Diabetes Diagnosis Using Machine Learning: High-Accuracy Predictive Modeling with Diverse Data Sources and Methods

Jasmin Patel[1], Pratham Patel[2] ,May Patel[3] and Shankar Paramar[4]
[1-2-3-4] Department of computer Engg., Government Engineering College, Bharuch, India
Email: {jasmin.patel2110, prathamsonara2005, maypatel9228}@gmail.com, shankar.parmar@gtu.edu.in

Abstract

The International Diabetes Federation (IDF) reports that in 2021, roughly 537 million adults between the ages of 20 and 79 were diagnosed with diabetes globally. Should current patterns persist, this figure is projected to increase to 643 million by 2030 and further escalate to 783 million by 2045. The research utilizes two main datasets: the widely recognized Pima Indians Diabetes Database (PIDD) and a collection of approximately 2,000 patient records from a Frankfurt, Germany hospital. The latter dataset shares a similar feature structure with PIDD. A comprehensive analysis was conducted by generating 120 combinations using different datasets, imputation techniques, scaling techniques and machine learning models, including K-Nearest Neighbors (KNN), Logistic Regression, Support Vector Machines (SVM), Decision Trees, and Random Forest models. Extensive exploratory data analysis (EDA) was performed to assess model performance based on accuracy metrics. Our experimental results demonstrated that KNN and SVM provided the highest prediction accuracy, with values of 0.97 and 0.91, respectively, outperforming other models. The study highlights the importance of tailored model and imputation method selection in enhancing predictive accuracy for diabetes diagnosis. By identifying optimal combinations that effectively handle missing data.

***Index Terms*— Diabetes Detection, Machine Learning, Pima Indians Diabetes Database (PIDD), k-NN Classifier, Random Forest Models.**

I. Introduction

A global health crisis has emerged in the form of diabetes mellitus, a chronic metabolic condition. The International Diabetes Federation reported that in 2019, more than 463 million adults globally were affected by diabetes, with expectations of a considerable rise by 2045 [1]. This widespread occurrence puts immense pressure on global healthcare systems, as diabetes is linked to serious health issues such as cardiovascular disease, kidney failure, and neuropathy, all of which contribute to elevated rates of illness and death [2], [3]. Early and accurate detection of diabetes is crucial to implementing timely interventions that can mitigate adverse outcomes and alleviate the healthcare system's burden [4].

Traditional diagnostic methods, such as fasting blood glucose and glycated haemoglobin (HbA1c) tests, are indispensable but often reactive; these methods depend on clinical thresholds, which may delay diagnosis until the disease has progressed [5]. Advancements in data science and ML have introduced innovative diagnostic approaches capable of improving predictive accuracy by identifying complex patterns within high-dimensional medical datasets [6], [7]. These ML algorithms have shown promise in detecting nuanced risk factors through multidimensional data analysis, making them valuable for predictive modelling in diabetes screening [8], [9].

In the field of diabetes prediction, several machine learning techniques have shown significant success. These include Support Vector Machines (SVM), K-Nearest Neighbours (KNN), Logistic Regression, and ensemble methods such as Random Forest. Notably, ensemble approaches have enhanced prediction accuracy by combining the advantages of multiple classifiers. Among these, Random Forest has achieved impressive accuracy rates, reaching up to 83.67% [10]. This study

evaluates the effectiveness of these algorithms using two datasets: the Pima Indian Diabetes Dataset (PIDD) [11] and a dataset from Frankfurt, Germany [12], allowing a comparative analysis across diverse demographics.
The findings revealed that Random Forest and KNN models yielded the highest accuracy rates, achieving scores of 0.95 and 0.948, respectively, outperforming other algorithms in predictive effectiveness. These results highlight the importance of algorithm selection in processing high-dimensional datasets, where the choice of model significantly influences predictive reliability [6], [7]. By adopting these advanced ML methods, this research aims to enhance early detection and enable data-driven interventions, thereby supporting more effective and personalized diabetes management.

