Explainable Machine Learning for Diabetes Prediction

by

Sadam Hussain

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Applied Science

in the Department of Electrical and Computer Engineering

We acknowledge and respect the Lək̓ʷəŋən (Songhees and Xʷsepsəm/Esquimalt) Peoples on whose territory the university stands, and the Lək̓ʷəŋən and W̱SÁNEĆ Peoples whose historical relationships with the land continue to this day.

Explainable Machine Learning for Diabetes Prediction

by

Sadam Hussain

Supervisory Committee

Dr. T. Aaron Gulliver, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Mihai Sima, Departmental Member
(Department of Electrical and Computer Engineering)

ABSTRACT

Diabetes is a growing global health concern, contributing to significant morbidity, mortality, and long-term economic burden. Machine Learning (ML) methods are increasingly applied to diabetes prediction, however, selecting appropriate classifiers and understanding the key features driving model decisions remain essential for reliable and clinically acceptable performance. This is particularly important in healthcare settings where clinicians may have limited familiarity with ML techniques and where transparency and trust in predictive outputs are critical. This study evaluates eight ML classifiers, Logistic Regression (LR), Random Forest (RF), Gradient Boosting (GB), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), AdaBoost (AB), Decision Tree (DT) and Neural Network (NN) using a dataset of 100,000 patient records for diabetes prediction. Models are evaluated using various configurations which includes baseline training and hyperparameter optimization using RandomizedSearchCV. The global and local interpretability is examined using SHapley Additive exPlanations (SHAP), Local Interpretable Model-Agnostic Explanations (LIME) and Explain Like I'm 5 (ELI5) to identify the most influential features contributing to predictions. These findings show that ensemble based models achieve strongest predictive performance with RF and GB outperforming other evaluated classifiers. Interpretability analyses consistently highlight that Hemoglobin A1c (HbA1c), blood glucose, Body Mass Index (BMI), and age are the dominant predictive features. A final evaluation using a reduced feature set derived with the help of Explainable AI (XAI) demonstrates that strong predictive accuracy can be maintained while improving model simplicity and interpretability. This work underscores the importance of combining ML performance with transparent feature explanations in order to support trustworthy and clinically meaningful decision support systems for diabetes prediction.

Contents

List of Tables

List of Figures

Abbreviations

AI	Artificial Intelligence
IDF	International Diabetes Federation
ML	Machine Learning
DL	Deep Learning
LR	Logistic Regression
RF	Random Forest
GB	Gradient Boosting
AB	AdaBoost
DT	Decision Tree
SVM	Support Vector Machine
KNN	K-Nearest Neighbors
NN	Neural Network
SHAP	SHapley Additive exPlanations
LIME	Local Interpretable Model-agnostic Explanations
ELI5	Explain Like I'm 5
HbA1c	Hemoglobin A1c
BMI	Body Mass Index
PIMA	Pima Indian Diabetes Dataset
NHANES	National Health and Nutrition Examination Survey

ACKNOWLEDGEMENTS

With sincere thanks to my thesis advisor, my mentor, and all who have inspired and supported me.

DEDICATION

To my parents, family, friends, teachers, and especially my mentor.

Chapter 1

Introduction

Diabetes mellitus is a chronic metabolic disorder characterized by persistent hyperglycemia resulting from impaired insulin secretion, insulin action or both. It is among the most widespread non-communicable or contagious diseases globally. As of 2024, an estimated 589 million adults aged 20-79 worldwide are living with diabetes, representing roughly one in nine adults (1). According to the World Health Organization (WHO), diabetes remains a leading underlying cause of serious complications such as cardiovascular disease, kidney failure, blindness, and lower-limb amputation (2). Given its high prevalence, severe complications and rising incidences, early prediction and prevention of diabetes including the identification of high risk individuals have become critical public health priority around the globe.

Diabetes develops gradually, transitioning through the early metabolic abnormalities like impaired fasting glucose or prediabetes before progressing to the clinical diagnosis (3). These intermediate stages provide a crucial path towards early detection. Accurately predicting models allow healthcare systems to actively identify individuals at risk and deliver timely lifestyle interventions, thereby preventing progress and reduce long term complications (4). Early prediction also reduces healthcare expenditures as diabetes places substantial economic burden on the health systems worldwide, which is estimated at over US $1,015 billion in 2024 with costs expected to rise further according to 2025 International Diabetes Federation (IDF) Diabetes Atlas (1).

Traditional statistical models such as Logistic Regression (LR) have long been used for diabetes risk assessment due to their simplicity and interpretability (5). However, as the complexity and volume of the clinical data have increased LR has shown limitations in

capturing the non-linear patterns and high dimensional interactions among risk factors (6). This has driven growing interest in Machine Learning (ML) approaches which can model intricate relationships among multiple clinical and lifestyle variables. Key predictors identified across recent studies include fasting plasma glucose, Hemoglobin A1c (HbA1c), Body Mass Index (BMI), age, triglycerides, blood pressure, lipid profiles, family history, smoking and drinking status, and physical activity (7; 8). ML methods including Random Forests (RFs), Gradient Boosting Machines (GBMs), Support Vector Machines (SVMs) and Neural Networks (NNs) have demonstrated superior performance compared to the traditional statistical models in diabetes prediction tasks (9; 10).

Despite their high predictive accuracy many ML and Deep Learning (DL) models operate as black box systems, making it hard to interpret how individual predictions are made. This lack of transparency limits trust and clinical deployment where interpretability and justification are essential for safe adoption (11). Explainable AI (XAI) addresses this challenge by providing methods to interpret model behavior. Techniques such as SHapley Additive exPlanations (SHAP) (12) and Local Interpretable Model-agnostic Explanations (LIME) (13) offer insights into both global and local feature contributions enabling clinicians to understand why a model classifies a patient as high or low risk.

1.1 Related Work

Despite significant advancements in computational methods, diabetes prediction remain a critical global health challenge because of its rising prevalence, complex multifactorial etiology and substantial economic burden. Type 2 diabetes, which accounts for more than 90% of cases, develop gradually through interactions between metabolic, behavioral, environmental and genetic risk factors. Early detection is necessary for preventing the complications, improving clinical outcomes and reducing healthcare costs (14). In most recent years, ML approaches have emerged as powerful tools for identifying at risk individuals through patterns in clinical, biochemical, and demographic data (15). This chapter synthesizes current literature on diabetes prediction using ML, discusses the role of explainability techniques in clinical deployment, highlights limitations of existing studies and establishes the foundation for the contributions of this thesis.

Traditional statistical models such as LR, Cox proportional hazards models and generalized linear models have been widely used for diabetes risk estimation (16). These models are

valued for their interpretability but they often struggle to capture non-linear interactions and higher order relationships within the clinical datasets. For instance, fasting glucose, BMI, age, blood pressure and lipid measures exhibit complex interdependencies that simpler statistical models may fail to model effectively (6). ML methods like RF, GB and SVM have demonstrated improved predictive performance by learning non-linear mappings and interaction effects from data (17). As a result, ML based diabetes predictions have received increasing attention in both research and applied clinical decision support setting.

Feature engineering plays a significant role in improving ML based diabetes prediction. Commonly used features include fasting blood glucose, HbA1c, BMI, age, cholesterol profiles, physical activity levels, family history, and lifestyle factors (7). Many studies have demonstrated that glycemic markers such as HbA1c and fasting glucose are often the strongest individual predictors, followed by BMI, age, lipids, and other metabolic measures (18). However, interactions among features are also important, and ML models may outperform traditional clinical scoring systems due to their ability to learn such interactions automatically (19). Nevertheless, inconsistencies in feature selection, lack of systematic evaluation across different algorithms and limited attention to reduced feature modelling remain persistent challenges.

Explainability has become a central theme in ML for healthcare due to the need for transparency, clinician trust, and regulatory compliance. Modern XAI tools like SHAP, LIME, ELI5, and permutation importance offer global and local interpretability which enables clinicians to understand which features most influence predictions (20). SHAP technique which is based on the cooperative game theory provides consistent and individualized explanations, making it appropriate for the clinical decision-making (12). In addition, LIME approximates the decision boundary locally by fitting the simple surrogate models while the ELI5 offers intuitive feature attribution visualizations. Despite their contributions, these XAI methods have not been widely integrated into the diabetes prediction pipelines and that many published studies only report accuracy or AUC without addressing the transparency of the model (21). This limits real world adoption as clinicians are hesitating in order to rely on the black box models for critical clinical decisions like diagnosing prediabetes or initiating further interventions.

Several studies highlight the importance of combining ML predictions with explainability to support personalized risk predictions. For example, SHAP based interpretation of the tree ensembles has revealed how a combination of glycemic markers (e.g. glucose) and lifestyle/metabolic indicators (e.g. BMI) drives individual risk profiles (22). Such insights

can guide targeted interventions, making the XAI a key enabler of actionable clinical decision support. However, integration of XAI remains inconsistent across current research, and a very few studies provide a comprehensive global and local explanations alongside rigorous model benchmarking.

In addition to methodological challenges, practical and clinical barriers impede the deployment of ML systems for diabetes prediction. Healthcare data often contains missing values, outliers, noise and heterogeneous formats which requires robust preprocessing pipelines and careful data cleaning before modeling (23). Many studies either ignore missing data or apply simplistic imputation methods without thoroughly evaluating the impact on model performance. In addition, models trained on old or controlled data may not work as well in real hospitals because patients, medical practices, and data patterns change over time. Prospective validation remains rare therefore, even strong retrospective performance may not guarantee real-time effectiveness in clinical workflows (24).

Due to the current limitations, there is a need for more reliable and interpretable ML frameworks for diabetes prediction, in particular using large, diverse datasets that reflect real world populations. This thesis aims to address these challenges through systematic model comparison, rigorous feature selection and integrated explainability as detailed in the section below.

1.2 Existing Gaps In The Literature and Thesis Contributions

Although ML has shown strong potential for diabetes prediction, current literature lacks several critical gaps that limit its robustness, scalability, interpretability and clinical applicability. These limitations span data quality, study design, methodology, explainability and deployment feasibility. This section provides major gaps and presents contributions of this thesis toward addressing these issues.

A prominent limitation in existing research is the heavy reliance on small or homogeneous datasets. For example, many studies continue to use the Pima Indian Diabetes Dataset (PIMA) which consists of only 768 records, all from a single ethnic population, which limits demographic and clinical diversity (25). These limited and small datasets often fail to capture the heterogeneity of modern populations, thereby hindering the development of

models that generalize beyond narrow settings. Even with larger public datasets such as NHANES (National Health and Nutrition Examination Survey), a few ML based diabetes prediction studies remain limited in scope or lack robust external validation, which may lead to optimistic performance estimates which may not be generalizable across more diverse clinical population (26). To overcome this gap, this work uses a large and contemporary dataset (100,000 individuals) from a publicly available source, offering a broader representation of demographic, metabolic and lifestyle variations. This enhances the potential for building the more generalizable and clinically relevant diabetes risk prediction models.

The second major gap involves lack of systematic multiple model comparisons under standardized conditions. Many studies evaluate only a few ML algorithms, making it difficult to conclude whether a model performs well intrinsically or only outperforms a weak baseline. Comparative studies often vary widely in data splits, evaluation metrics, handling of missing values and feature engineering strategies (27). These inconsistencies hinder reproducibility and prevent meaningful benchmarking across literature. The methodological diversity also makes it challenging for clinicians or policymakers to identify reliable modeling approaches. Therefore, this work performs a comprehensive evaluation of eight ML algorithms under uniform preprocessing and evaluation pipelines.

Interpretability represents another gap, particularly in studies involving complex models. Most of the published works emphasize predictive performance metrics such as accuracy or AUC, however, they offer little to no explanation for how models arrive at their decision. Clinicians require transparent models that justify their outputs to ensure trust, safety and accountability (21). While some studies employ SHAP or LIME, they typically apply them superficially or restrict analyses to a single model. Few studies integrate explainability across multiple algorithms or compare global and local explanations comprehensively. In addition, limited attention is given to the selected or reduced feature modeling, despite their potential to enhance interpretability and reduce data collection burden. This work addresses this gap through an integrated XAI framework applying SHAP, LIME and ELI5 to all the eight ML models. It further identifies the most influential features and evaluates reduced feature models, demonstrating that high predictive accuracy can be achieved with as few as two or four features.

Feature selection remains a relatively under explored aspect in ML based diabetes prediction. Although traditional predictors such as HbA1c, BMI, age, blood pressure and lipid profiles consistently emerge among top predictors in many studies, systematic evaluation of individual feature contributions or interactions is often lacking. In some works that do

conduct feature selection, glucose (or fasting plasma glucose), HbA1c, BMI and age still rank among the most influential variables, indicating their central importance for predictive models (28; 29). Nevertheless, many studies continue to adopt predefined feature sets or include large numbers of variables without rigorous feature importance analysis which may lead to overfitting or reduced interpretability. This thesis conducts systematic feature importance analysis using SHAP, permutation importance, and model specific measures, enabling deeper insights into how individual predictors influence outcomes.

This work addresses the limitations of the existing literature by integrating large scale data, multiple model evaluation, systematic feature analysis, reduced feature modeling and a unified explainability framework. Together, these contributions foster the development of transparent, generalizable and clinically meaningful ML models for diabetes prediction.

In this study, eight ML models; LR, RF, GB, Adaboost (AB), Decision Tree (DT), SVM, K-Nearest Neighbour (KNN) and NN are trained and evaluated using a real world diabetes dataset containing 100,000 patient records. Model performance is first assessed using baseline hyperparameters. Subsequently, hyperparameter tuning is performed using RandomizedSearchCV to optimize model performance. To interpret model decisions and identify key predictors, three XAI techniques SHAP, LIME and ELI5 are applied. These methods are used to determine the most contributing features with models retrained using the top four and top two features identified by XAI. Finally, the models are retrained using the selected subsets of features and their predictive performance is evaluated to assess how reducing the feature set impacts the performance.

The remainder of this thesis is organized as follows. Chapter 2 reviews relevant literature on diabetes prediction and XAI. Chapter 3 describes the details of dataset, preprocessing, methodology and experimental design. Chapter 4 presents classification, hyperparameter tuned optimization, explainability and selected features result.. Chapter 5 concludes the study and provides directions for future research.

Chapter 2

Methodology

In this chapter, dataset overview, data preprocessing techniques, implementation details, ML modes used for classification, post-hoc XAI techniques, hyperparameter tuning approach and the comprehensive methodology used for diabetes prediction is described. The detailed workflow diagram of the methodology is shown in Figure 2.1.

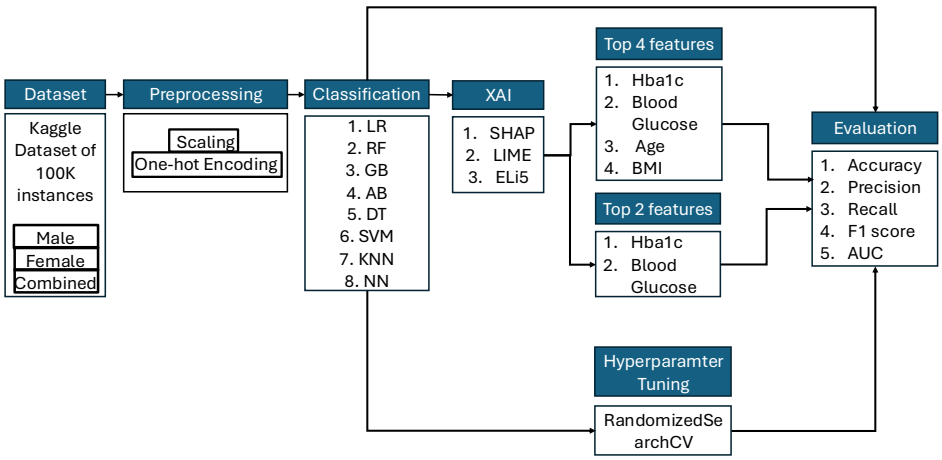

Figure 2.1: Workflow diagram of methodology for diabetes prediction.

2.1 Dataset

In this work, a Kaggle dataset consisting of 100,000 instances is used. The data was collected between 2015 and 2022. This dataset includes year, gender, age, race (African American, Asian, Caucasian, Hispanic and other), hypertension, heart disease, smoking history, BMI, HbA1c level, blood glucose level, and diabetes as features. The 15 features used in the dataset are shown in Table 2.1. The dataset is divided into two classes; negative and positive, which are represented as 0 and 1 respectively. This dataset is highly imbalanced; 91,500 are negative cases while 8,500 are only positive cases. The two genders; male and female are represented in the data, where females represent 59% while males represent 41% of the population.

2.2 Data Preprocessing

The diabetes dataset consists numerical (year, age, BMI, HbA1c, and blood glucose level) and categorical (gender, race, hypertension, heart disease, smoking history, and diabetes) features. Since ML models cannot directly interpret categorical data, these features need to be converted into a numerical form while preserving their discrete meaning. In order to address the issue, a preprocessing pipeline is constructed using the column transformer, which applies standard scaler to the numerical columns and one hot encoder to the categorical columns. This preprocessing step ensures that both numerical and categorical features are properly scaled and transformed into a representation that is comprehensible to ML algorithms thereby facilitating effective model training.

2.3 Implementation Details

All experiments in this study were executed on CPU only `fir-compute` nodes of the Fir supercomputer by the Digital Research Alliance of Canada (https://ccdb.alliancecan.ca). Each node provides 192 AMD EPYC CPU cores, 768 GB DDR5 system memory, and a high-speed InfiniBand interconnect. During development and evaluation, jobs were typically launched through the simple Linux utility for resource management (SLURM) workload manager using interactive allocations requesting 2 CPU cores and 4 GB of RAM per task. All models and experiments were implemented in Python. To evaluate compu-

tational efficiency, testing execution time was measured for each model on the entire test dataset. This measurement records the time taken to generate predictions for all test samples and was performed on the same `fir-compute` nodes to ensure consistency across models.

Feature	**Description**	**Distribution / Statistics**
year	Year of data record collection	2019: 79.7%, 2016: 8.8%, 2015: 8.8%, 2018: 2.7%
gender	Patient gender	Female: 58.6%, Male: 41.4%, Other: 0.0%
age	Age of patient (years)	Mean = 41.89 ± 22.52
race: African American	Race indicator for African American ethnicity (1 = Yes, 0 = No)	0: 79.8%, 1: 20.2%
race: Asian	Race indicator for Asian ethnicity (1 = Yes, 0 = No)	0: 80.0%, 1: 20.0%
race: Caucasian	Race indicator for Caucasian ethnicity (1 = Yes, 0 = No)	0: 80.1%, 1: 19.9%
race: Hispanic	Race indicator for Hispanic ethnicity (1 = Yes, 0 = No)	0: 80.1%, 1: 19.9%
race: Other	Race indicator for Other ethnicity (1 = Yes, 0 = No)	0: 80.0%, 1: 20.0%
hypertension	Presence of hypertension (1 = Yes, 0 = No)	0: 92.5%, 1: 7.5%
heart_disease	Presence of heart disease (1 = Yes, 0 = No)	0: 96.1%, 1: 3.9%
smoking_history	Smoking status of the patient	No Info: 35.8%, Never: 35.1%, Former: 9.4%, Current: 19.7%
bmi	Body Mass Index (weight-to-height ratio)	Mean = 27.32 ± 6.64
hbA1c_level	Average blood glucose over past 3 months (%)	Mean = 5.53 ± 1.07
blood_glucose_level	Current blood glucose measurement (mg/dL)	Mean = 138.06 ± 40.71
diabetes	Target variable indicating diabetes diagnosis (1 = Yes, 0 = No)	0: 91.5%, 1: 8.5%

Table 2.1: Summary of the diabetes dataset including feature descriptions and value distributions.

2.4 Models

In this work, eight ML classifiers; LR, RF, GB, AB, DT, SVM, KNN and NN are used for the diabetes classification. These models represent a diverse set of learning paradigms ranging from linear classifiers to tree-based ensembles and instance based learning, allowing a comprehensive comparison of different algorithmic behaviors.

2.4.1 Logistic Regression

LR is a linear classification model that estimates class probabilities using the logistic (sigmoid) function. It models the relationship between input features and the target variable through weighted coefficients and a bias term. In this work, it is configured with `max_iter = 1000` to ensure convergence. LR is widely used for binary and multiclass classification tasks (30). LR works by finding a decision boundary that best separates classes in a linear feature space. It optimizes the log-likelihood function using gradient-based methods. Despite its simplicity, LR performs well when the relationship between features and output is approximately linear. It is also highly interpretable, as the learned coefficients directly indicate the influence of each feature on the probability of diabetes. LR is computationally efficient and performs well in high-dimensional settings, making it a strong baseline for medical classification tasks.

2.4.2 Random Forest

RF is a tree-based ensemble learning classifier. It is based on bootstrap aggregation (bagging), where randomly selected samples of the training data are used to train individual decision trees using replacement. This provides each tree with unique training subsets, and the inherent randomness makes RF robust to overfitting. RF is used for both classification and regression tasks (31). Each tree contributes a vote toward the final prediction, and RF aggregates these votes through majority voting. It handles non-linear relationships, complex feature interactions, and noisy datasets effectively. RF also provides measures of feature importance, which is particularly useful for understanding the key predictors in diabetes risk. Its robustness to outliers and ability to handle correlated features make it a reliable choice for biomedical applications.

2.4.3 Gradient Boosting

GB is an ensemble method that builds sequential decision trees, where each tree is trained to correct the errors of the previous ones. It minimizes a differentiable loss function using gradient descent, resulting in strong predictive performance. In this work, GB is configured with `random_state=42` to ensure reproducibility (32). GB focuses on the residuals (errors) at each stage, gradually improving model accuracy. Because each tree is shallow, the method builds a strong learner through many weak learners. GB can model complex patterns, subtle interactions, and non-linear relationships, making it powerful for tabular medical data. However, it may require careful hyperparameter tuning to avoid overfitting. Its strong predictive capability often yields superior performance in classification tasks involving metabolic health indicators.

2.4.4 AdaBoost

AdaBoost (AB), or Adaptive Boosting, combines multiple weak learners into a strong classifier by iteratively assigning higher weights to misclassified samples. Each new learner focuses on correcting the mistakes of the previous ensemble. AB is implemented with `random_state=42` in this work (33). AdaBoost typically uses shallow decision stumps as base learners. Because it adaptively redistributes weights, the model becomes highly sensitive to minority patterns or hard-to-classify instances. This makes AB particularly useful when dealing with subtle indicators of diabetes risk. It works well with clean and moderately sized datasets but is sensitive to noise and outliers, as misclassified noisy samples may receive disproportionately high weights.

