Diabetes Classification

Lithigesh P G CB.EN.U4CCE23025
Dept. of Electronics and Communication Engineering(CCE)
Amrita Vishwa Vidyapeetham, Ettimadai,
Coimbatore – 641 112
cb.en.u4cce23025@cb.students.amrita.edu

Nivas G CB.EN.U4CCE23030
Dept. of Electronics and Communication Engineering(CCE)
Amrita Vishwa Vidyapeetham, Ettimadai,
Coimbatore – 641 112
cb.en.u4cce23030@cb.students.amrita.edu

Praveen S CB.EN.U4CCE23035
Dept. of Electronics and Communication Engineering(CCE)
Amrita Vishwa Vidyapeetham, Ettimadai,
Coimbatore – 641 112
cb.en.u4cce23035@cb.students.amrita.edu

***Abstract*— Diabetes classification is a vital application of machine learning that focuses on the early detection and effective management of diabetes. This project implements a multi-class classification approach to predict diabetes levels using various machine learning models. The classification framework categorizes individuals into three distinct classes: Class 0 (No Diabetes), Class 1 (Prediabetes), and Class 2 (Diabetes). To enhance predictive accuracy, the models incorporate feature engineering, data preprocessing, and advanced classification techniques. The performance of the proposed approach is evaluated using key metrics such as accuracy, precision, recall, and F1-score, ensuring a comprehensive assessment of its effectiveness.**

Keywords*— *Machine Learning, Multi-Class Classification, Diabetes Prediction, Data Preprocessing, Feature Engineering, Model Evaluation

I. Introduction

Diabetes is a chronic metabolic disorder that affects millions worldwide, characterized by high blood glucose levels due to insufficient insulin production or ineffective insulin utilization. If left undiagnosed or untreated, diabetes can lead to severe complications, including cardiovascular diseases, kidney failure, vision impairment, and nerve damage. Early detection and classification of diabetes are crucial for timely intervention, effective treatment, and better patient outcomes. Machine learning techniques offer a promising approach to analyzing health-related data and identifying individuals at risk with high accuracy and efficiency.

This project employs various machine learning algorithms to classify diabetes into three categories: No Diabetes, Prediabetes, and Diabetes. The classification models implemented include K-Means Clustering, K-Nearest Neighbors (KNN), Logistic Regression, Decision Tree, Random Forest, Support Vector Machine (SVM), Naïve Bayes, and a Neural Network. Each algorithm is assessed based on its performance using evaluation metrics to determine the most effective model for diabetes prediction.

The dataset used in this project is derived from the Behavioral Risk Factor Surveillance System (BRFSS) 2015 survey and consists of 253,680 data points with 21 features. The target variable, Diabetes_012, categorizes individuals into three classes: 0 (No Diabetes), 1 (Prediabetes), and 2 (Diabetes). The features include various health indicators such as BMI, blood pressure levels, cholesterol levels, physical activity, smoking habits, alcohol consumption, general health status, mental health, physical health, age, education, and income levels. The dataset is preprocessed to handle missing values, normalize features, and balance class distributions where necessary. The goal is to leverage this dataset to train robust models capable of accurately predicting diabetes risk and aiding in early medical intervention.

II. Data Preprocessing

A. Encoding

Categorical columns of the dataset that contain non numerical value, i.e., Smoker, Stroke, Heart Disease and sex which contain yes/no and male/female are label encoded.

B. Handling Missing Values

The missing values are handled by replacing them with mean for numerical columns and mode for categorical columns.

C. Duplicates Handling

Removing duplicate rows is essential to ensure data quality and prevent bias in analysis or model training. Duplicates can skew statistical measures, over represent certain patterns, and lead to overly optimistic model performance. These number of duplicates identified were 8581 and were removed.

