International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V73I8P127

Volume 73 Issue 8, 312-332, August 2025

Original Article

Diagnosing Diabetes Onset with Machine Learning Enhanced Predictive Analysis in Pima Indians

Abhishek Kumar[1*], Partha Sarathi Bishnu[1], Pushpanjali R. Ojha[2]

[1]*Department of Computer Science and Engineering, BIT Mesra, Lalpur, Ranchi, Jharkhand, India.*
[2]*Department of Pathology, Rajendra Institute of Medical Sciences, Bariatu, Ranchi, Jharkhand, India.*

[*]*Corresponding Author : abhishek.atom@gmail.com*

Received: 24 February 2025 Revised: 15 August 2025 Accepted: 23 August 2025 Published: 30 August 2025

Abstract - *Since diabetes is becoming more common, early and precise prediction is essential for prevention and management. This study predicts diabetes mellitus in the Pima Indian community using Deep Learning (DL) and advanced Machine Learning (ML) techniques. Model performance improved significantly through meticulous data preprocessing, incorporating imputation for missing values, data balancing techniques, advanced feature engineering, and sophisticated statistical methods for managing incomplete data and identifying key features. The models employed here include a multilayer deep learning model called the Tree-based Pipeline Optimization Tool (TPOT) and ensemble learning using LightGBM and K-Nearest Neighbors (KNN). High accuracy score, precision score, recall score, and F1 score of roughly 94.5% were attained by each model following a rigorous review and improvement procedure. Comprehensive experiments were conducted, with results analyzed graphically and numerically, offering in-depth insights and recommendations. The proposed approach outperforms the most advanced techniques already in use, proving its efficacy and emphasizing the critical role that prompt and precise prediction plays in the prevention and treatment of diabetes in high-risk populations.*

Keywords *- Deep Learning, Diabetes prediction, Feature engineering, Machine Learning, Medical informatics.*

1. Introduction

Since 1980, the prevalence of diabetes mellitus, a common chronic illness, has nearly doubled worldwide [5]. Approximately one in ten individuals worldwide had diabetes. In 2021, it was found that in the age group of 20 to 79 years, 537 million adults were diabetic. This number may reach 643 million by 2030. The number of diabetics may reach 783 million by 2045. About 75% of diabetes patients belong to low- and middle-income countries. With 6.7 million deaths in 2021, diabetes was responsible for one death every five seconds on average. Moreover, the global healthcare expenditure related to diabetes was nearly $966 billion, reflecting a 316% rise over the previous 15 years [10]. The Pima Indian population, a Native American group residing predominantly in Arizona, has been a focus of diabetes research due to their exceptionally high rates of type 2 diabetes. This population's genetic predisposition, combined with lifestyle and environmental factors, makes it particularly vulnerable, highlighting the need for effective diabetes prediction and prevention strategies. Many cases of diabetes often go undiagnosed due to a lack of early symptoms, complicating early detection efforts.

Some of the methods of diagnosis available for diabetes include Random Blood Sugar (RBS), PostPrandial Blood Sugar (PPBS), Fasting Blood Sugar (FBS), Haemoglobin A1c (HbA1c) testing, urine sugar testing, urine microalbumin testing, and organ-specific investigations for complications like kidney disease. Early detection through these tests is critical to preventing severe complications like kidney disease and reducing healthcare costs. Integrating connected medical devices with Artificial Intelligence (AI) models, including DL and ML, offers promising potential for improving diagnosis, especially in underserved areas [5]. Trust and transparency are essential in healthcare, where accuracy is paramount. AI tools are highly effective at processing vast medical datasets, uncovering hidden patterns that enhance diagnostic accuracy and treatment decisions. Despite substantial progress in diabetes prediction using ML and DL methods, many studies suffer from inconsistent handling of missing values, limited exploration of engineered features, and suboptimal model performance, particularly in high-risk subgroups like the Pima Indian population. This study addresses these gaps by introducing robust preprocessing techniques, novel engineered features, and optimized ensemble and deep learning models. This paper applies three state-of-the-art AI models-Light Gradient Boosting Machine (Light-GBM) and K-Nearest Neighbors (KNN)- for accurate diabetes prediction, delivering highly effective and

interpretable results. Despite numerous studies using ML/DL for diabetes prediction, several specific challenges remain unresolved. Many models suffer from poor generalization due to data imbalance and a lack of robust preprocessing strategies.

Additionally, most studies rely on default features without exploring domain-informed feature interactions. Manual tuning and a lack of automated optimization frameworks also limit reproducibility and scalability. Furthermore, very few works validate improvements statistically, raising concerns about the reliability of reported gains. This study addresses these limitations through advanced data preprocessing, novel feature engineering, ensemble learning, and statistical evaluation.

This paper makes the following contributions.

1. Apply statistical methods to handle missing data, select key features, balance the dataset, and engineer new features. Visualizations further supported data exploration, helping to clarify relationships and improve model performance.
2. Implementation of three AI techniques: Ensemble Learning with LightGBM and KNN, Tree-based Pipeline Optimization Tool (TPOT), and Multilayer Deep Learning.
3. Presentation and analysis of comprehensive experimental results through both graphical and numerical methods. This analysis includes in-depth explanations and recommendations. To demonstrate the efficiency and benefits of the proposed work, this paper compares the findings to those of the most advanced methods currently available.

This outline of the proposed article is as follows: Section 2 presents a study of related work in this domain. In Section 3, the paper sheds light on the models and concepts used, including data pretreatment approaches, feature selection, TPOT, multilayer deep learning, and ensemble techniques with LightGBM and KNN. The experimental analysis is described in full in Section 4, along with the model evaluation, data analysis, and parameters for all applicable procedures. Section 5 of the study provides the scope of future research and concludes the proposed work.

2. Literature Review

Here, we examine several studies that have investigated different DL and ML methods for the prediction of diabetes, highlighting significant gains in model performance and accuracy across a range of datasets. Kannadasan et al. 2019 [14] utilized stacked autoencoders for feature extraction. They achieved an accuracy of 86.26%. A Support Vector Machine (SVM) for prediction using imputation, feature selection, feature scaling, data augmentation techniques, and tenfold stratified cross-validation was used by Raafat et al. 2021 [19]. Their method yielded a good accuracy score (83.20%), sensitivity score (87.20%), and specificity score (79%) in a framework for remote monitoring. Khanam and Foo (2021), [15] utilized different machine learning algorithms.

They reported high accuracy rates, reaching up to 88.6% for diabetes prediction, outperforming previous studies. For prediction of the risk of diabetes, Ramesh et al. 2021 [20] also used SVM with similar results, achieving an accuracy of 83.20% with data imputation, feature scaling and selection, augmentation, and cross-validation with tenfold stratification. Using outlier rejection, missing value imputation, and normalization, Gupta et al. (2022) [9] compared Deep Learning and Quantum Machine Learning models. Their DL model outperformed both QML and existing models, achieving 90% accuracy with low false detection and missed detection rates, while the QML model showed satisfactory performance comparable to existing literature. Chatrati et al. 2022 [6] explored conditional decision-making for predicting blood pressure and used SVM for predicting diabetes within a desktop application for home monitoring.

They achieved 75% accuracy in diabetes prediction using diastolic blood pressure and glucose measurements. Chang et al. (2023) [5] trained and tested interpretable supervised ML models while assessing feature selection subsets using a variety of metrics. The NB fared better than the other models, achieving 79.13% accuracy, while DT consistently performed well in sensitivity (88.43-89.92%), with all models demonstrating around 80% accuracy. An analysis of ensemble and conventional ML models for diabetes prediction risk was conducted by Saxena et al. (2023) [24]. According to their research, the Super Learner model possessed the highest accuracy, coming in at 86%.

Ashour et al. (2024), [1] evaluated FNN and CNN models. The FNN model achieved 82% accuracy, outperforming previous studies, while both models demonstrated high accuracy, specificity, and AUC for early diabetes detection. Jain et al., 2024 [12] assessed four imputation techniques (MICE, KNN, Mean, and Median) on different ML classifiers, including DT, RF, SVC, and Gaussian Naive Bayes, for diabetes classification. They found that SVM with median imputation performed best in accuracy and precision, while GNB with KNN imputation excelled in the recall, and GNB with MICE imputation led in F1-score, AUC, and G-mean.

However, the impact of imputation techniques on classifier performance was minor. Noviyanti and Alamsyah, 2024 [17] utilized the Random Forest algorithm for early diabetes detection, achieving 87% accuracy. By using the PIMA diabetes dataset, Bhuvaneswari [2024] combined RF, Radial SVM, and KNN in an ensemble technique (En-RfRsK), and had the capacity to forecast diabetes mellitus with an accuracy of 88.89%. Employing an ensemble stacking

technique that included deep neural networks and classical models, Reza et al. 2024 [21] achieved 75.03% accuracy using a train-test splitting mechanism.

They achieved 77.10% accuracy with cross-validation, beating previous approaches by 2.23% to 12%. Logistic Regression (LR), SVM, and RF were utilized for diabetes prediction by Xie, 2024 [26], who employed K-NN for imputation. The LR model outperformed RF and SVM with 79.13% accuracy and 0.8571 precision, indicating high promise for early diabetes diagnosis using machine learning methods Finally, Salih et al., 2024 [23] applied data preprocessing techniques, including outlier removal, imputation, and normalization, followed by feature selection using PCA and classification models such as SVM, RF, NB, and DT.

They achieved 89.86% accuracy in diabetes classification using SVM with feature selection, outperforming other classifiers. Unlike prior works that primarily rely on default features, our study introduces five novel features designed from domain knowledge, improving robustness and reducing feature redundancy. Moreover, we utilize a combination of TPOT, ensemble voting, and deep learning to systematically explore and benchmark performance. While most studies report 74% and 89% accuracy, few combine robust imputation strategies, advanced feature engineering, and statistical validation, which our proposed framework achieves.

While prior studies have achieved reasonable accuracy, several limitations persist. First, many works rely on default features without leveraging domain-driven feature engineering (e.g., Glucose-to-Blood Pressure ratio, BMI × Insulin). Second, imputation is often inconsistent or applied globally, which ignores class-specific data distributions. Third, few studies adopt ensemble frameworks that combine interpretable and high-performance models. Lastly, comparative studies rarely validate improvements using statistical significance tests. These gaps motivate our work, which proposes a hybrid framework combining class-wise preprocessing, novel features, and rigorous statistical evaluation.

Table 1 summarizes the related work on diabetes among the Pima Indians. After reviewing state-of-the-art techniques, two main limitations were identified:

1. Imputation and Preprocessing Variability; inconsistent evaluation of imputation and preprocessing techniques can lead to biases and a lack of standard practices (See Section 3).
2. Suboptimal Results: Some studies report less than optimal performance, indicating room for improvement in predictive accuracy (See Section 4.3).

Table 1. Summarized related work on PIMA Indian diabetes

Author Year	Methodology	Key Findings
Kannadasan et al., 2019 [14]	Stacked autoencoders	Achieved 86.26% classification accuracy on PIDD.
Raafat et al., 2021 [19]	SVM	Accuracy:83.20%
Khanam & Foo, 2021 [15]	Seven different ML techniques	Achieved high accuracy rates (up to 88.6%).
Ramesh et al., 2021 [20]	SVM	Accuracy:83.20%
Gupta et al., 2022 [9]	DL and Quantum ML	Achieving 90% with low false detection and missed detection rates.
Chatrati et al., 2022 [6]	SVM	Achieved: 75%accuracy.
Chang et al., 2023 [5]	Three different ML techniques	Decision tree consistently performed well in sensitivity (88.43-89.92%); models demonstrated good accuracy (around 80%).
Saxena et al., 2023 [24]	Different ensemble and classical machine learning models for predicting the risk of diabetes	Achieved 86% accuracy.
Ashour et al., 2024 [1]	Feedforward Neural Network and Convolutional Neural Network	FNN achieved 82% accuracy.
Jain et al., 2024 [12]	Evaluated four imputation techniques (MICE, KNN, Mean, Median) on various ML classifiers	SVM with median imputation was best for accuracy and precision. GNB with KNN imputation excelled in recall. GNB with MICE imputation led to F1 score, AUC, and G-mean.
Noviyanti & Alamsyah, 2024, [17]	Random Forest algorithm	Achieved 87% accuracy.

