. It is particularly useful when you want a measure that considers both false positives and false negatives equally, and there is an uneven class distribution. The F1 score is calculated as:

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}. \tag{1.7}$$

An F1 score of 1 indicates perfect precision and recall, while an F1 score of 0 means that either the precision or the recall is zero.

1.5.5. Area Under the ROC Curve (AUC-ROC)

The receiver operating characteristic (ROC) curve is a graphical representation that displays the performance of a binary classifier as its discrimination threshold changes. It plots the true positive rate (Recall) against the false positive rate. The area under the curve (AUC) is then calculated, providing a singular value that represents the model's performance. An AUC of 1.0 indicates a perfect classifier, while an AUC of 0.5 represents a model that performs no better than random classification. This metric is especially beneficial in binary classification problems like loan default prediction, as it provides a comprehensive view of model performance across all possible classification thresholds. There isn't a simple equation for AUC-ROC as it requires integrating the area under the ROC curve.

1.6. Conclusion

This chapter delivered a comprehensive overview of the various methods and models that have been traditionally employed for loan default prediction. These range from traditional statistical models to more advanced machine learning and deep learning techniques, each with their unique advantages and drawbacks.

A significant emphasis was placed on the need for interpretability in machine learning models. This is paramount in the field of finance, where comprehensible decision-making processes can impact trust and regulatory compliance. In response to this need, the TabNet architecture was introduced. TabNet strikes a balance between offering superior predictive performance and delivering interpretability, making it a promising choice for the task at hand.

Moreover, this chapter also discussed the importance of accurate model evaluation in the context of loan default prediction. Various performance metrics including accuracy, precision, recall, F1 score, and AUC-ROC were explored. Each of these metrics provides different insights into a model's performance, and their proper understanding and application are crucial for assessing the effectiveness of the predictive models.

2 | The Tabnet model

2.1. Introduction

In the TabNet paper, Arik et al. (2019) introduce a novel deep learning model specifically designed for tabular data analysis, addressing the limitations of traditional machine learning methods and other deep learning techniques in handling structured data. Tabular data, prevalent in various domains such as finance, healthcare, and social sciences, is often characterized by missing values, categorical variables, and complex relationships between features. While traditional methods such as Gradient Boosting Machines and Random Forests have been successful in addressing some of these challenges, they lack the expressive power and scalability offered by deep learning models.

TabNet aims to bridge this gap by introducing a unique architecture that combines the strengths of both traditional models and deep learning techniques. The core components of TabNet include learnable sparse feature selection, attention mechanisms, and end-to-end training. This combination enables TabNet to effectively process and analyze tabular data by prioritizing the most informative features, capturing complex patterns and dependencies, and handling categorical variables efficiently.

Arik et al. demonstrate that TabNet consistently outperforms traditional machine learning methods and other deep learning models on various benchmark datasets, highlighting its effectiveness in extracting valuable insights from structured data. Moreover, TabNet's architecture is scalable and amenable to parallel processing, making it suitable for large-scale datasets often encountered in real-world applications.

The main contributions as summarized in the paper are:

- TabNet inputs raw tabular data without any preprocessing and is trained using gradient descent-based optimization, enabling flexible integration into end-to-end learning.
- TabNet uses sequential attention to choose which features to reason from at each decision step, enabling interpretability and better learning as the learning capacity

is used for the most salient features . This feature selection is instance-wise, e.g. it can be different for each input, and unlike other instance-wise feature selection methods, TabNet employs a single deep learning architecture for feature selection and reasoning.

- Above design choices lead to two valuable properties: (i) TabNet outperforms or is on par with other tabular learning models on various datasets for classification and regression problems from different domains; and (ii) TabNet enables two kinds of interpretability: local interpretability that visualizes the importance of features and how they are combined, and global interpretability which quantifies the contribution of each feature to the trained model.
- Finally, for the first time for tabular data, we show significant performance improvements by using unsupervised pre-training to predict masked features.

2.2. The Architecture

In this section, we delve into the details of the TabNet model, discussing its architecture and the key components that contribute to its effectiveness in handling tabular data. TabNet, as introduced by Arik et al. (2019), is a deep neural network specifically designed for processing and analyzing structured data. The architecture of TabNet is comprised of the following primary components:

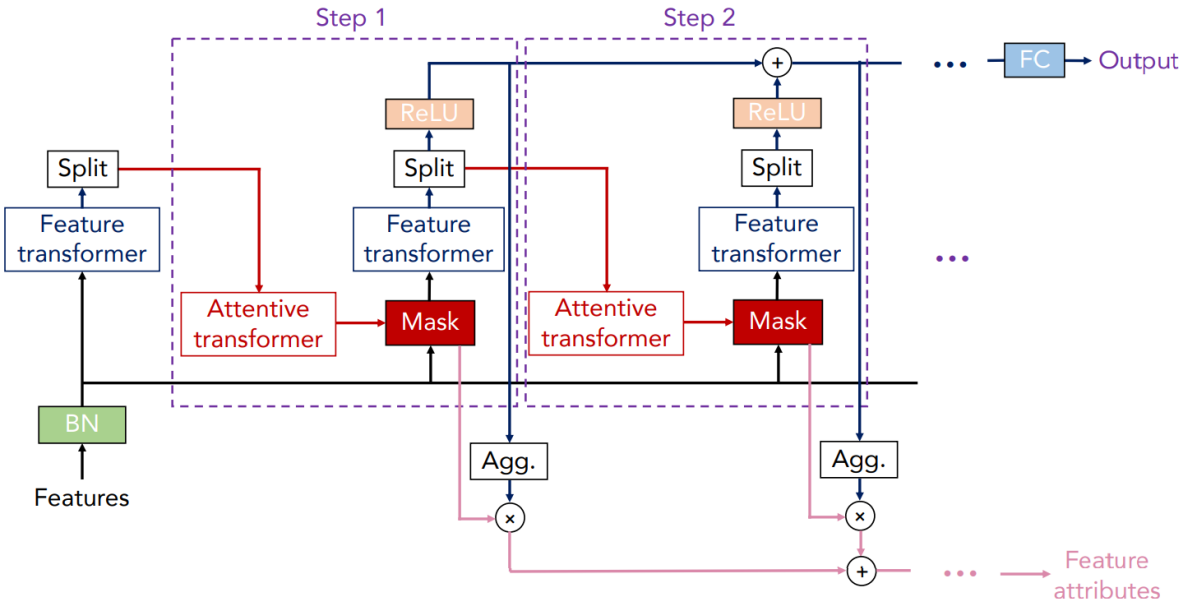

Figure 2.1: The Transformer Architecture

2.2.1. Feature Embeddings

TabNet's architecture begins with feature embeddings, which allow the model to efficiently handle both numerical and categorical variables within the raw tabular data. Each feature is passed through an embedding layer, transforming it into a continuous representation that can be effectively processed by the subsequent layers of the model. This approach eliminates the need for separate preprocessing steps and enables TabNet to handle mixed data types seamlessly. In the paper this layer is called feature transformer:

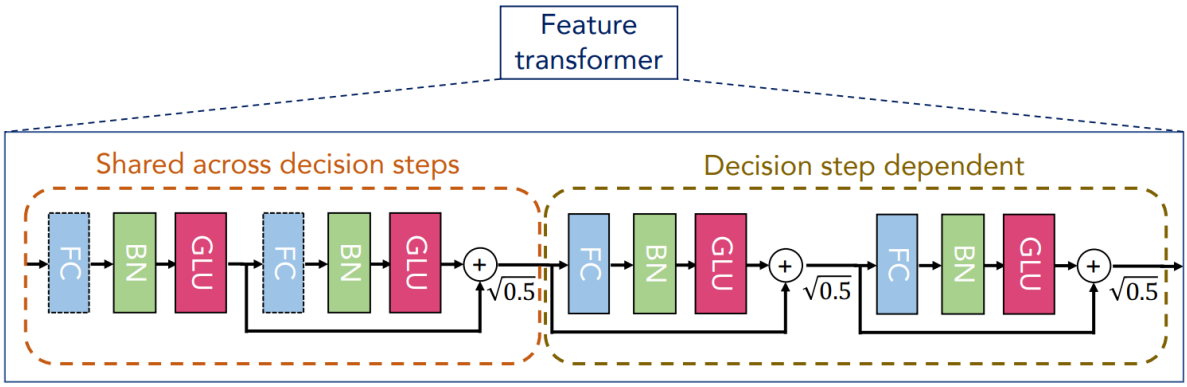

Figure 2.2: The Feature Transformer Architecture

2.2.2. Decision Step: Sequential Attention and Feature Selection

A key component of TabNet's architecture is its use of sequential attention and feature selection mechanisms. At each decision step, the model employs an attention mechanism to determine which features are the most salient for the current reasoning task. This instance-wise feature selection allows TabNet to focus its learning capacity on the most relevant features, thereby enhancing its interpretability and learning efficiency.

The attention mechanism in TabNet is implemented using a series of masked feature transformers, which are responsible for calculating the attention scores and applying masks to the input features. These masks are then used to modulate the feature importance, allowing the model to selectively focus on a subset of the input features at each decision step. Moreover these masks are responsible for the interpretability properties of Tabnet. The decision step is the modular part of the model and an arbitrary number of them can be concatenated at this phase.

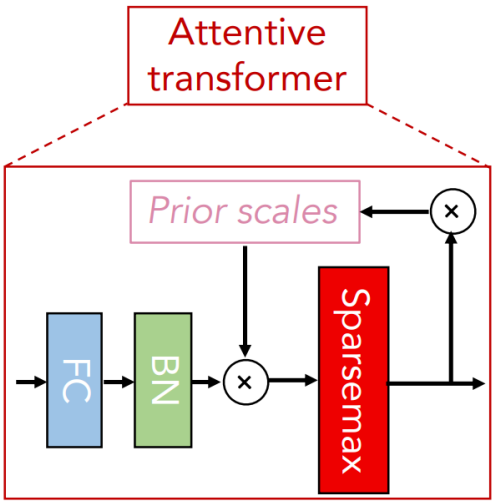

Figure 2.3: The Attentive Transformer Architecture

2.2.3. ReLU Activation and Element-wise Addition

After applying the attentive transformer, the feature masks and the feature transformers, the output is passed through a ReLU (Rectified Linear Unit) activation function. The ReLU activation function introduces non-linearity, allowing the model to learn more complex relationships between features.

Following the ReLU activation, an element-wise addition operation is performed to aggregate the intermediate representations generated at each decision step. This process effectively combines the information extracted from the most salient features at each step, producing a final representation that encompasses the relevant information from all decision steps.

2.2.4. Fully-Connected Layer (FC)

After the ReLU activation and the element-wise addition, TabNet incorporates a Fully Connected Layer within its architecture to model complex relationships between features and capture non-linear patterns in the data. The aggregated representation is processed by the Fully Connected Layer.

The depth and size of FC can be adjusted based on the complexity of the task and the available computational resources. This flexibility allows the model to adapt to various problem settings and data types effectively.

2.2.5. Output Layer

The final output representation produced by the FC layer is then passed through an output layer to generate the model's predictions. Depending on the task, the output layer can be a softmax layer for classification problems or a linear layer for regression problems.

2.2.6. Autoencoder and Unsupervised Training

Another significant aspect of the TabNet architecture is its ability to incorporate unsupervised training through the use of an autoencoder. Unsupervised training can lead to significant performance improvements by leveraging unlabeled data to learn meaningful representations of the input features before fine-tuning the model with supervised training on labeled data.

In the unsupervised setting, TabNet's architecture is modified to function as an autoencoder, which consists of an encoder and a decoder. The encoder maps the input features to a lower-dimensional latent space, while the decoder reconstructs the original input features from the latent representation. The primary objective in the unsupervised training phase is to minimize the reconstruction error, thereby forcing the model to learn a compact and informative representation of the input data.

To adapt TabNet for unsupervised training, a reconstruction loss is introduced into the model's objective function. The model's architecture remains largely unchanged, with the exception of the FC layer and the output layer, which is now designed to produce a reconstruction of the input features rather than a classification or regression output.

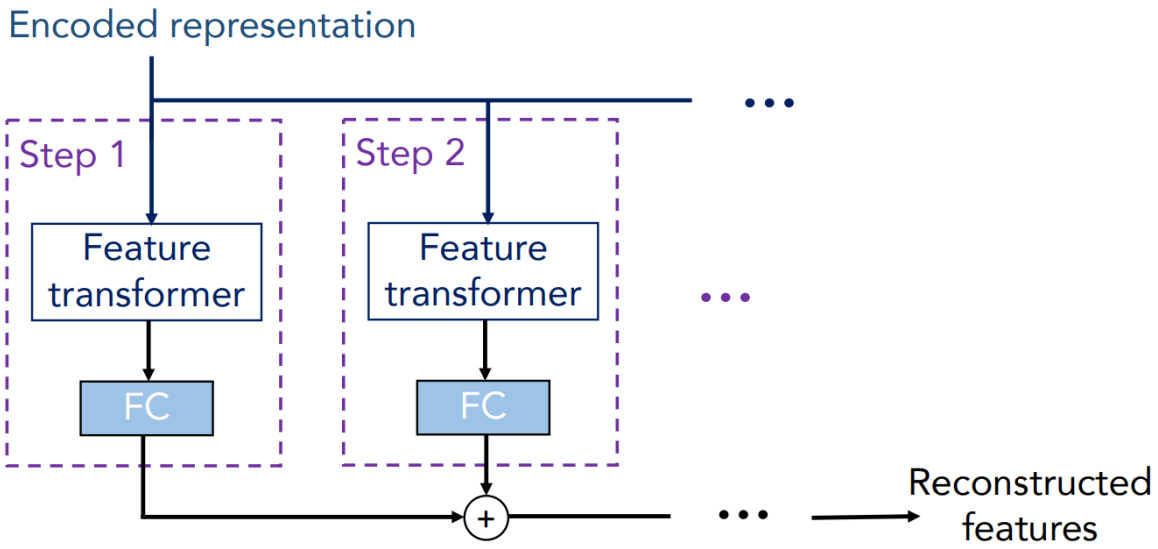

Figure 2.4: The Decoder Architecture

During unsupervised pre-training, the model learns to predict masked features from the remaining unmasked features. This process is inspired by the masked language modeling technique employed in the BERT architecture (Devlin et al., 2018). By predicting masked features, the model learns to capture the underlying structure and dependencies within the input data, even in the absence of labeled samples.

After the unsupervised pre-training phase, the autoencoder-based TabNet is fine-tuned using supervised training with labeled data. The output layer of the model is adjusted to produce classification or regression outputs, as required by the specific problem. By leveraging the pre-trained feature representations learned during the unsupervised phase, TabNet can achieve improved performance and generalization, particularly when the labeled data is scarce or noisy.

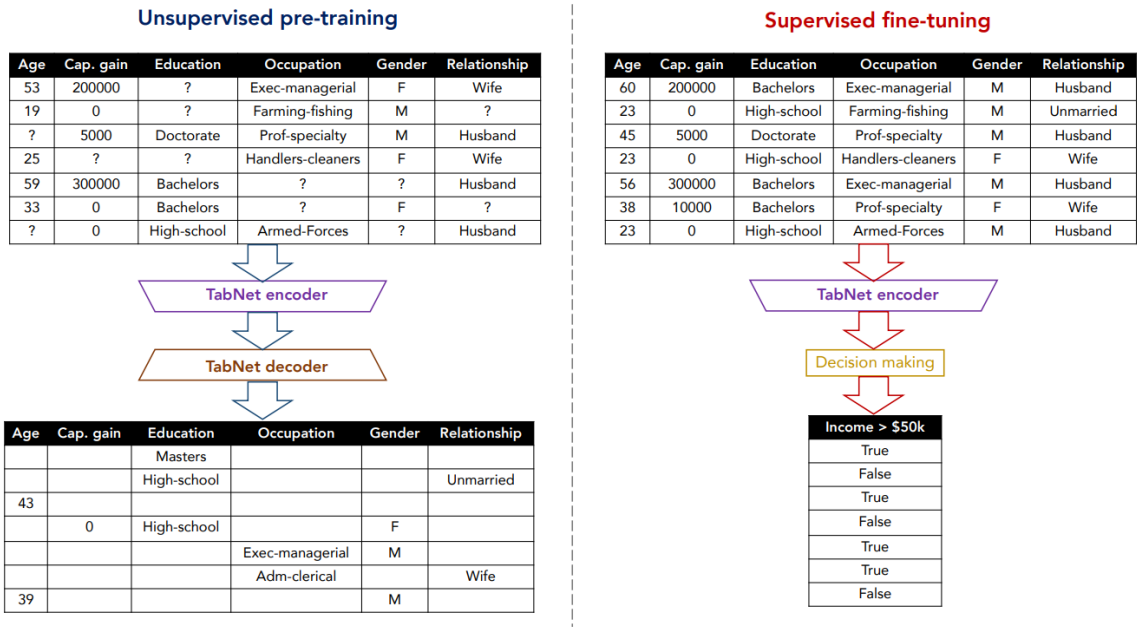

Figure 2.5: Example of Unsupervised and Supervised training phases

In summary, the integration of an autoencoder and unsupervised training into the TabNet architecture allows the model to benefit from both unlabeled and labeled data, enhancing its ability to learn meaningful feature representations and improving its performance on various tabular data tasks.

2.2.7. Summary

In conclusion, TabNet's architecture is a unique combination of feature embeddings, sequential attention mechanisms, masked feature transformers, ReLU activation, element-wise addition, fully-connected layer, and an output layer, specifically tailored to address the challenges associated with tabular data analysis. By leveraging these components, TabNet offers a powerful, scalable, and interpretable solution for processing and analyzing structured data, with competitive performance on various benchmark datasets.

3 | Dataset and Feature Preprocessing

3.1. The Lending Club Dataset

3.1.1. Overview

The Lending Club dataset is a comprehensive collection of loan data provided by Lending Club, the world's largest online lending platform. This dataset contains information on loans issued from 2007 to 2018, with details about borrowers, loan amounts, interest rates, loan grades, payment histories, and defaults. The dataset is widely used for credit risk modeling and loan default prediction, making it an ideal choice for this master thesis.