2.4.5 Decision Tree

DT is a non-parametric tree-structured classifier that recursively splits the dataset based on feature values to maximize class purity. It provides interpretable decision rules and can overfit small datasets if not regularized. DT is initialized with `random_state=42` (34). DTs use measures such as Gini impurity or entropy to select the best split at each node. They are easy to visualize and explain, which is advantageous for healthcare applications where interpretability is essential. DTs can model non-linear relationships and interactions effectively but tend to overfit without pruning. As a standalone model, DT provides a

transparent baseline to compare more complex ensemble methods.

2.4.6 Support Vector Machine

SVM is a margin-based classifier that identifies the optimal hyperplane separating classes with maximum margin. Kernel functions allow it to model non-linear decision boundaries. In this work, SVM is configured with `probability=True` and `random_state=42` to enable probability estimates and reproducibility (35). SVM seeks the boundary that maximizes class separation, making it highly effective for high-dimensional data. Using kernels such as RBF, SVM can capture complex patterns relevant for diabetes prediction. It is robust to outliers and works well when the dataset is not extremely large. However, SVM can be computationally expensive, especially with non-linear kernels, and its performance depends on careful selection of hyperparameters such as C and gamma.

2.4.7 K-Nearest Neighbors

KNN is a distance-based, instance-learning classifier that assigns a class to a sample based on the majority vote of its k nearest neighbors in the feature space. It is non-parametric and sensitive to feature scaling. In this work, KNN is used with default settings (36). KNN makes predictions based on similarity, making it straightforward and intuitive. Because it does not build an explicit model, its performance heavily depends on the choice of k and the distance metric. KNN performs well when similar samples cluster together in the feature space. However, it can be slow during inference and is sensitive to irrelevant or unscaled features. Despite these limitations, it provides a simple comparison point for more complex algorithms.

2.4.8 Neural Network

NN is a feed-forward model consisting of layers of interconnected neurons. It learns non-linear mappings from inputs to outputs using backpropagation. In this work, the network is implemented as an `MLPClassifier` with `max_iter=50`, `early_stopping=True`, and `random_state=42` to prevent overfitting and ensure reproducibility (37). MLPs can model complex relationships that are difficult for linear or tree-based models to capture. By stacking multiple hidden layers, NNs can learn hierarchical abstractions of the input data. Early

stopping prevents overfitting, which can occur easily in small or medium-sized medical datasets. NNs are powerful but require careful tuning of architecture, activation functions, and regularization techniques. They offer high flexibility and are suitable for capturing non-linear dependencies in diabetes-related features.

2.5 Explainability

In this section, three explainability techniques used in this work, SHAP, LIME and ELI5, are described.

2.5.1 SHAP

SHAP is a model agnostic explainability method rooted in the Shapley values from cooperative game theory. It provides a theoretically sound framework to assign an importance score to each feature by calculating the marginal contribution of that feature to the prediction outcome across all possible feature subsets. The fundamental idea treats the prediction as a "payout" of a cooperative game and each feature as a "player" ensuring fair and consistent attribution. Mathematically for a model f and feature i in a feature set N, the Shapley value is defined as

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

where $\phi_i(f)$ denotes the SHAP value representing the contribution of feature i to the prediction of model f. N is the set of all input features, and S denotes a subset of features excluding i. The notation $|S|$ represents the number of features in subset S, and $|N|$ represents the total number of features. The factorial operator ! is used to compute combinatorial weights. The term $f(S \cup \{i\})$ represents the model prediction when feature i is included in the subset S, while $f(S)$ represents the prediction when the feature is absent. In practice, $f(S)$ is estimated by marginalizing the absent features over a background dataset, ensuring that missing information is represented realistically. This ensures local accuracy, meaning the sum of feature attributions equals the model prediction, and consistency, meaning that if a model changes such that a feature contributes more, its SHAP value cannot decrease. The SHAP workflow involves generating subsets of features, computing marginal contribu-

tions, weighting them combinatorially, and aggregating contributions to determine feature importance. For practical scalability, TreeSHAP enables exact polynomial-time computation for tree-based models, while KernelSHAP provides a model-agnostic approximation using weighted linear regression. SHAP is capable of producing both local explanations for individual predictions and global explanations by aggregating feature contributions across the dataset. Its theoretical guarantees and interpretability make it highly suitable for high-stakes applications in healthcare, finance, and any domain requiring trustworthy AI explanations. SHAP also supports interaction effects between features allowing more nuanced insight into complex models. Despite its advantages, computational cost can be high for large datasets or models with many features unless approximation methods are employed (12).

2.5.2 LIME

LIME explains individual predictions by approximating a black-box model f locally with a simpler, interpretable surrogate model g. Given an instance x

$$\arg\min_{g \in G} L(f, g, \pi_x) + \Omega(g), \tag{2}$$

where $L(f, g, \pi_x) = \sum_{z \in Z} \pi_x(z)(f(z) - g(z))^2$ is the weighted loss over perturbed samples Z, with Z denoting the set of synthetic neighborhood samples generated by perturbing the original instance x, $\pi_x(z)$ is the proximity-based weight of perturbed instance z to the original instance x, and $\Omega(g)$ penalizes complexity to maintain interpretability.

The LIME algorithm consists of the following steps.

1. Select an instance to explain.
2. Perturb features to generate synthetic neighborhood data.
3. Predict outcomes for perturbed instances using the black-box model.
4. Weight perturbed instances based on proximity.
5. Fit the interpretable surrogate model on weighted data.
6. Use the surrogate model to assign feature importance values locally.

LIME provides local interpretability which is useful for explaining individual predictions of complex models like gradient boosting, deep neural networks, or ensemble methods. It is model agnostic, supports multiple data modalities (tabular, text, images), and allows practitioners to inspect which features most influence a prediction. However, its explanations may vary depending on sampling randomness and the choice of neighborhood, which can lead to instability if not properly controlled (13).

2.5.3 ELI5

ELI5 (38) is a Python library designed to make ML models intuitively understandable by providing human readable explanations. It supports a wide range of models including linear classifiers, decision trees, ensemble methods and text based models. ELI5 offers both global explanations, which describe overall feature importance across the model and local explanations which describe contributions of features to individual predictions. For linear models, ELI5 extracts feature coefficients w_i from the model function

$$f(x) = \sum_i w_i x_i + b, \tag{3}$$

where w_i are the feature weights and b is the bias term, and presents them as feature importances. For the tree based models, it traverses decision paths and computes the contribution of each node along the path, highlighting which features and splits led to a particular prediction. For text models, it identifies influential words affecting the model output. Its workflow involves detecting the model type, extracting contributions using built-in methods, aggregating these contributions for local or global explanations and rendering them in a human friendly format. ELI5 is model agnostic for many types of models, it can explain both individual and global predictions, visualizes decision logic effectively and emphasizes interpretability. It is particularly suited for practitioners who need clear actionable explanations without deep mathematical knowledge although it relies on the underlying model for correctness (38).

2.6 Evaluation Metrics

In this section, the metrics used for models evaluations are described. The metrics used are accuracy, precision, recall, F1-score, AUC and confusion matrix.

2.6.1 Accuracy

Accuracy is the most widely used evaluation metric for classification tasks. It measures the proportion of correctly classified instances out of the total number of instances. Accuracy provides an overall sense of model performance as given by

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{4}$$

where TP is the number of true positives, TN is true negatives, FP is false positives, and FN is false negatives. Accuracy is intuitive and simple to compute and it works best when the dataset has balanced classes. However, for highly imbalanced datasets, metrics like Precision, Recall, or F1-score are preferred.

2.6.2 Precision

Precision quantifies the proportion of correctly predicted positive instances out of all instances predicted as positive. It is a measure of exactness and indicates how many of the predicted positives are actually true positives. Precision is particularly important in situation where the cost of false positives is high and is given by

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{5}$$

High precision implies that the model makes very few false positive errors. However, precision alone does not account for false negatives, so it is often analyzed together with recall.

2.6.3 Recall (Sensitivity)

Recall also known as sensitivity or true positive rate, measures the proportion of correctly predicted positive instances out of all actual positive instances. It represents the model's ability to identify positive cases and is given by

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{6}$$

High recall indicates that the model successfully captures most positive instances making it crucial in medical diagnosis, fraud detection, and other high risk domains where missing

positive cases is costly. However, recall does not consider false positives which is why balancing with precision is necessary.

2.6.4 F1-Score

F1-Score is the harmonic mean of precision and recall, providing a single metric that balances both aspects of model performance. It is particularly useful when classes are imbalanced and is given by

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{7}$$

The harmonic mean ensures that both precision and recall contribute equally and a low value in either reduces the F1-score. F1-score is widely used in healthcare, fraud detection, and other domains where both false positives and false negatives are critical.

2.6.5 Receiver Operating Characteristic (ROC) and Area Under the Curve (AUC)

The ROC curve is a graphical representation of a classifier's performance across different threshold values. It plots the true positive rate (Recall) against the false positive rate and is given by

$$\text{FPR} = \frac{FP}{FP + TN}. \tag{8}$$

ROC-AUC measures the area under the ROC curve, providing a single value to summarize performance. An AUC of 1 indicates perfect classification, while 0.5 corresponds to random guessing. ROC-AUC is particularly useful for evaluating models under class imbalance and for comparing multiple classifiers as it accounts for all possible classification thresholds.

2.6.6 Confusion Matrix

The Confusion Matrix is a tabular representation of classification performance, providing a comprehensive view of true positives, true negatives, false positives and false negatives. For a binary classification problem the confusion matrix is given as

$$\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}. \tag{9}$$

The confusion matrix serves as the foundation for computing accuracy, precision, recall and F1-score. It is particularly useful for diagnosing the type of errors a classifier makes and for understanding model behavior in multiclass classification tasks by extending the matrix to multiple rows and columns.

2.7 Hyperparameter Tuning

Hyperparameter tuning is the process of searching for the most effective parameter values to achieve optimal model performance. In this work, the RandomizedSearchCV method is used for hyperparameter tuning. The technique is described in the following section.

2.7.1 Randomized Search Cross-Validation

Randomized Search Cross-Validation (RandomizedSearchCV) is an optimization technique used for hyperparameter tuning in ML models. It is designed to efficiently search through a predefined hyperparameter space by sampling a fixed number of parameter combinations from specified distributions. Unlike Grid Search which exhaustively evaluates all possible parameter combinations, Randomized Search explores only a subset, offering a favorable balance between computational efficiency and model performance (39). Let $\Theta = \{\theta_1, \theta_2, \ldots, \theta_n\}$ represent the hyperparameter space, where each θ_i corresponds to a particular configuration of model parameters. Instead of evaluating all combinations $|\Theta|$, Randomized Search selects k random samples $\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(k)}$ from the parameter distributions

$$\theta^{(i)} \sim P(\Theta), \tag{10}$$

where $P(\Theta)$ denotes the joint probability distribution over all hyperparameters. For each sampled configuration $\theta^{(i)}$, the model is trained and evaluated using cross-validation (CV), ensuring robustness to overfitting and variance due to random sampling. The cross-validation process partitions the training data into K folds and performance is averaged across all folds

as

$$\text{Score}(\theta^{(i)}) = \frac{1}{K}\sum_{k=1}^{K}\mathcal{M}(f_{\theta^{(i)}}, D_k), \tag{11}$$

where $\mathcal{M}$ represents the chosen evaluation metric (e.g., accuracy, F1-score, ROC-AUC), $f_{\theta^{(i)}}$ is the trained model with parameters $\theta^{(i)}$, and D_k denotes the validation fold. After all iterations the best performing parameter set is selected as

$$\theta^* = \arg\max_{\theta^{(i)}} \text{Score}(\theta^{(i)}). \tag{12}$$

2.7.2 Algorithmic Workflow

The RandomizedSearchCV algorithm operates through the following steps.

1. Define the model and specify the hyperparameter distributions.
2. Randomly sample k combinations of hyperparameters from the specified distributions.
3. For each sampled combination, perform K-fold cross-validation.
4. Compute the average validation performance for each configuration.
5. Select the hyperparameter combination that yields the best cross-validation score.

2.7.3 Advantages and Properties

- Efficiency: Evaluates only a limited number of parameter combinations, making it more computationally efficient than exhaustive Grid Search.
- Scalability: Well suited for high-dimensional hyperparameter spaces where exhaustive search is infeasible.
- Flexibility: Allows sampling from continuous, discrete or categorical parameter distributions.
- Global exploration: Avoids bias toward fixed grid points and increases the likelihood of discovering near-optimal regions of the parameter space.

- Parallelization: Evaluations of different hyperparameter configurations can be parallelized across multiple cores or GPUs.

2.7.4 Practical Considerations

The performance of Randomized Search depends on the number of iterations (n_{iter}) and the range or distribution of parameters defined by the user. Typically, uniform or log-uniform distributions are used for continuous parameters, while discrete distributions are used for categorical parameters. Increasing n_{iter} improves the probability of finding a near optimal solution but increases computation time. When combined with cross validation, RandomizedSearch provides a robust estimate of model generalization performance. For example, in scikit-learn, the RandomizedSearchCV() function automates this process, requiring specification of the model, parameter distributions, number of iterations and cross-validation folds.

2.7.5 Mathematical Insight

The expected improvement of Randomized Search over Grid Search arises from its probabilistic exploration of the hyperparameter space. As shown by (39), when only a subset of hyperparameters significantly influences performance random sampling yields an exponentially higher probability of selecting an optimal region compared to uniform grid coverage. This property makes Randomized Search particularly powerful in complex models such as neural networks and ensemble methods. RandomizedSearchCV provides a computationally efficient, flexible, and statistically grounded approach for hyperparameter tuning. By leveraging random sampling and cross-validation, it identifies near optimal parameter configurations without requiring exhaustive search making it a preferred choice for large-scale or resource constrained ML applications.

Chapter 3

Results

In this section, test results are presented in the order in which they were obtained using Juypter notebook and Python. The evaluation includes baseline performance, optimized performance after hyperparameter tuning using RandomizedSearchCV, the important feature extraction using XAI analyses and the performance based on the subset of chosen 4 features and 2 features respectively. All classifiers were executed with consistent preprocessing steps to ensure comparability across the male, female and combined datasets.

In the first step, eight classifiers were evaluated using the full dataset of 100,000 instances. These classifiers were selected to provide a wide variety of model behaviors, ranging from linear methods to ensemble approaches. For this stage, the dataset was split into training and testing subsets for the classification. The baseline test results for the male, female, and combined group are presented in the following tables. In the second step, hyperparameter tuning was performed using RandomizedSearchCV to identify optimized parameter configurations for each classifier. The hyperparameter tuned results for all three cohorts are reported sequentially, allowing direct comparison with their corresponding baseline performances.

In the third step, SHAP, LIME, and ELI5 explainable AI techniques were applied to interpret the predictions and identify the key features influencing behavior of the model. SHAP global bar plots, summary plots, and decision plots along with LIME bar plots and ELI5 weight based explanations, provide a comprehensive understanding of feature importance across the dataset. Finally, based on the most influential features identified through the explainability analysis, the baseline models were re-evaluated using the top four and top two features. These results are presented in separate tables to assess the effect of feature

reduction on classifier performance.

3.1 Baseline Performance

In the first step, eight classifiers were evaluated using the male, female and combined datasets. These classifiers were chosen to represent a broad range of model types, including linear models, tree based learners, ensemble methods, kernel based methods, and neural networks. For each experiment the dataset was divided into training and testing subsets to assess the baseline classification performance. The results for the male, female and combined groups are presented in Tables 3.1, 3.2, and 3.3 respectively.

3.1.1 Male Group

For the male group (Table 3.1), GB, RF and AB classifiers achieved the highest accuracy of 0.964. Among these models GB and RF produced strong precision scores of 0.981 and 0.972 respectively with relatively low false positives (FP = 10 for GB and FP = 15 for RF). AB reached perfect precision 1.0 but showed lower recall 0.631, indicating that while it avoided false positives entirely (FP = 0), it misclassified more true cases (FN = 298). The NN also performed strongly with an accuracy of 0.960 and an AUC of 0.965. LR achieved an accuracy of 0.951 with high precision 0.847 but a lower recall of 0.610 showing that many positive samples remained undetected. The DT classifier had the lowest AUC 0.842 due to higher variability in false positives (FP = 245). Overall, most classifiers achieved accuracy between 94% and 96%, with GB performing slightly better across multiple metrics. In terms of execution time, which is measured in seconds, DT appeared to be the most efficient model, requiring only 0.07 seconds, whereas SVM appeared to be the most computationally costly model, taking 24.54 seconds to execute.

3.1.2 Female Group

For the female group (Table 3.2), GB again achieved the highest accuracy at 0.976 followed by RF and AB 0.975. GB obtained the best AUC score of 0.979 and high precision of 0.986, with relatively low false positives (FP = 9). AB reached perfect precision 1.0 with no false positives (FP = 0) but similar to the male group its recall remained moderate at 0.677 (FN =

288). The NN performed competitively with an accuracy of 0.973 and a balanced profile of precision 0.945 and recall 0.691. LR achieved an accuracy of 0.965, performing better than in the male group, though recall remained moderate 0.624. The SVM and KNN models showed weaker recall values 0.576 and 0.556, resulting in F1 scores below 0.73. The DT classifier, despite achieving reasonable recall 0.739, produced the lowest AUC 0.857 due to higher false positive counts (FP = 270). Overall, most classifiers performed between 96% and 98% accuracy, with GB showing the strongest balance across metrics. For this group, LR appeared to be the most efficient model, requiring only 0.03 seconds, whereas SVM was the most computationally expensive, taking 43.19 seconds.

3.1.3 Combined Group

For the combined dataset (Table 3.3), GB classifier again achieved the highest accuracy 0.973, representing consistent strength across all groups. GB also achieved the highest precision among the top performers 0.985 and a strong AUC of 0.978, with only 18 false positives. The RF and AB classifiers both achieved an accuracy of 0.972. RF produced high precision 0.968 but moderate recall 0.687, while AB again achieved perfect precision 1.0 at the cost of higher false negatives (FN = 554). The NN also performed competitively with an accuracy of 0.972 and an AUC of 0.976. LR achieved a strong accuracy of 0.961 and high precision 0.870 but showed limited recall 0.633. The DT classifier had the lowest performance among the group, with an accuracy of 0.951 and the highest number of false positives (FP = 544). SVM and KNN achieved moderate accuracy 0.964 and 0.958 but exhibited lower recall 0.588 and 0.581 leading to lower F1 scores. For the combined group, LR was again the most efficient model, requiring only 0.06 seconds, while SVM was the most computationally expensive, taking 197.42 seconds.

Across all datasets, GB consistently demonstrated the strongest and most balanced performance, achieving top results in accuracy, precision and AUC. AB repeatedly delivered perfect precision but lower recall due to a higher number of false negatives. LR remained stable but performed poorly on recall, while DTs exhibited the lowest AUC and highest variability in misclassification. These baseline results establish a comprehensive performance benchmark prior to hyperparameter tuning and explainability analysis.

3.2 Hyperparameter Tuning

3.2.1 Hyperparameter Tuning Results (RandomizedSearchCV)

In the second step of the analysis, all eight classifiers were fine-tuned using RandomizedSearchCV with default settings to optimize their hyperparameters for better classification performance. This step aimed to improve the baseline performance of each model by identifying parameter configurations that enhance classification accuracy, F1 score, precisioin, recall and AUC while reducing misclassification errors. As in the baseline stage, the results are reported separately for the Male, Female and Combined groups to capture gender specific differences in model behavior. (Tables 3.4, 3.5, and 3.6) summarize the performance metrics obtained after hyperparameter tuning.

3.2.1.1 Male Group

For the male dataset, hyperparameter tuning led to substantial improvements in F1-scores, precision and recall across almost all classifiers. The RF, GB, AB and DT classifiers emerged as the top performing models achieving an accuracy of 0.964 and an F1-score between 0.960 and 0.961. These models also demonstrated strong minority class recognition, with recalls of 0.964 and high precision ranging from 0.964 and 0.965. Interestingly, RF achieved the fewest false positives (FP = 2) and a relatively low number of false negatives (FN = 296) indicating a balanced and robust performance. The LR classifier also improved significantly compared to the baseline, reaching an F1-score of 0.947 and recall of 0.951, though still behind the ensemble based models. The KNN classifier showed the weakest performance in the tuned setting, with an accuracy of 0.947 and a drop in performance for the minority class (TP = 389) consistent with its limitations when facing high dimensional tabular data. The SVM and NN classifiers produced competitive results with NN performing better than SVM in both accuracy 0.963 vs. 0.953 and F1-score 0.959 vs. 0.948. Overall, hyperparamter tuning significantly reduced the performance gap among classifiers in the Male group. In hyperparameter tuning setting, LR and DT appeared to be the most efficient models, requiring only 0.01 seconds, whereas KNN appeared to be the most computationally costly model, taking 2.15 seconds to execute.

3.2.1.2 Female Group

For the female dataset, hyperparameter tuning produced even more notable improvements. The RF, GB, AB, DT and NN classifiers all achieved accuracy values between 0.975 and 0.976, with corresponding F1-scores of 0.973 and 0.974, demonstrating consistent and high performance. The GB classifier achieved one of the best overall performance achieving an accuracy of 0.976, precision of 0.976, recall of 0.976 and AUC of 0.979. The AB and DT classifiers performed nearly identically with both achieving near perfect balance between precision and recall. RF resulted in zero false positives (FP = 0), indicating exceptional separation between classes in the Female dataset. Compared to the Male group, all classifiers showed stronger minority class predictive power, partly due to the richer feature separability observed in Female samples. As in the baseline setting SVM and KNN were comparatively weaker, though both still improved significantly after tuning. For female group, again LR and DT appeared to be the most efficient models, requiring only 0.01 seconds, whereas KNN again was the most computationally expensive, taking 2.94 seconds.

3.2.1.3 Combined Group

For the combined dataset, hyperparameter tuning improved all classifiers, reducing the gap in performance and leading to more consistent results. The ensemble based methods again emerged as the most effective. The RF, GB, AB, DT, and NN models each achieved an accuracy between 0.972 and 0.973 with F1-scores around 0.970. Among these classifiers, AB achieved the highest accuracy of 0.973, with a precision and recall of 0.973 and a low false-positive count (FP = 15). Similarly, GB achieved an AUC of 0.979, the highest among all models. The LR, SVM and KNN classifiers, although improved compared to their baseline performance, remained behind the ensemble based classifiers. SVM achieved an accuracy of 0.962 but exhibited a higher number of false negatives (FN = 687), which impacted its recall. KNN performed the weakest, consistent with earlier observations with an accuracy of 0.958 and the largest number of false negatives (FN = 802). Overall, hyperparameter tuning notably strengthened the performance of all classifiers on the combined dataset with ensemble models showing superior generalization and robustness. For the combined group, LR and DT was again the most efficient models, requiring only 0.01 seconds, while KNN was the most computationally expensive, taking 6.11 seconds.

