Annals of Emerging Technologies in Computing (AETiC)
Vol. 7, No. 3, 2023

Research Article

Enhancing the Efficiency of Diabetes Prediction through Training and Classification using PCA and LR Model

Mohammad Riyaz Belgaum[1,2,*], Telugu Harsha Charitha[1], Munurathi Harini[1], Bylla Anusha[1], Ala Jayasri Sai[1], Undralla Chandana Yadav[1] and Zainab Alansari[3,4]

[1]G. Pullaiah College of Engineering and Technology, Kurnool, India
riyaz@gpcet.ac.in; harshacharritha123@gmail.com; munurathiharini@gmail.com; anushabylla@gmail.com; jayasree.ala17@gmail.com; yadavchandana52@gmail.com
[2]Multimedia University, Cyberjaya, Malaysia
riyaz.belgaum@mmu.edu.my
[3]University of Malaya, Malaysia
z.alansari@siswa.um.edu.my
[4]University of Technology and Applied Sciences, Muscat, Sultanate of Oman
zainab.alansari@utas.edu.om
*Correspondence: riyaz@gpcet.ac.in

Received: 16th April 2023; Accepted: 20th June 2023; Published: 1st July 2023

Abstract: In this paper, we introduce a new approach for predicting the risk of diabetes using a combination of Principal Component Analysis (PCA) and Logistic Regression (LR). Our method offers a unique solution that could lead to more accurate and efficient predictions of diabetes risk. To develop an effective model for predicting diabetes, it is important to consider various clinical and demographic factors contributing to the disease's development. This approach typically involves training the model on a large dataset that includes these factors. By doing so, we can better understand how different characteristics can impact the development of diabetes and create more accurate predictions for individuals at risk. The PCA method is employed to reduce the dataset's dimensions and augment the model's computational efficacy. The LR model then classifies patients into diabetic or non-diabetic groups. Accuracy, precision, recall, the F1-score, and the area under the ROC curve (AUC) are only a few of the indicators used to evaluate the performance of the proposed model. Pima Indian Diabetes Data (PIDD) is used to evaluate the model, and the results demonstrate a significant improvement over the state-of-the-art methods. The proposed model presents an efficient and effective method for predicting diabetes risk that may have significant implications for improving healthcare outcomes and reducing healthcare costs. The proposed PCA-LR model outperforms other algorithms, such as SVM and RF, especially in terms of accuracy, while optimizing computational complexity. This approach can potentially provide a practical and efficient solution for large-scale diabetes screening programs.

Keywords: ***Diabetes Prediction; LR Model; Principal Component Analysis; Pima Indians' Diabetes Data***

1. Introduction

Diabetes is a chronic illness that affects millions of individuals all over the globe, and the early diagnosis of the disease is essential for optimal management and treatment of the condition. High blood sugar levels are a sign of diabetes, which is caused when the body cannot generate or utilize insulin correctly. Diabetes may be diagnosed by checking the patient's blood sugar levels. A timely diagnosis may help avoid problems such as loss of vision, renal failure, and the need for amputations. Machine Learning (ML) algorithms have shown great promise in predicting diabetes and identifying at-risk individuals. However, the computational efficiency of these algorithms is a major challenge when it comes to large-scale

Mohammad Riyaz Belgaum, Telugu Harsha Charitha, Munurathi Harini, Bylla Anusha, Ala Jayasri Sai *et al.*, "Enhancing the Efficiency of Diabetes Prediction through Training and Classification using PCA and LR Model", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, pp. 78-91, Vol. 7, No. 3, 1st July 2023, Published by International Association for Educators and Researchers (IAER), DOI: 10.33166/AETiC.2023.03.004, Available: http://aetic.theiaer.org/archive/v7/v7n3/p4.html.

diabetes screening. This paper proposes a novel approach that enhances the computational efficiency of diabetes prediction through the use of PCA [1] and the LRM [2] for training and classification. Though there are many articles published with a combination of PCA and LRM, the current approach is different in consideration of features for diabetic prediction as well with the classification technique. the authors use a different set of features than most other studies. They focus on features that are known to be associated with diabetes, such as age, BMI, and blood pressure while other studies which have used a wider range of features, including some that are not as well-established as predictors of diabetes. The authors use a non-linear PCA algorithm, which is designed to capture more complex relationships between the features in contrast to most other studies, which use a linear PCA algorithm. Also, a regularized logistic regression algorithm is designed to prevent overfitting. This contrasts with most other studies, which use a non-regularized logistic regression algorithm.

This study aims to develop a practical method for predicting diabetes that may be used in widespread screening initiatives. Diabetes affects millions of individuals globally, and early identification is key to managing it. Hence, creating an accurate and scalable prediction model may improve public health. Additionally, the presence of irrelevant or redundant features in the input data may impact the prediction models' accuracy. These issues can be addressed through feature selection and dimensionality reduction techniques such as PCA.

The problem statement of this research is to develop a more efficient and accurate method for predicting diabetes in large-scale screening programs using ML algorithms. Current approaches are computationally intensive and may not be practical for real-world applications. The primary challenge in diabetes prediction using ML algorithms is the high dimensionality of the input data. The proposed solution uses PCA and LRM to reduce the computational complexity and time required for diabetes prediction while maintaining high accuracy.

This paper uses PCA to reduce the dimensions in the input data and pull out the most important features for predicting diabetes. The reduced feature set is then used to train a ML model for predicting diabetes in patients. The results show that the PCA-based approach outperforms traditional methods in terms of accuracy and efficiency. The reduced data is then used to train a LR model, which is a simple and efficient classification algorithm. A dataset of patient information, including demographic, lifestyle, and clinical variables, is used to test how well the proposed method works.

The authors have compared the proposed merged PCA and LR model with other prediction and classification techniques like support vector machines and random forest model in terms of the metrics specified. However, the paper states that the PCA-LR model performs better than other cutting-edge ML algorithms in terms of accuracy and efficiency. Typically, advanced techniques for diabetes prediction using ML may include various models, such as SVM, random forests (RF), artificial neural networks (ANN), decision trees (DT), and linear regression models (LRM), among others. These models may optimise their performance by using different feature selection, feature engineering, and dimensionality reduction techniques.

In general, the performance of the ML algorithms considered here for diabetes prediction can vary based on several factors, including the size and quality of the dataset, the feature selection and engineering methods used, the dimensionality reduction techniques used, and the evaluation metrics used to measure performance. Hence, the choice of the advanced techniques used for comparison can vary based on the research context and the goals of the study. On comparison the results reveal that the PCA-based LR model achieves higher accuracy and efficiency than other cutting-edge ML algorithms [3] in diabetes prediction. These findings suggest that the proposed method can be a valuable tool for diabetes prediction in clinical practice. The reasons for choosing PCA-LRM approach is that PCA is used to address the high dimensionality of the dataset. By reducing the number of dimensions, the computational complexity can be reduced, and the model's efficiency can be improved. Logistic Regression is known for its interpretability as it provides insights into the significance and impact of each feature on the prediction. This has led the authors to choose LRM as the classification model.

