engineering proceedings

MDPI

Proceeding Paper

An Artificial Intelligence Framework for Early Prediction of Diabetes Using Hybrid Data Analytics †

Chippada Dhanush, Paravasthu Pattarabhiran Venkat, Khayam Khani Fazulu Rahaman and Balasubramaniam Doraiswamy *

Department of Biomedical Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai 600062, Tamil Nadu, India; vtu27201@veltech.edu.in (C.D.); vtu27244@veltech.edu.in (P.P.V.); vtu25027@veltech.edu.in (K.K.F.R.)

* Correspondence: drbalasubramaniamd@veltech.edu.in

† Presented at the 13th International Electronic Conference on Sensors and Applications, 18–20 November 2026; Available online: https://sciforum.net/event/ECSA-13.

Abstract

Identifying the possibility of diabetes at an early stage can help individuals seek appropriate medical evaluation before complications become severe. This work investigates a machine learning-based framework that combines several data preparation techniques with multiple classification algorithms for diabetes prediction. The Pima Indians Diabetes Dataset is considered for the experimental study. Before model development, zero entries in selected physiological variables are treated as invalid observations and replaced using feature-wise median values. The processed data is examined for distributional characteristics, transformed using the Yeo–Johnson method, and standardized using StandardScaler. Since the two outcome classes are not equally represented, SMOTEENN is incorporated to balance and clean the training data. Logistic Regression, Decision Tree, Support Vector Machine, and Random Forest are then trained and compared. GridSearchCV is used to select the Random Forest configuration. The resulting accuracies are 91.04%, 94.02%, 92.53%, and 95.52% for Logistic Regression, Decision Tree, SVM, and Random Forest, respectively. Random Forest gives the best result among the four tested approaches. Its feature analysis places glucose at the top of the importance ranking, followed by age, insulin, BMI, skin thickness, pregnancies, diabetes pedigree function, and blood pressure. The study shows how combining data preparation and ensemble learning can produce a useful computational screening framework, although the resulting model is not intended to replace professional medical diagnosis.

Keywords: diabetes prediction; machine learning; artificial intelligence; clinical data analysis; Random Forest; SMOTEENN; predictive analytics

Check for updates

Academic Editor: Stefano Mariani

Published: 12 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Diabetes requires continuous attention because changes in blood glucose levels can gradually affect several organs and body systems. Detecting people who may be at risk before serious complications develop is therefore an important part of preventive healthcare [1,2]. In practice, assessment is based on a combination of patient information and clinical measurements rather than a single characteristic.

The increasing availability of healthcare datasets has created an opportunity to use computational methods for this type of assessment. Machine learning algorithms can learn relationships between several input variables and a known outcome and can subsequently

use those relationships to classify new observations [3,4]. However, the quality of such predictions depends not only on the classifier but also on how the input data is prepared.

The Pima Indians Diabetes Dataset is frequently used for experimental diabetes classification. The dataset contains measurements including pregnancy count, glucose level, blood pressure, skin thickness, insulin, body mass index, diabetes pedigree function, and age, along with a binary outcome. The original work by Smith et al. demonstrated the use of computational learning for forecasting diabetes onset using this dataset [1].

A direct application of machine learning to raw clinical data may not always provide reliable results. Some physiological attributes in the dataset contain zero values that are not meaningful as actual measurements. In addition, numerical variables can have highly uneven distributions, while the two outcome categories are not represented equally. These characteristics motivated the preprocessing strategy adopted in this study.

The proposed approach combines invalid-value treatment, distributional transformation, scaling, class balancing, and classification. Four algorithms are evaluated under the same general workflow so that their predictive performance can be compared.

The main aim of the study is not to develop a clinical diagnostic device, but to examine whether a carefully prepared machine learning pipeline can provide useful diabetes classification from routinely available numerical attributes.

2. Related Work

Research into computational diabetes prediction has used statistical learning, decision-tree methods, support vector approaches, and ensemble classifiers [3,4]. The Pima dataset has remained useful in this area because it provides a fixed collection of clinical variables with a binary outcome, allowing different algorithms and preprocessing strategies to be compared.