3.3 Explainable AI Results

In this section, explainability analysis is presented for all eight ML models trained on the combined dataset using SHAP, LIME and ELI5. These techniques were applied to understand both the global behavior of the models and the local reasoning behind individual predictions. To maintain clarity, results are presented in a general form followed by representative examples for detailed understanding.

3.3.1 SHAP Global Feature Importance (Bar Plots)

SHAP bar plots were generated for all eight models to identify the average contribution of each feature. These plots rank features based on the mean absolute SHAP values. While individual models exhibit some variation, several features consistently appear among the most influential across all classifiers. The complete SHAP bar plots for each model are presented in Figures 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, and 3.8. In these bar plots, the y-axis lists the input features ranked according to their importance, while the x-axis represents the mean absolute SHAP value for each feature. The SHAP value measures how much a feature contributes to the model's prediction relative to the baseline prediction. By taking the absolute value and averaging it across all observations, the plot reflects the overall magnitude of each feature's influence on the model predictions. Therefore, longer bars indicate features that have a stronger overall impact on the model's decision-making process across the dataset. For example, in the GB model, `hbA1c_level`, `age`, `blood_glucose_level`, and `bmi` exhibit the highest contributions, demonstrating strong global influence over the prediction outcomes. It is important to note that SHAP bar plots represent the magnitude of feature importance only and do not show whether a feature increases or decreases the predicted probability. The direction of influence is instead illustrated in SHAP summary (beeswarm) plots.

3.3.2 SHAP Summary (Beeswarm) Plots

SHAP summary plots are used to interpret ML models by quantifying the contribution of each input feature to the model's predictions. These plots provide both global interpretability, by showing the overall importance of features, and local interpretability, by illustrating how individual feature values influence predictions for specific observations. In the sum-

mary plot, the y-axis lists the model input features, ordered by their overall importance based on the mean absolute SHAP values across the dataset. Features appearing at the top have a larger overall impact on the model's predictions, while features closer to the bottom have comparatively smaller contributions.

The x-axis represents the SHAP value, which measures the contribution of a feature to the predicted outcome relative to the model's baseline prediction. A positive SHAP value indicates that the feature increases the predicted probability of the positive class (diabetes), whereas a negative SHAP value indicates that the feature decreases the predicted probability and pushes the prediction toward the negative class (non-diabetes). The magnitude of the SHAP value reflects the strength of the feature's influence on the prediction. Each point in the plot represents an individual observation (patient) in the dataset. The horizontal position of the point shows how strongly that feature affects the prediction for that specific observation. The color of each point represents the raw feature value, which corresponds to the original value of the feature in the dataset before any transformation or scaling used for model training. A color gradient is used to visualize these values, where blue indicates lower feature values and red indicates higher feature values. This allows the reader to observe how different ranges of feature values influence the model's predictions.

Figures 3.9, 3.10, 3.11, 3.12, 3.13, 3.14, 3.15, and 3.16 present the SHAP summary plots for the eight evaluated models. The results indicate that features such as `hbA1c_level` and `blood_gulocose_level` consistently have the largest SHAP values, demonstrating their strong influence on diabetes prediction. Higher values of these features tend to shift the prediction toward the positive class, reflecting their well-established clinical relevance in diabetes diagnosis. In contrast, features such as race categories or certain smoking history indicators generally produce SHAP values close to zero, suggesting a relatively smaller or more variable influence on the predictions. Overall, these plots confirm that the models rely primarily on clinically meaningful predictors while also highlighting variability in feature contributions across different observations.

3.3.3 SHAP Decision Plots and Corresponding Tables

SHAP decision plots illustrate how feature contributions accumulate to produce a specific prediction. For each model, representative samples were selected and both the decision plot and a corresponding table listing individual feature contributions are provided Figures 3.17, 3.18, 3.19, 3.20, 3.21, 3.22, 3.23, and 3.24, and Tables 3.7, 3.8, 3.9, 3.10, 3.11,

3.12, 3.13, and 3.14. Each table shows raw SHAP values for individual features, representing their positive or negative contribution to the predicted outcome relative to the model's base value. For instance, in the GB model one sample's prediction is primarily driven by `hbA1c_level` (+8.950), followed by `age` (+0.213), while `blood_glucose_level` (-0.573) slightly reduces the prediction score. These tables and plots allow precise interpretation of local decision making, showing how the model transitions from the base value to the final output through cumulative feature contributions.

3.3.4 LIME Local Explanations

LIME was applied to approximate the behavior of each trained model around selected instances. LIME explains individual predictions by constructing a local surrogate model that approximates the complex model in the neighborhood of a specific observation. This allows the identification of which features contribute most to the prediction for that particular instance. The resulting LIME explanation plots are presented as bar charts. In these plots, the y-axis lists the input features that most strongly influence the prediction for the selected instance, while the x-axis represents the contribution weight assigned by LIME to each feature. These contribution values indicate how strongly each feature affects the model's prediction for that specific observation. In addition, the feature conditions displayed on the y-axis (e.g., $\texttt{hbA1c\_level} \leq -0.68$ or $-0.93 < \texttt{blood\_glucose\_level} \leq 0.05$) correspond to discretized intervals of the feature values used by LIME to locally approximate the model. Because the dataset features were standardized prior to model training, these threshold values are expressed as standardized values (z-scores), which indicate how many standard deviations a feature value is above or below the dataset mean rather than representing the raw measurement units. Consequently, negative threshold values simply indicate that the feature value is below the dataset mean, while positive values indicate that it is above the mean.

Since the presented plots explain class 1 (diabetes), positive contribution values indicate features that support the prediction of diabetes, while negative contribution values represent features that oppose the prediction and push the model toward the non-diabetes class. Therefore, the positive and negative values on the x-axis correspond to the weights learned by the local surrogate model, indicating the direction and strength of each feature's contribution to the predicted probability. It is important to note that the sign of the feature threshold (i.e., whether the z-score is positive or negative) is independent of the direction

of the contribution shown in the bar chart. A feature may have a negative standardized value (below the dataset mean) but still contribute positively to the prediction if the local surrogate model determines that such a value increases the probability of the predicted class. The length of each bar reflects the magnitude of the contribution, where longer bars indicate a stronger influence on the prediction. Figures 3.25, 3.26, 3.27, 3.28, 3.29, 3.30, 3.31, and 3.32 present the LIME explanations for the eight models. For example, in LR prediction instance, LIME indicates that features such as `hbA1c_level` and `blood_glucose_level` strongly oppose the prediction of diabetes, while features such as `smoking_history_No current` slightly support the prediction. These explanations illustrate how different feature values influence the model's decision at the individual prediction level.

3.3.5 ELI5 Feature Weights and Explanations

In this study, ELI5 was applied using permutation importance, which estimates the contribution of each feature by measuring the decrease in model performance when the feature values are randomly shuffled. The resulting ELI5 outputs present feature importance rankings in a tabular format, where each row corresponds to an input feature and its associated importance weight. These weights indicate how much the model's predictive performance decreases when a given feature is permuted, thereby reflecting its relative contribution to the model's predictions.

Higher importance values indicate that the feature plays a greater role in the model's decision-making process, while values close to zero suggest that the feature has little influence on the predictions. Small negative values may occasionally appear due to random variation during the permutation process and generally indicate negligible importance. Figures 3.33, 3.34, 3.35, 3.36, 3.37, 3.38, 3.39, and 3.40 present the feature importance rankings for all evaluated models. These results complement the SHAP and LIME analyses by highlighting which features consistently influence predictions across different models. In particular, clinically relevant variables such as `hbA1c_level`, `blood_glucose_level`, `BMI` and `age` are consistently identified as strong predictors of diabetes risk.

3.3.6 Consolidated Feature Importance Across All Explainability Methods

By integrating SHAP bar plots, SHAP summary plots, SHAP decision tables, LIME explanations and ELI5 weights, a consolidated list of the most influential features was created. Features consistently appearing among the top contributors across multiple models and explainability techniques were selected. Based on this analysis, two reduced feature sets were extracted.

1. Top four most important features
2. Top two most important features

Baseline classifiers were retrained using these reduced feature subsets. The performance results for the four feature subset and two feature subset are given in Table 3.15 and Table 3.16 respectively. This consolidated approach ensures that feature selection is both data driven and interpretable across models.

3.4 Results for Selected Feature Subsets

Based on the consolidated feature importance analysis from SHAP, LIME, and ELI5, two reduced feature sets were selected for retraining and evaluation of the baseline classifiers.

- **Four features:** `HbA1c`, `Blood Glucose`, `Age`, `BMI`
- **Two features:** `HbA1c`, `Blood Glucose`

3.4.1 Results Using Four Selected Features

Table 3.15 summarizes the performance metrics of all eight ML models trained on the combined dataset using the four feature subset. Metrics include Accuracy, F1-score, Precision, Recall, AUC and CM. The results indicate that models maintain high predictive performance even with a reduced feature set. The GB and AB models achieve the highest accuracy of 0.972 and AUC of 0.976 and 0.970, respectively, demonstrating that these four features capture most of the relevant predictive information. With reduced set of features,

four features in this case, LR appeared to be most cost effective model with 0.03 seconds execution time, while SVM appeared to be the most expensive model with 109.15 execution time.

3.4.2 Results Using Two Selected Features

Table 3.16 shows the performance of the same models when trained with only the top two features: `HbA1c` and `Blood Glucose`.

Even with only two features most models retain strong predictive performance, particularly ensemble methods (RF, GB, AB), indicating that `HbA1c` and `Blood Glucose` are highly informative and can be sufficient for efficient predictions with minimal feature inputs. A comprehensive performance comparison, which compares AUC across baseline, hyperparameter tuning, four features and two features is shown in Table 3.17. The highest AUC of 0.979 across all techniques was achieved by the GB model with hyperparameter tuning, whereas the DT model showed the lowest performance, with an AUC of 0.856 in both the baseline and the 4-feature subsets. With a limited number of features (2 features in this case), the GB model achieved the highest AUC of 0.943, while KNN obtained the lowest AUC of 0.869. This highlights that even with fewer features, competitive performance can be maintained. With the subset of two-features, DT appeared to be most cost effective model with 0.03 seconds execution time, while SVM once again appeared to be the most expensive model with 151.87 execution time.

Reducing the feature set to the most influential features identified via XAI techniques allows for more interpretable models and potentially faster computation without significant loss of predictive power. The results confirm that a small subset of features can adequately capture the risk patterns in the combined dataset with GB, AB and RF consistently performing the best across both feature sets.

Model	Accuracy	F1	Precision	Recall	ET (s)	AUC	CM
LR	0.951	0.709	0.847	0.61	0.12	0.958	(7389, 89) (315, 493)
RF	0.964	0.777	0.972	0.647	1.45	0.962	(7463, 15) (285, 523)
GB	0.964	0.775	0.981	0.64	1.98	0.976	(7468, 10) (291, 517)
AB	0.964	0.774	1.000	0.631	0.47	0.970	(7478, 0) (298, 510)
DT	0.943	0.710	0.703	0.717	0.07	0.842	(7233, 245) (229, 579)
SVM	0.954	0.697	0.959	0.547	24.54	0.876	(7459, 19) (366, 442)
KNN	0.944	0.647	0.843	0.525	2.08	0.875	(7399, 79) (384, 424)
NN	0.960	0.754	0.941	0.629	1.75	0.965	(7446, 32) (300, 508)

Table 3.1: Baseline classification performance for male group using ML models across different metrics; Accuracy, Precision, Recall, F1 score, AUC and CM. Execution time for each model (in seconds) is provided in the ET (s) column.

Model	Accuracy	F1	Precision	Recall	ET (s)	AUC	CM
LR	0.965	0.730	0.880	0.624	0.03	0.963	(10743, 76) (335, 557)
RF	0.975	0.809	0.978	0.689	1.94	0.960	(10805, 14) (277, 615)
GB	0.976	0.814	0.986	0.693	2.60	0.979	(10810, 9) (274, 618)
AB	0.975	0.807	1.000	0.677	0.65	0.973	(10819, 0) (288, 604)
DT	0.957	0.724	0.709	0.739	0.10	0.857	(10549, 270) (233, 659)
SVM	0.967	0.724	0.973	0.576	43.19	0.876	(10805, 14) (378, 514)
KNN	0.961	0.687	0.897	0.556	2.64	0.887	(10762, 57) (396, 496)
NN	0.973	0.798	0.945	0.691	2.47	0.976	(10783, 36) (276, 616)

Table 3.2: Baseline classification performance for female group using ML models across different metrics; Accuracy, Precision, Recall, F1 score, AUC and CM. Execution time for each model (in seconds) is provided in the ET (s) column.

Model	Accuracy	F1	Precision	Recall	ET (s)	AUC	CM
LR	0.961	0.733	0.870	0.633	0.06	0.961	(18139, 161) (624, 1076)
RF	0.972	0.804	0.968	0.687	3.46	0.960	(18262, 38) (532, 1168)
GB	0.973	0.810	0.985	0.688	4.43	0.978	(18282, 18) (531, 1169)
AB	0.972	0.805	1.000	0.674	1.18	0.971	(18300, 0) (554, 1146)
DT	0.951	0.719	0.698	0.741	0.17	0.856	(17756, 544) (440, 1260)
SVM	0.964	0.733	0.971	0.588	197.42	0.892	(18270, 30) (700, 1000)
KNN	0.958	0.702	0.888	0.581	4.90	0.896	(18175, 125) (712, 988)
NN	0.972	0.805	0.984	0.681	4.40	0.976	(18281, 19) (543, 1157)

Table 3.3: Baseline classification performance for combined group using ML models across different metrics; Accuracy, Precision, Recall, F1 score, AUC and CM. Execution time for each model (in seconds) is provided in the ET(s) column.

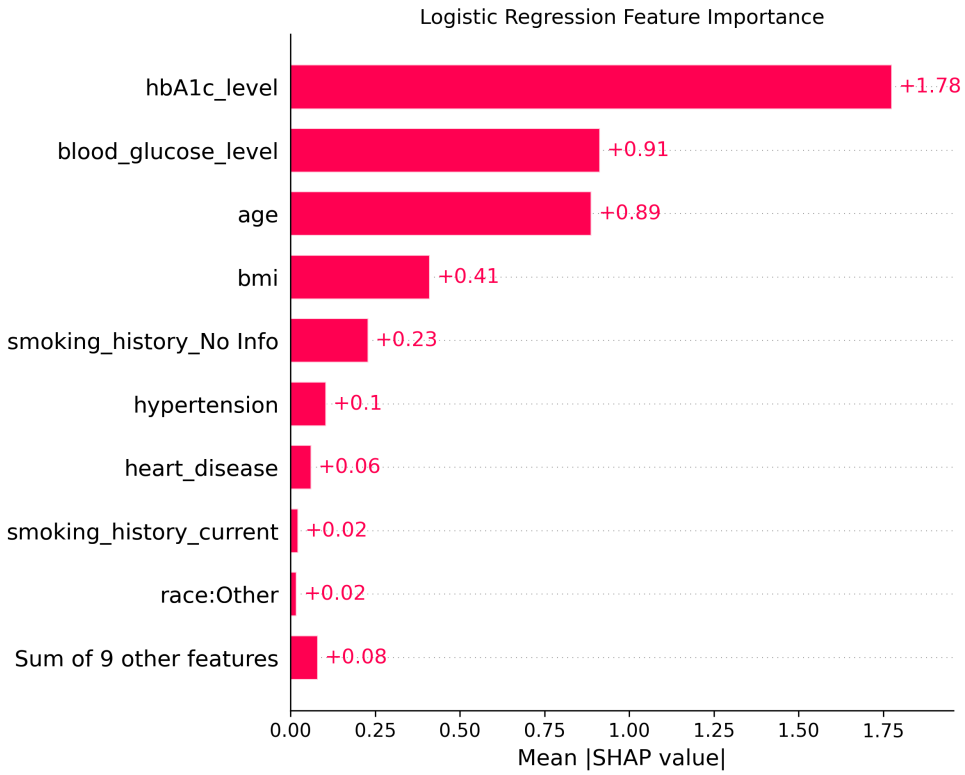

Figure 3.1: Mean absolute SHAP value bar plot showing the global feature importance using the LR model.

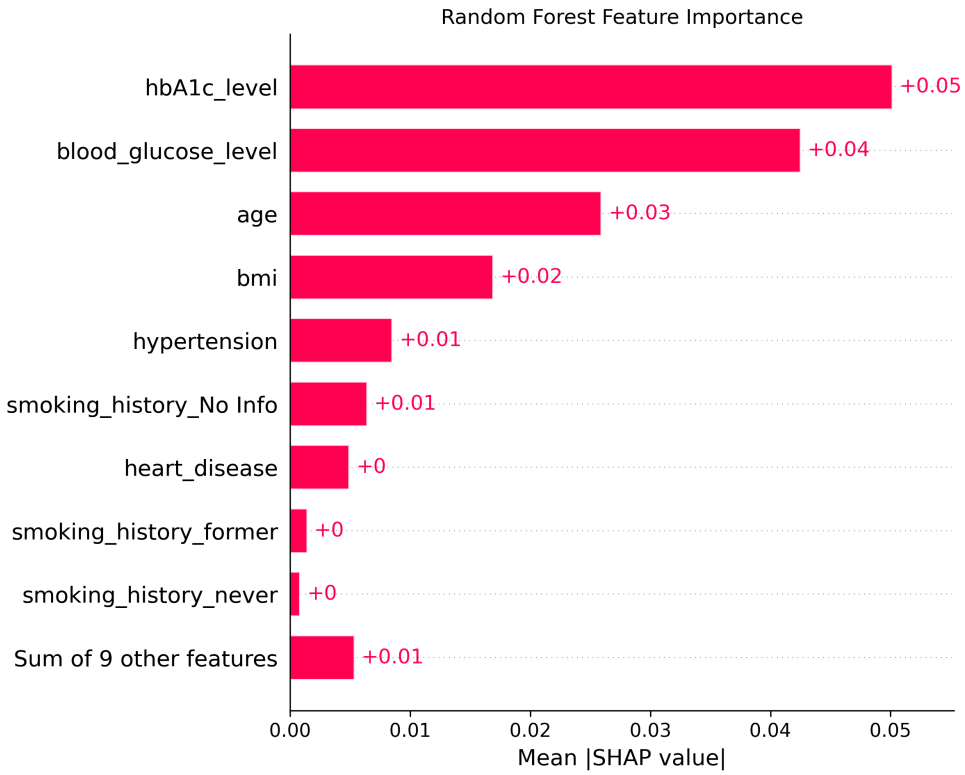

Figure 3.2: Mean absolute SHAP value bar plot showing the global feature importance using the RF model.

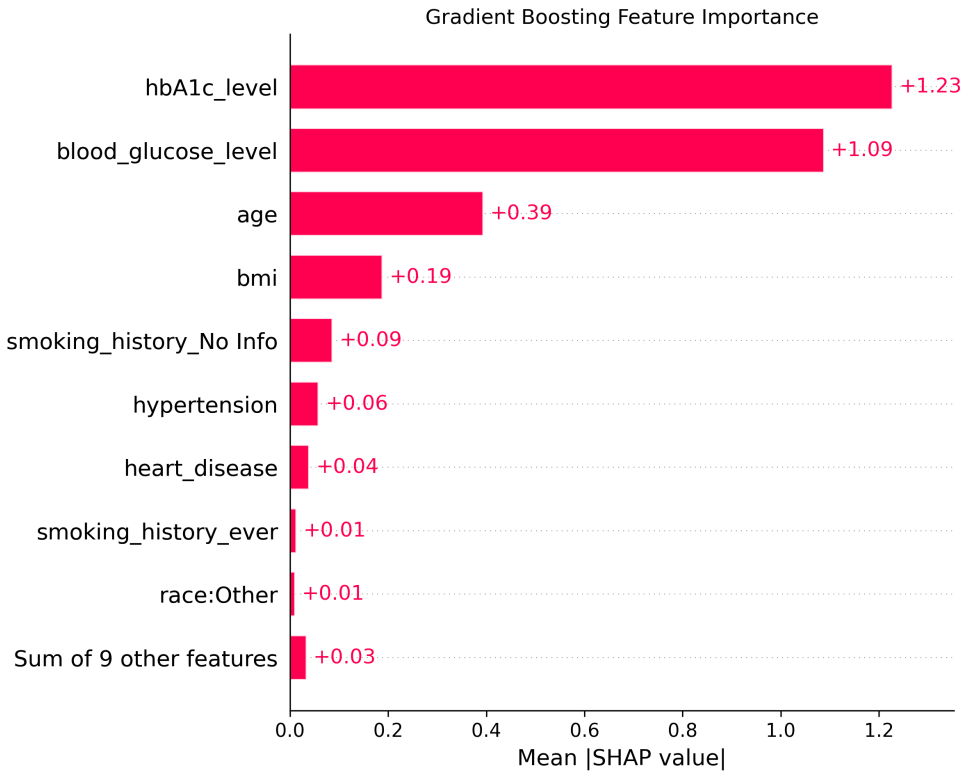

Figure 3.3: Mean absolute SHAP value bar plot showing the global feature importance using the GB model.

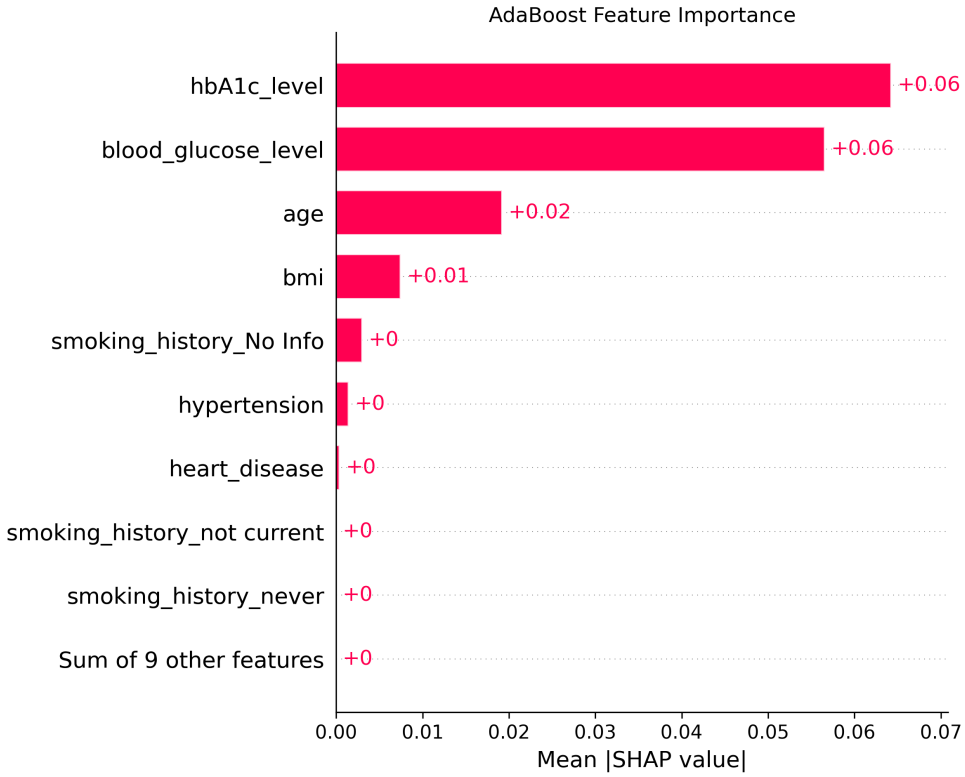

Figure 3.4: Mean absolute SHAP value bar plot showing the global feature importance using the AB model.