3.1.2. Previous Use Cases in Research

The Lending Club dataset has been widely used in academic research and industry applications for credit risk assessment and loan default prediction. Some notable studies and their key findings include:

Chen [7] addressed the issue of imbalanced datasets in loan default prediction. Their novel approach, applied to the Lending Club dataset, provides a roadmap for dealing with similar imbalance issues in future research.

Meanwhile, Malekipirbazari [14] proposed a risk assessment model combining Logistic Regression, Random Forests, and Gradient Boosting Machines. Their model yielded an 78% default prediction accuracy when applied to the Lending Club dataset.

Zhu [19] explores loan default prediction in P2P online lending platforms. The researchers use the Random Forest algorithm, real-world data from Lending Club, and the SMOTE method for class imbalance. After data cleaning and dimensionality reduction, the Random Forest algorithm shows superior performance in predicting default samples compared to other algorithms like logistic regression and decision trees.

These studies highlight the value of the Lending Club dataset in exploring various aspects of credit risk modeling and loan default prediction.

3.1.3. Dataset Structure

The foundation of our exploration into loan default prediction is the Lending Club dataset, encompassing data from 2007 to 2018. This rich dataset provides a comprehensive array of loan characteristics, borrower demographics, and loan performance, making it an ideal resource for building our predictive model.

The dataset contains an impressive total of 2,260,668 samples, each representing an individual loan issued within the specified period. These samples are characterized by 151 features, divided into 113 numerical and 38 categorical attributes. The information encapsulated within these features is broad, covering aspects such as borrower's income, employment status, loan purpose, and many more, providing a multifaceted perspective on loan default prediction.

One of the challenges associated with this dataset is the presence of missing data. Approximately 31.78% of the dataset contains missing values, necessitating strategic handling and pre-processing to ensure data integrity and robustness of the resulting model. We can see the extent of missing values in figure 3.1, where the black areas represent missing values in the dataset.

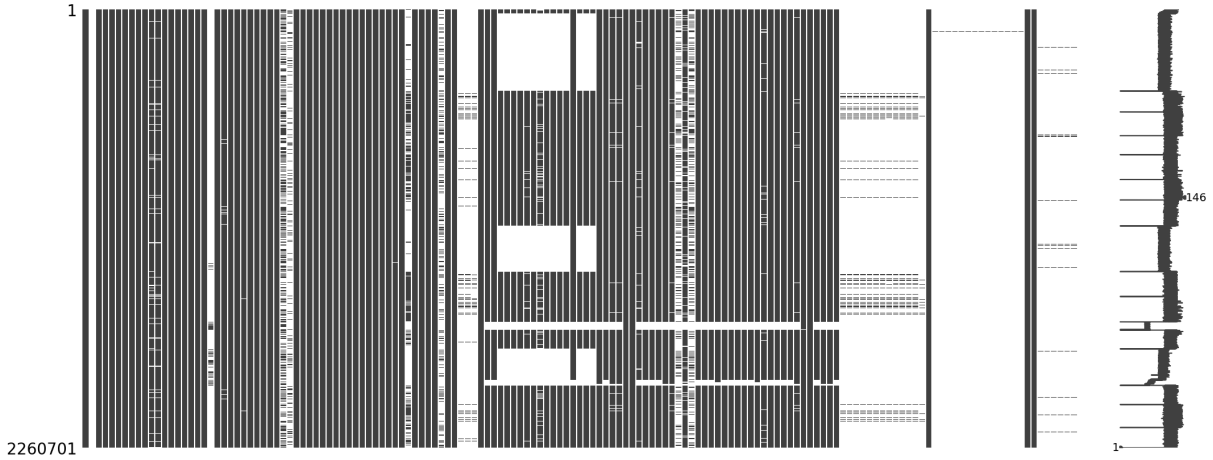

Figure 3.1: Missing Values displayed in a matrix

Among the numerical features, 'loan_amnt' is of particular interest as it represents the loan amount for each sample. It is directly tied to a borrower's financial obligation and their capacity to repay. The loan amount varies from as low as $500 to as high as $40,000,

with an average value of around $15,046.93. The standard deviation of approximately $9,190.25 reflects the substantial dispersion in loan amounts across borrowers, see figure 3.2.

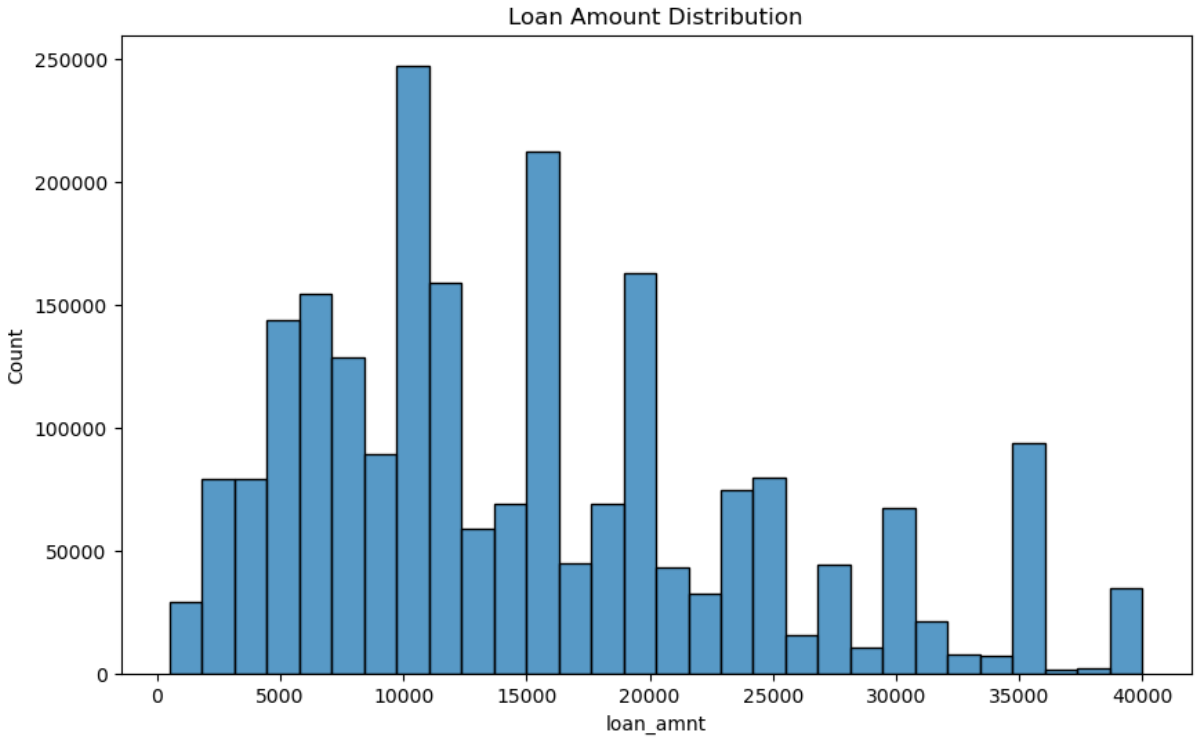

Figure 3.2: Loan Amount Distribution

Diving deeper, the loan amount is further dissected by the borrower's grade, categorized from A to G. The borrower's grade is an assessment of the borrower's creditworthiness, with A being the highest grade and G the lowest. Intriguingly, the mean loan amount tends to increase as the grade worsens. This observation underlines the riskier nature of lower-grade borrowers, who typically face higher interest rates due to increased credit risk, figure 3.3.

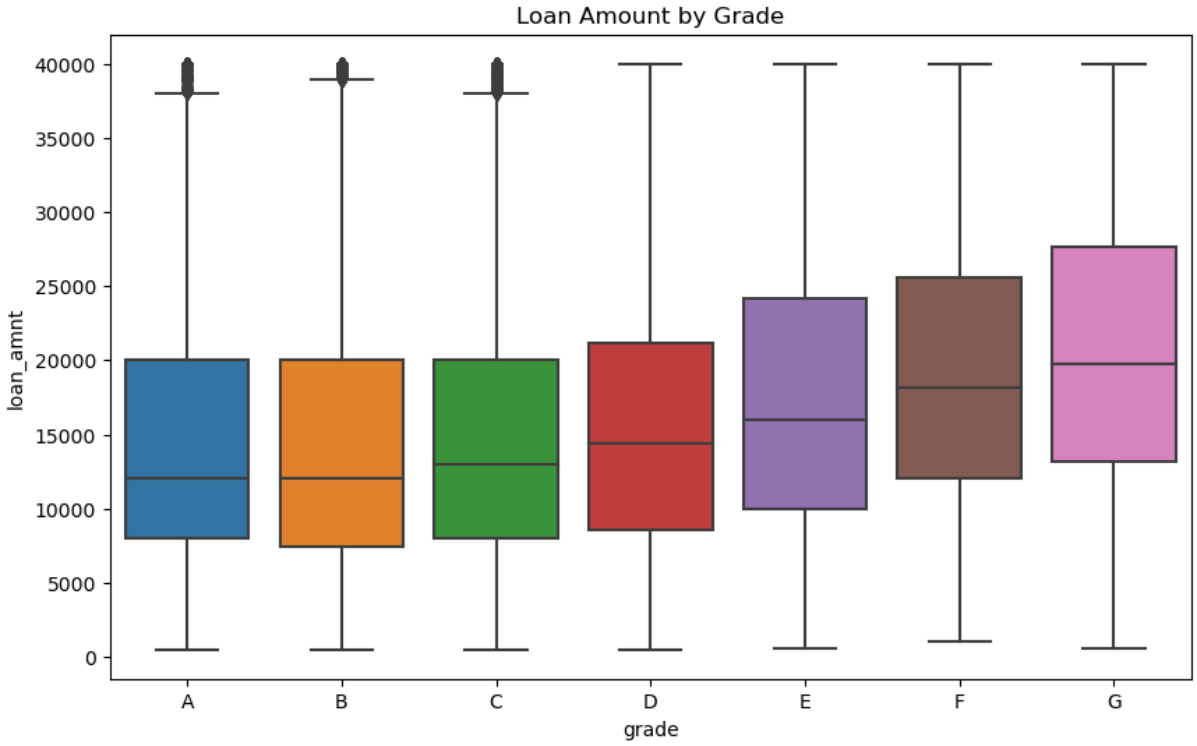

Figure 3.3: Loan Amount by Grade

Overall, despite its inherent challenges, this dataset offers a solid foundation for building a predictive model. By employing careful preprocessing and feature engineering, we can transform this raw data into a structured format suitable for the TabNet model, thereby facilitating the development of a reliable loan default prediction model.

3.2. Feature Preprocessing

3.2.1. Overview

Feature preprocessing is an essential step in the process of developing machine learning models. The primary purpose of preprocessing is to convert raw data into a format that can be efficiently used by machine learning algorithms. It can also help in improving the accuracy and performance of the models. In this chapter, we discuss the minimal feature preprocessing and data cleaning approaches employed in our study, with the goal of preserving the original real-world data.

3.2.2. Missing values

Missing values are a common occurrence in real-world datasets and can arise for various reasons, such as data entry errors, unavailability of information, or data collection issues. It is crucial to handle missing values appropriately during the preprocessing stage because their presence can lead to several challenges and adverse effects on machine learning models. The machine learning algorithms of our study are not designed to handle missing values and require complete data to function correctly. Feeding a dataset with missing values into these algorithms may result in errors or unexpected behavior [8]. For categorical features, missing values were replaced with the string 'missing,' while for numerical features, they were replaced with -1. This approach maintains the original data distribution and allows for easy interpretation of the results [8]. Furthermore, features with more than 90% missing values were removed from the dataset, as they do not contribute enough information for machine learning models and may introduce noise or increase the dimensionality of the dataset without providing meaningful insights [8].

3.2.3. Scaling and Encoding

The importance of scaling and encoding in preprocessing is crucial for various reasons. Appropriate scaling and encoding help improve the performance of machine learning models by ensuring that features are represented in a format suitable for the algorithm and contribute equally to the model training process. Scaling is particularly important for gradient-based optimization algorithms, as it can help improve the convergence speed and stability of the optimization process. Encoding categorical features as numerical values allows machine learning models to process the data more efficiently and enables better interpretability of the results.

Proper scaling and encoding can also lead to more efficient computation and faster model training, as they help reduce the dimensionality of the data and optimize the representation for machine learning algorithms. In order to ensure that all features contribute equally to the model, we standardized the numerical features using the StandardScaler from the sklearn.preprocessing module. This technique scales the features to have a mean of 0 and a standard deviation of 1, which helps in reducing the impact of differing units and magnitudes among features [10]. We didn't apply scaling of continuous features in the Tabnet preprocessing, since the model handles it autonomously. Categorical features were converted to numerical form using label encoding. This technique assigns a unique integer value to each category, which simplifies data representation and requires less memory compared to one-hot encoding.

3.2.4. Data Leakage

Data leakage is a critical issue in machine learning that occurs when information from the target variable, or information not available at the time of prediction, is inadvertently included in the training data. This can lead to overly optimistic performance metrics and models that fail to generalize well to new, unseen data. In our study, we used the Lending Club dataset of issued loans from 2007 to 2018, and it was essential to identify and remove features that could potentially cause data leakage.

The following features were removed due to their potential to leak information:

- 'out_prncp' and 'out_prncp_inv': These represent the outstanding principal and outstanding principal invested by investors for a loan, respectively. Both features provide information about the loan's future performance and are not available at the time of prediction.
- 'total_pymnt' and 'total_pymnt_inv': These indicate the total payment received to date and the total payment received to date by investors for a loan, respectively. Including these features in the model would provide information about the loan's future performance.
- 'total_rec_prncp', 'total_rec_int', 'total_rec_late_fee': These features represent the total received principal, total received interest, and total received late fees for a loan, respectively. They are related to the future performance of the loan and should not be included in the model training.
- 'recoveries' and 'collection_recovery_fee': These features represent the post-charge-off gross recovery and post-charge-off collection fee, respectively. They provide information on the loan's performance after it has already been charged off, which is not available at the time of prediction.
- 'last_pymnt_d' and 'last_pymnt_amnt': These features represent the date and amount of the most recent payment on the loan. Including these features in the model would leak information about the loan's future payment history.
- 'next_pymnt_d': This feature represents the scheduled date of the next payment for the loan. As this information is related to the future payment history of the loan, it should not be included in the model training.
- 'last_credit_pull_d': This feature represents the date when the most recent credit report was pulled for the borrower. Including this feature in the model would provide information about the borrower's future credit behavior.

- 'debt_settlement_flag', 'debt_settlement_flag_date', 'settlement_status', 'settlement_date', 'settlement_amount', 'settlement_percentage', 'settlement_term': These features are related to the debt settlement process, which occurs after the loan has already defaulted. Including these features would leak information about the loan's future performance.
- 'last_fico_range_high' and 'last_fico_range_low': These features represent the borrower's FICO score range at the time of the most recent credit report. Including these features in the model would provide information about the borrower's future credit behavior.

By removing these features, we mitigated the risk of data leakage in our study, ensuring that the developed machine learning models are trained on information that would be available at the time of prediction. This approach contributes to the development of models that generalize better to new, unseen data and provides a more accurate representation of their performance in real-world scenarios.

3.2.5. High-Dimensional Categorical Features

High dimensional categorical features can pose challenges in machine learning due to the large number of unique categories, which can lead to increased model complexity, overfitting, and computational inefficiency. Additionally, some high dimensional categorical features may be noisy or provide misleading information, further hampering the performance of machine learning models. In our study, we used the Lending Club dataset of accepted loans from 2007 to 2018 and removed several high dimensional categorical features deemed noisy or misleading.

The following features were removed due to their noisy nature:

- 'id': This feature represents the unique identifier for each loan. Including this feature in the model would not provide any meaningful information for prediction, as each loan has a distinct identifier that is unrelated to its performance.
- 'url': This feature represents the URL for each loan on the Lending Club website. Similar to the 'id' feature, the URL does not provide any valuable information for predicting loan performance and merely serves as a reference to the loan's online location.
- 'emp_title': This feature represents the job title provided by the borrower at the time of loan application. While employment information can be relevant for credit risk assessment, the 'emp_title' feature contains a vast number of unique job titles,

making it a high dimensional categorical feature. Moreover, the job titles may not be standardized, and borrowers may use various titles for the same or similar job positions, introducing noise and inconsistency in the data.

- 'title': This feature represents the loan purpose as described by the borrower. Although the loan purpose can be informative, the 'title' feature suffers from similar issues as the 'emp_title' feature, with a large number of unique and potentially inconsistent entries.

By removing these noisy high dimensional categorical features, we reduced the complexity of the dataset and ensured that our machine learning models focused on more meaningful and informative features for predicting loan performance. This approach contributes to the development of more accurate and interpretable models, while also reducing the risk of overfitting and improving computational efficiency.

3.2.6. Labels Cleaning and Encoding

In this subsection, we discuss the custom encoding of labels for the Lending Club dataset, which is used for loan default prediction. The purpose of this encoding process is to simplify the loan status labels, ensuring that the machine learning models can effectively learn from the data while providing interpretable and actionable insights.

Before proceeding with the custom encoding, we first filtered the dataset to include only the relevant rows for our study. The dataset was narrowed down to include loan records with the following loan statuses: 'Fully Paid,' 'Charged Off,' and 'Defaulted.' This step was performed to focus our analysis on the loans with clear outcomes, thereby enabling the development of accurate models for loan default prediction.

After filtering the dataset, we applied the following custom encoding to the 'loan_status' feature:

Algorithm 3.1 Custom Encoding for Loan Status

1: Create a custom encoding dictionary:
2: `'Fully Paid'` $\rightarrow 0$
3: `'Charged Off'` $\rightarrow 1$
4: `'Default'` $\rightarrow 1$

This custom encoding approach simplifies the loan status labels by mapping them to binary outcomes: 0 for 'Fully Paid' loans, indicating a successful loan repayment, and 1 for 'Charged Off' and 'Default' loans, indicating unsuccessful loan repayments. By

consolidating the 'Charged Off' and 'Default' statuses into a single category, we capture the essence of loan default prediction, focusing on the primary objective of identifying loans that are likely to default or not.

4 | Unsupervised and Supervised Training

4.1. Unsupervised Training

4.1.1. Overview

Unsupervised training of the TabNet model has demonstrated significant performance gains over gradient boosting methods for tabular data modeling. One of its main strengths lies in its ability to learn expressive feature representations from raw data without relying on labeled examples. In contrast, gradient boosting methods require handcrafted features and cannot automatically learn representations.