II. Literature Review

The accuracy of machine learning models for predicting diabetes is significantly affected by missing values in datasets. These gaps in data can negatively influence model performance. To tackle this challenge, researchers have investigated various data imputation methods. Among these, the K-Nearest Neighbour (KNN) imputer has been noted for its efficacy in filling data gaps, thus improving the precision and dependability of predictive models [13] [14]. Research has shown that the choice of imputation technique plays a crucial role in determining the accuracy of machine learning models used in diabetes prediction systems. K-Nearest Neighbours Imputation (KNNI) has been shown to outperform other methods, achieving over 80% accuracy and precision in diabetes prediction tasks, particularly when critical attributes are missing [15]. However, conventional techniques like mean imputation often fail to enhance classifier performance, highlighting the necessity for more sophisticated methods like Bayesian multiple imputation and Expectation Maximization, which can improve predictive outcomes when data is missing completely at random [16]. Overall, addressing missing values through appropriate imputation techniques is crucial for developing robust predictive models in diabetes detection [17] [18].
The accuracy of machine learning models in diabetes prediction systems is greatly affected by various normalization methods, including standardization and min-max scaling. Standardization, a technique that transforms data to have a zero mean and unit standard deviation, can improve model performance by reducing the impact of outliers. This is particularly beneficial for algorithms that are affected by data distribution, such as Support Vector Machines (SVM) [19]. Conversely, min-max normalization, which scales features to a fixed range, can improve the convergence speed of gradient-based algorithms but may not perform as well with outliers [20]. Studies indicate that models like Decision Trees and Random Forests benefit from these techniques, achieving higher accuracy rates, with Random Forests often outperforming others in diabetes prediction tasks [21] [22]. Ultimately, the choice of normalization technique can lead to variations in model accuracy, emphasizing the need for careful preprocessing in diabetes prediction systems [23].
Research has demonstrated that the performance of these classifiers is substantially improved through feature selection. Random Forest and Decision Trees have proven particularly effective in this regard, owing to their capacity to manage intricate data interactions [24] [25]. Additionally, the use of hyper-parameter tuning and diverse algorithms has been emphasized to enhance model performance [24].
For predicting diabetes, researchers have utilized a range of machine learning techniques, including Logistic Regression, Support Vector Machines (SVM), K-nearest neighbours (KNN), Decision Trees, and Random Forests. These methods have been applied to datasets such as the Pima Indian Diabetes Database and information collected from Frankfurt, Germany. Studies indicate that these algorithms can achieve significant accuracy in predicting diabetes risk factors, with logistic regression and random forest showing robust performance metrics. For instance, one study reported an accuracy of over 82.2% and an AUC of 87.2% using these methods [26]. A separate study demonstrated that KNN and random forest algorithms were successful in forecasting blood glucose levels, yielding high levels of accuracy [27]. These results collectively emphasize the promising applications of machine learning approaches in improving the prediction and management of diabetes [28].

III. Methodology

The methodology begins with selecting two datasets: the Pima Indians Diabetes Dataset (PIDD) and a dataset of approximately 2,000 patient records from a Frankfurt, Germany hospital. To address missing data, six imputation methods were employed, including No Imputation, Mean, Median, k-Nearest Neighbour (KNN), Random Sample, and Iterative Imputation (MICE).
Preprocessing techniques such as standardization and normalization were applied to optimize data for model training. Five machine learning models were tested: Logistic Regression, Support Vector Machine (SVM), k-Nearest Neighbours (KNN), Decision Tree, and Random Forest.
To evaluate the impact of different combinations, 120 unique configurations of datasets, imputation methods, preprocessing techniques, and classification models were tested. Each configuration was assessed based on accuracy metrics, providing a comprehensive analysis of predictive performance.
The flowchart illustrates one complete cycle in the process of evaluating the 120 unique combinations of dataset configurations. Starting with the raw dataset, an imputation method is applied to handle missing data, followed by the use of a scaling technique (e.g., standardization or normalization) to prepare the data for machine learning models. The pre-processed data is then fed into a specific machine learning model, and the results are evaluated based on accuracy metrics.

Repeating this cycle with various combinations of imputation methods, scaling techniques, and models ensures a thorough exploration of their impact on predictive performance as described in Fig. 1.