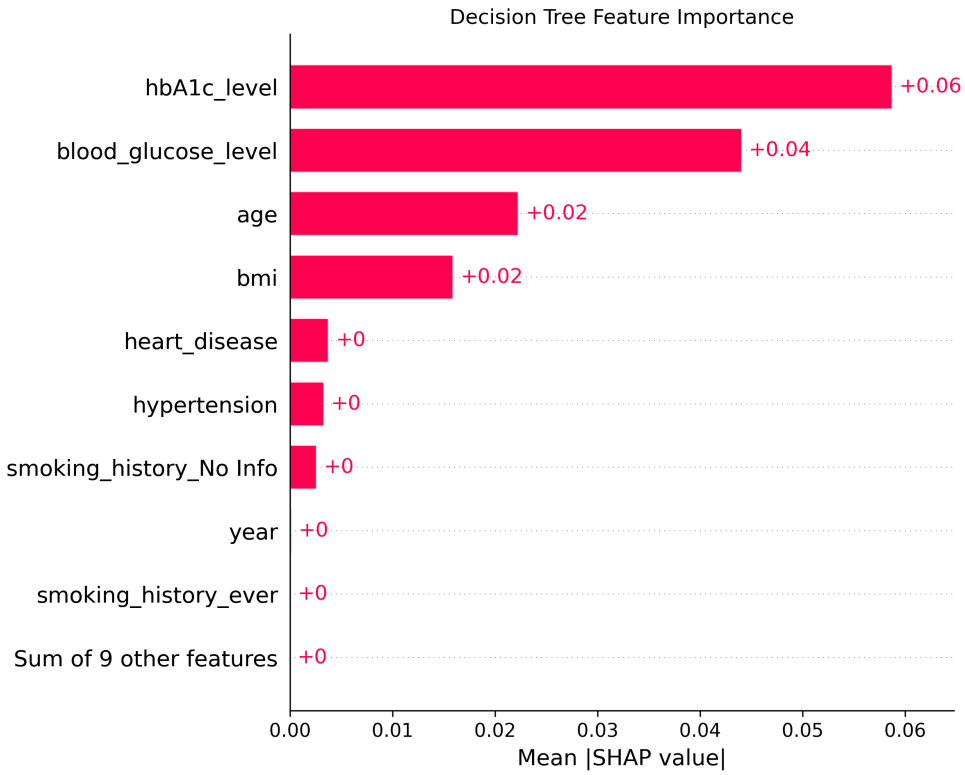

Figure 3.5: Mean absolute SHAP value bar plot showing the global feature importance using the DT model.

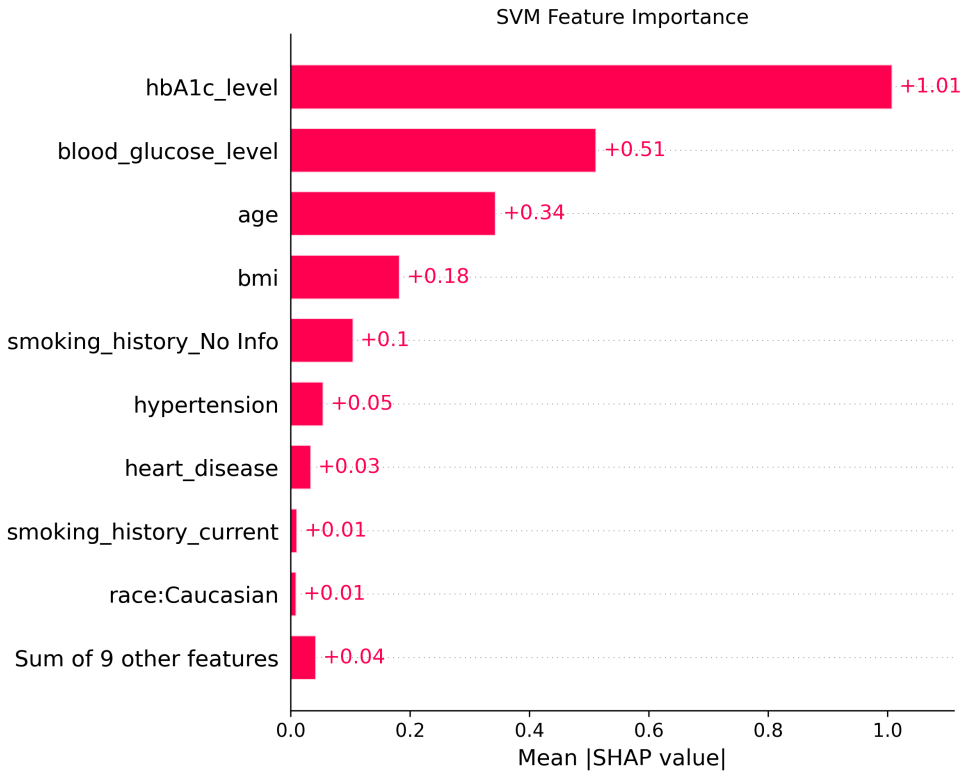

Figure 3.6: Mean absolute SHAP value bar plot showing the global feature importance using the SVM model.

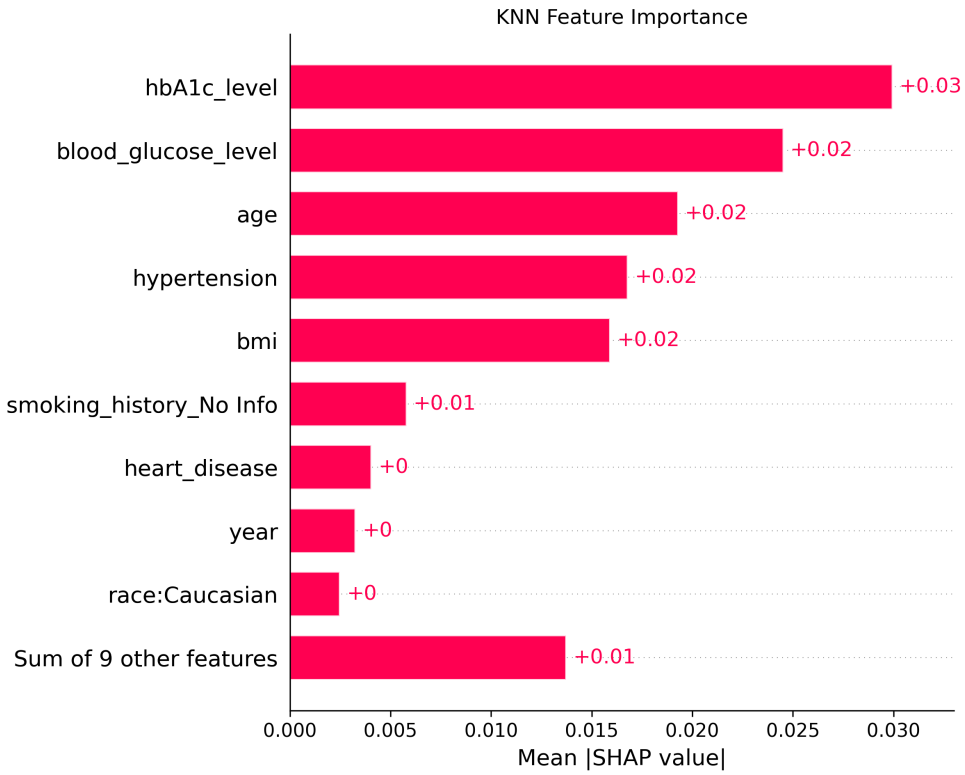

Figure 3.7: Mean absolute SHAP value bar plot showing the global feature importance using the KNN model.

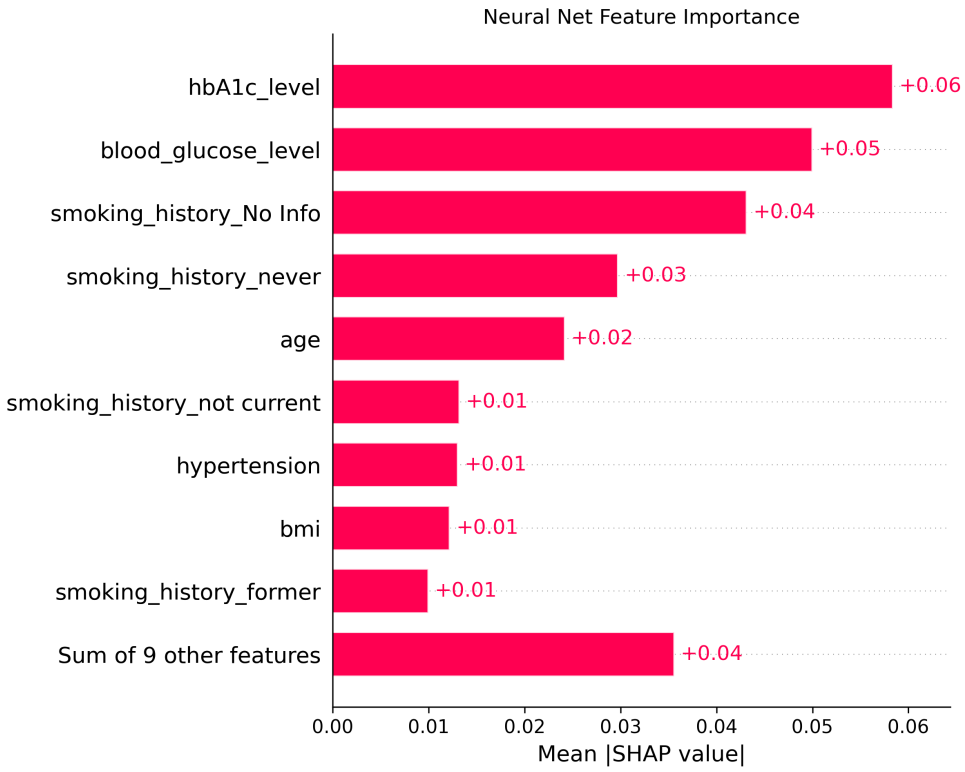

Figure 3.8: Mean absolute SHAP value bar plot showing the global feature importance using the NN model.

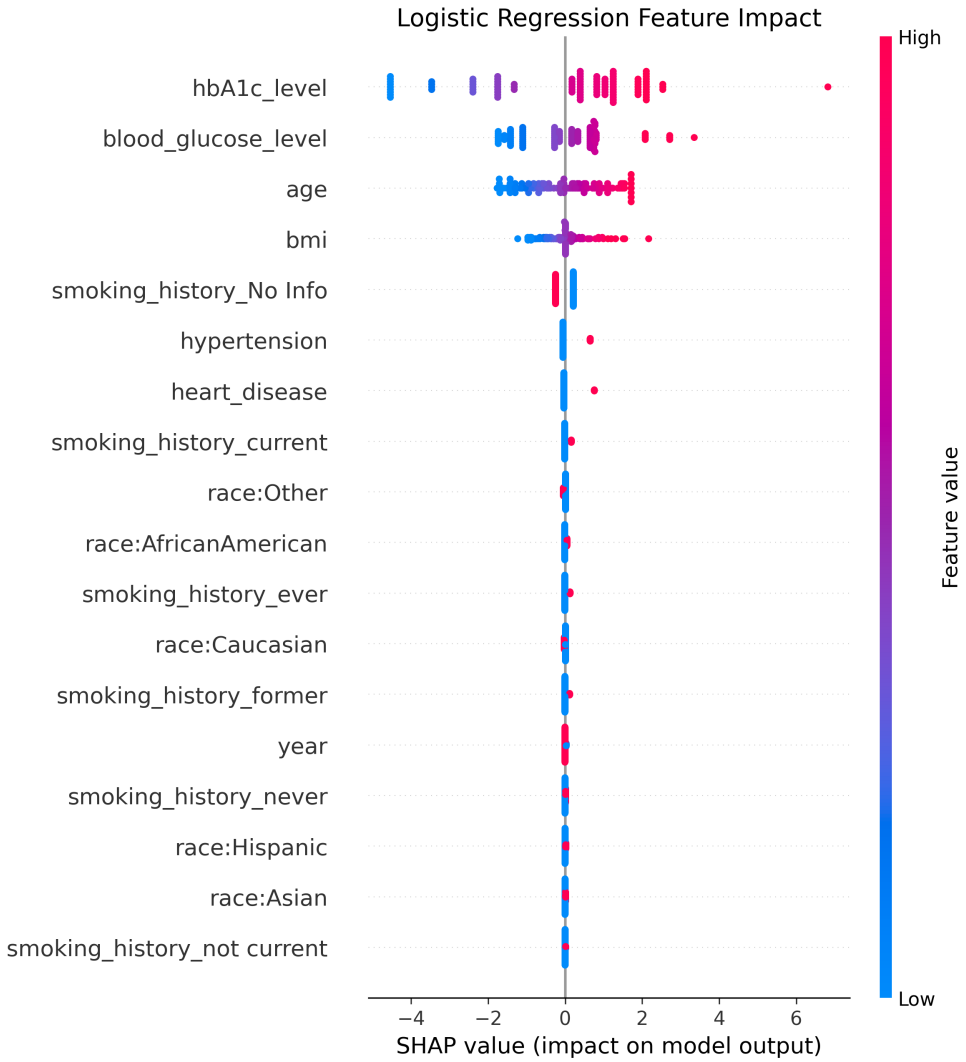

Figure 3.9: SHAP summary plot showing the feature impact on the LR model predictions.

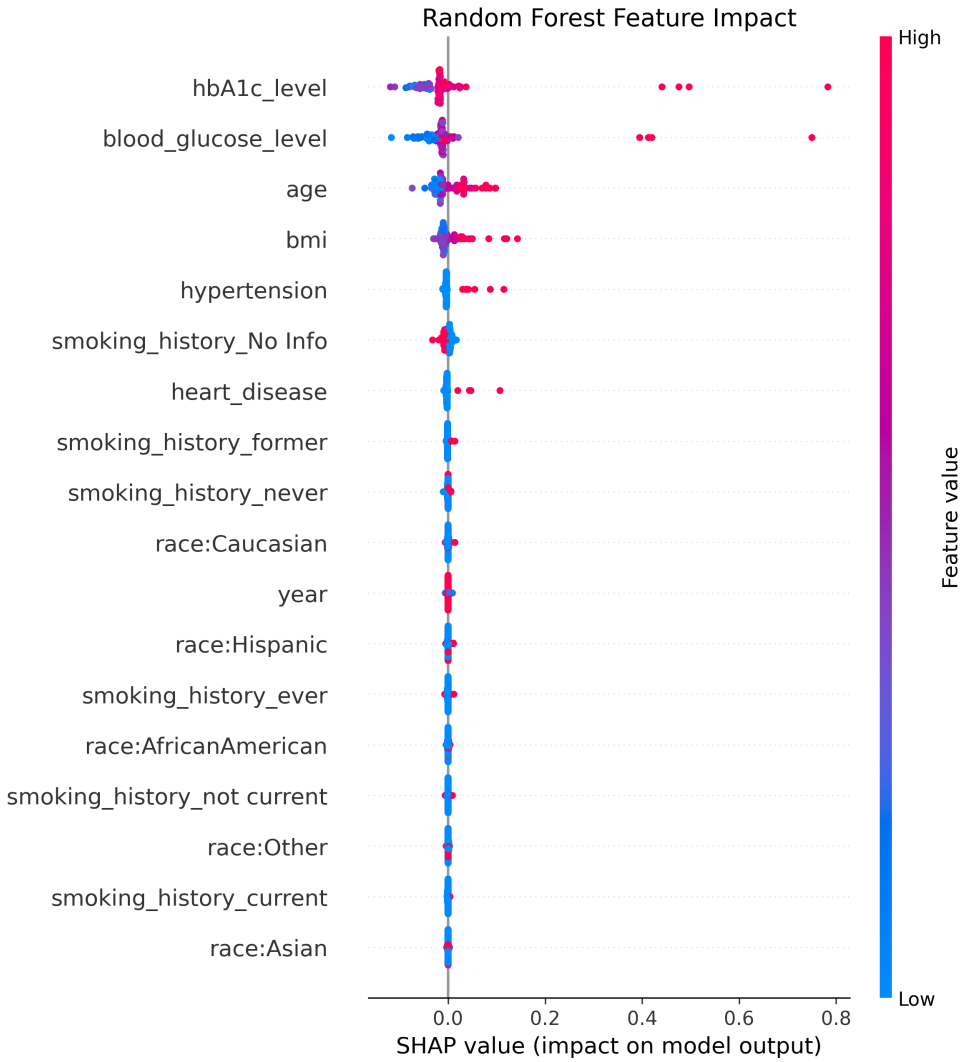

Figure 3.10: SHAP summary plot showing the feature impact on the RF model predictions.

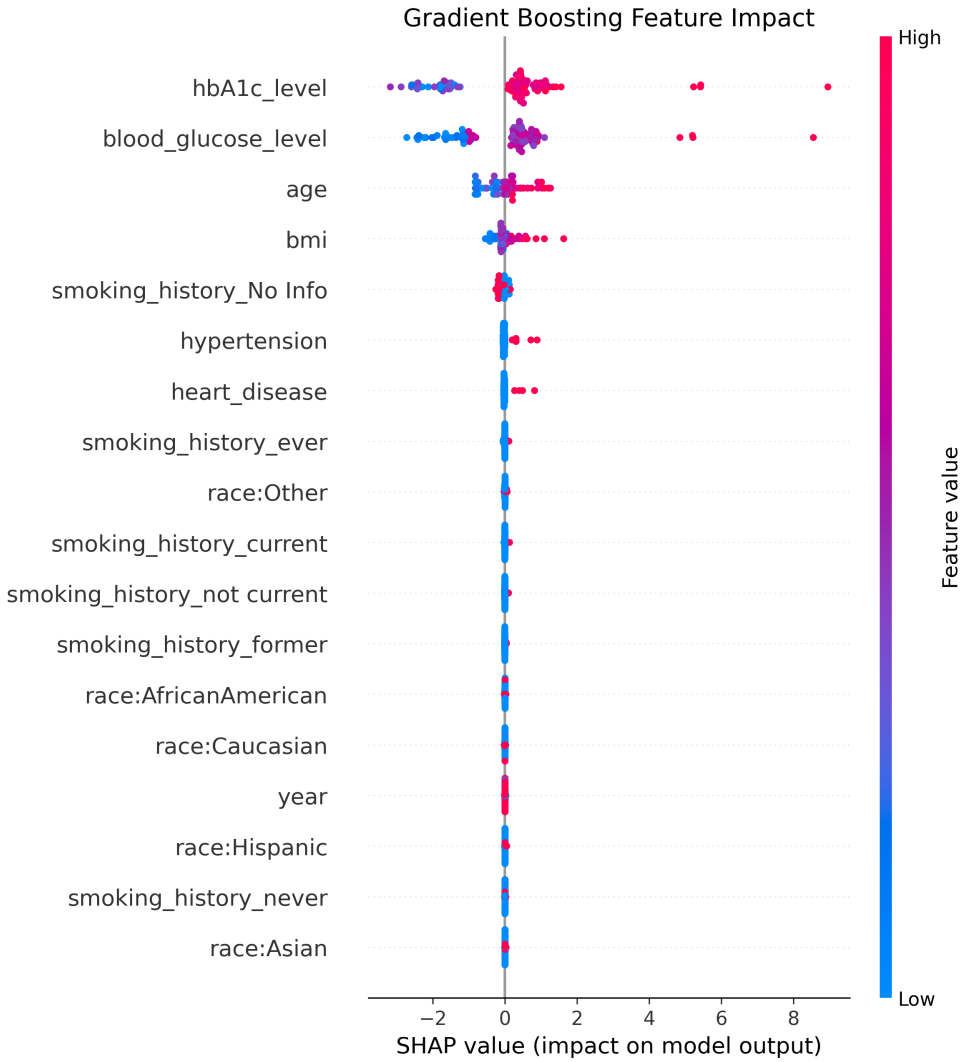

Figure 3.11: SHAP summary plot showing the feature impact on the GB model predictions.

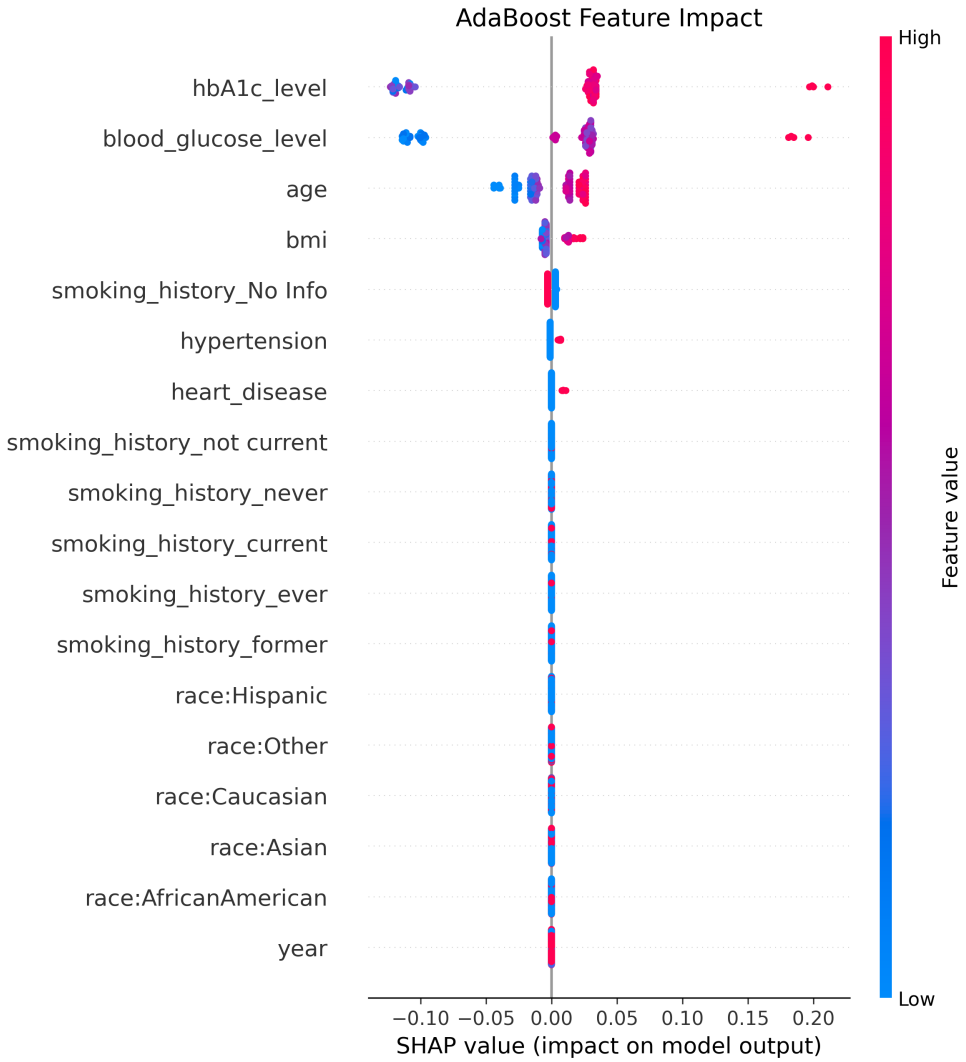

Figure 3.12: SHAP summary plot showing the feature impact on the AB model predictions.