TabNet is also highly scalable and efficient, enabling it to handle large-scale datasets with ease. Its architecture incorporates sparse attention mechanisms that promote feature selection and reduce computation. Gradient boosting methods, particularly tree-based ones, are less scalable due to their sequential training process and tendency to overfit when dealing with high-dimensional data.

Unsupervised training of the TabNet model has shown remarkable robustness and generalization capabilities across various tasks and domains. The self-supervised pretraining technique encourages learning invariant features that are less susceptible to noise and other confounding factors, resulting in a more robust model. In comparison, gradient boosting methods often require extensive hyperparameter tuning and may be prone to overfitting. In the original Tabnet paper the advantages in performance for unsupervised training were shown in figure 4.1. With pre-training the model learns faster and achieves a better final accuracy :

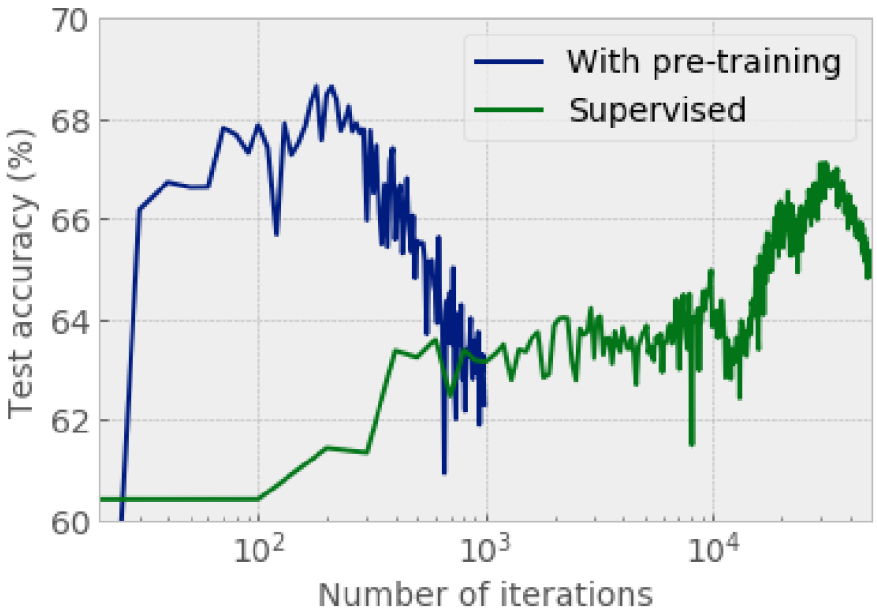

Figure 4.1: Test Accuracy with and without Pretraining

Lastly, the unsupervised training approach used by TabNet offers increased flexibility and adaptability. It can easily adapt to new tasks and domains by fine-tuning the pretrained model on a relatively small amount of labeled data, while gradient boosting methods typically require retraining from scratch and extensive feature engineering. In summary, unsupervised training of the TabNet model presents numerous advantages over gradient boosting methods, making it a promising approach for advancing the state of the art in tabular data modeling and analysis.

4.1.2. Hyperparameters

In the unsupervised training of the TabNet model, numerous hyperparameters must be tuned to attain optimal performance. To guide this tuning process, the original TabNet paper was taken as a foundational reference. A manual search over the hyperparameter space was then conducted, as it was deemed more feasible in terms of memory and computation resources compared to a random search. This decision was made to ensure the efficiency of the training process while adhering to the limitations imposed by the available computing infrastructure.

- **n_d and n_a (64 each)**: Define the dimensions of the decision and attention layers of the TabNet architecture, playing a crucial role in determining the model's capacity to capture complex patterns in the data.

Hyperparameter	Value
n_d	64
n_a	64
n_steps	5
gamma	1.6
n_independent	2
n_shared	2
optimizer_fn	torch.optim.Adam
optimizer_params	{lr: 2e-2, weight_decay: 1e-5}
mask_type	entmax
scheduler_fn	torch.optim.lr_scheduler.CosineAnnealingWarmRestarts

Table 4.1: Selected hyperparameters for unsupervised training of the TabNet model.

- **n_steps (5)**: Represents the number of decision steps in the TabNet model, influencing its ability to learn hierarchical representations.
- **gamma (1.6)**: Is the scaling factor for feature sparsity in the attention mechanism, encouraging the model to select a smaller set of relevant features at each decision step.
- **n_independent (2) and n_shared (2)**: Are the number of independent and shared feature transformers, respectively, controlling the balance between model complexity and sharing of learned representations across decision steps.
- **optimizer_fn (torch.optim.Adam)** and **optimizer_params (learning rate: 2e-2, weight_decay: 1e-5)**: Define the optimization algorithm and its parameters, impacting the convergence and generalization of the model during training.
- **mask_type ('entmax')**: Specifies the attention mechanism's sparsity-inducing function, with entmax promoting sparse and interpretable attention maps.
- **scheduler_fn (torch.optim.lr_scheduler.CosineAnnealingWarmRestarts)** and **scheduler_params**: Control the learning rate schedule, enabling the model to explore different regions of the loss landscape and potentially find better minima.

4.2. Supervised Training

This chapter focuses on the application of supervised learning techniques to predict loan defaults. The models utilized in this study include LightGBM, XGBoost, and TabNet, which have emerged as popular and efficient techniques for handling structured data in machine learning.

The dataset used in this study consists of diverse features related to loan applications. To ensure the robustness and generalizability of the predictive models, the dataset was divided into three distinct sets: a training set (60%), a validation set (20%), and a test set (20%).

The decision to split the data in this way adheres to the standard practice in machine learning research. The training set is used to build and train the models, the validation set is used to tune the parameters and prevent overfitting, and the test set is used to evaluate the final model's performance, providing an unbiased assessment of how well the model would perform on unseen data.

It is important to note that only the training set underwent undersampling, for a few reasons. Firstly, this process can help the model to learn a decision boundary that is not biased towards the majority class, which should improve its ability to predict the minority class.

Secondly, the validation and test sets should ideally reflect the real-world distribution of the data. Applying undersampling to these sets would distort their class distributions, leading to an overly optimistic estimate of the model's performance. As a result, the validation and test sets were left untouched to provide a more realistic evaluation of the model's ability to predict loan defaults.

4.2.1. LightGBM Training

Overview

After a thorough evaluation of various machine learning algorithms, LightGBM was selected for loan default prediction due to its superior performance in handling large-scale datasets with a high dimensionality, coupled with its efficient gradient boosting framework that effectively mitigates overfitting and expedites model training. To further enhance its predictive capabilities for loan default outcomes, the binary log loss function was employed as the evaluation metric, which was chosen for its ability to quantify the deviation between predicted probabilities and true binary labels, providing a comprehensive and

interpretable assessment of the model's performance. In addition to its inherent advantages, the hyperparameters for LightGBM were carefully optimized using random search, ensuring the best possible model configuration. The training loss followed this plot:

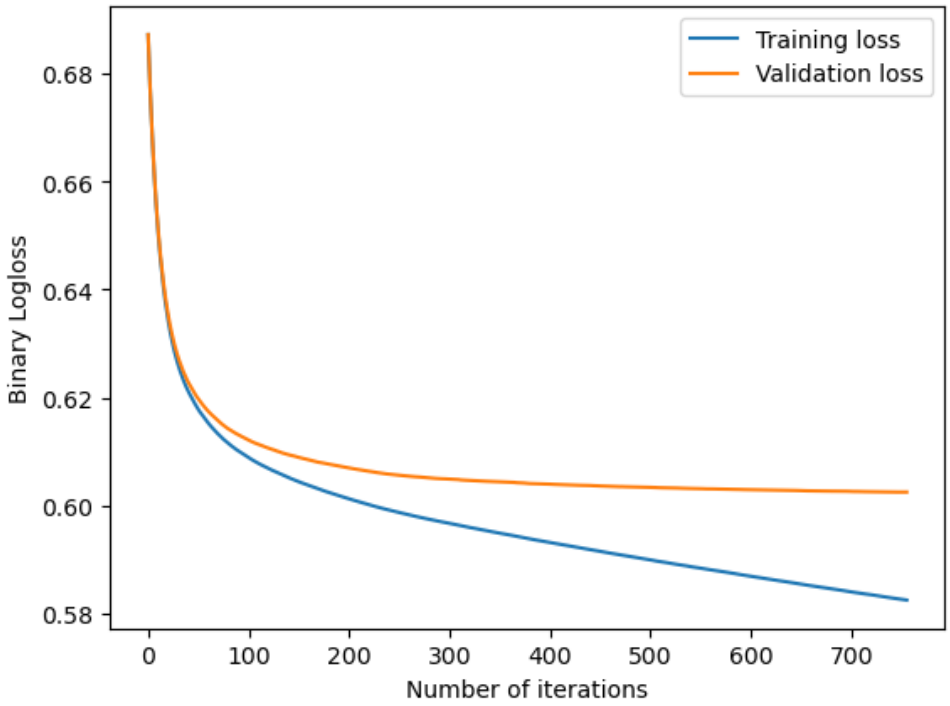

Figure 4.2: LightGBM training and validation loss

Evaluation Metrics

In this section, we present the evaluation metrics obtained from our LightGBM model, providing a comprehensive view of its performance. These metrics include accuracy, precision, recall, F1-score, and AUC-ROC score. Accuracy measures the proportion of correctly classified instances out of the total instances. Precision is the proportion of true positives among the positive predictions, while recall is the proportion of true positives among all actual positive instances. F1-score is the harmonic mean of precision and recall, providing a balanced evaluation metric. Lastly, the AUC-ROC score represents the area under the Receiver Operating Characteristic curve, which illustrates the trade-off between the true positive rate and false positive rate.

Below table 4.2 summarizes the performance of our model on both training and test sets, with values truncated to the third decimal point:

The LightGBM model exhibits a validation and test accuracy of 0.660, with a relatively lower precision of 0.332. This suggests that while the model is moderately accurate,

Metric	Value
Validation Accuracy	0.660
Test Accuracy	0.660
Test Precision	0.332
Test Recall	0.693
Test F1-score	0.449
Test AUC-ROC score	0.738

Table 4.2: LightGBM Evaluation Metrics

it may be predicting a significant number of false positives. The model demonstrates a reasonably high recall score of 0.693, indicating a good ability to correctly identify positive instances. The F1-score, which combines precision and recall into a single metric, is 0.449, representing a balanced harmonic mean of the two metrics. The AUC-ROC score, a metric indicative of the model's ability to distinguish between classes, stands at a reasonably strong 0.738.

4.2.2. XGBoost Training

Overview

XGBoost is another state of the art model in the loan prediction landscape, owing to its exceptional performance in managing large-scale datasets with high dimensionality, in tandem with its powerful gradient boosting framework that effectively minimizes overfitting and accelerates model training. To further bolster its predictive prowess for loan default outcomes, the binary log loss function was utilized as the evaluation metric, which was selected for its capacity to measure the discrepancy between predicted probabilities and true binary labels, offering a thorough and intelligible appraisal of the model's performance. Alongside its intrinsic benefits, the hyperparameters for XGBoost were diligently optimized using random search, guaranteeing the optimal model configuration. The training and validation loss converged smoothly as you can see in figure 4.3

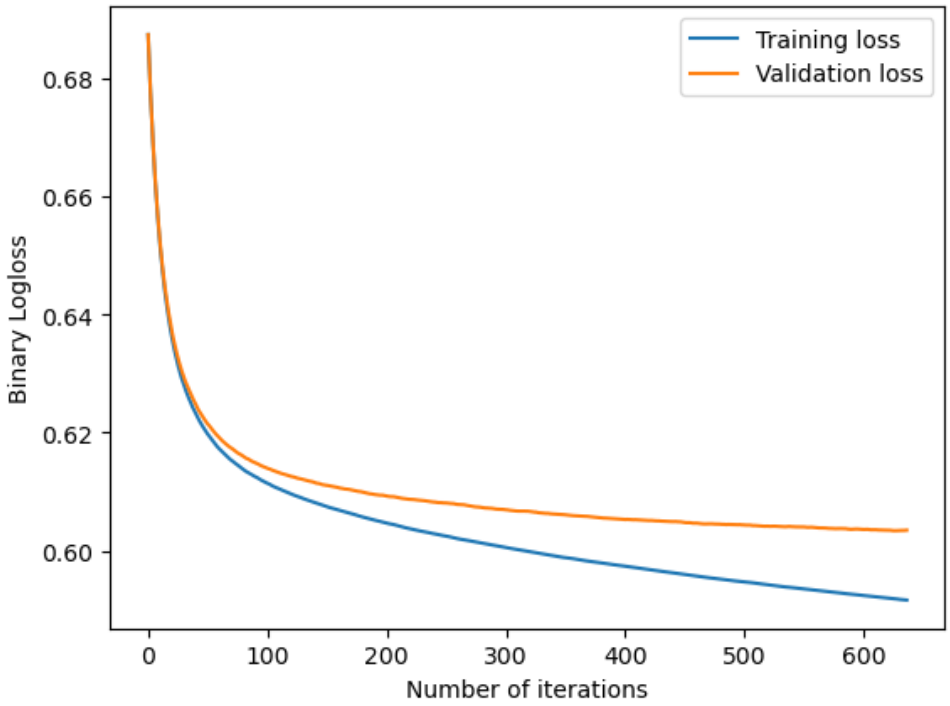

Figure 4.3: XGBoost training and validation loss

Evaluation Metrics

The same evaluation metrics are chosen for all models. The performance of XGBoost is the following:

Metric	Value
Validation Accuracy	0.660
Test Accuracy	0.661
Test Precision	0.333
Test Recall	0.691
Test F1-score	0.449
Test AUC-ROC score	0.737

Table 4.3: XGBoost Evaluation Metrics

The XGBoost model, like the previous LightGBM, also shows a similar validation and test accuracy of around 0.660. Precision is slightly improved at 0.333, but still suggests a substantial rate of false positives. The recall is marginally lower at 0.691, hinting at a slight decrease in the model's ability to correctly identify positive instances compared to LightGBM. The F1-score remains identical at 0.449, reflecting similar trade-offs between precision and recall. The AUC-ROC score is slightly lower than the previous model at 0.737.

4.2.3. Tabnet Training

Overview

In the training phase, TabNet exhibited different characteristics compared to LightGBM and XGBoost. The most notable aspect was the non-monotonic trend observed in the loss reduction during the training process. In contrast to the relatively smoother and more consistent loss decrease seen in LightGBM and XGBoost, the loss function for TabNet displayed more fluctuations, see figure 4.4.

This behavior can be attributed to the unique architecture of TabNet, which incorporates feature selection in a different way compared to traditional gradient boosting methods. The use of sparse attention mechanisms in TabNet can lead to such non-monotonic trends in loss reduction, as the model learns to select and focus on different subsets of features in different iterations.

Despite this irregularity, it should be noted that a non-monotonic loss decrease does not necessarily indicate a problem with the model's learning process. The final model performance should be evaluated based on the prediction accuracy on the validation set, rather than the smoothness of the training loss decrease.

Another noteworthy point was the increased training time of TabNet compared to LightGBM and XGBoost. TabNet's architecture is more complex and computationally intensive due to its use of self-attention mechanisms. These mechanisms enable it to capture complex interactions between features, which can be highly beneficial for prediction accuracy, but at the cost of increased computational time.

Despite the longer training time and the non-monotonic loss decrease, TabNet has been recognized for its ability to provide interpretable predictions and handle high-dimensional datasets effectively. The following sections will explore the prediction results and compare the performance of the three models in more detail.

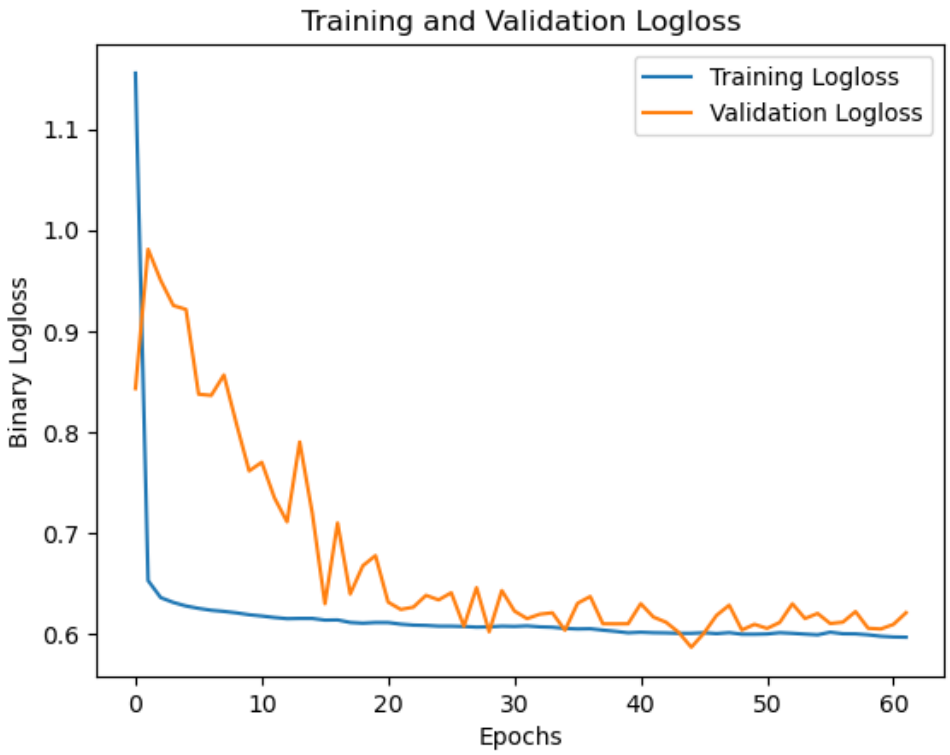

Figure 4.4: TabNet training and validation loss

Evaluation Metrics

The same evaluation metrics are chosen for all models. The performance of TabNet is the following:

Metric	Value
Validation Accuracy	0.675
Test Accuracy	0.673
Test Precision	0.334
Test Recall	0.641
Test F1-score	0.439
Test AUC-ROC score	0.723

Table 4.4: TabNet Evaluation Metrics

The TabNet model displays an improvement in validation and test accuracy, reaching values of 0.675 and 0.673, respectively. It also presents a slight improvement in precision to 0.334, which suggests a relatively lower rate of false positives compared to the previous models. Interestingly, the recall drops significantly to 0.641, suggesting that while overall accuracy has increased, the model's ability to correctly identify all positive instances has declined. The F1-score is marginally lower at 0.439, hinting at the effect of the lower

recall. Despite an increased accuracy, the AUC-ROC score drops to 0.723, suggesting a decreased ability in distinguishing between classes compared to both LightGBM and XGBoost.