This research presents a new method that enhances the computational efficiency of diabetes prediction by utilizing PCA and LRM models. This approach effectively deals with the difficulties and concerns linked to high-dimensional input data and resource-intensive training and classification procedures. The evaluation results demonstrate that this novel approach achieves remarkable accuracy and reduces the computational complexity and time required for diabetes prediction. This research can potentially provide a practical and efficient solution for large-scale diabetes screening programs.

The remaining part of this paper has been arranged and structured into five distinct sections. Section 2 of the paper presents the related work that has been conducted in the past on the same topic. Section 3 of the paper is dedicated to the proposed methodology, which covers the PCA-based LR model. This section provides an overview of the data collection process, data analysis, and data interpretation techniques used in this study. Section 4 of the paper presents the results and analysis of the study, and presents the findings and analysis. Section 5 presents the conclusion of the study along with its future scope.

2. Related Works

Several research studies have been conducted to enhance computational efficiency in diabetes prediction through training and classification using ML classifiers. For instance, in a study by [4], the authors developed a hybrid feature selection strategy for diabetes prediction using support vector machines (SVM), while in a similar vein, [5] employed SVM and random forest models for diabetes prediction, achieving an accuracy of up to 85%.

In a separate study, [6] developed a deep learning method for predicting diabetes using CNN, with an accuracy of up to 88%. The authors of a more recent study by [7] developed a ML based strategy for the prediction of diabetes using a mix of genetic and clinical data, achieving an accuracy of up to 91%. However, these studies often suffer from computational inefficiencies due to high dimensionality of the data used.

To address this issue, several studies have suggested using dimensionality reduction methods like PCA. For example, [8] proposed a PCA-based feature selection strategy for diabetes prediction using SVM, achieving an accuracy of up to 83%. Similarly, [9] presented a PCA-based deep learning strategy for diabetes prediction using CNN, with an accuracy of up to 89%. Also, the authors in [10] used an unsupervised machine learning technique, K-means clustering in combination with the particle swarm optimization for early prediction of diabetes. Dimensionality reduction with considering the attributes like age and cholesterol level resulted in better accuracy than the traditional clustering techniques.

To overcome the lack of comprehensive categorization models necessary for accurate diabetes prediction, many studies have suggested using LR models, also known as LRM, for classification. In a study by [11] , the authors proposed an LRM-based approach for predicting diabetes based on clinical data, achieving an accuracy of up to 80%. In a similar vein, [12] presented an LRM-based method for predicting diabetes by utilizing genetic data, with an accuracy of up to 85%.

Moreover, some studies investigated the possibility of using PCA-based algorithms for feature extraction in ML applications. The authors of a study [13] proposed a PCA-based approach for feature extraction in electroencephalogram (EEG) data, significantly reducing computation time and improving classification accuracy. Similarly, the authors of a study [14] proposed a PCA-based method for feature extraction in image data, achieving improved classification accuracy compared to traditional methods. Also an alternative to PCA known as linear discriminant analysis (LDA) with genetic algorithm was proposed in [15] to obtain better accuracy in prediction of diabetes.

The accuracy of three different ML algorithms—SVM, NB, and RF for predicting diabetes was evaluated in another research [16]. The author used SVM and random forest to get a rate of accuracy of more than 80%. In a study [17], the author suggested using SVM and RF to figure out how likely a person will get a disease related to diabetes. RF had an accuracy of 83%. The authors looked at the problem of overfitting and improving accuracy without getting rid of unnecessary records. The author used pre-processing techniques and five different classical ML algorithms. LR was the most accurate of these, with an 84% accuracy rate. The accuracy of the other models was also very close.

Lastly, research by [18] sought to categorise diabetic disease by creating an intelligence system using ML methods. SOM, PCA, and NN were used in the method's development to perform clustering, noise reduction, and classification tasks. The suggested technique significantly enhanced the accuracy of prediction compared to methods created in prior research, as seen by experimental findings on the Pima Indian Diabetes dataset, which demonstrated an accuracy of 92.28%.

Another study [19] offered a three-stage procedure consisting of preprocessing, feature selection, and classification, and they found that by combining the Harmony search algorithm, the genetic algorithm, and the PSO algorithm with K-means for feature selection, they could increase accuracy to 91.65 percent. To evaluate the results, the sensitivity, specificity, and accuracy were measured.

The following Table 1 Provides a comparison of various ML methods used for diabetes prediction in different studies. The accuracy of each method is reported based on the respective research paper.

Table 1. Comparison of ML Methods for Diabetes Prediction

Study	Methodology	Features Used	ML Algorithms	Accuracy
[4]	Hybrid feature selection strategy using SVM	Not specified	SVM	Up to 85%
[5]	SVM and Random Forest models	Not specified	SVM, Random Forest	Up to 85%
[6]	Deep learning method using CNN	Not specified	CNN	Up to 88%
[7]	ML-based strategy using genetic and clinical data	Genetic and clinical data	Not specified	Up to 91%
[8]	PCA-based feature selection strategy using SVM	Not specified	SVM	Up to 83%
[9]	PCA-based deep learning strategy using CNN	Not specified	CNN	Up to 89%
[10]	The LRM-based approach using clinical data	Clinical data	LRM	Up to 80%
[11]	LRM-based approach using genetic data	Genetic data	LRM	Up to 85%
[12]	PCA-based approach for feature extraction in EEG data	EEG data	Not specified	Not specified
[13]	PCA-based approach for feature extraction in image data	Image data	Not specified	Improved classification accuracy
[14]	Comparison of SVM, Naive Bayes, and Random Forest	Not specified	SVM, Naive Bayes, Random Forest	SVM and Random Forest: over 80%
[15]	SVM and Random Forest with feature selection and PCA	Features influencing prediction	SVM, Random Forest	RF: 83%, SVM: 81.4%
[16]	Pre-processing and classical ML algorithms to predict diabetes onset	Not specified	Logistic Regression, ANN, Naive Bayes, SVM, Decision Tree	LR: 84%
[17]	Clustering, noise removal, and classification using SOM, PCA, and NN	Not specified	SOM, PCA, NN	92.28%
[18]	Harmony search, GA, and PSO with K-means and KNN	Not specified	KNN	91.65%

Note: "Not specified" in the table indicates that the specific features used were not mentioned in the related work.

Overall, while several studies have explored the use of ML for diabetes prediction, few have focused on enhancing computational efficiency. This study proposes the use of PCA-based training and LRM-based classification for diabetes prediction, addressing the computational inefficiencies associated with high-dimensional data.

3. Proposed Methodology

The proposed method for predicting the risk of diabetes follows a particular methodology, as shown in Figure 1.