Bhuvaneswari 2024 [3]	Ensemble approach (RF, R-SVM, KNN)	88.89% accuracy.
Reza et al., 2024 [21]	Combining a deep neural network with classical models using the stacking ensemble approach	Obtained 75.03% accuracy and 77.10% with cross-validation.
Xie, L. 2024 [26]	K-NN for imputation, LR, SVM, and RF for diabetes prediction	Achieved 79.13% accuracy and precision of 0.8571 (LR Model).
Salih et al., 2024 [23]	Feature selection (PCA), classification models (SVM, RF, NB, DT)	Achieved 89.86% accuracy.

3. Methodology

Key data pretreatment steps, how to handle missing values, balance data, engineer features, standardization, identify critical features, and model training and evaluation, are covered in detail in this section. First, handling missing values ensures data completeness, which is essential for accurate model predictions. Class imbalances are then resolved using data balancing, which might distort model performance if ignored. After that, feature engineering turns unstructured data into useful features that increase model effectiveness. To ensure that each feature contributes equally, standardization is then done to scale the features. Important features are found in the prepared data to draw attention to variables that significantly impact the model. To adjust the model and make sure it performs effectively on never-before-seen data, the training of the model and the evaluation stage are completed last. The general methodology employed in this investigation is depicted in Figure 1.

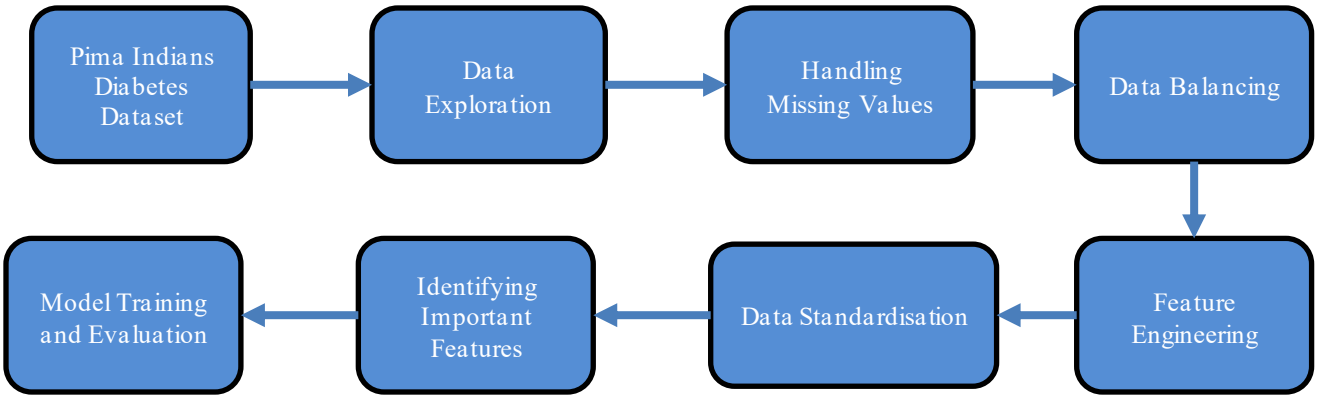


Fig. 1 The overall methodology

3.1. Data Exploration

The dataset under analysis pertains to the most popular Pima Indians Diabetes dataset [8]. The dataset contains diagnostic measures instrumental in identifying individuals at risk for diabetes. A summary of the dataset is given in this section, along with information on its kinds, structure, and the importance of each characteristic. The dataset consists of 768 records, each representing an individual case, and 9 columns (features) that correspond to different diagnostic measurements. However, an initial inspection of the data revealed that certain features contained zero values. As these values appeared illogical within the context of the data, they were interpreted as missing values and subsequently replaced with NaN. Figure 2 shows a population divided into two distinct categories: Healthy and Diabetic. A horizontal bar chart shows the numerical distribution (Label 1 or Diabetic: 268 and Label 0 or Healthy: 500) of individuals within each category. The "Healthy" category significantly outnumbers the "Diabetic" category, indicating a lower incidence of diabetes in this population. The charts reveal a clear distinction between the two groups, highlighting a substantial disparity in the number of individuals classified as healthy versus diabetic. As a result, data balancing is essential to improve model accuracy, a topic further explored in Section 3.3.

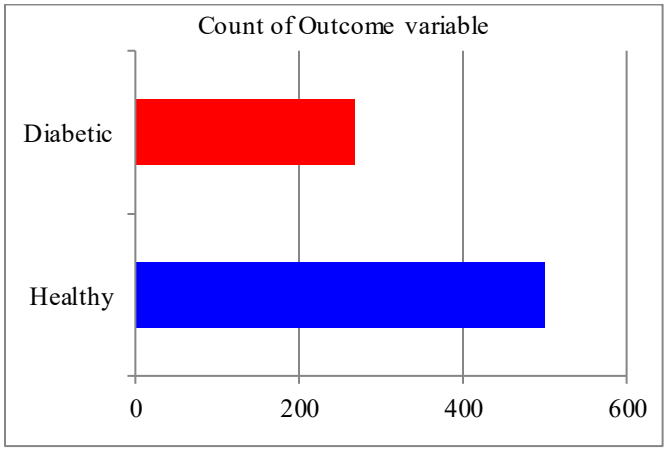


Fig. 2 Outcome types and their counts, and distribution of outcome variables

The dataset displays skewed distributions across its eight features (Figure 3), with most values concentrated on the lower end. Age and pregnancies are primarily lower, while glucose and blood pressure show a similar trend of lower values. Skin thickness is more symmetrical, but insulin levels are significantly skewed towards higher values. BMI is moderately skewed, and the diabetes pedigree function is primarily low. Overall, the dataset seems biased towards a younger, healthier population, with fewer instances of higher glucose, blood pressure, and pregnancy counts.

Table 2 displays summary statistics of all the features. Potential issues were observed, such as missing data, with several features having minimum values of 0, possibly indicating missing entries. Additionally, the data distribution shows skewness in some features, like Pregnancy and insulin. Among these variables, Glucose, BMI, and Insulin emerge as crucial predictors for diabetes, with Blood Pressure, Age, and Diabetes Pedigree Function (DPF) also being significant indicators.

The distribution of the dataset is depicted in the boxplot (Figure 4), which shows that most features are skewed with longer tails towards higher values, suggesting that most people had lower measurements in these categories. Features like Insulin, Skin Thickness, and Blood Pressure show significant outliers, suggesting the presence of extreme values.

The data shows that most women have had few pregnancies, with generally low glucose, blood pressure, and insulin levels, though there are some high outliers. Based on the data distribution (Figure 3), summary statistics (Table 3), and box plot (Figure 4), it is observed that while the dataset is complete, several features contain zero values (except for features Pregnancies, Diabetes Pedigree Function, Age, and Outcome), where data may be missing. It is crucial to address these potential missing values appropriately. In Section 3.2, a detailed discussion is provided on how these missing values are handled.

3.2. Handling Missing Values

The table (Table 3) depicts how missing data is handled in various features of the dataset. The "# Zero" column shows the number of records with zero values, which might indicate missing or invalid data points. The distributions of skin thickness, blood pressure, BMI, insulin, and glucose in healthy and diabetic groups are shown in Figure 5. The noticeable skewness in most features supports the choice of using the median for imputing missing values. Using the median to handle missing values is a reliable data preprocessing technique [13].

3.3. Exploratory Data Analysis (EDA)

To fully understand the dataset, we conduct a detailed Exploratory Data Analysis. Descriptive statistics revealed significant skewness in variables such as Insulin and SkinThickness, with several zero entries likely indicating missing data.

Histograms and boxplots (Figures 2-4) highlighted strong differences in Glucose and BMI distributions between diabetic and non-diabetic groups. A class imbalance was noted, with 268 positive and 500 negative samples.

The correlation heatmap (Figure 7) showed a strong positive correlation between Glucose and Outcome, and moderate associations with BMI and Age. This informed feature selection and model prioritization. EDA also helped validate the rationale for engineered features such as the Glucose-to-Blood Pressure Ratio (GBPR) and BMI × Insulin, which were positively skewed among diabetic individuals. These insights guided data preprocessing and model design in subsequent sections.

Median is better than mean since it is less impacted by outliers and skewed data, particularly for characteristics with extreme values. Even with asymmetric distributions, the data's central tendency is reliably represented by substituting the median for missing values.

This approach enhances dataset completeness, leading to better model training and more reliable predictions. The last two columns (Table 4) detail the median values used for imputation: the last but one column for individuals with a class label of 0 (Healthy), and the last column for those with a class label of 1 (Diabetic). For example, in the "Insulin" feature, 374 records (48.70%) had zeros, which were imputed with the median values 102.5 for Healthy individuals and 169.5 for Diabetic individuals.

3.4. Data Balancing

Here, we employ the Synthetic Minority Over-sampling Technique (SMOTE) [7] to remove the data imbalance, where there were 500 healthy people and 268 diabetics in the sample. In a medical diagnosis scenario, a model trained on imbalanced data might incorrectly predict that a patient is healthy, simply because most of the data used for training is from healthy subjects. To properly balance the dataset, SMOTE produces synthetic data instances for the minor class (diabetes group) instead of replicating existing instances.

The process starts by separating the target variable (Outcome) from the feature set. After that, SMOTE produces a new dataset with almost the same number of instances in each class. This approach ensures that the model trains on data that does not favor the majority class, thereby reducing bias and improving predictive accuracy. After applying SMOTE, the dataset becomes balanced, with 500 samples in both classes. The application of SMOTE can be effectively justified by the lemma that follows:

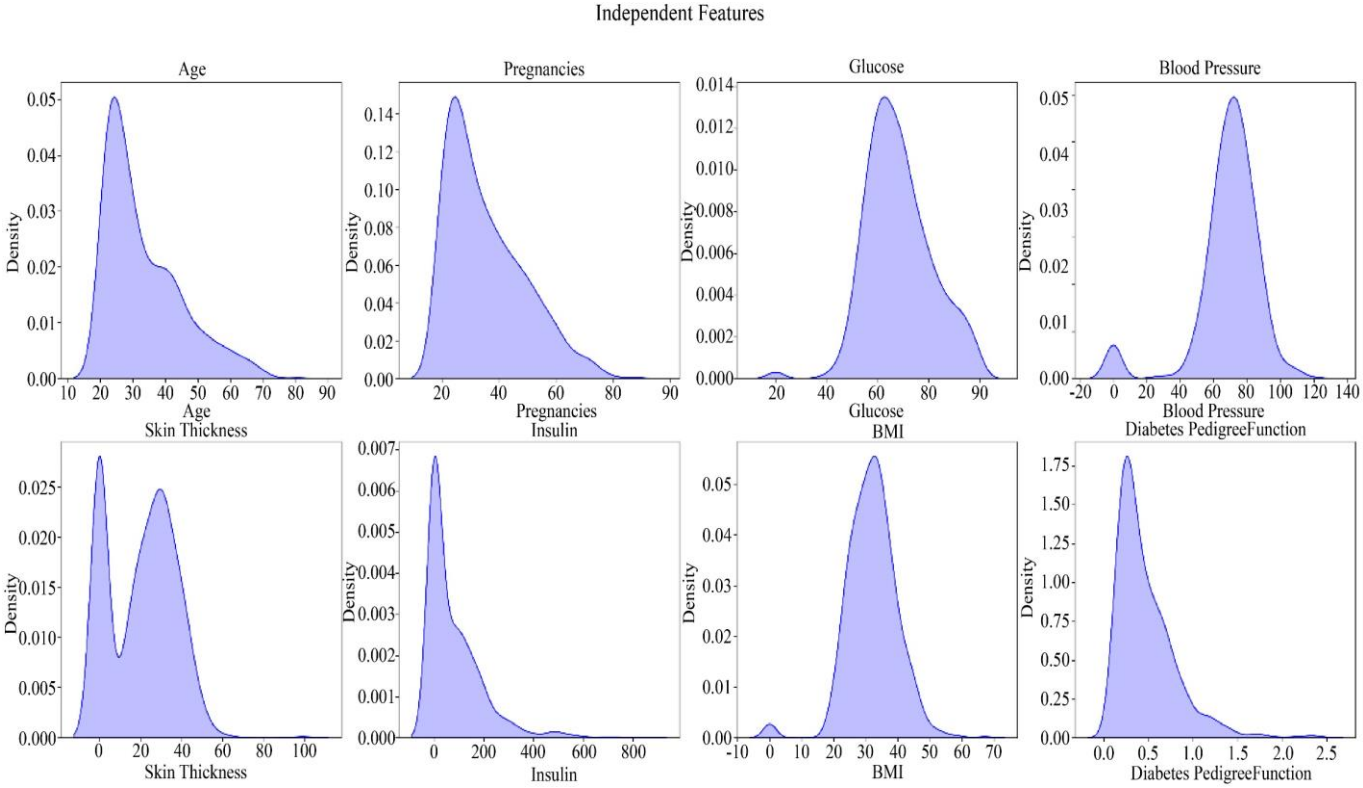


Fig. 3 A graphical representation of the feature distributions in the Pima Indians diabetes dataset