4.2.4. Random Forest Training

Overview

The Random Forest model is another powerful ensemble machine learning algorithm that was utilized for loan default prediction. Random Forest employs multiple decision trees to make predictions, and it is less prone to overfitting compared to individual decision trees, due to the randomness introduced in both feature selection and data sampling.

The model was trained using Gini impurity as the criterion for splitting nodes, a choice that was made for its ability to maximize the gain in information. The hyperparameters such as the number of trees, maximum depth of the trees, and the minimum samples required to split a node were optimized using random search to achieve the best possible model configuration.

Notably, Random Forest can handle a high dimensionality dataset without the need for explicit feature selection or scaling, making it particularly useful for this application. Also, it offers a form of interpretability through the importance score assigned to each feature based on their contribution to the model's prediction, although the interactions between features can be harder to interpret than in a model like logistic regression.

Evaluation Metrics

The Random Forest performance is the following:

Metric	Value
Validation Accuracy	0.654
Test Accuracy	0.653
Test Precision	0.323
Test Recall	0.668
Test F1-score	0.435
Test AUC-ROC score	0.659

Table 4.5: Random Forest Evaluation Metrics

The Random Forest model demonstrates a slightly lower validation and test accuracy

compared to the previous models, coming in at 0.654 and 0.653 respectively. The precision also dips to 0.323, hinting at an increased number of false positives. The recall score of 0.668 is higher than TabNet but lower than LightGBM and XGBoost, indicating a decreased but still reasonable ability to find all the positive samples. The F1-score, at 0.435, reflects the lower precision and recall. Furthermore, the AUC-ROC score significantly drops to 0.659, suggesting that the Random Forest model has a substantially lower ability to distinguish between classes compared to the previous models.

4.2.5. Logistic Regression Training

Overview

The logistic regression model, despite being a relatively simpler model compared to the others, offers unique advantages. This model is highly interpretable and provides direct probabilities for the outcome variable. In the context of loan default prediction, logistic regression was trained on the feature matrix by using a sigmoid function that maps any real-valued number into the range [0,1]. The binary cross-entropy loss function was employed as the evaluation metric, which, similar to the binary log loss function, quantifies the difference between the predicted probabilities and the true binary labels.

The strength of logistic regression lies in its simplicity and interpretability. It offers a clear understanding of the relationship between each independent variable and the outcome, as the coefficients can be directly used to quantify the effect of a unit change in the independent variable on the odds of defaulting. However, it should be noted that logistic regression may fail to capture more complex non-linear relationships between variables.

Unlike the gradient boosting methods, logistic regression does not necessitate extensive hyperparameter tuning, thereby reducing the training time. However, feature engineering, such as handling of categorical variables and feature scaling, is crucial in this approach to ensure model accuracy and robustness.

Evaluation Metrics

The Logistic Regression performance is the following:

The Logistic Regression model presents a validation and test accuracy of 0.663, which is a slight improvement over the Random Forest model but still lower than the TabNet model. The precision stands at 0.329, which is slightly higher than the Random Forest

Metric	Value
Validation Accuracy	0.663
Test Accuracy	0.663
Test Precision	0.329
Test Recall	0.656
Test F1-score	0.438
Test AUC-ROC score	0.660

Table 4.6: Logistic Regression Evaluation Metrics

model, suggesting a minor reduction in false positives. The recall score drops slightly to 0.656, indicating a marginal reduction in the model's ability to find all the positive samples. The F1-score of 0.438 is similar to the TabNet and Random Forest models, and the AUC-ROC score of 0.660 is only marginally better than the Random Forest model, suggesting a limited ability to distinguish between classes.

4.3. Overall Results

Metric	LightGBM	XGBoost	TabNet	Random Forest	Logit
Validation Accuracy	0.660	0.660	**0.675**	0.654	0.663
Test Accuracy	0.660	0.661	**0.673**	0.653	0.663
Test Precision	0.332	0.333	**0.334**	0.323	0.329
Test Recall	**0.693**	0.691	0.641	0.668	0.656
Test F1-score	**0.449**	**0.449**	0.439	0.435	0.438
Test AUC-ROC score	**0.738**	0.737	0.723	0.659	0.660

Table 4.7: Comparison of Evaluation Metrics for LightGBM, XGBoost, TabNet, Random Forest, and Logistic Regression

This chapter presents a comparative analysis of the performance of five machine learning models - LightGBM, XGBoost, TabNet, Random Forest, and Logistic Regression - trained on a loan default prediction task. The evaluation metrics considered, as detailed in Table 4.7, include Validation Accuracy, Test Accuracy, Test Precision, Test Recall, Test F1-score, and Test AUC-ROC score.

Starting with Validation Accuracy, TabNet outperforms all other models, delivering a score of 0.675. This suggests that during the validation phase, TabNet was able to cor-

rectly classify a larger proportion of instances compared to the rest of the models. It is closely followed by the Logistic Regression model with a score of 0.663, and then LightGBM and XGBoost both at 0.660, while Random Forest lagged slightly with 0.654.

The Test Accuracy of TabNet remains the best among all models with a score of 0.673, meaning that it has the highest proportion of correct predictions on the test dataset. Logistic Regression came next with a score of 0.663, while Random Forest scored the lowest with 0.653.

In terms of Test Precision, which measures the proportion of positive identifications that were actually correct, TabNet still stands out with a score of 0.334, followed closely by XGBoost and LightGBM. This implies that when TabNet predicts a loan will default, it is slightly more likely to be correct than when other models make the same prediction.

However, in the Test Recall metric, which quantifies the model's ability to find all the positive instances, LightGBM leads with a score of 0.693. Despite TabNet's superior precision, it falls behind when it comes to correctly identifying all actual positives, with Random Forest and Logistic Regression also performing better than TabNet in this regard.

For the Test F1-score, which provides a balance between Precision and Recall, LightGBM and XGBoost come out on top with a score of 0.449, indicating an effective balance between identifying positive instances (recall) and limiting false positives (precision). Meanwhile, TabNet, Logistic Regression, and Random Forest show similar performance, scoring 0.439, 0.438, and 0.435, respectively.

The Test AUC-ROC score, a critical metric that evaluates the model's ability to distinguish between positive and negative classes, is highest for LightGBM with a score of 0.738. XGBoost closely follows with a score of 0.737, and TabNet is slightly behind with 0.723. Both Random Forest and Logistic Regression demonstrate notably lower AUC-ROC scores at 0.659 and 0.660, respectively.

In summary, while each model has areas of relative strength, TabNet shows the highest scores in Validation Accuracy, Test Accuracy, and Test Precision, indicating a particular strength in reducing false positives. Despite trailing in Test Recall, Test F1-score, and Test AUC-ROC score, it still demonstrates competitive performance. However, the lower scores of Random Forest and Logistic Regression, particularly in the Test AUC-ROC score, indicate limitations in their ability to distinguish between loan default and non-default cases.

These results are particularly significant given that TabNet was designed with an emphasis on model interpretability without compromising on performance. Therefore, while it

might not lead in every metric, its competitive performance across the board combined with its superior interpretability makes TabNet a compelling choice for tasks that require both high performance and interpretability, such as loan default prediction.

This analysis provides valuable insights for both academics and industry professionals interested in applying machine learning models for loan default prediction, and highlights the competitive performance and interpretability that TabNet brings to the table.

5 | Tabnet Interpretability Analysis

5.1. Global Interpretability

The global interpretability analysis provides an all-encompassing view of the feature importance in the prediction process of the model. While usually this is achieved by measuring the extent to which variations in the input features' values impact the model's output, TabNet has a unique advantage when it comes to interpretability. Its attention mechanism allows us to track which features the model is focusing on at each decision step, providing transparency into the model's internal workings. The mask values, ranging from 0 to 1, indicate the degree of attention given to each feature at each decision step. In this context, we investigate the global importance of features in the TabNet model, trained on the Lending Club dataset.

In the TabNet model, the authors introduced an aggregate feature importance measure that provides insights into the model's decision-making process. This is achieved by combining feature selection masks at different decision steps.

The formula for the aggregate decision contribution at the i^{th} decision step for the b^{th} sample is given by:

$$\eta_b[i] = \sum_{c=1}^{N_d} ReLU(d_{b,c}[i]), \tag{5.1}$$

Where:

- $\eta_b[i]$ represents the aggregate decision contribution.
- $ReLU(d_{b,c}[i])$ is the application of the Rectified Linear Unit (ReLU) activation function on the decision step output $d_{b,c}[i]$. The ReLU function is defined as $ReLU(x) = max(0, x)$.

Therefore, if $d_{b,c}[i]$ is less than 0, then the output is 0, indicating that all features at the i^{th} decision step should have zero contribution to the overall decision. However, as $d_{b,c}[i]$ increases, it plays a greater role in the overall decision.

The aggregate decision contribution $\eta_b[i]$ is then used to scale the decision mask at each decision step, resulting in an aggregate feature importance mask. This mask provides a measure of the importance of each feature in the model's decision-making process.

The goal of creating such an aggregate feature importance mask is to produce a model that is interpretable and provides insights into which features are most influential in the final decision-making process of the TabNet model.

The aggregate feature importance mask is proposed in the paper as a way to weigh the relative importance of each feature in the decision-making process. The formula is given as:

$$M_{agg-b,j} = \frac{\sum_{i=1}^{N_{steps}} \eta_b[i] M_{b,j}[i]}{\sum_{j=1}^{D} \sum_{i=1}^{N_{steps}} \eta_b[i] M_{b,j}[i]}, \tag{5.2}$$

Where:

- $M_{agg-b,j}$ represents the aggregate feature importance mask for the j^{th} feature of the b^{th} sample.
- $\eta_b[i]$ is the aggregate decision contribution at the i^{th} decision step for the b^{th} sample, which we previously defined.
- $M_{b,j}[i]$ is the feature selection mask for the j^{th} feature of the b^{th} sample at the i^{th} decision step.
- D is the total number of features, and N_{steps} is the total number of decision steps.

The numerator of the equation is the sum of the product of the aggregate decision contribution and the feature selection mask for each decision step. This value is then normalized by the sum of these products over all features and decision steps, which is the denominator of the equation.

This aggregate feature importance mask $M_{agg-b,j}$ provides a normalized measure of how much each feature contributes to the model's decision-making process across all decision steps, for each sample.

Finally, to obtain a global feature importance measure, we sum the aggregate feature importance masks across all samples and normalize by 1000. This gives a measure that

reflects the average importance of each feature across all samples in the dataset, thereby providing a global view of feature importance. The formula for this is given as:

$$G_j = \frac{1}{1000} \sum_{b=1}^{N_{samples}} M_{agg-b,j}, \tag{5.3}$$

Where:

- G_j represents the global feature importance for the j^{th} feature.
- $M_{agg-b,j}$ is the aggregate feature importance mask for the j^{th} feature of the b^{th} sample, which we previously defined.
- $N_{samples}$ is total number of samples

With this global feature importance measure G_j, we can gain insights into the overall importance of each feature in the TabNet model's decision-making process, taking into account its behavior across a large number of samples. This aids in a better understanding of the model's behavior and can help in feature selection and engineering for further improvements in model performance.

A bar plot was constructed to visually represent the global feature importance, revealing a degree of sparsity.

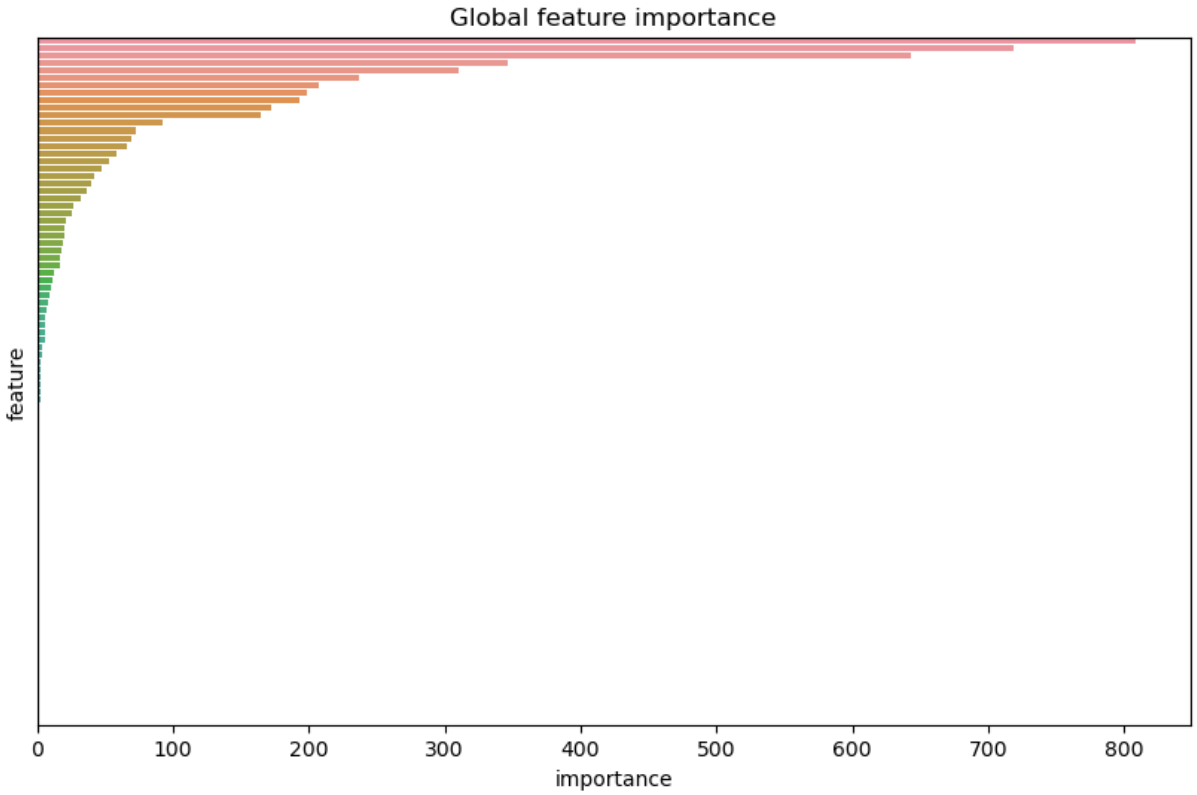

Figure 5.1: Features Global Importance

A small set of features was found to account for a significant portion of the variance in the model's output. In order to delve into the analysis the 30 most influential global features have been plotted:

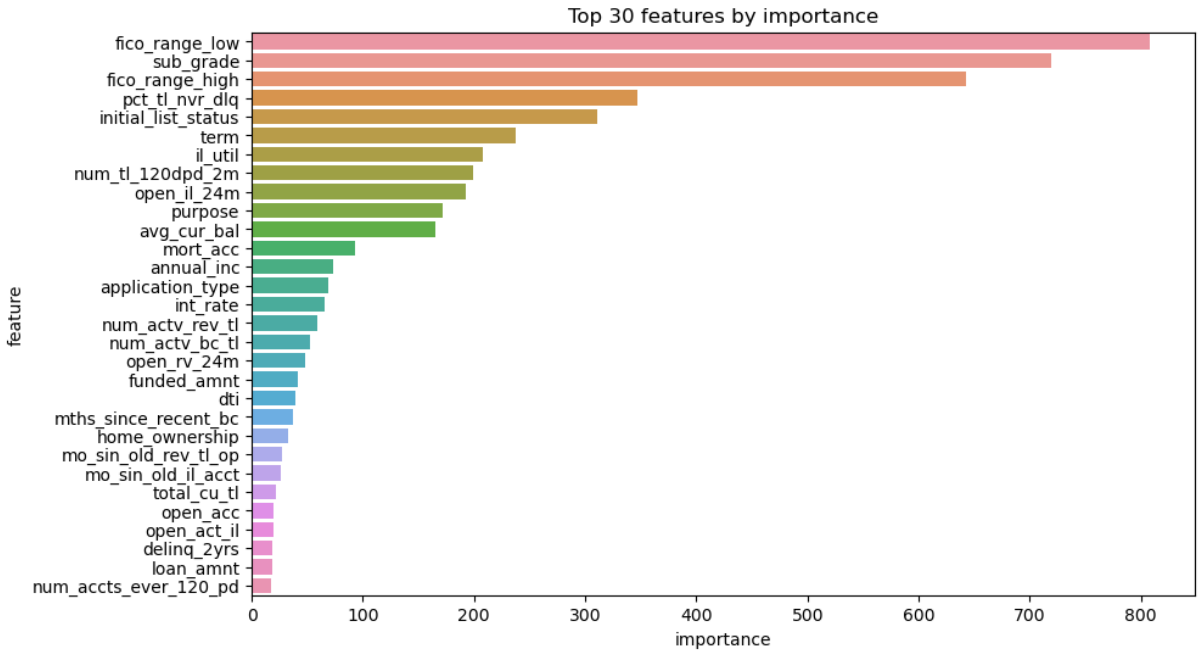

Figure 5.2: Top 30 Features Global Importance

From this plot, we can see that FICO Range (both high and low), Sub Grade, and the percentage of trades never delinquent are the most important features in predicting loan defaults according to the TabNet model. This insight aligns with common financial wisdom, as credit scores and past payment behavior are often the most significant factors in evaluating creditworthiness.

Additionally, it is worth noting that loan characteristics like term, interest rate, and loan amount, while important, seem to have less influence on the model's predictions compared to the borrower's credit profile. This observation highlights the emphasis placed by the model on the borrower's ability to repay, as opposed to the loan's characteristics.

5.1.1. Feature Importance at Each Decision Step

Beyond the global importance, we analyzed the importance of each feature at each decision step. A heatmap was created to visualize the average mask of each feature across the decision steps in TabNet. The heatmap revealed a varying distribution of feature importance across the different decision steps, reflecting the dynamic attention mechanism of TabNet.