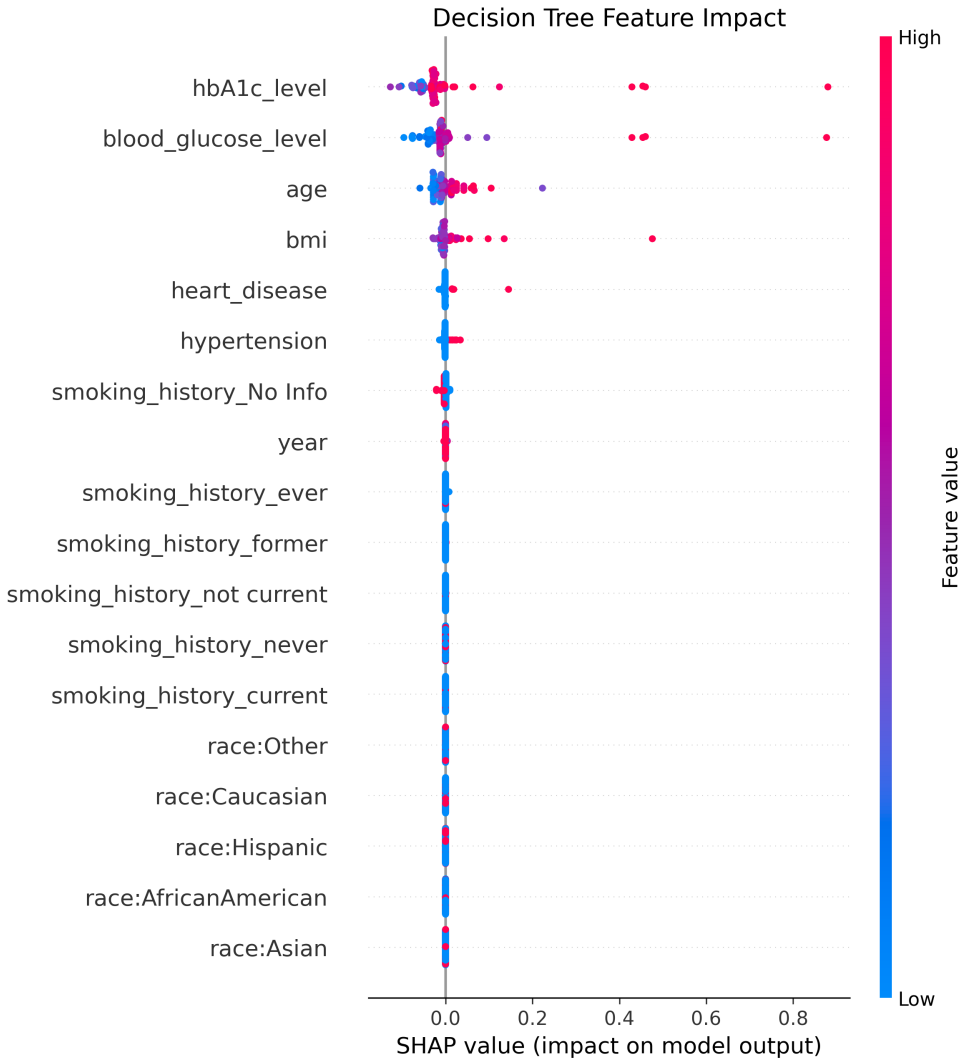

Figure 3.13: SHAP summary plot showing the feature impact on the DT model predictions.

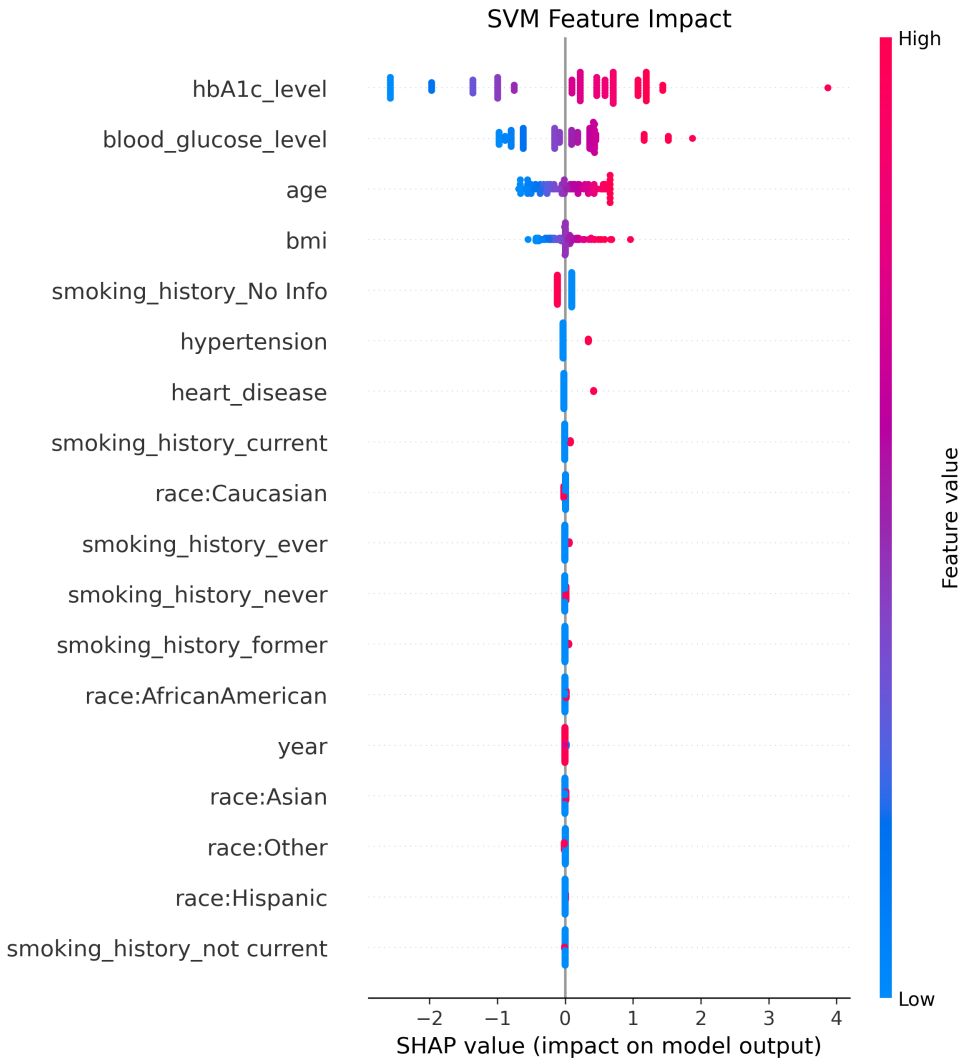

Figure 3.14: SHAP summary plot showing the feature impact on the SVM model predictions.

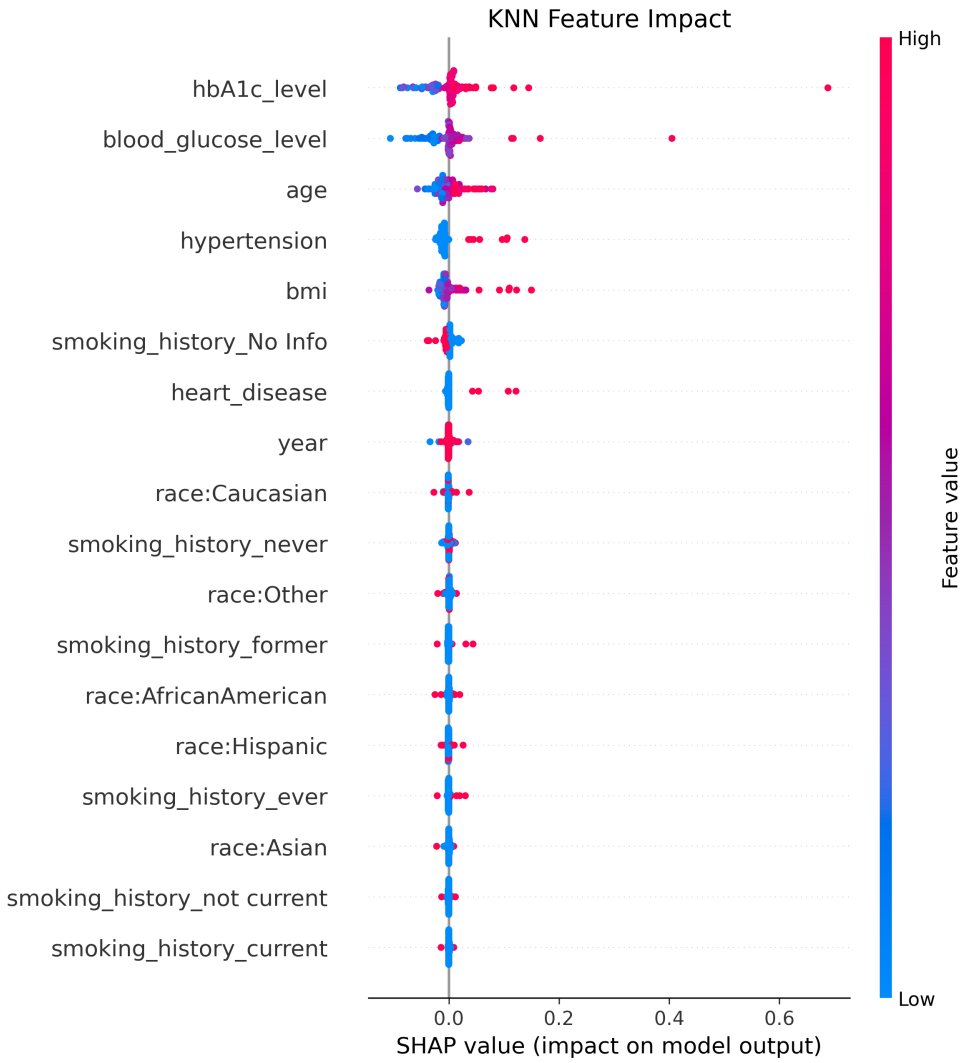

Figure 3.15: SHAP summary plot showing the feature impact on the KNN model predictions.

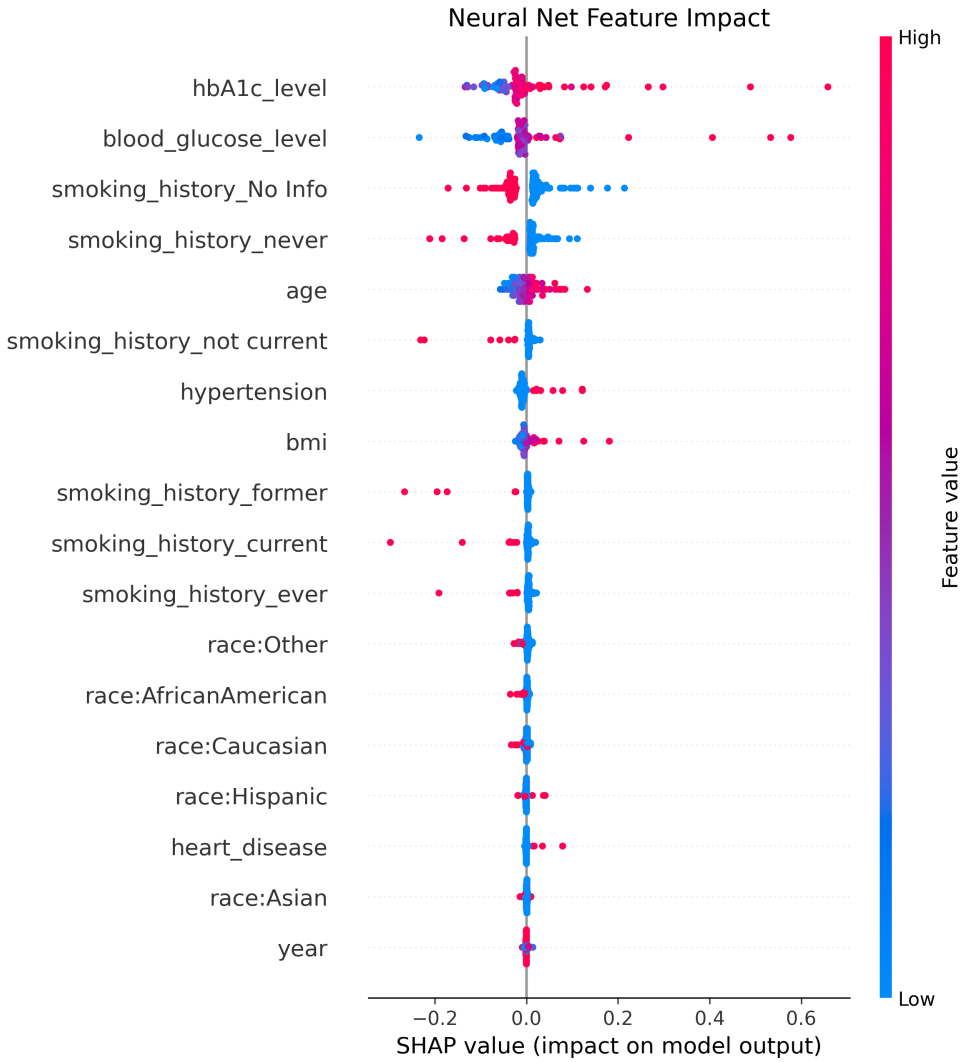

Figure 3.16: SHAP summary plot showing the feature impact on the NN model predictions.

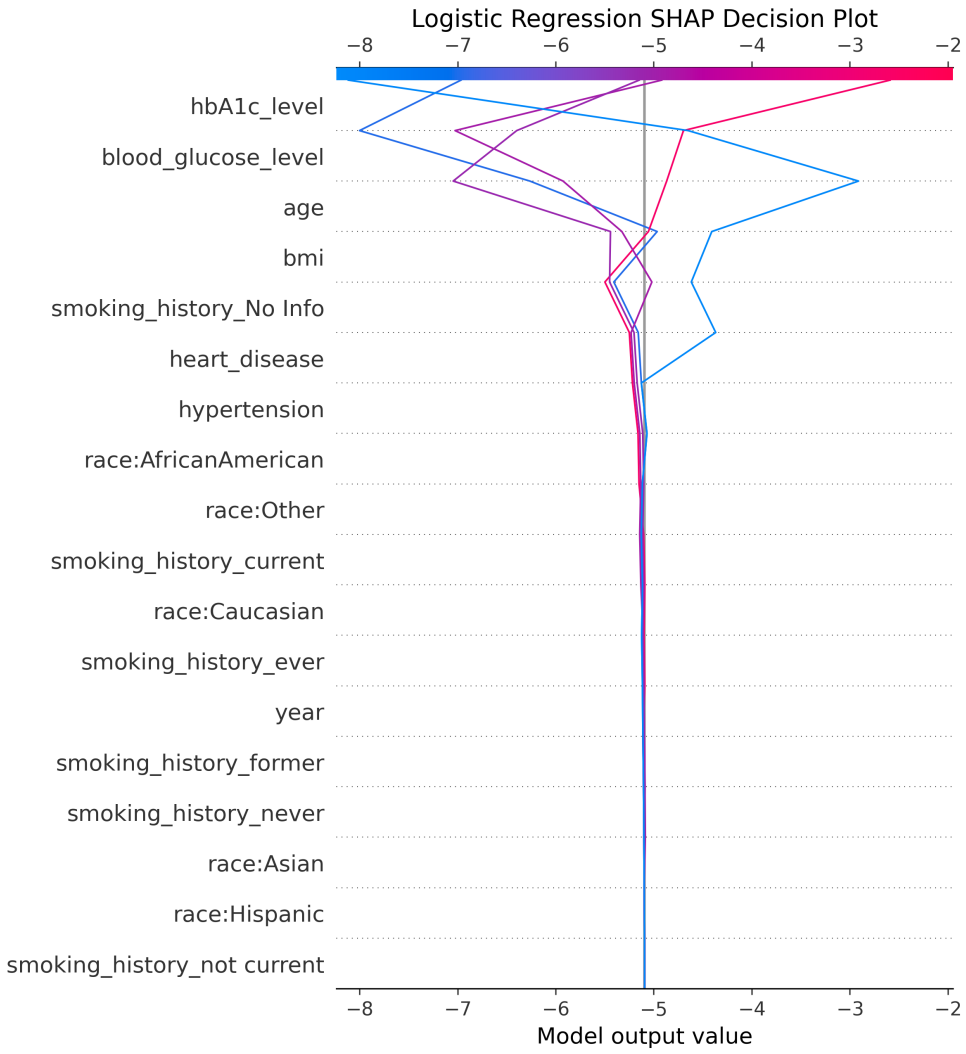

Figure 3.17: SHAP decision plot illustrating the cumulative contribution of individual features to the model output for selected samples using the LR model.

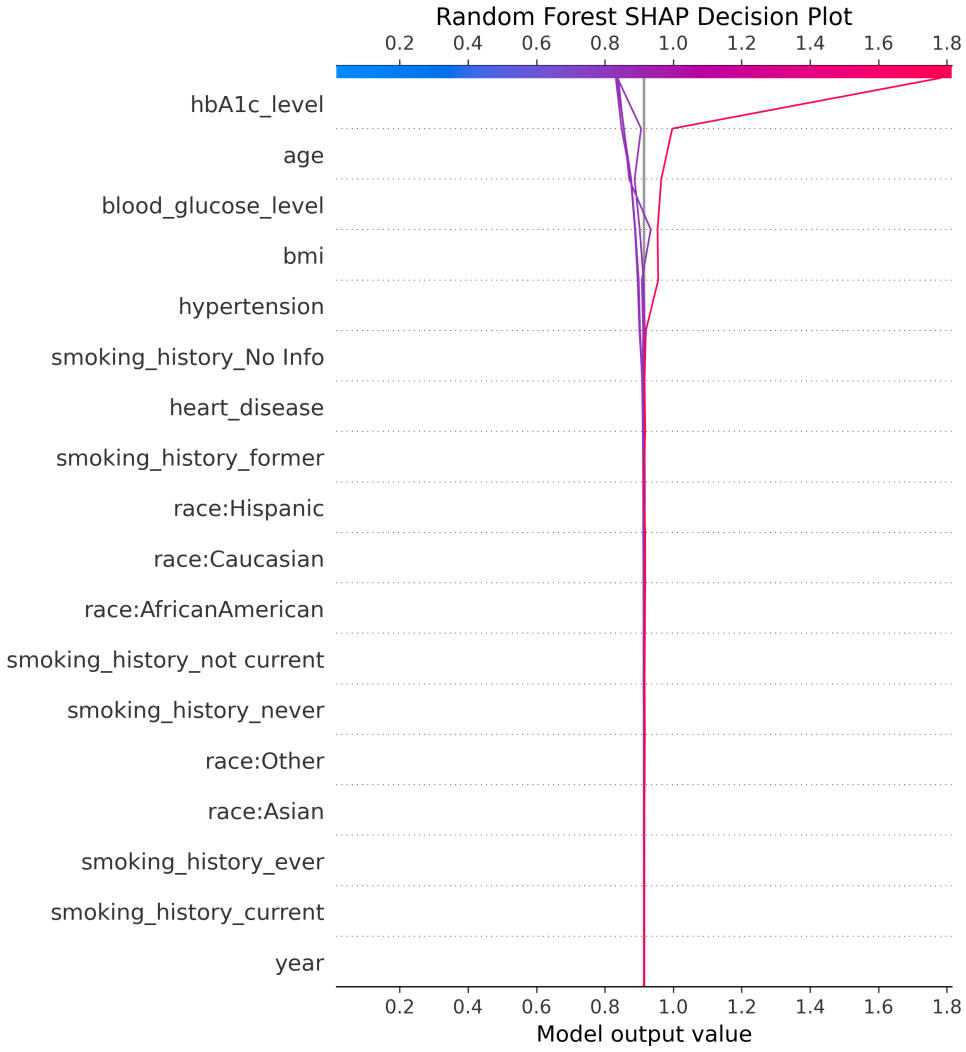

Figure 3.18: SHAP decision plot illustrating the cumulative contribution of individual features to the model output for selected samples using the RF model.