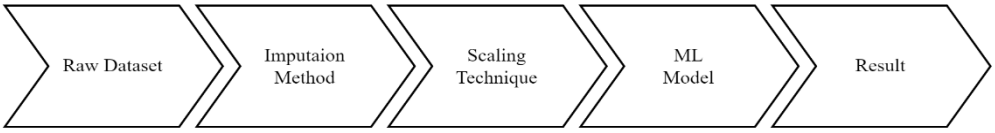

Figure 1. Process Flow Diagram

A. Dataset Description

The Pima Indian diabetes (PID) and the Germany diabetes datasets. Both are publicly available. The PID dataset, obtained from Kaggle, originated from the "National Institute of diabetes and digestive and kidney diseases (sNIDDK)" and was employed to determine diabetes presence in patients. It comprises information on 768 females across eight attributes. The second dataset, provided by Hospital Frankfurt in Germany, includes 2,000 cases with eight features. Both datasets contain patient information such as glucose levels, insulin, and age. Missing data for specific features is denoted by '0' values.

B. Data Imputation

In our datasets, we have missing values in the form of 0's in the columns excluding Pregnancy. In pregnancy 0 value is possible. To address missing data, six imputation techniques were applied: no imputation (0 values retained), mean imputation, median imputation, k-Nearest Neighbour (k-NN) imputation, random sample imputation, and iterative imputation (MICE).

No imputation refers to the approach where missing values are left unaltered, either as zeros (if initially replaced) or left blank. This approach serves as a baseline to understand the impact of missing data on model performance. In the case of missing values, various imputation techniques can be employed. One approach involves substituting absent data points with the arithmetic mean of the observed values for each variable, known as mean imputation. Another method, median imputation, fills in gaps using the middle value of each feature's data set. Alternatively, k-Nearest Neighbours (k-NN) imputation utilizes information from the k most similar data points to estimate and replace missing values. This method finds the k samples with the most similar values to the row with the missing data and takes an average (or weighted average) of those values. Random sample imputation fills missing values with randomly selected observed values from the same feature. By sampling from existing values, this approach retains the original distribution and variance of the data. Multiple Imputation by Chained Equations (MICE), or iterative imputation, is an advanced method that imputes missing values through an iterative process. Each feature with missing data is modelled as a function of the other features in the dataset, and values are imputed in a chained sequence of regressions until they converge. These methods are compared to assess their impact on model performance across different preprocessing and modelling stages.

C. Pre-Processing Phase

Two scaling techniques, standardization and normalization, were applied to improve model performance by ensuring feature comparability.

Data that has been rescaled to have a mean of 0 and a standard deviation of 1 is called Standardization. Transforms features to have a consistent scale without bounding values to a fixed range, which helps reduce bias from features with larger scales.

$$z = \frac{x - \mu}{\sigma} \qquad (1)$$

Normalization rescales data to fit within a fixed range, usually [0, 1]. Transforms features to a bounded range, reducing the influence of large feature values and ensuring consistency in range across features.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \qquad (2)$$

A 70/30 split was used to separate the data into training and testing sets after imputation.

D. Algorithm Selection and Hyperparameter Tuning

The process of fine-tuning a machine learning model's setting, or hyperparameters, to optimize its performance on a particular task is known as hyperparameter tuning. The behaviour of the model and training dynamics are impacted by hyperparameters, which are different from model parameters in that they are set before training rather than being learnt from the data. In order to find the combination of hyperparameter values that maximizes model performance according to a selected evaluation metric, GridSearchCV, a machine learning hyperparameter tuning technique, employs a brute-force methodology.

If a model has k hyperparameters, with each parameter having $n_1, n_2, \ldots, n_k$ possible values, the total number of combinations, N, that GridSearchCV evaluates is:

$$N = n_1 \times n_2 \times \ldots \times n_k \qquad (3)$$

Using k-fold cross-validation, GridSearchCV splits the training set into k equal-sized subsets, or "folds." One-fold is utilized for validation and k−1 folds are used for training for every combination of parameters. Each fold serves as a validation set once during the k repetitions of this operation. A dataset with N observations is mathematically divided into k subsets of size N/k.

