Paris 1 Panthéon-Sorbonne University

École d'Économie de la Sorbonne (EES)

Applied Machine Learning

Final Report

by: Fatima-Zahra BRICHA & Loan de TRAVERSAY

May 30, 2023

Contents

List of Tables

List of Figures

EXPLAINABLE AI FOR CREDIT RISK MANAGEMENT

Fatima-Zahra Bricha*[1] and Loan de Traversay†[2]

[1,2,3]**Master 2 Finance Technology Data , Paris 1 Panthéon-Sorbonne University**

May 30, 2023

Abstract

This paper analyzes the application of Explainable Artificial Intelligence (XAI) in the context of credit scoring for credit risk management. The main objective of this work is to evaluate the feasibility and potential benefits of employing an XAI approach into credit risk assessment. To achieve this goal, we evaluated three different scoring models, namely Random Forest, XGBoost, and a Neural Network. Additionally, we applied multiple tools to gain insights into the interpretability of these models and understand their results. The findings of our proof of concept (POC) provide valuable understanding of the advantages and limitations of XAI techniques in credit scoring, enabling informed decision-making for companies operating in the credit risk management domain.

1 Introduction

Credit risk management plays a vital role in the financial industry by enabling lenders to assess the creditworthiness of borrowers and make informed decisions regarding loan approvals. The digitization of financial services has significantly transformed the operations of institutions, emphasizing efficiency and customer convenience as key priorities. As technology continues to advance rapidly and new tools become available, complex machine learning techniques have emerged as powerful tools for analyzing vast amounts of data and extracting valuable insights. However, it is important to recognize that these techniques can introduce challenges such as a lack of transparency and interpretability, making it difficult to understand the factors driving credit decisions. Moreover, it is crucial to evaluate the appropriateness of these techniques for the specific use case. The adoption of any credit scoring model should be approached with caution, ensuring that the chosen techniques are justified by their proven effectiveness and are aligned with the specific requirements of credit risk assessment.

*Email: bricha.fz@gmail.com

†Email: Loan.de-Traversay@etu.univ-paris1.fr

One of the key requirements in financial services is interpretability and explainability. Financial institutions need to understand the factors influencing decisions and be able to clearly communicate it, particularly when it comes to credit risk assessment, in simple terms any lender should be able to explain the variables that drove their decision to deem a borrower as likely or unlikely to default on a loan. Therefore ensuring transparency, fairness, accountability, and compliance with regulatory guidelines.

However, machine learning techniques operate in an opaque manner providing little to no explanation or understanding of how they arrive at their decisions. This is why XAI (eXplainable Artificial Intelligence) has gained prominence as it offers transparency and interpretability in machine learning models. XAI techniques provide insights into the decision-making process by bridging the gap between complex algorithms and human understanding.

In the following report, we compare and analyze the application of XAI techniques to three different scoring models, namely Random Forest, XGBoost and Neural Networks. The rest of the paper is organized as follows: Section 2 introduces our methodology. Section 3 shows the results of the scoring models. Section 4 analyzes the explainability of the models Section 5 concludes and presents possible future developments.

2 Data

Our dataset, called Credit Risk Dataset was extracted from kaggle. It originally contains 32 581 of historical lending and default records and contains 12 variables, Table (1) shows the description of each feature.

Table 1: Description of Features

Feature Name	**Description**
person_age	Age
person_income	Annual Income
person_home_ownership	Home ownership
person_emp_length	Employment length (in years)
loan_intent	Loan intent
loan_grade	Loan grade
loan_amnt	Loan amount
loan_int_rate	Interest rate
loan_status	Loan status (0 is non-default, 1 is default)
loan_percent_income	Percent income
cb_person_default_on_file	Historical default
cb_preson_cred_hist_length	Credit history length

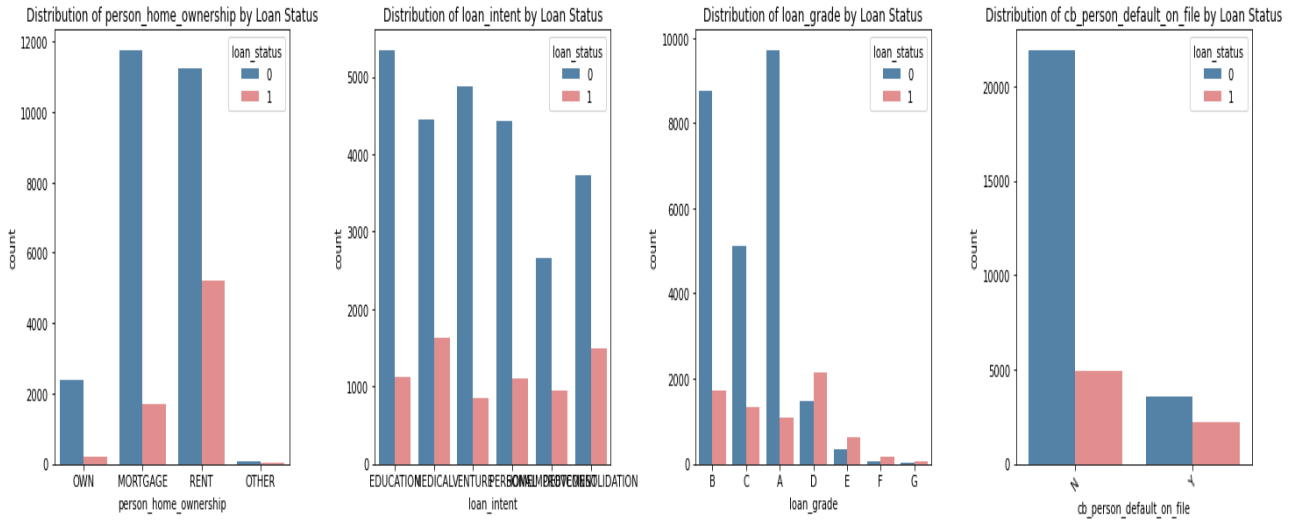
Source : kaggle

Both Employment length and Interest rate features has missing values of 895 (3%) and 3116 (9.5%) respectively. In order to keep all records, we decided to handle the missing values by applying linear imputation which estimated the linear relationship between the missing variables and the other numerical variables in the data set. The choice to apply linear imputation is motivated by:

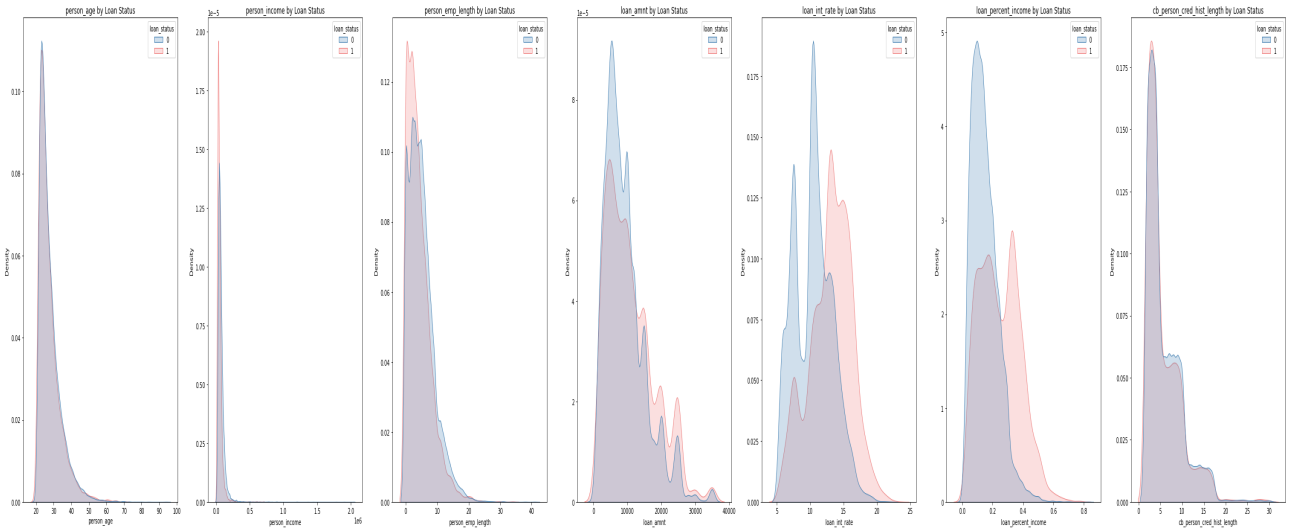
1. The very low percentage of missing data
2. The random pattern of the missing data

We then discarded outliers and values that seemed illogical in the context of our use case, namely person ages that are ≥ 123 and employment length $= 123$.

Figure 1: Brief descriptive statistics



(a) Categorical features' repartition by Loan Status



(b) Distribution of Numeric Columns by Loan Status using KDE

Source: Authors' calculations

Figure (1) Graph (a) displays the categorical features of the dataset according to loan status. The plot reveals that across various variables, such as person home ownership, loan intent, loan grade, and cb person default on file, the majority of categories have a higher proportion of no default (0) values compared to default (1) values. This could indicate a lower likelihood of default for individuals across different home ownership categories and loan intents. However, as loan grade deteriorates, the likelihood of default increases.
The majority of instances are associated with no default (0 values) This shows a clear imbalance of loan status in the dataset, which could impact the performance of our analyses and predictive models along the line, as they can be influenced by the uneven distribution of the target variable.

Indeed, the original data which consists of 32 574 records, over 78% (25 467) of which are non defaulters, meaning that just less than 22% of the data (7107) are labeled as defaulters Table (2). To address this issue, we have performed oversampling on the dataset, ensuring a more balanced representation of default and non-default instances. By oversampling, we aim to improve the accuracy and reliability of our work. Through this method, we have 50 934 records that are equally (50%) each are (0) or (1). Other than the already described data manipulation, we also performed data normalization to ensure that variables with different scales and units have a comparable impact on our modeling.

Table 2: Dataset Overview

	Original Dataset	Oversampled Dataset
Total Records	32,574	50,934
Non-defaulters	25,467 (78%)	25,467 (50%)
Defaulters	7,107 (22%)	25,467 (50%)

Source: Authors' calculations

In Figure (1) Graph (b), the distributions of person age for loan status 0 and 1 are overlapping, suggesting that age may not strongly differentiate defaulters and non-defaulters. Person income indicates that individuals with lower incomes are more likely to default. The density of person emp length is higher for lower length, indicating that individuals with shorter employment lengths have a higher likelihood of default. In the case of loan amnt, the density for loan status 0 is relatively higher for lower loan amounts, while for defaulters, the density is larger after the $12,000 mark, implying that larger loans are associated with a higher likelihood of default. Additionally, loan interest rate shows that defaulters have a higher interest rate, suggesting that riskier borrowers are charged higher interest rates. Lastly, loan percent income demonstrates that loan defaulters have a larger percentage of income dedicated to loan repayment, which is logical given their lower income, higher loan amounts, and higher interest rates.

3 Models & Results

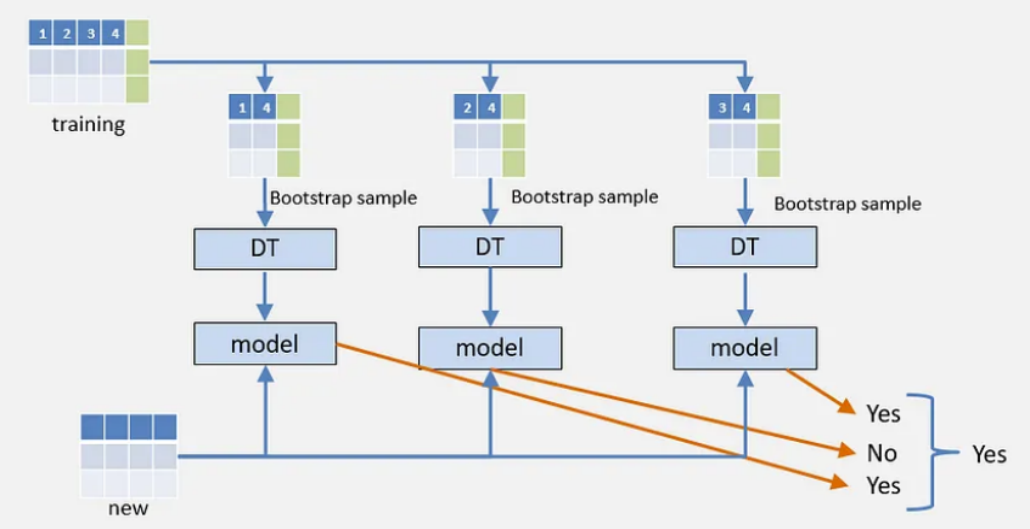
We performed Random Forest classification, XGBoost, and Neural Network models to address the classification problem. These machine learning techniques were applied to effectively classify and predict the target variable (loan status) using ensemble learning, gradient boosting, and artificial neural network approaches, respectively.

3.1 Models Description

3.1.1 Random Forest

Random Forest (Breiman, 2001) is an ensemble learning method that combines multiple decision trees by using bootstrap aggregation (or bagging) and random feature selection. Each tree in the forest is built independently on a bootstrap sample from the original dataset, where the samples are drawn with replacement. Additionally, as seen on Figure (2) at each split in the tree, only a random subset of features is considered, further enhancing the diversity of the trees. The final prediction of the Random Forest is obtained by aggregating the predictions of all the individual trees, either through majority voting in classification or averaging in regression.