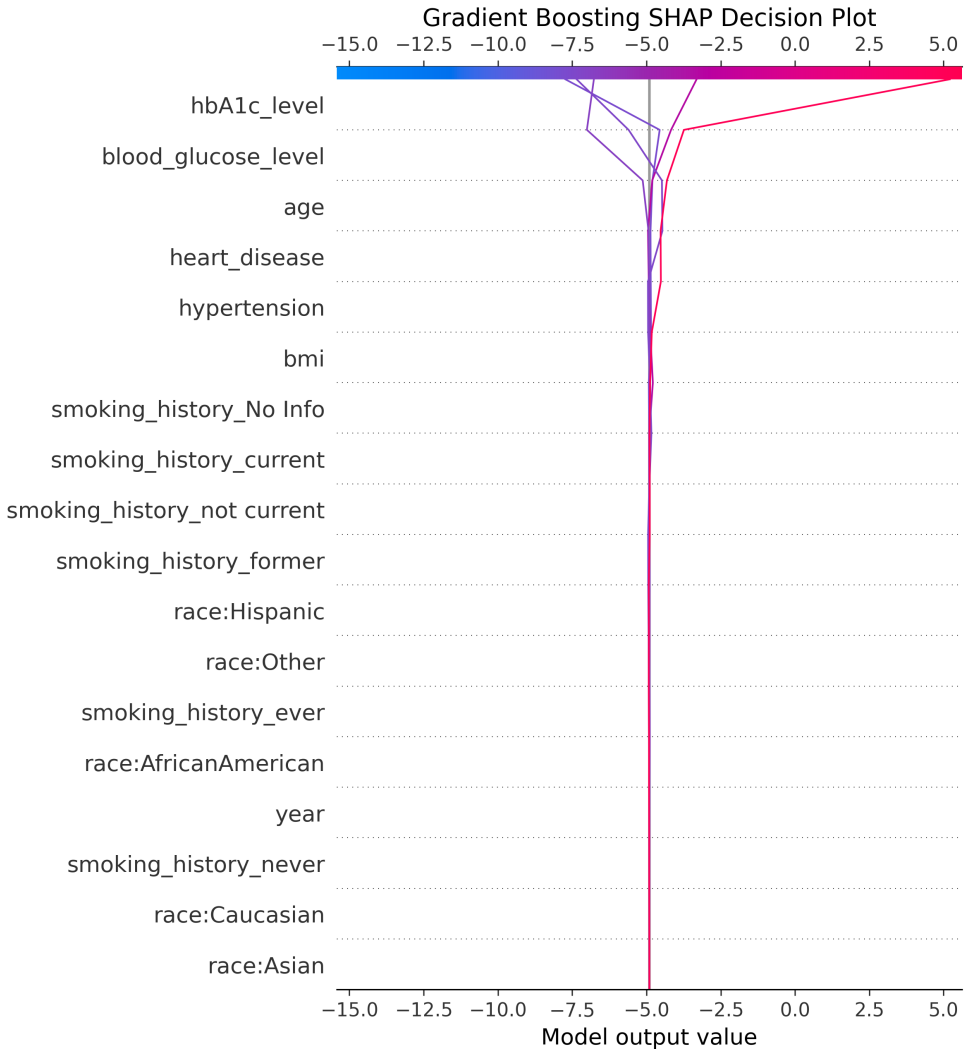

Figure 3.19: SHAP decision plot illustrating the cumulative contribution of individual features to the model output for selected samples using the GB model.

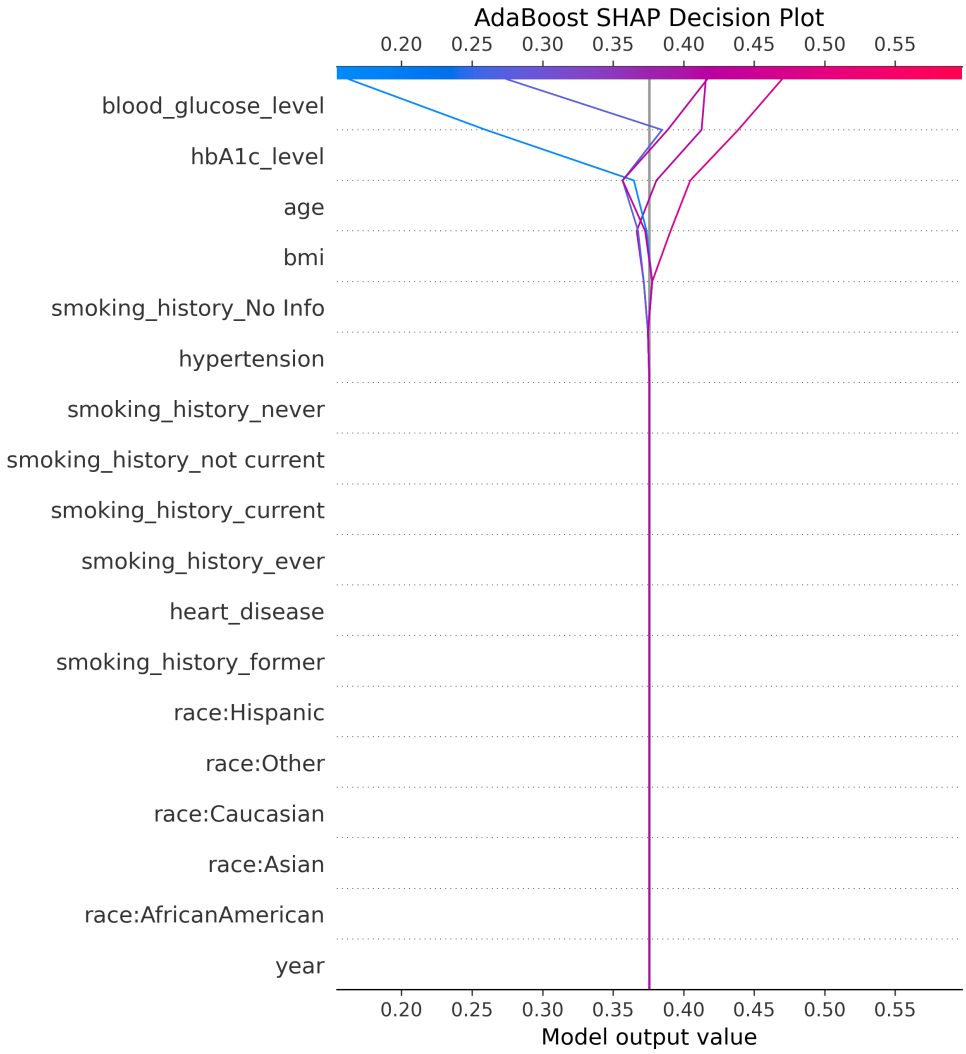

Figure 3.20: SHAP decision plot illustrating the cumulative contribution of individual features to the model output for selected samples using the AB model.

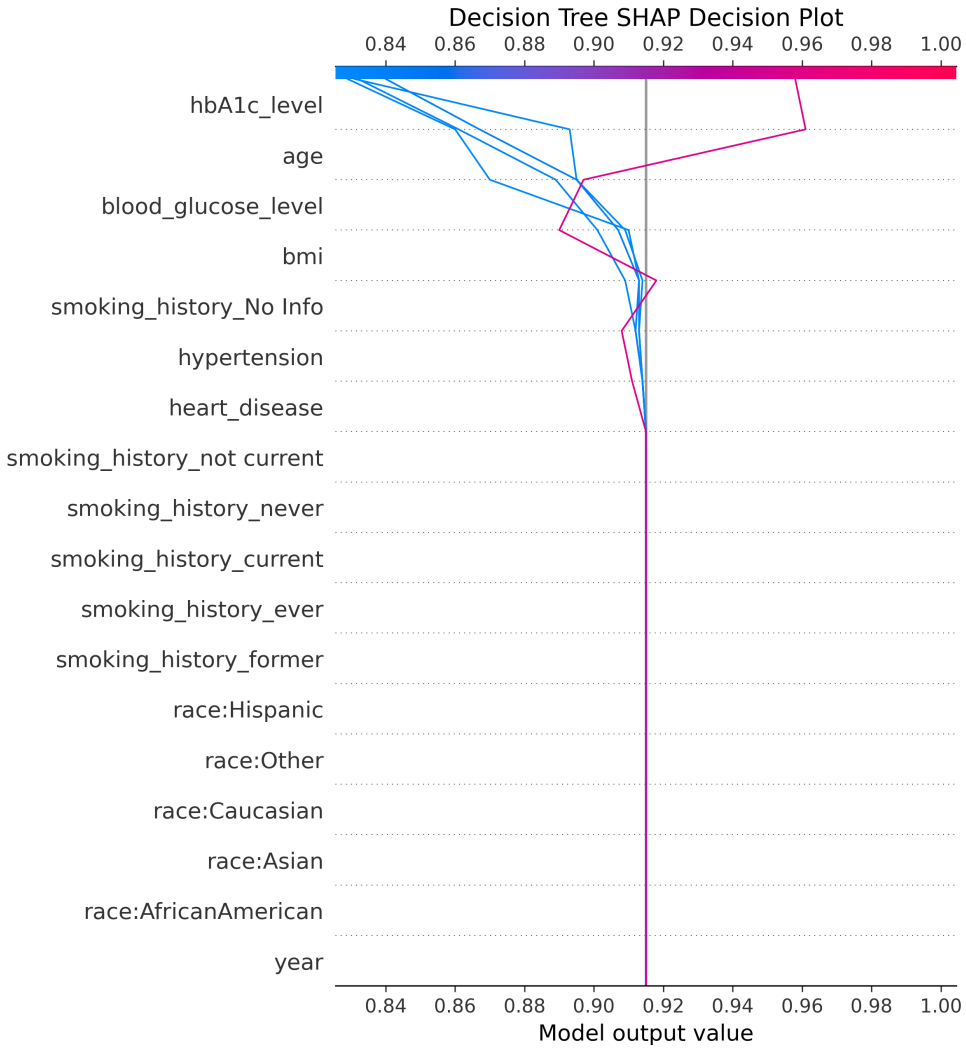

Figure 3.21: SHAP decision plot illustrating the cumulative contribution of individual features to the model output for selected samples using the DT model.

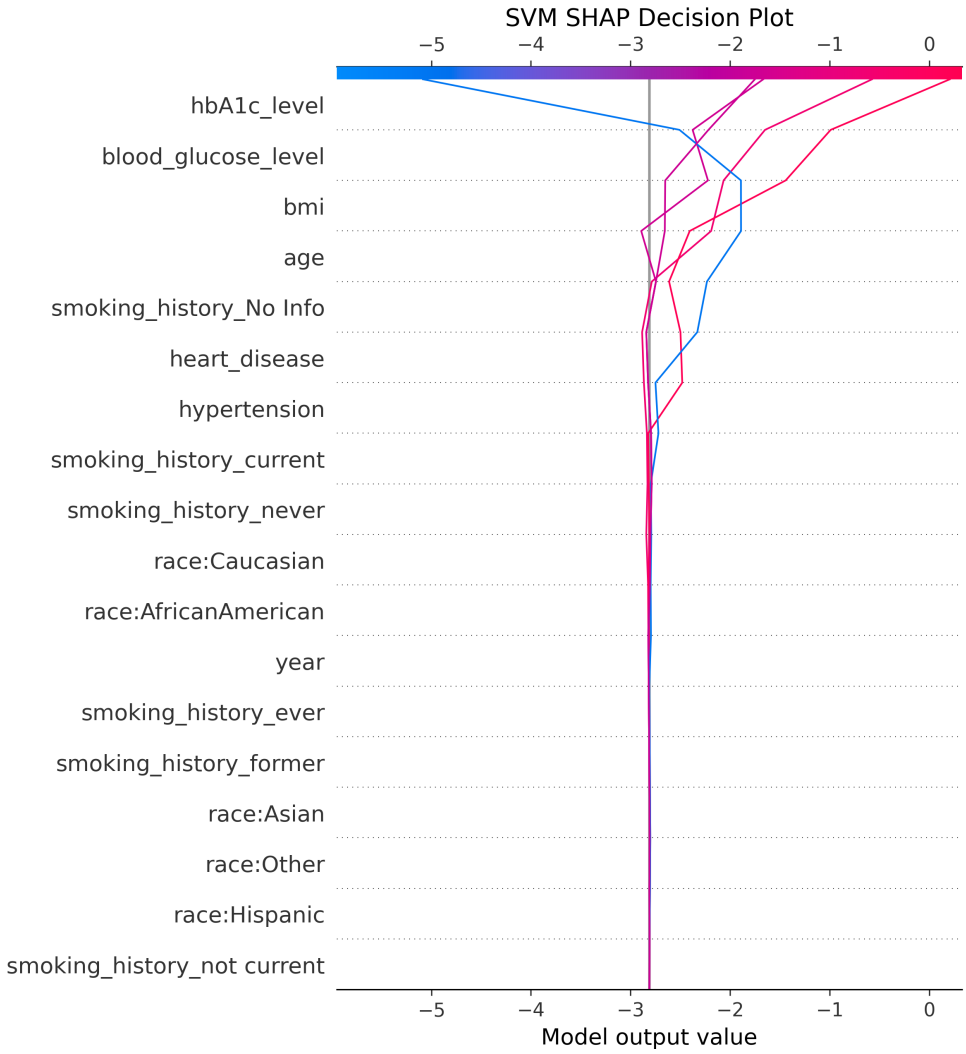

Figure 3.22: SHAP decision plot illustrating the cumulative contribution of individual features to the model output for selected samples using the SVM model.

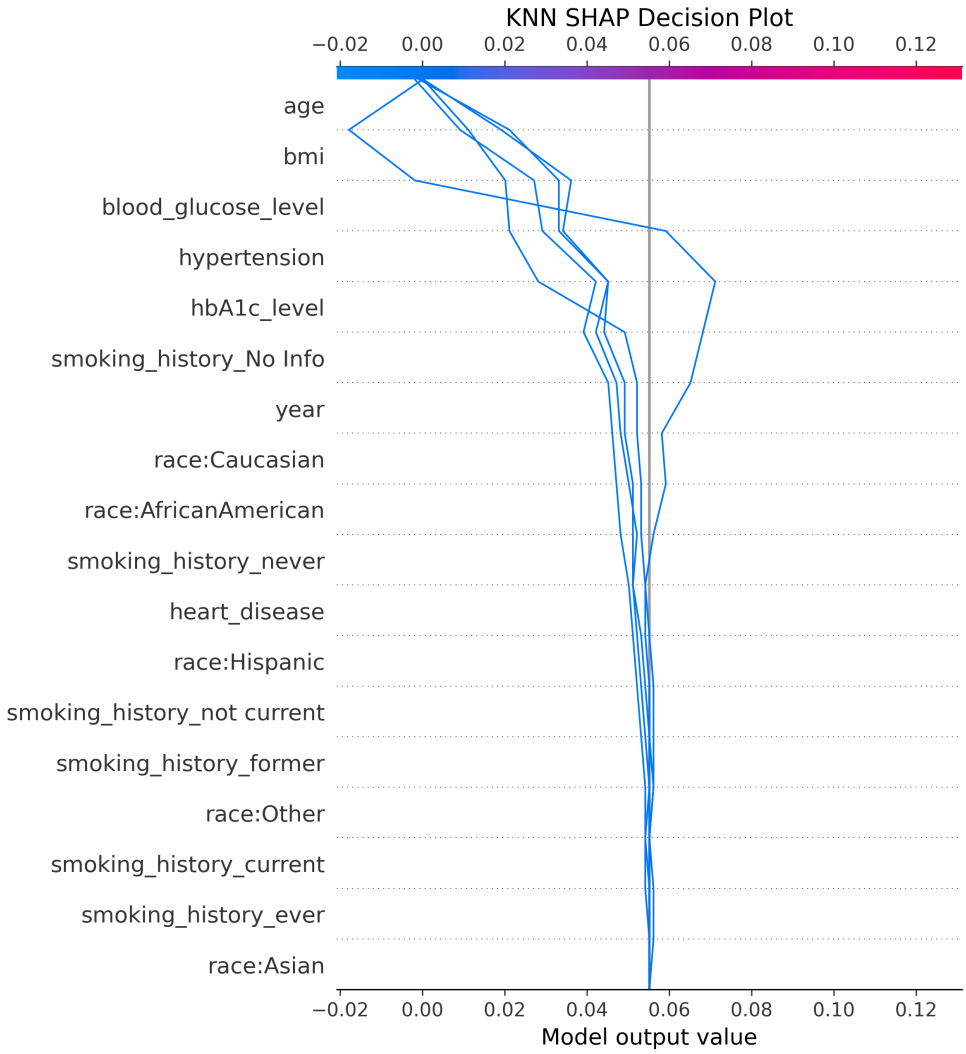

Figure 3.23: SHAP decision plot illustrating the cumulative contribution of individual features to the model output for selected samples using the KNN model.

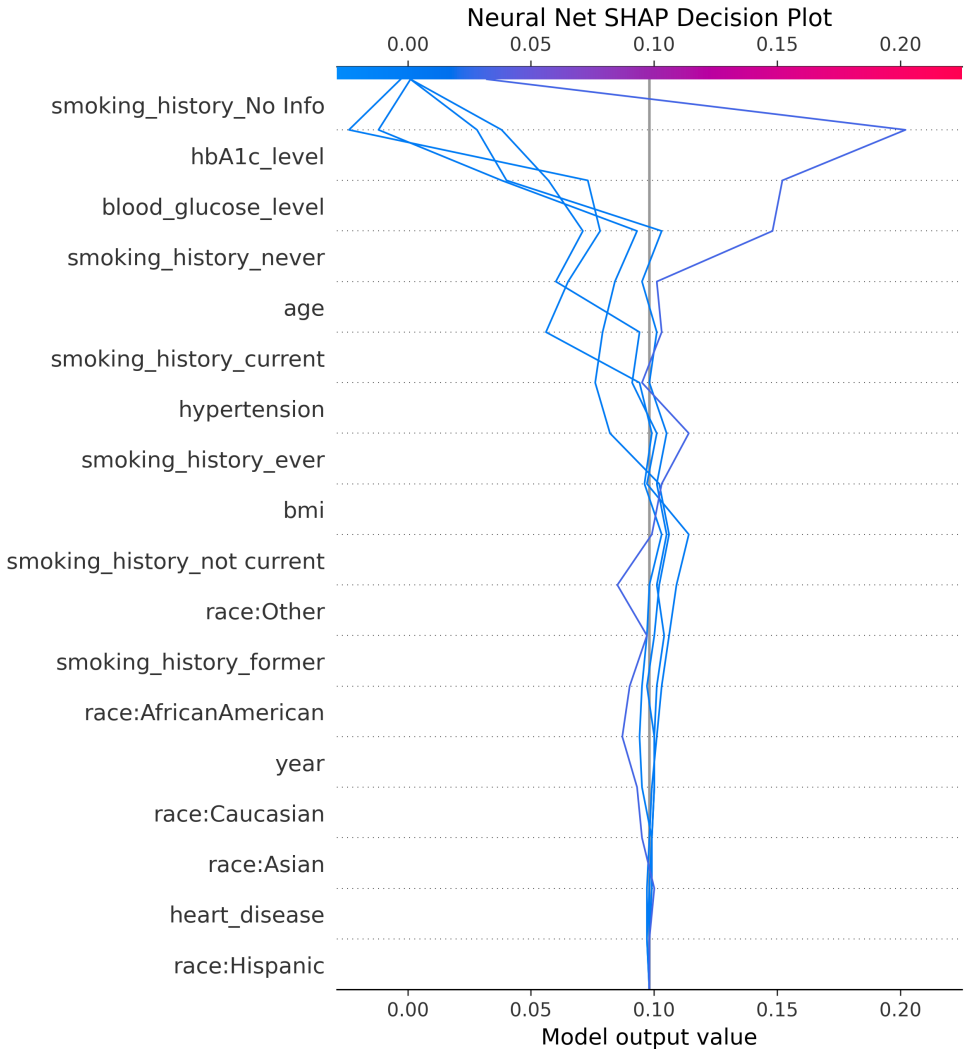

Figure 3.24: SHAP decision plot illustrating the cumulative contribution of individual features to the model output for selected samples using the NN model.

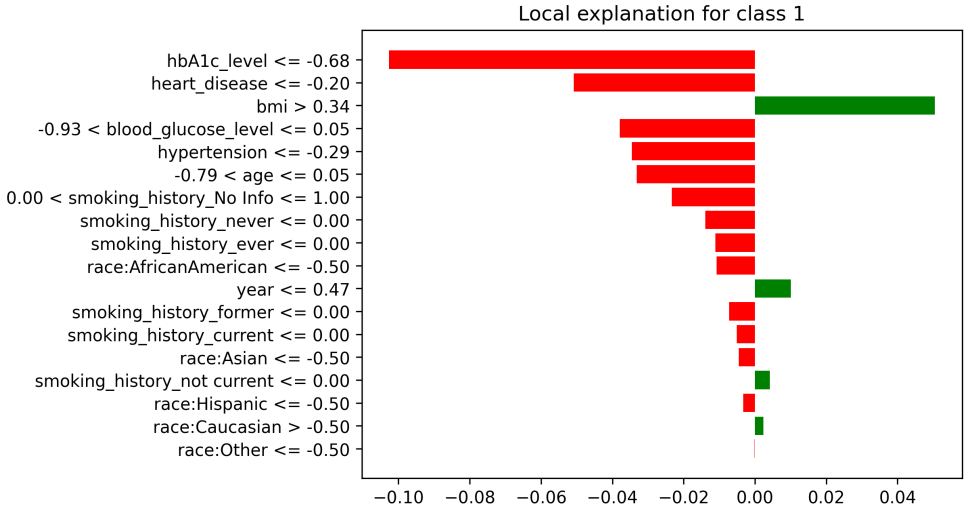

Figure 3.25: LIME plot for LR classifier: Local explanation illustrates how individual features contributed to the model's predicted probability.

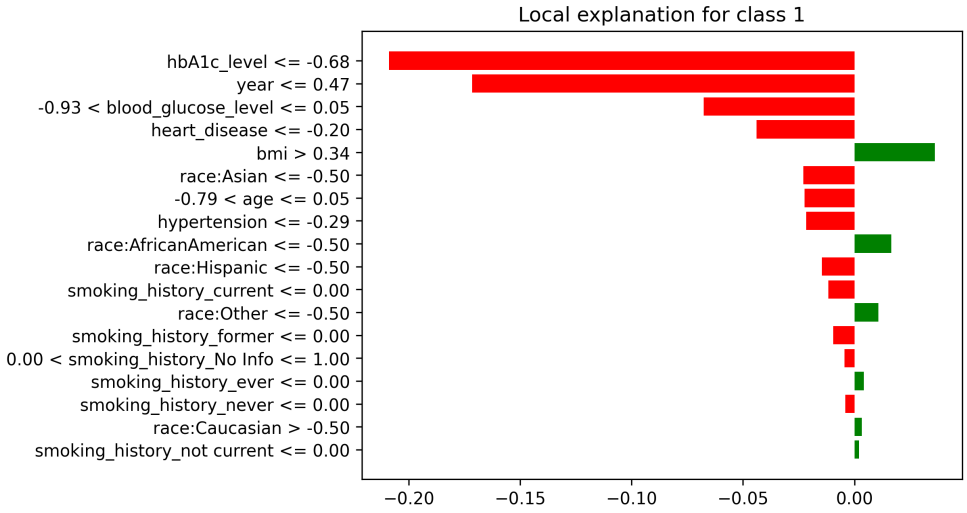

Figure 3.26: LIME plot for RF classifier: Local explanation illustrates how individual features contributed to the model's predicted probability.

