VRIJE UNIVERSITEIT AMSTERDAM

Interpretable AI: On the explainability of machine learning decision-making through model-agnostic explainability methods

By Canalytics Group:

Nando Beijaard (2624391) c.f.beijaard@student.vu.nl
Nick Peetam (2704149) n.r.peetam@student.vu.nl
Kevin Landegent (2663630) k.landegent@student.vu.nl
Rony Munnik (2674125) r2.munnik@student.vu.nl

A Thesis Submitted In Partial Fulfillment Of The Requirements For The Degree Of

Bachelor of Science in Business Analytics

Amsterdam, Noord-Holland, The Netherlands
June, 2023

SUPERVISOR

Prof. Dr Shujian Yu

THESIS COORDINATOR

Prof. Dr Sandjai Bhulai

Abstract

This research paper delves into explainable Artificial Intelligence (AI), focusing on the explainability of machine learning decision-making processes. The study explores the application of model-agnostic methods, aiming to make the often 'black box' models more understandable and transparent. A significant part of our research involves the implementation of Kernel SHAP, a universal adaptation of the Shapley Additive Explanations (SHAP) method. as well as Local Interpretable Model-agnostic Explanations (LIME). This research focuses on comparing both techniques and validating the results with the leave-one-covariant-out (LOCO) method. Furthermore, it investigates how the information uncovered by these methods increases the ability to detect bias. Our finding showed the reliability of the two methods studied and their power to provide valuable insights for both researchers and practitioners in all sectors associated with AI.

1 Introduction

Machine Learning algorithms have risen in popularity in the flourishing area of Artificial Intelligence (AI) due to their ability to extract meaningful insights from vast amounts of data and their applicability in numerous fields. This rise, however, is not without any critiques; there have been some challenges with this adoption, specifically, the need for more transparency and interpretability of these machine learning models. Deep learning, a prime example, allows machines to process and learn hierarchical data representations automatically, such as those used in classification tasks. However, interpreting the decision-making behind the output often requires more work.(Linardatos et al., 2020)

In 2021, the European Parliament proposed new privacy and transparency regulations on AI. These regulations address these concerns, moving away from black box models and ensuring machine learning models' lawfulness, safety, and trustworthiness. It is crucial to have explainability in high-risk AI systems that deal with sensitive environments or data; this is also the focus of the regulations. This legislative move signals a shift in AI, pushing for more transparency and ethical practices in machine learning model utilisation.

This research was motivated by the pressing need for interpretability in crowd flow prediction. The client collaborated with us to gain deeper insights into the citywide crowd flow prediction models. The goal was to identify potential biases in these models, particularly concerning a prominent touristic area where their measurements and predictions exhibited discrepancies. As a result, this study investigated one of their chosen machine-learning algorithms.

The tragedies that unfolded during the Travis Scott Astroworld Festival in 2021 and the crowd crush incident at Seoul Halloween in 2022 are stark reminders of the severe consequences that can arise from deficiencies in crowd flow prediction and management systems. Crowd flow prediction exemplifies a high-risk system dealing with data concerning people and their behaviour. Enhancing the trust in and the accuracy and efficiency of prediction models can improve safety measures and enable timely interventions when necessary.

Similarly, another unique high-risk system lies in condition diagnosis within the healthcare domain, which forms the basis of the dataset utilised in this research. Machine learning can play a significant role in this domain, offering valuable insights. However, it is crucial to acknowledge that relying solely on predictions without considering interpretability can pose substantial dangers.

Researchers and developers are constantly developing methods to gain insights into model decision-making with the increasing relevance and popularity of Explainable AI (XAI). Key among these XAI methods are the two widely accepted and standard procedures, SHAP (SHapley Additive exPlanations) using Shapley values and Local Interpretable Model-Agnostic Explanations (LIME), as discussed by (Molnar, 2022). These techniques revolve around feature- and instance importance, which refers to each input variable or feature's impact on a model's global and local prediction. Understanding which features or data points are the most important for the model is of great value and can be used to improve performance, identify biases, and provide valuable insights into the model's decision-making.

Building on the discussion of established methods for model interpretability, it is essential to recognise the ongoing efforts and challenges in the field. Researchers are striving to enhance the reliability of these techniques and improve their com-

munication with stakeholders. (Bhatt et al., 2020) used interviews to develop a framework encompassing clear goals of explainability, considering stakeholders, and discussing standard practices in explainability. Research as (Saarela and Jauhiainen, 2021) on different feature importance methods suggests a combination of measures for more reliable and trustworthy results, particularly on local explanations. However, significantly more research must be done on dealing with limitations and effectively communicating these methods' results to stakeholders. (Bhatt et al., 2020)

Through this research, we aim to fill this gap and contribute to the ongoing discussion on transparency and explainability in AI. This study will primarily focus on established techniques for model interpretability, aiming to utilise their reliability and transparency to improve communication with stakeholders by providing clear insights into the models. With this, stakeholders will be able to comply with the forthcoming regulations while expanding the utility of their machine-learning models. This paper delves into the established methods of SHAP and LIME, their backgrounds, and how they can be applied in practical settings. It also explores how the results of these methods can be effectively communicated to stakeholders through validation and visualisations. In addition, we will show how far these methods go in explaining the model's decision-making and uncovering its bias by answering the following research question:

How effectively can model-agnostic explainability methods like SHAP and LIME reveal machine learning models' biases and decision-making processes?

We will focus on the importance of explainable AI due to the lack of- and need for trust in specific fields for proper implementation. This need for trust is significant in sectors less familiar with machine-learning models while dealing with sensitive data or decision-making.

2 Literature Review

2.1 Terminology

This section defines and explains the key terms and concepts in this research on machine learning interpretability for decision-making in AI with a focus on LIME and SHAP and high-risk systems.

- **Explainable AI (XAI)/Interpretability** refers to the operation of narrowing the gap between complex (black box) machine learning models and human understanding. They address the need for transparency in decision-making processes focused on predictions that often carry irreversible consequences, such as decisions that concern society as a whole or a person's health.

- **Data and Cost Imbalance** refers to the uneven distribution of classes in the dataset and the associated costs or consequences of classification errors in a machine learning model. In cost-sensitive learning, it is crucial to consider the varying severity of misclassifications, particularly in high-risk applications such as healthcare. Table 1 illustrates the confusion matrix used to evaluate classification performance.

Actual / Predicted	**Positive**	**Negative**
Positive	Correct decision	Type I error
Negative	Type II error	Correct decision

Table 1: Confusion Matrix: Type I and Type II Errors

- **Local Explanation**, in the context of machine learning, local explanation refers to the process of explaining individual predictions or decisions made by a model. It involves identifying the factors or features that contributed to a specific output or outcome and, for example, attributing weights to the contribution of the factors or features to give insights into the decision-making. Local explanations can help to increase transparency and trust in machine learning models and allow for better debugging and model enhancement.

- **Global Explanation** Global explanation refers to the ability of a model to provide full insight into its decision-making process across all possible instances. This stands in contrast to local explanations, which focus on individual predictions. A global explanation is often used for model interpretability on a broad scale. For instance, it could help explain how various features are weighted and interact within the model. Achieving global explanations can be challenging due to many machine

learning models' complexity and non-linear nature, especially with neural networks.

- **Explainability Methods**
  - **Model-Agnostic Methods** are a key component of XAI. These methods aim to improve interpretability by providing a general approach to understanding (a part of) the decision-making process. They achieve this by not focusing on the internal workings of the model but rather approximating the model by analysing its transformations on changes in the input data.
  - **Model-Based Methods**, on the other hand, enhance the interpretability of a machine learning model by leveraging its internal workings, parameters, or architecture. These methods are designed with a specific model or group of algorithms in mind, making them less applicable to other models. While model-based methods are instrumental in XAI, this research focuses on a universal approach to interpretability, so it was decided not to include these methods.
- **Shapley Values** give a better understanding of how much each feature contributes to the overall prediction of the model and originate from cooperative game theory (Roth, 1988). To determine the Shapley value of a feature, we must factor in all potential combinations of features, including the current one, and assess the effect on the prediction when a feature (or a group of features) is added or removed. By averaging the marginal contributions of a feature across all possible coalitions, we can arrive at its Shapley value. They provide a fair importance distribution by evaluating each feature's marginal contribution to the prediction. Furthermore, one of the main advantages and possibly reasons why Shapley values are so popular in XAI is that its properties make it reasonable to be considered a fair payout and guarantee fairly distributed differences between predictions and the average prediction, which makes it stand out from other explainability methods.(Molnar, 2022).

In this terminology section, we have introduced a set of key terms and concepts foundational to understanding machine learning interpretability, with a particular emphasis on model-agnostic methods like LIME and SHAP. These terms form the basis for discussing the role of interpretability in AI decision-making, especially in contexts such as diabetes classification in healthcare, where decisions have critical implications. By understanding terms like Explainable AI, data and cost imbalance, global and local explanation and model-based and -agnostic methods, readers will be better equipped to understand the complexities and nuances of making machine learning models transparent and accountable.

These terms frequently reoccur as we progress through the literature review and delve into existing research, applications, challenges, and future directions in machine learning interpretability. Readers can refer to this section if they encounter a term they need to familiarise themselves with or need clarification on a concept.