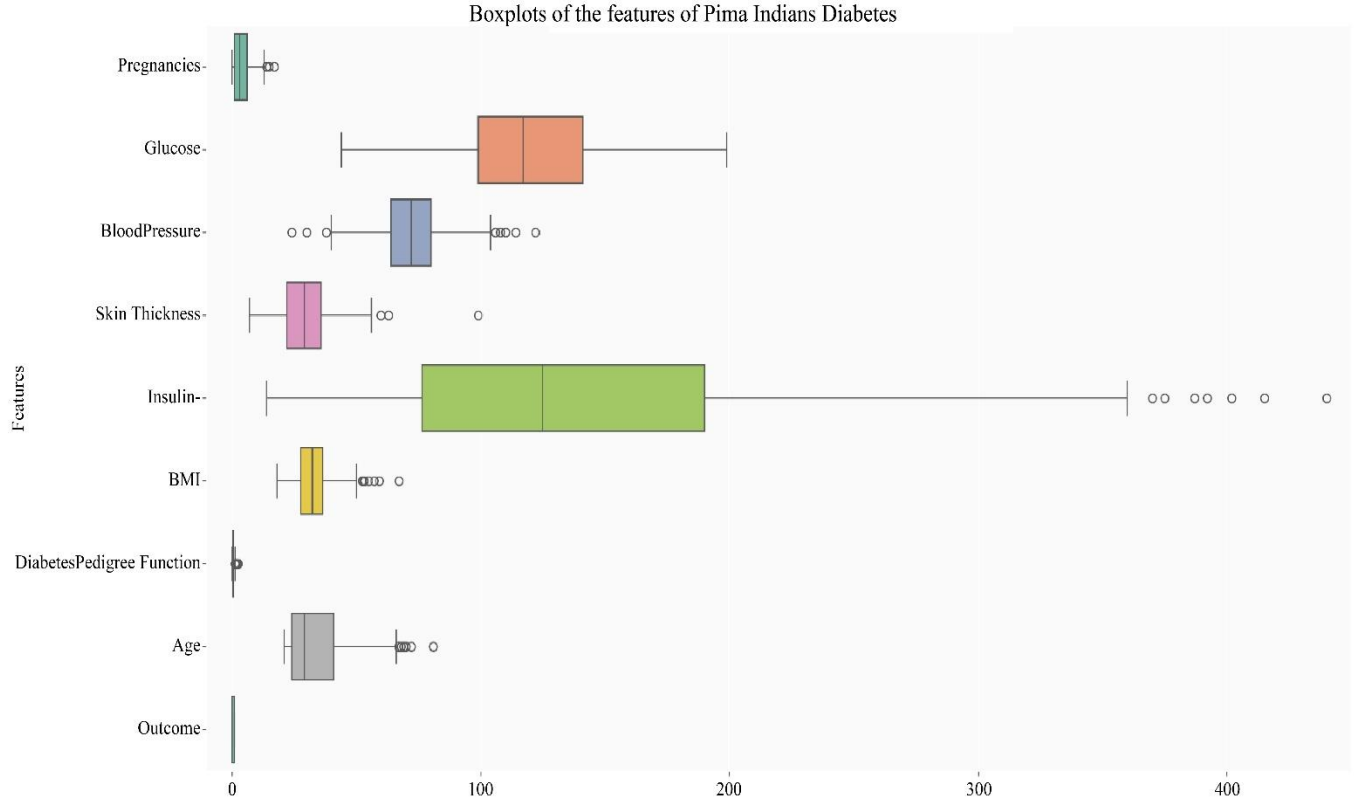


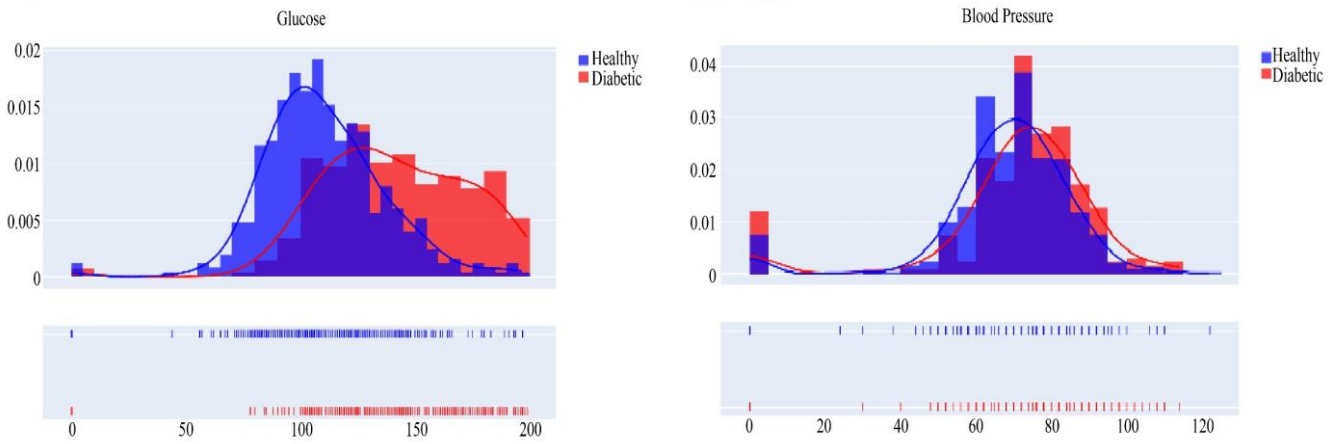
Fig. 4 Displays the distribution of the features using the boxplots

Table 2. Summary STATISTICS of the Pima Indian diabetes dataset

Features	Mean	Std Dev.	Minimum	Quartile 1	Median	Quartile 3	Maximum
Pregnancies	3.854	3.370	0	1.000	3.000	6.000	17.000
Glucose	120.895	31.973	0	99.000	117.000	140.250	199.000
Blood Pressure	69.105	19.356	0	62.000	72.000	80.000	122.000
Skin Thickness	20.536	15.952	0	0	23.000	32.000	99.000
Insulin	79.799	115.244	0	0	30.500	127.250	846.000
BMI	31.993	7.884	0	27.300	32.000	36.600	67.100
DPF	0.472	0.331	0.078	0.244	0.372	0.626	2.420
Age	33.241	11.760	21.000	24.000	29.000	41.000	81.000
Outcome	0.349	0.477	0	0	0	1	1

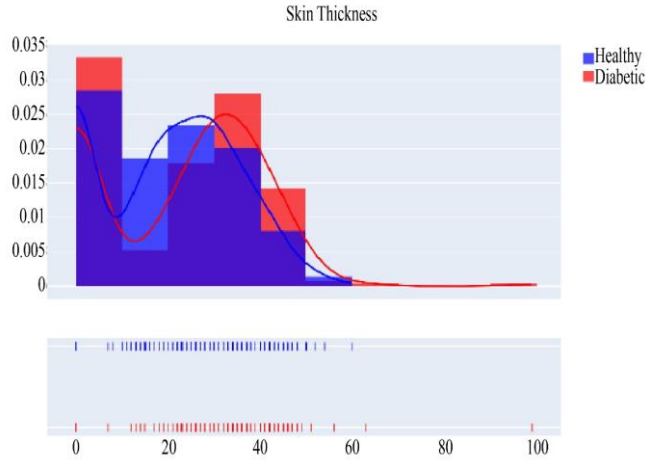
Table 3. Identification and handling of missing values

Features	# Zero	% Missing	Values Replaced for 0 class	Values Replaced for 1 class
Pregnancies	111	NA	NA	NA
Glucose	5	0.65	107.0	140.0
Blood Pressure	35	4.56	70.0	74.5
Skin Thickness	227	35.42	27.0	32.0
Insulin	374	48.70	102.5	169.5
BMI	11	1.43	30.1	34.3
DPF	0	0	NA	NA
Age	0	0	NA	NA

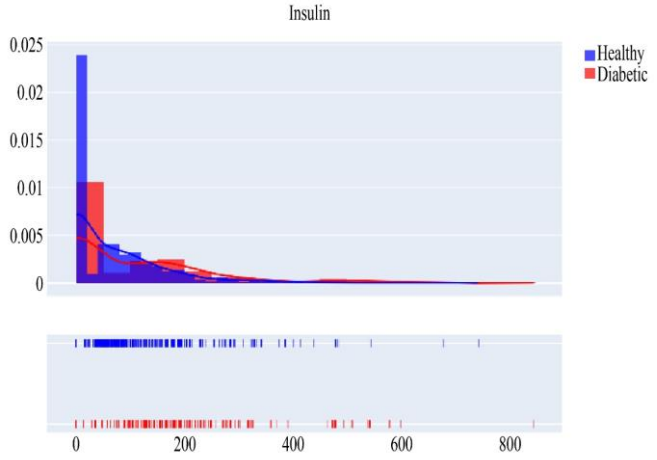


(a)

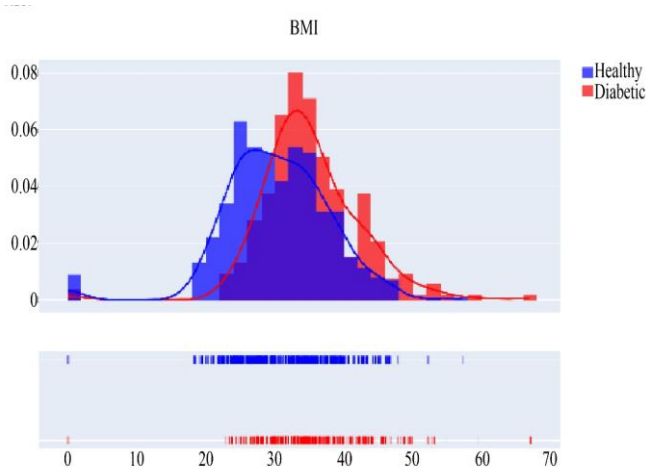
(b)



(c)



(d)



(e)

Fig. 5 Comparison of features distributions between healthy (class 1) and diabetic groups (class 0)

3.4.1. Lemma 1 (Impact of Data Imbalance on Model Precision)

In a binary classification task, it is anticipated that if the minority class is underrepresented, the classifier's precision will decrease, particularly if data balancing strategies such as SMOTE are not used.

