Optimizing Diabetes Mellitus Classification using Deep Neural Networks on the PIMA Indians Dataset

Aadarsh Anantha Ramakrishnan[1], Selvanayagam S[2], and Dr. R. Murugesan[3]

[1 2 3]National Institute of Technology, Tiruchirappalli, 620015, Tamil Nadu, India

Abstract: Diabetes is a chronic illness that affects a significant number of individuals worldwide. Survey reports that the Pima Indians have one of the highest prevalence of diabetes mellitus globally. This study analyzes the Pima Indians Diabetes Dataset which contains diagnostic measurements of 768 Pima Indian women. Though prior research has predominantly employed several predictive models such as SVM, NB, DT and RF classifiers for predicting diabetes in the PIMA Indians Dataset, all these techniques have been identified with some limitations. This indicates a need for a more accurate predictor and a scope for further study. Hence, this study proposes a Deep Neural Network (DNN) model on the PIMA Indians Dataset. This study shows that the proposed DNN model performs better (81.72%), when compared to other popularly used ML classifiers. Our findings suggest that DNNs are a highly effective supervised learning algorithm, which can be extended to other related fields in medical research.

Keywords: PIMA Indians; Deep Learning; Neural Networks; Diabetes Mellitus

1 Introduction

Diabetes mellitus is a chronic metabolic condition that results in high blood sugar levels due to the body's inability to make or use insulin. [1] It is associated with various health complications such as cardiovascular disease, kidney failure, and stroke. Diabetes prediction is crucial because early identification and intervention can prevent or delay complications. [2] However, we currently lack an accurate predictor as the development of diabetes is influenced by a complex interplay of genetic, environmental, and lifestyle factors, making it difficult to predict with high accuracy.

The Pima Indian population has been extensively studied as they have one of the highest prevalences of diabetes mellitus in the world. [3] The Pima Indians of Arizona have a genetic predisposition to insulin resistance and diabetes mellitus, making them an ideal population for studying the disease's underlying mechanisms. [4] The Pima Indians Diabetes Dataset is a well-known dataset which has been used by several researchers for developing and evaluating machine learning models for predicting diabetes.

Prior research has predominantly employed several predictive models such as Support Vector Machines (SVMs) and Naive Bayes (NB) classifiers separately [5], for this classification task, which are identified with certain limitations. This is indicated by the extant literature, a need for an accurate and better predictor. Hence, the primary aim of this study is to apply a Deep Neural Network and compare its performance with other popular ML techniques such as SVM, KNN, Gaussian Naïve Bayes and Logistic Regression. Our proposed method for predicting diabetes involves a data preprocessing pipeline which includes exploratory data analysis, outlier removal using the Z Score and Inter Quartile Range method and missing data imputation. The DNN model achieves an accuracy of 81.72%, which surpasses the performance of other classifiers. Our prediction model has significant potential in the medical field as a valuable tool for predicting diabetes and helps prevent long-term complications associated with diabetes.

The paper is structured in the following manner: The current section deals with the introduction. The second section discusses the relevant literature pertaining to the proposed model. The third section presents the methodology that includes details about the dataset and the conventional ML techniques used on it. Details about the proposed DNN model have also been explained in detail. The fourth section presents the results and discussions, followed by conclusions in the last section.

2 Literature Survey

The PIMA Indian Diabetes Dataset [6] has been the subject of much research in the field of machine learning, with various methods proposed to achieve accurate classification results. Self-Organizing Map (SOM) method and cross-validation techniques were utilized in Deng and Kasabov's [7] work to successfully categorize the dataset with an accuracy of 78.4%. A helpful tool for grouping and visualization tasks, SOM is an unsupervised learning technique that converts high-dimensional data into lower-dimensional space. Several machine learning techniques were examined in Yu et al.'s [8] study, including Quantum-behaved Particle Swarm Optimization (QPSO), Support Vector Machines (SVM), Weighted Least Squares (WLS), and Neural Network (NN). According to their data, categorization accuracy ranged from 68.6% to 77.8%.

The c4.5 decision tree approach was used in Al Jarullah et al.'s [9] study to categorize the dataset, with an accuracy of 71.1%. The c4.5 algorithm is a well-known decision tree algorithm that constructs a decision tree using information gain and entropy measurements. To reach a classification accuracy of 75.29%, Pasi Lukka's study [10] used feature selection techniques based on fuzzy entropy measurements and similarity classifiers.

In their study [11], Seera et al. suggested a hybrid intelligent system that integrates the DT, RF, and Fuzzy Min-Max neural network to classify the dataset with an accuracy of 71.35%. A particular kind of neural network called a fuzzy min-max network employs fuzzy logic to deal with data imprecision and uncertainty. The Genetic Algorithm (GA) for feature selection and Naive Bayes (NB) for classification were suggested in Choubey et al. [12] study, which had a 78.69% accuracy rate. Naive Bayes is a probabilistic classifier that presupposes independence among features, and Genetic technique is a metaheuristic optimization technique that draws inspiration from the process of natural selection.

The work by Kumari et al. [13] investigated the performance of several kernels and used the SVM model for PIMA classification. Their research claimed a classification accuracy of 75.50% utilizing the RBF kernel and used a cross-validation approach to optimize the SVM hyper-parameters. In a high-dimensional space, the SVM classification method locates the ideal hyperplane to divide data points. In their investigation, Somu et al. [14] combined the Random Forest classification approach with the Rough Set based K-Helly (RSKHT) feature selection strategy. Their study used a variety of machine learning techniques to attain classification accuracy levels ranging from 73.11% to 75.11%. An ensemble learning technique called Random Forest creates several decision trees and then aggregates the results to produce predictions.

Overall, the use of machine learning methods such as SVM, Random Forest, Decision Trees, and Neural Networks has proven effective in the classification of the PIMA Indian dataset. While each method has its own strengths and weaknesses, the development of hybrid intelligent systems and the application of feature selection techniques have shown promise in achieving even greater accuracy in the future.

