JOURNAL OF SOFT COMPUTING AND DATA MINING
e-ISSN: 2716-621X

JSCDM

Vol. 6 No. 1 (2025) 1-20
https://publisher.uthm.edu.my/ojs/index.php/jscdm

Diabetes Prediction Through Classification Using Pima Dataset: Survey and Evaluation

Ahmad Adel Abu-Shareha[1]*, Mosleh M. Abualhaj[2], Mohammad A. Alsharaiah[1], Adeeb Al-Saaidah[2], Anusha Achuthan[3]

[1] *Department of Data Science and Artificial Intelligence, Al-Ahliyya Amman University, Amman, 19328, JORDAN*
[2] *Department of Networks and Cybersecurity, Al-Ahliyya Amman University, Amman, 19328, JORDAN*
[3] *School of Computer Sciences, Universiti Sains Malaysia, Gelugor, Penang 11800, MALAYSIA*

*Corresponding Author: a.abushareha@ammanu.edu.jo
DOI: https://doi.org/10.30880/jscdm.2025.06.01.001

Article Info

Received: 27 July 2024
Accepted: 8 May 2025
Available online: 30 June 2025

Keywords

Diabetes, prediction, diagnosis, prognosis, dataset, classification, evaluation, Pima

Abstract

Diabetes prediction using machine learning techniques has been extensively investigated in the literature, resulting in diverse prediction and evaluation approaches. This diversity often leads to inconsistent comparisons between these approaches. This paper offers a comprehensive survey of state-of-the-art diabetes prediction through classification, encompassing various preprocessing techniques and machine learning methods. As such, the primary objectives of this paper are as follows: 1) To analyze the performance of existing machine learning methods and trace the advancements in diabetes prediction outcomes over time. 2) To establish a baseline for evaluating and benchmarking different diabetes prediction approaches. 3) To assess the performance of common machine learning methods. 4) To propose future research directions to enhance diabetes prognosis methodologies further. 5) To provide state-of-the-art results for the performance of the common machine learning methods on the Pima dataset. The review outcomes show significant variations in the existing prediction and evaluation approaches. The results of the proposed evaluation approach showed that scaling, feature selection, and over-sampling improve the results of the prediction approaches. Besides, the results of the machine learning techniques varied, with the best results mostly achieved by the random forest algorithm.

1. Introduction

The development of data collection techniques, database management systems, and medical data warehousing approaches have formed a robust infrastructure for computer-based medical case processing and analysis [1-3]. Electronic Medical Records (EMR) systems introduced in the 1960s offer an electronic way of medical record-keeping [4], which influences medical diagnosis, prognosis, and treatment [5]. EMR systems enable efficient data management and quick medical record retrieval and provide healthcare professionals with seamless data access and sharing capabilities. Besides, EMR systems allow the sharing of data and facilitate easy access to information. Moreover, EMR systems enable the development of customized disease-specific datasets that provide essential information for computer-based medical diagnosis of various health conditions [6,7].

The development of EMR systems and datasets has been driven by data mining algorithms that enable powerful analysis of content, pattern discovery, and information extraction. Besides, the value of these systems has increased due to technological advancements and data collection across various domains and the concept of 'big data' and its machine learning applications. Clinical systems and specialized datasets contain rich information, including patients' demographic details, historical records, and laboratory results, offering valuable insights that can be automatically gleaned to enhance disease understanding [8,9].

Data mining and machine-learning techniques, such as data clustering, and classification, are used to analyse the relations and patterns [10]. In healthcare, data mining is used for processing and analyzing data, irrespective of data structure or format, including structured and unstructured data. However, processes and techniques used to analyze healthcare data can vary significantly, as shown in Fig. 1 [10]. Data mining applications in the medical domain are extensive; for example, prognosis development can be revealed by mining the hidden patterns through frequent pattern mining or association rules mining [11,12], critical risk factors can be identified using feature selection [13-15], disease models can be created using clustering [16], and prediction can be performed through classification [17].

Diabetes, one of the most common and serious diseases globally, is a condition linked to insulin, a hormone that controls blood sugar. Diabetes Type I occurs when the pancreas produces insufficient insulin, while Type II occurs when the produced insulin cannot be used effectively. Type I mainly affects children and youths due to genetic disorders, while type II affects adults over 40 due to high blood sugar. Pregnancy diabetes, retinopathy, and neuropathy are other diabetes types [18]. Generally, uncontrolled insulin leads to elevated blood sugar, damaging body systems, especially the kidneys, heart, and lower limbs. Moreover, diabetes can cause blindness, stroke, damage to blood vessels and nerves, and, in severe cases, limb amputation [19]. According to the World Health Organization (WHO), diabetes prevalence in low- and middle-income countries has been rapidly increasing. In 2016, diabetes was responsible for 1.6 million deaths, ranking as the seventh leading cause of mortality worldwide [20]. The number of adults with diabetes grew from 463 million in 2019 to 537 million in 2021 [21].

The diagnosis of diabetes has greatly benefited from the data stored in clinical systems and medical datasets. Various health institutions, research organizations, and collaborative bodies are dedicated to developing automated diagnosis and prognosis systems for diabetes, with the ultimate goal of saving lives. Early and precise diagnosis is a significant contribution to the fight against diabetes. Thus, detecting and diagnosing diabetes is crucial to promote public health and well-being [22]. Prognosis and diagnosis are complex processes that rely on expertise within the healthcare field. Early diagnosis of diabetes hinges on medical knowledge and experience applied by physicians. Automated diagnosis involves collecting raw data from diabetes patients' diagnostics and then using data mining techniques for analysis [23]. Successful diagnosis depends on two main factors: 1) the availability and maintenance of diabetes datasets containing relevant factors and features that reveal hidden information and hidden patterns, and 2) the use of data mining techniques and algorithms for ongoing analysis and classification of raw data [24].

The efficiency of diabetes prediction applications has advanced with the expansion of datasets, simplifying implementation, and testing of diagnosis techniques. A primary focus is on applying classification algorithms to customized datasets [25], leveraging well-researched features that enhance diagnosis accuracy and build confidence in automated diabetes prediction systems. These diabetes datasets encompass diverse features varying in size, type, and range according to specific diagnostic and prognostic criteria. A key distinction is in the output type, requiring either classification (e.g., Yes/No) or regression (e.g., stage as real values) for prediction. Additionally, data mining techniques for diabetes prognosis are diverse, using various techniques, such as classification, feature selection, and preprocessing. This field has a well-established history, with various algorithms like linear regression (LinReg), artificial neural networks (NN), and support vector machines (SVM).

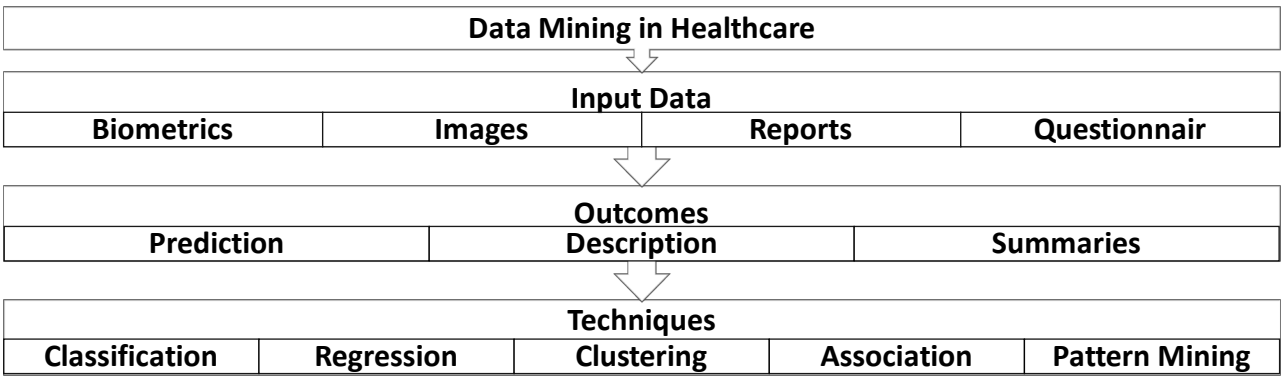

Fig. 1 *Components of data mining applications in healthcare*

The vast number of approaches in automatic diabetes prognosis and the differences in their techniques and results have urged the need to review, categorize, and evaluate these approaches to foster innovation and improve diagnostic outcomes. This paper provides a structured framework for benchmarking various diabetes prediction techniques, which are used to survey and assess the state-of-the-art methods in diabetes diagnosis through classification. The conducted survey offers a clear understanding of the strengths and limitations of the reviewed approach, allowing for more informed decisions in diagnosis applications. The aims of this paper are as follows: 1) To assess and examine the effectiveness of diabetes prediction through classification. 2) To establish a baseline for evaluating and benchmarking different diabetes prediction approaches. 3) To assess the performance of common machine learning methods. 4) To propose future research directions to enhance diabetes prognosis methodologies further. 5) To provide state-of-the-art results for the performance of the common machine learning methods on the Pima dataset. The rest of this paper is organized as follows: Section 2 reviews the existing surveys on diabetes prediction. Section 3 discusses the components of the prediction approaches, including datasets, preprocessing, feature selection, balancing, and classification. Section 4 presents an overview of existing approaches and their results. Section 5 provides an empirical comparison, and the conclusion is given in Section 6.