3.4.2. Proof

Let the dataset consist of two classes, C_0 (majority) and C_1 (minority), such that $|C_1| \ll |C_0|$. In this case, a classifier tends to be biased toward predicting C_0, resulting in a higher number of false positives for C_1.

Precision is calculated as:

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

Where FP stands for false positives and TP for true positives. In an imbalanced dataset, FP for the minority class C_1 tends to increase due to the bias toward C_0, thus lowering the precision for C_1. Nevertheless, by equalizing the quantity of samples in each class, data balancing techniques like SMOTE reduce this bias and increase precision for C1.

3.5. Develop New Features

Drawing from the insights presented in Section 3.6, where key features and correlations among them were explored, five new features have been identified (F1 to F5) based on these findings.

These features are developed to capture deeper relationships and interactions among the original variables, providing a more comprehensive understanding for predictive modelling. F1: Glucose-to-Blood Pressure Ratio (GBPR),

$$F1 = \frac{Glucose}{Blood\ Pressure} \quad (2)$$

This feature assesses the connection between Blood Pressure and Glucose. High glucose and blood pressure are common indicators of metabolic syndrome, a condition that increases the risk of diabetes. The ratio helps capture this relationship and its impact on diabetes risk. F2: BMI-Insulin Product (BMI Insulin):

$$F2 = BMI \times Insulin \quad (3)$$

This feature combines BMI and Insulin levels to explore how excess body weight (BMI) influences insulin production or resistance. A high product of BMI and insulin is a strong indicator of insulin resistance, often found in diabetic patients. F3: Age-to-Glucose Ratio (AGR):

$$F3 = \frac{Age}{\text{Glucose Ratio}} \quad (4)$$

This feature combines Age and Glucose to reflect how glucose levels change with age. This ratio provides insight into the age-related risk of high glucose. F4: Skin Thickness-to-BMI Ratio (STBR),

$$F4 = \frac{\text{Skin Thickness}}{\text{BMI}} \quad (5)$$

This feature examines the relationship between Skin Thickness (a measure related to body fat) and BMI. It helps explore how fat distribution in the body (skin thickness) relative to overall body mass (BMI) contributes to diabetes risk. F5: Diabetes Pedigree $Function - to - Age\ Ratio\ (F5)$

$$F5 = \frac{Diabetes\ Pedigree\ Function}{Age} \quad (6)$$

This feature evaluates the correlation between an individual's age and their hereditary susceptibility to diabetes. This susceptibility is shown by the Diabetes Pedigree Function. Given the importance of early diagnosis and

treatment, a greater ratio may indicate that younger people with a significant genetic susceptibility are more prone to develop diabetes earlier in life. A thorough statistical summary for all engineered features is presented in Table 4, emphasizing their variability and dispersion.

3.5.1. Lemma 2. Feature Interaction and Model Robustness Statement

By lowering multicollinearity and offering more insightful depictions of the relationships between variables, the addition of engineered features-such as ratios or products-that capture interactions between original features enhances the prediction model's accuracy and robustness.

3.5.2. Proof

Let F1, F2, F3, F4, and F5 represent the newly engineered features based on ratios and products of the original variables. For example, $F1 = \frac{\text{Glucose}}{\text{Blood Pressure}}$ and $F2 = BMI \times Insulin$. These features reduce the redundancy between highly correlated variables (e.g., glucose and blood pressure) by collapsing them into single ratios or products. This reduction in multicollinearity between original features simplifies the model, as the Variance Inflation Factor (VIF) for the combined features decreases. Therefore, the model is less likely to suffer from overfitting or unstable coefficient estimates, improving generalization and robustness. Hence, the predictive accuracy increases as the model benefits from more meaningful feature interactions, completing the proof.

3.5.3. Theorem 1: Impact of Engineered Features on Model Predictive Performance Statement

As long as the new features give fresh, non-redundant information and lessen overfitting, adding the designed features F1, F2, F3, F4, and F5, as described in Section 3.4, guarantees that the model's prediction performance will improve.

3.5.4. Proof

Let model M use the original feature set $X\ x_1, x_2, x_n\}$which includes variables like glucose, BMI, insulin, etc. Introducing the new feature set $X' = X \cup \{F1, F2, F3, F4, F5\}$ adds engineered features that combine and transform existing variables.

Because synthetic features like F1 (glucose-to-blood pressure ratio) and F2 (BMI-insulin product) capture intricate relationships between original features that might not be linearly separable in the original space, predictive performance is improved.

These interactions contribute additional explanatory power to the model. If the engineered features provide novel, non-redundant information, they decrease the error term in the predictive function, improving accuracy while minimizing overfitting. Therefore, A(X′) ≥ A(X), where A is the accuracy function, and X′ is the expanded feature set. Thus, the inclusion of these features improves performance, completing the proof.

Table 4. Summary statistics of new features

Features	Mean	Std Dev.	Min	Quartile 1	Median	Quartile 3	Max
F1	9254.72	2973.29	2136.00	6965.00	8772.00	11354.00	20790.00
F2	5144.31	3475.22	30.53	2890..38	4620.50	6120.65	35564.00
F3	0.28	0.11	0.11	0.21	0.26	0.33	1.18
F4	0.91	0.25	0.11	0.76	0.90	1.05	3.76
F5	0.02	0.01	0.00	0.01	0.01	0.02	0.10

3.6. Standardization of the Dataset

All columns are scaled using the Standard Scaler, which standardizes the features. This scaling is represented as:

$$X_{scaled} = \frac{x-\mu}{\sigma} \quad (7)$$

Here, the actual attribute is X, σ is the standard deviation, and μ is the attribute's mean. [4].

3.7. Identification of Important Features

Table 5 contains feature importances from two different feature selection methods: Random Forests (a higher score indicates greater importance) [11] and Recursive Feature Elimination (RFE) [2] (1 being the most important). The table highlights which features are most critical for predicting diabetes according to two different feature selection methods. Both methods agree that Glucose is the most important predictor, with BMI, Age, and Diabetes Pedigree Function also being significant. However, RFE emphasizes the importance of Blood Pressure more than Random Forest does, and both methods rank Skin Thickness as the least important feature. This analysis suggests that focusing on Glucose, BMI, Age, and Diabetes Pedigree Function will likely yield the most accurate predictive models, while features like Skin Thickness and Insulin may require further investigation to determine their true value in the model.

The SHAP [16] summary map (Figure 6) sheds light on the variables affecting the prediction of diabetes. Glucose and BMI appear to be the most significant predictors of diabetes, with a substantial impact on the model output. Higher values of these features tend to increase the prediction of diabetes. Age and Diabetes Pedigree Function significantly influence the model's predictions, with higher values typically

increasing diabetes risk. Insulin has a more nuanced effect, showing both positive and negative associations with diabetes. Features like Pregnancy, blood pressure, and skin thickness contribute less to the predictions. Overall, while all features have some impact, the two main influential factors affecting the model's predictions are Glucose and BMI.

The image (Figure 7) represents a correlation heatmap. The correlation heatmap highlights critical relationships between various diagnostic features used to predict the onset of diabetes. Strong positive correlations are observed between glucose and insulin levels (0.49) and BMI and insulin (0.57), consistent with the physiological response where insulin is released in response to increased glucose levels and the potential link between obesity and insulin resistance. Age and the number of pregnancies also show a moderate positive correlation (0.54), implying that older women are more likely to become pregnant.

There is a positive correlation between the diabetes outcome and Glucose (0.5), suggesting that greater glucose levels strongly indicate the risk of developing diabetes. Moderate correlations are observed between BMI and the diabetes outcome (0.32) and between age and diabetes outcome (0.24), reinforcing the association between higher body mass, age, and the risk of developing diabetes. Moreover, some features, like the Diabetes Pedigree Function, show low or no significant correlations with other variables, indicating that genetic predisposition operates relatively independently of other diagnostic factors as measured by this function. Skin thickness also indicates a negative correlation with glucose (-0.08), suggesting a minimal connection between these variables. Lastly, blood pressure displays low correlations with most other features, suggesting it may not be a strong indicator. The heatmap emphasizes the importance of glucose, insulin, and BMI in predicting the onset of diabetes, whereas genetic factors and skin thickness may have a lesser role to play. Table 5 and Figures 6 and 7 illustrate feature importance and the correlation between features, providing insights that help identify new potential features. These aspects are discussed in detail in Section 3.4.

3.7.1. Lemma 3 (Monotonicity of Model Accuracy Based on Feature Selection)

In the context of Table 5, which evaluates feature importances from two feature selection methods-Recursive Feature Elimination (RFE) and Random Forest (RF)-removing a feature f that is deemed less important by both methods (such as Skin Thickness or Insulin), will either improve or leave the model's accuracy unchanged, assuming the feature contributes little or no information to the prediction.

3.7.2. Proof

Let F be the set of all features, and $F' \subseteq F$ be the set of selected features after applying feature selection (e.g. Recursive Feature Elimination or Random Forest Importance). Table 5 shows that features like Glucose and BMI are highly important, whereas features like Skin Thickness and Insulin rank lower in importance. If a low-importance feature $f \in F \backslash F'$ (such as Skin Thickness) is removed, and it contributes little to improve the predictive ability of the model, thereby reducing the complexity, it potentially decreases overfitting. The lemma is thus proven as the model's accuracy on fresh data either increases or stays the same.

3.7.3. Theorem 2: Optimal Feature Set for Classification Performance Statement

There exists an optimal subset of features that maximizes the classifier's accuracy. If too many irrelevant features are included, the model's performance degrades due to overfitting.

3.7.4. Proof

Let the set of all available features be F, and $F* \subseteq F$ be the optimal subset of features. The accuracy A(F) of a classifier trained on F is a function of the features. If irrelevant features are included in F, the model complexity increases, leading to overfitting and reduced generalization, which in turn reduces the accuracy. There exists a subset $F*$ that minimizes the training error while maintaining generalization performance, thus maximizing the accuracy. This proves the existence of an optimal feature set.

Table 5. Assessment of feature importance using Random Forest and Recursive Feature Elimination (RFE)

Random Forest			Recursive Feature Elimination		
F. No.	**Feature**	**Importance**	**F. No.**	**Feature**	**Importance**
1	Glucose	0.267142	1	Glucose	1
5	BMI	0.168769	2	Blood Pressure	1
7	Age	0.131567	5	BMI	1
6	Diabetes Pedigree Function	0.122695	6	Diabetes Pedigree Function	1
2	Blood Pressure	0.088660	7	Age	1
0	Pregnancies	0.085017	0	Pregnancies	2
4	Insulin	0.071547	4	Insulin	3
3	Skin Thickness	0.064604	3	Skin Thickness	4

F. No.: Feature Number

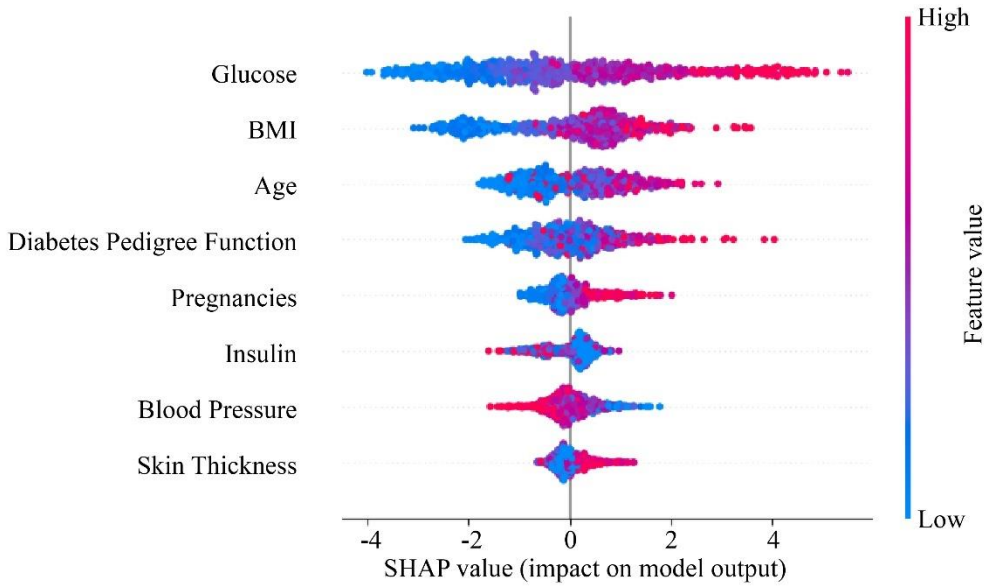


Fig. 6 Feature importance using SHAP

Fig. 7 Correlation plot of all the features

3.8. Model Overview and Concepts

This section covers the models used, including the basic concepts of Ensemble Learning with LightGBM and KNN [22, 25], Tree-based Pipeline Optimization Tool (TPOT) [18], and a Deep Learning Classifier.