The work of Smith et al. used the ADAP learning algorithm to investigate diabetes onset and established an early example of computational prediction using the Pima dataset [1]. Subsequent machine learning studies have continued to use similar clinical variables as inputs to predictive models.

Preprocessing is particularly important when medical measurements contain unusual distributions. Yeo and Johnson proposed a power transformation that can improve symmetry and normality while allowing the input variable to contain values on either side of zero [5]. This property makes the transformation convenient for general numerical preprocessing.

Another issue is the unequal distribution of classes. If a classifier receives substantially more examples from one category, its learned decision boundary can be influenced by the majority group. Chawla et al. introduced SMOTE to create synthetic minority-class observations rather than simply duplicating existing samples [6]. SMOTEENN extends this idea by combining synthetic oversampling with Edited Nearest Neighbours cleaning [7,8].

Tree-based ensemble methods have also shown strong potential for structured data. Breiman introduced Random Forest as an ensemble of decision trees whose combined predictions can provide robust classification performance [9]. In addition to prediction, tree ensembles can provide estimates of feature importance [10].

Based on these observations, this study combines transformation, scaling, class balancing, and model comparison instead of depending on a single preprocessing operation or classifier.

3. Dataset and Preprocessing

3.1. Dataset

The experimental dataset contains 768 observations. Eight variables are used as predictors, while the Outcome field is used as the binary target. Table 1 summarizes the input variables used in the study.

https://doi.org/10.3390/engproc2026163106

Table 1. Input variables used in the study.

Variable	Description
Pregnancies	Number of recorded pregnancies
Glucose	Plasma glucose concentration
Blood Pressure	Diastolic blood pressure
Skin Thickness	Triceps skin fold measurement
Insulin	Serum insulin measurement
BMI	Body mass index
Diabetes Pedigree Function	Hereditary risk-related measure
Age	Age of the individual

The eight predictors were retained as numerical input features. No one-hot encoding was required because the variables used by the model are numerical.

3.2. Handling Zero Measurements

One of the first preprocessing steps was to examine the numerical entries for values that do not represent realistic physiological measurements. Zero values in Glucose, Blood Pressure, Skin Thickness, Insulin, and BMI were considered invalid rather than genuine measurements. Instead of removing the corresponding patient records, these entries were replaced using the median value of their respective features. This approach preserves the available observations while limiting the effect of extreme values on the replacement process [11,12].

3.3. Distribution Analysis

The numerical variables were examined before model development to identify skewed distributions and possible relationships between features. The analysis showed that some variables did not follow a symmetric distribution. This observation was considered when designing the preprocessing pipeline. Rather than applying the same assumption to every feature, the Yeo–Johnson transformation was introduced to reduce the effect of skewness [5].

3.4. Yeo–Johnson Transformation

The Yeo–Johnson method was selected because it can transform numerical variables without requiring all input values to be strictly positive. For a non-negative value, its transformation can be expressed as:

$$x^{(\lambda)} = \begin{cases} \frac{(x+1)^{\lambda}-1}{\lambda}, & \lambda \neq 0 \\ ln(x+1), & \lambda = 0 \end{cases} \quad (1)$$

The transformation parameter is estimated from the available training data. The objective is to obtain a more suitable numerical representation before classification rather than to change the interpretation of the original clinical variable.

3.5. Standardization

Following transformation, the numerical features are standardized. For each feature, the standardized value is calculated using:

$$z_i = \frac{x_i - \mu}{\sigma} \quad (2)$$

where μ is the feature mean and σ is its standard deviation. This step prevents variables with larger numerical ranges from dominating the learning process. It is particularly relevant to Logistic Regression and SVM, where feature magnitude can influence the optimization process.

4. Proposed Framework

The complete analytical workflow is organized as a sequence of data preparation and learning stages:

$$D \rightarrow C \rightarrow T \rightarrow N \rightarrow B \rightarrow M \rightarrow R \quad (3)$$

where D represents the original dataset, C represents data cleaning, T represents transformation, N represents normalization, B represents balancing, M represents model training, and R represents result evaluation.