GridSearchCV selects the hyperparameter set with the highest average metric score (or lowest, if optimizing a loss). If θ^* represents the best hyperparameters and $M(\theta)$ represents the metric score of parameters θ, then the best hyperparameters are:

$$\theta^* = arg\ \max_{\theta} \bar{M}(\theta) \quad (4)$$

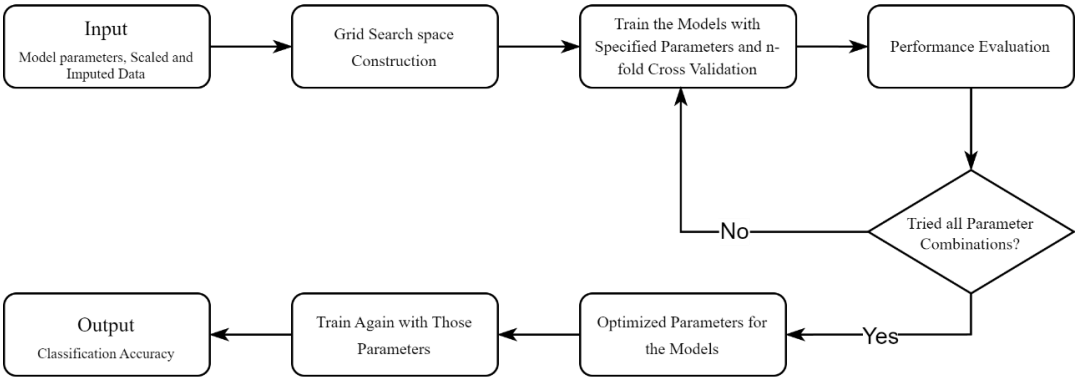

Figure 2. Implementation Flow

Once the best hyperparameters θ^* are identified, GridSearchCV retrains the model on the full training dataset using θ^*. This final model is ready for testing or deployment.

GridSearchCV is useful but occasionally limited for large datasets or models with a broad hyperparameter space because of its high number of evaluations, which can make it computationally costly despite its thoroughness.

Logistic regression, Support Vector Machine (SVM), k-Nearest Neighbours (KNN), Decision Tree, and Random Forest are the five machine learning methods used for this study. Each algorithm offers unique advantages in terms of handling data, interpretability, and predictive power, making them valuable choices for exploring patterns within the datasets.

Logistic Regression is a linear statistical model extensively utilized for binary classification. By using a logistic function that scales output probabilities between 0 and 1, it calculates the likelihood that a given input belongs to a specific class, making it appropriate for probabilistic interpretation. Despite being simple and very interpretable, logistic regression may have trouble with data that has non-linear connections. GridSearchCV was used in this work to adjust its solver parameter, specifically investigating the "lbfgs" and "liblinear" solvers, in order to enhance convergence and computing efficiency according to the properties of the data.

Support Vector Machine (SVM) is a powerful classification model, particularly effective in high-dimensional spaces due to its capability to separate classes with a high-margin hyperplane. By minimizing misclassification, this margin maximization improves the robustness of the model. In order to capture intricate, non-linear correlations, the SVM method also includes kernel functions, which map data into higher dimensions. For this study, we explored various hyperparameters using GridSearchCV: the kernel type ("linear" or "rbf") to determine whether linear or non-linear separation fits the data best, the regularization parameter (C) to control the trade-off between maximizing the margin and minimizing classification errors, and the kernel coefficient (gamma), which influences decision boundaries in non-linear kernels.

An instance-based, non-parametric learning approach called k-Nearest Neighbours (KNN) classifies a data point according to the majority class of its k-nearest neighbours. Although it can be sensitive to the choice of k and distance metric, it is computationally straightforward and efficient for some workloads. While smaller values of k are more affected by noise, larger values of k may smooth forecasts but run the risk of ignoring local structure. In our implementation, GridSearchCV was used to optimize the number of neighbours (n_neighbours), the weighting method (either "uniform" or "distance-based"), and the distance metric (Euclidean or Manhattan) to balance accuracy with computational feasibility.

For both classification and regression applications, decision trees offer a hierarchical, tree-like structure. At each node, they generate decision rules by recursively dividing the data according to feature values. Decision trees can handle non-linear data and are simple to understand, but they are prone to overfitting. To counteract this, GridSearchCV was used to explore several parameters, including the maximum tree depth (max_depth) to control overfitting, the splitting criterion (either "gini" or "entropy") to determine information gain for each split, and the minimum samples required to split a node (min_samples_split), which helps prevent overly complex tree structures.