2. Related Work

Several surveys and comparative studies have delved into machine learning algorithms for predicting diabetes. For instance, Larabi-Marie-Sainte et al. [26] reviewed machine learning and deep learning approaches within a specific timeframe (2013-2019). The datasets, the utilized techniques, and the outcomes of these approaches were reported. According to their findings, all machine learning algorithms were applied to the Pima Indian diabetes dataset (PIDD), achieving accuracies ranging from 68.23% to 74.48%. In contrast, deep learning approaches reached a higher accuracy of 95%, although these results were reported for different datasets. An empirical evaluation was carried out on seldom-used and underexplored machine learning classifiers for diabetes prediction, achieving an accuracy of 74.48%. Such meta-analysis studies provide comprehensive summaries of the existing approaches and their development through diverse machine-learning algorithms. Furthermore, they enable the identification of advancements over time.

Some surveys have investigated the potential of preprocessing and feature selection alongside machine learning algorithms. As such, Khan et al. [27] reviewed and categorized diagnostic and predictive approaches. Based on their conclusions, the performance of diabetes prediction approaches depends on factors such as dataset, preprocessing methods, and feature selection. Consequently, data preprocessing and hybrid methods were recommended to enhance disease detection accuracy. Effective preprocessing, which includes dimensionality reduction, de-noising, feature selection, and extraction, complements classification and prediction, thereby improving performance. Similarly, Jaiswal et al. [28] reviewed and categorized diabetes diagnosis and prediction approaches based on the evaluation datasets. The survey indicated that techniques varied in accuracy across different datasets, consistent with previous surveys. These surveys provided a comprehensive overview of the prediction components and their impact on the output accuracy performance.

Chaki et al. [29] surveyed the use of machine learning for diabetes prediction, with a focus on datasets. The survey examined structured and image-based datasets, the features extracted, preprocessing steps, classification approaches, and performance measures. Besides, the classification results from reviewed studies were gathered and compared. Kodama et al. [30] conducted a survey to assess the literature on using various machine learning models for diabetes prediction, particularly focusing on classification methods using the PIDD. Results from reviewed studies were collected and compared. Similarly, Mohsen et al. [31] reviewed the literature on machine learning applications for diabetes prediction using diverse data sources, including images and reports. Instead of comparing performance on specific datasets, the machine learning algorithms, features, and performance measures utilized in the reviewed literature were compared. A comparison and comprehensive overview of these reviews and meta-analyses is provided in Table 1.

Table 1 *A comparison of the related work*

Ref.	Survey Approaches	Trends Overview	Benchmarking using PIDD Dataset
Larabi-Marie-Sainte et al. [26]	Yes	No	Limited
Khan et al. [27]	Yes	No	No
Jaiswal et al. [28]	Yes	No	No
Chaki et al. [29]	Yes	Yes	No
Kodama et al. [30]	Yes	No	No
Mohsen et al. [31]	Yes	No	No
proposed	Yes	Yes	Yes

Building on these efforts, this paper aims to deepen the comparative analysis of algorithms and components for diabetes prediction. Consequently, common machine learning methods are assessed across the PIDD, and their results are compared to address existing gaps in the literature. This study aims to make a meaningful contribution to diabetes prediction research by systematically evaluating these algorithms and analyzing their predictive performance.

3. Diabetes Prediction Framework Components

Diabetes prediction using classification typically follows a multi-stage process, as illustrated in Fig. 2. Initially, datasets are preprocessed to improve data quality and ensure consistency. After preprocessing, balancing techniques, such as oversampling or undersampling, and feature selection are applied to improve model performance. In workflows using oversampling, preprocessing is followed by oversampling and then feature selection, while in undersampling, feature selection is performed first. The refined data is then partitioned before being fed into various classification algorithms. Finally, the performance of each algorithm is thoroughly evaluated using selected performance metrics, taking the dataset partitioning model into account.

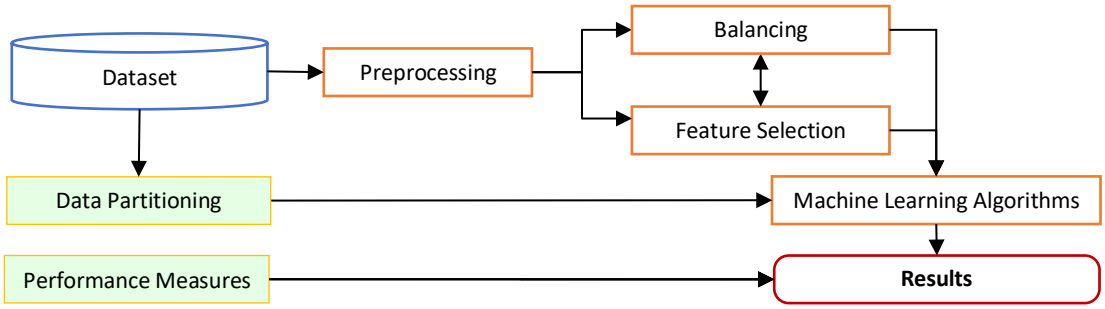

Fig. 2 *Diabetes prediction framework*

3.1 PIDD Dataset

Typically, diabetes is identified through a range of laboratory tests and assessments, including diastolic blood pressure, plasma glucose concentration, and blood serum levels (such as insulin). Additionally, symptoms and demographic factors, such as genetic predisposition, age, obesity, and environmental influences, play a role in diagnosis [32]. Various other elements influence diagnosis, prognosis, and treatment. As a result, these factors are used as features for diabetes prediction. Among structured datasets, three are commonly referenced: PIDD [25], the Diabetes Progression Dataset (DPD) by Efron et al. [33], and the Early Risk (Sylhet Hospital Diabetes) dataset [34]. PIDD, unlike the other datasets, presents challenges due to its limited samples and features; accordingly, various methods have been developed to process this dataset.

PIDD was collected from the Pima Indian female population due to their significantly high risk of diabetes, established PIDD. The community's residents have diabetes rates 19 times higher than those of a typical town in Minnesota [35]. Residents of the target community underwent oral glucose tolerance tests, and a diabetes diagnosis was given if their plasma glucose level was at least 200 mg/dL. Variables such as age, blood pressure, and body mass are included in the dataset for diabetes classification. The dataset comprises 500 non-diabetic and 268 diabetic women, totaling 768 samples and 8 features. These features are listed in Table 2 [36].

J. of Soft Computing and Data Mining Vol. 6 No. 1 (2025) p. 1-20

Table 2 *PIDD description*

#	Feature	Description	Type	Range
1	Pregnant	Number of pregnant times	Numeric	0–17
2	Plasma–glucose	Plasma–glucose concentration		0–199
3	Diastolic	Diastolic blood pressure		0–122
4	Triceps	Triceps skin fold thickness		0–99
5	Serum–insulin	Serum–insulin 2-h		0–846
6	Body mass	Weight		0–67.1
7	DPF	DPF Diabetes pedigree function		0.078–2.42
8	Age	Age in years		21–81
Class	Diabetes	Diabetes diagnostic	Nominal	Yes/No

The dataset was donated by Vincent Sigillito, RMI Group Leader at the Applied Physics Laboratory, The Johns Hopkins University, Laurel, MD. The data is publicly available through repositories such as Data World and other archival sources.

3.2 Preprocessing Stage

Preprocessing stages vary depending on dataset characteristics, with normalization being a common step to standardize the feature ranges. Normalization improves the stability and convergence speed of classification algorithms, such as SVM and k-nearest neighbors (KNN). Besides, normalization ensures fair feature contributions and reduces biases from scale variations [37]. Various normalization techniques exist, including Min-Max scaling, which scales the feature values between 0 and 1; robust scaling, which reduces the impact of outliers by scaling the values based on the median and interquartile range; and z-score normalization, which standardizes values to a mean of 0 and standard deviation of 1. Another preprocessing step is feature encoding, which converts categorical variables into numerical representations (e.g., one-hot encoding), allowing algorithms to interpret categorical data effectively [38]. Finally, filling in missing values using imputations is used to maintain dataset completeness. Imputation can be implemented using methods such as mean, median, and mode for simple cases or advanced imputation techniques such as KNN and multiple imputation by chained equations (MICE) for more complex data [39]. A summary of these preprocessing steps is given in Table 3.

Table 3 *Machine-learning preprocessing steps*

	Aim	Advantage(s)	Challenges(s)
Normalization	Standardize feature value ranges.	Enhances model stability and ensures fair contribution.	Difficult to choose the appropriate method.
Feature Encoding	Convert categorical variables to numerical ones.	Enables algorithms to process categorical features and facilitates accurate interpretation.	Introduce high dimensionality and loss of correlations.
Handling Missing Values	Address missing data to prevent biases.	Maintains dataset integrity.	Introduce bias.