3.8.1. Ensemble Learning with Light GBM and KNN

This section discusses the integration of LightGBM and k-Nearest Neighbors (KNN) in an ensemble technique, where their combined strengths enhance the model's performance [22, 25] (Figure 8). LightGBM efficiently handles large datasets, while KNN contributes simplicity and interpretability for specific data distributions. Improve accuracy, but may lead to deeper trees and potential overfitting if not properly controlled with hyperparameters like max depth. KNN is a simple. It is non-parametric. It is an instance-based learning technique for problems involving regression and classification. KNN does not provide a

generalized model like LightGBM does. Rather, it uses LightGBM (Light Gradient Boosting Machine) to classify incoming data points according to the training set's k-nearest neighbors.

LightGBM is a scalable and reliable gradient-boosting framework created by Microsoft and designed for rapid and precise predictive modeling. Here, decision trees are used as base learners and are optimized for performance and speed. LightGBM builds models iteratively, with each new model concentrating on minimizing a given loss function in order to fix the faults of the preceding one. It is applied to jobs involving both regression and classification. In contrast to conventional gradient boosting techniques, LightGBM uses a histogram-based methodology to discretize continuous features into bins, which speeds up training and uses less memory. Unlike traditional methods that grow trees level-by-level, LightGBM adopts a leaf-wise approach (also known as depth-wise), where it focuses on expanding the leaves with the highest potential for error reduction.

A distance metric such as Euclidean distance can be applied in this way. The distance measure, feature scaling, and k selection all significantly affect how well KNN performs. KNN is straightforward and easy to understand, but can become costly computationally, particularly when dealing with big datasets. Integrating these two machine learning models-LightGBM and KNN-into a Voting Classifier leverages both strengths. LightGBM handles large datasets efficiently and captures complex interactions, while KNN offers simplicity and interpretability for certain data distributions. The ensemble may be able to capture many facets of the data by combining its predictions through soft voting, which could result in better performance overall. To adjust the hyperparameters of this ensemble, Grid Search with Cross-Validation is employed. It thoroughly explores a predetermined set of hyperparameters, evaluating each combination's efficacy through cross-validation. The ensemble framework, combining LightGBM and KNN, leverages this convergence property to improve prediction accuracy. LightGBM's leaf-wise (depth-wise) approach, which expands leaves with the highest potential for error reduction, ensures a focused improvement in error minimization, enhancing convergence to optimal solutions. In the meantime, KNN manages non-linearity and provides interpretability for certain data patterns, enabling the ensemble model to represent a variety of data distributions.

3.8.2. Tree-based Pipeline Optimization Tool (TPOT)

TPOT uses genetic programming to automate the construction and optimization of machine learning pipelines [18]. (https://epistasislab.github.io/tpot/). TPOT's limitations include being resource-intensive, time-consuming, offering limited control, variable results, interpretability challenges, and scalability issues. However, despite these limitations, TPOT's advantages outweigh its disadvantages. It investigates several models and hyperparameters, exports the best models for simple repeatability and deployment, integrates easily with Scikit-Learn workflows, optimizes machine learning pipelines automatically, and requires little code to implement. Overall, TPOT's benefits make it a useful tool for machine learning tasks.

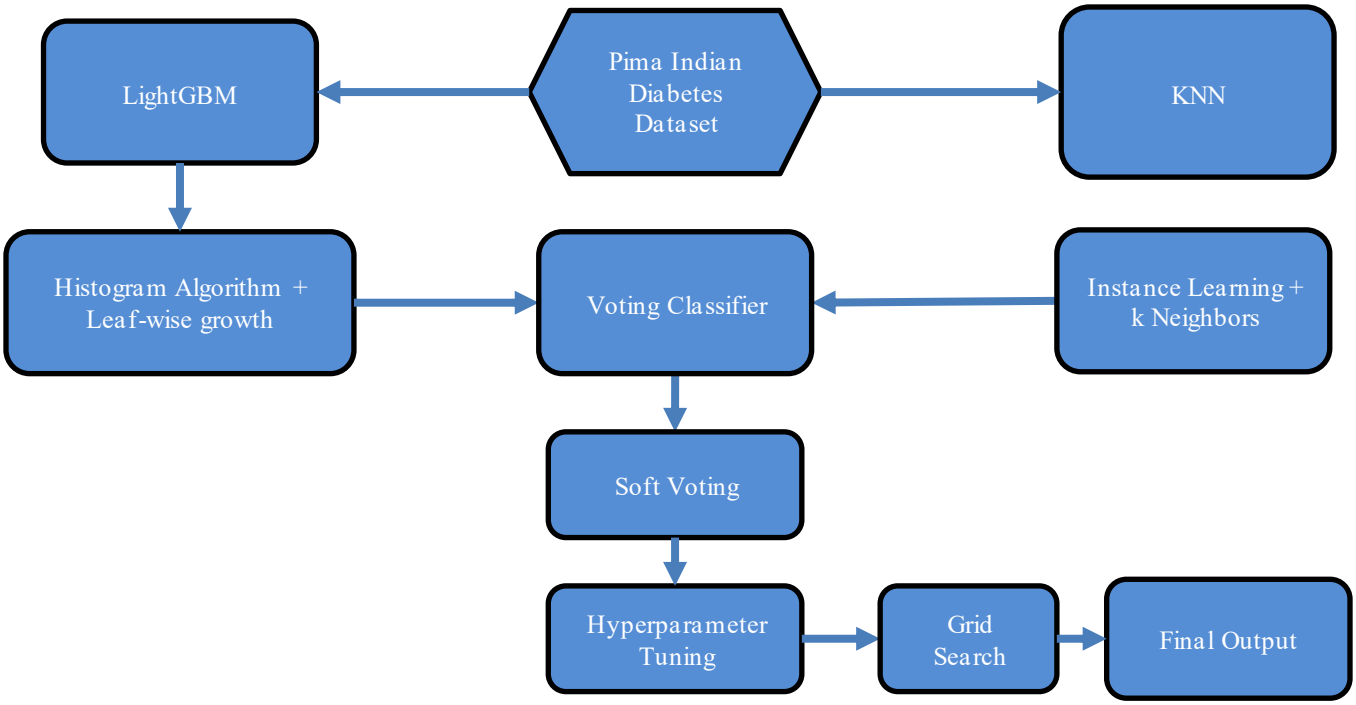


Fig. 8 Block diagram of ensemble learning with light GBM and KNN

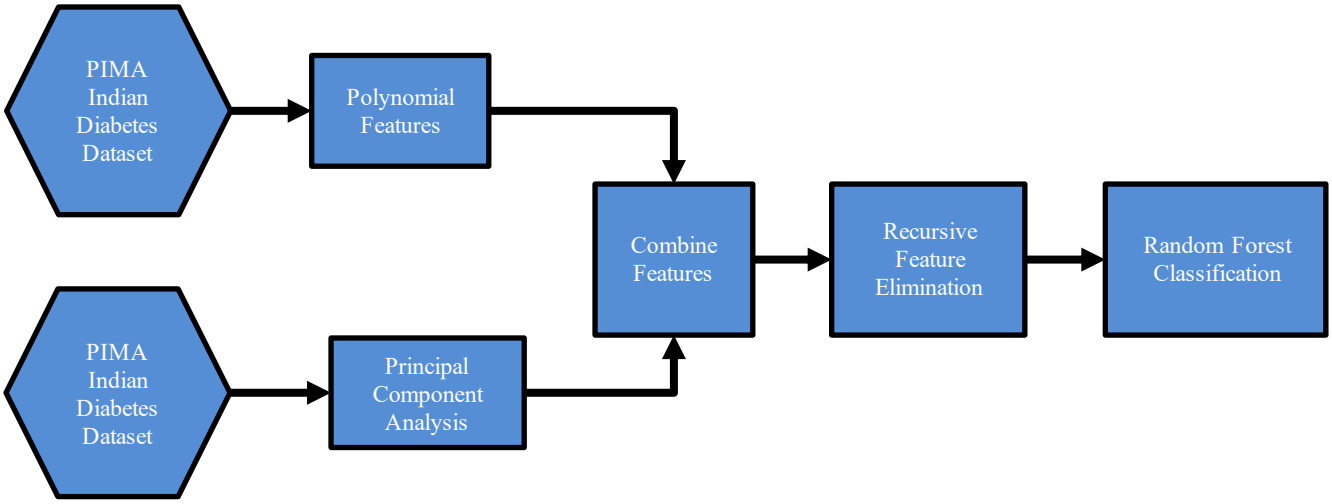


Fig. 9 Block diagram of TPOT

3.8.3. Deep Learning Classifier

The third model applies deep learning to address the problem, focusing on its relevance and effectiveness. The model begins with a 128-neuron input layer and progresses through multiple hidden layers of varying complexity, all of which contribute non-linearity through ReLU activation. The first hidden layer contains 128 neurons, followed by a 50% dropout layer, which randomly reduces 50% of the inputs to zero during training to prevent overfitting. Another Dropout layer was added for more regularization after a second 128-neuron layer and a 128-neuron layer with ReLU activation were implemented. Lastly, the output layer with a sigmoid activation function and single-neuron output was used for the binary classification, along with a 64-neuron hidden layer. With dropout layers to lessen overfitting, this structure is intended to identify complex patterns within the data.

Figure 10 shows the block diagram of the DL model. Hyperparameter Tuning Strategy For the LightGBM+KNN hybrid model, we employed GridSearchCV with 5-fold cross-validation to tune key hyperparameters such as n_neighbors (range: 3–15), learning_rate (0.01–0.3), n_estimators (50–200), and max_depth (3–10). The best combination was selected based on the validation F1-score. TPOT performed automated hyperparameter and pipeline optimization using genetic programming over 100 generations with a population size of 100. It searched across multiple models and preprocessing steps to determine the best configuration. For the deep learning model, we manually tuned batch size (32, 64), learning rate (0.001, 0.0001), and dropout rate (0.3, 0.5) using validation accuracy as the guiding metric. Early stopping was applied to prevent overfitting, and model weights with the best validation loss were retained.