3 Methodology

3.1 Pima Indians Diabetes Dataset

3.1.1 Data Description

The PIMA Indians Diabetes Dataset [7] contains 8 features:

- Pregnancy History: Number of times pregnant
- Glucose Levels: Plasma glucose concentration in a 2-hour glucose tolerance test
- Blood Pressure: Diastolic blood pressure (mm Hg)
- Skin Thickness: Triceps skin fold thickness (mm)
- Insulin: 2-hour serum insulin (mu/U/ml)
- BMI: Body Mass Index
- Diabetic Pedigree Function: Scores the likelihood of diabetes using family history.
- Age

There are more instances of non-diabetic (labeled 0) individuals (65%) than diabetic (labeled 1) ones (35%). Sample data from the Pima Indians Dataset has been shown in Table 1 and a summary of the dataset (count, mean, standard deviation, minimum/maximum value, 25/50/75%) is presented in Table 2, which is self-explanatory.

S.No	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Table 1: Raw data of Pima Indians Diabetes

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigree Function	Age	Outcome
Count	768	768	768	768	768	768	768	768	768
Mean	3.8	120.8	69.1	20.5	79.9	31.9	0.47	33.2	0.3
Std	3.3	31.9	19.3	15.9	115.2	7.8	0.33	11.7	0.4
Min	0	0	0	0	0	0	0.07	21	0
25%	1	99	62	0	0	27.3	0.24	24	0
50%	3	117	72	23	30.5	32	0.37	29	0
75%	6	140.2	80	32	127.2	36.6	0.62	41	1
Max	17	199	122	99	846	67.1	2.42	81	1

Table 2: Descriptive statistics of the raw data

3.1.2 Exploratory Data Analysis

Exploratory Data Analysis (EDA) is an important initial step, which involves analyzing the dataset to understand its key relationships and characteristics. The goal of EDA is to generate insights and hypotheses, through different graphs, charts, and statistical summaries.

During the process of EDA, it is important to carefully consider missing values, outliers, and anomalies, as they can significantly affect the accuracy of the analysis results. To address missing values, imputation techniques such as using mean/median values or removing the records altogether can be used. Outliers and anomalies can be identified using statistical measures such as standard deviation or interquartile range and can be addressed using techniques such as truncation or winsorization. It is imperative to handle these issues in a careful and considerate manner to ensure that the analysis results are both accurate and reliable. EDA is particularly useful in identifying potential errors and inconsistencies in the data, which can be corrected before further analysis.

In this paper, we analyzed how the likelihood of diabetes is affected by multiple features using EDA. In this EDA, we focused on multivariate data analysis for finding relationships between features. Visualizations and insights found during the EDA have been presented in Figures 1, 2 and 3. Note that in the figures, an outcome of 0 represents non-diabetic people and an outcome of 1 represents diabetic people.

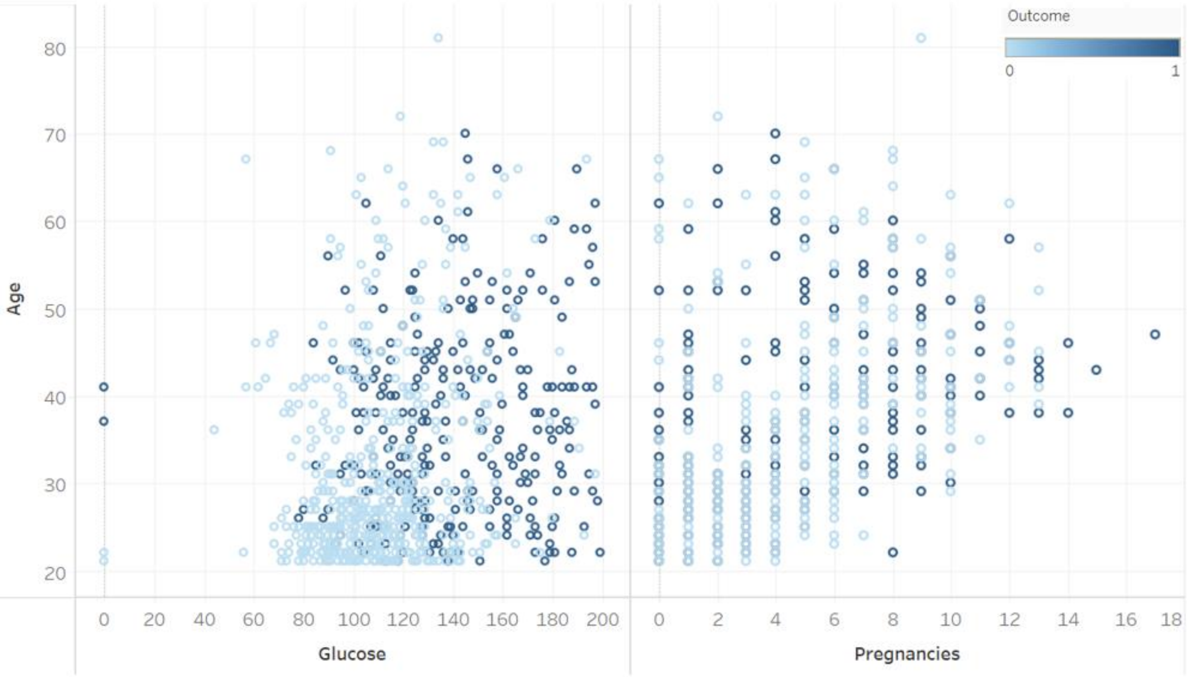

Figure 1: Glucose/Pregnancies vs Age

Figure 1 shows when a person's age is less than 30, and their blood glucose level is less than 120 mg/dL or their number of pregnancies is less than 6, they are less likely to have diabetes.