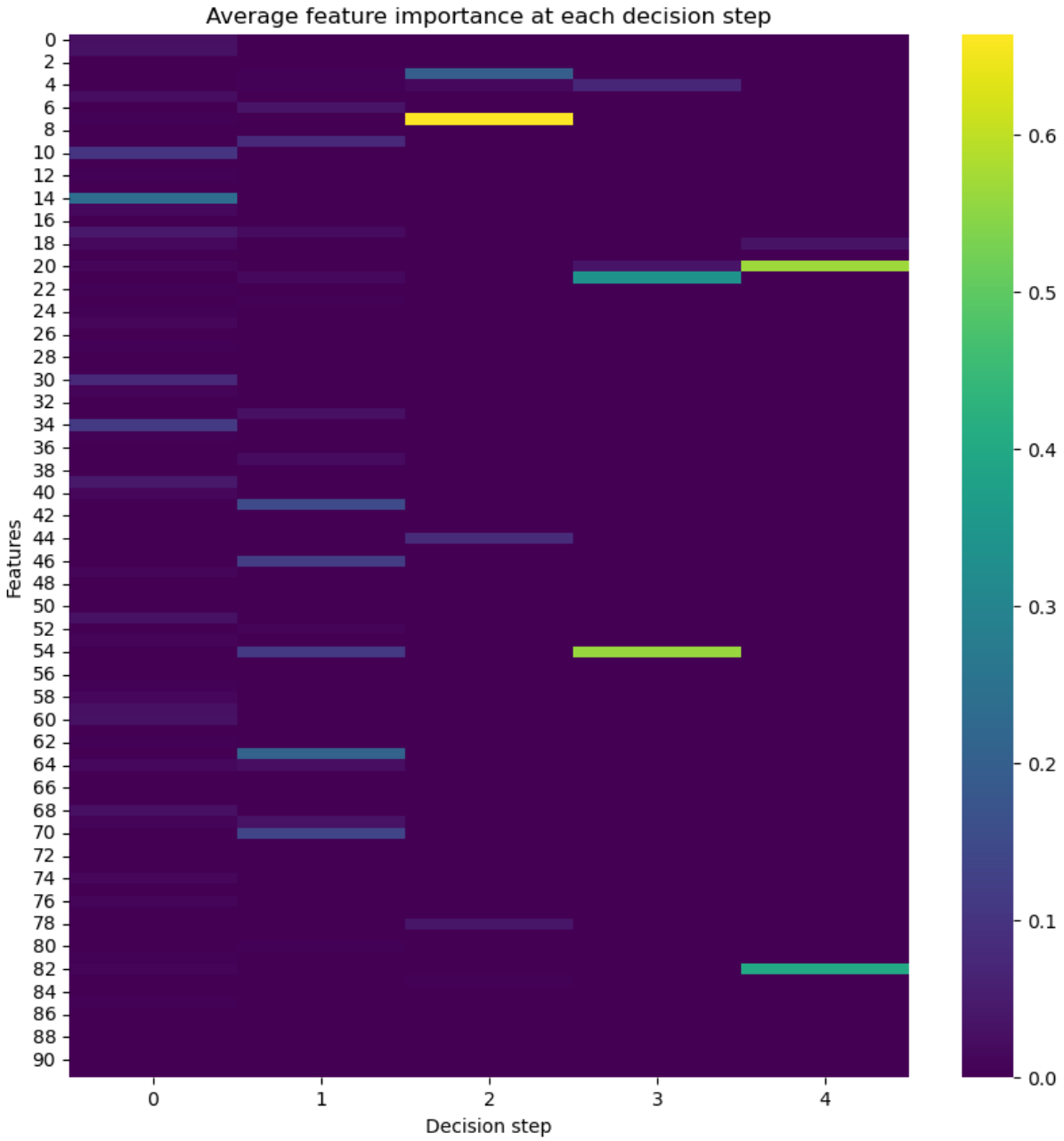

Figure 5.3: Average Masks at Each Decision Step

This distribution of feature importance across the decision steps further showcases TabNet's unique strength in interpretability. Unlike traditional tree-based models, which split on features based on a static hierarchy, TabNet dynamically adjusts the importance of features at each decision step, providing a more nuanced and flexible representation of complex relationships in the data.

For a more detailed analysis, we further investigated the top 10 features for each decision

step. The specific features and their importance scores are as follows:

Decision Step 0

Decision step 0

	Feature	Importance
1	purpose	0.23
2	application_type	0.11
3	annual_inc	0.10
4	initial_list_status	0.07
5	open_act_il	0.04
6	dti	0.04
7	mo_sin_old_rev_tl_op	0.03
8	funded_amnt	0.03
9	total_cu_tl	0.03
10	num_accts_ever_120_pd	0.03

Table 5.1: Decision step 0: Feature importance.

At the first decision step, the model primarily focuses on the purpose of the loan, application type, and annual income of the borrower. This aligns with an initial evaluation phase, where the purpose of the loan and the borrower's income are crucial to assess the feasibility and risk associated with the loan.

Decision Step 1

Decision step 1

	Feature	Importance
1	mort_acc	0.21
2	open_il_24m	0.15
3	num_actv_rev_tl	0.14
4	open_rv_24m	0.12
5	avg_cur_bal	0.11
6	home_ownership	0.08
7	grade	0.04
8	num_actv_bc_tl	0.03
9	policy_code	0.03
10	mths_since_recent_bc	0.02

Table 5.2: Decision step 1: Feature importance.

At this stage, the model seems to shift its focus towards the number of mortgage accounts, the number of installment accounts opened in the last 24 months, and the number of active revolving trades. This suggests that the model is evaluating the borrower's existing credit obligations, which is an important factor in determining the borrower's ability to manage additional debt.

Decision Step 2

Decision step 2

	Feature	Importance
1	sub_grade	0.66
2	term	0.20
3	il_util	0.08
4	num_tl_120dpd_2m	0.04
5	int_rate	0.02
6	percent_bc_gt_75	nil
7	all_util	nil
8	num_tl_30dpd	nil
9	open_rv_24m	nil
10	mo_sin_rcnt_rev_tl_op	nil

Table 5.3: Decision step 2: Feature importance.

The most significant feature at this decision step is the sub-grade, followed by the term and IL utilization. The sub-grade, which is a measure of credit risk, naturally becomes a focal point for the model. The term of the loan and the IL utilization further provide information about the duration and the proportion of installment accounts being used, respectively, contributing to the risk assessment.

Decision Step 3

Decision step 3

	Feature	Importance
1	avg_cur_bal	0.56
2	fico_range_high	0.33
3	int_rate	0.07
4	fico_range_low	0.03
5	num_rev_accts	nil
6	initial_list_status	nil
7	inq_last_6mths	nil
8	funded_amnt_inv	nil
9	num_actv_rev_tl	nil
10	all_util	nil

Table 5.4: Decision step 3: Feature importance.

At this decision step, the model places a high importance on the average current balance, FICO range (both high and low), and interest rate. This indicates that the model is focusing on the borrower's credit history, current financial status, and the cost of the loan, which are significant in evaluating the borrower's ability to repay the loan and the overall risk of the loan.

Decision Step 4

Decision step 4

	Feature	Importance
1	fico_range_low	0.57
2	pct_tl_nvr_dlq	0.40
3	delinq_2yrs	0.03
4	policy_code	nil
5	all_util	nil
6	total_bal_ex_mort	nil
7	fico_range_high	nil
8	tot_coll_amt	nil
9	open_rv_12m	nil
10	inq_last_6mths	nil

Table 5.5: Decision step 4: Feature importance.

At the final decision step, the model focuses heavily on the FICO range (low), the percentage of trades never delinquent, and the number of delinquencies in the last 2 years. This suggests that the model is making its final prediction based on the borrower's credit score, past payment behavior, and recent credit history, which are arguably among the most critical factors in assessing credit risk.

In summary, the TabNet model's decision-making process reflects a logical progression, similar to how a human expert might assess a loan application. It starts with a preliminary assessment based on the purpose of the loan and the borrower's income, followed by an evaluation of the borrower's existing credit obligations. Then, it shifts its focus to the borrower's credit risk and loan characteristics, and finally makes its prediction based on the borrower's credit history and past payment behavior. This stepwise progression of feature importance not only provides insights into the model's decision-making process but also underscores the complex, multi-faceted nature of credit risk assessment.

5.1.2. Mean SHAP values

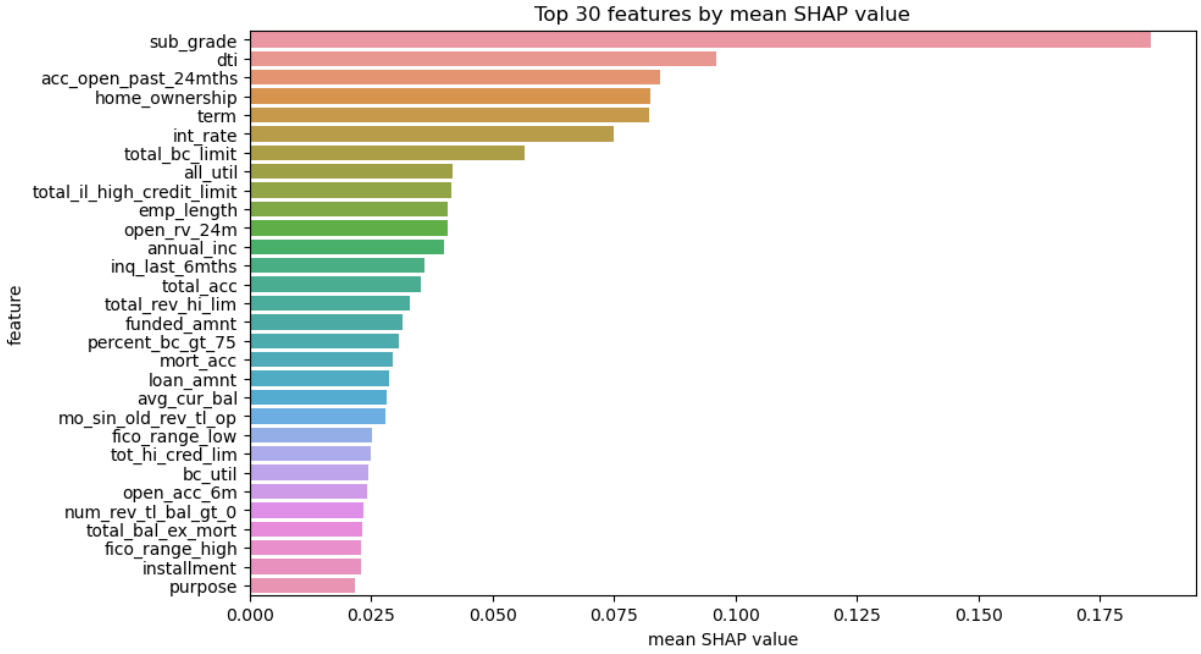

Figure 5.4: Mean SHAP values

SHAP values offer a game-theoretic approach to feature importance and are computed per instance, which means they consider every possible feature combination to evaluate the contribution of a specific feature.

The feature sub_grade exhibits the highest SHAP values, 0.187, indicating that it carries substantial influence in the decision-making process of the model. sub_grade is a measure of risk assigned by lenders to borrowers, and its strong influence reflects the validity of the risk grading system used by lenders. As higher grades indicate greater default risk, our model evidently assigns significant weight to this indicator in making its predictions.

The dti (Debt-to-Income) ratio, which measures a borrower's monthly debt payments relative to their gross monthly income, holds a mean SHAP value of 0.1. This substantial influence is expected as this ratio is a common tool used by lenders to assess the borrower's ability to manage monthly payments and repay debts. A higher dti implies a higher risk of default.

Features such as acc_open_past_24mths (the number of trades opened in the past 24 months), home_ownership (whether the borrower owns a home or not), term (length of the loan in months), and int_rate (interest rate on the loan) show slightly lower SHAP values, from 0.089 to 0.08. This suggests that while these factors do play a part in the

model's decision-making process, their influence is comparatively less than sub_grade and dti.

The SHAP values computation, though effective, is computationally intensive, particularly for models with a large number of features. This computational demand can pose difficulties in certain scenarios, especially in real-time systems or when dealing with vast feature spaces.

Furthermore, the SHAP values method assumes that the features are independent from one another, which is often not the case in real-world datasets. Correlations between features could potentially confound our interpretations of feature importance, leading to misleading conclusions about the impact of individual predictors.

SHAP values focus on the average contribution of a feature across all possible coalitions of features and, as such, they might not provide complete insight into the complexity of our model. They don't explicitly capture interactions between features which could play a significant role in prediction outcomes, particularly in cases of high feature interdependencies.

Also, being an aggregate measure, SHAP values can't provide insight into specific instances where a feature might be particularly important or unimportant. While they help us understand the average effect of a feature, they may miss out on the nuances of its effect across different contexts and subsets of data.

5.1.3. Methodologies comparison

When contrasting the TabNet mask-based method and the SHAP values method, both similarities and differences in their identification of significant features, such as sub_grade and term, become apparent. However, differences such as the high significance of the FICO Range in TabNet and the dti ratio in SHAP values highlight the distinct approaches of these methods.

TabNet masks not only offer an efficient way to discern feature importance, but they also consider the interdependencies and nonlinear relationships between features at each decision step. This ability to account for intricate relationships proves particularly useful in real-world scenarios, where features often interact in complex and nonlinear ways, something that's not adequately captured by the SHAP values method.

On the other hand, the SHAP values approach provides a robust measure of feature importance, grounded in game theory. However, it's essential to recognize its limitations: it is computationally intensive, particularly with a large number of features. It also

assumes features are independent from one another, which may not hold true in most real-world datasets, including ours, potentially leading to distortions in interpreting feature importance.

The computational demands of SHAP values can present substantial challenges in real-time systems or vast feature spaces. Furthermore, SHAP values, being an aggregate measure, may not fully expose the nuances of feature importance across different contexts and data subsets.

Therefore, when determining the choice of interpretability approach within the context of loan default prediction, one should consider the specific strengths and limitations of these techniques. In particular, the ability of TabNet masks to better capture feature interdependencies and non-linearities should be weighed against the theoretical robustness of SHAP values, considering the specific requirements and constraints of the task at hand.

5.2. Local Interpretability

Local interpretability of a machine learning model is an essential aspect of model evaluation, especially in high-stakes fields such as loan default prediction. While global interpretability provides an overall understanding of the model's functionality, local interpretability is focused on explaining why and how a model made a specific decision for a single instance.

In predictive models, local interpretability can help in identifying the main features that drive the model's prediction. This is crucial in domains like banking, where a model's decision can significantly impact a person's financial situation. Providing an explanation for the model's predictions not only improves transparency but also helps in identifying potential biases and errors in the model.

In our research, we employ the TabNet model for loan default prediction. TabNet, a high-performance and interpretable model, generates masks for each decision step, allowing us to visually inspect which features are most important at each step. We provide an analysis of local interpretability by randomly selecting three samples from our test set—two where TabNet predicts default and one where it predicts full repayment.

These local interpretations, coupled with the visual interpretation from the masks, provide insights into the model's decision-making process. However, the interpretability of TabNet does not stop at identifying important features. The model also shows how it combines these features at each decision step, providing an even more granular understanding of its predictions.

This local interpretability approach enhances our trust in the model, as we can dissect and understand its predictions for individual cases. More importantly, it provides actionable insights that can be used to improve the model and make fair and accurate predictions. For instance, handling missing data like the 'il_util' in the second sample could potentially improve the model's performance.