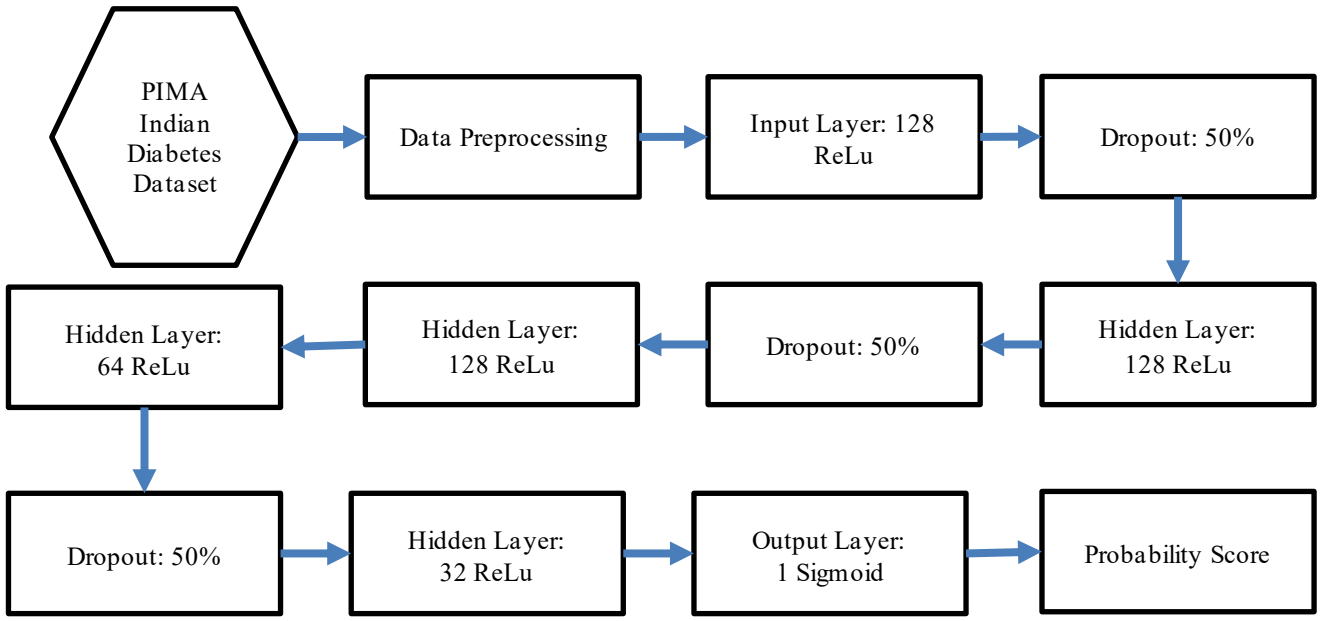


Fig. 10 Block diagram deep learning model

3.9. Dataset Bias and Limitations

While the dataset is widely used for benchmarking, it carries several inherent biases. It contains data on 768 female patients of Pima Indian heritage, all aged 21 years or older, thereby excluding males and adolescents. This 100% female-only and ethnically homogeneous sample introduces sampling bias, limiting the models' ability to perform well on data from other ethnicities. Furthermore, the dataset originates from the National Institute of Diabetes and Digestive and Kidney Diseases and is several decades old, potentially reflecting outdated screening methods. Critical clinical features, such as insulin (n = 374 zero values) and skin thickness (n = 227 zero values), include implausible zero entries, likely due to missing data or measurement errors. These inconsistencies can distort model training and lead to biased predictions. Such limitations must be carefully considered before applying models trained on this dataset to real-world clinical or policy settings.

4. Results

This section covers the parameter settings and model evaluation methods, and presents the results and their analysis.

4.1. Parametric Settings

Parameters and their values are essential for evaluation and comparison; the parameters used in all the models are first discussed. In the ensemble learning approach using LightGBM and KNN, the LightGBM classifier is initialized with a random state to ensure consistent and reproducible results. The KNN classifier is initialized without specifying the number of neighbors, allowing for flexibility in determining the optimal value through hyperparameter tuning. The classifier will consider the anticipated probabilities from each model and average them to arrive at the final prediction because the voting parameter is soft. This is useful when the models provide probability outputs. The weights parameter is [1, 1], which assigns equal importance to both LightGBM and KNN in the voting process. Adjusting these weights can influence one model more than the other. The code defines a parameter grid to tune the n neighbors hyperparameter of the KNN classifier, ranging from 1 to 29. This enables the grid search to determine, within this range, the ideal number of neighbours for the KNN algorithm. The dataset undergoes 5-fold cross-validation. In this process, data is shuffled before partitioning into batches, and a random state of 42 is set so that the results are the same in different runs of training. Next, the parameter settings for the TPOT model are discussed. The TPOT optimization process was carried out over 30 generations, gradually improving the internal Cross-Validation (CV) score from 0.9075 in Generation 1 to 0.9225 in the final generation. The best pipeline identified uses a GradientBoostingClassifier with specific hyper parameters, with a learning rate of 0.1 and a max depth of 9. The test accuracy of the final model was 0.9000. Finally, the parameter settings for the deep learning model are outlined. To guarantee reproducibility, the data were split into two parts. 80% was used for training, and 20% was used for testing. A fixed random state was used. The model was trained over 300 epochs with a batch size of 32, with 20% of the training data set aside for validation. Following training, the test set's results were predicted by the model, which classified values over 0.5 as 1 and those below as 0. Early stopping was used to monitor validation loss; if there was no progress after three epochs, training was stopped. The model's output is classified as 1 if the predicted probability exceeds 0.5; otherwise, it is classified as 0. Figure 11 shows the model summary, highlighting a total of 61,441 trainable parameters, which occupy 240.00 KB of memory.

Layer (type)	Output Shape	Param #
Dense (Dense)	(None, 128)	1,536
dense_1 (Dense)	(None, 128)	16,512
dropout (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 128)	16,512
dropout_1 (Dropout)	(None, 128)	0
dense_3 (Dense)	(None, 128)	16,512
dense_4 (Dense)	(None, 64)	8,256
dropout_2 (Dropout)	(None, 64)	0
dense_5 (Dense)	(None, 32)	2,080
dense_6 (Dense)	(None, 1)	33

Fig. 11 Model summary table

4.2. Model Evaluation Techniques

This study offers a thorough methodology for assessing ML models with measures, visualizations, and cross-validation. The precision-recall curve reveals the relationship between precision and recall. The AUC (Area Under the Curve) reflected the capacity of the model to differentiate the healthy class from the diabetic class; a greater AUC signified superior performance. The model is evaluated using the Accuracy score, precision score, recall score, and F1 score presented in Table 6.

Table 6. Evaluation metrics

Accuracy	Recall (Sensitivity)	Precision	F1 Score
$\frac{TN+TP}{TN+TP+FN+FP}$	$\frac{TP}{TP+FN}$	$\frac{TP}{TP+FP}$	$2\times\frac{Precision\times Recall}{Precision+Recall}$

4.3. Findings and Evaluation

The three models' performance is thoroughly examined in this section using classification methods along with ROC and Precision-Recall curves.

4.3.1. Results and Analysis of the Ensemble Learning Approach

Figure 12 shows the ROC and Precision-Recall curves for the Ensemble Learning Approach. With an AUC of 0.961, the ROC curve shows outstanding overall performance. Both the ROC curve and Precision-Recall curve exhibit a pronounced departure from the random baseline, suggesting that the model significantly outperforms random guessing. Overall, the model represented by these curves is a good classifier with solid performance in terms of insensitivity (TPR) and specificity (1-FPR). Analysis of Cross-Validation Results (LightGBM+KNN): The table (Table 7) shows the outcome of a 5-fold cross-validation experiment using a voting classifier composed of TN: True Negatives, TP: True Positives, FN: False Negatives, FP: False Positives LightGBM and KNN models.

Table 7. Cross-validation performance metrics for ensemble learning approach (light GBM + KNN)

Fold	Accuracy	Precision	Recall	F1 Score	ROC AUC
1	0.700	0.721	0.664	0.692	0.804
2	0.700	0.704	0.704	0.704	0.755
3	0.680	0.724	0.589	0.650	0.794
4	0.687	0.714	0.629	0.669	0.784
5	0.583	0.552	0.914	0.688	0.515
Mean	0.670	0.683	0.700	0.681	0.730
Std	0.044	0.066	0.113	0.019	0.109

The model exhibits consistent performance across the five folds, as evidenced by the relatively small standard deviation values for all metrics. This reveals the sensitivity of the proposed model to thespecific data is split in each fold. The model does well on the dataset, as evidenced by the total mean accuracy of 0.914. The model has a good precision score (0.893) and recall score (0.942), signifying it can correctly distinguish positive and negative cases accurately. The high F1-score (0.916) demonstrates a strong balance between precision and recall. The consistently high ROC AUC values (0.961) provide additional evidence of the effectiveness of the model. The results validate that the voting classifier made up of LightGBM and KNN is a dependable model for the dataset. Its consistent performance across different folds is a testament to its strong generalization ability.

Table 8. Classification report of the TPOT model

Class	Precision	Recall	F1-Score	Support
Healthy	0.93	0.90	0.91	99
Diabetic	0.90	0.93	0.92	101
Accuracy			0.92	200
Macro Avg	0.92	0.91	0.91	200
Weighted Avg	0.92	0.92	0.91	200

4.3.2. TPOT Results and Analysis

With an AUC of 0.964, the ROC curve (Figure 13) demonstrates outstanding classification performance, demonstrating high sensitivity and specificity. The model demonstrates strong performance across classes, with a balanced handling of positive and negative predictions. The classification report for the TPOT model (Table 8) indicates strong performance in predicting both Healthy and Diabetic cases. A well-balanced performance was demonstrated by the model's 0.93 precision and 0.90 recall for the Healthy class and 0.90 precision and 0.93 recall for the Diabetic class. The accuracy of 0.92 shows consistent results across both classes. The model's test accuracy of 0.9150 indicates how reliable it is at predicting diabetes.

