Turkish Journal of Computer and Mathematics Education *Vol.12 No.14 (2021), 1244- 1259*
Research Article

A Comparison of Five Machine Learning Algorithms in the Classification of Diabetes Dataset

Rasha Mahdi Abdulkader[1], Assist.Prof. Dr.Abdulbasit K. Alazzawi[2]

[1]Diyala University / College of Basic Education / Baqubeh / Diyala / Iraq
[2]Diyala University / College of Science /Baqubeh/Diyala/ Iraq

Article History: *Do not touch during review process(xxxx)*

Abstract: Diabetes is a disease that has no permanent cure; hence early detection is required with high accuracy. This study aims to compare five machine learning (ML)algorithms and achieve the best accuracy for predicting early stage diabetes. The dataset from the hospital Frankfurt, Germany includes information on 2000 patients as well as nine distinct characteristics for each of them is used in this work. Five ML Algorithms used for datasets to predict diabetes are Random Forest (RF), K-Nearest Neighbor (KNN), Gaussian Naïve Bayes (NB), support vector machine (SVM), and Logistic Regression (LR). However, according to the obtained results, it is observed that the proposed model with RF has achieved an excellent result of accuracy value = 99% during the comparison with a rest classification algorithm that is used in the proposed model. In addition, the proposed model's efficiency has been compared to previous work, and it has achieved the highest accuracy.

Keywords: Gaussian Naïve Bayes (NB); Machin Learning ML; Random Forest (RF); K-Nearest Neighbor (KNN); support vector machine (SVM); Logistic Regression (LR).

Introduction

Diabetes mellitus: More commonly referred to as “diabetes”—a chronic disease associated with abnormally high levels of the sugar glucose in the blood. Diabetes is due to one of the two mechanisms: Inadequate production of insulin (which is made by the pancreas and lowers blood glucose), or Inadequate sensitivity of cells to the action of insulin. Diabetes mellitus also may develop as a secondary condition linked to another disease, such as pancreatic disease, a genetic syndrome, such as myotonic dystrophy, or drugs, such as glucocorticoids. Gestational diabetes is a temporary condition associated with pregnancy. In this situation, blood glucose levels increase during pregnancy but usually returns to normal after delivery. Based on the data from the 2011 National Diabetes Fact Sheet, diabetes affects an estimate of 25.8 million people in the US, which is about 8.3% of the population. Additionally, approximately 79 million people have been diagnosed with pre-diabetes[1]. In this big data era, a large volume of data is generated and machine learning has become an imperative tool to analyze the complexity of the generated data. A plethora of techniques have been applied for data analytic in medical diagnosis, including single classifier and classifier ensemble [2]. With ML models, it can also be possible to improve quality of medical data, reduce fluctuations in patient rates, and save in medical costs. Therefore, these models are frequently used to investigate diagnostic analysis when compared with other conventional methods. To reduce the death rates caused by chronic diseases (CDs), early detection and effective treatments are the only solution. Therefore, most medical scientists are attracted to the new technologies of predictive models in disease forecasting [3]. Applying machine learning and data mining methods in Diabetes Mellitus (DM) research is a key approach to utilizing large volumes of available diabetes-related data for extracting knowledge. The severe social impact of the specific disease renders DM one of the main priorities in medical science research, which inevitably generates huge amounts of data [4]. Data mining represents a significant advance in the type of analytical tools. It has been proven that the benefits of introducing data mining into medical analysis are to increase diagnostic accuracy, to reduce costs and to save human resources [5]. The motive of this study is to compare the performance of the most effective machine learning techniques, used to predict diabetes diseases. In this works will be using diabetes dataset that collected from The dataset from the hospital Frankfurt. The dataset contains information about 2000 patients and their corresponding nine unique attributes. Algorithms used for datasets to predict diabetes are Naïve Bayes (NB) , Random Forest (RF) , K-Near Neighbor Classification algorithms (KNN),support vector machine (SVM),and Logistic Regression (LR). However, according to the obtained results it is observed that the proposed model with RF is achieved an excellent result of accuracy during the comparison with a rest classification algorithm that using in the proposed system.

The rest of the paper is laid out as follows: The second section delves into the related works. The discretion Diabetes Dataset and five machine learning models that are used in the proposed model are presented in Section

three along with the relevant methods and materials of this work. Section four illustrates the design proposed model. Sections five explain the simulation proposed method, the evaluation metrics, and experimental results and discussion respectively. Finally, Section six concludes the proposed model.