3.3 Dimensionality Reduction

Feature selection and dimensionality reduction are two approaches that address the challenges posed by high-dimensional datasets, including increased computational complexity, overfitting, and reduced model interpretability. While both aim to simplify data representation, they differ in methodology and focus. Feature selection narrows down the dataset to the most relevant features without altering the feature space itself. In contrast, dimensionality reduction techniques transform the data into another representation that reduce their dimensionality but preserves the essential characteristics. Feature selection identifies the most relevant features for a predictive task, removing those deemed redundant or irrelevant. This approach can be categorized into filter-based, wrapper-based, and embedded-based methods [40].

Filter-based methods evaluate individual features independent of any learning algorithm, using statistical measures such as variance, correlation, or information gain. For example, variance thresholding removes low-variance features that likely contribute little information to the model, and correlation-based methods, such as Chi-Square, eliminate features that are highly correlated with others. Although filter-based techniques are fast and scalable, they may overlook complex feature interactions, making them best suited for high-dimensional datasets where computational resources are constrained [41].

Wrapper-based methods assess feature subsets by integrating them into the model evaluation process. Recursive Feature Elimination (RFE), for instance, iteratively removes the least important features based on model performance until an optimal subset is achieved. Forward and backward selection strategies similarly add or remove features sequentially. While wrapper-based methods can capture feature interactions, they are computationally intensive and highly overfit in large feature spaces. However, they are particularly useful in tasks where feature interactions influence performance [42].

Embedded-based feature selection is a method where feature selection is integrated directly into the process of model training. These methods typically involve algorithms that automatically select features as part of the model-building process, considering the importance or relevance of each feature for the predictive task at hand. Common embedded feature selection techniques include Lasso, Elastic Net, and SVM. Embedded feature selection methods are advantageous because they select features directly relevant to the predictive task, potentially improving model performance and generalization. Additionally, they streamline the modeling process by integrating feature selection and model training into a single step [43].

Dimensionality reduction differs from feature selection by transforming the original feature space into a lower-dimensional representation while preserving essential data structures. Techniques such as Principal Component Analysis (PCA) and t-distributed Stochastic Neighbor Embedding (t-SNE) are commonly used for this purpose.

3.4 Balancing

Balancing data is an essential step for dealing with datasets where one class is significantly underrepresented compared to the other. Such class imbalances can lead to biased models that perform poorly on the minority class. Balancing can be implemented using both oversampling and undersampling techniques. Oversampling is typically applied before feature selection, while undersampling is more often used after feature selection and dimensionality reduction. Undersampling for data balancing involves reducing the number of majority class samples, either through random elimination or by selecting samples based on their relevance or contribution to the model. Conversely, oversampling increases the number of samples in the minority class with either duplicating samples or synthetically generated samples. One common oversampling method is the Synthetic Minority Oversampling technique (SMOTE) [44]. SMOTE balances the class distribution by generating synthetic samples for the minority class through interpolation between existing instances. SMOTE avoids sample duplication and reduces the risk of model overfitting. However, it is crucial to apply SMOTE solely to the training data to prevent data leakage and ensure that the model's performance is validated on the original dataset without any syntactic samples [45].

3.5 Classification Algorithms

There are various machine learning algorithms proposed to address the diverse nature of real-world data and the complexity of underlying problems. No single algorithm is universally superior; each comes with its own set of assumptions, strengths, and limitations. Some algorithms are more effective at managing high-dimensional data or non-linear relationships. In contrast, others focus on interpretability or computational efficiency, and different algorithms can provide complementary benefits when combined to form ensemble methods. Therefore, testing various algorithms is crucial to identify the most suitable algorithm based on the processed data.

KNN is a straightforward and intuitive algorithm that assigns a class label for a new instance as the majority class among the K nearest neighbors. KNN does not require explicit training and can adapt to complex decision boundaries. However, KNN performs poorly with high-dimensional data and imbalanced class distributions, and its prediction cost can be substantial for large datasets [46]. Similarly, the Nearest Centroid Classifier (NC), or Rocchio classifier, assigns a class label for a new instance as the class of the nearest centroid. The NC calculates the centroid of each class by averaging its feature vectors and assigning the class label. The simplicity of NC and the KNN and their interpretability make such classifiers particularly well-suited for tasks like text categorization and document clustering [47].

Naive Bayes (NB) is a probabilistic classifier based on Bayes' theorem and inforced assumption of feature independence. Despite this assumption, NB performs effectively in a range of real-world classification tasks, especially when dealing with high-dimensional and sparse data [48]. Logistic Regression (LogReg) is a linear classifier that models the classification problem using a logistic function. This algorithm estimates the relationship between the independent variables and the class label using the binary output function, making it well-suited for binary classification tasks. Although LogReg is straightforward, it performs well only when the decision boundary is linear or close to linear [49].

SVM finds the optimal hyperplane that separates two classes in the feature space while maximizing the margin between them. As such, SVM is effective in handling high-dimensional data and is robust against overfitting, especially in cases where the number of features exceeds the number of samples. SVM can manage non-linear relationships through the use of kernel functions, allowing them to learn complex decision boundaries. As such,

SVM is widely used in various fields, including image classification, text categorization, bioinformatics, and other unsupervised machine-learning tasks [50].

Decision Tree (DT) is a non-parametric algorithm that constructs a classification tree that divides the feature space into a hierarchical structure of decision nodes. Each leaf node stands for a regression value in the regression task or a class label in the classification task, with each node representing a feature. Such an algorithm has the advantage of being intuitive and interpretable. DT algorithm is prone to overfitting, especially when dealing with intricate datasets. Pruning and ensemble techniques like Random Tree (RT) and Random Forest (RF) help reduce overfitting and enhance Decision Tree performance [51]. RT is an ensemble learning technique that constructs several decision trees during training. Each decision tree is constructed using a random subset of the training data and a random subset of the features to diversify the trees. As such, RT is used with large datasets and high-dimensional feature spaces for its computational efficiency. Besides, DT is used to construct robust ensemble techniques like Gradient Boosting Machines (GBM) and RF. Multiple decision trees are combined in RF, with a random subset of features chosen at each split, and each tree in the forest is trained using a bootstrap sample of the training data. When making predictions, the RF's output is calculated by combining the predictions of each tree using voting and averaging techniques. RF provides estimates of feature importance, manages high-dimensional data effectively, and resists overfitting [52]. An ensemble learning method called GBM builds a sequence of decision trees one after the other. By fitting each tree to the residual errors of the preceding trees, GBM minimizes a loss function and produces stronger models. Consequently, this model is a popular option in competitions and real-world applications due to its strong predictive performance and resilience to overfitting [53].

An ensemble learning technique called Adaptive Boosting (AdaBoost) builds a strong classifier by combining several weak classifiers. It focuses on cases that were incorrectly classified by earlier models and train several weak learners in succession on altered datasets. AdaBoost iteratively improves classification accuracy by giving these incorrectly classified instances more weight. AdaBoost exhibits resilience against overfitting and is especially useful for binary classification tasks [54].

The term "deep learning" (DL) describes a family of artificial neural network (ANN) architectures with several hidden layers that make it possible to extract intricate patterns and representations from unprocessed data. Although deep learning models provide cutting-edge performance across a range of domains, their training necessitates significant computational resources and large amounts of data [55]. These algorithms offer diverse tools for solving the classification problem across various domains, each with strengths and weaknesses. A comparison between these algorithms is given in Table 4.

Table 4 *A summary of the classification algorithms*

Classifier	Description	Pros	Cons
KNN	Based on the majority class of the *nearest neighbors.*	Intuitive, no training required, and adaptable to complex decision boundaries.	Sensitive to noise and irrelevant features and computationally expensive for large datasets.
NB	Based on Bayes' theorem with the assumption of independence.	Efficient, simple, and suitable for high-dimensional and sparse data.	Perform poorly if the independence assumption is violated.
NC	Based on the nearest centroid.	Simple and interpretable.	Sensitivity to feature scaling.
SVM	Build a hyperplane to separate classes in feature space.	Effective for high-dimensional and non-linear data and robust against overfitting.	Computationally expensive for large datasets.
DT	Build a hierarchical structure of decision nodes.	Intuitive, interpretable, implements classification and regression, and can be used for feature selection.	Prone to overfitting and poorly performed with imbalanced data.
RT	Using an ensemble with multiple decision trees with randomness in feature and data sampling.	Reduces overfitting, computationally efficient, and easy to parallelize.	Less interpretable than individual decision trees.
RF	Ensemble with multiple decision trees.	Resistant to overfitting and provides estimates of feature importance.	Less interpretable and computationally expensive.

GBM	Ensemble of decision trees sequentially that minimize a loss function.	Robust against overfitting.	Computationally expensive, sensitive to noisy data and outliers, and requires parameter tuning.
AdaBoost	Ensemble of multiple weak classifiers.	Robust to overfitting.	Sensitive to noise and computationally expensive.
LogReg	Using a logistic function.	Simple and interpretable.	Assumes a linearity and is sensitive to outliers.
Elastic Net	Combines L1 and L2 regularization penalties.	Handles multicollinearity and high-dimensional data.	Requires parameter tuning and not suitable for highly correlated data.
DL	Composed of interconnected nodes.	Effective in capturing complex patterns and can handle high-dimensional data with noise and outliers.	Computationally expensive and prone to overfitting.

3.6 Evaluation Measurements

Different evaluation metrics are employed to assess the outcomes of diabetes prediction. Among many others, accuracy, precision, recall, and f-measure are widely employed in classification evaluations. These metrics' computations are outlined in Table 5.