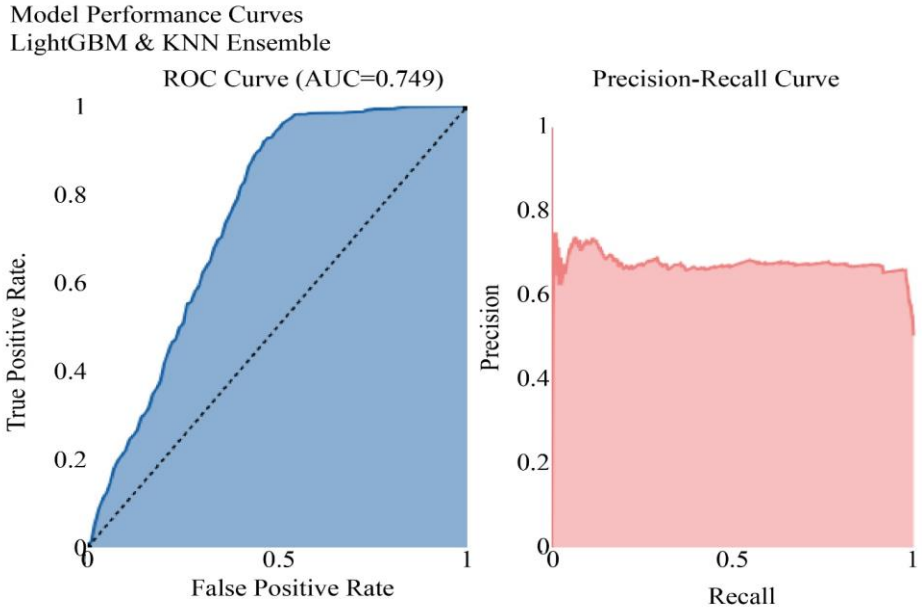


Fig. 12 ROC and precision-recall curve for ensemble learning approach

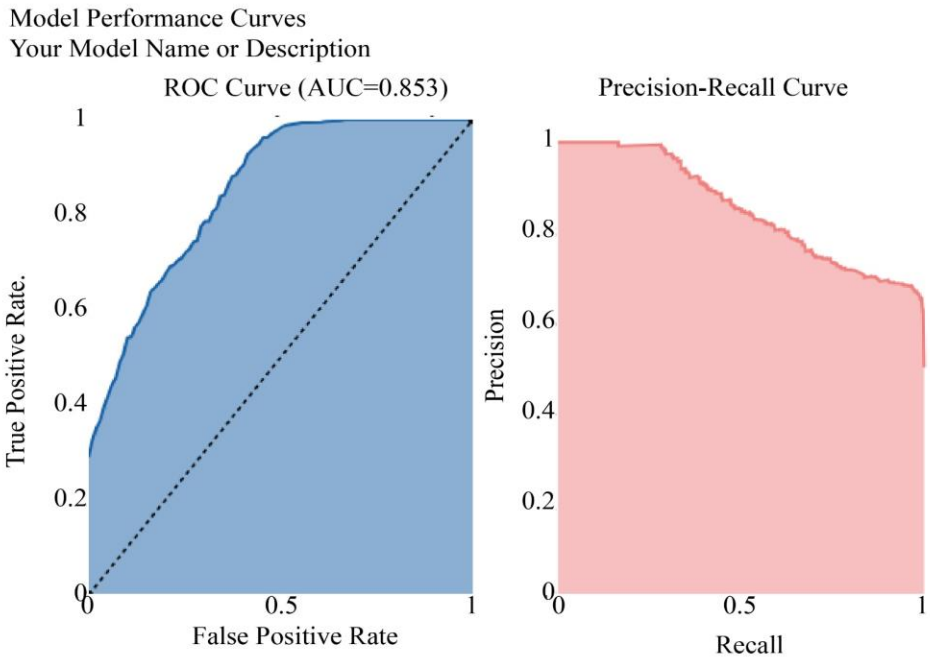


Fig. 13 ROC and precision-recall curve for TPOT

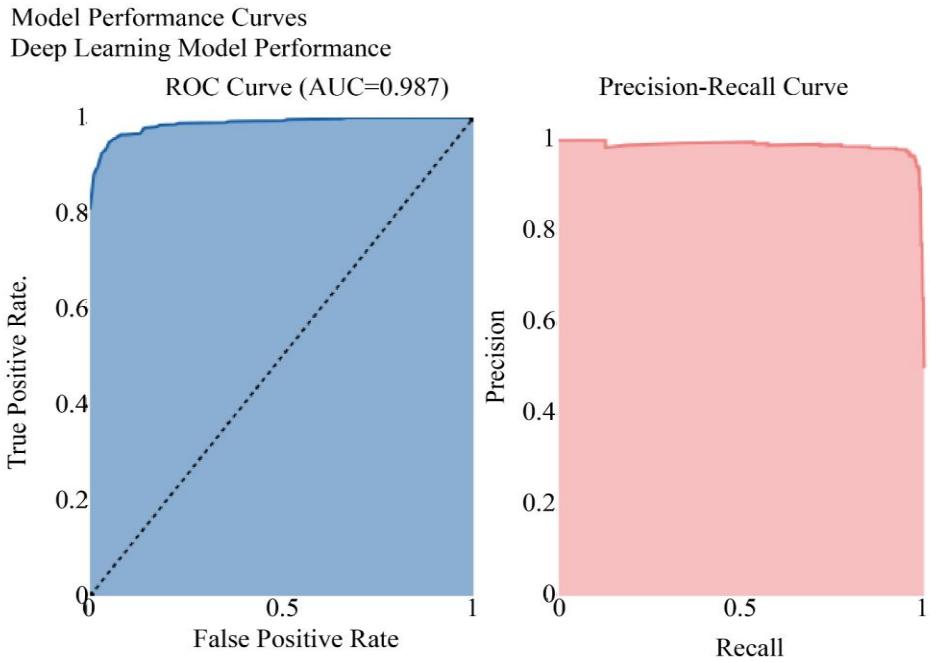


Fig. 14 ROC and precision-recall curve of the proposed deep learning model

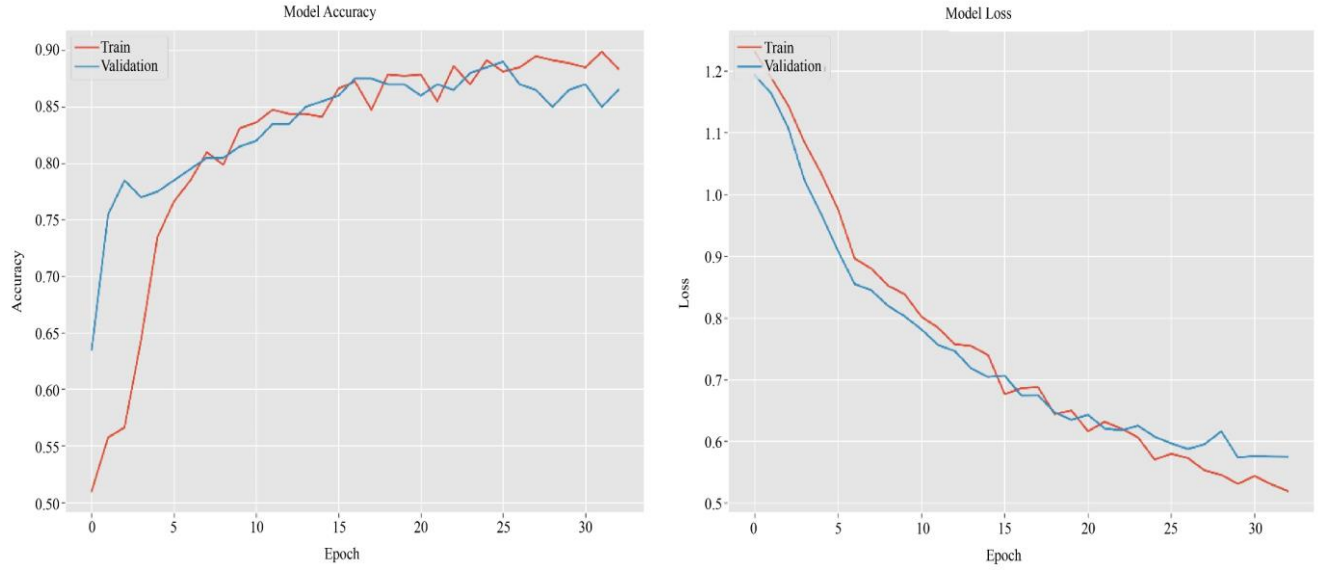
4.3.3. Findings and Evaluation Utilizing Deep Learning Techniques

Next, the results and analysis using deep learning techniques are discussed as follows: The Precision-Recall curve and ROC curves, which show the DL performance, are shown in Figure 14. With an AUC of 0.984, the ROC curve shows outstanding overall performance. The Precision-Recall and ROC curves clearly deviate from the random baseline, indicating that the model outperforms random guessing by a large margin. All things considered, the model that these curves depict is an ideal classifier with excellent performance for specificity (1-FPR) and sensitivity (TPR). With a dense neural network, the accuracy levels for testing, validation, and training are 0.9300, 0.8938, and 0.9266, respectively. Furthermore, the F1 Score, Precision score, Recall (Sensitivity) score, and Specificity score are 0.9307, 0.9307, 0.9293, and 0.9307, respectively.

Table 9. Classification report of the proposed deep learning model

Class	Precision	Recall	F1-Score	Support
0 (Healthy)	1.0	0.89	0.940	99
1 (Diabetic)	0.90	1.00	0.950	101
Accuracy	-	-	0.945	200
Macro Avg	0.95	0.94	0.940	200
Weighted Avg	0.95	0.94	0.940	200

Table 9 gives an examination of the model for the Healthy (0) and Diabetic (1) classes. The classification report highlights strong model performance. The model's 0.89 recall indicates that 11% of healthy cases are incorrectly labeled as diabetic, while its 1.00 accuracy indicates that there are no false positives for the Healthy class. The model accurately detects all cases of diabetes for the Diabetic class, demonstrating 0.90 precision and a faultless 1.00 recall. Strong performance with an F1-score of 0.95. Overall, the model achieves 0.95 accuracy and consistent performance across both classes, as shown by the macro and weighted averages of 0.94 for all key metrics. In the Healthy class, a small trade-off between recall and precision is balanced by robust diabetic predictions. As illustrated in Figure 15, the model accuracy and loss curves suggest potential overfitting. While validation accuracy plateaus, training accuracy keeps increasing, suggesting that the model might be memorizing training data instead of generalizing. Similar trends can be seen in the loss curves, with validation loss plateauing or rising and training loss declining. Future research could explore techniques to address this overfitting issue.



(a) (b)

Fig. 15 Model performance: accuracy and loss curves

Comparison with Leading-Edge Methods

A comparison of several ML methods for diabetes prediction is shown in Table 10, which is compared to existing methods. Here is an analysis: Stacked Autoencoders (86.26% Accuracy) – As reported by K. Kannadasan et al. (2019) [14], the stacked autoencoder model performs well with a reasonable accuracy of 86.26%. This approach likely utilizes deep learning techniques suitable for complex data representations, but may not be as effective for relatively simpler datasets like PIMA. SVM- Based Models – Two studies employed Support Vector Machines (SVM) with different kernels: SVM with an RBF kernel was used by Ramesh et al. (2021) [20] to reach an accuracy of 83.20%, indicating a reasonable performance. SVM- RBF is often sensitive to kernel parameters and may require extensive tuning. Using a plain SVM model, Chatrati et al. (2022) [6] reported a lower accuracy of 74.00%, suggesting that the kernel choice significantly impacts model performance.

Ensemble Models – Ensemble methods show varying performance: Saxena et. al. (2023) [24] used an ensemble approach, achieving 86.00% accuracy, showing that combining multiple models can improve prediction performance. Noviyanti & Alamsyah (2024) [17] applied Random Forest with an accuracy of 87.00%, which suggests that this approach can handle complex relationships between variables effectively. En-RfRsK Ensemble Approach – Amma (2024) [3] employed a specific ensemble method (En-RfRsK) combining Random Forest, Radial SVM, and KNN, achieving 88.89% accuracy. The TPOT model proposed in this research achieved an accuracy of 92.00%, outperforming most other methods. The proposed ensemble approach combining LightGBM and KNN achieved 91.10% accuracy, highlighting the effectiveness of combining gradient boosting with simpler KNN models to capture complex interactions and local patterns.

With an accuracy of 94.00%, the Deep Learning-based model beats all other approaches, demonstrating neural networks' capacity to identify complex data patterns in this diabetes dataset. Compared to machine learning techniques, the suggested models-TPOT, LightGBM + KNN, and the deep learning-based model-show notable gains.

With the maximum accuracy of 94%, deep learning models in particular are more suited to capturing intricate patterns. However, ensemble methods still offer competitive performance, showing that combining diverse models often yields robust predictions.

Table 10. Comparison with leading-edge methods

References	Techniques	Test Accuracy (%)
K. Kannadasan et al. 2019 [14]	Stacked autoencoders	86.26
Ramesh et al., 2021 [20]	SVM-RBF	83.20
Chatrati et al., 2022 [6]	SVM	74.00
Saxena et al., 2023 [24]	Ensemble models	86.00
Noviyanti & Alamsyah, 2024 [17]	Random Forest	87.00
Amma 2024 [3]	Ensemble approach (En-RfRsK)	88.89
Proposed	LightGBM+KNN	91.10
Proposed	TPOT	92.00
Proposed	Deep Learning based	94.50

4.4. Statistical Analysis

Here we present a comparison of the performance of the three models by precision score, recall score, and F1-score using a paired t-test (Table 11). Table 12 presents the findings in terms of paired t-test findings.

Paired t-test results yield crucial insights: Precision: The ensemble learning model (LightGBM + KNN) is greatly outperformed by both the TPOT and Deep Learning models. Additionally, the Deep Learning model outperforms the TPOT model for Precision. Recall: There is no difference in Recall for the three models, indicating similar performance in identifying true positives across the models. F1-Score: The Deep Learning model significantly outperforms both the TPOT and ensemble learning models, while there is no statistical difference between the TPOT and ensemble learning models for this metric. In summary, the Deep Learning (DL) model consistently performs in all three metrics. It significantly outperforms both TPOT and the ensemble learning models in Precision and F1-score. While the TPOT model puts up a good fight against the ensemble learning approach, it falls short of the DL model in critical metrics. The ensemble learning model, while delivering reasonable performance, is outperformed by both TPOT and DL in these areas. However, the recall performance is tied across all models, indicating equal effectiveness in correctly identifying diabetic cases.