D. Class Imbalance handling using SMOTE

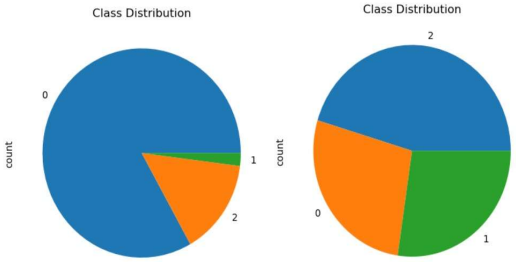

Fig 1. Class distribution before and after SMOTE

Class imbalance occurs when some classes in a dataset are significantly underrepresented compared to

others, causing machine learning models to perform poorly on minority classes.

SMOTE (Synthetic Minority Over-sampling Technique) helps repair this imbalance by generating synthetic (artificial) examples for minority classes rather than just duplicating data.

E. Outlier Handling

Outlier handling identifies and manages extreme values that distort data analysis and model performance. Outlier Bounds are set as 25th percentile – 1.5* IQR and 75th percentile + 1.5*IQR and any values outside the bounds are considered outliers and these are clipped to the bounds.

F. Normalization

The numerical columns were normalized using MinMaxScaler to rescale all features to a [0, 1] range, ensuring equal contribution during model training and preventing variables with larger scales from dominating the learning process.

G. Feature Extraction

Feature extraction was performed to reduce dimensionality and highlight the most informative patterns in the data, improving model efficiency and performance by transforming raw variables into a more discriminative representation. Here correlation is computed between features and the target variable and the features resulting in NAN is dropped.

III. METHODOLOGY

A. K – Means Clustering

K-Means clustering is an unsupervised machine learning algorithm designed to partition data into distinct groups, or clusters, based on similarity. The algorithm works by iteratively assigning each data point to the nearest cluster centroid and then recalculating the centroids as the mean of all points in the cluster. This process continues until the centroids stabilize, minimizing the within-cluster variance. In the context of diabetes classification, K-Means clustering serves as a powerful tool for uncovering hidden patterns in the data.

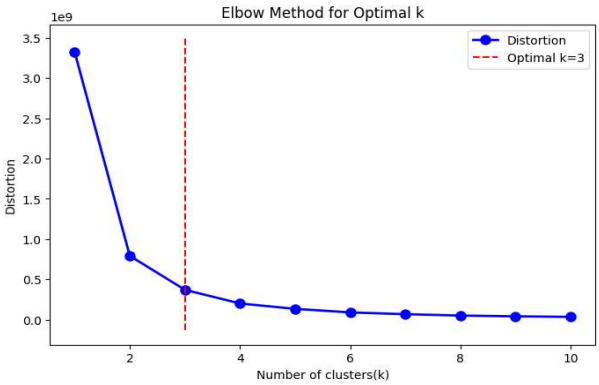

Fig 2. Elbow method for K-means clustering

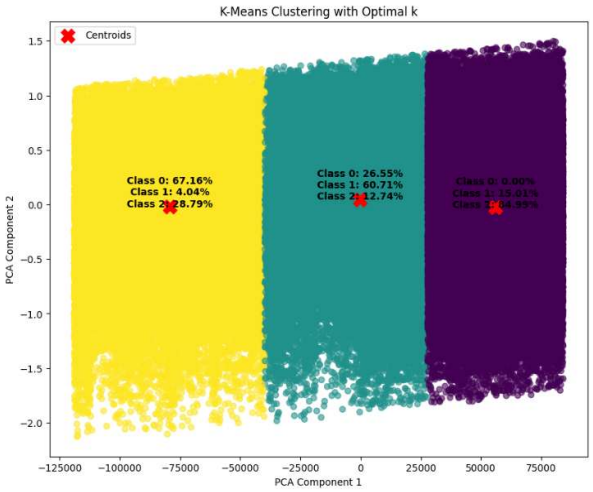

Fig 3. K-means clustering with optimal K

The results from the analysis highlight the effectiveness of K-Means in this application. The Elbow Method identified k=3 as the optimal number of clusters, suggesting that the diabetes dataset naturally divides into three distinct groups. Each cluster is predominantly represented by one class, with Class 0, Class 1, and Class 2 being the majority in their own groups.