Related Works

In the previous work of Miao L., et. al.[6] develop a model for long-term risk cardiovascular disease (CVD) as a result of type 2 diabetes (T2D). On data from the Framing-ham Heart Research longitudinal study, they used the K-nearest neighbors and Support Vector Machine (SVM) algorithms to construct the prediction models. The dataset was first aligned using the synthetic Minority Oversampling Technique algorithm. After adjusting the parameters and training 1000 times, the average precision for correctly predicting the prevalence of CVD due to T2D was 96.5 %, with an average recall rate of 89.8 %. They also used the KNN algorithm to train the dataset, with a 92.9 % recall rate.

In the bygone work of Cherradi B., et. al. [7] to predict with or without type 2 diabetes mellitus patients, researchers used and evaluated four Machine Learning algorithms (Artificial Neural Network, K-Nearest Neighbors, Decision Tree, and Deep Neural Network). The first was retrieved from the Germany Frankfurt Hospital while the second is a well-known dataset of Pima Indian, which contains the same feature composed of risk factors, mixed data, and some clinical data. The proposed model achieved the best accuracy rate with KNN 97.53% and DeepNN 96.35%.

In the past work of Anwar N.K &SaianR.[8] utilized two different diabetes datasets the Frankfurt Germany diabetes dataset and the Pima Indian diabetes dataset, various machine learning models involved in this study like Naïve Bayes, AdaBoost M1, K-nearest neighbor, and RIPPER. The main algorithm that they used in this research is Ant-Miner to make a comparison in terms of accuracy value. The highest accuracy obtained for the dataset is when they implement the Ant-Miner algorithm which is 73.64% compared to other algorithms.

In the previous work of Maniruzzaman M., et. al. [9] built a diabetic patient prediction algorithm based on machine learning (ML). Logistic Regression (LR) is used to classify risk factors for diabetes disease using p-values and odds ratios (OR). To forecast diabetic patients, they used four classifiers: Naive Bayes (NB), decision tree (DT), Ada boost (AB), and Random Forest (RF). These protocols were also followed and replicated in 20 trials by three groups of partition protocols (K2, K5, and K10). The accuracy (ACC) and region under the curve (AUC) of these classifiers are used to assess their performance (AUC) The ACC of the ML-based method as a whole is 90.62 %. The K10 protocol has a 94.25 % ACC and 0.95 AUC thanks to a mix of LR-based feature collection and RF-based classifier.

These studies have mainly focused on obtaining acceptance accuracy rates to diagnose and detect diabetic patients using different classification methods. Unlike these approaches, the main focus of the proposed model is to design a diabetes classification model based on 5 advanced Machine Learning Algorithms to achieving higher accuracy.

Research and Materials

In this work, a diabetes classification model is proposed based on advanced machine learning algorithms for diagnostic system accuracy enhancement. So in this section will be description test diabetes datasets, all classification algorithm that using in this work are explained in this section, and finally, show the evaluation measure.

Diabetes Dataset

A short overview of the datasets used will be given in this section. Downloading the diabetes dataset for classification was done via theKaggle machine learning repository. Every dataset comprises a set of numerical attributes instances. This data was gathered at the Frankfurt Hospital in Germany [10]. The dataset has 10 attributes and 2000 instances. The first column identifies each instance with an ID number while the last column of the data table is that of the class label that defines the diagnose of the diabetic that class variable 1 means is diabetes patient and class variable 0 meaning are non-diabetes patient [7]. Table 1 [7,11] lists the diabetes dataset instances and characteristics, as well as some statistical evidence.

Table 1. The diabetes dataset description

No	Feature Name	Feature Cod	Description	Rang
1	Pregnancies	PG	Number of women who are pregnant	0-17
2	Glucose	GL	In an oral glucose tolerance test, plasma glucose concentration was measured after 2 hours.	0-199
3	Blood Pressure	BP	Diastolic blood pressure (mm Hg)	0-122
4	Skin Thickness	ST	Thickness of Triceps skin fold (mm)	0-99
5	Insulin	IS	2-Hour serum insulin (mu U/ml)	0-846
6	BMI	BMI	Index mass of the body (weight in kg/(height in m)²)	0-67.1
7	Diabetes Pedigree Function	DPF	Diabetes pedigree function	0.078-2.42
8	Age	AGE	Age	21-81
9	Outcome	1 = yes 0 = no	Healthy=0 Diabetes=1	

Prediction by Machine Learning Algorithms

Some of the aspects of the supervised Machine Learning algorithms that have been considered would be covered in this section. For a comparative study of forecasting diabetes diseases from the datasets presented above, the following algorithms are considered.

Naive Bayesian for Classification

One of the most common classification algorithms is the Naive Bayesian approach which derives from Bayesian theory for probability theory.