The overfitting issues with individual decision trees are addressed by the Random Forest ensemble learning technique. Random Forest combines predictions from several models to increase accuracy and stability by creating a "forest" of trees trained on various subsets of data and attributes. In order to balance model complexity and generalization, GridSearchCV was used in this study to adjust the number of estimators (n_estimators), which controls the number of trees in the forest; the splitting criterion (either "gini" or "entropy") for tree node splits; and the maximum depth (max_depth) of each tree.

E. Evaluation Metrics

In machine learning and classification tasks, evaluating model performance accurately is crucial, and various metrics serve this purpose, each with unique insights. Metrics like accuracy, precision, recall, and F1-score were used to assess the model's performance.

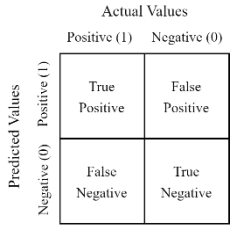

Figure 3. Confusion Matrix

The simplest statistic is Accuracy, which shows the proportion of cases that are properly classified out of all the occurrences. While it provides an overall measure of correctness, accuracy can be misleading if the dataset is imbalanced, as it does not distinguish between different types of errors.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

Precision, which emphasizes positive predictions, is determined by dividing the total number of anticipated positives (false positives and true positives) by the number of true positives (accurate positive predictions). Precision shows how many of the model's positive predictions are truly accurate, which is very helpful when false positives are expensive.

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

Recall (or true positive rate) is the ratio of true positives to total actual positives meaning it includes all the correctly predicted correct output. It evaluates the model's efficiency to detect all relevant cases and it is essential in algorithms like disease detection, where false negatives are very costly since this considers how well a terrific algorithm can catch real positives.

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

F1-Score computes the harmonic mean of precision and recall to merge them into a single metric. It offers a more nuanced perspective of performance in situations where accuracy alone might not be adequate, making it especially helpful when striking a balance between precision and recall is required. Higher values on the F1-score scale, which goes from 0 to 1, indicate a better balance between recall and precision. When taken as a whole, these indicators provide a thorough picture of model performance, enabling more intelligent model evaluation and selection across a range of applications.

$$F1-Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

These metrics were chosen for their ability to capture different aspects of classification performance, particularly in healthcare contexts where false negatives and positives carry different risks.

IV. RESULTS

The accuracy, precision, recall, and F1 scores of each model applied to the variously imputed and pre-processed datasets are presented in this part, together with information on the PID and Frankfurt hospital datasets.

As described in Fig. 4(a) Random Forest shows the highest accuracy with No Imputation (0.779), followed by Mean Imputation (0.762) and KNN Imputation (0.762). This suggests that Random Forest performs consistently well with minimal data manipulation. SVM also performs well across various imputations, with the highest accuracy observed for Mean Imputation (0.762). This performance is close to that of Random Forest, indicating that SVM is fairly robust with different imputation techniques. Decision Tree performs best with No Imputation (0.775), while other imputation methods, such as Median and Mean, yield slightly lower accuracies around 0.736 to 0.758. KNN Classifier has relatively lower performance, with Mean Imputation yielding the highest accuracy (0.688). This suggests that KNN may not be as effective on this dataset regardless of the imputation strategy used. Logistic Regression shows reasonably consistent results across different imputations, with the highest accuracy using Random Sample Imputation (0.762) and No Imputation (0.758).

As described in Fig. 4(b) Random Forest again achieves the highest accuracy with No Imputation (0.766), which aligns with the results from Fig. 4(a), suggesting robustness without imputation. KNN Classifier shows improved performance with Random Sample Imputation (0.766), in contrast to its lower scores in Fig. 4(a). This suggests that KNN benefits more from Random Sample Imputation in this dataset. SVM and Decision Tree show moderate performance across all imputation methods, with No Imputation and Mean Imputation generally providing the best results around 0.753 and 0.732, respectively. Logistic Regression displays consistent performance across imputations, with accuracy values around 0.736 to 0.753,