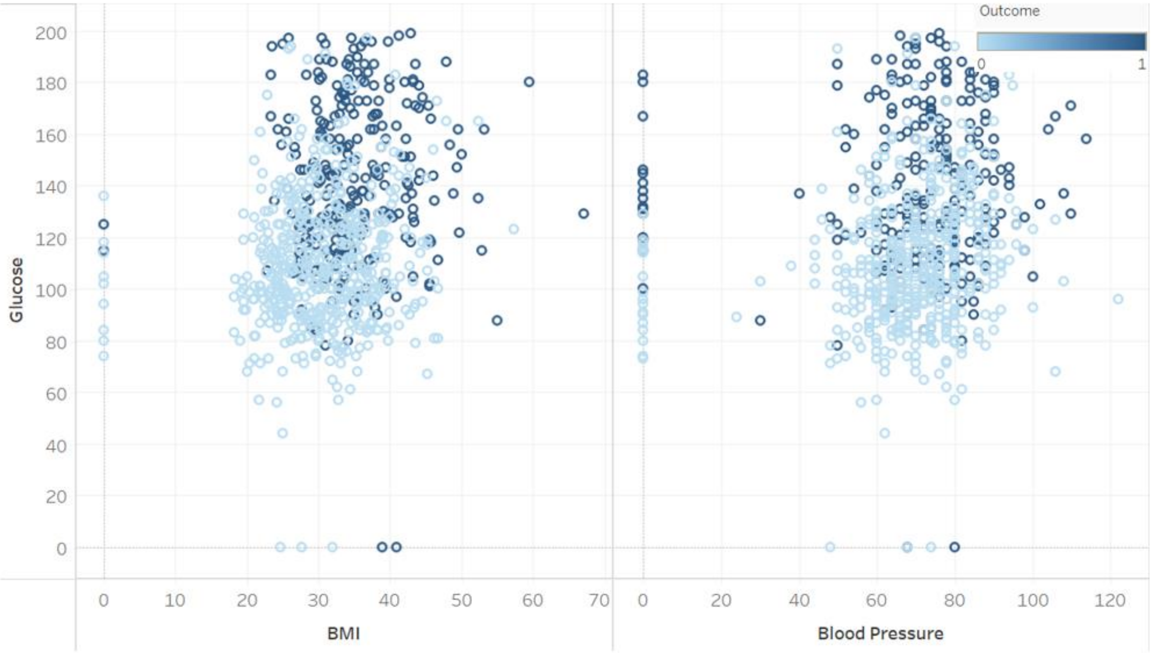

Figure 2: BMI/Blood Pressure vs Glucose

Figure 2 shows when a person's blood glucose level is less than 105 mg/dL, and their BMI less than 30 or blood pressure is less than 80 mmHg, they are less prone to diabetes.

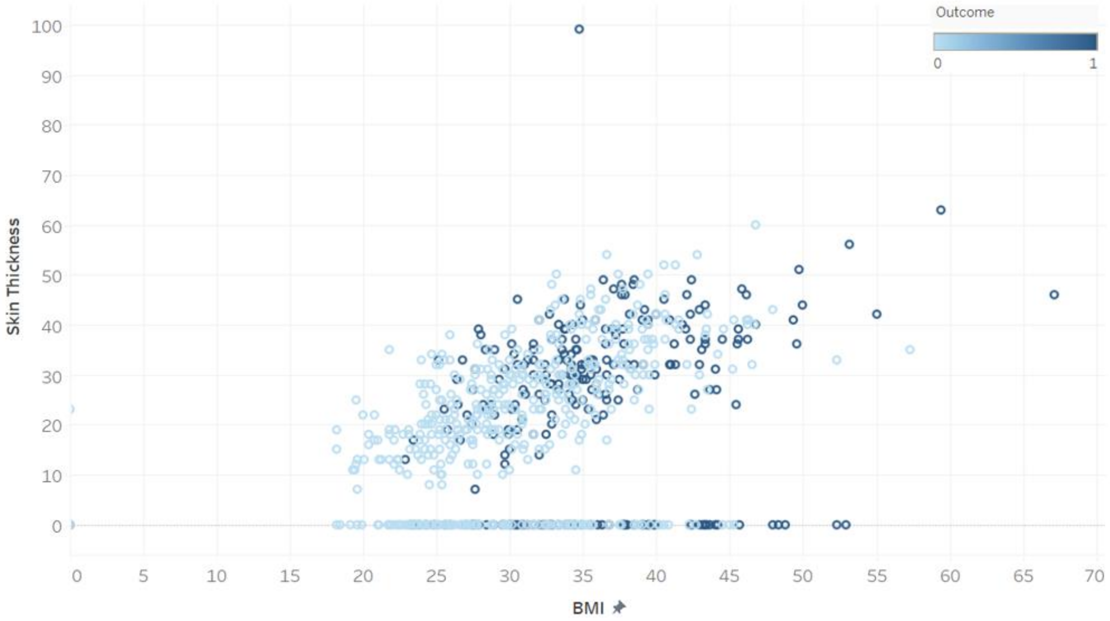

Figure 3: BMI vs Skin Thickness

Figure 3 shows when a person's BMI is less than 30 and their skin thickness is less than 20 mm, they are less susceptible to diabetes.

3.1.3 Data Preparation

One of the biggest challenges with the PIMA Indians Diabetes Dataset is the presence of missing data in the Glucose, Blood Pressure, Skin Thickness, Insulin and BMI features. To combat this issue, this paper uses the following steps:

- Step 1: Remove outlier data points using Z Score and Inter Quartile Range Method.
- Step 2: Replace missing values (having value zero) with the mean value of the feature.
- Step 3: Check the values of all features after adjustments, to ensure the completeness of data.

An outlier is an observation that deviates significantly from the rest of the dataset. These observations can have a significant impact while training ML models, leading to inaccurate results and predictions. By removing outliers, we can better understand the underlying patterns in our data and gather meaningful insights from it. The dataset's outliers have been visualized using a box plot in Figure 4. This paper uses the Z-Score and Inter Quartile Range methods for removing these outliers from the dataset.