Probability Theory Basics

The probability derived by splitting the number of times A occurs into the overall number of times A occurs is the likelihood of an occurrence A occurring. The chance value is always in the range of 0 to 1, and it can also be expressed as a percentage. For example, the likelihood of selecting one of the letters C, D, E, F, G, or H at random is p (sound) = 1/6, or roughly 16.67 %. When two unrelated events do not influence each other's probability, they are referred to as independent events in probability theory. If the conditional probability of B resulted in A equaling the probability of B, A and B are referred to as two separate events, defined by P (A) and P (B) [12].

$$P(B \mid A) = p(A).(B)/P(A) \quad (1)$$

Bayesian Classification Basic

The Bayesian classification is based on Thomas Bayes' principle (1702-1761). After his death in 1763, his theory was published in the Philosophical Transactions of the Royal Society of London in an essay titled "On the Problem of Opportunity" [12]. The Bayesian approach is concerned with determining the probability of a certain trend (sample) X given a set of observations. The posterior likelihood, on the other hand, is used to measure pattern X (that is, the probability of a certain model belonging to class I that gives its observed functional values) of a class or hypothesis is computed as shown in equations (2).

h, P (h|X), is denoted as $P(h|X) = \frac{P(X|h)P(h)}{P(X)}$ (2)

where P (h) $=\frac{|h|}{N}$ is the estimated h prior probability (given |h| is class h number of patterns and N is the total patterns number and assumes that all assumptions are equally probable), P (X|h) is the X conditional probability which conditioned on h, and P(X) is the X prior probability. The maximum posterior (MAP) hypothesis is used to assign the class h having maximum P (h|X). Equation (3) expresses that as shown:

$h_{MAP} \equiv \arg\max_{h \in H} = \arg\max_{h \in H} P(X|h)P(h)$ (3)

Where H is the hypotheses set.

The Bayesian classifier can efficiently perform the minimum error rate if the distribution of data probabilities is given. In this context, the expected loss of decision-making (i.e. conditional risk) will be minimum. This statistically optimal classification rule is widely used as a benchmark to which other classification algorithms are often measured [13].

Naive Bayesian Classifier

This is a widespread classification of probabilistic usage in a number of applications. The Bayes classification is based on the theorem from Bayes, and the naive adjective comes from the presumption that a data set features each other. In other words, considering the nature of the lesson, it is presumed that all attributes (features) of the training examples are independent of one another [14]. The NB classifier represents each pattern X as an n-dimensional vector of attribute values [a_1, a_2… a_n] Given that there are l class's c_1, c_2… cn. The classifier anticipates that an undefined pattern X corresponds to the class with the highest posterior probability conditional on X., i.e., X is assigned to the class C_i if and only if

$P(C_i|X) > P(C_j|X)$ (4)

For 1 <= j <i and j ≠ i. using the equation (2), getting the equation (5).

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \quad (5)$$

The classifier, however, makes the naive or simplified statement that the features (whose cumulative number is denoted by n) are conditionally independent of one another to save on computational costs. The class-conditional independence can be written as:

$$P(X|C_i) = \prod_{j=1}^{n} P(f_j|C_i) \quad (6)$$

As P(X) is a constant for each class, and $P(C_i) = \frac{|C_i|}{N}$, NB classifier needs to maximize only the $P(X|C_i)$. Since it just counts the class distribution, this greatly decreases computing costs. [12].

Gaussian Naive Bayes

When working with continuous data, it's common to assume that the continuous values associated with each class follow a Gaussian distribution. The mean and variance of each class are determined after the training data is segmented by class. As a result, the following approximation can be used to approximate the probabilities of a continuous dataset [14].

$$P(X=x|C=c) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \quad (7)$$

And;

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (8)$$

Finally;

$$\sigma = [\frac{1}{n-1}\sum_{i=1}^{n}(xi - \mu)^2]^{0.5} \quad (9)$$

Where: n is the number of instances; x_i is a specific value of the x variable for the i^{th} instance.

Random Forest Classifier

A Random Forest is a classifier that consists of several different tree classifiers {h (x, θ _k) k = 1,2,….}, where Θ k are irregularly distributed scattering vectors and each tree selects a unit for the common class at the input x [15]. A Random Forest is created by arbitrating a set of trees.

Breiman used the following steps to create each tree in a Random Forest: if N is the size of the files in the prep set, at this point, N files are displayed especially but by division, from the initial information; this is a bootstrap test. This example will be a preparation for the development of the tree. In cases where there are input elements M, the number m << M is chosen with the ultimate goal that m elements are randomly selected in M in each position, and the best distribution of these m units is to separate the center. The evaluation of m in the background field development is maintained [16].