Table 5 *Evaluation measures*

Measure	Calculation	Description	Problem
Accuracy	$accuracy = \frac{TP+TN}{TP+TN+FP+FN}$	The ratio of correctly predicted output to the total number of samples to be predicted.	Classification
Precision (Specificity)	$precision = \frac{TP}{TP+FP}$	The ratio of correctly predicted positive samples to the total positives predicted.	Classification
Recall (Sensitivity)	$recall = \frac{TP}{TP+FN}$	The ratio of correctly predicted positive samples to the total positives in the samples.	Classification
F-Measure	$Fmeasure = \frac{2 * precision * recall}{precision + recall}$	A combination of precision and recall.	Classification

3.7 Data Partitioning

Data splitting divides the dataset into training, validation, and testing sets, allowing the model to be trained on one subset, validated on another, and tested on a further unknown portion. This approach guarantees a dispassionate evaluation of new information. Two common techniques for data splitting are cross-validation and percentage split. Cross-validation divides the dataset into k folds, or equal-sized subgroups. To train the model, k−1 folds are used, while the remaining fold is used for testing. The model's performance measures are averaged over all folds to provide a more accurate assessment. In contrast, a percentage split divides the dataset into two subsets: a training set and a testing set. The model is trained on the training set and evaluated on the testing set. In contrast, by testing the model on many subsets of the data, cross-validation produces a more accurate assessment of the model's performance. This approach reduces the impact of data variability and ensures that the model's performance is independent of a random split. These strategies are contrasted in Table 6.

J. of Soft Computing and Data Mining Vol. 6 No. 1 (2025) p. 1-20

Table 6 *Comparison of the data partitioning techniques*

Aspect	*Cross-Validation*	*Percentage Split*
Splitting	*k* folds for training and testing	Training and testing subsets
# Runs	Trains model k times	Trains model only once
Model Evaluation	Averaged across all folds	A single testing set
Robustness	More robust	Less robust
Cost	Higher computational cost	Lower computational cost
Sensitivity	Less sensitive to data variability due to multiple folds	Impacted by data variability depending on the split

4. Literature Review on the Diabetes Prediction Approaches

Various approaches were developed using variations of the machine learning algorithms and components. Here is a review of these approaches.

One of the earliest approaches for diabetes conducted on the PIDD was presented by Smith et al. [25] using ANN with 576 instances randomly selected for training with 192 test samples. No preprocessing or post-processing was implemented in this approach except for the discretization of the continuous output. The value of 0.48 was chosen experimentally as a threshold. All output values lower than 0.48 are classified as negative diabetes, while the rest are positive. The results reported based on the sensitivity and specificity were equal to 76%. Similarly, Shanker [56] proposed an approach based on ANN on a variation of the dataset as the one utilized by Smith et al. [25]. The data samples were selected from the dataset based on constraints, such as the age above 21. Accordingly, there were 768 cases, of which 268 cases were diagnosed with diabetes. The approach used a different number of hidden nodes, different activation functions, and variations of the input feature set, which achieved an accuracy of 81%. Using the same dataset, Carpenter and Markuzon [57] proposed an approach using NN (enhanced fuzzy logic and adaptive resonance theory (ART) neural networks) and achieved an accuracy of 81%. A summary of the proposed approaches that were implemented based on the PIDD is given in Table 7. The advances in the classification techniques led to enhance the accuracy of PIDD classification from 73.8% [36] to 84.24% Ganji and Abadeh [58], in cross-validation and up to 100% with train-test splitting Tigga and Garg [59]. Various algorithms have been utilized for the classification, such as ANN [25,56,57,60-62], LogReg [63-65], and SVM [66-68]. A reviewed summary of the literature on PIDD classification is given in Table 7. The preprocessing steps, classification algorithms, training and testing percentages, and results are detailed in the subsequent columns of Table 7. The last column lists the baseline methods used to compare each of these approaches.

Table 7 *Diabetes prediction approaches using PIDD*

Ref.	Preprocess	Algorithm	Train	Test	Results	Comparison
Smith et al. [25]	None	NN: ADAP (adaptive learning NN algorithm).	75%	25%	76% sensitivity and specificity	The original paper utilized the PIDD.
Shanker [56]	None	NN with various hidden nodes, activation functions, and input feature set	75%	25%	81% Accuracy	ADAP and LogReg
Carpenter and Markuzon [57]	None	ARTMAP-IC (enhanced fuzzy logic)	75%	25%	81% Accuracy	ARTMAP.
Au and Chan [69]	None	A fuzzy inference process learned from the training data and validated by experts.	70%	30%	77.6% Accuracy	C4.5, CBA (rule-based), and FID (fuzzy tree).
Breault [36]	Samples and feature selection, and discretization	RS	300	92	73.8% Accuracy	The results of previous studies

Zhang et al. [70]	None	Attribute weighted KNN	100	668	75.6% Accuracy	KNN
Kayaer and Yildirim [60]	None	NN (general regression NN (GRNN))	75%	25%	80.2% Accuracy	MLP and general regression NN (GRNN)
Pobi and Hall [71]	None	RF with a different version of PIDD of 42 features.	10-Fold Cross-validation		80.3% Accuracy	ANN and C4.5 and ensembles.
Ghosh and Hasley [63]	None	A modified version of the logistic regression with different cost function	65%	35%	83.5% Accuracy	LogReg
Tinos and Junior [61]	None	ANN, k-means, and GA.	75%	25%	80.2% Accuracy	Other activation functions.
Polat et al. [72]	None	GDA-LSVM	10-Fold Cross-validation		82.05% Accuracy	LS-SVM
Patil et al. [73]	Samples selection (625)	Hybrid Prediction Model (HPM) of K-means clustering, and C4.5.	10-Fold Cross-validation of 625 samples		92.38% Accuracy	22 algorithms, as reported by Michie et al. [74]
Dogantekin et al. [75]	None	Linear Discriminant Analysis (LDA) and Adaptive Network Based Fuzzy Inference System (ANFIS)	90%	10%	84.61% Accuracy	The results of previous studies
Karegowda et al. [76]	Samples selection (392) and feature selection	Hybrid-system of ANN and GA	60%	40%	84.7% Accuracy	DT
Calisir and Dogantekin [77]	None	LDA and Morlet Wavelet SVM (MWSVM)	90%	10%	89.74% Accuracy	The results of previous studies
Al Jarullah [78]	Samples selection (724), feature selection, missing values, and discretization	C.4.5 Decision Tree	10-Fold Cross-validation of 724 samples		76.68% Accuracy	None
Ganji and Abadeh [58]	None	Ant Colony-based classification creates a Set of fuzzy rules	10-Fold Cross-validation		84.24% Accuracy	The results of previous studies
Karatsiolis and Schizas [79]	None	Hybrid-system of SVM and Radial Basis Function	Using training set		82.2% Accuracy	Polynomial SVM
Almasi et al. [66]	None	Least square SVM (LS-SVM)	10-Fold Cross-validation		79.66% Accuracy	The results of previous studies
Karthikeyani and Begum [62]	Samples selection	C4.5, SVM, KNN, ANN, BLR, MLR, CRT, CS-CRT, PLS-DA and PLS-LDA	623	145	74% Accuracy	The results of previous studies

J. of Soft Computing and Data Mining Vol. 6 No. 1 (2025) p. 1-20

Mandal et al. [80]	None	C4.5, Rough Set, Bayesian Network, Neural Network	10-Fold Cross-validation	75.78% Accuracy	Various other classification algorithms.
			80% 20%	83.33% Accuracy	
Vijayan and Ravikumar [81]	None	Amalgam KNN and Adaptive Neuro-Fuzzy Inference System (ANFIS)	10-Fold Cross-validation	80% Accuracy	EM, KNN, K-means, amalgam KNN and ANFIS
Zangooei et al. [82]	None	Propose a new Algorithm by combining SVR and the Non-dominated Sorting GA (NSGA-II)	10-Fold Cross-validation	86.13% Accuracy	The results of previous studies
Nadimi-Shaharaki and Ghahramani [83]	Feature selection	GA then C4.5	10-Fold Cross-validation	75.13% Accuracy	None
Farahmandian et al. [68]	None	SVM	80% 20%	81.77% Accuracy	KNN, ID3, NB, CART, C4.5 and C5.0
Mercaldo et al. [65]	None	LogReg	80% 20%	0.75 Precision and 0.75 recall	KNN, NB, DT, RF, and SVM.
Islam and Jahan [64]	Samples selection (755)	LogReg	10-Fold Cross-validation of 755 samples	78.01% Accuracy	KNN, OneR, SVM, ANN, NB, and Bagging
Nnamoko et al. [84]	None	Proposed ensemble method of five base classifiers, SMO, RBF, C4.5, NB, and RIPPER	10-Fold Cross-validation	83% Accuracy	NB, RBF, SMO, and C4.5
Mahmud et al. [85]	None	NB	70% 30%	74% Accuracy	ANN, SVM, LogReg, DT, and RF.
Wei et al. [86]	None	DL	10-Fold Cross-validation	77.86% Accuracy	LogReg, SVM, DT, and NB
Choudhury and Gupta [87]	None	LogReg	250 518	77.61% Accuracy	ANN, DT, RF, NB, KNN and SVM
Guldogan et al. [88]	None	ANN	60% 40%	78.1% Accuracy	RBF
Abedini et al. [89]	None	Hybrid approach of DT and LogReg in layer one and ANN in layer two	70% 30%	83.08% Accuracy	Compare the results with DT and LogReg
Kumar et al. [90]	None	Deep learning: Multi-Layer Feed Forward NN (MLFNN)	90% 10%	81.73% Accuracy	NB and RF