B. K Nearest Neighbor (KNN)

KNN is a supervised machine learning algorithm used for both classification and regression tasks. As a instance-based learning method, KNN makes predictions by finding the most similar training neighbors to a new data point and taking a majority vote for classification of these neighbors.

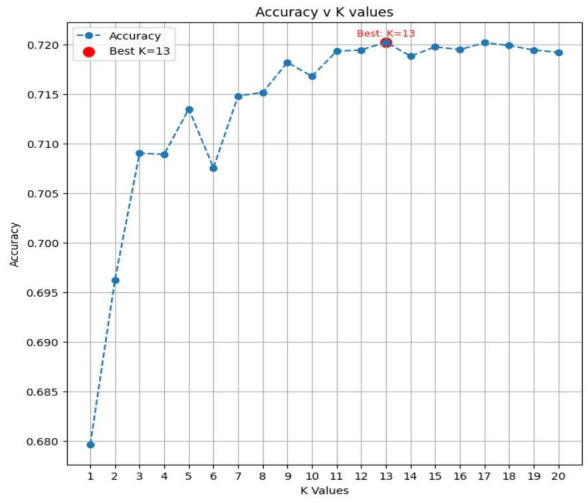

Fig 4. Accuracy vs K values plot

The KNN classifier was evaluated for diabetes classification, with performance metrics analyzed across different values of k. The highest accuracy was achieved when k=13, indicating this number of neighbors provides the optimal balance between model complexity and predictive power.

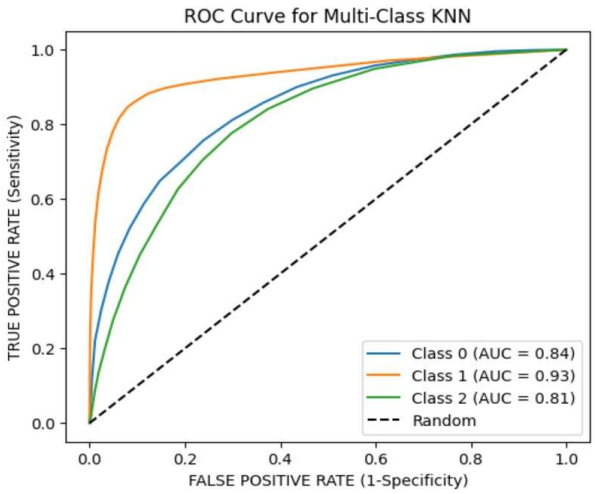

Fig 5. ROC curve for KNN

The ROC curve analysis reinforced these findings, with Class 1 achieving an outstanding AUC of 0.93, indicating near-perfect discriminative ability. Class 2's weaker performance (AUC=0.81) suggests the model struggles more with this class. Possible improvements might involve tackling class imbalance or optimizing features to enhance class separation.

C. Logistic Regression

Logistic Regression is a supervised learning algorithm widely used for classification tasks, including medical diagnostics like diabetes prediction. While traditional logistic regression employs a sigmoid function for binary classification, multiclass problems require an extension using the softmax function. The softmax function generalizes logistic regression by calculating probabilities for each class, ensuring they sum to one, which is essential for distinguishing between multiple diabetes categories (e.g., non-diabetic, prediabetic, diabetic).