Thus, many trees are created in the background forest; Ntree size preselects the size of the tree. The amount of variables items (m) selected in each center is indicated as m, t, r, y in the item. The size of the perimeter (for example, the number of events in the leaves, nodes), which is normally set to one, will limit the depth of the tree.

When the forest is ready or designed as above, to organize another event, it is run with all the trees that are filled. The new event is allocated to each tree, and it is registered as a vote [17].
All of the trees' votes are put together, and the class with the most votes e.g, the largest share of votes is presented as a description of the new event [15]. As measuring the tree's formation, when the starter test kit is created by examining and swapping each tree, about 1/3 of the instances cases are missing. This setting is called (OOB) data. Each tree has its OOB information index that is used to estimate individual errors. It is called OOB error evaluation. A forest random generalization error is reported as [18]:

$$RE^* = f_{xy}\big(mg(X.Y)\big) < 0 \quad (10)$$

The margin function is given as,

$$mg(X.Y) = av_k I(h_k(x) = Y) - max_{j \neq Y} av_k(h_k(X) = j) \quad (11)$$

The margin function measures the extent to which the average number of votes at (X, Y) for the right class exceeds the average vote for any other class [19]. Strength of Random Forest is given in terms of the expected value of margin function as,

$$S = E_{X.Y}\big(mg(X.Y)\big) \quad (12)$$

The following equation gives an upper bound for generalization error if ρ is the mean value of the association between base trees:

$$PE^* \leq P(1 - S^2) / S^2 \quad (13)$$

As a result, to improve Random Forest precision, the base decision trees must be diverse and precise [19].

Algorithm of K-Nearest Neighbours

K-Nearest Neighbours is a straightforward algorithm that produces excellent results. It's an instance-based, lazy, non-parametric learning algorithm. This algorithm can be used to solve problems including classification and regression. In classification mode, KNN is used to determine the class the new unlabeled object belongs to. The "k" is decided (where k is the neighbor's element number) which is generally odd and the distance between the data points closest to the objects is measured using methods such as Manhattan distance, Euclidian's distance, Minkowski distance, or Hamming distance. Figure 1 depicts an example of binary classification in action [7].

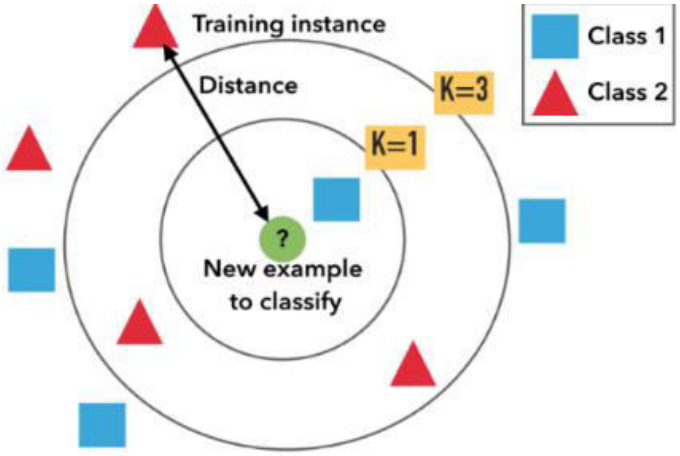


Figure 1: KNN with two groups to predict a new example [7].

The positive integer k value is calculated by examining the dataset, as seen in Figure 1. Cross-validation is a technique for retrospectively determining a strong value of k by validating the k with an independent data set. In this analysis, ten cross-validations will be used, and the values (k =1) will be used because they yield the best outcomes [7].

Support Vector Classifier

SVM is a set of similar supervised learning techniques for the classification of medical diagnosis. SVM maximizes the geometric margin while minimizing the empirical classification error. As a result, SVM stands for Maximum Margin Classifiers. The kernel trick allows SVMs to do non-linear classification effectively by indirectly mapping their inputs into high-dimensional feature spaces. The kernel trick enables the classifier to be built without having to recognize the feature space directly. An SVM model is a representation of the examples as points in space, mapped in such a way that the examples of the different groups are separated by a clear gap as large as possible. An SVM, for example, considers a hyperplane with the greatest possible fraction of points from the same class on the same plane given a collection of points from one of the two classes. The ideal separating hyperplane (OSH) is a separating hyperplane that maximizes the distance between the two parallel hyperplanes

and reduces the probability of misclassifying test dataset instances. As data points of the form, given labeled training data [20].

$$M = \{(x_1,y_1),(x_2,y_2),\ldots,(x_n,y_n)\} \quad (14)$$

Where y_n is constant (1/-1) that denotes the class that point $x_n$belongs, where n is the data sample number. Each $x_n$is a p-dimensional real vector. The SVM classifier converts the input vectors into a decision value before performing classification with the aid of a threshold value. Divide the hyperplane, which can be represented as an equation, to view the training data (15) [20].