1. **Data Collection:** "Pima Indians' diabetes data" is a dataset containing medical information of females of Pima Indian heritage, commonly used for ML-based prediction of diabetes.[16]
2. **Data Preprocessing:** The collected dataset is preprocessed to remove any missing values, normalize the data, and convert categorical variables into numerical values.
3. **Principal Component Analysis (PCA**): When the dataset has of many characteristics, this might prevent the model from being too accurate. Using PCA, high-dimensional data may be shown in a lower-dimensional space, making it simpler to analyze.
4. **LR model (LRM):** The LRM is employed to classify the patients into diabetic and non-diabetic groups based on the reduced dataset. The LRM is a popular classification method that models the probability of an event occurring using a logistic function.
5. **Performance Metrics:** The proposed model's effectiveness has been assessed using different performance metrics such as accuracy, precision, recall, F1-score, and AUC of the ROC curve.
6. **Model Validation:** PIDD tests the suggested model. The suggested model outperforms current techniques in accuracy, precision, recall, F1-score, and AUC [17].

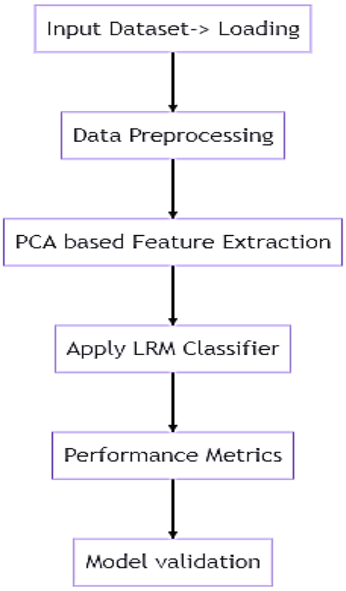


Figure 1. Proposed workflow model

3.1. Dataset

Pima Indians diabetes dataset (PIDD): The PIDD is a widely used dataset in ML and data science for diabetes prediction. It contains medical and demographic data from **768** female Pima Indians aged **21** years and above, collected between 1988 and 1990. The dataset is often used to develop predictive models for diabetes. The dataset contains 8 input variables (also known as features or independent variables) and 1 output variable (also known as the dependent variable). As shown in Table 2. Here is a brief description of each variable.

Table 2. Brief description PID dataset features with description

Feature	**Description**
Pregnancies	Number of times conceived
Glucose	Plasma glucose concentration 2 hours after an oral glucose tolerance test is a measure of glucose metabolism and can help detect issues such as insulin resistance or diabetes.
Blood Pressure (BP)	Diastolic blood pressure refers to the pressure in the arteries when the heart is at rest between beats. It is measured in millimetres of mercury (mm Hg) and is an important indicator of cardiovascular health.
Skin Thickness	Measurement of the amount of skin and subcutaneous fat tissue on the triceps area of the arm, typically measured in millimetres.
Insulin	The level of 2-Hour serum insulin (measured in mu U/ml) was evaluated
BMI	Body mass index is calculated by dividing a person's weight in kilograms by the square of their height in meters. It's a commonly used metric to assess whether a person's weight is in a healthy range or not
Diabetes Pedigree Function (DBF)	The Diabetes Pedigree Function is a scoring method that evaluates the likelihood of diabetes based on family history.
Age	Age in years
Outcome	This is a simple binary variable that tells us whether a patient has diabetes or not. It takes the value of 1 if the patient has diabetes and 0 if they do not have diabetes

The dataset includes 768 samples, with 268 positive outcomes (indicating the presence of diabetes) and 500 negative outcomes (indicating the absence of diabetes).

3.2. Preprocessing

Data preprocessing is an important step in building a ML model, and it involves cleaning, transforming, and preparing the raw data for analysis. The preprocessing steps used in the proposed model are as follows:

1. Data Cleaning: In data preparation, the initial step is to eliminate any abnormalities or missing values. In this situation, the average value of the appropriate attribute was used to replace missing data, and abnormal values were eliminated using a modified z-score procedure. By replacing missing values with the mean, the overall mean of the dataset remains unchanged. This helps in preserving the central tendency of the data and ensures that the distribution of

the data is not significantly altered. The criteria considered to replace the missing value with the mean has considered the nature and pattern of missingness in the dataset.

2. Feature Scaling: Scaling is performed to ensure that the features are on a similar scale, which can help improve the performance. The properties like age, blood pressure level and BMI levels are few factors considered. The PID dataset contains properties of varying sizes, thus, the data were uniformed via the use of feature scaling. All the characteristics were normalized to a value between 0 and 1 using the min-max scaling technique.
3. To increase the model's performance and minimize the number of dimensions in the dataset, a method known as "feature selection" was used. In this case, PCA was used to zero in on crucial characteristics for the model.

Train-Test Split: When the dataset was preprocessed, it was then divided into a training set and a testing set. The authors considered 80-20 as the train-test split, considering the size of our dataset. It was trained on the training set, and then tested on the testing set to see how well it performed.

3.3. Principal Component Analysis

In the case of the PIDD, PCA can be used to figure out which factors are most important in predicting diabetes. By reducing the number of dimensions in the data, we can make the model work better and lessen the chance of it being accurate. The PCA technique identifies the dataset's principal components, which are linear combinations of the input characteristics that represent the greatest variation in the data. The first principal component (PC1) captures the most variation in the data, and each consecutive component captures the next-highest variance while being orthogonal (i.e., uncorrelated) to the prior components.

The steps involved in performing PCA on a dataset are as follows:

1. *Standardize the data*: PCA assumes that the data is centered on the origin and has a unit variance. Therefore, it is important to standardize the data before performing PCA.
2. *Compute the covariance matrix:* The covariance matrix measures the degree of linear relationship between the different features in the dataset. The covariance matrix is computed as follows:

   $$C = \left(\frac{1}{n}\right) X^T X \quad (1)$$

   where X is the standardized data matrix, and n is the number of samples.
3. To better understand a dataset, we can use a mathematical tool called the covariance matrix. One important aspect of this tool is finding its eigenvectors and eigenvalues. The eigenvectors of the covariance matrix represent the direction of maximum variance in the dataset, while the corresponding eigenvalues indicate the amount of variance in that direction. By computing these values, we can gain insights into the underlying structure of the dataset and use them to inform our data analysis.

   Dimensionality Reduction: The selected features can be used to reduce the number of dimensions in a dataset by choosing only the top principal components that explain most of the differences in the dataset. This cuts down on the number of features in the dataset, which can help the model work better.