Moving forward, we will explore the significance of model-agnostic methods and their application in high-stakes decision-making environments such as healthcare. Additionally, we will critically examine how these methods can help understand and mitigate biases, especially in datasets with data and cost imbalance, and discuss the potential challenges and future directions in the field.

2.2 Importance in Healthcare

The usage of data in the healthcare sector to predict severely ill patients' risk of death through data analysis has been conducted for approximately Forty years (Gall et al., 1984), but were highly generalised and focused on the patient's vitality instead of the type of disease present (Vincent and Moreno, 2010). Machine learning models directly assisting doctors with specific conditions is a new application of machine learning models. These models, known as clinical prediction models, can be categorised into three groups: diagnostic, therapeutic, and prognostic. Diagnostic models predict the probability that a certain disease is present. Therapeutic models respond to a certain form of treatment, and predictive models will have a well-defined outcome for an individual patient (Labarère et al., 2014). For example, different studies have been conducted to predict certain diseases (diagnostic) (Dalal et al., 2022) and assess the severity of a certain health problem (prognostic) (Li et al., 2022). For all three types of models, the benchmark is that these models have sufficient accuracy of prediction. However,

more factors must be assessed before these models can be used widely within the sector.

Predicting if a patient has contained a certain disease through machine learning can be beneficial only if the doctor trusts the model used. Trust is an important characteristic for a model to have practical use. When a model is not trusted by its operator, who is literate within the field of study, it will never be used to assess a case.

In addition, A model can perform well on a dataset of a certain healthcare facility, but performance can deteriorate when implemented at another facility, as discovered by (Reps et al., 2022). Therefore, the doctor must understand the model's inner workings to validate its decision-making process clinically, ensuring that it aligns with current medical procedures. Nonetheless, when a model does not follow the standard diagnostic procedure, it may uncover unknown critical factors and increase understanding of disease mechanisms, leading to better diagnostic and treatment strategies.

Moreover, it is important that a doctor can explain the prediction made by any model to their patients. By transparently explaining the model's predictions, patients can understand the reasoning behind treatment recommendations, leading to improved patient engagement and compliance. (van den Heuvel et al., 2022) uncovered that personalised prediction models for Parkinson's are in demand if the models' prediction can be explained.

A process to validate clinical prediction models by examining a model's transparency, performance, and ethicality has been researched by (Labarère et al., 2014). Unfortunately, The models used for prediction are often black-box models which do not possess an ability for easy explanation. These models are known to be complex, and the process proposed by Labarère lacks the power to increase such models' interpretability by looking at their decision-making process and characteristics through model-agnostic methods.

2.3 Bias

For machine learning models to discern patterns and relationships within datasets, they will become susceptible to introducing biases. The bias of a model refers to systematic errors made by these models towards specific features or combinations of features. Models with high levels of bias will generate false outputs with significant implications concerning outcomes' accuracy and fairness. This leads to the desire to minimise the bias in models. To reduce bias, it has to be discerned first, and to achieve that, it is split into two categories, data bias and model bias.

2.3.1 Data Bias

Data bias refers to the training data not being representative of real-world applications. This could lead to unwanted favouritism in the model towards one or multiple features. Three types of bias can be found in data (Gu and Oelke, 2019). The first type is covariate shift, which refers to a feature not being uniformly covered. For example, when the age feature of a database only goes up to 50 and not above. It would be fine if the age is always under 50; however, if the model would have to predict higher ages, the model would end up biased. The second type of bias is sample selection bias. This bias occurs when there is a correlation between one or multiple features in the training data that do not exist in the real-world application. Lastly, the Imbalance bias refers to when the distribution of the samples between target labels is big. The imbalance bias will be thoroughly discussed further in the report.

2.3.2 Model Bias

Model bias refers to the model showing unwanted favouritism towards a feature or subset of features. It is caused by bias passing through the machine learning pipeline, originating from the data(Hellström et al., 2020). This bias is also known as algorithmic bias and is recently being discussed by the European Union. This is caused by these biases' ability to discriminate by using features that reflect negatively on specific subsets of the data.

2.4 Data and Cost Imbalance

Class imbalance in machine learning can lead to biased model training and subpar performance. This is particularly important in healthcare, where misclassifying patients can have severe consequences. In dealing with cost imbalance (Elkan, 2001) first suggests re-balancing the training set by changing the proportion of positive and negative instances based on a cost matrix. (Sun et al., 2007) also discusses three categories of cost-sensitive classification techniques on the data level resampling, algorithm level adapting existing models and using boosting algorithms in ensemble learning is the main topic of the paper. Cost-sensitive learning

assigns higher costs to more critical errors (Turney, 2002), and ensemble techniques like XGBoost can effectively handle imbalanced datasets. Techniques like SMOTE can generate synthetic minority class samples to alleviate class imbalance issues (Kumar et al., 2022).

Addressing class and cost imbalance regarding the gravity of classification errors is critical in healthcare applications like diabetes classification. Cost-sensitive learning helps in directing the model to minimise the more costly mistakes. Coupled with interpretability methods like LIME and SHAP, it enables more informed and reliable decision-making processes.

2.4.1 SMOTE

The Synthetic Minority Over-sampling Technique (SMOTE), introduced by (Chawla et al., 2002), is a method for dealing with the class imbalance problem in machine learning. SMOTE enhances classifier performance by generating synthetic examples by interpolating between existing minority instances to balance class distribution, thus enabling better generalisation.

2.5 Model-Agnostic Explainability Methods

In this subsection of the methodology section, we will discuss the model-agnostic methods used in this research to find an answer to the original problem statement. The selected methods are SHAP (SHapley Additive exPlanations), LIME (Local Surrogate), and their background and implementations will be elaborated.

2.5.1 SHAP (SHapley Additive exPlanations)

To achieve interpretability using SHAP, a combination of Shapley values and LIME, the trained models are analysed to understand the contribution of each feature towards the model's predictions. SHAP by (Lundberg and Lee, 2017), a machine learning application of Shapley values as a linear model, is derived from cooperative game theory and calculated to quantify each feature's importance for each prediction. SHAP values provide a suitable measure of feature importance that considers interactions between features. The SHAP values can be used to generate global and local explanations, which help to understand the overall behaviour of the model and the reasoning behind individual predictions.

The properties mentioned in the terminology section regarding Shapley being a popular method are the solid theory behind the Shapley value. The four properties are defined as follows:

- **Efficiency**, where the feature contributions add up to the difference between the prediction for x and the average (expected value).

$$\sum_{i=1}^{p} \phi_i = \hat{f}(x) - E_X(\hat{f}(X)) \tag{1}$$

  The sum of ϕ_i is the sum of all feature contributions, and the right side of the equation is the difference between $\hat{f}(x)$, the prediction on x and the average prediction.

- **Symmetry**, where the contributions of two feature values i, j should be equal when the contribution to all coalitions is the same for both.

$$\begin{aligned} \phi_i &= \phi_j \\ \text{if } val(S \cup \{i\}) &= val(S \cup \{j\}) \\ \forall S &\subseteq \{1, ..., p\} \backslash \{i, j\} \end{aligned} \tag{2}$$

  Where $val(S)$ is the prediction for the feature values in Set S marginalised over all features, not in S.

- **Dummy**, if the feature does not affect the predicted value no matter which coalition it is added to, its Shapley value should be 0.

$$\begin{aligned} \phi_i &= 0 \\ \text{if } val(S \cup \{i\}) &= val(S) \forall S \subseteq \{1, ..., p\} \end{aligned} \tag{3}$$

- **Additivity**. To calculate the Shapley value for a feature in a game with combined payouts, such as a random forest or XGBoost, we can use additivity. This involves summing up the individual Shapley values for each tree and averaging them for the feature. The Shapley values for each tree are as follows:

$$\phi_i + \phi_i^+ \text{ if combined payouts: } val + val^+ \tag{4}$$

These properties make for a solid foundation for the explanations given by Shapley values and, together with the fairly distributed prediction across features, make the method of great interest and importance. This is especially the case when considering the AI Act and the required regulations

that are coming up, as these are unique properties in the current state of XAI methods.

(Molnar, 2022) mentions another advantage of Shapley, such as contrastive explanations, which suggests that, distinct from other methods like LIME, Shapley allows for comparison between predictions and subsets or single instances instead of the average prediction, which can result in valuable insights.

In addition to its advantages, Molnar also discusses the method's drawbacks. One major concern is that it involves heavy computation and is labelled "NP-Complete." "NP-complete" problems are notoriously difficult computational problems for which no efficient algorithm has yet been discovered. They offer a challenge because the time required to solve them increases exponentially as the problem increases in size. Moreover, the possibility of misinterpretation of values, the necessity for data access, and unrealistic data instances to simulate the absence of feature values from a coalition are mentioned as drawbacks.

2.5.2 KernelSHAP

The SHAP method has multiple variations, and the variation used in this paper is the kernelSHAP (Molnar, 2022). The kernelSHAP approximates the contributions of every feature for an instance, the Shapley values. It uses a background dataset to find all possible feature values and compare results. The kernelSHAP works ad follows:

First we create coalitions z'_k. *Coalitions* are sets with the same length as the model's number of features. This set contains exclusively 0's and 1's. Where one stands for the real value of the feature and zero stands for a random other value in the data for this feature.