$$\text{Mapping}: W^T.x + b = 0 \quad (15)$$

where w is a scalar and b is a p-dimensional weight vector The dividing hyperplane is perpendicular to the vector w. The margin can be increased by using the offset parameter b. Select these hyperplanes such that there are no points between them while the training data is linearly separable, and then attempt to maximize the distance between them. SVM has found out the distance between the hyperplane as $2 / |x|$ as shown in Figure (2).To minimize$|W|$, then will need to ensure for all i either using equation (16)[21].

$$w.x_i - b \geq 1 \; or \; w.x_i - b \leq -1 \quad (16)$$

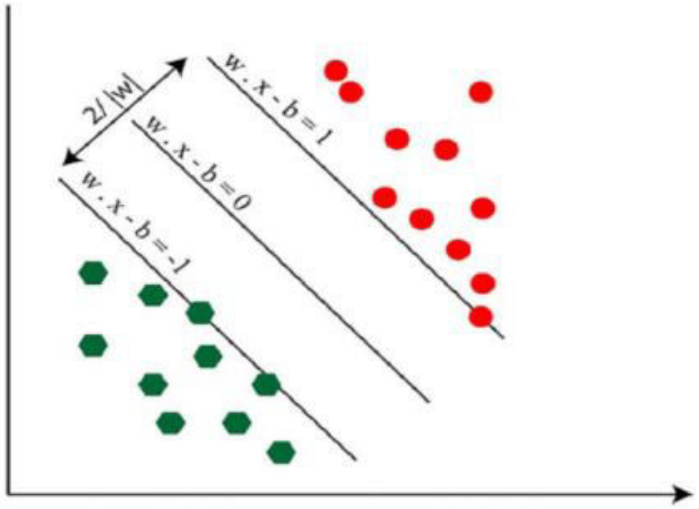


Figure 2. SVM hyperplane with maximum margin trained for two-class samples [21].

Logistic Regression Classifier

The Logistic regression (LR) model has been used extensively in a variety of fields, including the biological sciences. Where the aim is to distinguish data objects, the logistic regression algorithm is used. In most cases, the goal variable in logistic regression is binary, so it only includes data that can be labeled as 1 or 0, and this example applies to a patient who is positive or negative for diabetes. The logistic regression algorithm aims to find the best fit for describing the relationship between the target variable and the predictor variables that is diagnostically rational. The algorithm of logistic regression is based on the following linear regression in equation (17) [22]:

$$y = h_{\ominus}(x) = \ominus^T x \quad (17)$$

Equation (17) is very inefficient for binary values to be predicted$(y^{(i)}\varepsilon\{0,1\})$, The function is therefore introduced in equation (18) to forecast the probability of a particular patient belonging to the "1" class (positive) versus the probability of being of the "0" class (negative) [22].

$$P(y = 1/x) = h_{\ominus}(x) = \frac{1}{1 + \exp(-\ominus^T x)} = \sigma(\ominus^T x)$$
$$P\left(y = \frac{0}{x}\right) 1 - P\left(y = \frac{1}{x}\right) = 1 - h_{\ominus}(x) \quad (18)$$

The equation (19), called the sigmoid function, will maintain the value of $\ominus^T x$ within the [0, 1] range. Then LR look up a value θ as if the probability $P(y = 1/x) = h_{\ominus}(x)$ is high if x is of the class "1" and small, if x is of the class "0 (i.e.P (y = 0|x) is large) [22].

$$\sigma(t) = \frac{1}{(1 + e^{-1})} \quad (19)$$

Criteria of Performance Evaluation

A confusion matrix is a type of tool used to monitor the accuracy of the classifier in classification [23]. This tool explains the connection between the classes concerned and the forecast. The efficiency level of the classification

model is determined using a correct and incorrect number of classifications classified in the confusion matrix for each possible variable. Table 2 reveals the two-class confusion matrix [12]:

Table 2: Confusion Matrix of Two Classes

		predicted	
		Negative	**Positive**
Actual	**Negative**	**TN**	**FP**
	Positive	**FN**	**TP**

Where

- TP and TN stand for True Positive and True Negative, respectively, indicating the proportion of positive and negative states that were correctly identified.

- FP stands for False Positive, which refers to all negative cases that were falsely classified as positive, and FN stands for False Negative, which refers to all positive cases that were incorrectly classified as negative.