Tigga and Garg [59]	None	RF	75%	25%	100% Accuracy	LogReg, KNN, SVM, NB, and DT
			10-Fold Cross-validation		77.4% Accuracy	
Naz and Ahuja [91]	Samples selection	Deep learning: Multi-Layer Feed Forward NN (MLFNN)	NA	207	98.07% Accuracy	DT, ANN, and NB
Abdulhadi and Al-Mousa [92]	None	Median-based filling missing data, then RF	Unknown		82% Accuracy	LDA, LogReg, and voting
Kumari et al. [93]	Samples selection	Median-based filling missing data, followed by Ensemble soft voting.	561	207	79.04% Accuracy	Various classifiers, such as SVM and RF.
Aamir et al. [94]	Samples selection	Fuzzy logic system	400	368	96.47% Accuracy	The results of previous studies
Edeh et al. [95]	Filling missing values	SVM	80%	20%	83.1% Accuracy	RF, NB, and DT
Chang et al. [96]	Samples selection	PCA then RF	538	230	79.13% Accuracy	NB and C4.5
Perdana et al. [97]	None	KNN (K=22)	90%	10%	83.12% Accuracy	None
Reza et al. [98]	Samples selection, Missing values, balancing, and scaling	SVM	10-Fold Cross-validation		85.5% Accuracy	The results of previous studies
Khan et al. [99]	Feature selection	ANN	70%	30%	99.36% Accuracy	RF, GBM, SVM and TabNet
Reza et al. [100]	Samples selection, Missing values, balancing, and scaling	Stacking NN	5-Fold Cross-validation		77.1% Accuracy	RF, DT, SVM, and Other Stacking

As noted, there is inconsistency in the evaluation model, in which some approaches use cross-validation while others use a percentage split with different splits. Filtering or sample selection has also been implemented in some approaches, leading to a different number of evaluation samples. Besides, some approaches used preprocessing, while others did not. Accordingly, there is a need to unify such issues to compare these approaches fairly.

5. Comparative Study

In this comparative study, a series of scenarios are implemented based on commonly utilized stages in prediction techniques. The frameworks and stages employed are discussed, and the results are presented in this section.

5.1 Implementation

The implementation consists of a transformation step, which is included for all the experiments to allow testing the dataset using all the classifiers, as some of the classifiers cannot process non-numerical data. The preprocessing is implemented first (scaling and filling missing values), followed by feature selection (Chi-Square). Chi-Square feature selection was chosen for this study due to its ability to evaluate the independence between categorical features and the target variable. This statistical method assesses how well each feature contributes to the prediction of the target class by measuring the discrepancy between the observed and expected frequencies of the feature values. Given that the dataset contains both categorical and continuous variables, Chi-Square is particularly effective as it can identify relevant features that have a significant association with the outcome. Additionally, its computational efficiency makes it suitable for handling larger datasets, allowing for quicker

J. of Soft Computing and Data Mining Vol. 6 No. 1 (2025) p. 1-20

feature selection without sacrificing predictive power. This approach helps to enhance model performance by eliminating irrelevant features and reducing the risk of overfitting, ultimately leading to more accurate predictions. Then, oversampling (SMOTE) is implemented with the aim of improving the accuracy of the classification. Accordingly, the framework for classification is formulated for these stages, as given in Fig. 2.

5.2 Results

To evaluate the effect of each processing step, these steps were incrementally augmented while the results were assessed after each addition. The results of using various classifiers without preprocessing, oversampling, or feature selection are compared with those obtained using scaled features (a preprocessing step) alone, as given in Table 8. The scaling process, utilizing the Min-Max scaler, slightly improved the performance of most of the classifiers but did not change the highest accuracy, which remained at 77.2% with the LogReg classifier. In contrast, LogReg and RF achieved an accuracy of 76.4% with scaling. LogReg also attained the best precision of 72.2% for both scaled and unscaled features, where precision consistently exceeded recall across all classifiers. RF achieved the highest recall of 59.3% with unscaled features, while the best recall significantly improved to 69% with scaled features using the NC classifier. The best F-measure for unscaled features was 64%, also achieved by RF, while NC produced a slightly higher F-measure of 64.5% for scaled features. Overall, there is variability in the results for scaled and unscaled features, leading to an ambiguous influence on classification outputs. However, RF demonstrated stable results across both feature sets, achieving the best accuracy for scaled features and the best recall and F-measure for unscaled features.

Table 8 *Evaluation of the classification results with scaling step*

#	Classifier	No Processing				Scaled Features			
		Acc.	P	R	F	Acc.	P	R	F
1	KNN	0.703	0.579	0.549	0.563	**0.740**	0.643	0.571	0.605
2	NB	**0.757**	0.671	0.593	0.630	**0.757**	0.671	0.593	0.630
3	NC	0.633	0.472	0.437	0.453	**0.734**	0.605	0.690	0.645
4	SVM	**0.766**	0.706	0.563	0.627	0.651	0.623	0.129	0.232
5	DT	0.737	0.651	0.530	0.584	**0.745**	0.667	0.537	0.595
6	RF	**0.767**	0.694	0.593	0.640	0.764	0.688	0.593	0.637
7	GBM	0.729	0.624	0.563	0.592	**0.730**	0.624	0.571	0.596
8	AdaBoost	**0.753**	0.668	0.578	0.620	**0.753**	0.668	0.578	0.620
9	**LogReg**	**0.772**	0.722	0.563	0.633	0.764	0.738	0.504	0.599
10	DL	0.656	0.524	0.160	0.246	**0.671**	0.642	0.127	0.212

Addressing missing values is a critical step, as these values can significantly impact the performance of machine learning models. Missing data can lead to biased estimates and reduced model accuracy if not handled properly. In this step, the median value was utilized for the imputation to fill missing values, as it is a robust method that can effectively mitigate the influence of outliers and preserve the integrity of the dataset. The results after augmenting the dataset with filled missing values are presented in Table 9. Notably, filling in the missing values led to improvements in most classifiers, with the best accuracy significantly increasing to 88.2% using the GBM, compared to 77.2% without this preprocessing step. This substantial jump in accuracy highlights the critical role that handling missing data plays in enhancing model performance, as the filled values provide more context and information for the classifiers to learn from, ultimately leading to better generalization on unseen data. It is important to highlight that the influence of scaling remains ambiguous at this stage. Nevertheless, RF has consistently demonstrated stable performance across both scaled and unscaled features, achieving the best results in both scenarios alongside GBM.

Table 9 *Evaluation of the classification results with filling missing values step*

#	Classifier	Unscaled with Filling Missing Values				Scaled with Filling Missing Values			
		Acc.	**P**	**R**	**F**	**Acc.**	**P**	**R**	**F**
1	KNN	**0.849**	0.784	0.784	0.784	0.762	0.659	0.657	0.658
2	NB	**0.768**	0.680	0.634	0.656	**0.768**	0.680	0.634	0.656
3	NC	**0.839**	0.747	0.813	0.779	0.757	0.628	0.743	0.680
4	SVM	**0.775**	0.707	0.604	0.652	0.651	0.000	0.000	0.000
5	DT	**0.868**	0.841	0.769	0.803	0.867	0.837	0.769	0.802
6	RF	**0.879**	0.840	0.806	0.823	**0.879**	0.830	0.821	0.826
7	**GBM**	**0.882**	0.847	0.806	0.826	**0.882**	0.847	0.806	0.826
8	AdaBoost	0.870	0.823	0.799	0.811	0.870	0.823	0.799	0.811
9	LogReg	0.777	0.716	0.601	0.653	0.777	0.734	0.567	0.640
10	DL	**0.783**	0.650	0.817	0.724	0.651	0.000	0.000	0.000

The results after augmenting the dataset with oversampling using SMOTE are provided in Table 10. As such, up to this stage, scaling, filling missing values, and oversampling are implemented prior to the classification step. Oversampling consistently improves classifier performance across both scaled and unscaled features. The highest accuracy, 90.5%, was achieved using both the RF and GBM with scaled features. Notably, all classifiers except NB demonstrated better performance in terms of accuracy, precision, recall, and f-measure after oversampling was implemented. The results indicate that both scaled and unscaled features benefit from oversampling. However, performance varies slightly: some classifiers performed better with scaled features, while others yielded better results with unscaled features. Overall, the combination of oversampling and filling in missing values significantly impacted classifier performance for the diabetes detection task on the PIDD dataset. Specifically, the best accuracy, reported as 90.5%, improved from 88.2% achieved before applying oversampling. Precision, recall, and f-measure also showed notable gains after feature selection was applied.