After creating the coalitions and replacing the values according to these coalitions, these results are computed with the original model. The weights are computed for all coalitions when the results are computed for multiple coalitions. SHAP uses these weights of the coalition since the more isolated a feature is, the more information this coalition gives about the feature in this instance. To compute these weights, the following formula is used:

$$\pi_x\left(z'\right) = \frac{(M-1)}{\binom{M}{|z'|}\,|z'|\,(M-|z'|)} \tag{5}$$

In this equation, M is the number of features, and $|z'|$ is the number of original features. This formula returns the highest weights for the biggest and smallest coalition. The weighted linear regression model can be built after the weights are calculated. This model is built according to the following formula:

$$g\left(z'\right) = \phi_0 + \sum_{j=0}^{M} \phi_j z'_j \tag{6}$$

The ϕ_j' represents the Shapley values. This model is trained by minimising the following loss function:

$$L\left(f, g, \pi_x\right) = \sum_{z' \in Z} \left[f\left(h_x\left(z'\right) - g\left(z'\right)\right)\right]^2 \pi_x\left(z'\right) \tag{7}$$

Where the difference between our original model and approximation model is squared to make outliers more important, this method of SHAP has higher computational efficiency than others. However, it still explodes when more instances in the background dataset are used, and since it is an approximation, the obtained Shapley values are not the true values.

While this method is less computationally heavy than the original SHAP, it is important to remember that this method only approximates the true Shapley values. This can be explained due to it being computed over a subset of the data; as a result, the values obtained may have a margin of error.

2.5.3 LIME (Local Interpretable Model-agnostic Explanations)

Local Interpretable Model-agnostic Explanations (LIME) is a method developed by (Ribeiro et al., 2016) that provides interpretable explanations for any machine learning model's predictions. LIME is designed to explain individual predictions of any machine learning model by approximating it locally with a simple interpretable model.

The main idea behind LIME is to understand the model's behaviour by making permutations $z \in Z$ of the examined instance x and observing outcomes of these permutations on the original model f. The perturbed data points are then weighted by their proximity π_x to the instance x, and a simple interpretable model g out of a family of possible explanations G is learned on the weighted permuted dataset. To obtain the simple interpretable model, g function 8, is used. After g is determined, the feature importance $\xi(x)$, for instance, x, can be obtained. If g is a simple linear model, the weights of g can be used for feature importance $\xi(x)$.

$$\xi(x) = \arg\min_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{8}$$

To ensure the model g is a locally good approximation of f, the loss function 9 is used. To enforce simplicity, the complexity measure $\Omega(g)$ is summed with the loss function, and this sum is minimised to find model g

$$\mathcal{L}(f, g, \pi_x) = \sum_i \pi_x(z^i)(f(z^i) - g(z^i))^2 \quad (9)$$

This approach allows LIME to provide a faithful explanation of the model's decision in the local region of the instance being explained while keeping the model simple to assure interpretability. Since it is model-agnostic, it can be applied to any machine-learning model without requiring any modifications to the model itself. Furthermore, the explanations generated by LIME are interpretable, meaning humans can easily understand them. This is particularly important in fields such as healthcare, where stakeholders need to understand the reasoning behind a model's predictions before making decisions.

LIME only provides local explanations, meaning it may not accurately represent the model's behaviour in other regions of the input space. Additionally, the quality of the explanations generated by LIME can be sensitive to the choice of the local interpretable model and the method used to permutate the data.

3 Methodology

This section outlines the methodology employed in this research to achieve interpretability in machine learning models using model-agnostic methods. The focus is on two widely used model-agnostic techniques: SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations).

3.1 Data Pre-Processing

In the current paper, a dataset that satisfies the need for interpretability has been used. The dataset is about the prediction of diabetes and is available through Kaggle and can be found here (Kaggle, 2023). The predicted variable is diabetes, where zero represents a negative diagnosis of diabetes and one a positive diagnosis. The variables are explained in Table 2 considering their relationship to diabetes in the context of current medical knowledge. The values of the variable Age range from 0 to 80 in the dataset, with ages below two represented by floats, with the residual value being the age in months as a fraction.

Variable	Explanation
Age	The age of the patient in years.
Gender	Gender refers to the biological sex of the individual.
Hypertension	Whether the patient has hypertension.
Heart Disease	Whether the patient has a heart disease.
Smoking History	Smoking history of the patient.
BMI	The patient's Body Mass Index.
Hemoglobin A1c level	Patient's average blood sugar level over the past 2-3 months.
Blood Glucose Level	The patient's fasting blood glucose level.
Diabetes	Diagnosis of diabetes.

Table 2: Dataset Variables

The dataset consists of 100,000 instances but reduces to 96,146 instances after removing duplicate rows. The distribution of positive and negative diabetes diagnoses can be seen in Figure B.1. The dataset is randomly split into training (70%) and test data (30%) using the train_test_split function of the scikit-learn library. For reproduction purposes, all functions involved with randomness are set to a random state 42 as seed. One-hot encoding separates the variable "Smoking History" in current- and former smokers. The other variables created due to the one-hot encoding are removed because these were too general or provided not enough information. The head of the data is presented in Appendix A, and the correlation matrix can be found in Figure B.3.

3.1.1 Correlations

Correlation matrices are a popular tool first to indicate possible relations between variables. The value ranges from -1 to 1 and shows casual changes between 2 variables, where -1 indicates a perfect negative correlation, a value of zero suggests no correlation, and a value of one indicates a perfect positive correlation. A high correlation concerning the target variable may imply that the variable is a suitable predictor. In contrast, a high correlation between predicting variables may indicate that a variable is negligible to the model when the other is present but crucial when the other is removed.

The correlation matrix of the Diabetes Prediction Dataset in Figure B.3 shows significant positive correlations between Hemoglobin A1c levels, Blood glucose level and diabetes, which indicates that

these variables might be important variables for the predicting power of the model.

3.1.2 XGBoost Classifier

This study employs an XGBoost classifier, part of the gradient-boosting algorithms (GBA) family. As stated by (Chen and Guestrin, 2016), the XGBoost classification algorithm has great prediction power and efficiency and has gained vast popularity in recent years. The algorithm optimises a set of decision trees through an iterative process. This optimisation is achieved sequentially, minimising the loss function and enhancing the model's accuracy. The model implemented in this paper has been partially optimised and could be improved. However, since this is not the main subject of the paper, sub-optimality is accepted.

The XGBoost algorithm used in this paper has the following parameters:

- **Number of estimators:** 100
- **Maximum depth:** 5
- **Random state:** 42

All unmentioned parameters are set to the default value, which is present in the XGBoost library.

3.2 Bias

Bias can decrease the performance of our models. This makes removing the bias significant to create better-performing models. In this paper, one of the goals is to find bias through model-agnostic methods; the results of the model-agnostic methods are used to determine the most influential features. If the difference in importance of the features is significant, the model is biased towards the features with the most influence. This new information about the model gives new opportunities for improvements.

3.3 Class Imbalance

Class imbalance presents considerable challenges in classification settings. It occurs when the distribution of classes is heavily skewed, resulting in a disproportionate representation of one class compared to others. This imbalance significantly impacts the performance and fairness of classification algorithms. The ultimate goal is to mitigate the impact of class imbalance and improve the overall performance of the classification model. Although this is not the primary objective of the research, a reliable model is crucial in building a foundation for interpretability.

For this research, we used a commonly used method, which implemented downsampling of the majority class, to deal with the class imbalance in the diabetes dataset. The SMOTE method was not preferred since it created unrealistic feature values for the synthetic instances because of the non-continuous nature of some of the features, and together with SHAP, this could create previously non-existing bias. In the context of this research, the XGBoost classifier, a powerful ensemble method, is chosen as it effectively handles imbalanced datasets, incorporates cost-sensitive learning, and remains close to the intended research with the initial client.

The majority class (True Negatives) is, with respect to the minority class, downsampled from an approximately 11:1 ratio to a 4:1 ratio to improve the model's accuracy and recall. Different ratios for the minority and majority classes were considered. The downsampling was only performed on the training set so that for validation on the test set, a realistic composition of the classes representative of reality gets tested. Figure 1 shows how closely the ratios perform in ROC plot and AUC scores, a metric used to evaluate the performance of binary classification models. It gauges the ability of a model to discern positive and negative classes by plotting the true positive rate (TPR) against the false positive rate (FPR). A classifier is considered better the closer the value gets to 1, a value of 1 is considered a perfect classifier.

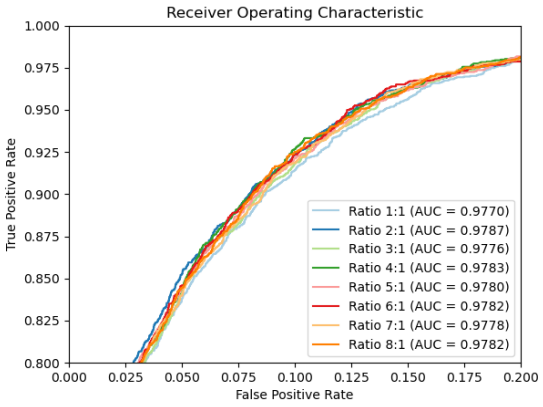


Figure 1: Top Left Corner of Receiver Operating Characteristic (ROC) Plot comparing various training set ratios

Table 3 presents the accuracy and recall for the different ratios. The table shows that the 4:1 ratio has a high accuracy with a high false recall. The 5:1

ratio does improve the accuracy by a small margin. However, the true recall does decrease below 75%.