Accuracy

Accuracy measures the classifier's capability to produce the level of accurate diagnosis [23]. Equation (20) shows the accuracy formula.

$$Accuracy = \left(\frac{TP + TN}{TP + TN + FP + FN}\right) * 100 \quad (20)$$

Precision

It simply shows "what number of selected data items are relevant". In other words, out of the observations that an algorithm has predicted to be positive, how many of them are actually positive. According to formula (21), the precision equals the number of true positives divided by the sum of true positives and false positives.[25]

$$Precision = \frac{TP}{TP + FP} \quad (21)$$

Sensitivity or Recall

Recall: It presents "what number of relevant data items are selected". In fact, out of the observations that are actually positive, how many of them have been predicted by the algorithm. According to formula (22), the recall equals the number of true positives divided by the sum of true positives and false negatives[25]:

$$Recall = \frac{TP}{TP + FN} \quad (22)$$

Specificity

Specificity is the metric that evaluates a model's ability to predict true negatives of each available category. To compute the specificity value by applied equation (23)
[25]:

$$Specificity = \frac{TN}{TN + FP} \quad (23)$$

The proposed Model

This work aims to apply comprehensive studies to the diabetes dataset using a different ML classification technique to effectively detect diabetes. The detailed design of the proposed model is shown in Figure 3.

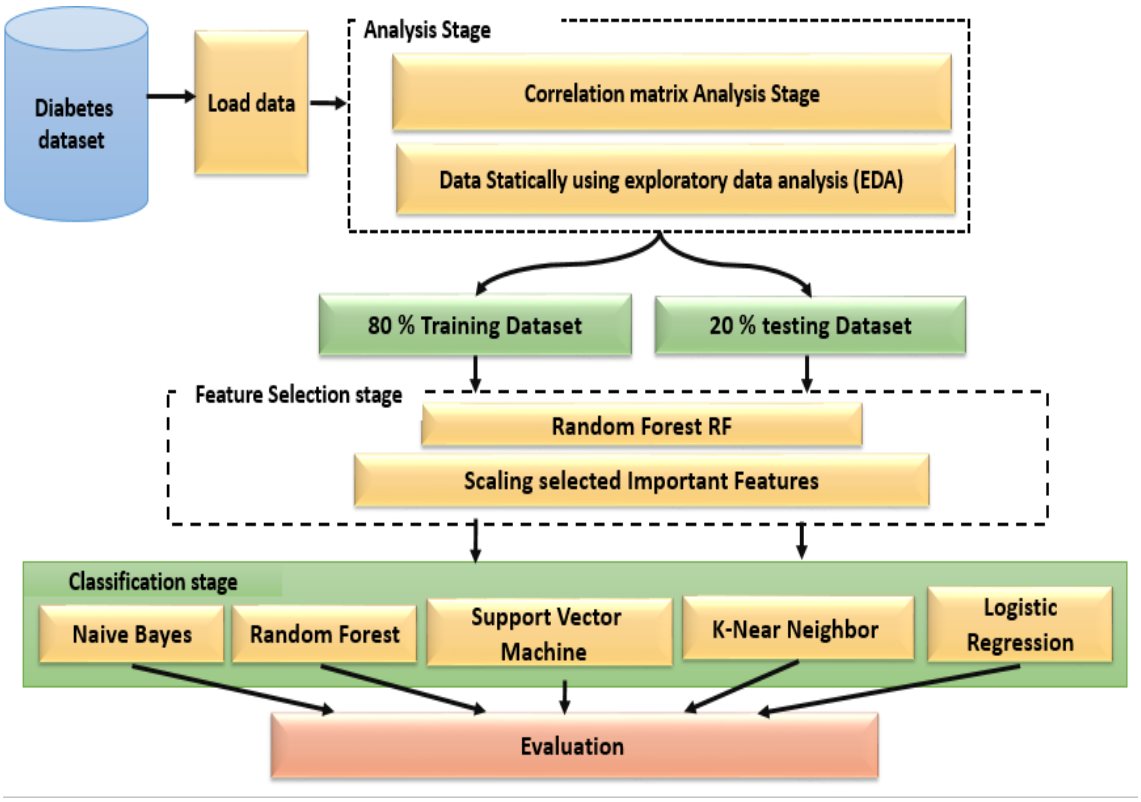


Figure 3. General Block Diagram of the Proposed Model.

As shown in Figure 2. the proposed work consists of five main stages: - Load diabetes, Analysis dataset using correlation matrix, preprocessing using data statistically, split dataset to 80%Train and 20%test, feature selection using Random Forest Feature analysis to Scaling selected Important Features, and classification stage (Naive Bayes, Random Forest, Support Vector Machine, K-Near Neighbor and Logistic Regression), and finally the result. Each stage in the proposed model will be explained in details in the subsection below:

Load Diabetes Dataset

Load diabetes dataset which includes 2000 samples with 9 attributes as illustrated in table (1) in section (3.1).