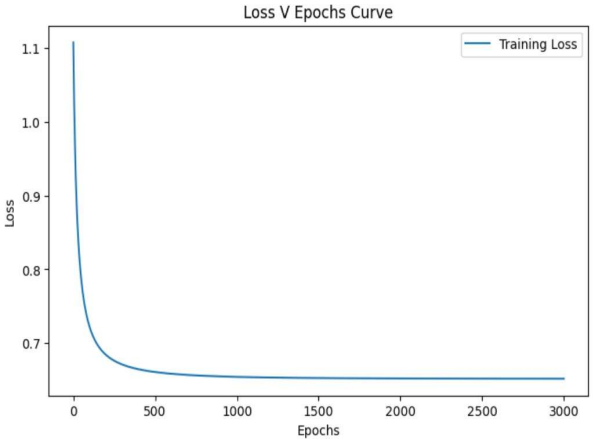

Fig 6. Loss vs epochs curve

The Loss vs. Epochs curve depicts how the training loss evolves as the model iteratively learns from the data. In this case, the loss starts at 0.9 and gradually decreases to 0.7 before plateauing around 500 epochs, indicating that the model's learning rate slows significantly beyond this point. The early decline suggests that the model initially captures meaningful patterns in the data, but the early plateau implies diminishing returns—further training epochs do not substantially reduce the loss.

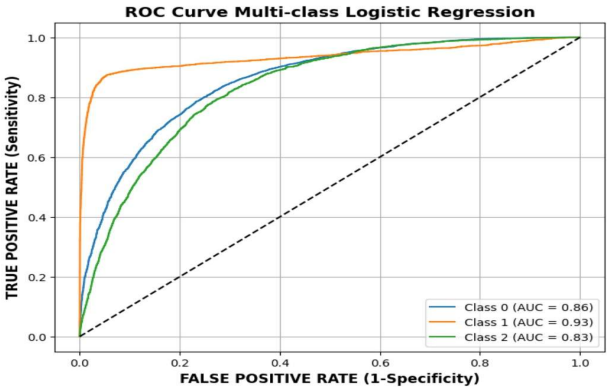

Fig 7. ROC curve for Logistic Regression

The ROC curve analysis of the multi-class logistic regression model reveals strong but uneven performance across classes, with Class 1 showing excellent discrimination (AUC = 0.93), while Classes 0 (AUC = 0.86) and 2 (AUC = 0.83) demonstrate good but comparatively weaker separation. To address this issue, fine-tune decision thresholds per-class to balance sensitivity/specificity based on clinical needs (e.g., higher sensitivity for diabetic cases).

D. Decision Tree

A decision tree is a supervised machine learning algorithm that models decisions and their potential consequences using a tree-like structure. It splits the dataset into branches based on feature values, making sequential decisions to classify data points into target categories. Each node represents a decision rule, each branch represents an outcome of that rule, and each leaf node holds the final prediction (e.g., diabetic or non-diabetic).

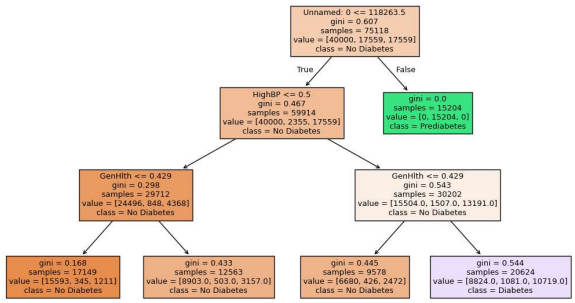

Fig 8. Decision Tree

Gini impurity measures the likelihood of misclassifying a randomly selected sample from a node, with values ranging from 0 (perfect purity) to higher values indicating mixed class distributions. As we traverse down the tree, the Gini impurity systematically decreases at each split, demonstrating how the algorithm partitions the data to create

increasingly pure subsets. This reduction occurs because the tree selects feature thresholds that maximize the separation between classes. Leaf nodes, which represent the final classifications, typically exhibit very low or zero Gini impurity, indicating high confidence in the predicted class. To prevent overfitting and ensure generalizability, thresholds were implemented to control tree growth, including minimum sample requirements for splits and constraints on maximum depth.