5.2.1. Borrower 1

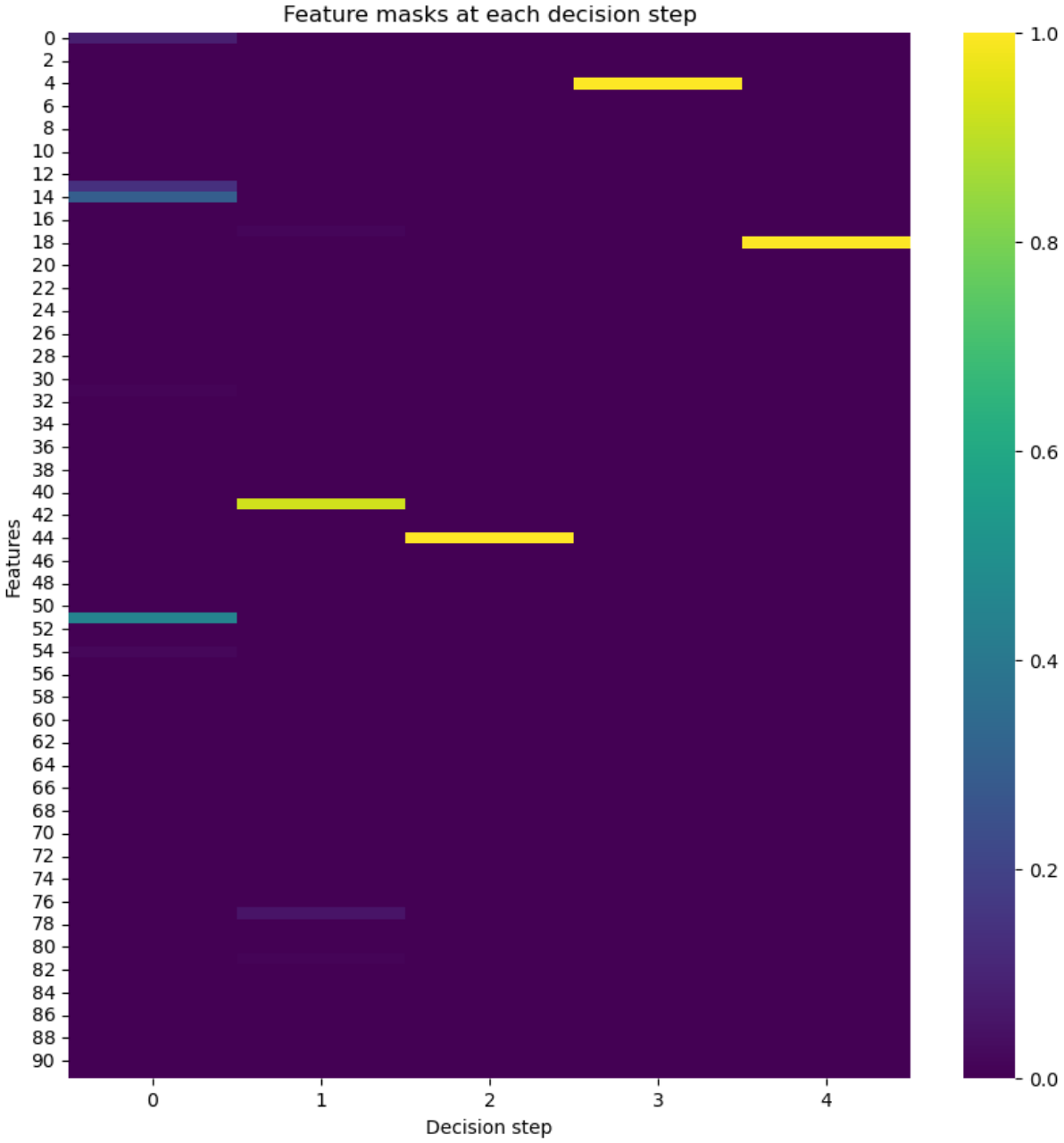

Figure 5.5: Borrower 1 masks at Each Decision Step

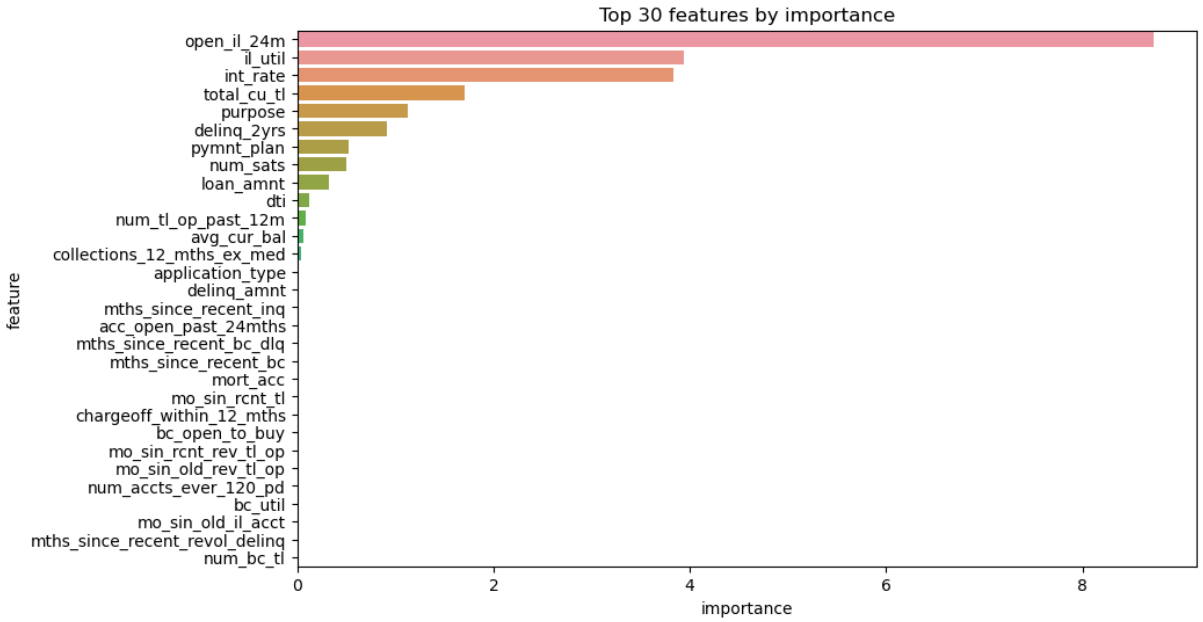

Figure 5.6: Borrower 1 Top 30 Features by Importance

Top 10 Feature Importances (Sample 1)

	Feature	Importance	Value
1	open_il_24m	8.724	2.0
2	il_util	3.943	43.0
3	int_rate	3.832	20.39
4	total_cu_tl	1.710	6.0
5	purpose	1.124	other
6	delinq_2yrs	0.916	1.0
7	pymnt_plan	0.526	n
8	num_sats	0.494	26.0
9	loan_amnt	0.321	24000.0
10	dti	0.116	26.91

Table 5.6: Top 10 Feature Importances and their values for sample 1.

In this case, we can see a picture of a borrower who might be under some financial stress. The high importance of open_il_24m and il_util suggests that this individual has been opening new credit lines recently and is using a significant portion of their available installment credit, both of which can be signs of financial strain. Coupled with

the high int_rate and dti (debt-to-income ratio), it seems this borrower is handling a considerable amount of debt relative to their income. The purpose of the loan, labeled as 'other', might not be geared towards improving their financial situation, such as debt consolidation, which could be another red flag. In essence, the model is picking up on a pattern of overleveraging, which could lead to default.

SHAP values

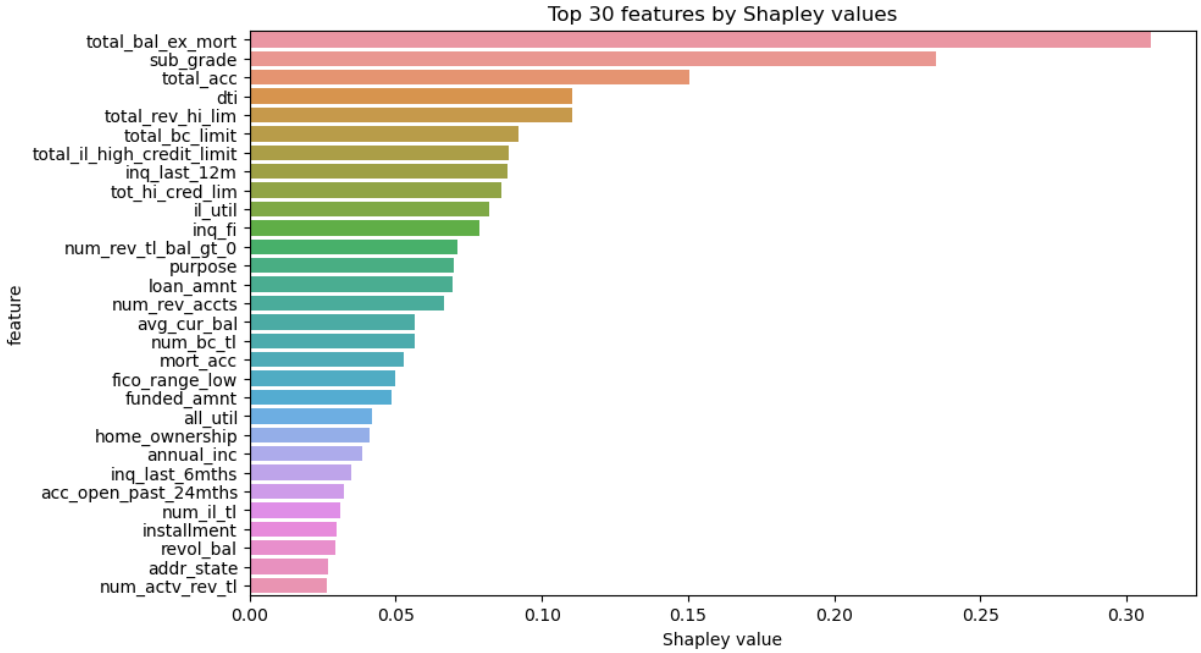

Figure 5.7: SHAP values Borrower 1

Top 10 Feature SHAP values (Sample 1)

	Feature	SHAP values	Value
1	total_bal_ex_mort	0.308	296613.0
2	sub_grade	0.235	D4
3	total_acc	0.151	74.0
4	dti	0.111	26.91
5	total_rev_hi_lim	0.110	103600.0
6	total_bc_limit	0.092	48500.0
7	total_il_high_credit_limit	0.088	244248.0
8	inq_last_12m	0.088	12.0
9	tot_hi_cred_lim	0.086	736987.0
10	il_util	0.082	43.0

Table 5.7: Top 10 Feature SHAP values and their values for sample 1.

For the rejected Borrower 1 (Table 5.7), the SHAP values highlighted that total_bal_ex_mort played a pivotal role in the loan default prediction. The high balance across all the borrower's accounts, excluding mortgage, indicated substantial financial obligations, possibly leading to an increased default risk and, consequently, loan rejection.

The credit rating sub_grade of D4 was also a significant influencer in the model's decision. This grade implies a higher risk borrower, strengthening the model's prediction towards loan rejection. Other notable features include the total number of credit lines (total_acc), the borrower's debt-to-income ratio (dti), and the total revolving high credit/credit limit (total_rev_hi_lim). These measures reflect the borrower's credit utilization and overall indebtedness.

5.2.2. Borrower 2

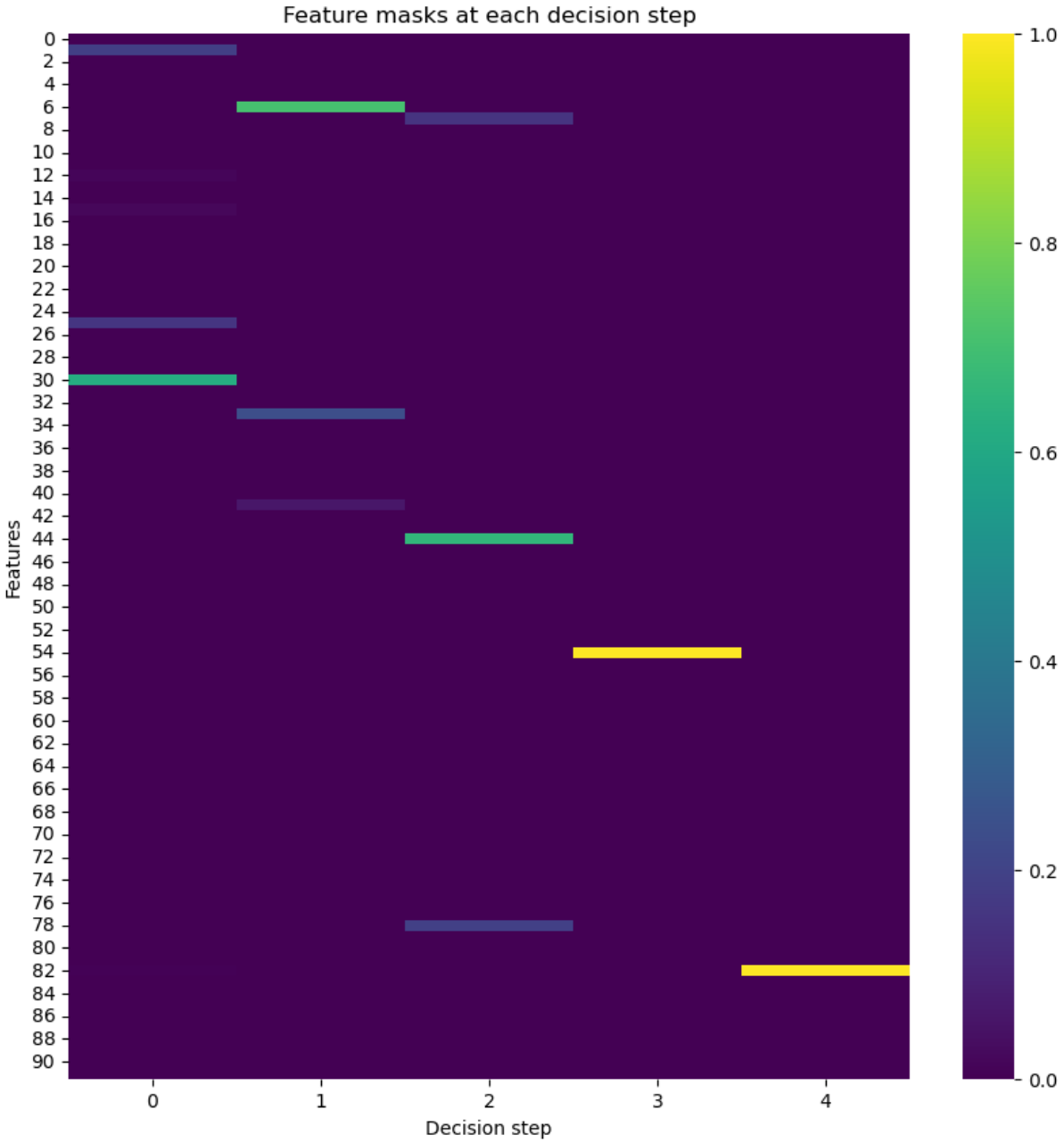

Figure 5.8: Borrower 2 masks at Each Decision Step

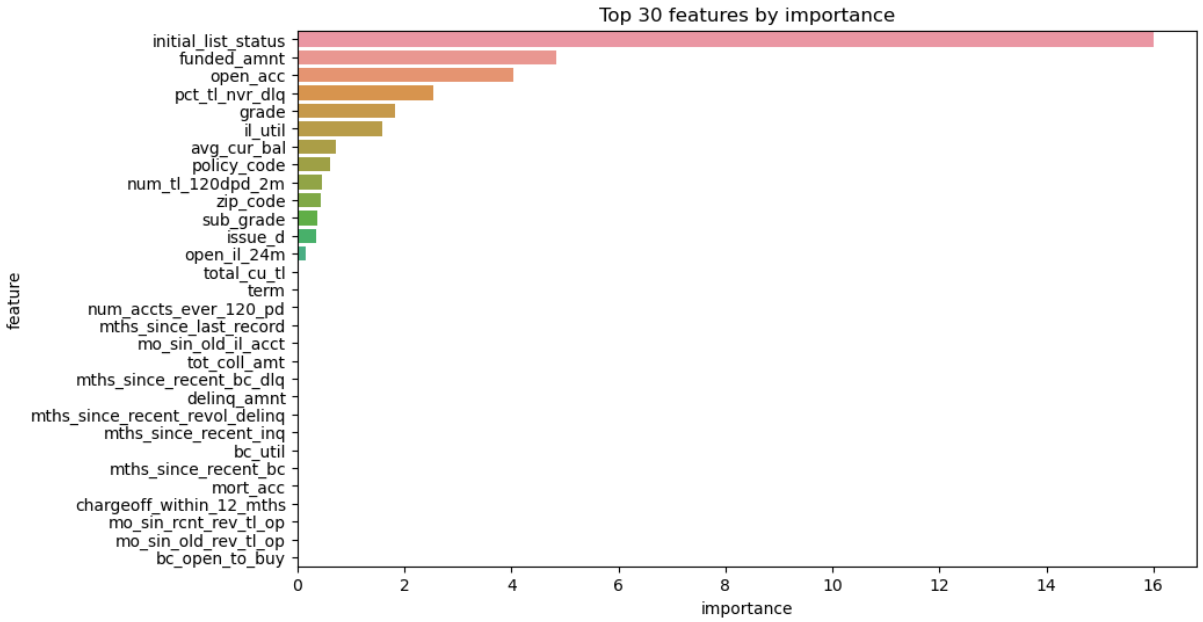

Figure 5.9: Borrower 2 Top 30 Features by Importance

Top 10 Feature Importances (Sample 2)

	Feature	Importance	Value
1	initial_list_status	16.002	w
2	funded_amnt	4.832	24000.0
3	open_acc	4.042	13.0
4	pct_tl_nvr_dlq	2.551	92.9
5	grade	1.818	E
6	il_util	1.590	NaN
7	avg_cur_bal	0.710	3888.0
8	policy_code	0.606	1.0
9	num_tl_120dpd_2m	0.456	0.0
10	zip_code	0.435	112xx

Table 5.8: Top 10 Feature Importances and their values for sample 2.

The second borrower shows a different story. They have a loan that was fully available to investors (initial_list_status), indicating high investor confidence at the outset. However, they have a large funded amount (funded_amnt) and a relatively low percentage of trades never delinquent (pct_tl_nvr_dlq), which implies a history of missed payments. The

borrower's loan is also classified as grade E, a lower grade suggesting higher risk. The lack of il_util data could mean that the borrower doesn't have any installment accounts or there's missing data - the uncertainty here could be a cause for concern. Overall, while this borrower might have seemed a good bet initially, their payment history and potentially risky financial behavior have led the model to predict a default.

SHAP values

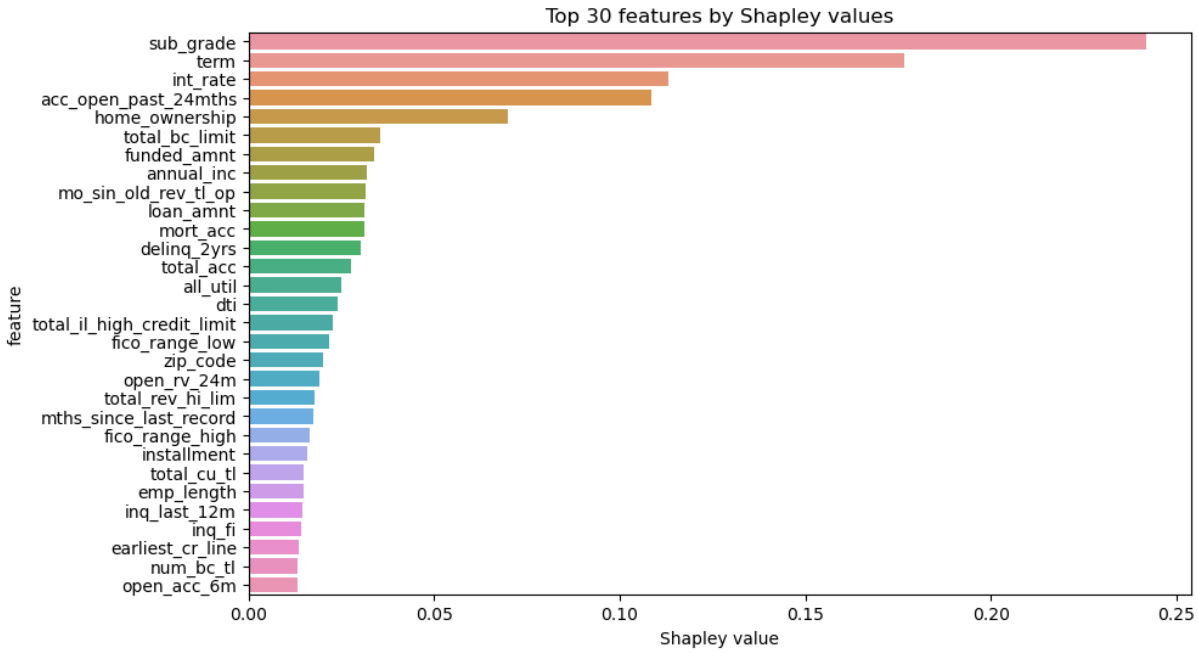

Figure 5.10: SHAP values Borrower 2

Top 10 Feature SHAP values (Sample 2)

	Feature	SHAP values	Value
1	sub_grade	0.242	E4
2	term	0.177	60 months
3	int_rate	0.113	19.99
4	acc_open_past_24mths	0.108	9.0
5	home_ownership	0.070	RENT
6	total_bc_limit	0.035	9100.0
7	funded_amnt	0.034	24000.0
8	annual_inc	0.032	90000.0
9	mo_sin_old_rev_tl_op	0.031	48.0
10	loan_amnt	0.031	24000.0

Table 5.9: Top 10 Feature SHAP values and their values for sample 2.