Table 10 *Evaluation of the classification results with oversampling*

#	Classifier	Unscaled, Filled, and Oversampling				Scaled, Filled, and Oversampling			
		Acc.	**P**	**R**	**F**	**Acc.**	**P**	**R**	**F**
1	KNN	**0.877**	0.852	0.912	0.881	0.819	0.774	0.902	0.833
2	NB	0.767	0.795	0.720	0.756	**0.774**	0.790	0.746	0.767
3	NC	**0.843**	0.854	0.828	0.841	0.763	0.770	0.750	0.760
4	SVM	**0.849**	0.827	0.882	0.854	0.746	0.772	0.698	0.733
5	DT	0.878	0.861	0.902	0.881	**0.879**	0.853	0.916	0.883
6	RF	0.899	0.880	0.924	0.901	**0.905**	0.887	0.928	0.907
7	**GBM**	0.900	0.895	0.906	0.901	**0.905**	0.898	0.914	0.906
8	AdaBoost	0.893	0.888	0.900	0.894	**0.893**	0.891	0.896	0.893
9	LogReg	**0.791**	0.794	0.786	0.790	0.784	0.797	0.762	0.779
10	DL	**0.808**	0.777	0.864	0.818	0.797	0.738	0.922	0.820

The results following feature selection based on the Chi-Square test are presented in Table 11. Interestingly, the same Chi-Square method selected different feature subsets for scaled and unscaled data: for unscaled data, seven out of eight features were retained, excluding the DPF feature, while for scaled data, the Diastolic feature was excluded instead (see Table 2). The impact of feature selection on classifier performance varied across models. Specifically, the NB and AdaBoost classifiers showed consistent accuracy with unscaled data and improved accuracy with scaled data after feature selection. In contrast, classifiers like NC, SVM, and LogReg demonstrated lower accuracy for both scaled and unscaled data post-feature selection. Other classifiers, KNN, DT, RF, GBM, and DL, achieved better accuracy after feature selection was implemented.

Notably, the highest accuracy reached 91.5% after feature selection, up from 90.5% before feature selection was applied. Precision, recall, and f-measure scores were similarly enhanced by incorporating feature selection. These results, as shown in Table 7, are competitive with or surpass prior literature, and all evaluations were conducted with cross-validation on the complete dataset to ensure robust performance assessment.

J. of Soft Computing and Data Mining Vol. 6 No. 1 (2025) p. 1-20

Table 11 *Evaluation of the classification results with feature selection*

#	Classifier	Unscaled, Filled, Oversampling, and Feature Selection				Scaled, Filled, Oversampling, and Feature Selection			
		Acc.	**P**	**R**	**F**	**Acc.**	**P**	**R**	**F**
1	KNN	**0.887**	0.860	0.924	0.891	0.833	0.785	0.918	0.846
2	NB	0.767	0.791	0.726	0.757	0.787	0.806	0.756	0.780
3	NC	0.837	0.849	0.820	0.834	0.760	0.769	0.744	0.756
4	SVM	0.826	0.806	0.858	0.831	0.735	0.769	0.672	0.717
5	DT	0.880	0.868	0.896	0.882	0.874	0.857	0.898	0.877
6	**RF**	0.915	0.906	0.926	0.916	0.910	0.893	0.932	0.912
7	GBM	0.910	0.902	0.920	0.911	0.910	0.899	0.924	0.911
8	AdaBoost	0.881	0.885	0.876	0.880	0.893	0.891	0.896	0.893
9	LogReg	0.788	0.789	0.786	0.788	0.781	0.791	0.764	0.777
10	DL	0.838	0.825	0.858	0.841	0.875	0.860	0.896	0.878

Overall, RF demonstrated the best performance, significantly improving at each stage of preprocessing: starting from an initial accuracy of 76.7% with feature scaling, increasing to 87.9% after filling missing values, then 89.9% with oversampling, and reaching 91.5% after feature selection. Similarly, GBM showed substantial improvements across the process, with accuracy progressing from 72.9% initially, to 88.2% after filling missing values, 90% with oversampling, and ultimately 91% following feature selection. These results underscore the impact of each preprocessing stage on enhancing classifier performance.

6. Conclusion

This paper presented a survey of existing approaches for diabetes prediction using the PIDD. The survey investigated the inconsistencies in evaluation methods, with some approaches using cross-validation while others used various percentage splits, and some implemented sample selection or filtering, resulting in differing sample sizes. In this study's evaluation, cross-validation with all data samples was used for a consistent comparison across models. The implemented framework, including scaling, filling missing values, oversampling and feature selection, and oversampling, provided critical insights into how these steps influence classifier performance for diabetes detection in the PIDD. Initially, the application of scaled features alone, using the Min-Max scaler, led to only marginal performance improvements in most classifiers without substantially affecting the highest accuracy. The LogReg classifier achieved the best accuracy of 77.2% with unscaled features, while the best accuracy with scaled features was 76.4%, achieved by both LogReg and RF. Filling missing values based on the median significantly improved model performance, with GBM, in particular, reaching an accuracy of 88.2%. The improvement was likely due to the median's ability to address outliers effectively, which enhances stability and prediction accuracy in complex models like GBM, where imbalances in data can skew residual estimates during tree building. This suggests that handling missing values is particularly impactful for tree-based models.

Oversampling using SMOTE had a strong positive effect, particularly for RF and GBM, which achieved the highest accuracy of 90.5% after oversampling. This improvement is due to the classifiers' capacity to leverage diverse, balanced samples, enhancing its ensemble accuracy through reduced bias and variance across trees. In contrast, NB, LogReg and DL did not show the same level of improvement, likely because NB assumes feature independence, which oversampling does not directly enhance, and DL requires more complex adjustments, such as architectural changes, to significantly improve performance. The Chi-Square feature selection method showed varying impacts depending on the classifier. KNN, DT, RF, GBM, and DL achieved better accuracy with feature selection on scaled data, while NC, SVM, and LogReg showed slight decreases. This variability suggests that certain models, such as KNN, benefit from a refined feature space, which enhances their performance in high-dimensional data by reducing noise. Conversely, NC's resilience to feature selection is likely due to their innate handling of feature importance, which can reduce sensitivity to less relevant features. Future work will explore various weighting techniques and additional preprocessing, feature selection, and oversampling methods to refine diabetes prediction in PIDD further.

Acknowledgements

This work was supported by Al-Ahliyya Amman University–Jordan research grant.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of the paper.

Author Contribution

Conceptualization: *Ahmad Adel Abu-Shareha;* ***methodology:*** *Mohammad A. Alsharaiah;* ***software:*** *Mosleh M. Abualhaj;* ***validation:*** *Adeeb Al-Saaidah****;*** ***formal analysis:*** *Anusha Achuthan;* ***resources:*** *Ahmad Adel AbuShareha;* ***writing—original draft preparation:*** *Ahmad Adel Abu-Shareha;* ***writing—review and editing:*** *Anusha Achuthan;* ***supervision:*** *Ahmad Adel Abu-Shareha;* ***funding acquisition:*** *Ahmad Adel Abu-Shareha".*

References

[1] Hamoud, A., Hashim, A. S., & Awadh, W. A. (2018). Clinical data warehouse: a review. *Iraqi Journal for Computers and Informatics*, *44*(2).

[2] Riegler, M., Lux, M., Griwodz, C., Spampinato, C., de Lange, T., Eskeland, S. L., . . . Gurrin, C. (2016). Multimedia and medicine: Teammates for better disease detection and survival. Proceedings of the 24th ACM international conference on Multimedia,

[3] Yang, C.-T., Liu, J.-C., Chen, S.-T., & Lu, H.-W. (2017). Implementation of a big data accessing and processing platform for medical records in cloud. *Journal of medical systems*, *41*(10), 1-28.

[4] Canelón, S. P., Burris, H. H., Levine, L. D., & Boland, M. R. (2021). Development and evaluation of MADDIE: Method to Acquire Delivery Date Information from Electronic health records. *International journal of medical informatics*, *145*, 104339.

[5] Garets, D., & Davis, M. (2006). Electronic medical records vs. electronic health records: yes, there is a difference. *Policy white paper. Chicago, HIMSS Analytics*, 1-14.

[6] Shortliffe, E. H., Buchanan, B. G., & Feigenbaum, E. A. (1979). Knowledge engineering for medical decision making: A review of computer-based clinical decision aids. *Proceedings of the IEEE*, *67*(9), 1207-1224.

[7] Jia, Z., Zeng, X., Duan, H., Lu, X., & Li, H. (2020). A patient-similarity-based model for diagnostic prediction. *International journal of medical informatics*, *135*, 104073.

[8] Alonso, S. G., de la Torre-Díez, I., Hamrioui, S., López-Coronado, M., Barreno, D. C., Nozaleda, L. M., & Franco, M. (2018). Data mining algorithms and techniques in mental health: A systematic review. *Journal of medical systems*, *42*(9), 1-15.

[9] Milovic, B., & Milovic, M. (2012). Prediction and decision making in health care using data mining. *International Journal of Public Health Science (IJPHS)*, *1*(2), 69-78.