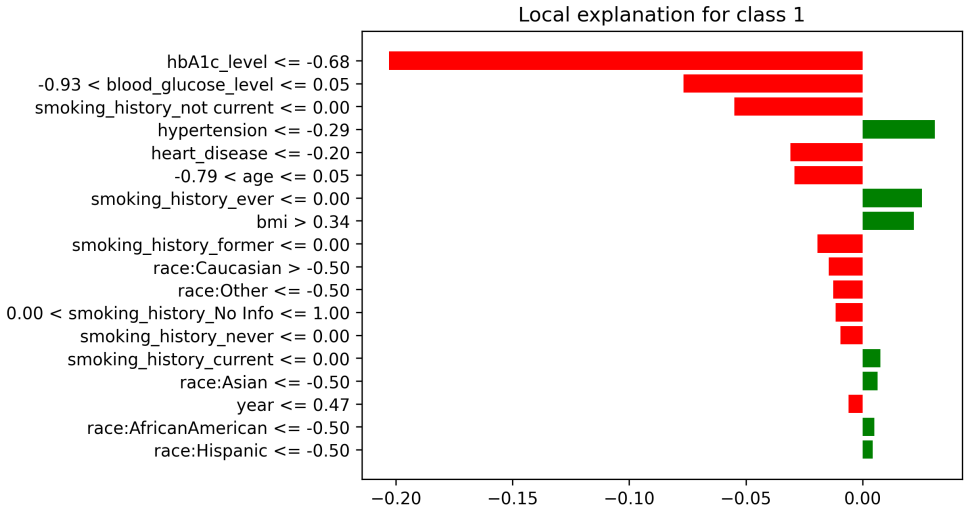

Figure 3.27: LIME plot for GB classifier: Local explanation illustrates how individual features contributed to the model's predicted probability.

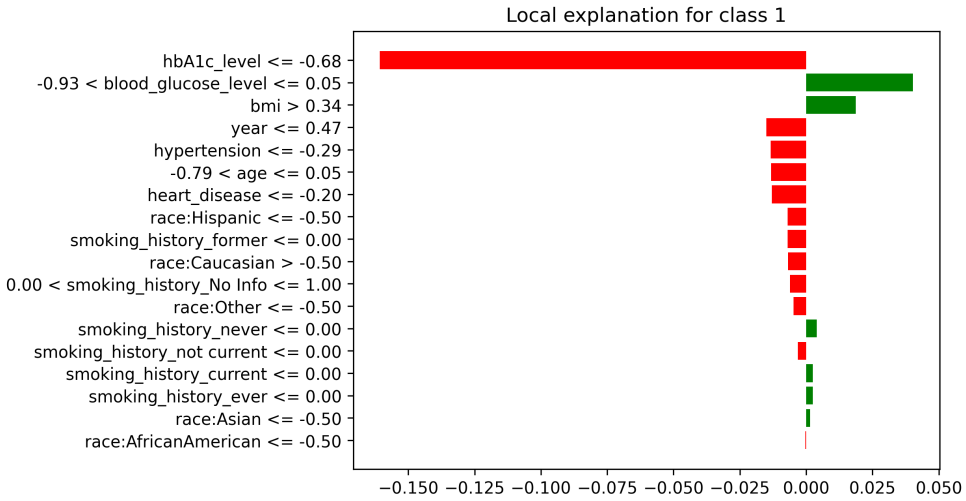

Figure 3.28: LIME plot for AB classifier: Local explanation illustrates how individual features contributed to the model's predicted probability.

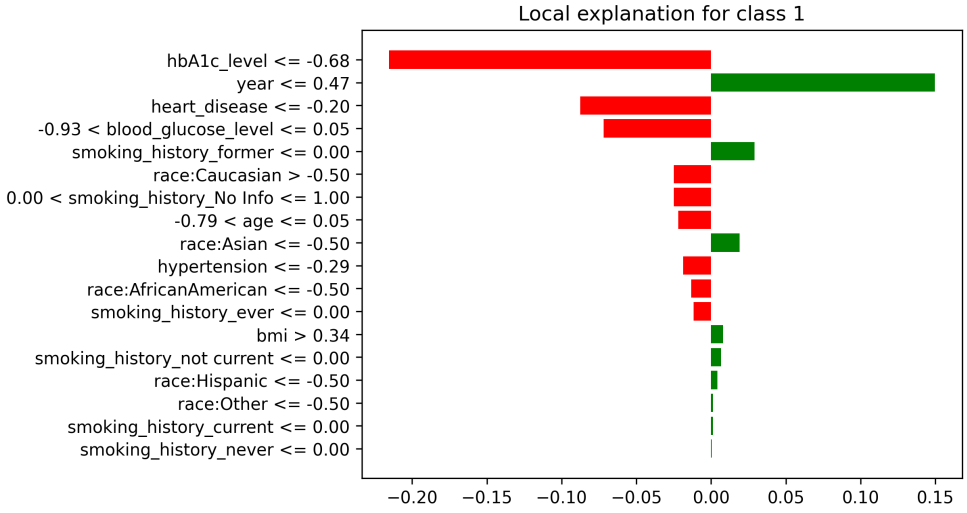

Figure 3.29: LIME plot for DT classifier: Local explanation illustrates how individual features contributed to the model's predicted probability.

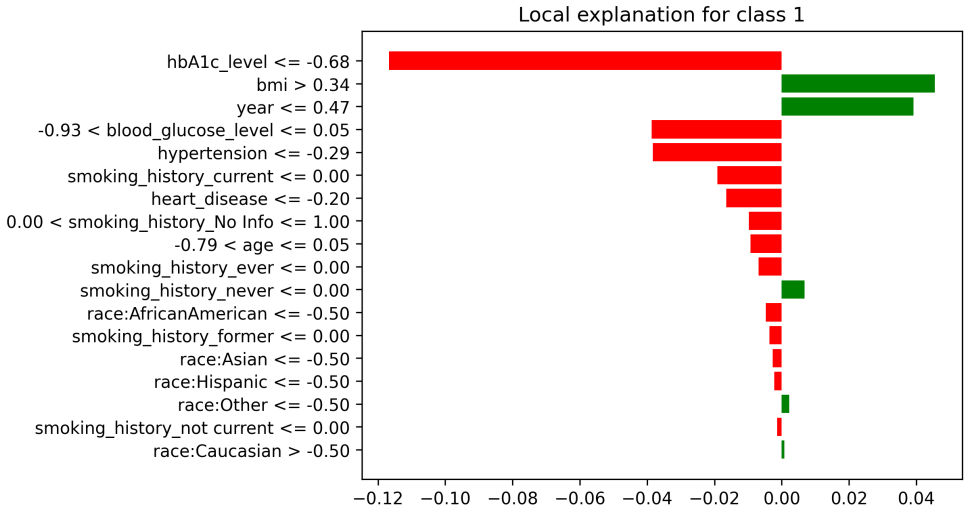

Figure 3.30: LIME plot for SVM classifier: Local explanation illustrates how individual features contributed to the model's predicted probability.

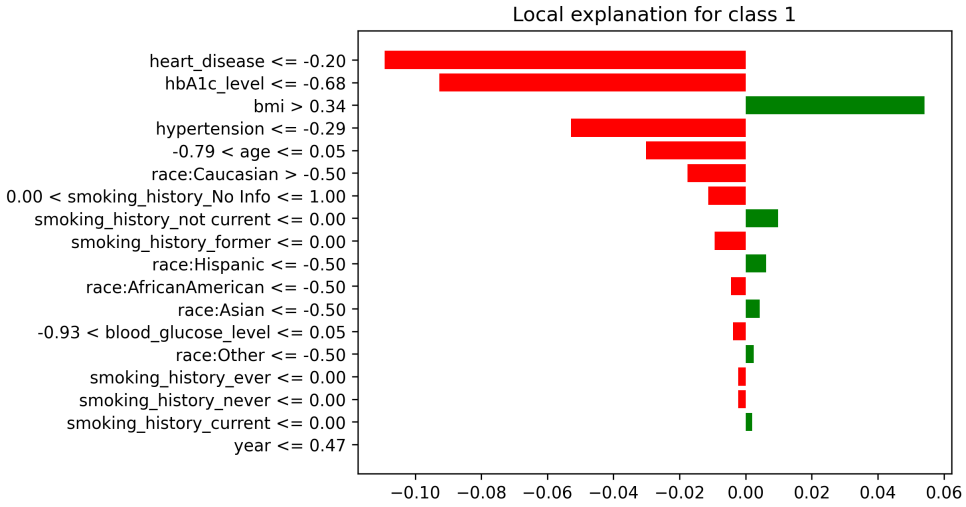

Figure 3.31: LIME plot for KNN classifier: Local explanation illustrates how individual features contributed to the model's predicted probability.

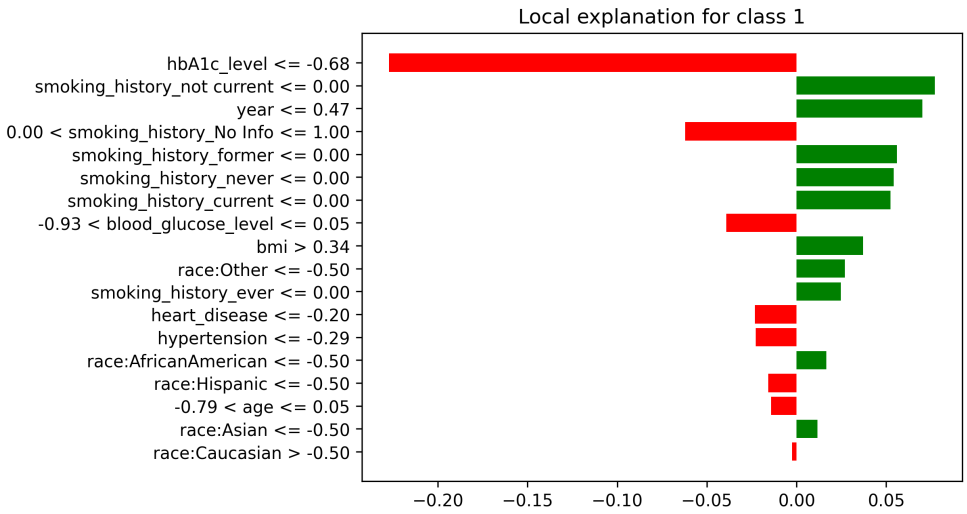

Figure 3.32: LIME plot for NN classifier: Local explanation illustrates how individual features contributed to the model's predicted probability.

Weight	Feature
0.0446 ± 0.0016	hbA1c_level
0.0299 ± 0.0004	blood_glucose_level
0.0067 ± 0.0010	age
0.0030 ± 0.0006	bmi
0.0004 ± 0.0003	heart_disease
0.0002 ± 0.0001	year
0.0002 ± 0.0001	race:Other
0.0001 ± 0.0004	hypertension
0.0001 ± 0.0009	smoking_history_No Info
0.0001 ± 0.0001	race:Asian
0.0001 ± 0.0002	smoking_history_current
0.0001 ± 0.0001	race:Hispanic
0.0000 ± 0.0001	race:Caucasian
0.0000 ± 0.0002	smoking_history_never
0.0000 ± 0.0001	smoking_history_not current
-0.0001 ± 0.0000	smoking_history_ever
-0.0001 ± 0.0001	race:AfricanAmerican
-0.0001 ± 0.0001	smoking_history_former

Figure 3.33: Permutation importance of features for the LR model as computed by ELI5.

Weight	Feature
0.0578 ± 0.0007	hbA1c_level
0.0467 ± 0.0011	blood_glucose_level
0.0003 ± 0.0002	bmi
0.0001 ± 0.0002	age
0.0001 ± 0.0001	heart_disease
0.0000 ± 0.0001	hypertension
0.0000 ± 0.0001	race:Hispanic
0.0000 ± 0.0000	race:Caucasian
0.0000 ± 0.0001	race:Asian
0.0000 ± 0.0001	smoking_history_not current
0.0000 ± 0.0001	smoking_history_never
0.0000 ± 0.0001	smoking_history_ever
-0.0000 ± 0.0001	smoking_history_current
-0.0000 ± 0.0001	race:AfricanAmerican
-0.0000 ± 0.0001	year
-0.0001 ± 0.0001	smoking_history_No Info
-0.0001 ± 0.0001	smoking_history_former
-0.0001 ± 0.0001	race:Other

Figure 3.34: Permutation importance of features for the RF model as computed by ELI5.

Model	Accuracy	F1	Precision	Recall	ET (s)	AUC	CM
LR	0.951	0.947	0.948	0.951	0.01	0.958	(7404, 74) (330, 478)
RF	0.964	0.960	0.965	0.964	0.20	0.968	(7476, 2) (296, 512)
GB	0.964	0.960	0.964	0.964	0.02	0.976	(7466, 12) (288, 520)
AB	0.964	0.961	0.965	0.964	0.12	0.975	(7471, 7) (289, 519)
DT	0.964	0.961	0.964	0.964	0.01	0.970	(7461, 17) (282, 526)
SVM	0.953	0.948	0.951	0.953	0.78	0.956	(7427, 51) (342, 466)
KNN	0.947	0.939	0.948	0.947	2.15	0.937	(7461, 17) (419, 389)
NN	0.963	0.959	0.963	0.963	0.04	0.973	(7466, 12) (295, 513)

Table 3.4: Optimized performance after hyperparameter tuning using Randomized-SearchCV for male group using ML models across different evaluation metrics; Accuracy, Precision, Recall, F1 score, AUC and CM. Execution time for each model (in seconds) is provided in the ET (s) column.

Model	Accuracy	F1	Precision	Recall	ET (s)	AUC	CM
LR	0.965	0.962	0.964	0.965	0.01	0.963	(10760, 59) (348, 544)
RF	0.975	0.973	0.976	0.975	0.09	0.971	(10819, 0) (287, 605)
GB	0.976	0.974	0.976	0.976	0.04	0.979	(10805, 14) (271, 621)
AB	0.976	0.974	0.976	0.976	0.16	0.978	(10812, 7) (279, 613)
DT	0.975	0.973	0.976	0.975	0.01	0.973	(10811, 8) (281, 611)
SVM	0.966	0.962	0.965	0.966	1.28	0.961	(10789, 30) (374, 518)
KNN	0.961	0.955	0.962	0.961	2.94	0.939	(10806, 13) (441, 451)
NN	0.975	0.973	0.975	0.975	0.04	0.976	(10801, 18) (278, 614)

Table 3.5: Optimized performance after hyperparameter tuning using RandomizedSearchCV for female group using ML models across different evaluation metrics; Accuracy, Precision, Recall, F1 score, AUC and CM. Execution time for each model (in seconds) is provided in the ET (s) column.

Model	Accuracy	F1	Precision	Recall	ET (s)	AUC	CM
LR	0.961	0.958	0.959	0.961	0.01	0.962	(18152, 148) (636, 1064)
RF	0.972	0.970	0.973	0.972	0.52	0.972	(18295, 5) (547, 1153)
GB	0.972	0.970	0.973	0.972	0.19	0.979	(18279, 21) (530, 1170)
AB	0.973	0.970	0.973	0.973	0.25	0.977	(18285, 15) (535, 1165)
DT	0.972	0.970	0.972	0.972	0.01	0.972	(18277, 23) (535, 1165)
SVM	0.962	0.958	0.960	0.962	4.25	0.960	(18217, 83) (687, 1013)
KNN	0.958	0.953	0.959	0.958	6.11	0.938	(18270, 30) (802, 898)
NN	0.972	0.969	0.972	0.972	0.03	0.976	(18278, 22) (548, 1152)

Table 3.6: Optimized performance after hyperparameter tuning using RandomizedSearchCV for combined group using ML models across different evaluation metrics; Accuracy, Precision, Recall, F1 score, AUC and CM. Execution time for each model (in seconds) is provided in the ET (s) column.

Feature	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
year	-0.005	-0.005	-0.005	-0.005	0.019
age	-1.604	-0.600	-1.298	1.496	0.186
race:AfricanAmerican	-0.010	-0.010	0.047	0.047	-0.010
race:Asian	0.011	-0.003	-0.003	-0.003	-0.003
race:Caucasian	0.008	-0.023	0.008	0.008	0.008
race:Hispanic	-0.003	-0.003	-0.003	-0.003	-0.003
race:Other	0.011	0.011	0.011	0.011	-0.047
hypertension	-0.057	-0.057	-0.057	-0.057	-0.057
heart_disease	-0.032	-0.032	-0.032	0.758	-0.032
bmi	0.009	-0.306	0.442	0.209	0.444
hbA1c_level	1.249	2.107	1.035	-3.465	2.107
blood_glucose_level	0.648	-1.101	-1.737	-1.737	0.171
smoking_history_No Info	-0.249	0.212	-0.249	-0.249	-0.249
smoking_history_current	-0.012	-0.012	-0.012	-0.012	-0.012
smoking_history_ever	-0.008	-0.008	-0.008	-0.008	-0.008
smoking_history_former	-0.006	-0.006	-0.006	-0.006	-0.006
smoking_history_never	-0.003	-0.003	-0.003	-0.003	-0.003
smoking_history_not current	-0.004	-0.004	-0.004	-0.004	-0.004

Table 3.7: SHAP feature contributions (importance scores) for five LR decision plot samples.

Feature	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
year	0.000	-0.000	0.000	-0.000	0.000
age	0.032	0.019	-0.022	-0.027	-0.015
race:AfricanAmerican	0.001	-0.000	-0.001	-0.000	-0.001
race:Asian	0.001	-0.000	0.000	-0.000	0.001
race:Caucasian	-0.000	0.002	-0.001	0.001	0.000
race:Hispanic	-0.005	-0.000	-0.000	-0.000	-0.000
race:Other	0.001	0.000	-0.001	0.000	0.000
hypertension	0.036	-0.004	-0.004	-0.005	-0.004
heart_disease	-0.002	-0.002	-0.002	-0.003	-0.003
bmi	-0.002	-0.011	-0.010	-0.008	0.026
hbA1c_level	0.783	-0.069	-0.016	-0.017	-0.021
blood_glucose_level	0.011	-0.015	-0.012	-0.012	-0.063
smoking_history_No Info	0.004	0.003	-0.006	-0.009	0.002
smoking_history_current	-0.000	-0.000	-0.000	-0.000	-0.000
smoking_history_ever	-0.000	-0.000	-0.000	-0.001	-0.000
smoking_history_former	0.005	-0.001	-0.001	-0.001	0.002
smoking_history_never	0.000	0.000	0.000	0.000	0.000
smoking_history_not current	0.003	0.000	-0.000	0.000	-0.001

Table 3.8: SHAP feature contributions (importance scores) for five RF decision plot samples.

Feature	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
year	-0.006	-0.001	0.001	-0.001	-0.004
age	0.213	0.064	-0.196	0.143	-0.016
race:AfricanAmerican	0.000	-0.004	0.004	0.008	0.002
race:Asian	-0.001	-0.003	0.001	-0.000	-0.000
race:Caucasian	-0.005	-0.001	0.002	-0.001	0.000
race:Hispanic	-0.003	-0.012	-0.000	-0.000	0.028
race:Other	0.003	-0.009	-0.004	0.005	-0.014
hypertension	0.306	-0.010	-0.025	-0.049	-0.005
heart_disease	-0.010	-0.008	-0.017	-0.025	0.486
bmi	0.057	0.075	0.012	-0.090	-0.063
hbA1c_level	8.950	-3.176	0.236	0.860	-1.750
blood_glucose_level	0.573	0.244	-1.879	0.634	-1.122
smoking_history_No Info	0.019	-0.013	0.006	0.123	-0.048
smoking_history_current	-0.006	-0.005	-0.001	-0.003	0.066
smoking_history_ever	-0.008	-0.004	-0.004	-0.007	-0.004
smoking_history_former	0.032	-0.004	-0.002	-0.004	-0.002
smoking_history_never	0.001	0.000	0.001	0.001	-0.001
smoking_history_not current	-0.008	0.000	0.000	0.002	0.002

Table 3.9: SHAP feature contributions (importance scores) for five GB decision plot samples.

Feature	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
year	0.000	0.000	0.000	0.000	0.000
age	-0.016	-0.009	-0.011	0.014	0.014
race:AfricanAmerican	0.000	0.000	0.000	0.000	0.000
race:Asian	0.000	0.000	-0.000	0.000	0.000
race:Caucasian	0.000	0.000	0.000	0.000	0.000
race:Hispanic	0.000	0.000	0.000	0.000	0.000
race:Other	0.000	0.000	0.000	0.000	0.000
hypertension	-0.001	-0.001	-0.001	-0.001	-0.001
heart_disease	-0.000	-0.000	-0.000	-0.000	-0.000
bmi	-0.005	-0.004	-0.004	-0.005	0.013
hbA1c_level	0.032	-0.105	0.028	0.032	0.034
blood_glucose_level	0.028	-0.097	-0.110	0.003	0.031
smoking_history_No Info	0.003	0.003	-0.003	-0.003	0.003
smoking_history_current	0.000	0.000	0.000	0.000	0.000
smoking_history_ever	0.000	0.000	0.000	0.000	0.000
smoking_history_former	0.000	0.000	0.000	0.000	0.000
smoking_history_never	0.000	0.000	0.000	0.000	0.000
smoking_history_not current	0.000	0.000	-0.000	0.000	0.000

Table 3.10: SHAP feature contributions (importance scores) for five AB decision plot samples.

Feature	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
year	-0.000	0.000	-0.000	-0.000	-0.000
age	0.064	-0.010	-0.028	-0.002	-0.028
race:AfricanAmerican	0.000	0.000	0.000	0.000	0.000
race:Asian	0.000	0.000	0.000	0.000	0.000
race:Caucasian	0.000	0.000	0.000	0.000	0.000
race:Hispanic	0.000	0.000	0.000	0.000	0.000
race:Other	0.000	0.000	0.000	0.000	0.000
hypertension	-0.003	-0.001	-0.002	-0.001	-0.002
heart_disease	-0.004	-0.001	-0.001	-0.001	-0.001
bmi	-0.028	-0.003	-0.008	-0.005	-0.006
hbA1c_level	-0.003	-0.031	-0.029	-0.062	-0.027
blood_glucose_level	0.007	-0.040	-0.012	-0.014	-0.012
smoking_history_No Info	0.010	0.000	-0.003	0.001	0.001
smoking_history_current	0.000	0.000	0.000	0.000	0.000
smoking_history_ever	0.000	0.000	0.000	0.000	0.000
smoking_history_former	0.000	-0.000	-0.000	-0.000	-0.000
smoking_history_never	0.000	0.000	0.000	0.000	0.000
smoking_history_not current	0.000	0.000	-0.000	0.000	0.000

Table 3.11: SHAP feature contributions (importance scores) for five DT decision plot samples.