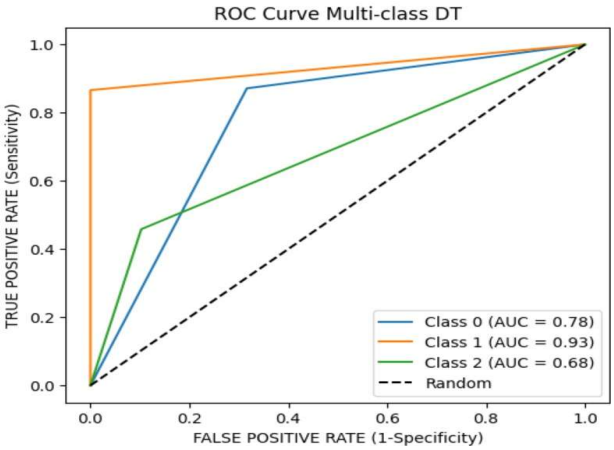

Fig 9. ROC curve for Decision Tree

The ROC curves plot the true positive rate (sensitivity) against the false positive rate (1-specificity) at varying probability thresholds, with the area under the curve (AUC) serving as a key metric to evaluate model effectiveness. Class 1 achieves the highest AUC of 0.93, indicating excellent discriminatory power, as the curve closely follows the top-left corner, reflecting high sensitivity and low false positive rates. Class 0 shows strong performance with an AUC of 0.78, while Class 2 has a comparatively lower AUC of 0.68.

E. Random Forest

Random Forest is an ensemble learning method that builds upon the foundation of decision trees to improve predictive accuracy and control overfitting. Unlike a single decision tree, which can be prone to high variance, a Random Forest combines multiple decision trees—each trained on a random subset of the data and features—to produce a more robust and generalizable model. By aggregating predictions through majority voting for classification, the algorithm reduces individual tree errors and enhances overall stability.

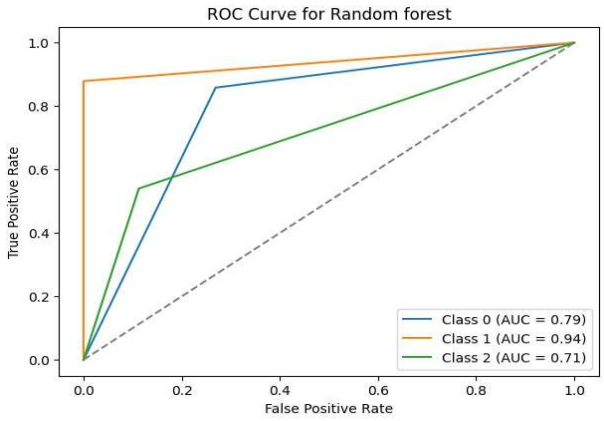

Fig 10. ROC curve for Random Forest

The Random Forest model demonstrates improved performance compared to the single Decision Tree, as evidenced by the higher AUC (Area Under the Curve) values across all three classes. While Random Forest is inherently powerful, its advantage diminishes when dealing with very few classes (e.g., binary or small multi-class problems) because the diversity among trees may not significantly outweigh the simplicity of a single well-tuned Decision Tree. However, in this case, Random Forest still proves beneficial by reducing overfitting—a common issue with individual Decision Trees—through feature and data subsampling, as well as aggregated voting across multiple trees. The Random Forest outperforms the Decision Tree, particularly in improving Class 2's classification, while maintaining robustness against overfitting.

F. Naive Bayes

Naive Bayes is a probabilistic classification algorithm based on Bayes' Theorem, which assumes that features are conditionally independent given the class label—a "naive" assumption that simplifies computation. Naive Bayes calculates the probability of each class directly from the training data, making it fast and scalable even for large datasets.