Borrower 2 (Table 5.9), which was also rejected, provided interesting insights through the SHAP values. The sub_grade of E4 stands out, indicating a borrower at a higher credit risk than Borrower 1. The lengthy loan term of 60 months, coupled with a higher interest rate (int_rate), escalated the overall risk, leading to the loan's rejection.

Other influential features, such as the number of credit lines opened in the past 24 months (acc_open_past_24mths), home ownership status, and total bankcard limit (total_bc_limit), also played a part in the model's decision.

5.2.3. Borrower 3

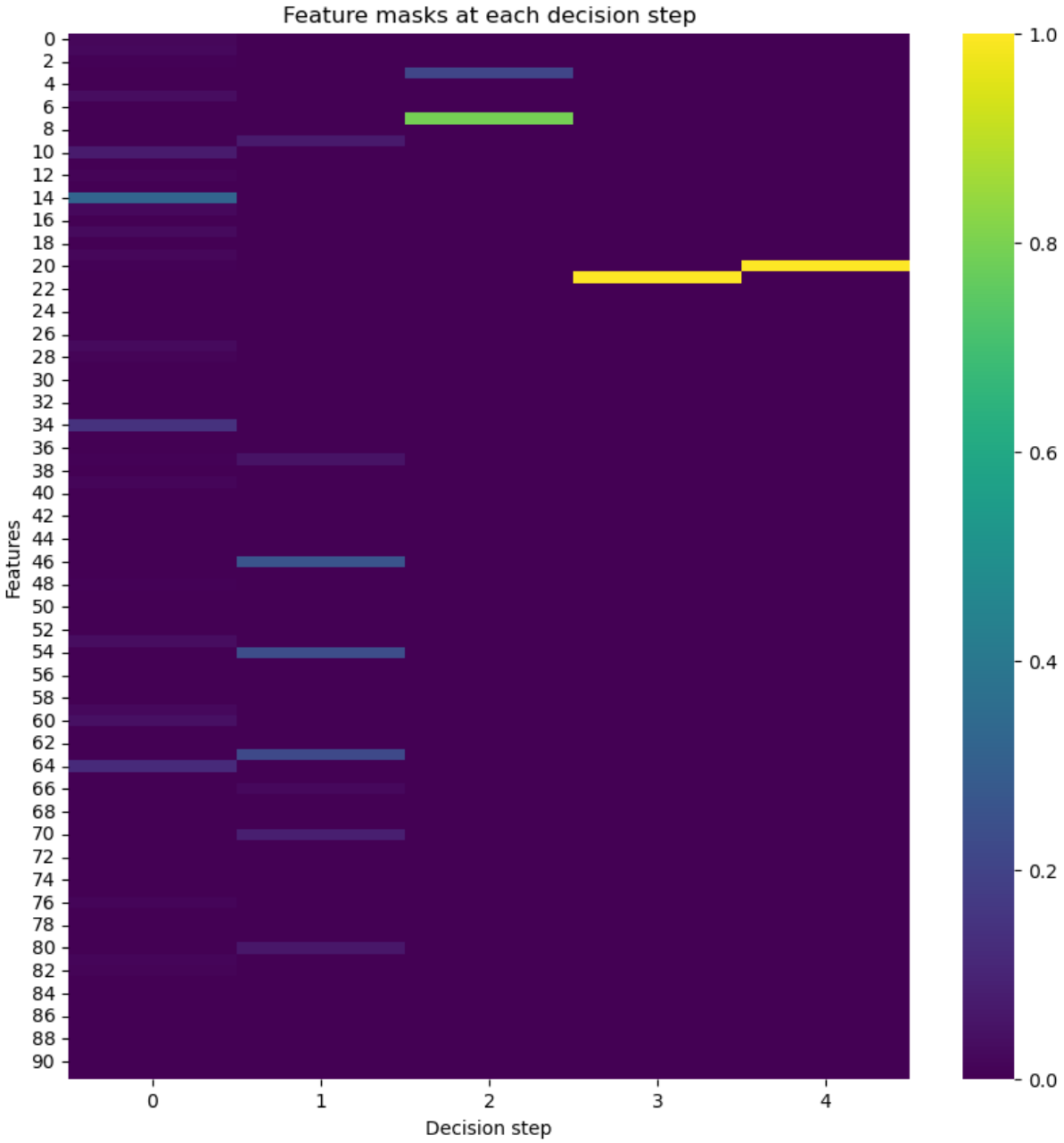

Figure 5.11: Borrower 3 masks at Each Decision Step

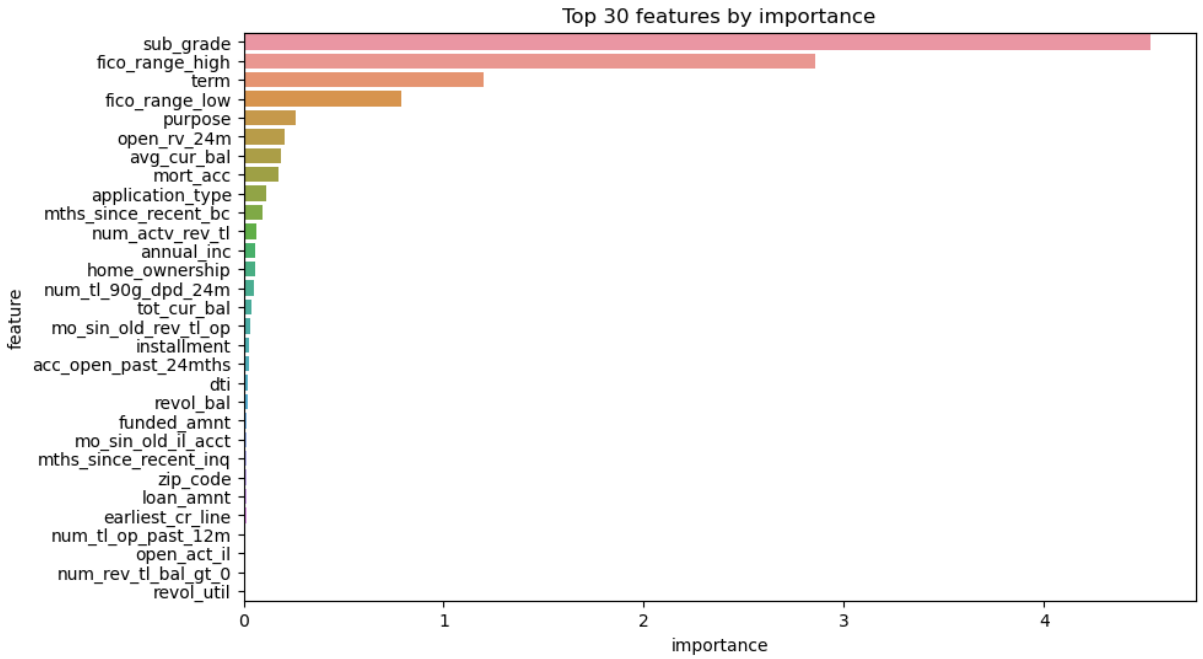

Figure 5.12: Borrower 3 Top 30 Features by Importance

Top 10 Feature Importances (Sample 3)

	Feature	Importance	Value
1	sub_grade	4.529	C5
2	fico_range_high	2.855	674.0
3	term	1.199	60 months
4	fico_range_low	0.789	670.0
5	purpose	0.260	debt_consolidation
6	open_rv_24m	0.207	NaN
7	avg_cur_bal	0.190	27360.0
8	mort_acc	0.177	2.0
9	application_type	0.115	Individual
10	mths_since_recent_bc	0.095	120.0

Table 5.10: Top 10 Feature Importances and their values for sample 3.

For the third borrower, we see a more stable financial profile. This person has a solid FICO score (fico_range_high and fico_range_low), indicating good creditworthiness. The loan purpose, debt_consolidation, suggests a proactive approach to managing their finances. Their sub-grade C5 shows that they are in the middle risk group, yet the model

predicts that they will fully repay the loan. It's possible that their good credit score, combined with a responsible approach to debt management, outweighs the potential risk indicated by their sub-grade. The 60-month term might also suggest that this borrower is confident in their ability to repay the loan over a longer period. The model is picking up on these signs of financial stability and responsibility, which leads to the prediction of full repayment.

SHAP values

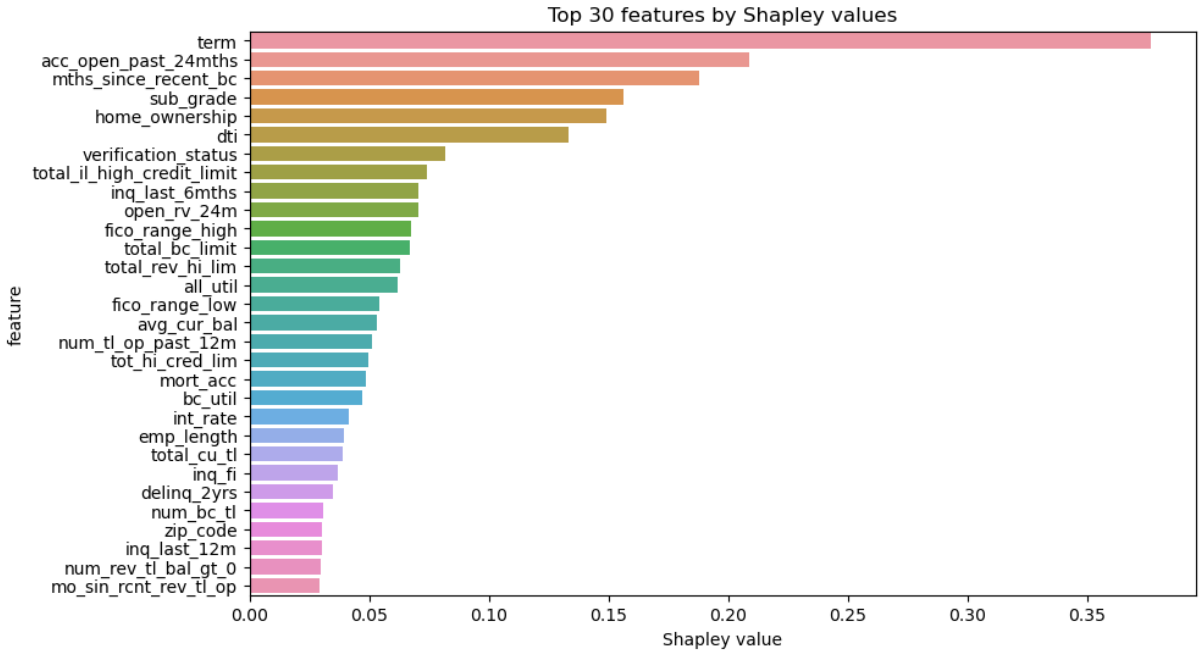

Figure 5.13: SHAP values Borrower 3

Top 10 Feature SHAP values (Sample 3)

	Feature	SHAP values	Value
1	term	0.377	60 months
2	acc_open_past_24mths	0.209	2.0
3	mths_since_recent_bc	0.188	120.0
4	sub_grade	0.156	C5
5	home_ownership	0.149	MORTGAGE
6	dti	0.133	24.32
7	verification_status	0.082	Not Verified
8	total_il_high_credit_limit	0.074	52881.0
9	inq_last_6mths	0.071	0.0
10	open_rv_24m	0.070	NaN

Table 5.11: Top 10 Feature SHAP values and their values for sample 3.

In the case of the approved Borrower 3 (Table 5.11), despite the loan term of 60 months, which usually signals a higher risk, the model perceived it as acceptable, possibly due to counterbalancing factors. The SHAP values suggested that the low number of credit lines opened in the past 24 months (acc_open_past_24mths) and the better credit sub-grade of C5 lowered the perceived default risk.

Although a high mths_since_recent_bc value and a MORTGAGE status in home ownership could typically escalate the risk profile, these were offset by factors such as a relatively low debt-to-income ratio (dti) and a Not Verified verification status. Despite the verification status, the loan was approved, indicating the borrower's strong profile as inferred by the model.

5.2.4. Methodologies comparison

The interpretation of model predictions in machine learning is of utmost importance, especially in sectors where the consequences of decisions carry far-reaching implications. SHAP values and TabNet Masks are two methodologies utilized for understanding feature importance within predictive models, each offering unique insights based on their underlying mechanisms.

SHAP values, a methodology hailing from cooperative game theory, ensures a fair allocation of feature importance. This is achieved by considering all potential permutations

of features, which guarantees a comprehensive perspective on the contribution of each feature. Such a wide-ranging approach is applicable across all models, offering consistent and equitable interpretations.

However, despite these strengths, the SHAP values methodology may not always yield the most granular insights. While it provides a thorough overview of feature importance, the method assumes that the features are independent, which is not often the case with real-world datasets. Correlations between features could potentially confuse our interpretations of feature importance, potentially leading to false conclusions about individual predictors' impacts.

Moreover, the SHAP values method can be computationally intensive, especially when working with models with large feature spaces. This can pose challenges in scenarios such as real-time systems, where computational speed is critical.

In addition, SHAP values focus on the average contribution of a feature across all potential coalitions of features. Consequently, they may not fully reveal the complexity of a model's decision-making process as they do not explicitly account for feature interactions, which could significantly influence prediction outcomes, particularly in cases of high feature interdependencies.

On the other hand, the TabNet Masks methodology provides a more detailed look into the decision-making process. This technique assigns feature importance at each decision step within the TabNet model, offering insights into how features interact and influence the prediction at various stages of the process. This granular perspective allows for a richer understanding of the complex interplay of features, which is beneficial in intricate domains like credit risk, where comprehending the sequence and interaction of feature importance can provide crucial insights.

For example, in the case of our three borrowers, TabNet Masks highlighted distinct sets of features for each borrower as important. This reflects the complexity and individuality of their financial profiles and underscores how the model's decisions are influenced by diverse factors.

The strength of TabNet Masks lies in its ability to elucidate these complex feature interactions and their sequential importance. This rich, detailed analysis contrasts with the simpler, more aggregated insights provided by SHAP values and makes TabNet Masks a powerful tool for interpreting complex prediction tasks.

In summary, while both SHAP values and TabNet Masks have their utility in model interpretation, the ability of TabNet Masks to provide more nuanced insights into the

complex interactions and decision-making processes makes it an invaluable tool in intricate prediction tasks. By providing diverse and detailed insights, it offers a robust foundation for understanding and interpreting model predictions.

6 | LightGBM Interpretability Analysis

The LightGBM algorithm offers an in-built mechanism to calculate feature importances, thereby providing valuable insights into the features that are most significant in determining the predictions of the model.

LightGBM computes feature importances in two ways: split-based and gain-based. The split-based method counts the number of times a feature is used to split the data, while the gain-based method adds up the total gain brought about by each feature when it is used for a split. A higher gain indicates that the feature is more useful for the model in making accurate predictions.

We focus on the gain-based feature importances. For each feature, the importance score computed by the LightGBM model represents the total gains of splits that this feature contributes to, across all trees in the model. The gains are calculated as the reduction in the loss function brought about by the splits on that feature. The loss function quantifies the difference between the predicted and the actual values. Therefore, a greater total gain implies that the splits on that feature significantly improve the accuracy of the predictions.

6.1. SHAP values

The SHAP value of a feature represents the average contribution of that feature to the prediction of the model across all possible combinations of features. Higher SHAP values correspond to features that, on average, contribute more to the prediction of the model.

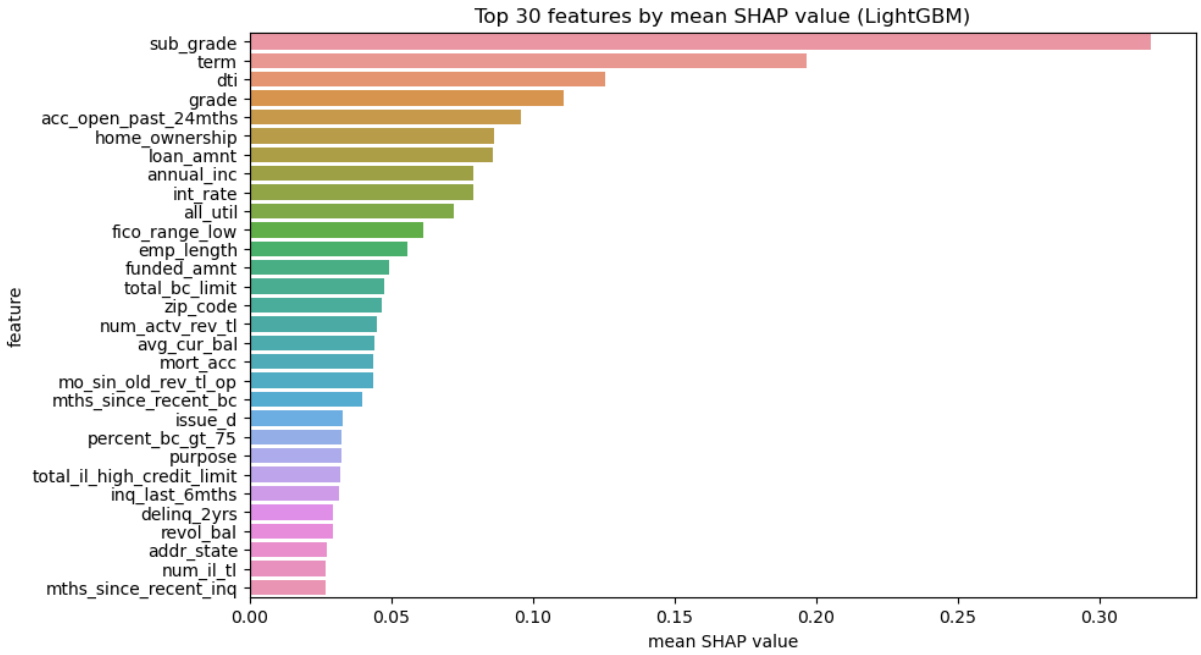

Figure 6.1: Mean SHAP values

Analyzing the SHAP values for our LightGBM model, we see that the 'sub_grade' feature has the highest mean SHAP values, followed by 'term' and 'dti'. This indicates that these features generally contribute the most to the model's decision-making process across all instances. 'sub_grade' and 'term' may likely be capturing information about the credit-worthiness of the loan applicant, while 'dti', or debt-to-income ratio, can be a significant determinant of an applicant's ability to service a loan.