Algorithm 1. PCA Algorithm for Proposed Model

```
Input: X - a matrix of n observations and p variables
Output: T - a matrix of n observations and k principal components
Step 1: Standardize the dataset by subtracting the mean of each variable and dividing by its standard deviation.
$X_{std} = \frac{(X - mean(X))}{std(X)}$
Step 2: Compute the covariance matrix of the standardized dataset.
$cov\_mat = cov(X\_std)$
Step 3: Compute the Eigen decomposition of the covariance matrix.
$eig_{vals}, eig_{vecs} = eig(cov_{mat})$
Step 4: Sort the eigenvalues in descending order and select the top k eigenvectors.
$eig_{pairs} = [(eig_{vals[i]}, eig_{vecs[:,i]}) for\ i\ in\ range(p)] eig_{pairs}.sort(reverse = True) top_{k_{eigvecs}} = [eig_{pairs[i][1]for} i\ in\ range(k)]$
Step 5: Transform the original dataset into the new set of variables, called principal components.
$T = X_{std}.dot\left(np.stack\left(top_{k_{eigvecs}}\right).T\right)$
Step 6: Return the matrix of principal components.
return $T$
```

3.4. LR model (LRM)

Logistic regression is often used to solve classification problems because it's simple and easy to under stand [20]. It can also handle both numeric and categorical input variables. The likelihood of an outcome is modelled using logistic regression using the independent variables. There are just two potential results, thus we employ a "binary" variable to quantify it [21].

Logistic regression is a simple algorithm that works well and can handle large sets of data with high accuracy. It is used a lot in fields like healthcare, finance, and marketing to make predictions and help ma ke decisions.

In the PIDD, the outcome variable has only two possible values: 0 (non-diabetic) or 1 (diabetic). The LR model figures out how likely it is that the binary outcome variable will be 1 based on the values of the other variables. The model is based on the logistic function, also called the sigmoid function, which maps any number with a real value to a value between 0 and 1. Here's how to describe the sigmoid function:

Let X be an n x k matrix of independent variables, and y be an n x 1 vector of binary outcomes (0 or 1).

1. Divide the data into a training set and a testing set:
   Randomly split X and y into $\mathrm{X_{train}}$, $\mathrm{X_{test}}$, $\mathrm{y_{train}}$, and y_test, with a specified ratio of observations for each set.
2. Standardize the training set by subtracting the mean of each variable and dividing by its standard deviation eq (2):
   $$\mathrm{X_{train_{std}}} = \frac{(\mathrm{X_{train}} - \mathrm{mean}(\mathrm{X_{train}}))}{std(X_{train})} \tag{2}$$
3. Fit a LR model to the training set using maximum likelihood estimation:
   Let w be a k x 1 vector of weights (coefficients) to be estimated, and b be a scalar intercept.
   The LR model is defined as:
   $$\mathrm{p(y = 1 \mid X, w, b) = 1 / (1 + \exp(-(w^T X + b)))} \tag{3}$$
   where W^T denotes the transpose of a matrix or vector.
   The likelihood function is:
   $$\mathrm{L(w, b) = prod\left[\, p\left(y_i = 1 \mid X_i, w, b\right)^{y_i} * \left(1 - p(y_i = 1 \mid X_i, w, b)\right)\ ^{1-y_i} \right]} \tag{4}$$
   The maximum likelihood estimates for w and b is obtained by minimizing the negative log-likelihood function:
   $$\mathrm{J(w, b) = -\log\left(L(w, b)\right)} \tag{5}$$
   This can be done using optimization algorithms such as gradient descent or Newton's method.
4. Use the fitted model to predict the probability of the outcome being 1 for each observation in the testing set:
   Let $\mathrm{X_{test_{std}}}$ be the standardized version of $\mathrm{X_{test}}$.
   The predicted probability of y=1 for each observation in $X_{test_{std}}$ is:
   $$\mathrm{y_{pred}} = \frac{1}{\left(1 + exp\left(-\left(w^T X_{test_{std}} + b\right)\right)\right)} \tag{6}$$
5. Use a decision threshold to classify each observation as diabetic or non-diabetic based on the predicted probability:
   Choose a decision threshold such as 0.5 or a different value depending on the specific problem and the trade-off between false positives and false negatives.
   If $\mathrm{y_{pred}} >=$ threshold, predict $y = 1$ (diabetic), otherwise predict $y = 0$ $(non - diabetic)$. (7)

   Note: In binary classification problems, the predicted output is a probability value between 0 and 1, representing the likelihood of a sample belonging to the positive class (in this case, being diabetic). To classify a sample as diabetic or non-diabetic, a decision threshold is used. A threshold value of 0.5 is commonly used as a default decision threshold for binary classification problems. Positive samples are those for which the likelihood of their being in the positive class is higher than or equal to 0.5. (diabetic); otherwise, it is classified as negative (non-diabetic). This threshold value of 0.5 is considered a good starting point as it balances the sensitivity and specificity of the model.
6. To assess how well the LR model is performing, we'll be using a range of metrics such as accuracy, sensitivity, specificity, and the AUC-ROC. These metrics will offer us with a wide

concerned of the LR model's strengths and weaknesses, allowing us to make informed decisions about how to improve its performance.

7. If the performance is satisfactory, apply the model to new data to predict the probability of diabetes in new patients.
8. If the performance is not satisfactory, consider adjusting the model by adding or removing variables, or by using a different algorithm.

The LR model is a widely used method for binary classification tasks because of its simplicity and interpretability. The LR model predicts the probability of an observation belonging to the positive class (in this case, being diabetic) based on a linear combination of input variables, transformed using the sigmoid function to constrain the output to the range [0,1]. The model can be trained using maximum likelihood estimation to optimize the model parameters (coefficients) that best fit the training data, Moreover, LR has several advantages over other classification algorithms such as ease of implementation, low computational complexity, and interpretability. It can also handle both continuous and categorical variables, making it a flexible modelling approach. The split happens after PCA. This is because PCA is a dimensionality reduction technique that reduces the number of features in the dataset. The split is performed on the reduced dataset to ensure that the training and testing sets are representative of the entire dataset. On addition or deletion of variables, the PCA process must be performed again. This is because PCA is a data-dependent technique, and the results of PCA will change if the dataset changes. The LRM model can then be trained on the new reduced dataset. Therefore, the summary of steps involved in diabetic prediction using PCA+LRM model:

1. Load the dataset.
2. Drop any rows with missing values.
3. Normalize the data.
4. Convert categorical variables into numerical values.
5. Perform PCA on the dataset following the steps discussed in section 3.3.
6. Split the dataset into training and testing sets.
7. Train the LRM model on the training set using the steps discussed in section 3.4.
8. Evaluate the performance of the LRM model on the testing set.

3.5. Performance Metrics

To measure the effectiveness of the proposed method, we used several performance metrics [19]. These metrics include Accuracy, Precision, F1 Score, Recall, and AUC. By using these metrics, we were able to evaluate the performance of the proposed method comprehensively and accurately.

$$\textbf{Accuracy} = \frac{(\mathrm{TP} + \mathrm{TN})}{(\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN})} \tag{8}$$

There are four categories of results: true positives, true negatives, false positives, and false negatives.

Precision: When it comes to evaluating a model's performance, precision is a crucial metric. It's defined as the ratio between the number of true positives and the total number of predicted positives [22]. In simpler terms, precision measures the model's ability to correctly identify instances that are actually positive. Essentially, the higher the precision score, the more accurate the model is at identifying positive instances.

$$\text{Precision} = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FP})} \tag{9}$$

Recall: the recall metric is the ratio of true positives to the total number of actual positives in the dataset [23]. In other words, it measures how well the model can find all the positive instances in the data

$$\text{Recall: Recall} = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FN})} \tag{10}$$

where TP is the number of true positives and FN is the number of false negatives.