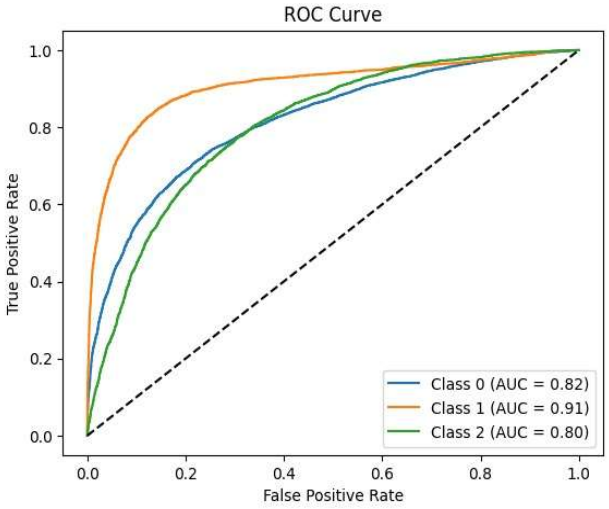

Fig 11. ROC curve for Naïve Bayes

It's key advantages include rapid training (single data pass), natural handling of categorical features, and robustness to irrelevant variables, though its performance may suffer when features interact strongly. Despite its simplicity, the model competes closely with more complex methods (Random Forest AUC: 0.79-0.94), suggesting that for this dataset, feature independence may be a reasonable approximation. Its inherent capacity for incremental learning allows real-time model updates without full retraining - a capability absent in most other algorithms.

G. Neural Network

Neural networks represent a powerful class of machine learning models inspired by the structure and function of biological neurons, capable of learning complex patterns through interconnected layers of artificial nodes. Unlike traditional models such as Naive Bayes or decision trees that rely on explicit statistical assumptions or rule-based splits, neural networks employ a series of hierarchical

transformations to automatically extract and combine features from raw data. However, they typically require substantial computational resources, careful tuning of hyperparameters (e.g., layers, activation functions), and larger training datasets to avoid overfitting.

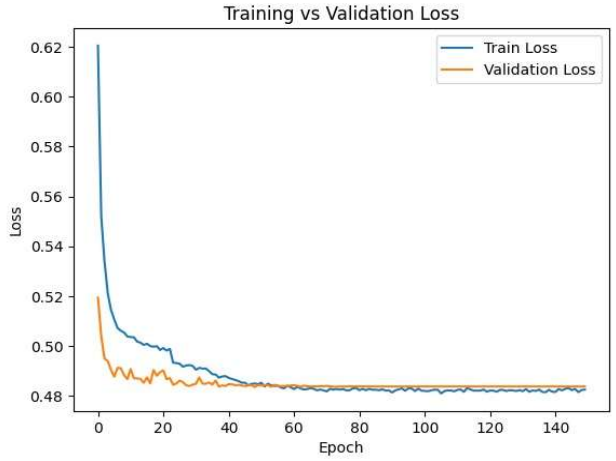

Fig 12. Training vs Validation loss curve

The training and validation loss curves of the neural network demonstrate effective learning for diabetes classification, with both metrics decreasing smoothly over 140 epochs to converge at low values (training loss: ~0.48, validation loss: ~0.50). The parallel decline and minimal gap between the curves indicate the model is generalizing well without overfitting, while the absence of erratic fluctuations suggests stable training with well-chosen hyperparameters like learning rate and batch size. The plateau around epoch 100 implies diminishing returns in later training stages, highlighting the potential for early stopping to optimize computational efficiency.