The workflow begins by loading and inspecting the dataset. Invalid zero measurements are then identified and replaced. Exploratory analysis is performed to understand the numerical variables and target distribution. The Yeo–Johnson transformation is subsequently applied, followed by standardization. SMOTEENN is then used to address the unequal representation of the outcome classes. The resulting training data is supplied to the four selected classifiers. Their predictions are evaluated on the reserved test data.

This sequence is referred to as a hybrid data analytics framework because the final prediction is supported by several complementary data-processing and learning techniques rather than a single algorithm.

5. Class Balancing

The target variable does not contain an equal number of observations for the two outcome categories. This imbalance can influence a model toward the more frequently represented class. SMOTEENN was selected to address this issue [6–8]. The SMOTE portion creates new minority observations between existing minority samples. A synthetic observation can be represented as:

$$x_{new} = x_i + \alpha\left(x_j - x_i\right) \quad (4)$$

where xi is a minority observation, xj is one of its selected minority neighbours, and α is a randomly selected value in the range $[0, 1]$. After synthetic oversampling, Edited Nearest Neighbours removes samples whose local neighbourhood indicates that they may not belong to the same class. This two-stage procedure was included to both increase minority representation and reduce potentially noisy samples around class boundaries.

6. Classification Models

Four models were selected to provide different types of decision behaviour:

6.1. Logistic Regression

Logistic Regression was used as a baseline model because it provides a relatively simple relationship between the input variables and the probability of the target class [13,14]. The probability is represented by:

$$P(y = 1) = \frac{1}{1 + exp\left(-\left(\beta_0 + \sum_{i=1}^{n} \beta_i x_i\right)\right)} \quad (5)$$

Its comparatively simple structure makes it useful for judging whether more complex models provide a meaningful improvement.

https://doi.org/10.3390/engproc2026163106

6.2. Decision Tree

The Decision Tree classifier divides observations through a sequence of feature-based decisions [15]. Each split attempts to separate the target classes more effectively. Unlike a strictly linear model, the tree can represent nonlinear relationships between variables.

6.3. Support Vector Machine

SVM attempts to identify a separating boundary between the classes while maximizing the margin between observations [9,15]. The method can also represent nonlinear boundaries depending on the selected kernel. Standardized features are important because the model relies on geometric relationships between observations.

6.4. Random Forest

Random Forest was selected as the ensemble model in the comparison. Instead of depending on one decision tree, it constructs multiple trees and combines their predictions [8,9]. For a classification problem, the final prediction can be described as:

$$\hat{y} = mode\{T_1(x), T_2(x), \ldots, T_k(x)\} \quad (6)$$

where each *Ti* is an individual tree. GridSearchCV was used to examine predefined Random Forest parameter combinations and identify the configuration used for the final model [16].

7. Experimental Procedure

The dataset was divided into training and testing portions, with approximately 80% used for training and 20% reserved for testing. The training data was used for model development, while the held-out test data was used to obtain the reported classification accuracy.

The preprocessing workflow included invalid-value treatment, Yeo–Johnson transformation, standardization, and SMOTEENN class balancing. Four models were then trained using the resulting processed data. Python 3.13.16 was used for data processing and machine learning implementation [16]. Standard machine learning libraries were used for classification, transformation, scaling, and hyperparameter search.

The primary comparison measure reported in this study is classification accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (7)$$

where *TP*, *TN*, *FP*, and *FN* denote true-positive, true-negative, false-positive, and false-negative predictions, respectively.

8. Results

The final classification results are presented in Table 2.

Table 2. Prediction accuracy obtained from the four models.

Model	Accuracy
Logistic Regression	91.04%
Decision Tree	94.02%
Support Vector Machine	92.53%
Random Forest	95.52%

Random Forest produced the highest accuracy among the evaluated models, reaching 95.52%. Decision Tree was the second-best model at 94.02%. SVM produced 92.53%,

while Logistic Regression recorded 91.04%. The difference between the highest and lowest reported accuracies is 4.48 percentage points. Random Forest also exceeded the Decision Tree result by 1.50 percentage points.

The results suggest that the ensemble representation used by Random Forest was better suited to the relationships present in the processed dataset than the other evaluated classifiers [9,11,12]. The performance of Decision Tree was also relatively high, indicating that tree-based models were effective for the available feature set.