Figure 2: Understanding Random Forest's Functioning

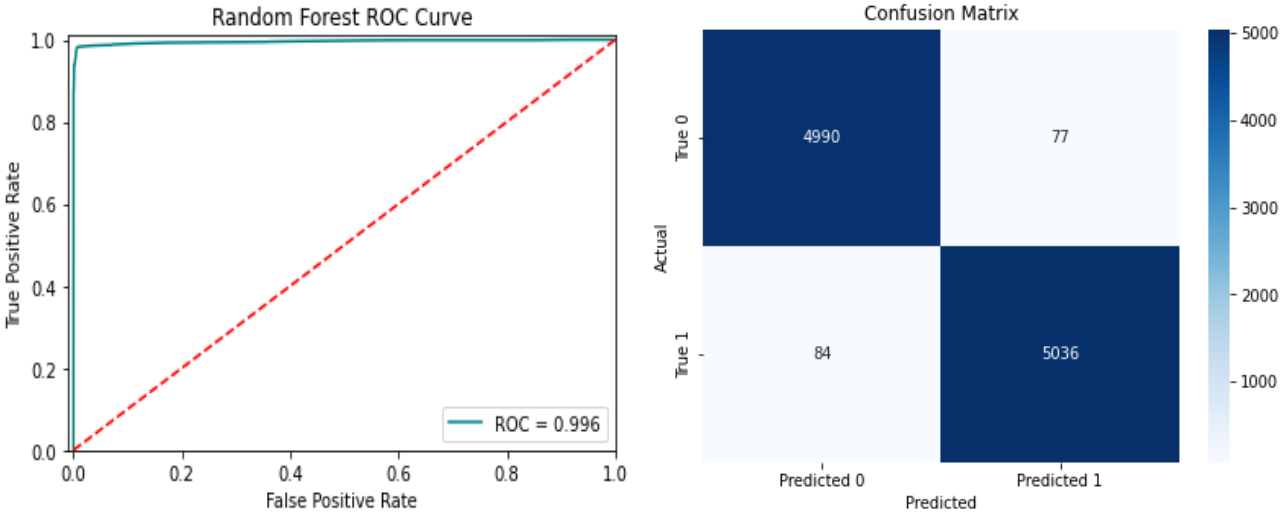


Source: Medium

On our data set the Random Forest algorithm, achieved an accuracy of over 0.983 on the test data. The training process took a only 10.42 seconds. The confusion matrix in Fig. (3) Graph (b) shows that out of 5067 records labeled as 0 (non-defaulters), 4990 were correctly predicted (with only 77 false positives), and for the 5120 instances labeled as 1(defaulters), 5036 were correctly predicted (with 84 false negatives). The classification report demonstrated high precision, recall, and f1-scores for both classes. These results emphasize that the Random Forest algorithm performs exceptionally well in credit risk classification, achieving accurate predictions within a short training time. The ROC (Receiver Operating Characteristic) seen in Fig. (3) Graph (a) yielded an exceptional score of 0.996, indicating its strong ability to discriminate between default and non-default instances.

We also performed cross validation to avoid any potential model overfitting, with a mean score of 0.98, it confirms that the Random Forest model demonstrates high accuracy and generalization performance on the training data.

Figure 3: Random Forest Performance Evaluation



(a) ROC Curve

(b) Confusion Matrix

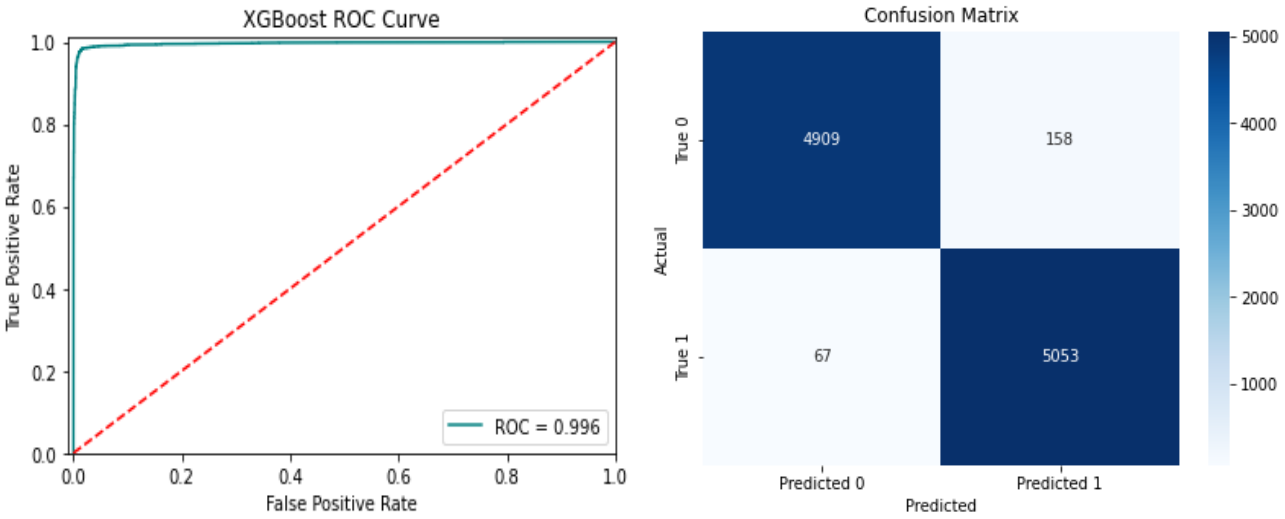
Source : Authors' calculations

3.1.2 Extreme Gradient Boosting

Extreme Gradient Boosting (Chen and Guestrin, 2016) is an optimized and scalable implementation of gradient boosting that leverages a combination of regularization techniques, parallel processing, and tree pruning. It is a system that focuses on performance and scalability, aiming to deliver efficient and accurate models for various machine learning tasks.

XGBoost is one of the algorithms we used for this project. To optimize its performance and avoid overfitting, we performed a grid search to identify the best set of hyperparameters. The model is fairly fast, fitting the training data (80% of the dataset) in only 66 seconds. It achieved an impressive ROC AUC of 0.98 Fig. (4) Graph (a) and an accuracy of 0.98 on the test data.
On the test data, the confusion matrix Fig. (4) Graph (b) reveals that out of 5067 records labeled as 0, 4909 were correctly predicted (158 falsely labeled as defaulters), while for the 5120 records labeled as 1, 5053 were correctly predicted (67 falsely labeled as non defaulters). The classification report demonstrates high precision, recall, and f1-scores for both classes.
To ensure the generalization of the model, we performed cross-validation to mitigate overfitting issues. Overall, these results indicate that XGBoost with carefully tuned hyperparameters shows great potential for credit risk classification tasks.

Figure 4: XGBoost Performance Evaluation



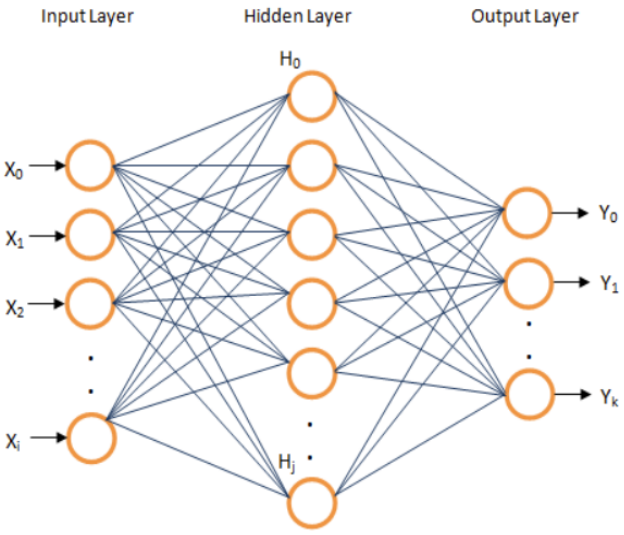
(a) ROC Curve (b) Confusion Matrix

Source : Authors' calculations

3.1.3 Neural Networks

A Multi-Layer Perceptron (MLP) as seen on Figure (5) is a type of Artificial Neural Network (ANN) that consists of multiple layers of interconnected nodes (neurons) Taud and Mas (2018). It is a powerful modeling tool that applies a supervised training procedure using labeled data examples. MLPs are capable of handling linear and nonlinear functions by learning from the relationships in the data and generalizing to unseen situations. By combining multiple layers of neurons and employing activation functions, MLPs can create complex models that enable the prediction of output data based on given input data.