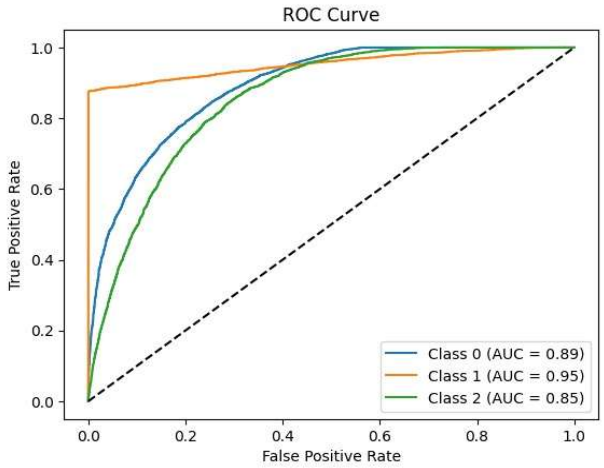

Fig 13. ROC curve for Neural Network

The neural network model demonstrates excellent performance in diabetes classification, as evidenced by its high AUC scores of 0.89 (Class 0), 0.95 (Class 1), and 0.85 (Class 2) on the ROC curve, indicating strong discriminatory power across all categories. The model's architecture - featuring 4 hidden layers with ReLU activation functions - enables it to effectively capture complex, hierarchical patterns in the data while avoiding vanishing gradient issues during training. The use of softmax activation in the output layer provides probabilistic class predictions that are clinically interpretable, while the categorical cross-entropy loss function ensures precise optimization for this multi-class problem.

H. Support Vector Machine (SVM)

Support Vector Machine (SVM) is a supervised learning algorithm that excels at classification tasks by finding the optimal hyperplane that maximally separates different classes in the feature space. It works particularly well with high-dimensional data and can handle both linear and non-linear decision boundaries through kernel functions. SVM focuses on the data points closest to the decision boundary (support vectors), making it robust to outliers and effective even with limited training samples. For multi-class problems like diabetes classification, SVM typically uses strategies like One-vs-Rest (OvR) to extend binary classification to multiple classes.

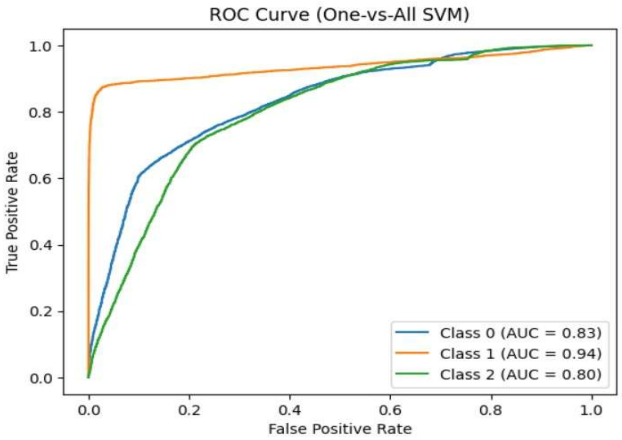

Fig 14. ROC curve for Support Vector Machine

The strong performance, especially for Class 1, highlights SVM's effectiveness in finding optimal decision boundaries when classes are separable, though its slightly lower AUC for Class 2 (0.80) may indicate more challenging edge cases in this category. The model's performance validates its use as a competitive alternative to more complex neural networks for this classification task, particularly when considering SVM's advantages in computational efficiency and clearer margin-based interpretability compared to deep learning approaches.

IV. RESULTS

S.No	Algorithm	AUC (Class 0)	AUC (Class 1)	AUC (Class 2)	AUC (Average)	Accuracy
1	KNN	0.84	0.93	0.81	0.86	0.73
2	Logistic Regression	0.86	0.93	0.83	0.87	0.75
3	Decision Tree	0.73	0.93	0.68	0.78	0.77
4	Random Forest	0.79	0.94	0.71	0.81	0.79
5	Naïve Bayes	0.82	0.91	0.80	0.84	0.71
6	Neural Network	0.89	0.95	0.85	0.90	0.79
7	SVM	0.83	0.94	0.80	0.86	0.70