Analysis Diabetes Dataset

The second stage in the proposed model is the analysis diabetes dataset using the correlation matrix technique, it is a table showing correlation coefficients between variables. In the Diabetes Dataset, given two features, each feature represents a set of values or array components, and the correlation coefficient between two features must be determined using equation (24) to calculate the intensity of the relationship between two variables [11].

$$\mathbf{r} = \frac{N\sum xy - (\sum x)(\sum y)}{\sqrt{[N\sum x^2 - (\sum x)^2][N\sum y^2 - (\sum y)^2]}} \quad (24)$$

Where

N = the pairs of scores number

$\sum \boldsymbol{xy}$ = the products of paired scores sum

$\sum \boldsymbol{x}$ = the x scores sum

$\sum \boldsymbol{y}$ = the y scores sum

$\sum \boldsymbol{x^2}$ = the squared x scores sum

$\sum \boldsymbol{y^2}$ = the squared y scores sum

The cross-correlation coefficient is another name for the correlation coefficient. The correlation coefficient is always in the range of -1 to +1, with -1 indicating that X and Y are negatively correlated and +1 indicating that X and Y are positively correlated.

Preprocessing based on EDA Technique

Before the feature selection method, various preprocessing techniques were used because every data has a lot of hidden information. To discover the secret patterns, this hidden information had to be investigated. These trends

will assist in making procedure decisions, removing uncertainty, and gaining key business insights. As a result, exploratory data analysis was implemented as a dataset preprocessor [9].

The aim of exploratory data analysis is to gain a broad understanding of the data. It is primarily carried out in order to determine its properties, patterns, and visualizations. It assists us in ensuring the data is accurate and ready to be used by machine learning algorithms. In this work, the EDA used descriptive statistics which is represent attribute type, class distribution, mean, standard deviation, median, quartile, Skewness, correlation [24]. In this stage, divide the distribution into four classes and compute the value of quartile measures, which is above and below the mean the distribution of values. The quartile breaks down the data into quarters such that 25% of the measurements are less than the lower quartile, 50% are less than the mean, and 75% are less than the upper quartile, while the mean divides the data in half so that 50% of the measurements are below the median and 50% are above it. The mean is equal to the sum of all the values in the data. As a result, a data set can contain n values, each of which has a values $x_1, x_2, x_3,\ldots..x_n$, the sample mean, usually denoted by x (pronounced "x bar"), is as shown in the formula (25) is usually written in a slightly different manner using the Greek capital letter, ∑, pronounced " sigma", which means "sum of":

$$x^{-} = \sum_{i=1}^{n} \frac{x_i}{n} \qquad (25)$$

where :

∑ = Summation of… ;x^{-} = sample mean; n = scores in the sample number

Where the standard deviation formula in (26)

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_{i-x^{-}})^{2}}{n-1}} \qquad 26)($$

Where:

s = Sample standard deviation; ∑ = Summation of…; x^{-} = sample mean;n = scores in the sample number

1.1 Splitting Dataset to Test and Train

This work divides the dataset into 80% training set for the classifier and 20% a testing set used to evaluate classification system performance accuracy. The total samples in Diabetes Dataset are = 2000, in this stage splitting database into training and test as shown in table 3.

Table 3. Splitting Diabetes Dataset

Total Sample	80%traning	20% testing
2000	1600	400

Feature Selection by Random Forest Algorithm.

The Random Forest (RF) algorithm is a type of ensemble algorithm. The RF algorithm is often used for feature selection, and it works as follows: The first phase is splitting all features in diabetes dataset and building tree for each feature. The second phase includes the dataset is divided randomly, transformed into groups, and drawn for each group an independent tree .

Figure 4 shows distribution of all features ((Pregnancies(PG), Glucose(GL), Blood Pressure(BP), Skin Thickness(ST), Insulin(IS), BMI(BMI), Diabetes Pedigree Function (DPF),and age (AGE)) in diabetes dataset ,the red line represent class (1) and blue line represents class (0).