Figure 5: Multi-Layer Perceptron



Source: dotnetlovers

For the purpose of this project, we carried out a sequential neural network model with three layers. It consists of an input layer with 32 neurons, a hidden layer with 16 neurons, and an output layer with a single neuron. The model uses the ReLU activation function in the hidden layer and the sigmoid activation function in the output layer for binary classification. The training is performed for 42 epochs with a batch size of 32 Table (3).

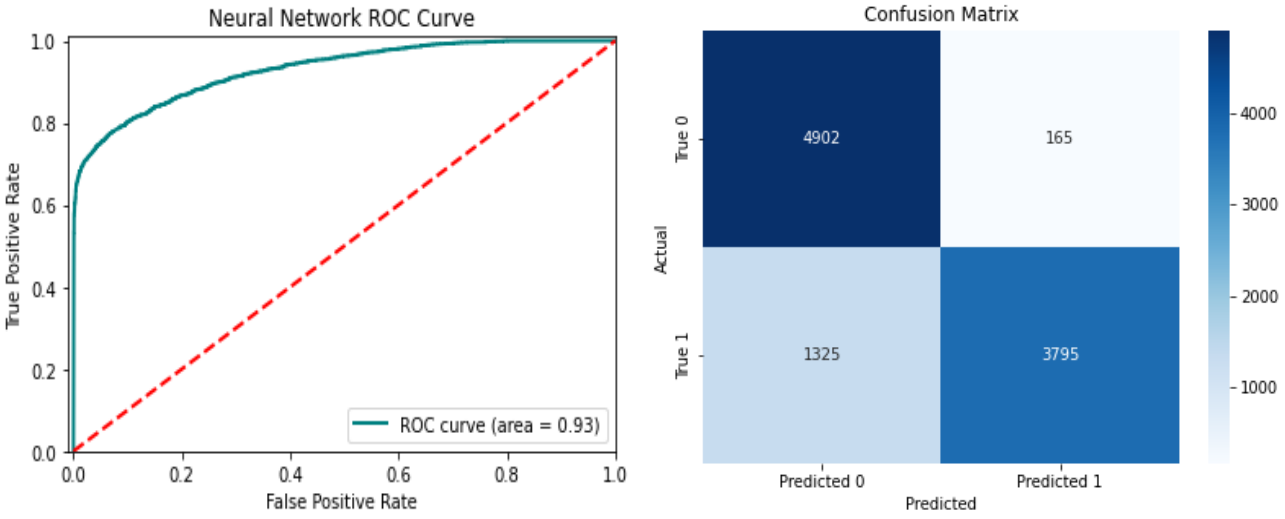
Table 3: Model Configuration and Training Details

Layers	**Neurons**	**Activation Functions**
Input Layer	32	-
Hidden Layer	16	ReLU
Output Layer	1	Sigmoid
Training Details	**Epochs**	**Batch Size**
-	42	32

Our model achieved an accuracy of 0.85 on the test set. From Fig. (6) Graph (b) we can see that the model correctly predicted 4902 of non-default records (true negatives) and 3795 records as default (true positives). However, it has over 165 records of false positives and 1325 of false negatives, which is the largest of any models. The precision for no default was 0.80, For default the precision was 0.96 suggesting that when the model predicts an instance as default, there is a high likelihood that it is indeed a default.

In Fig. (6) Graph (a), we can clearly see that this model has the lowest AUC suggesting that the neural network had relatively less ability to differentiate between the two classes compared to the random forest and XGBoost models.

Figure 6: Neural Network Evaluation

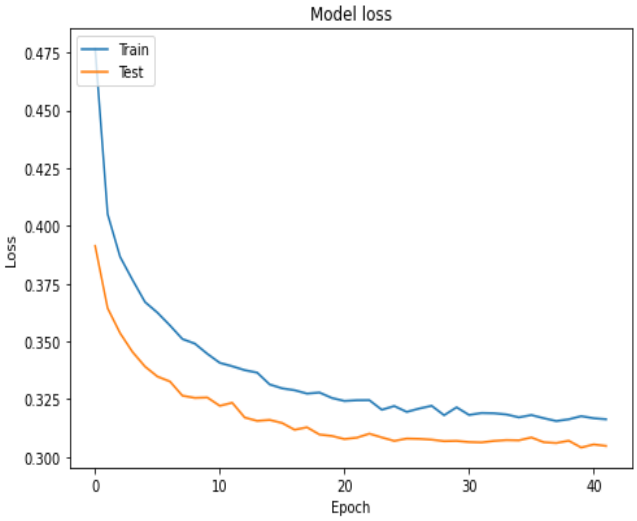


(a) ROC Curve (b) Confusion Matrix

Source : Authors' calculations

In Fig. (7) loss plot, both the train and test curves show a decreasing trend as the number of epochs increases. The fact that the test curve follows a similar trend to the train curve suggests that the model is not overfitting and is able to generalize well to unseen data. The convergence of both curves to lower loss values further indicates that the model is learning and improving its performance.

Figure 7: Neural Network Model Loss



Source : Authors' calculations

3.2 Results Overview

As indicated in Tab. (4) The random forest and XGBoost models outperformed the neural network in this specific scenario. The higher performance could be attributed to several factors. Firstly, random forest and XGBoost are ensemble learning methods that combine multiple decision trees, which can capture complex relationships in the data in a very effective way. Secondly, the random forest and XGBoost models demonstrated better accuracy, precision, recall, and F1-scores, indicating their ability to handle the specific classification task more accurately. It is important to note that model performance is data-dependent, and the performance of random forest and XGBoost on this dataset does not guarantee the same outcome on different data or on larger-scale data.
In fact, when considering big data, neural networks may be worth considering as they can be faster to train, are highly scalable and can capture non linearity in the data.

Table 4: Performance Metrics of Different Models

Model	Accuracy	Precision		Recall		F1-Score	
		(Label 0)	(Label 1)	(Label 0)	(Label 1)	(Label 0)	(Label 1)
Random Forest	0.984	0.98	0.98	0.98	0.98	0.98	0.98
XGBoost	0.977	0.99	0.97	0.97	0.99	0.98	0.98
Neural Network	0.854	0.79	0.96	0.97	0.74	0.87	0.84

Source : Authors' calculations

4 eXplainable Artificial Intelligence

Models in machine learning often achieve high performance in terms of predictive accuracy but lack explainability. There is however often a tradeoff between performance and explainability, where the most (likely) accurate models are least explainable, while the most explainable models like decision trees are less accurate. Explainable AI (XAI) systems aim to make model behavior more understandable by providing explanations. However, the context in which explanations are provided depends on the task, user abilities, and expectations. Therefore, definitions of interpretability and explainability are domain-specific and cannot be universally defined (Gunning et al., 2019).