The Z-Score method is a type of statistical technique, used for identifying and removing outliers from the dataset. It is based on the Z-Score, which represents the number of standard deviations that the observation is from the mean. The formula for the Z-Score is given by Equation 1, where x is the observation, μ is the feature's mean value and σ is the standard deviation of the feature. In this paper, we have established a threshold value of 3 for Z-Score, above which the observation will be removed from the dataset.

$$Z = \frac{x - \mu}{\sigma}$$

(1)

The Inter Quartile Range (IQR) method is another outlier removal technique, which works by calculating the range between the 25th and 75th percentiles of the data, also known as the interquartile range. This technique is very useful in identifying outliers in non-normal distributions, as it is less sensitive to extreme values than the Z-Score method. Equation 2 describes the function R(x), which uses the IQR method for outlier rejection.

$$R(x) = \begin{cases} x, & if\ Q1 - 1.5 \times IQR \leq x \leq Q3 + 1.5 \times IQR \\ remove, & otherwise \end{cases}$$

(2)

where x is the feature vector, Q1 is the first quartile, Q3 is the third quartile, and IQR is the interquartile range of the attributes.

This paper employs the Z-Score method first, and then uses the IQR method on the remaining records. After outlier removal, the number of records in the dataset comes down to 619. The missing values are imputed by their feature's mean value. Imputation with the mean is useful due to the effective imputation of continous data without the introduction of outliers.

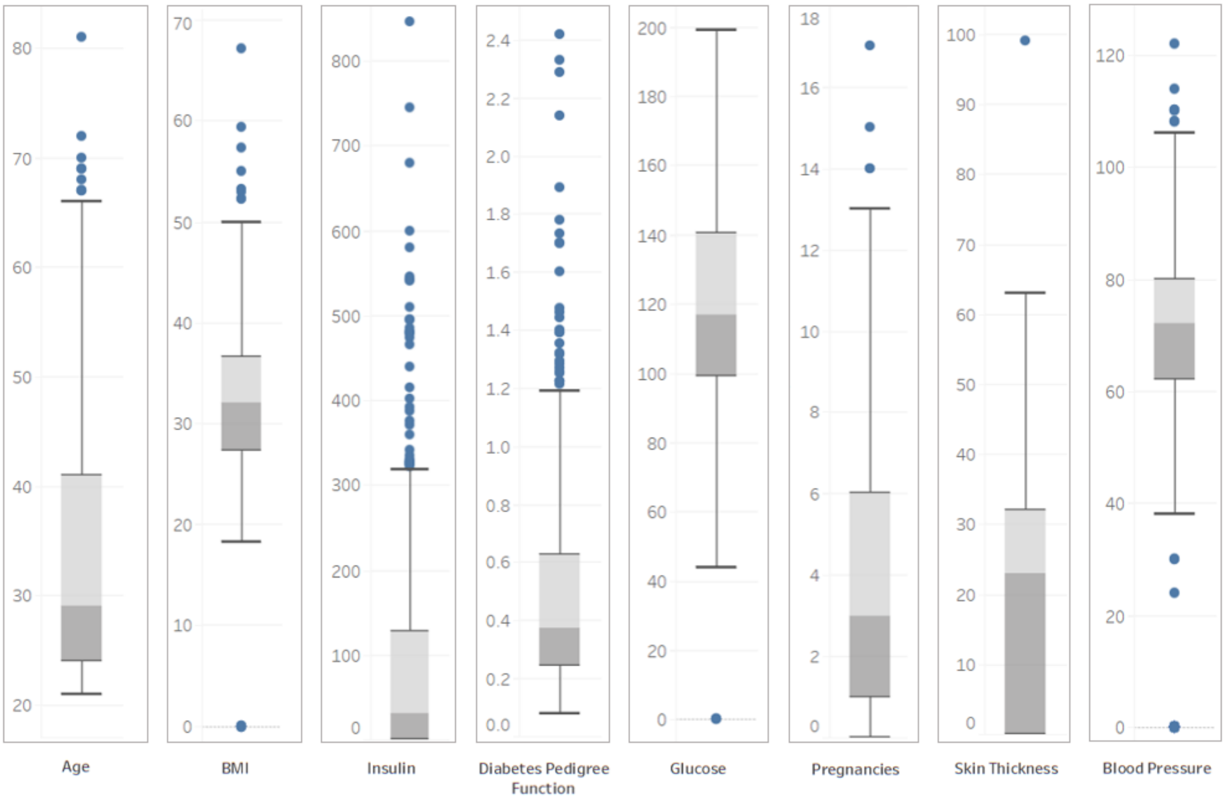

Figure 4: Box plot showing outliers in the PIMA Indians Dataset

3.1.4 Data Standardization

After the above process, it is essential to standardize the dataset as it is a common requirement for various machine learning estimators to perform well for the given dataset. This study utilized the Min Max Scaler standardization algorithm, which scales each feature to the range [0,1]. The formula used by this algorithm is described in Equation 3. The dataset after standardization has been presented in Table 3.

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{3}$$

where X is the input feature, X_{min} and X_{max} are the minimum and maximum values of the feature.

S.No	**Pregnancies**	**Glucose**	**BloodPressure**	**SkinThickness**	**Insulin**	**BMI**	**DiabetesPedigree Function**	**Age**	**Outcome**
0	0.46	0.67	0.46	0.52	0.37	0.49	0.51	0.67	1
1	0.07	0.26	0.36	0.41	0.37	0.26	0.25	0.23	0
2	0.61	0.90	0.33	0.40	0.37	0.16	0.55	0.25	1
3	0.38	0.46	0.50	0.40	0.37	0.23	0.11	0.20	0
4	0.23	0.22	0.10	0.47	0.24	0.40	0.15	0.11	1

Table 3: Raw data after outlier removal, data imputation and standardization

3.2 ML Techniques used for Prediction of Diabetes

In this section, different supervised learning algorithms previously used for classifying Pima Indian women as either diabetic or non-diabetic, are presented below.

3.2.1 Support Vector Machine

Support Vector Machines (SVMs) have been extensively researched and studied in the field of machine learning. The original idea behind SVMs was introduced by Vapnik and Chervonenkis in 1963 [15]. The modern version of SVMs, which incorporates the kernel trick, was developed by Boser, Guyon, and Vapnik in 1992 [16]. Since then, SVMs have been widely used in various applications such as image classification, text classification, and bioinformatics.