[10] Islam, M. S., Hasan, M. M., Wang, X., & Germack, H. D. (2018). A systematic review on healthcare analytics: application and theoretical perspective of data mining. Healthcare,

[11] Tandan, M., Acharya, Y., Pokharel, S., & Timilsina, M. (2021). Discovering symptom patterns of COVID-19 patients using association rule mining. *Computers in biology and medicine*, *131*, 104249.

[12] Mohapatra, A., Khare, S., & Gupta, D. (2021). Analysis of Tuberculosis Disease Using Association Rule Mining. In *Advances in Artificial Intelligence and Data Engineering* (pp. 995-1008). Springer.

[13] Al-Yarimi, F. A. M., Munassar, N. M. A., Bamashmos, M. H. M., & Ali, M. Y. S. (2021). Feature optimization by discrete weights for heart disease prediction using supervised learning. *Soft Computing*, *25*(3), 1821-1831.

[14] Husein, I., Noerjoedianto, D., Sakti, M., & Jabbar, A. H. (2020). Modeling of Epidemic Transmission and Predicting the Spread of Infectious Disease. *Systematic Reviews in Pharmacy*, *11*(6).

[15] Nasarian, E., Abdar, M., Fahami, M. A., Alizadehsani, R., Hussain, S., Basiri, M. E., . . . Acharya, U. R. (2020). Association between work-related features and coronary artery disease: A heterogeneous hybrid feature selection integrated with balancing approach. *Pattern Recognition Letters*, *133*, 33-40.

[16] Magesh, G., & Swarnalatha, P. (2020). Optimal feature selection through a cluster-based DT learning (CDTL) in heart disease prediction. *Evolutionary Intelligence*, 1-11.

[17] Choubey, D. K., Kumar, P., Tripathi, S., & Kumar, S. (2020). Performance evaluation of classification methods with PCA and PSO for diabetes. *Network Modeling Analysis in Health Informatics and Bioinformatics*, *9*(1), 1-30.

[18] Okur, M. E., Karantas, I. D., & Siafaka, P. I. (2017). Diabetes Mellitus: a review on pathophysiology, current status of oral pathophysiology, current status of oral medications and future perspectives. *ACTA Pharmaceutica Sciencia*, *55*(1).

[19] Association, A. D. (2021). Diabetes Care in the Hospital: Standards of Medical Care in Diabetes. *Diabetes Care*, *44*(Supplement 1), S211-S220.

[20] Roglic, G. (2016). WHO Global report on diabetes: A summary. *International Journal of Noncommunicable Diseases*, *1*(1), 3.

[21] Saeedi, P., Petersohn, I., Salpea, P., Malanda, B., Karuranga, S., & Unwin, N. (2019). Global and regional diabetes prevalence estimates for 2019 and projections for 2030 and 2045: Results from the International Diabetes Federation Diabetes Atlas. . *Diabetes research and clinical practice*, *157*(107843).

[22] Young-Hyman, D., De Groot, M., Hill-Briggs, F., Gonzalez, J. S., Hood, K., & Peyrot, M. (2016). Psychosocial care for people with diabetes: a position statement of the American Diabetes Association. *Diabetes Care*, *39*(12), 2126-2140.

[23] Kaur, S., Singla, J., Nkenyereye, L. J., S., Prashar, D., Joshi, G. P., & Islam, S. R. (2020). Medical Diagnostic Systems Using Artificial Intelligence (AI) Algorithms: Principles and Perspectives. *IEEE Access*.

[24] Alfian, G., Syafrudin, M., Ijaz, M. F., Syaekhoni, M. A., Fitriyani, N. L., & Rhee, J. (2018). A personalized healthcare monitoring system for diabetic patients by utilizing BLE-based sensors and real-time data processing. *Sensors*, *18*(7), 2183.

[25] Smith, J. W., Everhart, J., Dickson, W., Knowler, W., & Johannes, R. (1988). Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. Proceedings of the annual symposium on computer application in medical care,

[26] Larabi-Marie-Sainte, S., Aburahmah, L., Almohaini, R., & Saba, T. (2019). Current techniques for diabetes prediction: review and case study. *Applied Sciences*, *9*(21), 4604.

[27] Khan, F. A., Zeb, K., Al-Rakhami, M., Derhab, A., & Bukhari, S. A. C. (2021). Detection and prediction of diabetes using data mining: a comprehensive review. *IEEE Access*, *9*, 43711-43735.

[28] Jaiswal, V., Negi, A., & Pal, T. (2021). A review on current advances in machine learning based diabetes prediction. *Primary Care Diabetes*, *15*(3), 435-443.

[29] Chaki, J., Ganesh, S. T., Cidham, S., & Theertan, S. A. (2022). Machine learning and artificial intelligence based Diabetes Mellitus detection and self-management: A systematic review. *Journal of King Saud University-Computer and Information Sciences*, *34*(6), 3204-3225.

[30] Kodama, S., Fujihara, K., Horikawa, C., Kitazawa, M., Iwanaga, M., Kato, K., . . . Shimano, H. (2022). Predictive ability of current machine learning algorithms for type 2 diabetes mellitus: A meta‐analysis. *Journal of diabetes investigation*, *13*(5), 900-908.

[31] Mohsen, F., Al-Absi, H. R., Yousri, N. A., El Hajj, N., & Shah, Z. (2023). A scoping review of artificial intelligence-based methods for diabetes risk prediction. *npj Digital Medicine*, *6*(1), 197.

[32] Pinchevsky, Y., Butkow, N., Raal, F. J., Chirwa, T., & Rothberg, A. (2020). Demographic and clinical factors associated with development of type 2 diabetes: a review of the literature. *International Journal of General Medicine*, 121-129.

[33] Efron, B., Hastie, T., Johnstone, I., & Tibshirani, R. (2004). Least angle regression. *Annals of statistics*, *32*(2), 407-499.

[34] Islam, M. F., Ferdousi, R., Rahman, S., & Bushra, H. Y. (2020). Likelihood prediction of diabetes at early stage using data mining techniques. In *Computer Vision and Machine Intelligence in Medical Image Analysis* (pp. 113-125). Springer.

[35] Knowler, W. C., Bennett, P. H., Hamman, R. F., & Miller, M. (1978). Diabetes incidence and prevalence in Pima Indians: a 19-fold greater incidence than in Rochester, Minnesota. *American journal of epidemiology*, *108*(6), 497-505.

[36] Breault, J. L. (2001). Data mining diabetic databases: Are rough sets a useful addition. *Computing science and statistics*, *34*, 404.

[37] Raschka, S., & Mirjalili, V. (2019). *Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow 2*. Packt publishing ltd.

[38] Harris, E., Marcu, A., Painter, M., Niranjan, M., Prügel-Bennett, A., & Hare, J. (2020). Fmix: Enhancing mixed sample data augmentation. *arXiv preprint arXiv:2002.12047*.

[39] Emmanuel, T., Maupong, T., Mpoeleng, D., Semong, T., Mphago, B., & Tabona, O. (2021). A survey on missing data in machine learning. *Journal of Big data*, *8*, 1-37.

[40] Alyasiri, O. M., Cheah, Y.-N., Abasi, A. K., & Al-Janabi, O. M. (2022). Wrapper and hybrid feature selection methods using metaheuristic algorithms for English text classification: A systematic review. *IEEE Access*, *10*, 39833-39852.

[41] Bommert, A., Sun, X., Bischl, B., Rahnenführer, J., & Lang, M. (2020). Benchmark for filter methods for feature selection in high-dimensional classification data. *Computational Statistics & Data Analysis*, *143*, 106839.

[42] Wang, A., An, N., Chen, G., Li, L., & Alterovitz, G. (2015). Accelerating wrapper-based feature selection with K-nearest-neighbor. *Knowledge-Based Systems*, *83*, 81-91.

[43] Chen, C. W., Tsai, Y. H., Chang, F. R., & Lin, W. C. (2020). Ensemble feature selection in medical datasets: Combining filter, wrapper, and embedded feature selection results. *Expert Systems*, *37*(5), e12553.

[44] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321-357.

[45] Wongvorachan, T., He, S., & Bulut, O. (2023). A comparison of undersampling, oversampling, and SMOTE methods for dealing with imbalanced classification in educational data mining. *Information*, *14*(1), 54.

[46] Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, *13*(1), 21-27.

[47] Chaudhuri, B. (1996). A new definition of neighborhood of a point in multi-dimensional space. *Pattern Recognition Letters*, *17*(1), 11-17.

[48] Rish, I. (2001). An empirical study of the naive Bayes classifier. IJCAI 2001 workshop on empirical methods in artificial intelligence,

[49] Hosmer Jr, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression*. John Wiley & Sons.

[50] Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., & Scholkopf, B. (1998). Support vector machines. *IEEE Intelligent Systems and their applications*, *13*(4), 18-28.

[51] Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*, 81-106.

[52] Breiman, L. (2001). Random forests. *Machine learning*, *45*, 5-32.

[53] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.

[54] Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, *55*(1), 119-139.

[55] Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *67*(2), 301-320.

[56] Shanker, M. S. (1996). Using neural networks to predict the onset of diabetes mellitus. *Journal of chemical information and computer sciences*, *36*(1), 35-41.

[57] Carpenter, G. A., & Markuzon, N. (1998). ARTMAP-IC and medical diagnosis: Instance counting and inconsistent cases. *Neural networks*, *11*(2), 323-336.