6.2. Feature Importances

Let's examine the top five features according to the LightGBM model's feature importances, see table 6.1: sub_grade, grade, term, dti, and acc_open_past_24mths.

From the SHAP values values, sub_grade and grade have the highest mean SHAP values values, which corroborates the feature importance ranking of the LightGBM model. These variables reflect the risk level of the loan as determined by the lending institution, which is a critical determinant in loan default prediction.

The term of the loan, reflecting the duration in which the borrower is expected to repay the loan, also appears significant in both rankings. It stands to reason that a longer term might correspond to a higher risk of default due to more uncertainties over a more extended period.

Feature	Importance
sub_grade	228106.075174
grade	133784.966260
term	30774.391809
dti	20153.796853
acc_open_past_24mths	15419.500558
annual_inc	13567.750742
avg_cur_bal	13035.517462
zip_code	12675.682653
all_util	12505.249052
emp_length	12460.728152
int_rate	12169.775367
fico_range_low	12041.180121
loan_amnt	10719.932024
home_ownership	10426.620638
issue_d	9160.488230

Table 6.1: Top 15 most predictive features

The dti (debt-to-income ratio) ranks fourth in importance for both methodologies. This metric is a well-known risk factor in credit scoring, as it represents the proportion of a borrower's income that goes towards debt repayments. A higher dti often indicates a borrower's potential struggle to manage their debt load, hence increasing the risk of default.

The feature acc_open_past_24mths reflects the number of credit lines opened by the borrower in the last 24 months. Both the feature importances and the SHAP values values suggest that this feature plays a critical role in the model's predictions. This could be because opening multiple credit lines in a short period may indicate financial stress and could potentially increase the risk of default.

It is noteworthy that the rankings from feature importances and SHAP values values mostly align, reinforcing the significance of these features in predicting loan defaults. However, there are slight differences, for instance, annual_inc ranks relatively higher in feature importance compared to SHAP values values. This might be due to different perspectives of these methodologies, with feature importance focusing on the overall gain each feature brings to the model's performance and SHAP values values explaining the contribution of each feature to individual predictions.

6.3. SHAP values and Feature Importances Comparison

The SHAP values and feature importances convey two different perspectives on feature importance. SHAP values provide a model-agnostic, fair, and equitable distribution of importance based on average contribution across all possible combinations of features. They consider every possible configuration of features and calculate the marginal contribution of each feature to the prediction.

On the other hand, feature importances are a measure of how much a feature contributes to the improvement of the model's accuracy, based on how frequently the feature is used in constructing decision trees and the associated gains in model accuracy from those splits.

It's notable that while the ranking of features differs slightly between the two methods, the top features identified by both methods are largely consistent. Both methods identify 'sub_grade', 'term', and 'dti' as key features. However, the relative importance of these features varies between the two methods. For example, 'dti' ranks third in terms of SHAP values but only fourth in terms of feature importance. This could be because 'dti' interacts with other features in complex ways that may not always lead to high gains in model accuracy, but its marginal contribution when considered with all possible feature combinations is still significant.

It's also worth noting that some features rank significantly higher in feature importance than in SHAP values (e.g., 'grade'). This could be because these features contribute significantly to model accuracy when they are used in tree splits, even though their average marginal contribution across all feature combinations (as measured by SHAP values) may be less.

In conclusion, SHAP values and feature importances provide two complementary perspectives on feature importance. SHAP values highlight the average marginal contribution of a feature across all possible feature combinations, whereas feature importances highlight the contribution of a feature to model accuracy in terms of its usage in tree splits and the associated gains. Both methods can be useful in different scenarios and provide a more complete understanding of the model's decision-making process when used together.

6.4. Comparison between LightGBM and TabNet Interpretability

LightGBM and TabNet both possess interpretability features, however, TabNet distinctly surpasses LightGBM in this aspect, providing richer insights and deeper understanding of the decision-making process. LightGBM depends on SHAP values values and feature importances for interpretation. These tools offer a general overview of the features' cumulative influence on the model's prediction performance.

While examining the top ten features ranked by SHAP values, an interesting uniformity was observed across both LightGBM and TabNet models. This consistency can be interpreted as a testament to the strength of SHAP values in pinpointing the most stable and independent predictive features across various models. Although SHAP values might not delve deep into the intricate details of individual models, they provide overarching insights into the dataset itself.

Prominent and independent features tend to obtain higher SHAP values, which, while potentially overshadowing the more subtle variables, present an expansive view of the data's structure and impact.

This situation brings to light an intriguing comparison with LightGBM's feature importances, which bear similarities to SHAP values in their ability to provide a broad understanding of the data, rather than a granular interpretation. Herein lies the distinct strength of TabNet's design. With its unique interpretability approach based on attention masks, TabNet provides a more detailed and precise interpretability. Its feature importance is determined by the relevance of each feature in each decision, creating a more granular, case-specific view.

While LightGBM and SHAP values provide a wider lens into feature importance, TabNet's design caters to the demand for a detailed understanding of individual predictions. Therefore, while SHAP values hold their merit in identifying robust and consistent predictors across models, TabNet offers a compelling advantage in detailed interpretability.

TabNet, due to its unique architecture, offers interpretability inherently built into its design. TabNet Masks, a feature unique to TabNet, allows for a nuanced understanding of the sequential importance of features and their interdependencies. This capability is invaluable for intricate predictive tasks where understanding the synergistic effects of multiple features on predictions is vital. Furthermore, the temporal significance of features, where certain features gain relevance at different decision stages, is illuminated

by TabNet's design.

Conclusions and Future Developments

The focus of this thesis has been the application of the TabNet model for loan default prediction, with an emphasis on its interpretability. We explored its performance against other prevalent models and demonstrated its competitive performance, notwithstanding its explicit emphasis on interpretability - an attribute often compromised in high-performing machine learning models.

We found that the TabNet model outperformed other models in terms of Validation Accuracy, Test Accuracy, and Test Precision, with strong performance in other measures as well. Despite not being the absolute best in every metric, TabNet offers a robust compromise between performance and interpretability. Notably, its superior precision indicates its strength in reducing false positives, a crucial attribute in financial applications where wrongly classifying a loan as default could lead to significant economic consequences.

Our comprehensive analysis of TabNet masks has made a significant contribution to understanding the process of loan default prediction. By attributing feature importance at each decision step within the model, TabNet masks offer a more nuanced understanding of how features interact and influence the prediction process. This approach contrasts with traditional methods such as SHAP values, providing a more detailed perspective on feature interaction and decision-making processes.

We observed that TabNet's emphasis on interpretability didn't hamper its performance, but rather facilitated it. We believe this combination of interpretability and high performance is vital for high-stakes applications like loan default prediction where understanding the reasoning behind the predictions is as important as the accuracy of those predictions.

Looking towards the future, there is a clear potential for continued advancements in this field. We recommend further research on the application of TabNet and other interpretable models in related financial tasks, such as credit card default prediction or mortgage default prediction. These similar yet distinct tasks could provide new challenges and opportunities for these models.

In addition, future work could explore the applicability of the TabNet model across different geographies and economies. As our study focused primarily on a specific context, a broader scope could reveal variations in the performance of the TabNet model and expand its potential uses.

Furthermore, we believe that the development of more advanced and interpretable models like TabNet has the potential to revolutionize the financial industry by offering a clearer understanding of the credit risks involved in lending. This has the potential to reduce the incidence of financial crises and to contribute to a more stable financial system.

Lastly, we suggest future research into more computationally efficient methods for achieving high interpretability, as this continues to be a challenge in the field. Further work could also delve into better handling imbalanced datasets, a frequent issue in default risk prediction.

In conclusion, this thesis has successfully demonstrated the power and potential of the TabNet model as an interpretable yet high-performing machine learning model. Our work has broadened the understanding of TabNet's application in loan default prediction and opened up new avenues for future research in advancing interpretable machine learning for financial applications. We believe that the balance between performance and interpretability will continue to be a focal point in machine learning research, and that our findings will contribute to ongoing efforts in this direction.

Bibliography

[1] E. Alfaro, N. García, M. Gámez, and D. Elizondo. Bankruptcy forecasting: An empirical comparison of adaboost and neural networks. *Decision Support Systems*, 45(1):110–122, 2008.

[2] S. O. Arik and T. Pfister. Tabnet: Attentive interpretable tabular learning. *arXiv preprint arXiv:1908.07442*, 2019.

[3] T. Bellotti and J. Crook. Support vector machines for credit scoring and discovery of significant features. *Expert Systems with Applications*, 36(2):3302–3308, 2009.

[4] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[5] G. Cascarino, M. Moscatelli, and F. Parlapiano. Explainable artificial intelligence: interpreting default forecasting models based on machine learning. 2022. URL `https://www.dirittobancario.it/wp-content/uploads/2022/03/Explainable-Artificial-Intelligence_-interpreting-default.pdf`.

[6] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[7] Y.-R. Chen, J.-S. Leu, S.-A. Huang, J.-T. Wang, and J.-I. Takada. Predicting default risk on peer-to-peer lending imbalanced datasets. 2023. URL `https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9429248`.

[8] A. Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, Inc., 2019.

[9] D. J. Hand and W. E. Henley. Statistical classification methods in consumer credit scoring: A review. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 160(3):523–541, 1997.

[10] I. T. Jolliffe and J. Cadima. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202, 2016.

[11] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, 30:3146–3154, 2017.

[12] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[13] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30:4765–4774, 2017.

[14] M. Malekipirbazari and V. Aksakalli. Risk assessment in social lending via random forests. In *The European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, page 963–968, 2015.

[15] C. Molnar, G. König, J. Herbinger, T. Freiesleben, S. Dandl, C. A. Scholbeck, G. Casalicchio, M. Grosse-Wentrup, and B. Bischl. General pitfalls of model-agnostic interpretation methods for machine learning models. 2020.

[16] L. C. Thomas. A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers. *International Journal of Forecasting*, 16(2):149–172, 2000.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30:5998–6008, 2017.

[18] Y. R. Wei Bao, Jun Yue. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PloS one*, 12(7):e0180944, 2017.

[19] L. Zhu, D. Qiu, D. Ergu, C. Ying, and K. Liu. A study on predicting loan default based on the random forest algorithm. In *International Conference on Information Technology and Quantitative Management*, 2019.

A | Appendix A

The Python code utilized throughout the course of this Master's thesis, which amounts to several thousands of lines, has been organized and is accessible via a GitHub repository. This digital method of storage ensures that the code is neatly presented and easily navigable for readers. The repository encompasses all the Python scripts employed in this research.

To access the code, please follow the link provided:
`https://github.com/ArnaldoMll/Master-thesis-Tabnet`

For any inquiries, clarifications or issues related to the code, feel free to raise an issue in the repository or contact me directly.

B | Appendix B

This appendix presents a list in alphabetical order of all the features included in the dataset, after preprocessing, each accompanied by a brief explanation. Understanding these features is crucial for any subsequent data analysis, interpretation of results, and development of predictive models. The descriptions provided are concise, aiming to offer a preliminary understanding. For a more comprehensive definition of each feature, it is recommended to refer to the official Lending Club's data dictionary or accompanying documentation.

- **acc_now_delinq:** The number of accounts on which the borrower is now delinquent.
- **acc_open_past_24mths:** Number of trades opened in past 24 months.
- **all_util:** Balance to credit limit on all trades.
- **annual_inc:** The self-reported annual income provided by the borrower during registration.
- **application_type:** Indicates whether the loan is an individual application or a joint application with two co-borrowers.
- **avg_cur_bal:** Average current balance of all accounts.
- **bc_open_to_buy:** Total open to buy on revolving bankcards.
- **bc_util:** Ratio of total current balance to high credit/credit limit for all bankcard accounts.
- **chargeoff_within_12_mths:** Number of charge-offs within 12 months.
- **collections_12_mths_ex_med:** Number of collections in 12 months excluding medical collections.
- **delinq_2yrs:** The number of 30+ days past-due incidences of delinquency in the borrower's credit file for the past 2 years.

- **delinq_amnt:** The past-due amount owed for the accounts on which the borrower is now delinquent.
- **disbursement_method:** The method by which the borrower receives their loan.
- **dti:** A ratio calculated using the borrower's total monthly debt payments on the total debt obligations, excluding mortgage and the requested Lending Club loan, divided by the borrower's self-reported monthly income.
- **earliest_cr_line:** The month the borrower's earliest reported credit line was opened.
- **emp_length:** Employment length in years.
- **fico_range_high, fico_range_low:** The range of the borrower's FICO score.
- **funded_amnt:** The total amount committed to that loan at that point in time.
- **funded_amnt_inv:** The total amount committed by investors for that loan at that point in time.
- **grade:** Lending Club's assigned loan grade.
- **hardship_flag:** Flags whether the borrower is on a hardship plan.
- **home_ownership:** The home ownership status provided by the borrower during registration.
- **il_util:** Ratio of total current balance to high credit/credit limit on all install acct.
- **initial_list_status:** The initial listing status of the loan.
- **inq_fi:** Number of personal finance inquiries.
- **inq_last_12m:** Number of credit inquiries in past 12 months.
- **inq_last_6mths:** The number of inquiries by creditors in the last 6 months.
- **installment:** The monthly payment owed by the borrower if the loan originates.
- **int_rate:** Interest Rate on the loan.
- **issue_d:** The month which the loan was funded.
- **loan_amnt:** The listed amount of the loan applied for by the borrower.
- **loan_status:** Current status of the loan.
- **max_bal_bc:** Maximum current balance owed on all revolving accounts.

- **mo_sin_old_il_acct:** Months since oldest bank installment account opened.
- **mo_sin_old_rev_tl_op:** Months since oldest revolving account opened.
- **mo_sin_rcnt_rev_tl_op:** Months since most recent revolving account opened.
- **mo_sin_rcnt_tl:** Months since most recent account opened.
- **mort_acc:** Number of mortgage accounts.
- **mths_since_last_delinq:** The number of months since the borrower's last delinquency.
- **mths_since_last_major_derog:** Months since most recent 90-day or worse rating.
- **mths_since_last_record:** The number of months since the last public record.
- **mths_since_rcnt_il:** Months since most recent installment accounts opened.
- **mths_since_recent_bc:** Months since most recent bankcard account opened.
- **mths_since_recent_bc_dlq:** Months since most recent bankcard delinquency.
- **mths_since_recent_inq:** Months since most recent inquiry.
- **mths_since_recent_revol_delinq:** Months since most recent revolving delinquency.
- **num_accts_ever_120_pd:** Number of accounts ever 120 or more days past due.
- **num_actv_bc_tl:** Number of currently active bankcard accounts.
- **num_actv_rev_tl:** Number of currently active revolving trades.
- **num_bc_sats:** Number of satisfactory bankcard accounts.
- **num_bc_tl:** Number of bankcard accounts.
- **num_il_tl:** Number of installment accounts.
- **num_op_rev_tl:** Number of open revolving accounts.
- **num_rev_accts:** Number of revolving accounts.
- **num_rev_tl_bal_gt_0:** Number of revolving trades with balance >0.
- **num_sats:** Number of satisfactory accounts.

- **num_tl_120dpd_2m:** Number of accounts currently 120 days past due (updated in past 2 months).
- **num_tl_30dpd:** Number of accounts currently 30 days past due (updated in past 2 months).
- **num_tl_90g_dpd_24m:** Number of accounts 90 or more days past due in last 24 months.
- **num_tl_op_past_12m:** Number of accounts opened in past 12 months.
- **open_acc:** The number of open credit lines in the borrower's credit file.
- **open_acc_6m:** Number of open trades in last 6 months.
- **open_act_il:** Number of currently active installment trades.
- **open_il_12m:** Number of installment accounts opened in past 12 months.
- **open_il_24m:** Number of installment accounts opened in past 24 months.
- **open_rv_12m:** Number of revolving trades opened in past 12 months.
- **open_rv_24m:** Number of revolving trades opened in past 24 months.
- **pct_tl_nvr_dlq:** Percent of trades never delinquent.
- **percent_bc_gt_75:** Percentage of all bankcard accounts $> 75\%$ of limit.
- **policy_code:** Publicly available policy_code=1, new products not publicly available policy_code=2.
- **pub_rec:** Number of derogatory public records.
- **pub_rec_bankruptcies:** Number of public record bankruptcies.
- **purpose:** A category provided by the borrower for the loan request.
- **pymnt_plan:** Indicates if a payment plan has been put in place for the loan.
- **revol_bal:** Total credit revolving balance.
- **revol_util:** Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
- **sub_grade:** Lending Club assigned loan subgrade.
- **tax_liens:** Number of tax liens.

- **term:** The number of payments on the loan. Values are in months and can be either 36 or 60.
- **tot_coll_amt:** Total collection amounts ever owed.
- **tot_cur_bal:** Total current balance of all accounts.
- **tot_hi_cred_lim:** Total high credit/credit limit.
- **total_acc:** The total number of credit lines currently in the borrower's credit file.
- **total_bal_ex_mort:** Total credit balance excluding mortgage.
- **total_bal_il:** Total current balance of all installment accounts.
- **total_bc_limit:** Total bankcard high credit/credit limit.
- **total_cu_tl:** Number of finance trades.
- **total_il_high_credit_limit:** Total installment high credit/credit limit.
- **total_rev_hi_lim:** Total revolving high credit/credit limit.
- **verification_status:** Indicates if income was verified by Lending Club, not verified, or if the income source was verified.
- **zip_code:** The first 3 numbers of the zip code provided by the borrower in the loan application.

List of Figures

List of Tables