The concept of the kernel trick was initially proposed by Aizerman et al. in 1964 [17], and it was later formalized by Mercer in 1909 [18]. The kernel trick enables SVMs to implicitly map the data to be analyzed into a high-dimensional feature space where linear separation is possible. In summary, SVMs are a powerful and widely used machine learning technique that can handle high-dimensional and non-linear datasets while being less susceptible to overfitting.

3.2.2 K-Nearest Neighbors

A non-parametric technique used for classification and regression applications is K-nearest neighbors (KNN). Finding a data point's K nearest neighbors and assigning it to the class with the most support from those neighbors is how it operates. In other words, KNN identifies the K closest data points in the training set to a given test point and then classifies the test point by the most common class among its K nearest neighbors [19]. This algorithm is often implemented using KD-Tree, a powerful data structure capable of constructing a tree with points of K dimensions.

KNN is a powerful algorithm for classification tasks, particularly when the decision boundary is complex, or the data is not linearly separable [20]. KNN is considered a lazy learner, as it does not work by making assumptions on the underlying data distribution and instead relies on the training data for classification [21]. Even though its simplicity and interpretability make it a popular choice, limitations of the algorithm such as high memory requirements, sensitivity to the scale of data etc., should be considered when applying the algorithm to real-world datasets.

3.2.3 Gaussian Naïve Bayes

Gaussian Naive Bayes (GNB) is a simple and effective algorithm used for classification tasks in machine learning. This algorithm uses Bayes' theorem assuming independent features following a Gaussian distribution. It has been used widely in several fields, including text classification, image recognition, and medical diagnosis [22].

GNB estimates the variance and mean for each class in the training data, during the training phase. During testing, the algorithm calculates the probability of the input data belonging to each class, using the conditional probabilities of each feature given the class and the prior probability of the class itself. The class with the highest probability is then selected as the predicted class [23]. GNB has been shown to perform well on large datasets with many features and is computationally efficient due to its simplifying assumptions [24]. However, the assumption of independence between features can be limiting in some cases where the features are correlated. In such cases, more advanced algorithms such as the multivariate Gaussian Naive Bayes or the Bayesian network can be used [23].

In conclusion, GNB is a popular and effective classification algorithm that is simple to implement and can achieve high accuracy in many applications. However, it may not be suitable for all datasets and applications, and more advanced algorithms may be necessary to address complex and correlated features [24]. For example, it assumes that the features are independent of each other, and it may not be suitable for data with complex relationships between the features and the target variable. Researchers and practitioners should carefully consider the assumptions and limitations of GNB when selecting a classification algorithm for their specific application.

3.2.4 Logistic Regression

Logistic Regression is a frequently used machine learning algorithm for solving binary classification problems. Logistic Regression is essentially a statistical model that employs a logistic function to predict the probability of a binary response variable based on one or more predictor variables [25]. Logistic Regression is popular in different fields such as healthcare, finance, and marketing, owing to its simplicity, effectiveness, and interpretability [26].

One of the strengths of Logistic Regression is its ability to handle both continuous and categorical predictor variables. Moreover, this algorithm provides interpretable coefficients, which can be used to understand the importance of each predictor variable in the model. This interpretability also enables domain experts to validate the model and make informed decisions based on the model outputs. [26] Another major advantage of using this algorithm, is that it can handle high-dimensional data and perform well in sparse datasets [27]. This makes Logistic Regression a suitable choice for problems with many features or when the dataset has missing values.

One way to handle multi-class classification tasks in logistic regression is by utilizing methods like one-vs-all and softmax regression, as described in [23]. Additionally, logistic regression can be augmented with other machine learning (ML) algorithms like decision trees and support vector machines to enhance the model's efficacy, as mentioned in [28]. Therefore, Logistic Regression can be used as a baseline algorithm for the PIMA Indians Diabetes Dataset, and its performance can be improved by combining it with feature selection and engineering techniques or other machine learning algorithms.

3.3 Proposed Method

An Artificial Neural Network (ANN) is a computational model that draws inspiration from the structure and function of biological neurons. Just like the human brain, an ANN consists of many connected nodes, or artificial neurons, that work together to process information. ANN's typically include one or more hidden layers, where the neurons process data and pass it on to the next layer. Each neuron has an activation function, which helps to determine the output of that neuron. The output from the artificial neurons can be used for classification or prediction tasks, among others.

A dense neural network is a particular kind of artificial neural network, in which all the neurons are connected from one layer to the next layer. Each connection between neurons has a corresponding weight, which determines the strength of the connection [29].

Each neuron in a neural network has an activation function, which is used to determine the importance of an output from the neuron. This paper uses the Parametric Rectified Linear Unit (PReLU) and Sigmoid activation function. PReLU is a modification of the Rectified Linear Unit (ReLU) activation which sets all negative values to zero and leaves positive values unchanged. PReLU introduces a parameter that allows it to adjust the slope of the function for negative inputs. This means that PReLU can learn the appropriate slope for negative inputs during training. The equation of PReLU is described in Equation 4, where α is the learnable parameter that determines the slope for negative inputs and x is the input feature.

$$f(x) = \begin{cases} x, & x > 0 \\ \alpha x, & x \leq 0 \end{cases}$$

(4)

A mathematical function called the sigmoid activation converts any input value to a number between 0 and 1 has been used in the neural network. Additionally, because it is a non-linear function, neural networks may simulate intricate connections between inputs and outputs. The function has been described in Equation 5, where x is the input feature.

$$f(x) = \frac{1}{1 + e^{-x}}$$

(5)

We propose utilizing a deep neural network for this prediction task, which comprises an input layer, two dense layers utilizing a PReLU activation function, and a final output layer utilizing a Sigmoid activation function. The input layer contains 8 neurons, the first dense layer contains 128 neurons, the second dense layer contains 64 neurons, and the output layer contains one neuron. The overall architecture of the neural network can be found in Figure 5. A flowchart of our proposed method has been presented in Figure 6.