showing little sensitivity to the choice of imputation. MICE Imputation shows slightly lower performance across most models in both Fig. 4(a) and Fig. 4(b), indicating it may not be as effective in PID dataset.
From both Fig. 4(c) and Fig. 4(d), KNN imputation combined with the K-Nearest Neighbours (KNN) model consistently yields the highest accuracy, reaching up to 0.962 and 0.957, respectively. Mean and no imputation methods also perform relatively well, particularly with the KNN model, suggesting that handling missing data without complex imputation can still yield strong results for this model. Interestingly, Support Vector Machine (SVM) shows strong performance with KNN imputation (0.917) using standard scaling in Fig. 4(d), which contrasts with its moderate scores using min-max scaling in the Fig. 4(c), suggesting it may be sensitive to changes in data imputation. Models such as Logistic Regression and Decision Tree maintain similar accuracy across different imputation methods but generally score lower than KNN and SVM. Random Sample imputation tends to yield slightly lower scores across models, indicating that random sampling might be less effective in handling missing values for Frankfurt Dataset. KNN imputation paired with either KNN or SVM models appears to be the most effective approach in terms of accuracy, while random sampling is generally the least effective. This indicates that using more sophisticated imputation techniques like KNN or retaining the original data structure without imputation (in specific models) can enhance predictive accuracy for Frankfurt Dataset.