Table 3: Accuracy and Recall score for different majority/minority ratios of training dataset

Ratio	**1:1**	**2:1**	**3:1**	**4:1**	**5:1**
Accuracy	0.8987	0.9389	0.9567	0.9612	0.9666
Recall					
- True	0.92	0.85	0.80	0.77	0.74
- False	0.90	0.95	0.97	0.98	0.99

Table 4 is an extract from Appendix Table A.2 and displays several typical performance indicators in machine learning for different penalty weights and data ratios. Table 4 particularly zooms in on the chosen training set ratio and shows the changes in performance when the cost-sensitive approach of increased weight on Type II errors is implemented. The weight represents the scale_pos_weight parameter of the XGBoost classifier, a scalar for the penalty on Type II errors.

Table 4: Performance Results on Test Set for Different Type II Error Weight Values using 4:1 Training Set Ratio

Weight	Accuracy	F1	ROC AUC	FPR	TPR
1.0	**0.9626**	**0.882**	**0.9783**	**0.9824**	**0.9824**
1.5	0.9534	0.8656	0.978	0.9673	0.9673
2.0	0.9444	0.8487	0.9778	0.9553	0.9553
2.5	0.934	0.8311	0.9776	0.9413	0.9413
3.0	0.9273	0.821	0.9778	0.9323	0.9323
4.0	0.9133	0.8003	0.978	0.9145	0.9145
5.0	0.9031	0.7863	0.9776	0.9019	0.9019
10.4	0.869	0.7433	0.9763	0.8613	0.8613

The confusion matrix of the model trained after adjusting for class imbalance (Figure B.6) shows that in our test set, most predicted values correspond with their true label from the original dataset. The model has a nearly even amount of Type I and Type II errors, which can be seen in Table 4 in the row with 'weight' 1.

The model trained on the training data sampled from the original dataset (Figure B.5) shows an uneven split between these error types. It is for the model concerning the prediction of diabetes important that there are as few as possible Type II Errors, or at least the ratio between the errors and correctly predicted positive cases is greatly favoured towards the correct predictions. The model trained after adjusting for class imbalance has a better ratio than the original one. The class-imbalanced dataset shows a more even distribution. It is preferred for applying and analysing the model-agnostic methods as the performance is still great, whereas in most other scenarios in Table A.2, some form of performance has to be forfeited to improve the true positive rate.

3.3.1 Potential Effect on Data Bias

The correlation matrix of the dataset, adjusted to class imbalance, is presented in Figure B.4. The data suggests that diabetes is strongly and positively correlated with both HbA1c (0.53) and blood glucose levels (0.52). However, these two features are only moderately correlated with each other (0.27), which indicates that they are related but not too significantly, possibly due to individual variations in glucose metabolism or fluctuations in blood sugar levels.

Age and BMI also appear to be significantly positively correlated with diabetes. Age and BMI also carry the same characteristic cause they are significantly positively correlated. It seems that age is mildly positively correlated with all variables except gender, which is nearly 0. This benefits the analysis because data balanced towards men and women are preferred. In addition, we see that the current smoking history, which shows that someone who is currently smoking, is not correlated to age.

When we compare this heatmap to the correlation heatmap of the original dataset (see Figure B.3), we can observe an increase in correlation across all variables after adjusting the dataset for class imbalance. As the relations between features change with these class imbalance measures, potentially creating some form of bias, it is important to note that this change may impact the interpretability of the methods discussed in the next subsection.

3.4 Model-Agnostic Explainability Methods

In this subsection of the methodology section, we will discuss the model-agnostic methods used in this research to find an answer to the original problem statement. The selected methods are SHAP (SHapley Additive exPlanations), LIME (Local Surrogate) and their background and implementations, and the validator for these methods, LOCO (Leave-One-Covariate-Out), will be elaborated on.

3.4.1 SHAP (SHapley Additive exPlanations)

As mentioned in the terminology section, Shapley's calculations require considering every possible combination of feature coalitions. The computational complexity can grow exponentially as the number of features and instances increases. Since the chosen dataset and research timeframe made

this method too computationally expensive, we implemented a universal adaptation of SHAP called Kernel SHAP, using the SHAP library in Python. While we recognise the usefulness of TreeSHAP, it only works with Tree-based machine-learning algorithms. Therefore, we opted for Kernel SHAP, in line with utilising model-agnostic methods with broad applicability.

3.4.2 KernelSHAP

The kernelSHAP is a variant of the SHAP method that is less computationally expensive and can be used on various machine learning models. In this paper, the method uses a total of 1,000 instances, with a background dataset of another 1,000 instances. This choice is made to have an acceptable running time.

3.4.3 LIME

For instance, x, the LIME library, which provides a suite of tools for understanding and interpreting machine learning model predictions, was used in a Python environment to obtain the local explanation. A LimeTabularExplainer object was instantiated and configured to generate interpretable explanations with the training data and feature names from the training data. The classification mode and class names indicative of a binary classification problem were specified. A prediction function was created using the probability prediction method from the trained XGBClassifier model, enabling the computation of class probabilities. The instance explainer method from the LimeTabularExplainer object was then used to generate an explanation for the specific instance x. As an outcome, the instance explainer provided an overview of the significance of the features influencing the prediction, as seen in Figure 5.

The LIME method can also obtain a global explanation for the model. A method was developed to obtain a global explanation of the XGBoost classifier model. This process involved creating a function to extract weights from the LIME explanation object. This is done for the first 100 instances of the test set. This was only done for 100 instances to decrease run time. The feature weights are stored in a data frame with the feature names as columns. The absolute mean of the weights for each feature was calculated across all instances to measure average importance, providing a global view of feature importance as determined by the LIME explanations. This served as a powerful tool for understanding the overall decision-making process of the model, highlighting the features that, on average, have the most significant influence on the model's predictions. This overall process greatly enhances the interpretability of the machine learning model, contributing to the transparency and trust in its predictions.

3.5 Validation: Leave-One-Covariate-Out (LOCO)

(Lei et al., 2017) explores the LOCO (Leave-One-Covariate-Out) method, a technique used for model interpretation. The LOCO method operates by systematically excluding one covariate at a time from the model's input and observing the impact on the model's output. The significance of each covariate in the model's decision-making process can be determined by how the excluded covariate impacts the model's performance. To obtain the LOCO value of feature j, for instance, i, the formula 10 is used.

$$LOCO_{ij} = \hat{y}(x_i) - \hat{y}(x_{i,-j}) \tag{10}$$

$\hat{y}(x_i)$ is the prediction for the i^{th} instance when the model is trained without this instance. $\hat{y}(x_i, -j)$ is the prediction for the i^{th} instance when the model is trained without this instance and feature j. The median overall LOCO Values for the instances in the test set should be taken per omitted feature to get the global feature importance. Then these values are ranked to see the feature's importance. Since the LOCO method is such a straightforward process, it can be used to validate the results of the two methods used for explainability, SHAP and LIME.

To obtain the LOCO values per feature of the diabetes classifier, the accuracy of the test set can be used to evaluate the importance. First, the model on all features is trained to establish a baseline to compare performance. Then, a temporary copy is made from the test and training sets where feature j is omitted. The model is then trained on this altered training set and evaluated on the modified test set. The accuracy of the test set is then stored in a table for comparison. This process is repeated for all features. Then this table is ranked so that feature importance can be easily compared.

4 Results

This section presents the results obtained from implementing and applying the methods described in the previous sections. We discuss the explainability

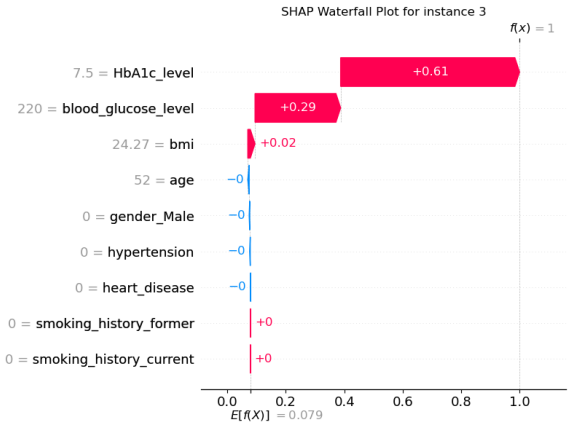
of the models using SHAP and LIME methods and apply the LOCO method to validate their results. We employed the SHAP and LIME methods in the explainability analysis to gain insights into the models' feature importance and behaviour, with LOCO as the validation method.