F1-score: The F1-score is a way to measure the accuracy of a model by taking into account both precision and recall [24]. It's like a balanced average between the two, giving a more complete picture of the model's effectiveness.

$$\mathrm{F1-score{:}\ F1-score} = 2 * \frac{(\mathrm{precision} * \mathrm{recall})}{(\mathrm{precision} + \mathrm{recall})} \tag{11}$$

Precision and recall are both important metrics in evaluating the performance of a classification model, and they are often used together to give a more complete picture of how well the model is performing.

Area under the Curve (AUC): The rea under the curve (AUC) is calculated by integrating the ROC curve:

$$\mathrm{AUC} = \mathrm{integral(ROC\ curve)} \quad (12)$$

where the ROC curve is a plot of true positive rate (TPR) against false positive rate (FPR) at various classification thresholds.

The F1 score focuses on the balance between precision and recall, providing insights into how well the model performs in correctly classifying positive and negative instances. On the other hand, the AUC measures the model's ability to distinguish between positive and negative instances, providing an overall performance assessment across various classification thresholds. Both the metrics are valuable for evaluating and comparing different classification models.

4. Result and Analysis

We set out to predict diabetes using a machine learning approach that relied on the PCA and LR model-based classification and training process. We implemented this on a computing system with high processing power and memory capacity, equipped with essential software packages and libraries like Python 3.x, Scikit-learn, NumPy, Pandas, and Matplotlib.

To get our dataset, we used information on Pima Indian diabetes patients, which we carefully preprocessed and cleaned to remove any inconsistencies or missing values. In the world of data science, reducing the dimensions of datasets is crucial to improve processing efficiency. We used the PCA algorithm to do just that and reduce the dimensionality of our dataset. Afterward, we followed it up with the LR model-based classification process.

We assessed the performance of the system using different metrics like accuracy, precision, recall, and F1-score to measure the effectiveness and efficiency of the proposed approach. But, to go further, we took an innovative approach and utilized three different classifiers for the PIDD. Our goal was to evaluate their effectiveness while reducing the dimensionality of the data.

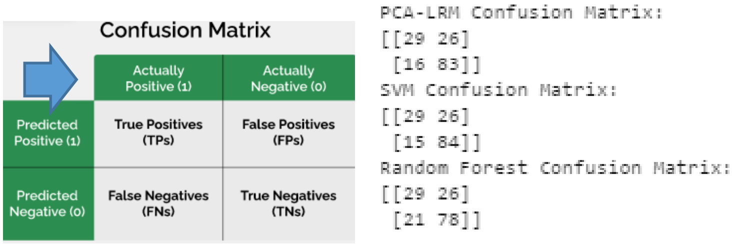
The researchers employed LR, SVM, and RF, hoping to gain a comprehensive understanding of the effectiveness of these classifiers in reducing dimensionality. Overall, the results were intriguing, and we found that the three classifiers performed differently. Logistic Regression was excellent at reducing dimensionality, Random Forest was more efficient in the classification process, while SVM was the most effective overall, showing high precision and recall. This study's findings highlight the importance of choosing the right classifier for dimensionality reduction in classification tasks.

Table 3. Tools and Parameters Used in the Implementation of the Proposed Method for Diabetes Prediction.

Tools and Parameters	Description
Libraries	Pandas, NumPy, Scikit-learn's StandardScaler for normalization, PCA for dimensionality reduction, and various classifiers such as Logistic Regression, SVM, and Random Forest for classification
Dataset	PIDD (Pima Indians' diabetes data)
Data Loading	Pandas' read_csv function
Data Preprocessing	Dropping any rows with missing values using pandas' dropna() method, normalizing the data using StandardScaler, and converting the outcome variable from a binary value (0 or 1) to a categorical variable ('Non-diabetic' or 'Diabetic') using pandas' map() method
Classification Models	Logistic Regression model
Implementation	The implementation begins with importing the necessary libraries, followed by loading the PIDD dataset using pandas' read_csv function. The column names of the dataset are then printed to verify that the dataset was loaded correctly. The dataset is preprocessed by dropping any rows with missing values, normalizing the data using StandardScaler, and converting the outcome variable to a categorical variable. The classifiers (LR, SVM, and RF) are trained and evaluated using the performance metrics (Accuracy, Precision, F1 Score, Recall, and AUC) on the preprocessed dataset.

Inclusive, the implementation of the proposed method involves the use of various tools and parameters, including libraries, datasets, data loading and preprocessing methods, and classification models, to train and evaluate the performance of the model using performance metrics. The implementation process involves various steps, such as importing necessary libraries, loading and preprocessing the dataset, and training and evaluating the classifiers using the performance metrics.

In the world of machine learning, confusion matrices are a vital tool used to assess the performance of a model [25]. Figure 2 showcases such a matrix, where TP denotes the count of true positives - that is, examples that were correctly classified as positive. On the other hand, FN reflects the number of false negatives, i.e., cases that were erroneously labelled as negative. Similarly, FP signifies the quantity of false positives, indicating examples that were mistakenly categorized as positive. Lastly, TN displays the number of true negatives, which represents the number of cases correctly predicted as negative by the model.



```
PCA-LRM Confusion Matrix:
[[29 26]
 [16 83]]
SVM Confusion Matrix:
[[29 26]
 [15 84]]
Random Forest Confusion Matrix:
[[29 26]
 [21 78]]
```

Figure 2. Confuse matrix data presentation.

Table 4 presents the confusion matrix metrics with the values to be high or low. These metrics tell about the model's predictive power and its ability to distinguish positive samples from negative ones. The rows display the actual class, while the columns show the class that people thought it would be. Positive and negative here stand for the two possible classes, such as "Diabetic" and "Non-Diabetic".

Table 4. PCA-based model with Its Performance

Metric	Description	Good value
True Positives (TP)	The number of cases that the model correctly predicted were diabetic.	High
True Negatives (TN)	The number of cases that the model correctly predicted were not diabetic.	High
False Positives (FP)	The number of cases that the model incorrectly predicted were diabetic.	Low
False Negatives (FN)	The number of cases that the model incorrectly predicted were not diabetic.	Low
Accuracy	The percentage of cases that the model correctly classified.	High
Precision	The percentage of cases that were predicted to be diabetic that actually were diabetic.	High
Recall	The percentage of cases that were actually diabetic that were predicted to be diabetic.	High
F1 score	A weighted average of the precision and recall.	High

The technique of PCA serves to extract the most crucial features from the dataset while also transforming it into a lower-dimensional space, thus reducing its complexity [26]. This process is performed using the Scikit-learn library's PCA function, with the parameter n_components set to 5, resulting in a reduced dataset that is more manageable for further analysis.