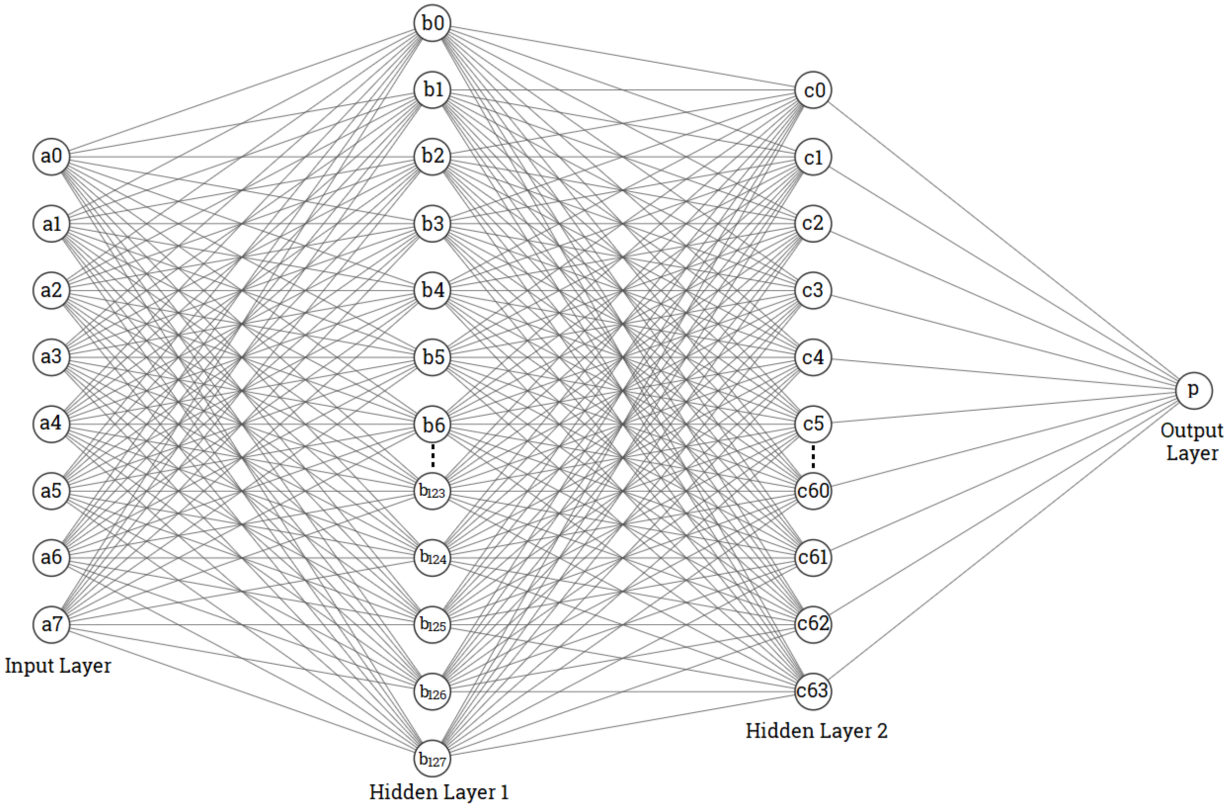

Figure 5: DNN Architecture

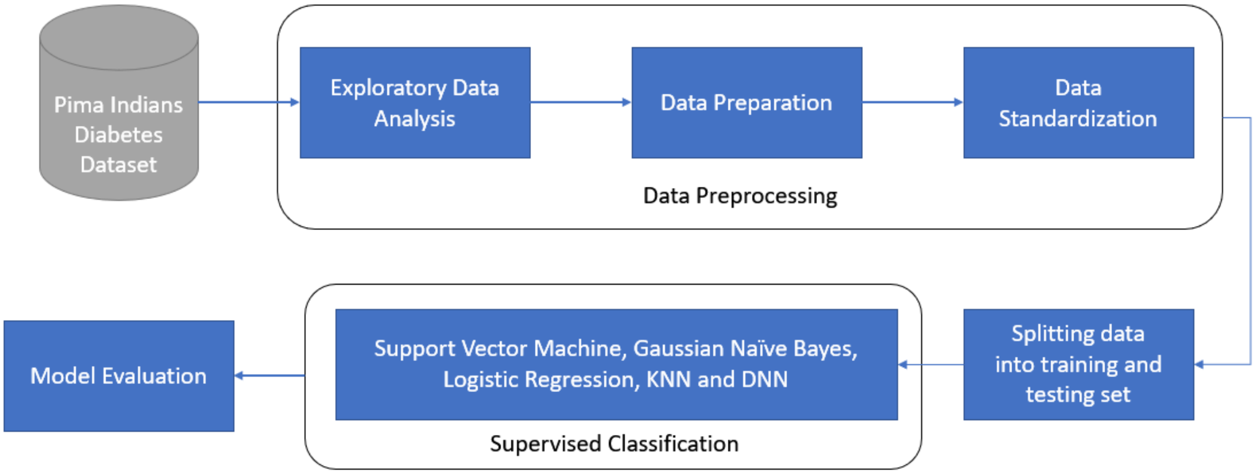
Pima Indians Diabetes Dataset
Exploratory Data Analysis
Data Preparation
Data Standardization
Data Preprocessing
Splitting data into training and testing set
Support Vector Machine, Gaussian Naïve Bayes, Logistic Regression, KNN and DNN
Supervised Classification
Model Evaluation

Figure 6: Flowchart of our Proposed Model

The proposed neural network has been trained with the Binary Crossentropy Loss Function, which is a model metric aimed at tracking the performance of the classification model. It penalizes the model if the predicted probability diverges from the actual label. The formula for the loss function is described by Equation 6, where y_i represents the actual class and $\log(\mathrm{P}(\mathrm{y_i}))$ is the probability of that class.

$$\mathrm{Loss} = -\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}} \mathrm{y_i} \cdot \log(\mathrm{P}(\mathrm{y_i})) + (1 - \mathrm{y_i}) \cdot \log(1 - \mathrm{P}(\mathrm{y_i}))$$

(6)

The DNN uses the Adam Optimizer for finding an optimal learning rate. Adaptive Moment Estimation (Adam) is a gradient-based optimization algorithm, which is an extension of the Stochastic Gradient Descent (SGD) optimizer. Adam uses estimates of the first and second estimates of the moments, to compute adaptive learning rates for each parameter. So, it allows the optimizer to adjust its learning rate optimally, for faster convergence and improved generalization performance. Adam is often the default choice for optimizing neural networks, due to its performance and ease of use.