4.1 XAI Categorization

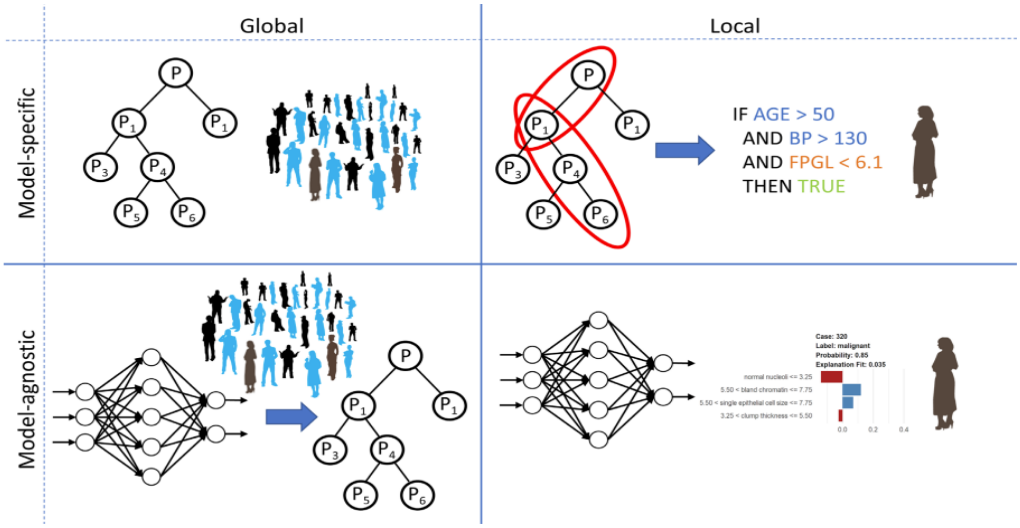
4.1.1 Model-Specific vs. Model-Agnostic Interpretability

Model-specific interpretation methods are limited to specific models and derive explanations by examining internal model characteristics. In contrast, model-agnostic methods, can be used on any machine learning model and treat the model as a black box (Stiglic et al., 2020). They offer flexibility and allow for comparison across different models, but they may face challenges in achieving a global understanding of complex models. In cases where exact explanations are necessary or interpretability takes precedence over model performance, model-specific should be considered (Elshawi et al., 2019).

4.1.2 Local vs. Global Interpretability

Local interpretability focuses on explaining individual predictions or decisions, it explain how a specific decision was taken. Global interpretability on the other hand, provides transparency about the model's behavior on an abstract level, giving insight into what is happening inside the model (Stiglic et al., 2020).

Figure 8: Visual representation of interpretability categories for ML-based predictive modeling



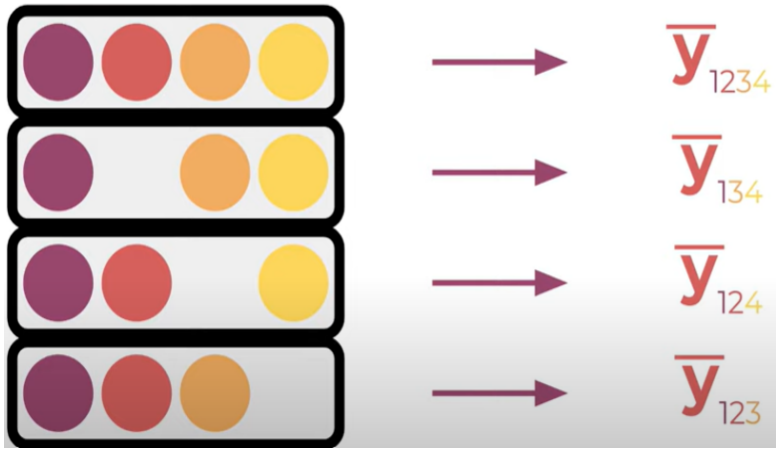
Source : (Stiglic et al., 2020)

4.2 Used libraries

4.2.1 SHapley Additive exPlanations

Shapley values is a concept derived from coalition game theory. In a coalition game, it involves distributing the gains fairly to each player according to their contribution to the outcome. To do so, all sub-coalitions and their payoffs are considered, in order to calculate the exact contribution of a player (Štrumbelj and Kononenko, 2014). Shap is **Local, Model-Agnostic** explanation model which simplifies the original complex models. It is defined as any interpretable approximation of the original model (Lundberg and Lee, 2017).

Figure 9: Shapley values feature permutation



Source : Youtube

In the specific case of shap package, we approximate the shapley values by passing values to the model of various permutations of the data we're trying to explain and replacing the feature studied with a fixed set of synthetic data. Then for each feature we take the average of the model output for all synthetic data points for that specific feature as the model output for that particular feature permutation which is the contribution for that feature. The contribution is then assigned by comparing the results with and without that specific feature.

4.2.2 DALEX

DALEX is a **model-agnostic**, **Global explainer** (for model understanding) and **local explainer** (for prediction understanding) (Baniecki et al., 2021) (Biecek, 2018). The model-agnostic approach DALEX offers is based on a permutational approach and works with any predictive model. It implements an interface for interactive explainability and fairness allowing for an interactive model and explicability analysis. DALEX is proposes both shapley values and Permutation Feature Importance (PFI) which is a technique for determining the importance of different features in a machine learning model by evaluating how much the model's performance decreases when the values of a particular feature are randomly shuffled, thereby breaking its correlation with the target variable. Unlike, shapley values who take into account the interactions between features, PFI breaks the relationship between the features and the target.

- **ELI5**

ELI5 Explain Like I'm 5, is a Python package which helps to debug machine learning classifiers and explain their predictions. ELI5 is a **Local, Model-Agnostic** tool.