SVM achieved a lower accuracy than the two tree-based approaches in this experiment. Logistic Regression produced the lowest accuracy among the four models, which may indicate that the relationships between the clinical variables and diabetes outcome are not fully represented by a simple linear decision boundary [13].

These observations apply specifically to the dataset and experimental configuration used in this study. They should not be interpreted as evidence that Random Forest will always outperform the other methods on independent clinical datasets.

9. Feature Importance

Feature importance was examined using the Random Forest model [9,10]. The resulting order is given in Table 3.

Table 3. Feature importance ranking.

Position	Feature
1	Glucose
2	Age
3	Insulin
4	BMI
5	Skin Thickness
6	Pregnancies
7	Diabetes Pedigree Function
8	Blood Pressure

Glucose occupies the first position in the importance ranking. Age and insulin follow, with BMI and skin thickness also contributing to the model. Pregnancies, diabetes pedigree function, and blood pressure appear lower in the observed ranking.

The importance values describe how the trained model used the available variables for prediction. They should not be interpreted as evidence that a higher feature importance directly represents medical causation.

The prominent position of glucose is nevertheless meaningful in the context of the dataset because glucose measurements provide substantial information for distinguishing the two outcome classes [2,11].

10. Discussion

The experimental results highlight the effect of combining several data-processing operations before classification [3,11,12]. The dataset is not treated as immediately ready for model training. Instead, invalid measurements are addressed first, feature distributions are examined, skewness is reduced, and the numerical variables are standardized.

The use of SMOTEENN provides an additional step for dealing with unequal class representation [6–8]. Rather than simply duplicating minority observations, SMOTE gen-

erates synthetic samples and the ENN stage removes observations that do not agree with their local neighbourhood.

The four models provide different levels of performance. Logistic Regression provides the simplest baseline, while Decision Tree and Random Forest can represent nonlinear relationships [15]. SVM provides another classification strategy based on separation between classes [9].

Random Forest obtained the strongest accuracy in the final comparison. Its ensemble structure allows the model to use multiple tree-based views of the same feature space. This may help explain its higher performance in the current experiment.

The feature analysis adds another perspective to the results. Glucose was ranked as the most influential variable, while age and insulin were next. This provides a basic interpretation of the variables used by the final model, although feature importance alone cannot establish clinical causality.

An important consideration is that the reported accuracy should not be viewed independently of the dataset. A model can obtain strong performance on a benchmark dataset while producing different results when applied to patients from another population [1,2].

11. Limitations

The study has several limitations.

First, the experiments are based on the Pima Indians Diabetes Dataset, which contains a limited number of observations and represents a specific population [1]. Therefore, the results cannot automatically be generalized to every patient population.

Second, the reported results come from the train-test configuration used in the project. An independent external dataset was not used for validation.

Third, only classification accuracy is reported in the present results. Accuracy does not reveal how well the model identifies positive cases or avoids false alarms. Future experiments should therefore include sensitivity, specificity, precision, recall, F1-score, and ROC-AUC [3,12].

Fourth, feature importance from Random Forest should be considered a model interpretation rather than a medical conclusion. Additional explainability techniques and clinical evaluation would be necessary before the predictions could be used in a healthcare environment [10].

Finally, the proposed framework is a computational screening approach. It should not be used as an independent diagnostic mechanism without appropriate medical assessment [2].

12. Future Work

The framework can be extended by testing it with larger datasets containing observations from multiple populations. External validation would help determine whether the model maintains its performance when the data distribution changes.

A future implementation can also include additional evaluation measures, repeated cross-validation, and confidence intervals. This would provide a more complete assessment than accuracy alone.

Explainable artificial intelligence can be incorporated to generate patient-level explanations. Methods such as SHAP can be investigated to determine how individual measurements influence a particular prediction [10].

Another possible extension is the development of a user-facing screening application. Such an application could accept the required clinical measurements and display a predicted risk category. Before such a system is used in practice, however, it would

require independent validation, clinical assessment, privacy controls, and appropriate regulatory review.

13. Conclusions

This paper developed and evaluated a hybrid machine learning framework for diabetes prediction. The approach combines invalid measurement handling, exploratory analysis, Yeo–Johnson transformation, feature standardization, SMOTEENN class balancing, and supervised classification.