4.1 Explainability Methods

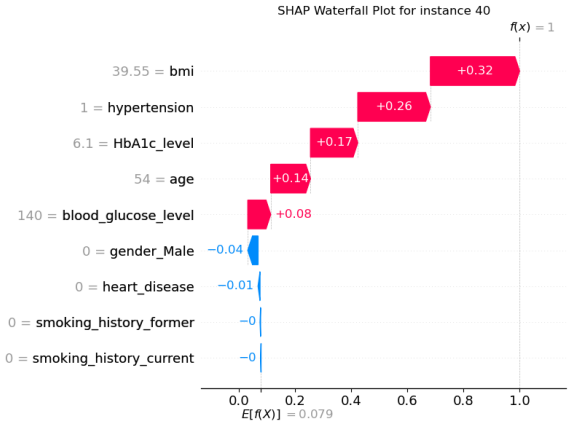
4.1.1 Kernel SHAP

We used the SHAP method to assess the local feature importance for individual instances. For data entries 3 and 40, the SHAP waterfall plots are presented in Figure 2a and Figure 2b, respectively. These plots depict, on the x-axis, the contributions of each feature on the y-axis with their actual value from the instance. They start from $E[f(x)]$, the expected value at the bottom of each plot leading to the two individual final predictions. While both are positive predictions, meaning a diagnosis of diabetes, instance 40, is a misclassification.

To get a comprehensive view of global feature importance using SHAP, a bar plot in Figure 3 illustrates the average feature importance for 1,000 instances, whereas Figure 4 exhibits the SHAP decision plot for the same number of instances. This plot depicts how the model arrived at the final classification, including the contributions of each feature for each instance shown as the change of direction from the start of a feature to the next one or prediction on the y-axis. Similarly, Appendix Figure B.7 also displays both a global and local view of the method results and the feature contributions for the same sample of instances, with the contribution and a relative value of the feature for each instance.



(a) SHAP Waterfall plot, (Local) feature importance for instance 3; Convergence from Expected value to final positive diagnosis prediction.



(b) SHAP Waterfall plot, (Local) feature importance for instance 40; Convergence from Expected value to final positive diagnosis prediction

Figure 2: Combined figure showing SHAP Waterfall plots for instances 3 and 40

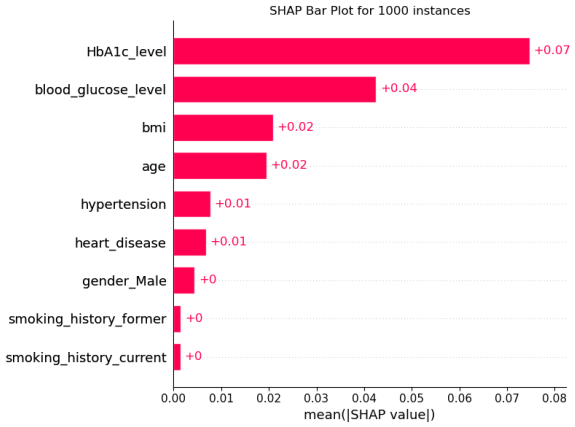


Figure 3: SHAP Bar plot, (global) average feature importance. Mean absolute SHAP value of variables (1,000 instances)

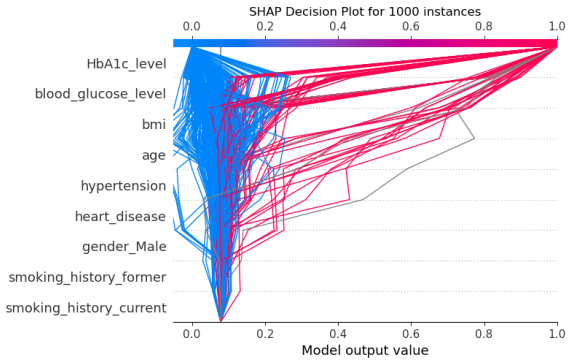


Figure 4: SHAP Decision Plot (local) feature importance. The contribution to the final prediction of variables (1,000 instances)

4.1.2 LIME

The explainer for the local explanation of instance 3 and instance 40 of the test set provided the result shown in Figure 5 and 6. The bars indicate how much that feature contributes to the final classification for each feature. Bars on the right contribute to a positive classification, and bars on the left contribute to a negative classification. For the global explanation, the mean absolute feature weight over the first 1000 instances of the test set per feature can be observed in bar plot 7. The mean absolute weight for a feature can be seen as the feature's importance.

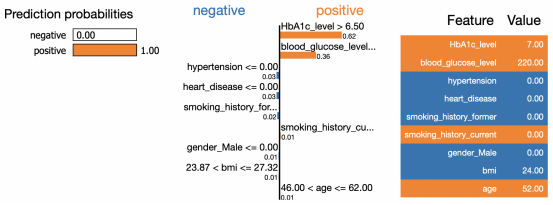


Figure 5: LIME explanation of instance 3 of the test set; Feature value of the instance and the positive/negative weights of these features for the final prediction contributing to the final prediction probabilities.

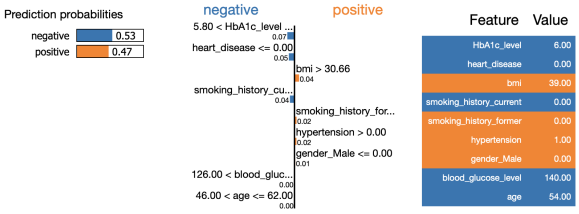


Figure 6: LIME explanation of instance 40 of the test set; Feature value of the instance and the positive/negative weights of these features for the final prediction contributing to the final prediction probabilities.

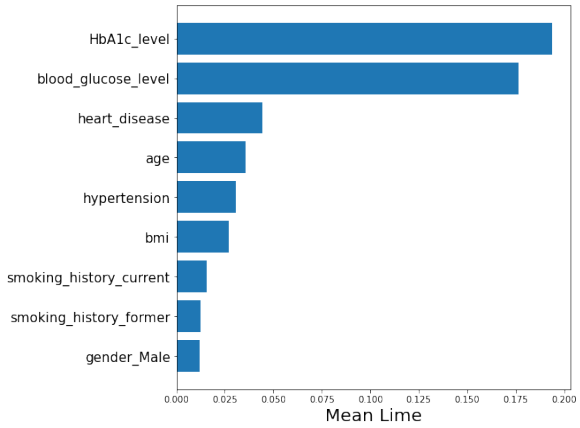


Figure 7: Mean feature importance. Mean absolute LIME value (first 1000 instances of the test set)

4.2 Validation: LOCO

For the LOCO method, the error rate per feature is plotted in Figure 8. The error rate is calculated by subtracting the accuracy of the test set from 1. "none" indicates the error rate for the model with all the features included to show the baseline.

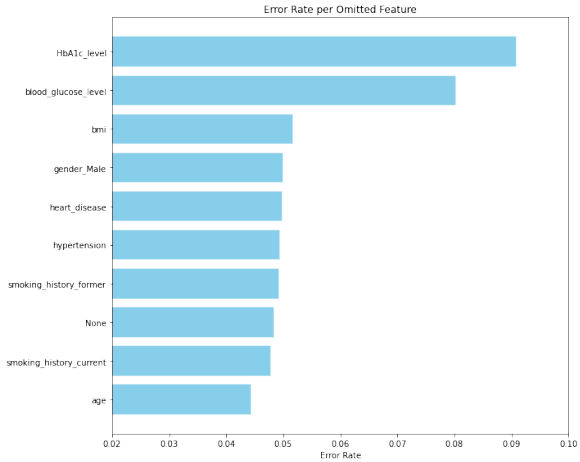


Figure 8: error rate using LOCO method; The error of the prediction when the variable is omitted from the model

4.3 Comparison of Methods

Figure 9 and Table 5 show the comparison of the methods and validator used in this research. Figure 9 is a bar plot that compares the mean absolute values from SHAP and LIME for each feature with the error rate obtained from applying LOCO to the model. The values on the x-axis are normalised values of each method, with the HbA1c level as the base for the normalisation, which is why this has a value of 1.

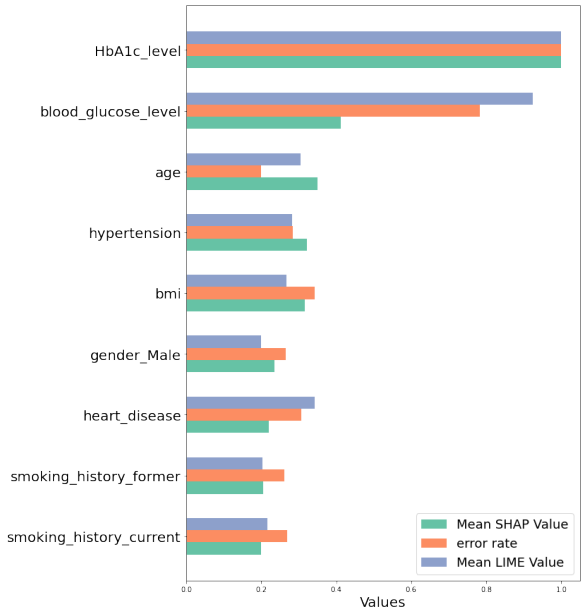


Figure 9: Normalised comparison of methods.

Table 5 depicts the correlation coefficients between the mean absolute values from the methods, the error rate from the validator and each feature correlation with the binary target variable 'Diabetes'. In this table, we observe a significant correlation between all considered, as a general rule of thumb for a strong correlation is around the value of 0.8. The values from Kernel SHAP and LIME, as well as LIME and LOCO, even seem almost perfectly correlated.