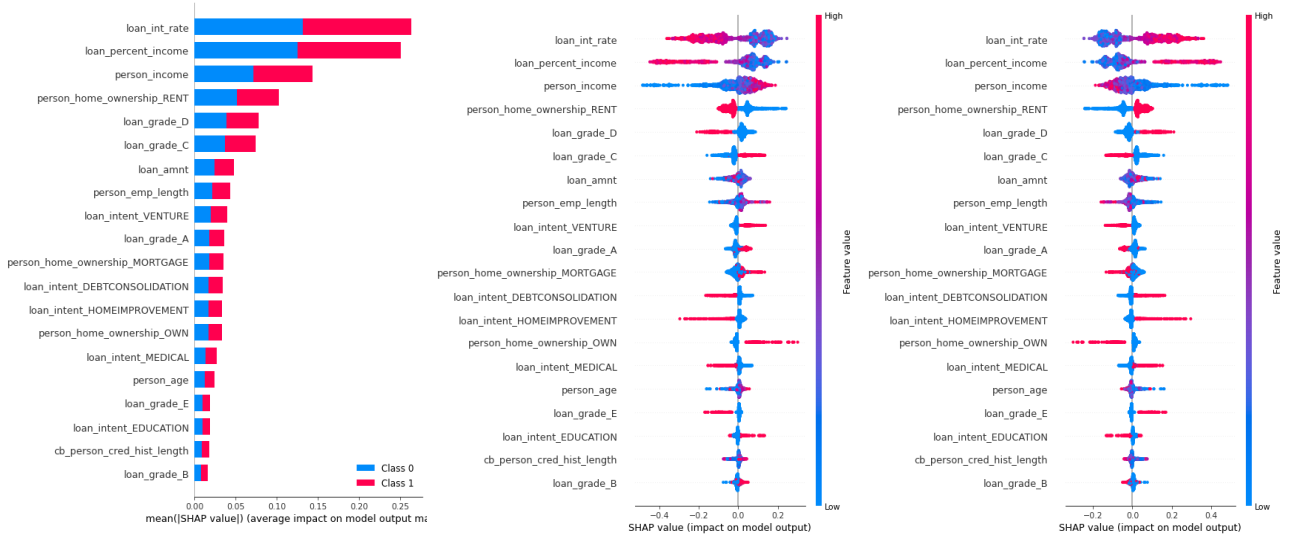
4.3 Results

4.3.1 Random Forest

- **Shap library**

Figure (10) shows Shapley results for 2000 datapoints. Figure (10) Graph (a), displays the absolute mean shap values for both classification classes. The Features are displayed according to their importance in influencing the model's output. We can see that loan interest rate, slightly followed by loan percent income impact the model's output regardless of the class. They are followed directly by person income. It is interesting to see that both classes are equally influenced by the features. Figure (10) Graph (b), plots the distribution of shapley values for each feature for all the data points according to 'No Default' class. It shows the features that drove the model to the 'No default' prediction. OWN category in person home ownership drive the model to predict 'No Default'. The higher the income the more the model's output will associated with 'No Default', whereas, loans associated with grade 'D' and 'E' drive the model into predicting Default.
Figure (10) Graph (c), plots the feature importance for all the data points according to 'Default' class. This plot reverses the values of Graph (b), therefore showing, that loan intent associated with 'HOME IMPROVEMENT' and 'DEBT CONSOLIDATION' are linked with a higher probability of default.

Figure 10: Shapley values for Random Forest



(a) Mean Feature Impact on Model Output per Class

(b) Feature Importance for 'No Default' Class

(c) Feature Importance for 'Default' Class

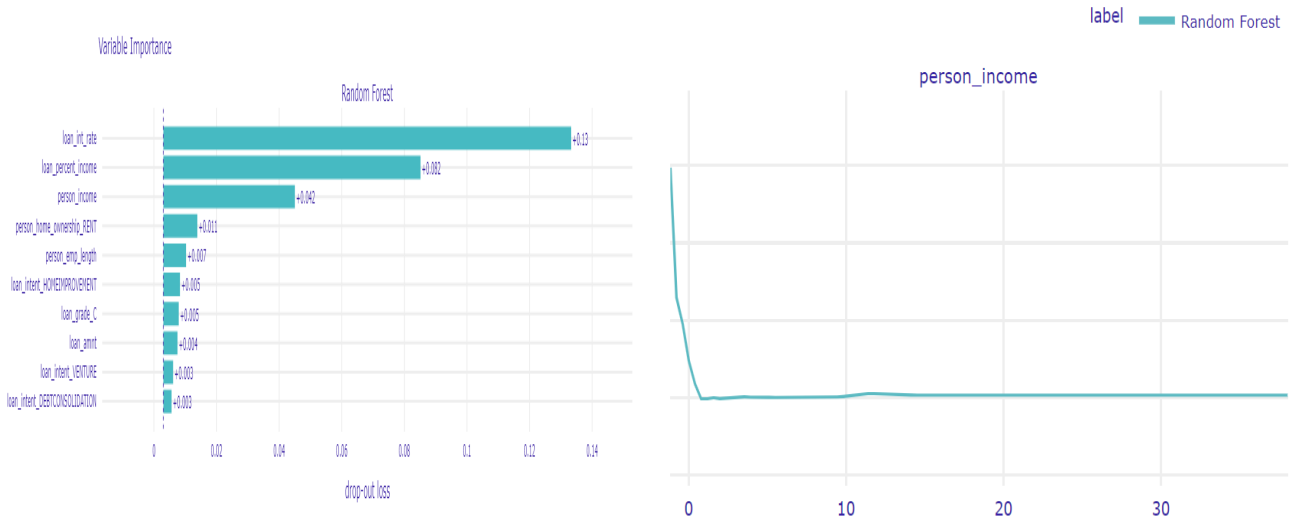
Source: Authors' calculations

- **DALEX library**

Figure (11) Graph (a), shows the importance of each feature on the predictions of the overall model. It is interesting to see that Dalex ranked the top features similariliy to shap in Figure (10) Graph (a). However, their Dalex importance range is different.

This library offers a module called **Model Profile**, which separately shows for each feature the influence of its values on the output. Figure (11) Graph (b) is the Model's profile for *Person Income*. We can see that extremely low values of income are associated with Default, but only for a small portion, once it is exceeded, the model prediction go towards no default.

Figure 11: DALEX Feature Contribution



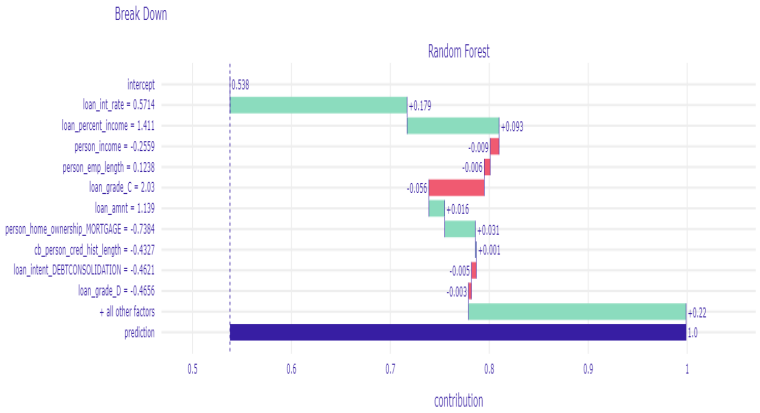
(a) Feature Importance

(b) Income Feature Influence on Output

Source : Authors' calculations

Figure (12) we computed the model's break down prediction for a single default value. For this specific person, their income, employment length and loan grade slightly pulled the prediction towards no default, whereas the remaining values all pulled the model to output default.