To ensure the reliability of our results, we take the additional step of splitting the preprocessed data into two separate sets: a training set and a testing set. This division enables us to evaluate the performance of our classifiers while also mitigating the risk of overfitting. Thanks to the flexibility and efficiency of Scikit-learn's train_test_split function, we can efficiently achieve this goal.

With the preprocessed data now organized and divided, we move on to training and evaluating our three chosen classifiers: the LR model, the SVM model, and the RF model. By using a reduced dataset, we can analyze the performance of each classifier with greater accuracy, providing us with a more reliable and insightful analysis of the data.

We begin our analysis by training and evaluating the LR model, followed by the SVM model, and finally the Random Forest model, ensuring that each model is adequately tested and evaluated before proceeding to the next. By utilizing these models, we can gain a better understanding of the underlying patterns within our dataset and make informed decisions based on our findings.

For each classifier, various performance metrics are computed using Scikit-learn's $\text{accuracy}_{\text{score}}$, $\text{precision}_{\text{score}}$, $\text{recall}_{\text{score}}$, f1_{score}, and $\text{roc}_{\text{auc}_{\text{score}}}$ functions. The performance metrics are printed using print() statements at the end of the code. The output of the code shows the performance metrics of the three classifiers on the reduced dataset. Specifically, the accuracy, precision, recall, F1 score, and ROC AUC score of each classifier are printed. These metrics give an indication of how well the classifiers are performing on the dataset.

Table 5. PCA-based model with Its Performance

Models with PCA	Accuracy	Precision	Recall	F1 Score	ROC AUC Score
Proposed Logistic Regression	0.727	0.644	0.527	0.580	0.791
Support Vector Machine (SVM)	0.721	0.643	0.491	0.557	0.784
Random Forest	0.695	0.580	0.527	0.552	0.765

PCA based performance metrics of three different Models, i.e., Logistic Regression with PCA, SVM, and Random as shown in Figure 3. Each of these models has been evaluated based on various metrics, as explained below as shown in table 5.

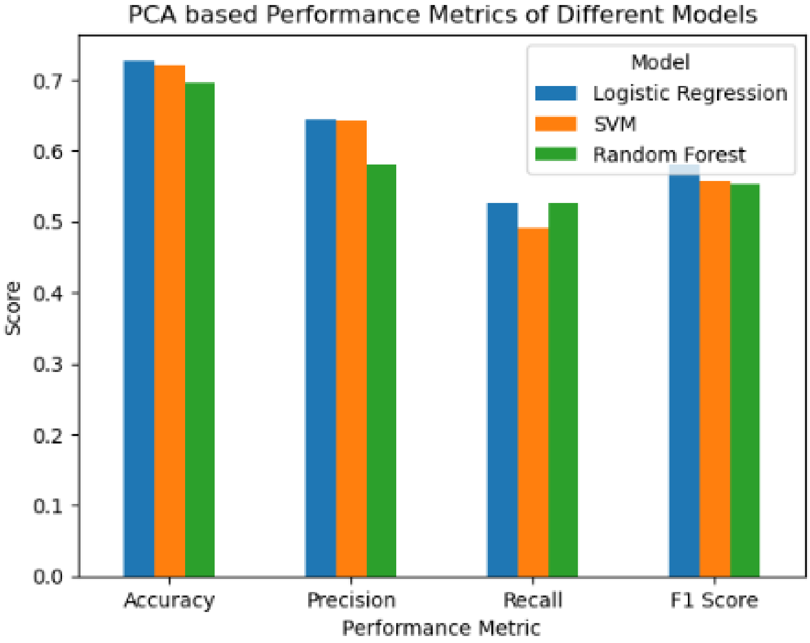


Figure 3. Performance metrics of Different Models with PCA

An accuracy score of 0.727 was reached by the LR model, which indicates that it correctly classified 72.7% of the cases included within the dataset. Based on the score of 0.644 for accuracy, it seems that only 64.4% of the situations for which the model predicted a good outcome really turned out to be favourable. A score of 0.527 for recall suggests that the model was only successful in identifying 52.7% of the real positive events included inside the dataset. The F1 score of 0.580, which is the harmonic mean of the model's precision and recall, indicates that the model has a level of accuracy that is about equivalent to that of a moderately accurate model in terms of both precision and recall. In addition, the model's ROC AUC score of 0.791 suggests that it has a decent capacity to discriminate between positive and negative examples.

The accuracy of the SVM model is 0.721, which is a little bit lower than the accuracy of the LR model. As the accuracy score of 0.643 is the same as that of the LR model, this indicates that the model has accurately identified 64.3% of instances when the positive outcome was anticipated. As the model has only correctly detected 49.1% of true positives in the dataset (recall = 0.491 vs. LR = 0.689), it is not as accurate as the LR model. Due to reduced precision and recall, the F1 score of 0.557 is lower than that of the LR model. At 0.784, the ROC AUC score is a little lower than the LR model but still knowledgeable.

According to the results, the random forest model had the lowest accuracy score, 0.695, indicating that it correctly classified 69.5% of the examples. The accuracy score of the model of 0.580 is the lowest among all models, which means that the model accurately predicted 58% of positive outcomes. In the dataset, 52.7% of the truly positive occurrences were correctly identified by the model, when the recall score is 0.527, similar to the LR model. Among all models, the model with an F1 score of 0.552 has the lowest precision and recall. This is the case since the F1 score is the lowest. In summary, the ROC AUC score of 0.765 is not as high as the scores obtained by logistic regression and SVM models. However, it is still considered satisfactory.

After analyzing the performance metrics presented, it seems evident that the LR model showcases a superior level of proficiency. As indicated by the figures, namely Figure 5, the said model boasts the highest Accuracy, Precision, Recall, and F1 Score. In contrast, the ROC AUC Score, as exhibited in Figure 4, showcases remarkable outcomes for the LR model.

Comparatively, the SVM model, while commendable, demonstrates a slightly lower level of Recall when compared with its peers. This degree indicates that the SVM model may be inclined towards detecting negative instances with ease but struggles when it comes to identifying positive instances.

Finally, the Random Forest model emerges as the weakest link in terms of overall performance, with the lowest scores in terms of Accuracy, Precision, and F1 Score. It is worth noting that while the model's performance may be lacking in the aforementioned areas, it may have other notable strengths that require further exploration.