Fig 15. AUC and Accuracy analysis

The comparative evaluation of diabetes classification models highlights how their inherent algorithmic properties influence performance differently. Neural networks demonstrate superior predictive capability by learning intricate hierarchical patterns through their deep architecture, making them particularly adept at capturing complex relationships in the data. Random forests achieve robust performance by combining multiple decision trees to reduce variance, though they may occasionally miss subtle class boundaries. Support Vector Machines excel at finding optimal separation margins between classes but can be sensitive to data distribution characteristics. Traditional approaches like logistic regression often outperform simpler methods such as KNN and naive Bayes, as the latter face challenges with feature dependencies and distance metric limitations. Notably, single decision trees may show misleading accuracy due to their tendency to overfit the training data. This analysis underscores the fundamental trade-off between model complexity and interpretability - while sophisticated algorithms can achieve higher performance, simpler models often provide more transparent decision-making processes. The choice among these approaches should consider not just predictive power but also clinical applicability, computational requirements, and the need for explainability in medical contexts. Each method brings distinct advantages that make it suitable for different scenarios within diabetes classification tasks.

S.No	Algorithm	Class 0	Class 1	Class 2	Average
1	KNN	0.78	0.81	0.49	0.69
2	Logistic Regression	0.80	0.83	0.52	0.72
3	Decision Tree	0.81	0.93	0.51	0.75
4	Random Forest	0.86	0.94	0.54	0.78
5	Naïve Bayes	0.76	0.74	0.54	0.68
6	Neural Network	0.82	0.93	0.55	0.77
7	SVM	0.72	0.80	0.59	0.70

Fig 16. F1 – score analysis

The F1-score analysis reveals random forests deliver the most balanced performance (average 0.78), while decision trees and neural networks excel in Class 1 identification (F1=0.93), with all models struggling significantly with Class 2 (0.49-0.59). For medical applications like diabetes classification, metric selection should be guided by clinical priorities: F1-scores are crucial for imbalanced data and critical classes (like Class 2), while AUC-ROC provides comprehensive threshold-agnostic assessment. Precision becomes vital when false positives carry high costs (e.g., unnecessary treatments), whereas recall matters most when missing true cases is dangerous (e.g., undiagnosed diabetes). Accuracy serves best for balanced datasets but proves less informative here. This analysis suggests that while random forests offer reliable overall performance, the clinical context should determine whether to prioritize models excelling in specific classes (neural networks for Class 1) or those minimizing particular error types, potentially warranting ensemble approaches to address the persistent Class 2 challenge.

V. CONCLUSION

In conclusion, this comparative analysis of machine learning models for diabetes classification demonstrates that each algorithm serves distinct purposes based on clinical requirements and operational constraints. Neural networks deliver superior predictive performance for complex pattern recognition when computational resources permit, while random forests offer the ideal balance of accuracy and interpretability for most clinical implementations. Support vector machines excel in clear margin-based classification scenarios, and logistic regression provides a simple yet effective baseline model. Decision trees serve well for transparent rule-based decision making, naive Bayes works efficiently in resource-limited settings, and KNN remains viable for small datasets with meaningful feature distances. The persistent challenge in classifying Class 2 cases across all models highlights an important area for future improvement through targeted data enhancement or ensemble techniques. Ultimately, model selection should be guided by specific clinical priorities - whether optimizing for recall to minimize missed diagnoses, precision to reduce false alarms, or interpretability to support medical decision-making - with random forests generally representing the most robust choice for balanced performance across all considerations in diabetes classification tasks.

VI. REFERENCES

[1] Rustam, F., Al-Shamayleh, A.S., Shafique, R. et al. Enhanced detection of diabetes mellitus using novel ensemble feature engineering approach and machine learning model. Sci Rep 14, 23274 (2024).

[2] Iparraguirre-Villanueva, O., Espinola-Linares, K., Flores Castañeda, R.O. and Cabanillas-Carbonell, M., 2023. Application of machine learning models for early detection and accurate classification of type 2 diabetes. *Diagnostics*, *13*(14), p.2383.

[3] Kopitar, L., Kocbek, P., Cilar, L. *et al.* Early detection of type 2 diabetes mellitus using machine learning-based prediction models. *Sci Rep* **10**, 11981 (2020). https://doi.org/10.1038/s41598-020-68771-z.