Figure 12: Feature Contribution Breakdown for a Default Prediction



Source : Authors' calculations

4.3.2 XGBoost

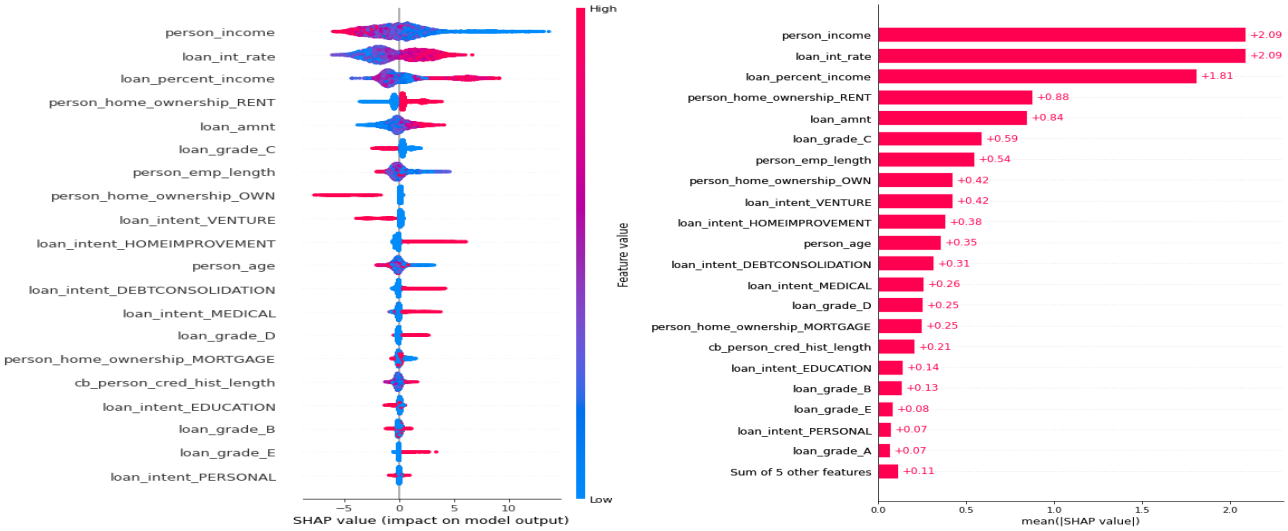
- **Shap library**

Fig. (10) Graph (a), plots an overview of the most important features displayed for every data point in the test set. It shows the impact of each feature on the model output. The plot proves that *person income* lowers the possibility of default. Higher values for both *loan interest rate* and *loan amount* increase the default predictions. We can clearly see that status of *home ownership* OWN is really pushing the model to predict no default, while some data point are close to 0, they have no predictive value on the model. *Home intent* IMPROVEMENT pushes the model to output default predictions. Fig. (11) Graph (b) display the model's absolute mean shapley values for each feature, indicating the overall contribution to the model's output. The plot indicates that both *person income* and *interest rate* have the most significant impact on the model's predictions.

- **DALEX library**

Fig. (14) display the XGBoost feature importance similarly to Fig. (11) Graph (b). We can notice here that both predictions gave similar importance to the top features namely *loan interest rate, person income,...* These features appear in both plots but not necessarily in the same rank. This indicates that these features are consistently important in influencing the model's predictions, but their relative importance varies according to the calculation method.

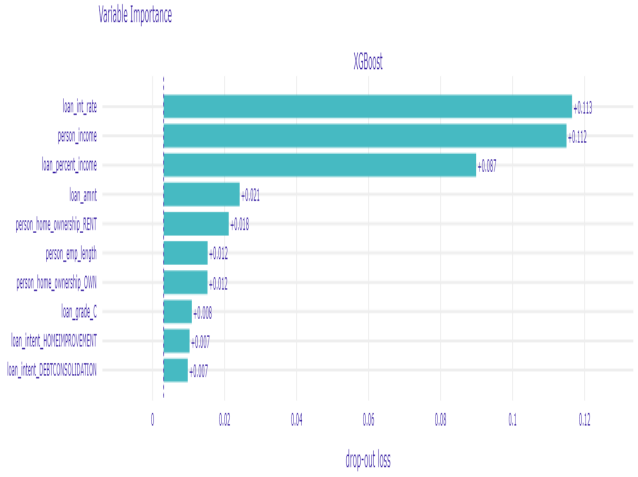
Figure 13: Shapley values for XGBoost



(a) Feature Importance (b) Absolute Mean Features Contribution

Source : Authors' calculations

Figure 14: DALEX XGBoost Feature Importance



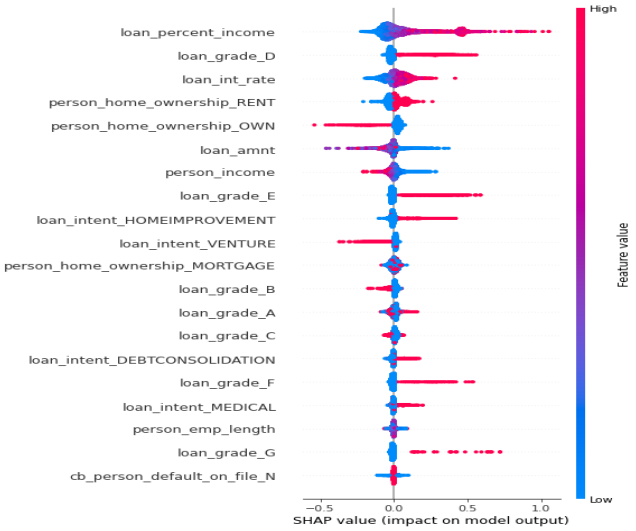
Source : Authors' calculations

4.3.3 Neural Network

- **Shap library** Figure (15) displays the neural network features' importance across all test set data points. Even though all three models achieve a fairly good performance, it is evident from the figure that they arrive at this performance in different ways. This becomes clear when we see that they do not attribute the same importance level to the features. The neural network decision is highly influenced by *loan percent income*, with higher values pulling towards default predictions. Inversly, both *loan amount & person income*, higher values drive the impact to the

left (no default predictions).

Figure 15: Neural Network's Shapley Feature Importance

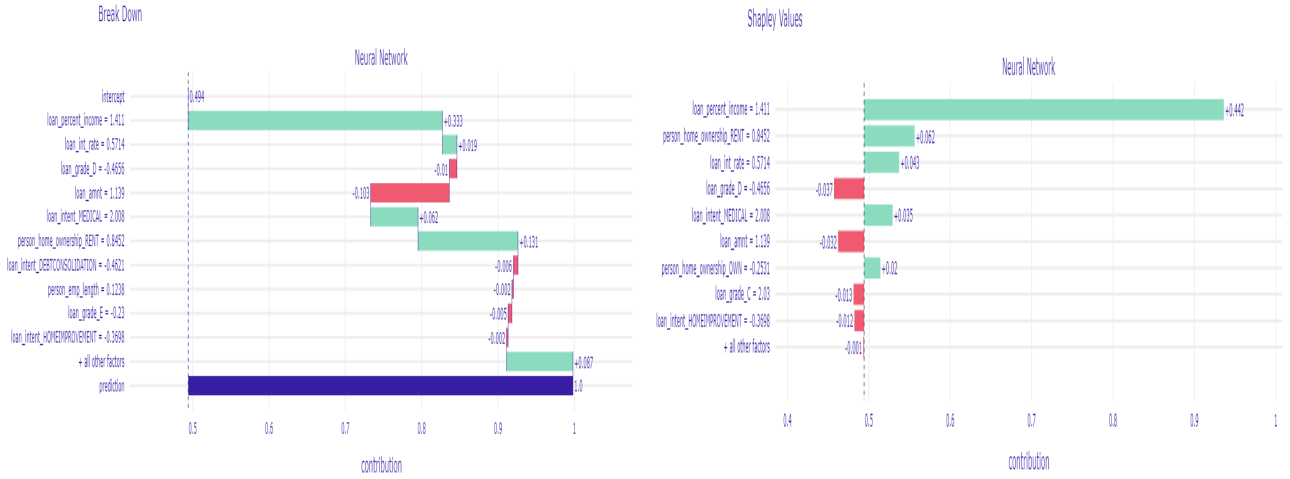


Source : Authors' calculations

- **DALEX library**

Dalex proposes two ways to calculate the features contributions namely **Permutation Feature Importance** and **Shapley Permutations**. In Figure (16), we computed both methods for a single prediction. Graph (a) shows features importance according to PFI while Graph (b) shows the importance according the shapley values. We can see that both calculation methods even though they arrive to the same results, they do not attribute the same feature importances. The fact that they do not perfectly align suggests that they capture different aspects of feature importance which we attribute to the underlying principles of each method.