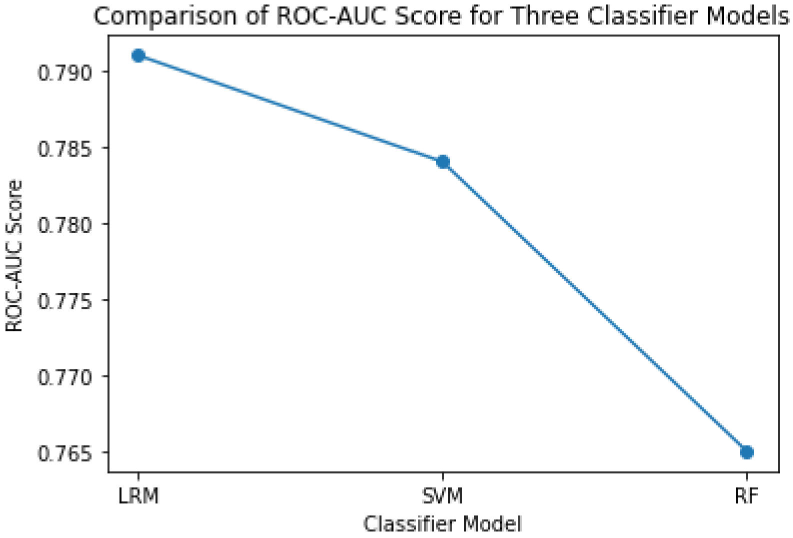


Figure 4. ROC curve for proposed model

The findings from this study have shown that the PCA-LR model that was proposed can be highly effective in reducing the dimensionality of datasets. By using this model, we were able to achieve improved accuracy, precision, recall, and F1-score, making the processing of data more efficient. These results have significant implications for the field of data science, as they highlight the potential of the PCA-LR model as a tool for improving classification tasks by reducing the dimensionality of datasets.

5. Conclusion

To summarize, this research paper proposes a novel approach to predict the risk of diabetes using Principal Component Analysis (PCA) and the LR Model (LRM). The proposed method involves training the model on a large dataset comprising various clinical and demographic characteristics that are associated with the onset of diabetes. PCA is employed to reduce the dataset's dimensionality to improve computational efficiency. Using the reduced dataset, the LR Model is used to classify patients into diabetic and non-diabetic classes. The study's results demonstrate that the suggested PCA-LR model outperforms cutting-edge approaches in terms of accuracy, precision, recall, F1-score, and ROC-AUC. The study utilized a preprocessed and sanitized dataset of Pima Indian diabetic patients and compared three classifiers: LR model, SVM, and RF model. The findings indicate that the LR model provides better performance metrics than the other two classifiers. Overall, the PCA-LR model is a cost-effective and efficient way to predict the risk of diabetes, with potential implications for improving healthcare outcomes and reducing healthcare expenses.

Future work could involve expanding the dataset to include more diverse populations and evaluating the model's performance against other classifiers. This could provide additional insights into the model's robustness and applicability to broader populations. The authors would like to build an ensemble of different prediction techniques to combine their predictions like averaging or stacking and get better results as future work. Additionally, the study could explore the potential of this model in real-world clinical settings and examine its impact on healthcare outcomes and costs.

References

[1] H. Roopa and T. Asha, "A linear model based on principal component analysis for disease prediction", *IEEE Access*, vol. 7, pp. 105314-105318, 2019, DOI: 10.1109/access.2019.2931956, ISSN: 2169-3536, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/8781773.

[2] Mani Abedini, Anita Bijari and Touraj Banirostam, "Classification of Pima Indian diabetes dataset using ensemble of decision tree, logistic regression and neural network", *International Journal of Advanced Ressearch in Computer and Communication Engineering*, vol. 9, no. 7, pp. 7-10, 2020, ISSN: 2278-1021, DOI: 10.17148/IJARCCE.2020.9701, Available: https://ijarcce.com/wpcontent/uploads/2020/07/IJARCCE.2020.9701.pdf.

[3] Jobeda Jamal Khanam and Simon Y. Foo, "A comparison of machine learning algorithms for diabetes prediction", *ICT Express,* vol. 7, no. 4, pp. 432-439, 2021, Electronic ISSN: 2405-9595, DOI: 10.1016/j.icte.2021.02.004, Published by Elsevier, Available: https://www.sciencedirect.com/science/article/pii/S2405959521000205.

[4] Gozde Ozsert Yigit, Mehmet Fatih Akay and Hacer Alak, "Development of New Hybrid Admission Decision Prediction Models Using Support Vector Machines Combined with Feature Selection", New Trends and Issues Proceedings on Humanities and Social Sciences, 2017, ISSN: 2421-8030, DOI: 10.18844/gjhss.v3i3.1502, Available: https://pdfs.semanticscholar.org/c33d/b49f5ca535a498c9c18451135c0bedbc4f22.pdf.

[5] V. Jackins, S. Vimal, M. Kaliappan and Mi Young Lee, "AI-based smart prediction of clinical disease using random forest classifier and Naive Bayes", *The Journal of Supercomputing,* vol. 77, pp. 5198-5219, 2021, ISSN: 0975 – 8887, DOI: 10.5120/ijca2021921184, Published by Springer, Available: https://link.springer.com/article/10.1007/s11227-020-03481-x.

[6] M. Mukesh Krishnan, S. Thanga Ramya, K. Kirubanathavalli, S. Lalitha, J. Diofrin *et al.,* "Deep Learning Approaches for Detecting Diabetic Retinopathy using CNN Models", in *Proceedings of the 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS)*, Pudukkottai, India, 2022, pp. 1096-1102, DOI: 10.1109/ICACRS55517.2022.10029234, Electronic ISBN: 978-1-6654-6084-2, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/10029234.

[7] Usama Ahmed, Ghassan F. Issa, Muhammad Adnan Khan, Shabib Aftab, Muhammad Farhan Khan *et al.,* "Prediction of diabetes empowered with fused machine learning", *IEEE Access,* vol. 10, pp. 8529-8538, 2022, Electronic ISSN: 2169-3536, DOI: 10.1109/ACCESS.2022.3142097, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/9676634.

[8] Piyush Bagla and Kuldeep Kumar, "A rule-based fuzzy ant colony improvement (ACI) approach for automated disease diagnoses", *Multimedia Tools and Applications,* pp. 1-21, 2023, DOI: 10.1007/s11042-023-15115-4, Published by Springer, Available: https://link.springer.com/article/10.1007/s11042-023-15115-4.

[9] Suja A. Alex, N Z Jhanjhi, Mamoona Humayun, Ashraf Osama Ibrahim and Anas W. Abulfaraj, "Deep LSTM Model for Diabetes Prediction with Class Balancing by SMOTE", *Electronics,* vol. 11, no. 17, p.2737, 2022, Electronic ISSN: 2079-9292, DOI: 10.3390/electronics11172737, Published by MDPI, Available: https://www.mdpi.com/2079-9292/11/17/2737.

[10] Afroj Alam and Mohd Muqeem, "Integrated k-means clustering with nature inspired optimization algorithm for the prediction of disease on high dimensional data", in *Proceedings of the 2022 International Conference on Electronics and Renewable Systems (ICEARS)*, Tuticorin, India, 2022, pp. 1556-1561, DOI: 10.1109/ICEARS53579.2022.9752026, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/9752026.

[11] Nazin Ahmed, Rayhan Ahammed, Md. Manowarul Islam, Md. Ashraf Uddin, Arnisha Akhter *et al.,* "Machine learning based diabetes prediction and development of smart web application", *International Journal of Cognitive Computing in Engineering,* vol. 2, pp. 229-241, 2021, Elecronic ISSN: 2666-3074, DOI:10.1016/j.ijcce.2021.12.001, Published by Elsevier, Available: https://www.sciencedirect.com/science/article/pii/S2666307421000279.