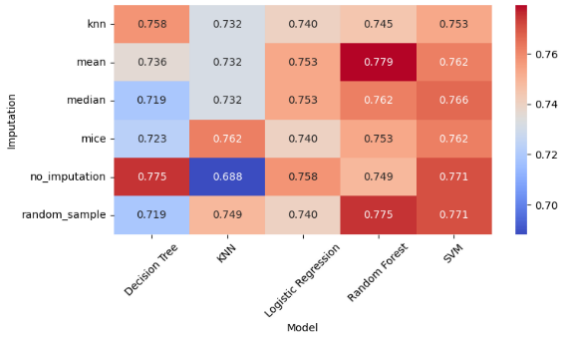

Figure 4(a). Accuracy Heatmap using min-max scaling for PIDD

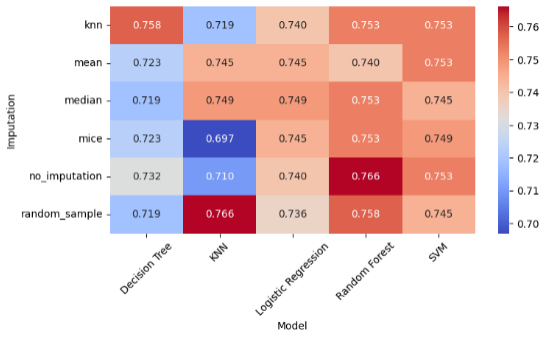

Figure 4(b). Accuracy Heatmap using standard scaling for PIDD

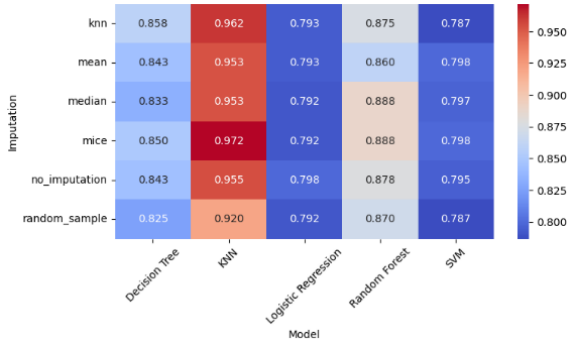

Figure 4(c). Accuracy Heatmap using min-max scaling for Frankfurt Dataset

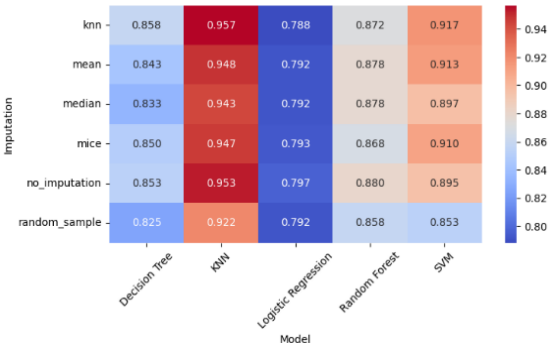

Figure 4(d). Accuracy Heatmap using standard scaling for Frankfurt Dataset

V. Conclusion

Based on the detailed methodology and results, we can draw several key conclusions regarding effective imputation and model selection for diabetes prediction across the datasets.
Five classification models (Logistic Regression, SVM, KNN, Decision Tree, and Random Forest) were tested on two different datasets, and six imputation approaches (No Imputation, Mean, Median, k-Nearest Neighbour, Random Sample, and Iterative Imputation) were investigated. The experimental design incorporated a variety of data preprocessing steps, including standardization and normalization, to ensure that each model received optimized inputs, and the comprehensive grid allowed us to assess 120 unique combinations.
The analysis found Random Forest and SVM to be the most robust models overall, particularly with simpler imputation methods like Mean and No Imputation on the first two datasets, indicating these models can maintain high accuracy without complex imputation. For the Frankfurt dataset, KNN Imputation paired with the KNN model achieved accuracies up to 0.962 and 0.957, demonstrating that using KNN for both imputation and modelling enhances data compatibility and accuracy. Additionally, the combination of mice imputation, normalization, and the KNN model reached the highest accuracy at 0.972 on the Frankfurt dataset. Conversely, Random Sample Imputation generally underperformed across models, likely due to its random nature affecting data distribution, while Logistic Regression and Decision Tree showed stable but lower

accuracy across methods, suggesting they are less impacted by imputation choice but may need further tuning to match the performance of KNN and SVM.

Future studies could use additional datasets with varied demographics and clinical factors to confirm how well the chosen model-imputation combinations work across different populations. Using more advanced imputation methods, like deep learning or ensemble techniques, could also boost model performance, particularly in datasets with high missing values or unusual distributions.

Further analysis of complex ensemble models and neural networks could reveal their fit with different imputation methods and highlight their strengths or limitations compared to traditional models. Additionally, this study's emphasis on standardization and normalization suggests that exploring other preprocessing methods, like robust scaling or dimensionality reduction, may enhance model accuracy and reliability.

REFERENCES

[1] "International Diabetes Federation, 'IDF Diabetes Atlas 9th Edition 2019,' International Diabetes Federation, 2019."

[2] Z. Zhuang, "Enhancing Diabetes Prediction and Management Through Machine Learning Innovations," 2024, pp. 53–59. doi: 10.2991/978-94-6463-540-9_7.

[3] H. Gunathilaka *et al.*, "3 - Non-Invasive Diagnostic Approach for Diabetes Using Pulse Wave Analysis and Deep Learning," *Informatics*, vol. 11, no. 3, 2024, doi: 10.3390/informatics11030051.

[4] N. Shende and P. Bhatele, "Enhancing Diabetes Prediction through Exploratory Data Analysis and Ensemble Learning," 2024.

[5] S. A. El-Aal, R. Salah El-Sayed, A. A. Alsulaiman, and M. A. Razek, "Using Deep Learning on Retinal Images to Classify the Severity of Diabetic Retinopathy." [Online]. Available: www.ijacsa.thesai.org

[6] S. Pang *et al.*, "A novel approach for automatic classification of macular degeneration OCT images," *Sci Rep*, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-70175-2.

[7] A. Hasan Mridul *et al.*, "Polycystic Ovary Syndrome (PCOS) Disease Prediction Using Traditional Machine Learning and Deep Learning Algorithms," 2024.

[8] R. Raman, V. Kumar, D. Saini, D. Rabadiya, S. Patre, and R. Meenakshi, "8 - Advancements in Diabetic Retinopathy and Cataract Identification Through Deep Learning," in *2024 International Conference on Data Science and Network Security (ICDSNS)*, 2024, pp. 1–5. doi: 10.1109/ICDSNS62112.2024.10691108.

[9] Q. Wang and Y. Yang, "Bioinformatics analysis of effective biomarkers and immune infiltration in type 2 diabetes with cognitive impairment and aging," *Sci Rep*, vol. 14, no. 1, p. 23279, Dec. 2024, doi: 10.1038/s41598-024-74480-8.

[10] S. Patil, A. B. Kathole, A. S. Mirge, and H. S. Pathak, "Computational methods for AI-based healthcare Engineering," *Frontiers in Health Informatics*, vol. 13, no. 3, 2024, [Online]. Available: www.healthinformaticsjournal.comwww.healthinformaticsjournal.com

[11] "Pima Indians Diabetes Database," Kaggle. Accessed: Oct. 28, 2024. [Online]. Available: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

[12] "Dataset of diabetes, taken from the hospital Frankfurt, Germany," Kaggle. Accessed: Oct. 28, 2024. [Online]. Available: https://www.kaggle.com/datasets/johndasilva/diabetes

[13] K. Alnowaiser, "Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000. Improving Healthcare Prediction of Diabetic Patients Using KNN Imputed Features and Tri-Ensemble Model", doi: 10.1109/ACCESS.2017.DOI.