Figure 16: DALEX Single Default Prediction Values



(a) PFI Feature Importance Break Down

(b) Shapley values Feature Importance

Source : Authors' calculations

4.3.4 ELI5 & Model-Specific Explailability

- **ELI5**

Keeping the same single prediction used earlier in the report, we computed the explanation for both Random Forest and XGBoost models through ELI5. Figure (17) shows the feature importance for each model respectively in Graph (a) representing the Random Forest model and graph (b) representing the XGBoost model. As expected, the feature importance and relationships captured by these two models are different.

The discrepancy in feature importance and relationships between the Random Forest and XGBoost models emphasizes that these models have distinct characteristics and approaches. Each model has its own way of weighing and considering the features when making predictions, resulting in differences in their interpretation of feature importance. It is important to note that different models may prioritize features differently based on their specific algorithms, training data, and optimization objectives.

Figure 17: ELI5 Single Default Prediction Values

y=1 (probability **1.000**) top features

Contribution[?]	Feature
+0.499	<BIAS>
+0.319	loan_percent_income
+0.069	loan_int_rate
+0.059	person_home_ownership_RENT
+0.058	loan_amnt
+0.032	person_home_ownership_MORTGAGE
+0.012	person_home_ownership_OWN
+0.011	loan_grade_A
+0.004	loan_grade_B
+0.002	person_age
+0.001	loan_intent_MEDICAL
+0.001	loan_intent_EDUCATION
+0.001	loan_intent_VENTURE
+0.000	loan_intent_PERSONAL
+0.000	loan_intent_HOMEIMPROVEMENT
+0.000	person_home_ownership_OTHER
-0.000	loan_grade_G
-0.001	loan_grade_F
-0.001	person_emp_length
-0.001	cb_person_default_on_file_N
-0.002	loan_intent_DEBTCONSOLIDATION
-0.002	cb_person_cred_hist_length
-0.002	cb_person_default_on_file_Y
-0.004	loan_grade_E
-0.005	person_income
-0.024	loan_grade_C
-0.026	loan_grade_D

(a) Random Forest Explainability

y=1 (probability **1.000**, score **8.726**) top features

Contribution[?]	Feature
+5.644	loan_percent_income
+2.733	person_home_ownership_RENT
+1.047	loan_amnt
+0.705	person_home_ownership_MORTGAGE
+0.372	loan_intent_MEDICAL
+0.219	person_home_ownership_OWN
+0.114	loan_grade_B
+0.078	person_age
+0.065	loan_intent_EDUCATION
+0.064	cb_person_default_on_file_N
+0.014	loan_grade_A
-0.001	<BIAS>
-0.005	person_home_ownership_OTHER
-0.005	loan_grade_F
-0.006	loan_grade_G
-0.014	loan_grade_E
-0.017	loan_intent_VENTURE
-0.018	loan_intent_DEBTCONSOLIDATION
-0.020	loan_grade_D
-0.059	loan_intent_PERSONAL
-0.085	loan_intent_HOMEIMPROVEMENT
-0.185	cb_person_cred_hist_length
-0.297	loan_int_rate
-0.419	person_income
-0.562	loan_grade_C
-0.636	person_emp_length

(b) XGBoost Explainability

Source : Authors' calculations

- **Model-Specific Global Explainability**

Both Random Forest a XGBoost algorithms have inherent model-specific explainability methods due to their structure. They provide feature importance rankings. However, the built-in capacities are limited to Global explanation and are not tailored to capture complex relationships in the data nor fine-grained explainability. Which is why we decided to keep this category of explainability out of the scope of this paper.

4.3.5 XAI Overview

Table 5: Comparison of models and XAI tools

Model	SHAP	DALEX	ELI5	Model-Specific
Random Forest	X	X	X	X
XGBoost	X	X	X	X
Neural Network	X	X		

5 Conclusion

In conclusion, after implementing Random Forest, XGBoost, and a Neural Network, we observed that both Random Forest and XGBoost achieved excellent performance compared to Artificial Neural Network. This highlights that complex models are not always the answer for achieving high performance. We then implemented SHAP, DALEX, and ELI5 for explainability of our models. The results highlighted the importance of feature selection and demonstrated that while models had similar predictive capacities, they achieve the results differently, therefore the XAI tools output different feature significance.
We also noticed that SHAP and DALEX offered a much more detailed explanation compared to ELI5, they provided insights that could be further leveraged to improve the model's performance in an iterative process. However, ELI5, with its simplicity and direct approach, offered us a quick and efficient way of interpreting the models.
It is important to highlight that the choice of machine learning models and the corresponding XAI tools are highly associated with the context, specific business needs, and the users they are destined for.

References

H. Baniecki, W. Kretowicz, P. Piatyszek, J. Wisniewski, and P. Biecek. dalex: Responsible machine learning with interactive explainability and fairness in python. *Journal of Machine Learning Research*, 22(214):1–7, 2021. URL http://jmlr.org/papers/v22/20-1473.html.

P. Biecek. Dalex: Explainers for complex predictive models in r. *Journal of Machine Learning Research*, 19(84):1–5, 2018. URL https://jmlr.org/papers/v19/18-416.html.

L. Breiman. Random forests. *Machine learning*, 45:5–32, 2001. URL https://link.springer.com/article/10.1023/a:1010933404324.

T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016. URL https://arxiv.org/pdf/1603.02754.pdf.

R. Elshawi, M. H. Al-Mallah, and S. Sakr. On the interpretability of machine learning-based model for predicting hypertension. *BMC medical informatics and decision making*, 19(1):1–32, 2019. URL https://doi.org/10.1186/s12911-019-0874-0.

D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, and G.-Z. Yang. Xai—explainable artificial intelligence. *Science robotics*, 4(37):eaay7120, 2019. URL https://www.science.org/doi/abs/10.1126/scirobotics.aay7120.

S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017. URL https://arxiv.org/pdf/1705.07874.pdf.

G. Stiglic, P. Kocbek, N. Fijacko, M. Zitnik, K. Verbert, and L. Cilar. Interpretability of machine learning-based prediction models in healthcare. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(5):e1379, 2020. URL https://doi.org/10.1002/widm.1379.

H. Taud and J. F. Mas. Multilayer perceptron (mlp). In *Geomatic approaches for modeling land change scenarios*, pages 451–455, 2018. URL https://link.springer.com/chapter/10.1007/978-3-319-60801-3_27.

E. Štrumbelj and I. Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41:647–665, 2014. URL https://doi.org/10.1007/s10115-013-0679-x.