Table 5: Correlation Coefficient Table Comparing Methods, SHAP, LIME, LOCO and Feature Correlations with the Target Variable

Correlation Coefficients	SHAP	LIME	LOCO	Feature Correlation
SHAP	1.000			
LIME	0.956	1.000		
LOCO	0.903	0.961	1.000	
Feature Correlation	0.816	0.791	0.873	1.000

5 Conclusion

The findings of our extensive investigation into two different techniques for model interpretability - SHAP and LIME, together with our validation technique LOCO have been presented in Figure 9 and Table 5. Our research revealed that these techniques share a high degree of correlation, with a significant agreement in the interpretations provided by each method. This consistent pattern across independently developed methodologies confirms the individual validity of each technique and significantly enhances the robustness of their collective outcomes.

While ideally, the explainability methods would agree to be able to make a stronger claim at understanding the decision-making of a model, as instance 3 in Figures 2a and 5 provide fairly similar local feature importance insights. For this data entry, this could also be explained by the fairly extreme values of the two most important features of HbA1c and blood glucose level for this instance. However, SHAP and LIME can differ at the local level; this can, for example, be observed in Figures 2b and 6 where the local explanations for data instance 40 are displayed; these two methods even suggest a different classification. This can be explained by their different approaches, where Kernel SHAP approximates the true Shapley values, the marginal contribution of the features, and LIME creates a local surrogate around the instance of interest to approximate the predictions of the original model, it could even be due to the margin of error that if the true Shapley values were computed, the explanations could be closer to one another. Another reason could be that Kernel SHAP requires dealing with missing values when features are omitted from a coalition, which in this case, is done by filling in the feature's average value from the training set. LIME does not have to deal with such complexities as it purely focuses on the local neighbourhood. In the appendix in figure B.8 till B.15 the explanations using both methods on several other instances can be seen. SHAP and LIME show agreement on the feature importance of features who are allocated the highest weights (blood glucose level and Hba1c level). But they mostly differ in their ranking of the features with lower importance.

The model-agnostic methods have illustrated that the model has two main features that it uses for its decisions. These are blood glucose level and Hba1c level, which is not unsurprising considering both concern blood sugar/glucose levels (see Table 2, for reference), commonly related to diabetes or microvascular complications. Since these two features influence the model the most, we can say that the model is biased towards these features. However, this does not suggest that this information leads to the possibility of model improvement,

it merely acknowledges the existence. Regardless, obtaining this kind of information from supposed "black box" models can be greatly beneficial when the information can be handled by specialists who can make educated conclusions and suggest altercations from the insights provided.

Our study emphasises the importance and value of these interpretability methods in providing clear and understandable insights into how complex machine learning models make predictions. The consistency in the patterns of feature importance across all the instances studied suggests that SHAP and LIME can be reliably used to guide model understanding and decision-making processes. By contributing to a more transparent and trustworthy AI landscape, these techniques enhance confidence in their utility and positively impact the integration of AI in various domains.

6 Discussion

When the model produces unexpected or incorrect predictions on unseen data, explanations can help diagnose the causes behind these errors. Examining the feature contributions in the explanations, one can identify which features or interactions might lead to erroneous predictions and take appropriate corrective actions. Our study has made a contribution by highlighting the effectiveness of SHAP and LIME as reliable interpretability methods. However, our study is not without limitations. Despite these limitations, our findings provide a foundation for further research.

Looking ahead, we believe our findings will inspire further research in this area, thereby contributing to developing more transparent and trustworthy AI systems and the possibility to comply with strict but necessary regulations to increase safety.

6.1 Limitations

In the range of this study, certain limitations were encountered that could have influenced the results and their interpretation. One of the main challenges was time constraints that limited the comprehensive application of the LIME and SHAP methods. This could have impacted our capacity to fully explore these methods' potential in revealing biases and the decision-making process of the machine learning model.

Although this study provides valuable insights into the decision-making process, it's vital to keep in mind that using a different model may result in different results as the explanations may depend on the chosen model's underlying mechanisms. In this case, the explainability methods are applied to a specific model, the XGBoost classifier, and their effectiveness can vary when these methods are applied to the same dataset with a different model.

Finally, it is vital to emphasize that the biases identified in this study are dataset-specific. This implies that the same approach may yield less clear results when applied to other models or data. The effectiveness of these methods could vary depending on the data they are applied to; for example, underlying biases could differ and greatly affect the explanations.

6.2 Further Research

While this study has made significant advancements in uncovering similarities and differences between major model-agnostic methods, several areas warrant further exploration.

Considering the findings of this paper, one field which can be researched is the improvement of models after conducting the analysis presented in this paper. These variables can be neglected during measurements and simplify the model by looking at which variables do not contribute to the predictions. In addition, other measures can be put in place to conduct future analyses on their importance for the prediction power of the model.

While the current methods for explanation validation have provided some insights, a more thorough sensitivity analysis could enhance trust in the explanations generated by LIME and SHAP. This analysis would investigate how alterations in feature values impact both the model's output and the explanations provided by LIME and SHAP. If these changes align with expectations based on the modifications made to the input features, this will bolster confidence in the explanations.

Furthermore, the consistency of these explanations in response to varying feature values could also serve as an indicator of their reliability. This approach would not only provide a more rigorous validation method for the explanations but also offer valuable insights for future research. It could help understand the stability of these explanation methods under different conditions and their adaptability to changes in the input data.

Another aspect which can be thoroughly examined is the application of the relations presented

in this paper concerning the European AI Act. While the current implications of this act are not finalised, combining the values and relationships of the model-agnostic methods presented can be of key importance for a model to be approved for application.

Moreover, applying the current methods to this dataset only shows how black-box models can be analysed externally. The XGBoost classification algorithm has internal workings that can be examined to improve interpretability. However, the idea of this paper is that the findings can be applied to all black-box models, such as deep neural networks, which do not possess these characteristics.

The current findings are only verified by the current dataset and model. Therefore, the conclusions are not guaranteed to be valid across all model types, and a more in-depth analysis of the behaviour of these models is necessary. The initial objective was to apply this analysis to a time-series dataset to understand crowd flow prediction models better. However, we could not apply the methods to this data due to time limitations and the complicated collaboration with the external data provider. The application could provide significantly different results and findings.

7 Acknowledgements

We want to express our sincere gratitude to all individuals whose contributions made this research project possible. Their support, guidance, and resources have been invaluable throughout this endeavour.

First and foremost, we would like to extend our heartfelt appreciation to our supervisor, Dr S. Yu. Their expertise and valuable insights have played a key role in shaping the direction of this paper. We are truly grateful for their guidance, which has greatly contributed to the successful completion of this research.

In addition, we would like to extend our gratitude to our project coordinator, Dr S. Bhulai. Their attention to detail and dedication during tempestuous times ensured the smooth execution of this research project. Their guidance and coordination were invaluable in navigating various stages of the study efficiently.

Last but not least, we are truly grateful to the Vrije Universiteit Amsterdam (VU University) for providing the necessary resources and facilities to conduct this research.

References

Bhatt, U., Xiang, A., Sharma, S., Weller, A., Taly, A., Jia, Y., Ghosh, J., Puri, R., Moura, J. M., and Eckersley, P. (2020). Explainable machine learning in deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 648–657. ACM.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM.

Dalal, S., Onyema, E. M., and Malik, A. (2022). Hybrid xgboost model with hyperparameter tuning for prediction of liver disease with better accuracy. *World Journal of Gastroenterology*, 28(46):6551–6563.

Elkan, C. (2001). The foundations of cost-sensitive learning. In *International joint conference on artificial intelligence*, volume 17, pages 973–978. Lawrence Erlbaum Associates Ltd.

Gall, J.-R. L., Loirat, P., Alperovitch, A., Glaser, P., Granthil, C., Mathieu, D., Mercier, P., Thomas, R., and Villers, D. (1984). A simplified acute physiology score for icu patients. *Critical Care Medicine*, 12(11):975–977.

Gu, J. and Oelke, D. (2019). Understanding bias in machine learning.

Hellström, T., Dignum, V., and Bensch, S. (2020). Bias in machine learning – what is it good for?

Kaggle (2023). Diabetes prediction dataset. https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset. Retrieved 29 June 2023.

Kumar, V., Lalotra, G., Sasikala, P., Rajput, D., Kaluri, R., and Lakshmanna, K. (2022). Addressing binary classification over class imbalanced clinical datasets using computationally intelligent techniques. *Healthcare*, 10(7):1293.

Labarère, J., Bertrand, R., and Fine, M. J. (2014). How to derive and validate clinical prediction models for use in intensive care medicine. *Intensive Care Medicine*, 40(4):513–527.

Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J., and Wasserman, L. (2017). Distribution-free predictive inference for regression.

Li, Y., Xu, Y., Ma, Z., Ye, Y., Gao, L., and Sun, Y. (2022). An xgboost-based model for assessment of aortic stiffness from wrist photoplethysmogram. *Computer Methods and Programs in Biomedicine*, 226:107128.

Linardatos, P., Papastefanopoulos, V., and Kotsiantis, S. (2020). Explainable ai: A review of machine learning interpretability methods. *Entropy (Basel, Switzerland)*, 23(1):18.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Molnar, C. (2022). *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*. Leanpub, 2nd edition.