3.4 Performance Evaluation

To evaluate different models, we will be using the Accuracy, Precision, Recall, and F1 metrics. Accuracy can be defined as the percentage of correct predictions done by the model. It can also be defined as the ratio between the sum of true positives and true negatives to the count of the entire dataset and has been described in Equation 7. Precision refers to the proportion of positive samples that are accurately classified out of all positive samples that were classified, as described in Equation 8.

On the other hand, recall is the ratio of correctly identified positive samples to the total number of actual positive samples in the dataset, as described in Equation 9. It evaluates the model's ability to detect all positive samples. The recall measures the model's ability to detect positive samples. The F1 score is the harmonic mean of precision and recall and has been described in Equation 10. It is a composite statistic that takes into account both recall and accuracy to provide a single number that summarizes the overall effectiveness of the model. Better performance is indicated by a higher F1 score, which has a maximum score of 1 and a minimum score of 0.

$$Accuracy\ (A) = \frac{\alpha + \beta}{\alpha + \beta + \gamma + \delta}$$

(7)

$$Precision\ (P) = \frac{\alpha}{\alpha + \beta}$$

(8)

$$Recall\ (R) = \frac{\alpha}{\alpha + \delta}$$

(9)

$$F1\ Score = \frac{2 * P * R}{P + R}$$

(10)

where α = True Positives, β = False Positives, γ = True Negatives and δ = False Negatives

4 Results

In this paper, we compared the performance of five classification algorithms (Gaussian Naïve Bayes, Logistic Regression, KNN, SVM and DNN) in predicting diabetes using the PIMA dataset. The dataset has been split into 70% training data and 30% testing data is used for model evaluation. It has been trained for 30 epochs, with a batch size of 32 and validation split of 30%.

Our results show that the DNN outperformed the other methods, achieving the highest accuracy of 81.72% and highest F1 Score of 0.6304. The second-best method was Logistic Regression, with an accuracy of 80.10% and the highest precision score of 0.7666. The SVM classifier achieved the highest recall (0.6037) and had an accuracy score of 72.58%. KNN achieved an accuracy score of 79.56%, while Gaussian Naïve Bayes had an accuracy of 77.95%. The results have been tabulated in Table 4 and a visualization is shown in Figure 7.

Method	Accuracy	Precision	Recall	F1 Score
Gaussian Naïve Bayes	0.7795	0.6200	0.5849	0.6019
Support Vector Machine	0.7258	0.5161	**0.6037**	0.5565
K-Nearest Neighbors	0.7956	0.7419	0.4339	0.5476
Logistic Regression	0.8010	**0.7666**	0.4339	0.5542
Deep Neural Network	**0.8172**	0.7435	0.5471	**0.6304**

Table 4: Performance Evaluation

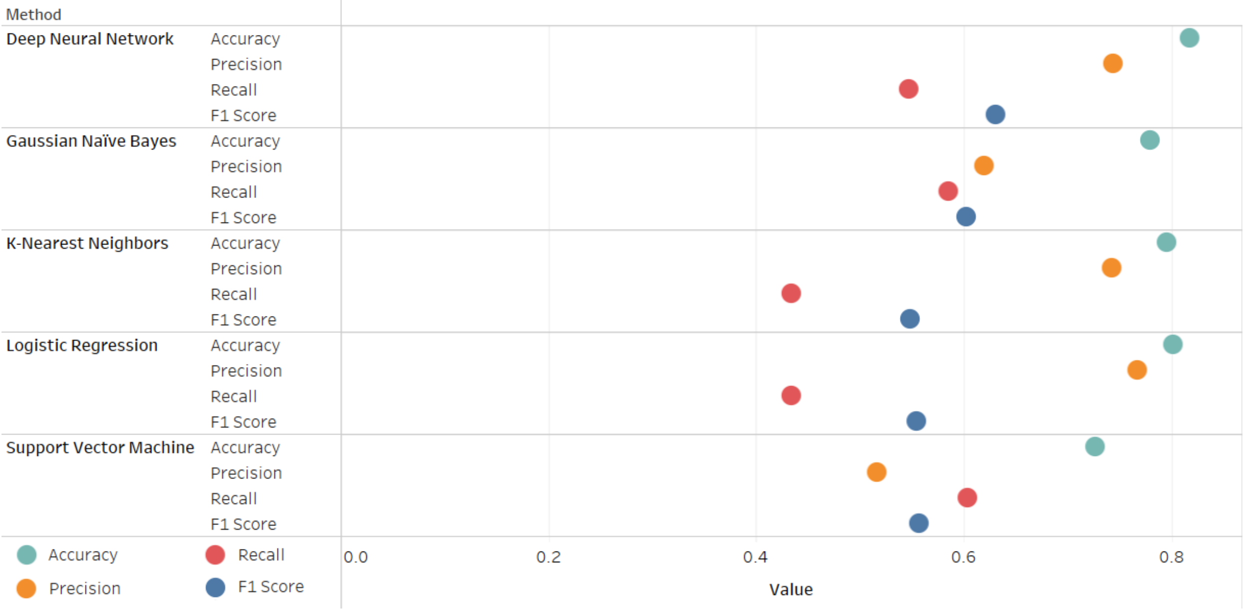

Figure 7: Performance Evaluation

The superior performance of DNN can be attributed to its ability to learn complex and nonlinear relationships between features and target variables. DNN can automatically extract high-level features from raw input data and use them to make accurate predictions. In contrast, the other methods we tested rely on linear relationships between features and target variables, which may not be sufficient for capturing the complexity of the PIMA dataset.