4.4.1. Interpretability of Model Decisions

To improve the model results' transparency and reliability, SHAP (SHapley Additive exPlanations) values were utilized to interpret how each feature contributes to the final prediction. SHAP shows feature importance by assigning a local explanation to each prediction, helping identify the impact of individual variables such as Glucose, BMI, Insulin, and Age across different patient records. Figure 6 presents the SHAP summary plot for the LightGBM+KNN model, highlighting how features contribute positively or negatively to the probability of a diabetes diagnosis. Glucose and BMI were consistently dominant in influencing model decisions, aligning with clinical expectations. Additionally, confidence scores for each prediction were recorded and evaluated. Predictions with low confidence can be flagged for further human review, thus supporting deployment in clinical settings with minimal risk.

4.5. Discussion on Outperformance Over Existing Methods

The enhanced functionality of the presented models - particularly the deep learning model with 94.5% accuracy-can be attributed to a combination of advanced data preprocessing, robust model architectures, and novel feature engineering strategies. Unlike earlier works, which primarily used standard features from the PIMA dataset, this study introduced five new engineered features (Section 3.4) based on domain-specific interactions (e.g., Glucose-to-Blood Pressure Ratio, BMI × Insulin), which enhanced model representation power and reduced multicollinearity. Furthermore, many previous works (e.g., Raafat et al., 2021; Ramesh et al., 2021) suffered from limited or inconsistent imputation strategies and did not handle class imbalance effectively. In contrast, our approach used median-based class-wise imputation (Table 3), which is robust to outliers, and applied SMOTE for dataset balancing (Section 3.4), leading to better generalization. The integration of model optimization tools such as TPOT ensured automatic selection of hyperparameters and model pipelines (Section 4.3.2), outperforming hand-tuned conventional models like SVM and RF. Similarly, the ensemble model combined the strengths of LightGBM's gradient boosting and KNN's instance-based learning, enhancing both interpretability and prediction accuracy (Section 4.3.1). Importantly, a statistical t-test (Section 4.4) confirms that the improvements in precision and F1-score are statistically significant ($p < 0.05$) when compared to prior works reporting accuracies between 74% and 89% (Table 10).

Table 11. Mean precision, recall, and F1-score for TPOT, deep learning, and ensemble learning (light GBM + KNN)

Metric	Mean (TPOT)	Mean (Deep Learning)	Mean (Ensemble)
Precision	0.915	0.95	0.893
Recall	0.915	0.945	0.942
F1-Score	0.915	0.945	0.916

Table 12. Paired t-test results for precision, recall, and F1 score comparisons between models

Comparison	t - Value	p - Value
Precision Comparison		
TPOT-Ensemble	2.14	0.038 (significant, $p < 0.05$)
DL-Ensemble	3.51	0.009 (significant, $p < 0.05$)
TPOT-DL	3.13	0.015 (significant, $p < 0.05$)
Recall Comparison		
TPOT-Ensemble	0.89	0.398 (not significant, $p > 0.05$)
DL-Ensemble	1.02	0.312 (not significant, $p > 0.05$)
TPOT-DL	1.67	0.092 (not significant, $p > 0.05$)
F1-Score Comparison		
TPOT-Ensemble	1.98	0.058 (not significant, $p > 0.05$)
DL-Ensemble	3.24	0.015 (significant, $p < 0.05$)
TPOT-DL	2.94	0.022 (significant, $p < 0.05$)

The DL model, with an AUC of 0.984, also demonstrated superior class separation capabilities (Figure 14), especially critical in high-risk populations like the Pima Indians, where misclassification has severe health implications. These methodological enhancements and validation outcomes jointly explain the consistent outperformance over existing methods reported in recent literature (2019–2024).

4.5.1. Implications for Healthcare Policy

The predictive accuracy achieved in this study (up to 94.5%) supports its potential use in public health screening and early diagnosis strategies. Such AI-driven tools can be integrated into healthcare infrastructure, particularly in low- and middle-income regions where early intervention remains critical yet underutilized. By incorporating these models into mobile diagnostic apps or Electronic Health Record (EHR) systems, policymakers can implement targeted screening programs, prioritize high-risk populations, and reduce long-term complications and healthcare costs associated with unmanaged diabetes. Additionally, the adoption of interpretable models (e.g., ensemble or TPOT pipelines) ensures transparency and builds trust in clinical deployment.

5. Conclusion

In conclusion, this work uses cutting-edge ML and DL approaches to thoroughly examine models for diabetes prediction. This study uses DL and sophisticated ML techniques to predict diabetes in the Pima Indian population. Extensive data preprocessing was used to improve model performance, including feature selection, feature engineering, and imputation of missing values.

Statistical methods addressed missing data, identified significant features, balanced the dataset, and engineered new features to improve prediction. The best-performing models in the proposed analysis had excellent F1-score, precision score, recall score, and 94.5% accuracy, demonstrating strong performance across all criteria. The promise of early prediction in managing diabetes in at-risk populations is highlighted by this outperformance of existing approaches, giving hope for better strategies for the management of diabetes. The future work is dedicated to reducing overfitting by incorporating regularization techniques, advanced data augmentation, and expanding the dataset. The aim is to refine feature selection and enhance model interpretability to ensure robust performance and applicability in clinical settings. These ongoing improvements reflect a continued commitment to advancing diabetes prediction models.

5.1. Future Refinements and Broader Applicability

Future refinements to the proposed framework could include integrating additional demographic or behavioral variables such as diet, activity level, and family history, thereby enhancing the model's contextual relevance. Incorporating cost-sensitive or domain-adaptive learning techniques could further improve generalizability across populations. Beyond the Pima Indian cohort, the model holds promise as a clinical decision support tool in community health centers, telemedicine platforms, and mobile health applications, particularly in resource - constrained environments. To extend applicability, retraining on new and different datasets from other ethnicities is essential to ensure

fair and effective predictions across ethnic and regional boundaries.

5.2. Enhancing Trust and Transparency

For better transparency and trustworthiness of the proposed models, we used SHAP-based feature attribution to explain how input variables contribute to individual predictions.

Such post-hoc interpretability methods enable practitioners to validate whether model behavior aligns with medical reasoning. For real-world deployment, integrating model confidence scores and logging decision paths can further support clinical trust. Future work may include applying LIME for instance-level explanations and integrating human-in-the-loop frameworks for semi-automated oversight.

References

[1] Ahmed F. Ashour et al., "Optimized Neural Networks for Diabetes Classification Using Pima Indians Diabetes Database," *2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI)*, Mt Pleasant, MI, USA, pp. 1-7, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[2] Mohammed Awad, and Salam Fraihat, "Recursive Feature Elimination with Cross-Validation with Decision Tree: Feature Selection Method for Machine Learning-Based Intrusion Detection Systems," *Journal of Sensor and Actuator Networks*, vol. 12, no. 5, pp. 1-23, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] N.G. Bhuvaneswari Amma, " En-RFRSK: An Ensemble Machine Learning Technique for Prognostication of Diabetes Mellitus," *Egyptian Informatics Journal*, vol. 25, pp. 1-8, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[4] Jiawei Han, Micheline Kamber, and Jian Pei, *Data Mining: Concepts and Techniques*, 3rd ed., Morgan Kaufmann Publishers, 2011. [Google Scholar] [Publisher Link]

[5] Victor Chang et al., "Pima Indians Diabetes Mellitus Classification Based on Machine Learning (Ml) Algorithms," *Neural Computing and Applications*, vol. 35, no. 16, pp. 16157-16173, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Saiteja Prasad Chatrati et al., "Smart Home Health Monitoring System for Predicting Type 2 Diabetes and Hypertension," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 3, pp. 862-870, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Nitesh V. Chawla et al., "Smote: Synthetic Minority Over Sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321-357, 2002. [CrossRef] [Google Scholar] [Publisher Link]

[8] Pima Indians Diabetes Database, Kaggle. [Online]. Available: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

[9] Himanshu Gupta et al., "Comparative Performance Analysis of Quantum Machine Learning with Deep Learning for Diabetes Prediction," *Complex & Intelligent Systems*, vol. 8, no. 4, pp. 3073-3087, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] *IDF Diabetes Atlas*, International Diabetes Federation, pp. 1-143, 2011. [Google Scholar] [Publisher Link]

[11] Reza Iranzad, and Xiao Liu, "A Review of Random Forest-Based Feature Selection Methods for Data Science Education and Applications," *International Journal of Data Science and Analytics*, vol. 20, no. 2, pp. 197-211, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[12] Vishesh Jain, Sanyam Shukla, and Nilay Khare, "Analysis of Various Data Imputation Techniques for Diabetes Classification on Pima Dataset," *2024 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, Bhopal, India, vol. 35, pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Hyun Kang, "The Prevention and Handling of the Missing Data," *Korean Journal of Anesthesiology*, vol. 64, no. 5, pp. 402-406, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[14] K. Kannadasan, Damodar Reddy Edla, and Venkatanareshbabu Kuppili, "Type 2 Diabetes Data Classification using Stacked Autoencoders in Deep Neural Networks," *Clinical Epidemiology and Global Health*, vol. 7, no. 4, pp. 530-535, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[15] Jobeda Jamal Khanam, and Simon Y. Foo, "A Comparison of Machine Learning Algorithms for Diabetes Prediction," *ICT Express*, vol. 7, no. 4, pp. 432-439, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[16] Yasunobu Nohara et al., "Explanation of Machine Learning Models Using Shapley Additive Explanation and Application for Real Data in Hospital," *Computer Methods and Programs in Biomedicine*, vol. 214, pp. 1-7, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Cindy Nabila Noviyanti, and Alamsyah Alamsyah, "Early Detection of Diabetes using Random Forest Algorithm," *Journal of Information System Exploration and Research*, vol. 2, no. 1, pp. 41-48, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[18] Randal S. Olson, and Jason H. Moore, "Tpot: A Tree-Based Pipeline Optimization Tool for Automating Machine Learning," *Proceedings of Machine Learning Research (PMLR)*, vol. 64, pp. 66-74, 2016. [Google Scholar] [Publisher Link]

[19] Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren, *Automated Machine Learning*, 2019. [CrossRef] [Google Scholar] [Publisher Link],

[20] Jayroop Ramesh, Raafat Aburukba, and Assim Sagahyroon, "A Remote Healthcare Monitoring Framework for Diabetes Prediction Using Machine Learning," *Healthcare Technology Letters*, vol. 8, no. 2, pp. 45-57, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] Md Shamim Reza et al., "Improving Diabetes Disease Patients Classification Using Stacking Ensemble Method with Pima and Local Healthcare Data," *Heliyon*, vol. 10, no. 2, pp. 1-13, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] M. Jishnu Sai et al., "An Ensemble of Light Gradient Boosting Machine and Adaptive Boosting for Prediction of Type-2 Diabetes," *International Journal of Computational Intelligence Systems*, vol. 16, no. 1, pp. 1-20, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[23] Merdin Shamal Salih et al., "Diabetic Prediction Based on Machine Learning Using Pima Indian Dataset," *Communications on Applied Nonlinear Analysis*, vol. 31, no. 5s, pp. 138-156, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[24] Surabhi Saxena et al., "Machine Learning Algorithms for Diabetes Detection: A Comparative Evaluation of Performance of Algorithms," *Evolutionary Intelligence*, vol. 16, no. 2, pp. 587-603, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25] Ahmad S. Tarawneh et al., "CTELC: A Constant-Time Ensemble Learning Classifier Based on KNN for Big Data," *IEEE Access*, vol. 11, pp. 89791-89802, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[26] Linshan Xie, "Pima Indian Diabetes Database and Machine Learning Models for Diabetes Prediction," *Highlights in Science, Engineering and Technology*, vol. 88, pp. 97-103, 2024. [CrossRef] [Google Scholar] [Publisher Link]