Reps, J. M., Williams, R. D., Schuemie, M. J., Ryan, P. B., and Rijnbeek, P. R. (2022). Learning patient-level prediction models across multiple healthcare databases: evaluation of ensembles for increasing model transportability. *BMC Medical Informatics and Decision Making*, 22:126.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Roth, A. E. (1988). *The Shapley value: essays in honor of Lloyd S. Shapley*. Cambridge University Press.

Saarela, M. and Jauhiainen, S. (2021). Comparison of feature importance measures as explanations for classification models. *SN Appl. Sci.*, 3(272):1–11.

Sun, Y., Kamel, M. S., Wong, A. K., and Wang, Y. (2007). Cost-sensitive boosting for classification of imbalanced data. *Pattern recognition*, 40(12):3358–3378.

Turney, P. D. (2002). Types of cost in inductive concept learning. *arXiv preprint cs/0212034*.

van den Heuvel, L., Knippenberg, M., Post, B., Meinders, M. J., Bloem, B. R., and Stiggelbout, A. M. (2022). Perspectives of people living with parkinson's disease on personalized prediction models. *Health Expectations : An International Journal of Public Participation in Health Care and Health Policy*, 25(4):1580–1590.

Vincent, J.-L. and Moreno, R. (2010). Clinical review: Scoring systems in the critically ill. *Critical Care*, 14(2):207.

A Appendix: Tables

Table A.1: Example of the first five instances of the dataset

gender	age	hypertension	heart_disease	smoking_history	bmi	HbA1c_level	blood_glucose_level	diabetes
Female	80.0	False	True	never	25.19	6.6	140	False
Female	54.0	False	False	No Info	27.32	6.6	80	False
Male	28.0	False	False	never	27.32	5.7	158	False
Female	36.0	False	False	current	23.45	5.0	155	False
Male	76.0	True	True	current	20.14	4.8	155	False

Table A.2: Performance of Class Imbalance and Cost Imbalance Measures

Ratio	Error Weight	Accuracy	F1	ROC AUC	False Recall	True Recall
1:1	1.0	0.8996	0.7811	0.977	0.8979	0.8979
	1.5	0.8813	0.7574	0.9762	0.8762	0.8762
	2.0	0.8691	0.7437	0.9767	0.8612	0.8612
	2.5	0.856	0.7287	0.9764	0.8462	0.8462
	3.0	0.8491	0.7214	0.9763	0.8378	0.8378
	4.0	0.8354	0.7069	0.9757	0.8221	0.8221
	5.0	0.8329	0.7041	0.9761	0.8195	0.8195
	10.4	0.8131	0.6848	0.9765	0.7967	0.7967
2:1	1.0	0.941	0.8443	0.9787	0.9494	0.9494
	1.5	0.9227	0.8138	0.9779	0.9265	0.9265
	2.0	0.9094	0.7956	0.9787	0.909	0.909
	2.5	0.8981	0.7796	0.9784	0.8957	0.8957
	3.0	0.8903	0.7699	0.978	0.886	0.886
	4.0	0.8796	0.7569	0.9775	0.873	0.873
	5.0	0.8645	0.7384	0.9774	0.8558	0.8558
	10.4	0.8373	0.7087	0.9772	0.8245	0.8245
3:1	1.0	0.9534	0.864	0.9776	0.9687	0.9687
	1.5	0.9406	0.8415	0.9778	0.9507	0.9507
	2.0	0.9298	0.8241	0.9774	0.9361	0.9361
	2.5	0.9209	0.8112	0.9779	0.924	0.924
	3.0	0.9111	0.7964	0.9774	0.9124	0.9124
	4.0	0.8997	0.7813	0.977	0.8979	0.8979
	5.0	0.8866	0.7639	0.9769	0.8827	0.8827
	10.4	0.8535	0.7259	0.9767	0.8432	0.8432
4:1	**1.0**	**0.9626**	**0.882**	**0.9783**	**0.9824**	**0.9824**
	1.5	0.9534	0.8656	0.978	0.9673	0.9673
	2.0	0.9444	0.8487	0.9778	0.9553	0.9553
	2.5	0.934	0.8311	0.9776	0.9413	0.9413
	3.0	0.9273	0.821	0.9778	0.9323	0.9323
	4.0	0.9133	0.8003	0.978	0.9145	0.9145
	5.0	0.9031	0.7863	0.9776	0.9019	0.9019
	10.4	0.869	0.7433	0.9763	0.8613	0.8613
5:1	1.0	0.9657	0.8885	0.978	0.9872	0.9872
	1.5	0.9602	0.8777	0.9782	0.9781	0.9781
	2.0	0.9522	0.8624	0.9782	0.9662	0.9662
	2.5	0.9447	0.8492	0.9779	0.9556	0.9556
	3.0	0.9382	0.8385	0.9777	0.9465	0.9465
	4.0	0.9248	0.8163	0.9771	0.9297	0.9297
	5.0	0.9152	0.8018	0.9771	0.9177	0.9177
	10.4	0.8825	0.7596	0.9768	0.8771	0.8771
6:1	1.0	0.9682	0.8934	0.9782	0.9916	0.9916
	1.5	0.9625	0.8802	0.9775	0.9835	0.9835
	2.0	0.9574	0.871	0.9777	0.9749	0.9749
	2.5	0.9521	0.8617	0.9779	0.9665	0.9665
	3.0	0.9465	0.8521	0.9781	0.9584	0.9584
	4.0	0.9341	0.8304	0.9775	0.9423	0.9423
	5.0	0.9258	0.8186	0.9775	0.9305	0.9305
	10.4	0.889	0.7674	0.9766	0.8851	0.8851
7:1	1.0	0.9691	0.8941	0.9778	0.9941	0.9941
	1.5	0.967	0.8914	0.9777	0.989	0.989
	2.0	0.9624	0.8814	0.9778	0.9821	0.9821
	2.5	0.9569	0.871	0.9779	0.9736	0.9736
	3.0	0.9518	0.8615	0.9779	0.9659	0.9659
	4.0	0.9413	0.8433	0.9776	0.9512	0.9512
	5.0	0.9336	0.8306	0.9775	0.9407	0.9407
	10.4	0.8996	0.7805	0.977	0.8984	0.8984
8:1	1.0	0.9699	0.8968	0.9782	0.9947	0.9947
	1.5	0.9672	0.8905	0.9779	0.9907	0.9907
	2.0	0.9639	0.8846	0.9776	0.9843	0.9843
	2.5	0.9606	0.8783	0.9782	0.9789	0.9789
	3.0	0.9565	0.8702	0.9778	0.9727	0.9727
	4.0	0.9474	0.8539	0.9775	0.9596	0.9596
	5.0	0.939	0.8391	0.9774	0.9481	0.9481
	10.4	0.9044	0.7864	0.9766	0.9046	0.9046

B Appendix: Figures

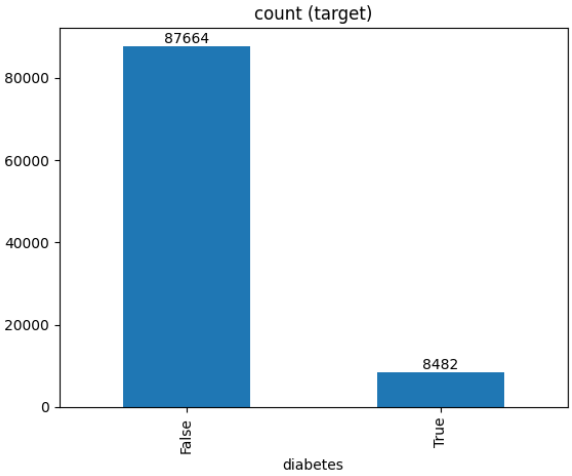


Figure B.1: distribution of diabetes (target variable) in original dataset

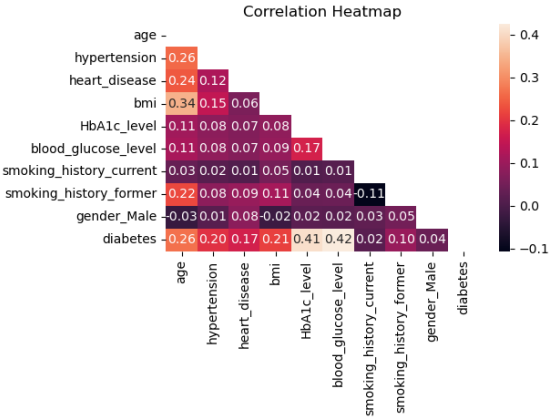


Figure B.3: Correlation Heatmap of the original dataset

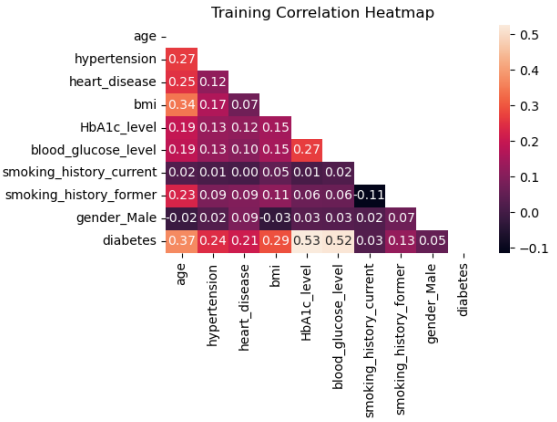


Figure B.4: Correlation Heatmap of the training dataset (4:1 ratio class imbalance)