Four algorithms were compared using the developed workflow. Logistic Regression achieved 91.04% accuracy, Decision Tree achieved 94.02%, SVM achieved 92.53%, and Random Forest produced the highest accuracy of 95.52%.

The Random Forest feature analysis placed glucose at the top of the importance ranking, followed by age, insulin, BMI, skin thickness, pregnancies, diabetes pedigree function, and blood pressure.

Overall, the experiment indicates that the combination of data preparation and ensemble learning can provide an effective classification approach for the selected diabetes dataset. The result, however, represents performance within the present experimental setting and should not be interpreted as clinical diagnostic accuracy. Broader datasets, independent validation, additional evaluation metrics, and clinical testing are required to establish the practical usefulness of the framework.

Author Contributions: Conceptualization, C.D., P.P.V. and B.D.; methodology, K.K.F.R.; software, C.D.; validation, C.D., P.P.V. and B.D.; formal analysis, C.D.; investigation, C.D.; data curation, C.D., P.P.V. and K.K.F.R.; writing—original draft preparation, C.D.; writing—review and editing, P.P.V., B.D. and K.K.F.R.; supervision, K.K.F.R. and B.D.; interpretation of results, C.D., P.P.V. and K.K.F.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The Pima Indians Diabetes Dataset analyzed in this study is publicly available on Kaggle at https://www.kaggle.com/datasets/kumargh/pimaindiansdiabetescsv, accessed on 5 February 2026.

Acknowledgments: The authors used ChatGPT (GPT-5.2) solely for spelling and grammar checking during manuscript preparation. The use of this AI tool was supervised by the authors, who reviewed and verified all suggested changes and retain full responsibility for the accuracy, originality, and integrity of the manuscript. The authors thank the Department of Biomedical Engineering at Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology for academic support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Smith, J.W.; Everhart, J.E.; Dickson, W.C.; Knowler, W.C.; Johannes, R.S. Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*; American Medical Informatics Association: Washington, DC, USA, 1988; pp. 261–265.
2. American Diabetes Association. Diagnosis and classification of diabetes mellitus. *Diabetes Care* **2014**, *37*, S81–S90. [CrossRef] [PubMed]
3. Kavakiotis, I.; Tsave, O.; Salifoglou, A.; Maglaveras, N.; Vlahavas, I.; Chouvarda, I. Machine learning and data mining methods in diabetes research. *Comput. Struct. Biotechnol. J.* **2017**, *15*, 104–116. [CrossRef] [PubMed]
4. Tigga, N.P.; Garg, S. Prediction of type 2 diabetes using machine learning classification methods. *Procedia Comput. Sci.* **2020**, *167*, 706–716. [CrossRef]

5. Yeo, I.K.; Johnson, R.A. A new family of power transformations to improve normality or symmetry. *Biometrika* **2000**, *87*, 954–959. [CrossRef]
6. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
7. Wilson, D.L. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Trans. Syst. Man. Cybern.* **1972**, *SMC-2*, 408–421. [CrossRef]
8. Batista, G.E.; Prati, R.C.; Monard, M.C. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explor. Newsl.* **2004**, *6*, 20–29. [CrossRef]
9. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
10. Lundberg, S.M.; Lee, S.I. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; pp. 4765–4774.
11. Zou, Q.; Qu, K.; Luo, Y.; Yin, D.; Ju, Y.; Tang, H. Predicting diabetes mellitus with machine learning techniques. *Front. Genet.* **2018**, *9*, 515. [CrossRef] [PubMed]
12. Diab, K.M.; Deng, J.; Wu, Y.; Yesha, Y.; Collado-Mesa, F.; Nguyen, P. Natural Language Processing for Breast Imaging: A Systematic Review. *Diagnostics* **2023**, *13*, 1420. [CrossRef] [PubMed]
13. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
14. Hosmer, D.W.; Lemeshow, S.; Sturdivant, R.X. *Applied Logistic Regression*, 3rd ed.; John Wiley & Sons: Hoboken, NJ, USA, 2013.
15. Quinlan, J.R. Induction of decision trees. *Mach. Learn.* **1986**, *1*, 81–106. [CrossRef]
16. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.