[14] A. Altamimi *et al.*, "An automated approach to predict diabetic patients using KNN imputation and effective data mining techniques," *BMC Med Res Methodol*, vol. 24, no. 1, p. 221, Dec. 2024, doi: 10.1186/s12874-024-02324-0.

[15] F. Luo *et al.*, "Missing Value Imputation for Diabetes Prediction," in *Proceedings of the International Joint Conference on Neural Networks*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/IJCNN55064.2022.9892398.

[16] X. Su, T. M. Khoshgoftaar, and R. Greiner, "Using Imputation Techniques to Help Learn Accurate Classifiers."

[17] K. Psychogyios, L. Ilias, C. Ntanos, and D. Askounis, "Missing Value Imputation Methods for Electronic Health Records," *IEEE Access*, vol. 11, pp. 21562–21574, 2023, doi: 10.1109/ACCESS.2023.3251919.

[18] S. Alam, M. S. Ayub, S. Arora, and M. A. Khan, "An investigation of the imputation techniques for missing values in ordinal data enhancing clustering and classification analysis validity," *Decision Analytics Journal*, vol. 9, Dec. 2023, doi: 10.1016/j.dajour.2023.100341.

[19] H. A. Abdelhafez and A. A. Amer, "Machine Learning Techniques for Diabetes Prediction: A Comparative Analysis," *Journal of Applied Data Sciences*, vol. 5, no. 2, pp. 792–807, May 2024, doi: 10.47738/jads.v5i2.219.

[20] J. Shin *et al.*, "Development of Various Diabetes Prediction Models Using Machine Learning Techniques," *Diabetes Metab J*, vol. 46, no. 4, pp. 650–657, Jul. 2022, doi: 10.4093/dmj.2021.0115.

[21] M. Soni and S. Varma, "Diabetes Prediction using Machine Learning Techniques." [Online]. Available: www.ijert.org

[22] M. O. Olusanya, R. E. Ogunsakin, M. Ghai, and M. A. Adeleke, "Accuracy of Machine Learning Classification Models for the Prediction of Type 2 Diabetes Mellitus: A Systematic Survey and Meta-Analysis Approach," Nov. 01, 2022, *MDPI*. doi: 10.3390/ijerph192114280.

[23] K. J. Rani, "Diabetes Prediction Using Machine Learning," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, pp. 294–305, Jul. 2020, doi: 10.32628/CSEIT206463.

[24] K. Kangra and J. Singh, "A genetic algorithm-based feature selection approach for diabetes prediction," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 2, pp. 1489–1498, Jun. 2024, doi: 10.11591/ijai.v13.i2.pp1489-1498.

[25] M. Kawarkhe and P. Kaur, "Prediction of Diabetes Using Diverse Ensemble Learning Classifiers," in *Procedia Computer Science*, Elsevier B.V., 2024, pp. 403–413. doi: 10.1016/j.procs.2024.04.040.

[26] I. J. Kakoly, M. R. Hoque, and N. Hasan, "Data-Driven Diabetes Risk Factor Prediction Using Machine Learning Algorithms with Feature Selection Technique," *Sustainability (Switzerland)*, vol. 15, no. 6, Mar. 2023, doi: 10.3390/su15064930.

[27] X. Fu *et al.*, "Implementation of five machine learning methods to predict the 52-week blood glucose level in patients with type 2 diabetes," *Front Endocrinol (Lausanne)*, vol. 13, Jan. 2023, doi: 10.3389/fendo.2022.1061507.

[28] A. Vimont, S. Béliard, R. Valéro, H. Leleu, and I. Durand-Zaleski, "Prognostic models for short-term annual risk of severe complications and mortality in patients living with type 2 diabetes using a national medical claim database," *Diabetol Metab Syndr*, vol. 15, no. 1, Dec. 2023, doi: 10.1186/s13098-023-01105-x.