5 Conclusions

In conclusion, our study compares the performance of five classification algorithms in predicting diabetes using the PIMA dataset. Our findings suggest that DNNs outperform other algorithms in terms of accuracy and F1-score metrics. Specifically, the DNN achieved an accuracy of 81.72% and an F1-score of 0.6304, demonstrating its potential as a valuable tool for predicting diabetes in the PIMA Indian population.

While DNNs show promise in healthcare, it is crucial to address the ethical implications of using predictive models as well as ensure that these models do not perpetuate biases or discrimination. Our study contributes to the growing body of research on the application of ML techniques in healthcare and provides evidence for the potential of DNNs in diabetes prediction. However, further optimization of the DNN model through hyperparameter tuning and additional training is necessary to improve its accuracy. Additionally, future research could explore the use of DNNs in other healthcare domains and investigate the impact of such models on clinical decision-making and patient outcomes.

References

[1] J. B. Buse, "Diabetes mellitus," in Harrison's Principles of Internal Medicine, 20th ed., D. L. Longo, Ed. New York, NY, USA: McGraw-Hill Education, 2018, ch. 342, pp. 2395-2410.

[2] World Health Organization. (2021). Diabetes. Retrieved from https://www.who.int/news-room/fact-sheets/detail/diabetes [Accessed 27 Mar. 2023].

[3] Ewan R. Pearson; Dissecting the Etiology of Type 2 Diabetes in the Pima Indian Population. Diabetes 1 December 2015; 64 (12): 3993–3995. https://doi.org/10.2337/dbi15-0016

[4] Schulz LO, Chaudhari LS. High-Risk Populations: The Pimas of Arizona and Mexico. Curr Obes Rep. 2015 Mar;4(1):92-8. doi: 10.1007/s13679-014-0132-9.

[5] Dong, Y., Xie, M., Jiang, X., & Yuan, Y. (2020). Application of machine learning methods in diabetes prediction: A systematic review. Frontiers in Endocrinology, 11, 1-14. Available: https://doi.org/10.3389/fendo.2020.578605 [Accessed 27 Mar. 2023].

[6] UC Irvine Machine Learning Repository. (n.d.). Pima Indians Diabetes Database. [Online]. Available: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database [accessed 28 Mar. 2023].

[7] Deng, L. and Kasabov, N., "Feature selection and classification of Pima Indians diabetes dataset using self-organizing maps and cross-validation," International Journal of General Systems, vol. 43, no. 4, pp. 375-391, 2014.

[8] Yu, W., Liu, T., Valdez, R., Gao, J. and Zeng, X., "Quantum particle swarm optimization and weighted least squares support vector machine for Pima Indian diabetes classification," Computational and Mathematical Methods in Medicine, vol. 2013, Article ID 398781, 11 pages, 2013.

[9] Al Jarullah, A., Fakhrzadeh, M., and Asadi-Shekari, Z., "An empirical comparison of different decision tree algorithms for predicting diabetes mellitus," International Journal of Healthcare Information Systems and Informatics (IJHISI), vol. 10, no. 2, pp. 18-38, 2015.

[10] Lukka, P., "Fuzzy entropy measures-based feature selection for Pima Indians diabetes dataset classification," International Journal of Computer Applications, vol. 55, no. 10, pp. 38-42, 2012.

[11] Seera, M., Yoo, P.D. and Kim, D.H., "An intelligent system for the diagnosis of diabetes mellitus using FMM neural network and decision tree," Journal of Medical Systems, vol. 36, no. 5, pp. 3135-3144, 2012.

[12] Choubey, S., "PIMA Indian diabetes dataset classification using Naive Bayes and Genetic Algorithm based feature selection," International Journal of Computer Applications, vol. 95, no. 6, pp. 1-8, 2014.

[13] Kumari, A., Kumar, A. and Kumar, V., "Classification of Pima Indians diabetes data using support vector machines," International Journal of Computer Science Issues, vol. 10, no. 2, pp. 20-26, 2013.

[14] Somu, R., Lakshmanaprabu, S.K., and Kannan, A., "Enhanced prediction of diabetes using rough set-based K-Helly feature selection algorithm with random forest classifier," International Journal of Computer Applications, vol. 103, no. 4, pp. 19-24, 2014.

[15] Vapnik, V. N., & Chervonenkis, A. Ya. (1963). On the uniform convergence of relative frequencies of events to their probabilities. Theory of Probability & Its Applications, 8(2), 288-29

[16] Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. Proceedings of the fifth annual workshop on Computational learning theory, 144-152.

[17] Aizerman, M. A., Braverman, E. M., & Rozonoer, L. I. (1964). Theoretical foundations of the potential function method in pattern recognition learning. Automation and Remote Control, 25(6), 821-837.

[18] Mercer, J. (1909). Functions of positive and negative type, and their connection with the theory of integral equations. Philosophical Transactions of the Royal Society A, 209, 415-446. doi: 10.1098/rsta.1909.0013

[19] Alpaydin, E. (2010). Introduction to machine learning (2nd ed.). MIT Press.

[20] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). Springer.

[21] Bramer, M. (2015). Principles of data mining (3rd ed.). Springer.

[22] Rish, I. (2001). An empirical study of the naive Bayes classifier. IJCAI Workshop on Empirical Methods in Artificial Intelligence, 41-46.

[23] Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.

[24] Domingos, P., & Pazzani, M. (1997). On the optimality of the simple Bayesian classifier under zero-one loss. Machine Learning, 29(2-3), 103-130.

[25] Hosmer Jr, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). Applied logistic regression (Vol. 398). John Wiley & Sons.

[26] Agresti, A. (2002). Categorical data analysis (Vol. 482). John Wiley & Sons.

[27] Peduzzi, P., Concato, J., Kemper, E., Holford, T. R., & Feinstein, A. R. (1996). A simulation study of the number of events per variable in logistic regression analysis. Journal of clinical epidemiology, 49(12), 1373-1379.

[28] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction (2nd ed.). Springer.

[29] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.