Feature	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
year	-0.003	-0.003	-0.003	0.017	-0.003
age	0.207	0.596	0.089	0.342	-0.148
race:AfricanAmerican	-0.003	-0.003	0.015	-0.003	0.015
race:Asian	-0.003	-0.003	-0.003	-0.003	-0.003
race:Caucasian	0.006	-0.018	0.006	0.006	0.006
race:Hispanic	0.005	-0.001	-0.001	0.005	-0.001
race:Other	0.003	0.003	0.003	0.003	0.003
hypertension	0.342	-0.030	-0.030	-0.030	-0.030
heart_disease	-0.018	-0.018	-0.018	0.420	-0.018
bmi	0.963	0.125	0.004	-0.001	0.670
hbA1c_level	1.196	1.074	0.466	-2.575	0.709
blood_glucose_level	0.452	0.416	0.434	-0.617	-0.154
smoking_history_No Info	-0.114	0.097	0.097	0.097	0.097
smoking_history_current	-0.006	-0.006	-0.006	0.077	-0.006
smoking_history_ever	-0.004	-0.004	-0.004	-0.004	-0.004
smoking_history_former	-0.003	-0.003	-0.003	-0.003	-0.003
smoking_history_never	-0.001	-0.001	-0.001	-0.001	-0.001
smoking_history_not current	-0.002	-0.002	-0.002	-0.002	-0.002

Table 3.12: SHAP feature contributions (importance scores) for five SVM decision plot samples.

Feature	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
year	-0.001	0.007	-0.001	-0.000	0.000
age	-0.011	0.018	-0.022	-0.011	-0.019
race:AfricanAmerican	-0.001	0.003	-0.002	-0.000	-0.000
race:Asian	0.000	0.001	-0.000	0.000	0.000
race:Caucasian	-0.001	-0.001	-0.002	-0.001	-0.002
race:Hispanic	-0.001	-0.001	-0.001	-0.001	-0.001
race:Other	-0.000	0.001	0.001	0.000	0.001
hypertension	-0.013	-0.012	-0.012	-0.007	-0.011
heart_disease	-0.001	-0.001	-0.002	0.000	-0.001
bmi	-0.018	-0.016	-0.012	-0.009	-0.017
hbA1c_level	0.003	0.003	0.003	-0.021	0.001
blood_glucose_level	-0.002	-0.061	-0.000	-0.001	0.002
smoking_history_No Info	-0.006	0.003	-0.005	-0.003	-0.005
smoking_history_current	-0.001	-0.001	-0.000	-0.000	-0.000
smoking_history_ever	-0.000	0.000	-0.000	-0.000	-0.001
smoking_history_former	-0.001	-0.000	-0.001	-0.000	-0.000
smoking_history_never	-0.000	-0.000	0.000	-0.000	0.000
smoking_history_not current	0.000	0.001	-0.000	0.000	-0.001

Table 3.13: SHAP feature contributions (importance scores) for five KNN decision plot samples.

Feature	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
year	0.002	0.000	-0.000	-0.006	-0.001
age	-0.034	0.005	-0.006	-0.002	0.009
race:AfricanAmerican	0.002	-0.003	0.001	0.003	0.001
race:Asian	0.001	0.000	0.001	-0.005	0.001
race:Caucasian	0.001	0.001	0.001	-0.002	-0.004
race:Hispanic	-0.001	0.000	0.000	0.000	-0.001
race:Other	0.003	0.002	-0.003	-0.012	0.001
hypertension	-0.010	-0.006	-0.007	-0.019	-0.005
heart_disease	0.000	0.001	0.000	0.002	0.001
bmi	-0.017	-0.004	-0.004	0.004	-0.007
hbA1c_level	-0.019	-0.050	-0.012	0.050	-0.097
blood_glucose_level	-0.014	-0.055	-0.063	0.004	-0.005
smoking_history_No Info	-0.037	0.013	-0.027	-0.170	0.021
smoking_history_current	0.003	0.003	0.003	0.008	-0.038
smoking_history_ever	0.004	-0.020	0.004	0.011	0.003
smoking_history_former	0.003	0.003	0.003	0.007	0.002
smoking_history_never	0.003	0.003	0.003	0.008	0.003
smoking_history_not current	0.002	0.001	0.001	0.004	-0.001

Table 3.14: SHAP feature contributions (importance scores) for five NN decision plot samples.

Weight	Feature
0.0579 ± 0.0007	hbA1c_level
0.0467 ± 0.0013	blood_glucose_level
0.0010 ± 0.0007	bmi
0.0007 ± 0.0004	age
0.0001 ± 0.0001	hypertension
0.0000 ± 0.0000	race:AfricanAmerican
0.0000 ± 0.0001	heart_disease
0.0000 ± 0.0001	smoking_history_never
0.0000 ± 0.0000	race:Asian
0.0000 ± 0.0001	smoking_history_not current
0.0000 ± 0.0001	year
-0.0000 ± 0.0000	race:Caucasian
-0.0000 ± 0.0001	race:Other
-0.0000 ± 0.0001	race:Hispanic
-0.0001 ± 0.0000	smoking_history_current
-0.0001 ± 0.0001	smoking_history_ever
-0.0001 ± 0.0003	smoking_history_No Info
-0.0001 ± 0.0001	smoking_history_former

Figure 3.35: Permutation importance of features for the GB model as computed by ELI5.

Weight	Feature
0.0609 ± 0.0024	hbA1c_level
0.0412 ± 0.0008	blood_glucose_level
0.0006 ± 0.0003	bmi
0.0002 ± 0.0001	hypertension
0.0001 ± 0.0004	smoking_history_No Info
0.0001 ± 0.0002	heart_disease
0.0000 ± 0.0003	age
0 ± 0.0000	smoking_history_not current
0 ± 0.0000	smoking_history_never
0 ± 0.0000	smoking_history_current
0 ± 0.0000	smoking_history_ever
0 ± 0.0000	smoking_history_former
0 ± 0.0000	race:Hispanic
0 ± 0.0000	race:Other
0 ± 0.0000	race:Caucasian
0 ± 0.0000	race:Asian
0 ± 0.0000	race:AfricanAmerican
0 ± 0.0000	year

Figure 3.36: Permutation importance of features for the AB model as computed by ELI5.

Weight	Feature
0.0578 ± 0.0005	hbA1c_level
0.0464 ± 0.0010	blood_glucose_level
0.0005 ± 0.0006	bmi
0.0003 ± 0.0002	age
0.0002 ± 0.0003	hypertension
0.0001 ± 0.0003	heart_disease
0.0001 ± 0.0001	smoking_history_No Info
0 ± 0.0000	smoking_history_not current
0 ± 0.0000	smoking_history_never
0 ± 0.0000	smoking_history_former
0 ± 0.0000	race:Caucasian
0 ± 0.0000	smoking_history_current
0 ± 0.0000	race:Hispanic
0 ± 0.0000	race:Other
0 ± 0.0000	race:AfricanAmerican
0 ± 0.0000	race:Asian
0 ± 0.0000	year
-0.0000 ± 0.0001	smoking_history_ever

Figure 3.37: Permutation importance of features for the DT model as computed by ELI5.

Weight	Feature
0.0457 ± 0.0007	hbA1c_level
0.0310 ± 0.0009	blood_glucose_level
0.0051 ± 0.0003	age
0.0024 ± 0.0013	bmi
0.0006 ± 0.0001	heart_disease
0.0006 ± 0.0003	smoking_history_No Info
0.0005 ± 0.0003	hypertension
0.0002 ± 0.0003	smoking_history_former
0.0001 ± 0.0004	smoking_history_current
0.0000 ± 0.0001	year
0.0000 ± 0.0001	smoking_history_ever
-0.0000 ± 0.0000	smoking_history_not current
-0.0000 ± 0.0000	race:Hispanic
-0.0001 ± 0.0001	race:AfricanAmerican
-0.0001 ± 0.0001	race:Other
-0.0001 ± 0.0001	race:Asian
-0.0001 ± 0.0001	race:Caucasian
-0.0001 ± 0.0001	smoking_history_never

Figure 3.38: Permutation importance of features for the SVM model as computed by ELI5.

Weight	Feature
0.0356 ± 0.0005	blood_glucose_level
0.0303 ± 0.0009	hbA1c_level
0.0056 ± 0.0007	age
0.0027 ± 0.0015	bmi
0.0008 ± 0.0002	hypertension
0.0008 ± 0.0005	heart_disease
0.0006 ± 0.0002	year
0.0000 ± 0.0001	smoking_history_ever
-0.0004 ± 0.0004	smoking_history_current
-0.0005 ± 0.0002	smoking_history_not current
-0.0005 ± 0.0003	smoking_history_former
-0.0005 ± 0.0004	smoking_history_No Info
-0.0008 ± 0.0002	race:Asian
-0.0009 ± 0.0010	smoking_history_never
-0.0009 ± 0.0004	race:AfricanAmerican
-0.0009 ± 0.0004	race:Hispanic
-0.0011 ± 0.0007	race:Caucasian
-0.0016 ± 0.0005	race:Other

Figure 3.39: Permutation importance of features for the KNN model as computed by ELI5.

Weight	Feature
0.0552 ± 0.0013	hbA1c_level
0.0453 ± 0.0018	blood_glucose_level
0.0213 ± 0.0010	smoking_history_never
0.0190 ± 0.0009	smoking_history_No Info
0.0092 ± 0.0003	smoking_history_former
0.0083 ± 0.0007	smoking_history_current
0.0063 ± 0.0004	smoking_history_not current
0.0041 ± 0.0001	smoking_history_ever
0.0011 ± 0.0003	age
0.0006 ± 0.0004	bmi
0.0003 ± 0.0002	race:Other
0.0003 ± 0.0000	heart_disease
0.0003 ± 0.0003	race:Caucasian
0.0002 ± 0.0003	race:Hispanic
0.0001 ± 0.0002	race:Asian
-0.0000 ± 0.0002	year
-0.0002 ± 0.0002	race:AfricanAmerican
-0.0002 ± 0.0004	hypertension

Figure 3.40: Permutation importance of features for the NN model as computed by ELI5.

Model	Accuracy	F1	Precision	Recall	ET(s)	AUC	CM
LR	0.960	0.726	0.875	0.62	0.03	0.958	(18150, 150) (646, 1054)
RF	0.968	0.788	0.908	0.696	3.73	0.952	(18180, 120) (517, 1183)
GB	0.972	0.807	0.998	0.677	3.24	0.976	(18298, 2) (549, 1151)
AB	0.972	0.805	1.000	0.674	0.82	0.970	(18300, 0) (554, 1146)
DT	0.955	0.731	0.735	0.727	0.10	0.856	(17855, 445) (464, 1236)
SVM	0.967	0.764	0.993	0.621	109.15	0.926	(18293, 7) (644, 1056)
KNN	0.967	0.776	0.910	0.676	0.48	0.906	(18186, 114) (551, 1149)
NN	0.972	0.804	0.996	0.675	3.97	0.974	(18295, 5) (553, 1147)

Table 3.15: Predicted classification performance (combined group), using the four selected features; HbA1c, Blood Glucose, Age and BMI is reported across Accuracy, Precision, Recall, F1-score, AUC and CM. Model execution time (in seconds) is listed in the ET(s) column.

Model	Accuracy	F1	Precision	Recall	ET(s)	AUC	CM
LR	0.956	0.682	0.886	0.554	0.06	0.925	(18179, 121) (758, 942)
RF	0.972	0.805	1.000	0.674	1.35	0.941	(18300, 0) (554, 1146)
GB	0.972	0.805	1.000	0.674	1.40	0.943	(18300, 0) (554, 1146)
AB	0.972	0.805	1.000	0.674	0.56	0.934	(18300, 0) (554, 1146)
DT	0.972	0.805	1.000	0.674	0.03	0.942	(18300, 0) (554, 1146)
SVM	0.969	0.773	1.000	0.629	151.87	0.916	(18300, 0) (630, 1070)
KNN	0.972	0.805	1.000	0.674	0.60	0.869	(18300, 0) (554, 1146)
NN	0.972	0.805	1.000	0.674	1.85	0.936	(18300, 0) (554, 1146)

Table 3.16: Predicted classification performance (combined group), using the two selected features; HbA1c, Blood Glucose is reported across Accuracy, Precision, Recall, F1-score, AUC and CM. Model execution time (in seconds) is listed in the ET(s) column.

Model	Baseline	Hyperparameter	4-Feature	2-Feature
LR	0.961	0.962	0.958	0.925
RF	0.960	0.972	0.952	0.941
GB	0.978	0.979	0.976	0.943
AB	0.971	0.977	0.970	0.934
DT	0.856	0.972	0.856	0.942
SVM	0.892	0.960	0.926	0.916
KNN	0.896	0.938	0.906	0.869
NN	0.976	0.976	0.974	0.936

Table 3.17: Comparison of AUC values across baseline, hyperparameter-tuned, four feature and two feature models.

Chapter 4

Conclusions

This work explored the prediction of diabetes using ML techniques on a large scale dataset obtained from Kaggle consists of 100,000 patient records. The study aimed to develop predictive models that not only achieve high performance but are also interpretable and clinically relevant. A systematic methodology was employed, integrating multiple model evaluation, hyperparameter tuning and XAI techniques to address the challenges associated with black box ML models and chose most influential features.

Initially, eight ML classifiers were trained on the combined dataset using a standard train test split. Performance metrics; accuracy, F1-score, precision, recall, AUC and CM were computed to assess model effectiveness. Among these, ensemble based models (GB, RF and AB) consistently demonstrated superior predictive performance. Neural networks also achieved competitive results highlighting the potential of non-linear models in diabetes risk prediction.

In addition, XAI techniques, including SHAP, LIME and ELI5 were employed to gain insights into feature importance and model behavior. These methods enabled both global interpretability, by identifying the most influential features across the dataset and local interpretability by explaining individual instance level predictions. Across multiple models and XAI techniques the consistently identified key features were HbA1c, blood glucose, age and BMI. Using this information, reduced feature subsets were created, comprising the top four and top two features. Classifiers retrained on these subsets maintained high predictive performance, demonstrating that simpler models with fewer features can achieve comparable results while enhancing interpretability.

The findings of this study confirm that integrating XAI with ML models provides a ro-

bust framework for both accurate and interpretable diabetes prediction. In particular, GB emerged as the best performing model when evaluated on the full feature set, whereas RF and NNs provided competitive results with reduced feature subsets. SHAP, LIME and ELI5 analyses demonstrated that HbA1c and blood glucose levels are the most critical predictors which aligns with clinical knowledge and validates the practical relevance of the models.

Future research can build on this work in several meaningful directions. First, incorporating additional data modalities such as medical imaging, continuous glucose monitoring or lifestyle information may enhance predictive accuracy and enable multimodal diabetes risk assessment. DL architectures, including CNNs and transformer-based models could also be explored to capture complex patterns and temporal dynamics, especially in longitudinal data. Beyond SHAP, LIME and ELI5, more advanced interpretability methods such as Integrated Gradients, Anchors or counterfactual explanations may provide complementary insights and further strengthen model transparency. Prospective validation remains essential as real-world clinical testing would help assess the robustness and reliability of the predictive models in practice. Future work may also include developing a clinical decision support system that integrates model predictions and explainability tools into a user friendly interface for healthcare providers. Additionally, building population specific models using diverse demographic datasets could improve generalizability and uncover subgroup dependent predictors. Finally, benchmarking this pipeline against alternative ML workflows, including AutoML frameworks may reveal opportunities for improved performance, efficiency and interpretability.

s

Bibliography

[1] B. B. Duncan, D. J. Magliano, and E. J. Boyko, "IDF diabetes atlas 11th edition 2025: global prevalence and projections for 2050," 2025.

[2] World Health Organization, "Diabetes: Key facts," https://www.who.int/news-room/fact-sheets/detail/diabetes, 2023, accessed: 2025-12-08.

[3] A. G. Tabák, C. Herder, W. Rathmann, E. J. Brunner, and M. Kivimäki, "Prediabetes: A high-risk state for diabetes development," *The Lancet*, vol. 379, no. 9833, pp. 2279–2290, 2012.

[4] A. D. Association, "2. classification and diagnosis of diabetes: standards of medical care in diabetes—2021," *Diabetes Care*, vol. 44, no. Supplement_1, pp. S15–S33, 2021.

[5] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*. Wiley Series in Probability and Statistics, 2013.

[6] J. Ma, P. Dhiman, C. Qi, G. Bullock, M. van Smeden, R. D. Riley, and G. S. Collins, "Poor handling of continuous predictors in clinical prediction models using logistic regression: a systematic review," *Journal of Clinical Epidemiology*, vol. 161, pp. 140–151, 2023.

[7] H. M. Deberneh and I. Kim, "Prediction of type 2 diabetes based on machine learning algorithm," *International Journal of Environmental Research and Public Health*, vol. 18, no. 6, p. 3317, 2021.

[8] A. Rajkomar, J. Dean, and I. Kohane, "Machine learning in medicine," *New England Journal of Medicine*, vol. 380, pp. 1347–1358, 2019.

[9] J. J. Khanam and S. Y. Foo, "A comparison of machine learning algorithms for diabetes prediction," *ICT Express*, vol. 7, no. 4, pp. 432–439, 2021.

[10] H. Lai, H. Huang, K. Keshavjee, A. Guergachi, and X. Gao, “Predictive models for diabetes mellitus using machine learning techniques,” *BMC Endocrine Disorders*, vol. 19, no. 1, p. 101, 2019.

[11] C. Rudin, “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead,” *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[12] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[13] M. T. Ribeiro, S. Singh, and C. Guestrin, “”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*, 2016, pp. 1135–1144.

[14] S. M. Marshall and A. Flyvbjerg, “Prevention and early detection of vascular complications of diabetes,” *BMJ*, vol. 333, no. 7566, pp. 475–480, 2006.

[15] E. Afsaneh, A. Sharifdini, H. Ghazzaghi, and M. Z. Ghobadi, “Recent applications of machine learning and deep learning models in the prediction, diagnosis, and management of diabetes: a comprehensive review,” *Diabetology Metabolic Syndrome*, vol. 14, no. 1, p. 196, 2022.

[16] G. S. Collins, S. Mallett, O. Omar, and L.-M. Yu, “Developing risk prediction models for type 2 diabetes: a systematic review of methodology and reporting,” *BMC Medicine*, vol. 9, no. 1, p. 103, 2011.

[17] A. Brnabic and L. M. Hess, “Systematic literature review of machine learning methods used in the analysis of real-world data for patient-provider decision making,” *BMC Medical Informatics and Decision Making*, vol. 21, no. 1, p. 54, 2021.

[18] M. Lugner, A. Rawshani, E. Helleryd, and B. Eliasson, “Identifying top ten predictors of type 2 diabetes through machine learning analysis of UK Biobank data,” *Scientific Reports*, vol. 14, no. 1, p. 2102, 2024.

[19] M. Kiran, Y. Xie, N. Anjum, G. Ball, B. Pierscionek, and D. Russell, “Machine learning and artificial intelligence in type 2 diabetes prediction: a comprehensive 33-year bibliometric and literature analysis,” *Frontiers in Digital Health*, vol. 7, p. 1557467, 2025.

[20] E. Tjoa and C. Guan, "A survey on explainable artificial intelligence (xai): Toward medical xai," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 11, pp. 4793–4813, 2020.

[21] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," *arXiv preprint arXiv:1702.08608*, 2017.

[22] P. Netayawijit, W. Chansanam, and K. Sorn-In, "Interpretable machine learning framework for diabetes prediction: Integrating smote balancing with shap explainability for clinical decision support," *Healthcare*, vol. 13, no. 20, 2025, art. 2588.

[23] M. Maniruzzaman, M. J. Rahman, M. Al-MehediHasan, H. S. Suri, M. M. Abedin, A. El-Baz, and J. S. Suri, "Accurate diabetes risk stratification using machine learning: role of missing value and outliers," *Journal of Medical Systems*, vol. 42, no. 5, p. 92, 2018.

[24] E. Otles, J. Oh, B. Li, M. Bochinski, H. Joo, J. Ortwine, E. Shenoy, L. Washer, V. B. Young, K. Rao *et al.*, "Mind the performance gap: examining dataset shift during prospective validation," in *Machine Learning for Healthcare Conference*. PMLR, 2021, pp. 506–534.

[25] C. C. Olisah, L. Smith, and M. Smith, "Diabetes mellitus prediction and diagnosis from a data preprocessing and machine learning perspective," *Computer Methods and Programs in Biomedicine*, vol. 220, p. 106773, 2022.

[26] P. B. Khokhar, C. Gravino, and F. Palomba, "Advances in artificial intelligence for diabetes prediction: insights from a systematic literature review," *Artificial Intelligence in Medicine*, 2025, art. 103132.

[27] I. Kavakiotis, O. Tsave, A. Salifoglou, N. Maglaveras, I. Vlahavas, and I. Chouvarda, "Machine learning and data mining methods in diabetes research," *Computational and Structural Biotechnology Journal*, vol. 15, pp. 104–116, 2017.

[28] J. Li, Z. Xu, T. Xu, and S. Lin, "Predicting diabetes in patients with metabolic syndrome using machine-learning model based on multiple years' data," *Diabetes, Metabolic Syndrome and Obesity: Targets and Therapy*, pp. 2951–2961, 2022.

[29] L. Li, Y. Cheng, W. Ji, M. Liu, Z. Hu, Y. Yang, Y. Wang, and Y. Zhou, "Machine learning for predicting diabetes risk in western china adults," *Diabetology Metabolic Syndrome*, vol. 15, no. 1, p. 165, 2023.

[30] M. P. LaValley, “Logistic regression,” *Circulation*, vol. 117, no. 18, pp. 2395–2399, 2008.

[31] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[32] A. Natekin and A. Knoll, “Gradient boosting machines, a tutorial,” *Frontiers in Neurorobotics*, vol. 7, p. 21, 2013.

[33] R. E. Schapire, “Explaining adaboost,” in *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik, Springer*, 2013, pp. 37–52.

[34] B. De Ville, “Decision trees,” *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 5, no. 6, pp. 448–455, 2013.

[35] S. Suthaharan, “Support vector machine,” in *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*. Springer, 2016, pp. 207–235.

[36] O. Kramer, “K-nearest neighbors,” in *Dimensionality Reduction with Unsupervised Nearest Neighbors, Springer*, 2013, pp. 13–23.

[37] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, pp. 533–536, 1986.

[38] TeamHG-Memex, “ELI5: Explain Like I’m 5,” https://github.com/TeamHG-Memex/eli5, 2023, accessed: 2025-12-02.

[39] J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *Journal of Machine Learning Research*, vol. 13, no. 1, pp. 281–305, 2012.