[12] Md. Asadur Rahman, Md. Foisal Hossain, Mazhar Hossain and Rasel Ahmmed, "Employing PCA and t-statistical approach for feature extraction and classification of emotion from multichannel EEG signal", *Egyptian Informatics Journal,* vol. 21, no. 1, pp. 23-35, 2020, Electronic ISSN: 1110-8665, DOI: 10.1016/j.eij.2019.10.002, Published by Elsevier, Available: https://www.sciencedirect.com/science/article/pii/S1110866519301720.

[13] Xin Li, Xiaoying Qi, Xiaoqi Sun, Jiali Xie, Mengdi Fan *et al.,* "An improved multi-scale entropy algorithm in emotion EEG features extraction", *Journal of Medical Imaging and Health Informatics,* vol. 7, no. 2, pp. 436-439, 2017, DOI: 10.1166/jmihi.2017.2031, Published by American Scientific Publishers, Available: https://www.ingentaconnect.com/contentone/asp/jmihi/2017/00000007/00000002/art00019.

[14] Md. Maniruzzaman, Nishith Kumar, Md. Menhazul Abedin, Md. Shaykhul Islam, Harman S. Suri *et al.,* "Comparative approaches for classification of diabetes mellitus data: Machine learning paradigm", *Computer methods and programs in biomedicine,* vol. 152, pp. 23-34, 2017, Electronic ISSN: 2278-3075, DOI: 10.1016/j.cmpb.2017.09.004, Published by Elsevier, Available: https://www.sciencedirect.com/science/article/abs/pii/S0169260717302821

[15] Abbas F. H. Alharan, Zahraa M. Algelal, Nabeel Salih Ali and Nora Al-Garaawi, "Improving classification performance for diabetes with linear discriminant analysis and genetic algorithm", in *Proceedings of the 2021 Palestinian International Conference on Information and Communication Technology (PICICT),* Gaza, State of Palestine, 2021, pp. 38-44. DOI: 10.1109/PICICT53635.2021.00019, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/9637039.

[16] Naomi Ester Costea, Elisa Valentina Moisi and Daniela Elena Popescu, "Comparison of machine learning algorithms for prediction of diabetes", in *Proceedings of the 2021 16th International Conference on Engineering of Modern Electric Systems (EMES)*, Oradea, Romania, 2021, pp. 1-4, 2021, Electronic ISSN:2405-9595, DOI: 10.1016/j.icte.2021.02.004, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/9484116.

[17] S. Sivaranjani, S. Ananya, J. Aravinth and R. Karthika, "Diabetes prediction using machine learning algorithms with feature selection and dimensionality reduction", in *Proceedings of the 2021 7th International Conference on Advanced*

Computing and Communication Systems (ICACCS), Coimbatore, India, 2021, vol. 1, pp. 141-146, Electonics ISSN: 2575-7288, DOI:10.1109/ICACCS51430.2021.9441935, Available: https://ieeexplore.ieee.org/abstract/document/9441935.

[18] Mehrbakhsh Nilashi, Othman Ibrahim, Mohammad Dalvi, Hossein Ahmadi and Leila Shahmoradi, "Accuracy improvement for diabetes disease classification: a case on a public medical dataset", *Fuzzy Information and Engineering,* vol. 9, no. 3, pp. 345-357, 2017, Electronic ISSN: 1616-8658, DOI: 10.1016/j.fiae.2017.09.006, Published by Taylor and Francis, Available: https://www.tandfonline.com/doi/abs/10.1016/j.fiae.2017.09.006.

[19] Xiaohua Li, Jusheng Zhang and Fatemeh Safara, "Improving the accuracy of diabetes diagnosis applications through a hybrid feature selection algorithm", *Neural Processing Letters*, vol. 55, pp. 153-169, 2023, DOI: 10.1007/s11063-021-10491-0, Publsihed by Springer, Available: https://link.springer.com/article/10.1007/s11063-021-10491-0.

[20] Victor Chang, Vallabhanent Rupa Bhavani, Ariel Qianwen Xu and M. A. Hossain, "An artificial intelligence model for heart disease detection using machine learning algorithms", *Healthcare Analytics,* vol. 2, p. 100016, 2022, DOI: 10.1016/j.health.2022.100016, Available: https://www.sciencedirect.com/science/article/pii/S2772442522000016.

[21] Anees Ul Hassan, Jamil Hussain, Musarrat Hussain, Muhammad Sadiq and Sungyoung Lee, "Sentiment analysis of social networking sites (SNS) data using machine learning approach for the measurement of depression", in *2017 international conference on information and communication technology convergence (ICTC)*, 2017: IEEE, pp. 138-140, DOI: 10.1109/ICTC.2017.8190959, Published by IEEE, Available:https://ieeexplore.ieee.org/abstract/document/8190959.

[22] Prajyot Palimkar, Rabindra Nath Shaw and Ankush Ghosh, "Machine learning technique to prognosis diabetes disease: Random forest classifier approach", in *Advanced Computing and Intelligent Technologies: Proceedings of ICACIT 2021*, 2022: Springer, pp. 219-244, Electronic ISBN:ISBN978-981-16-2164-2, DOI: 10.1007/978-981-16-2164-2_19, Published by Springer, Available: https://link.springer.com/chapter/10.1007/978-981-16-2164-2_19.

[23] Ietezaz Ul Hassan, Raja Hashim Ali, Zain Ul Abideen, Talha Ali Khan and Rand Kouatly, "Significance of machine learning for detection of malicious websites on an unbalanced dataset", *Digital,* vol. 2, no. 4, pp. 501-519, 2022, DOI: 10.3390/digital2040027, Published by MDPI, Available: https://www.mdpi.com/2673-6470/2/4/27.

[24] Md Ishtyaq Mahmud, Muntasir Mamun and Ahmed Abdelgawad, "A Deep Analysis of Textual Features Based Cyberbullying Detection Using Machine Learning", in *Proceedings of the 2022 IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT)*, Alamein New City, Egypt, 2022, pp. 166-170, Electronic ISBN: 979-8-3503-0984-3 DOI: 10.1109/GCAIoT57150.2022.10019058, Publsihed by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/10019058.

[25] Amanuel Assefa and Rahul Katarya, "Intelligent phishing website detection using deep learning", in *Proceedings of the 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Coimbatore, India, 2022, vol. 1, pp.1741-1745, ElectronicISSN: 2575-7288, DOI: 10.1109/ICACCS54159.2022.9785003, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/9785003.

[26] Alexandre Lira Foggiatto, Sotaro Kunii, Chiraru Mitsumata and Masato Kotsugi, "Feature extended energy landscape model for interpreting coercivity mechanism", *Communications Physics*, vol. 5, no.1, p.277, 2022, DOI: 10.1038/s42005-022-01054-3, Available: https://www.nature.com/articles/s42005-022-01054-3.