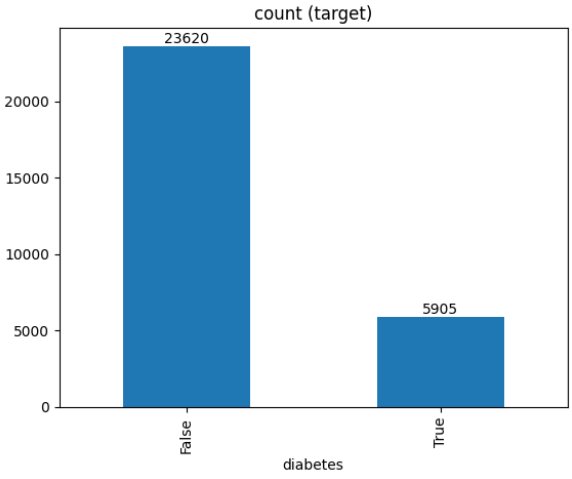


Figure B.2: distribution of diabetes (target variable) in the undersampled training dataset (ratio 4:1)

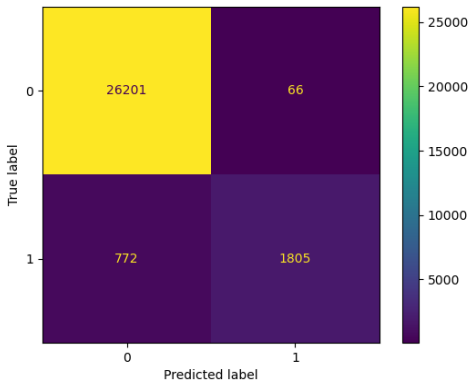


Figure B.5: Confusion matrix of the model's classifications on the test set before class imbalance measures

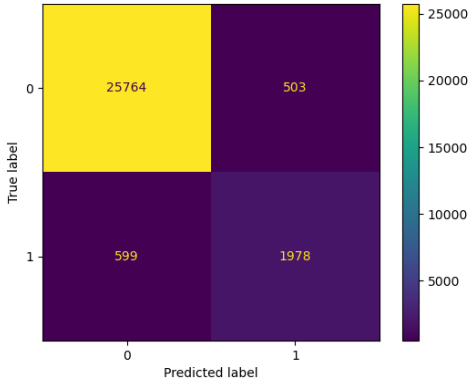


Figure B.6: Confusion matrix of the model's classifications on the test set after class imbalance measures

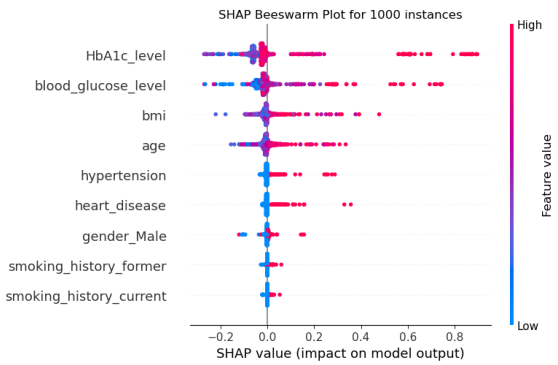


Figure B.7: SHAP Beeswarm plot (local) feature importance for 1,000 instances

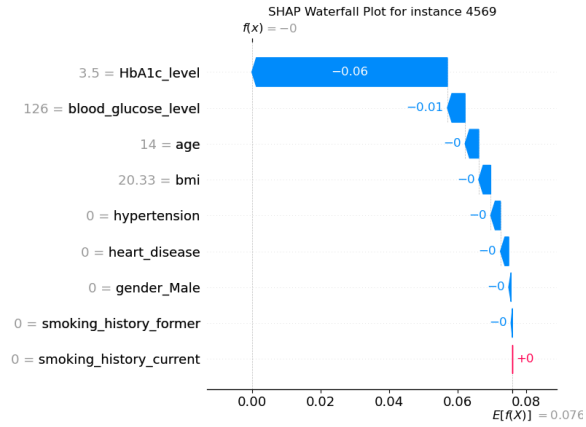


Figure B.8: SHAP Waterfall plot, (Local) feature importance for instance 4569.

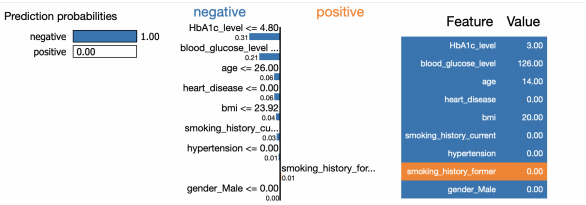


Figure B.9: LIME explanation of instance 4569 of the test set; Feature value of the instance and the positive/negative weights of these features for the final prediction contributing to the final prediction probabilities.

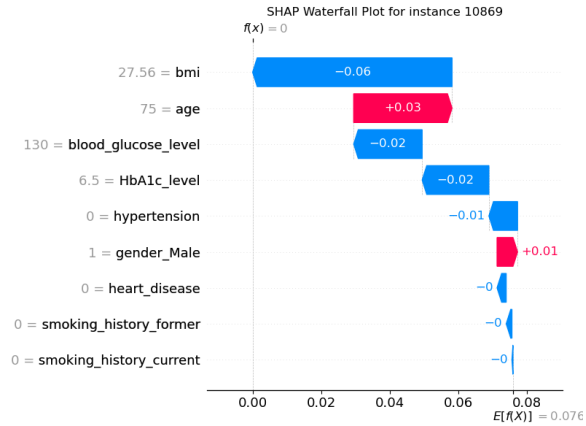


Figure B.10: SHAP Waterfall plot, (Local) feature importance for instance 10869.

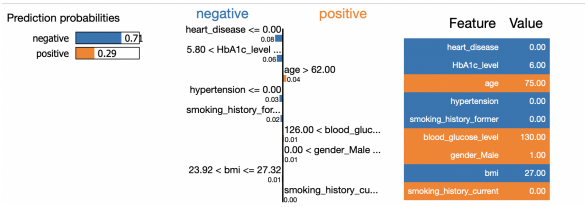


Figure B.11: LIME explanation of instance 10869 of the test set; Feature value of the instance and the positive/negative weights of these features for the final prediction contributing to the final prediction probabilities.

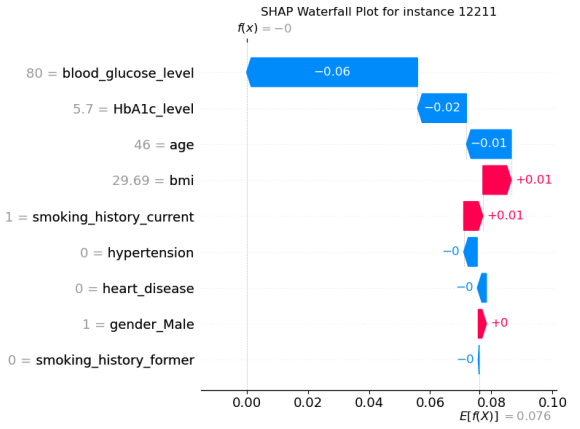


Figure B.12: SHAP Waterfall plot, (Local) feature importance for instance 12211.

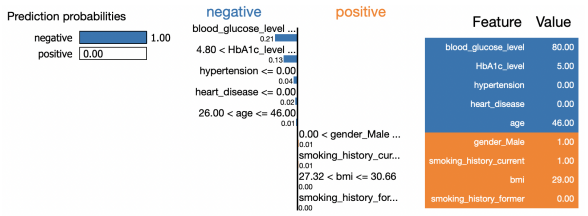


Figure B.13: LIME explanation of instance 12211 of the test set; Feature value of the instance and the positive/negative weights of these features for the final prediction contributing to the final prediction probabilities.

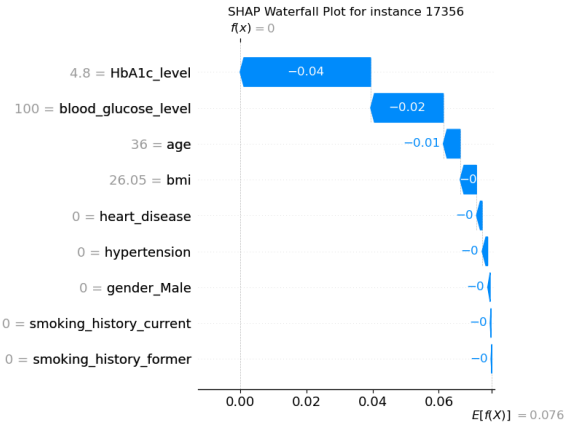


Figure B.14: SHAP Waterfall plot, (Local) feature importance for instance 17356.

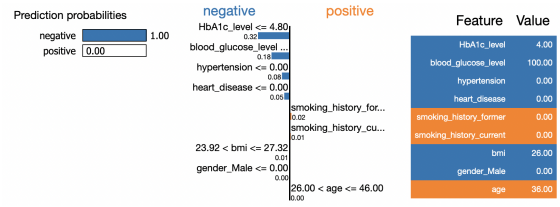


Figure B.15: LIME explanation of instance 17356 of the test set; Feature value of the instance and the positive/negative weights of these features for the final prediction contributing to the final prediction probabilities.