[58] Ganji, M. F., & Abadeh, M. S. (2011). A fuzzy classification system based on Ant Colony Optimization for diabetes disease diagnosis. *Expert systems with applications*, *38*(12), 14650-14659.

[59] Tigga, N. P., & Garg, S. (2020). Prediction of type 2 diabetes using machine learning classification methods. *Procedia computer science*, *167*, 706-716.

[60] Kayaer, K., & Yildirim, T. (2003). Medical diagnosis on Pima Indian diabetes using general regression neural networks. Proceedings of the international conference on artificial neural networks and neural information processing (ICANN/ICONIP),

[61] Tinos, R., & Junior, L. O. M. (2008). Selection of radial basis functions via genetic algorithms in pattern recognition problems. 2008 10th Brazilian Symposium on Neural Networks,

[62] Karthikeyani, V., & Begum, I. P. (2013). Comparison a performance of data mining algorithms (CPDMA) in prediction of diabetes disease. *International journal on computer science and engineering*, *5*(3), 205.

[63] Ghosh, B., & Hasley, J. (2007). Using Asymmetric Classification Cost Matrices in Predicting Diabetes. *ICDSS 2007 Proceedings*, 7.

[64] Islam, M. A., & Jahan, N. (2017). Prediction of onset diabetes using machine learning techniques. *International Journal of Computer Applications*, *975*, 8887.

[65] Mercaldo, F., Nardone, V., & Santone, A. (2017). Diabetes mellitus affected patients classification and diagnosis through machine learning techniques. *Procedia computer science*, *112*, 2519-2528.

[66] Almasi, B. N., Almasi, O. N., Kavousi, M., & Sharifinia, A. (2013). Computer-aided diagnosis of diabetes using least square support vector machine. *Journal of Advanced Computer Science & Technology*, *2*(2), 68.

[67] Sanakal, R., & Jayakumari, T. (2014). Prognosis of diabetes using data mining approach-fuzzy C means clustering and support vector machine. *International Journal of Computer Trends and Technology*, *11*(2), 94-98.

[68] Farahmandian, M., Lotfi, Y., & Maleki, I. (2015). Data mining algorithms application in diabetes diseases diagnosis: A case study. *MAGNT Res., Tech. Rep.*, *3*(1), 989-997.

[69] Au, W.-H., & Chan, K. C. (2001). Classification with degree of membership: A fuzzy approach. Proceedings 2001 IEEE International Conference on Data Mining,

[70] Zhang, L., Coenen, F., & Leng, P. (2002). An attribute weight setting method for k-NN based binary classification using quadratic programming. ECAI,

[71] Pobi, S., & Hall, L. O. (2006). Predicting juvenile diabetes from clinical test results. The 2006 IEEE International Joint Conference on Neural Network Proceedings,

[72] Polat, K., Güneş, S., & Arslan, A. (2008). A cascade learning system for classification of diabetes disease: Generalized discriminant analysis and least square support vector machine. *Expert systems with applications*, *34*(1), 482-487.

[73] Patil, B. M., Joshi, R. C., & Toshniwal, D. (2010). Hybrid prediction model for type-2 diabetic patients. *Expert systems with applications*, *37*(12), 8102-8108.

[74] Michie, D., Spiegelhalter, D. J., & Taylor, C. C. (1994). Machine learning, neural and statistical classification.

[75] Dogantekin, E., Dogantekin, A., Avci, D., & Avci, L. (2010). An intelligent diagnosis system for diabetes on linear discriminant analysis and adaptive network based fuzzy inference system: LDA-ANFIS. *Digital Signal Processing*, *20*(4), 1248-1255.

[76] Karegowda, A. G., Manjunath, A., & Jayaram, M. (2011). Application of genetic algorithm optimized neural network connection weights for medical diagnosis of pima Indians diabetes. *International Journal on Soft Computing*, *2*(2), 15-23.

[77] Calisir, D., & Dogantekin, E. (2011). An automatic diabetes diagnosis system based on LDA-Wavelet Support Vector Machine Classifier. *Expert systems with applications*, *38*(7), 8311-8315.

[78] Al Jarullah, A. A. (2011). Decision tree discovery for the diagnosis of type II diabetes. 2011 International conference on innovations in information technology,

[79] Karatsiolis, S., & Schizas, C. N. (2012). Region based Support Vector Machine algorithm for medical diagnosis on Pima Indian Diabetes dataset. 2012 IEEE 12th International Conference on Bioinformatics & Bioengineering (BIBE),

[80] Mandal, S., Saha, G., & Pal, R. K. (2014). A Comparative Study on Disease Classification using Different Soft Computing Techniques. *The SIJ Transactions on Computer Science Engineering & its Applications (CSEA)*, *2*(3), 45-52.

[81] Vijayan, V., & Ravikumar, A. (2014). Study of data mining algorithms for prediction and diagnosis of diabetes mellitus. *International Journal of Computer Applications*, *95*(17).

[82] Zangooei, M. H., Habibi, J., & Alizadehsani, R. (2014). Disease Diagnosis with a hybrid method SVR using NSGA-II. *Neurocomputing*, *136*, 14-29.

[83] Nadimi-Shaharaki, M. H., & Ghahramani, M. (2015). Efficient data preparation techniques for diabetes detection. IEEE EUROCON 2015-International Conference on Computer as a Tool (EUROCON),

[84] Nnamoko, N., Hussain, A., & England, D. (2018). Predicting Diabetes Onset: An Ensemble Supervised Learning Approach. 2018 IEEE Congress on Evolutionary Computation (CEC),

[85] Mahmud, S. H., Hossin, M. A., Ahmed, M. R., Noori, S. R. H., & Sarkar, M. N. I. (2018). Machine learning based unified framework for diabetes prediction. Proceedings of the 2018 International Conference on Big Data Engineering and Technology,

[86] Wei, S., Zhao, X., & Miao, C. (2018). A comprehensive exploration to the machine learning techniques for diabetes identification. 2018 IEEE 4th World Forum on Internet of Things (WF-IoT),

[87] Choudhury, A., & Gupta, D. (2019). A survey on medical diagnosis of diabetes using machine learning techniques. In *Recent developments in machine learning and data analytics* (pp. 67-78). Springer.

[88] Guldogan, E., Zeynep, T., Ayca, A., & Colak, C. (2020). Performance Evaluation of Different Artificial Neural Network Models in the Classification of Type 2 Diabetes Mellitus. *The Journal of Cognitive Systems*, *5*(1), 23-32.

[89] Abedini, M., Bijari, A., & Banirostam, T. (2020). Classification of Pima Indian Diabetes Dataset using Ensemble of Decision Tree, Logistic Regression and Neural Network. *The International Journal of Advanced Research in Computer and Communication Engineering*, *9*(7), 1-4.

[90] Kumar, S., Bhusan, B., Singh, D., & kumar Choubey, D. (2020). Classification of diabetes using deep learning. 2020 International Conference on Communication and Signal Processing (ICCSP),

[91] Naz, H., & Ahuja, S. (2020). Deep learning approach for diabetes prediction using PIMA Indian dataset. *Journal of Diabetes & Metabolic Disorders*, *19*, 391-403.

[92] Abdulhadi, N., & Al-Mousa, A. (2021, 14-15, July). *Diabetes detection using machine learning classification methods* international conference on information technology (ICIT), Amman, Jordan.

[93] Kumari, S., Kumar, D., & Mittal, M. (2021). An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier. *International Journal of Cognitive Computing in Engineering*, *2*, 40-46.

[94] Aamir, K. M., Sarfraz, L., Ramzan, M., Bilal, M., Shafi, J., & Attique, M. (2021). A fuzzy rule-based system for classification of diabetes. *Sensors*, *21*(23), 8095.

[95] Edeh, M. O., Khalaf, O. I., Tavera, C. A., Tayeb, S., Ghouali, S., Abdulsahib, G. M., . . . Louni, A. (2022). A classification algorithm-based hybrid diabetes prediction model. *Frontiers in Public Health*, *10*, 829519.

[96] Chang, V., Bailey, J., Xu, Q. A., & Sun, Z. (2023). Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms. *Neural Computing and Applications*, *35*(22), 16157-16173.

[97] Perdana, A., Hermawan, A., & Avianto, D. (2023). Analyze Important Features of PIMA Indian Database For Diabetes Prediction Using KNN. *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, *12*(1), 70-75.

[98] Reza, M. S., Hafsha, U., Amin, R., Yasmin, R., & Ruhi, S. (2023). Improving SVM performance for type II diabetes prediction with an improved non-linear kernel: Insights from the PIMA dataset. *Computer Methods and Programs in Biomedicine Update*, *4*, 100118.

[99] Khan, Q. W., Iqbal, K., Ahmad, R., Rizwan, A., Khan, A. N., & Kim, D. (2024). An intelligent diabetes classification and perception framework based on ensemble and deep learning method. *PeerJ Computer Science*, *10*, e1914.

[100] Reza, M. S., Amin, R., Yasmin, R., Kulsum, W., & Ruhi, S. (2024). Improving diabetes disease patients classification using stacking ensemble method with PIMA and local healthcare data. *Heliyon*, *10*(2).