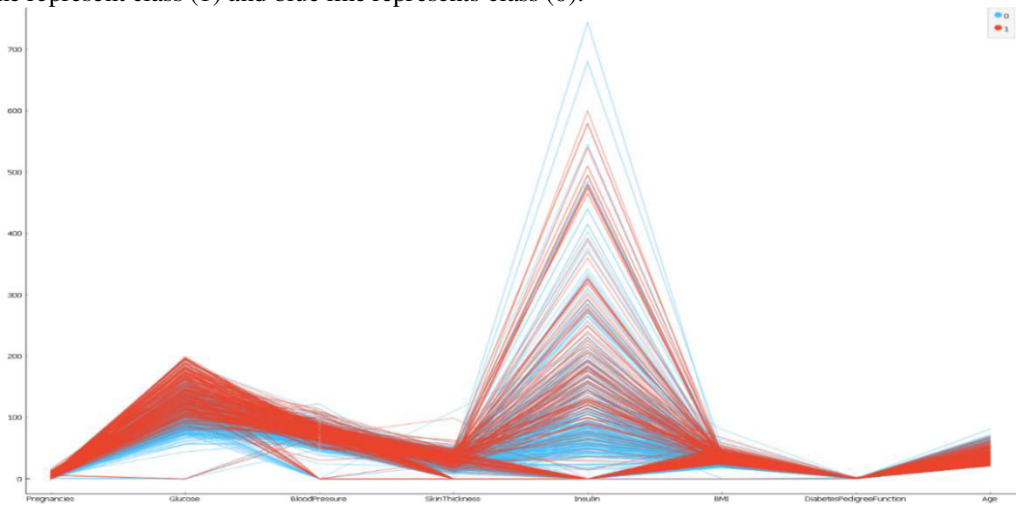


Figure 4. Distribution Data for All Features in The Dataset.

Random Forest algorithm takes all possibilities in a random way, each probability has drawn tree using RF algorithm, see Figure 5 that shown example of these tree.

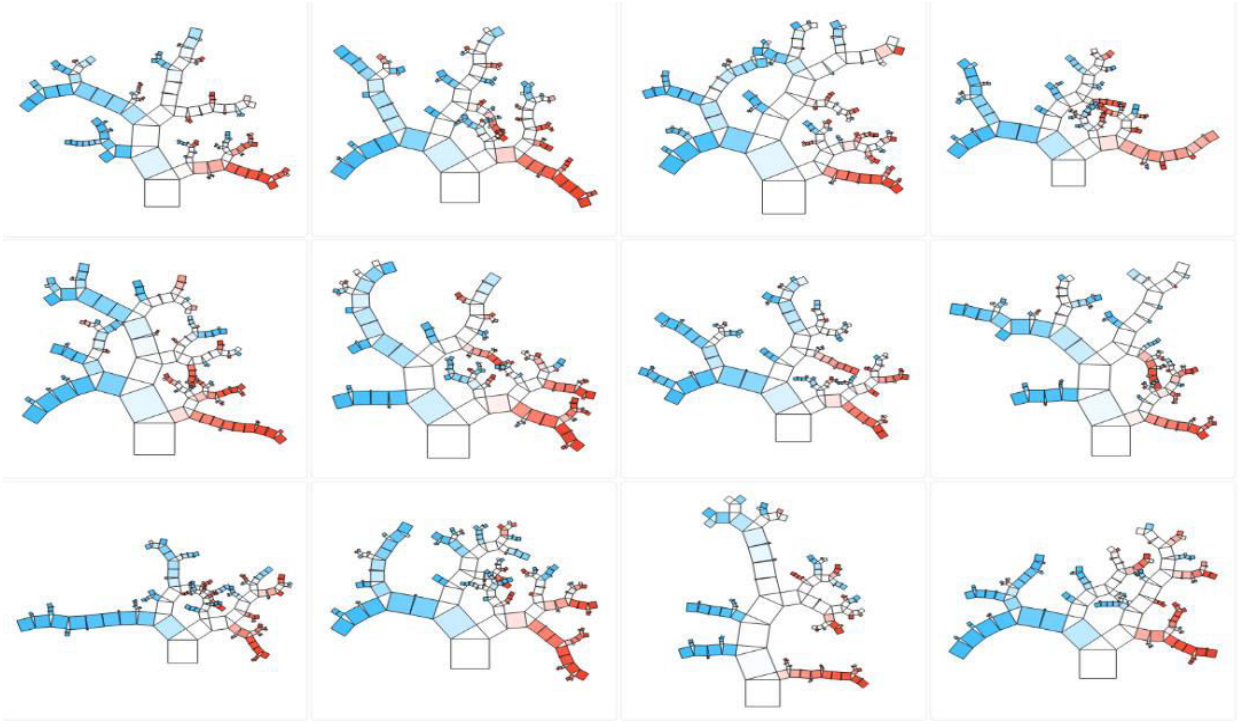


Figure 5. Example of Random Forest Tree for All Probability of the Dataset.

As shown in figure 5, note that the tree contains the red color, which indicates class 1, while the blue color indicates class 0. In addition, the color gets darker in the leaves of the tree, which depends on its value, which represents the best value for this group. Now the algorithm divides each class into groups and takes all possible possibilities by drawing independent trees for each group as shown in the figure 6 for class 0 .

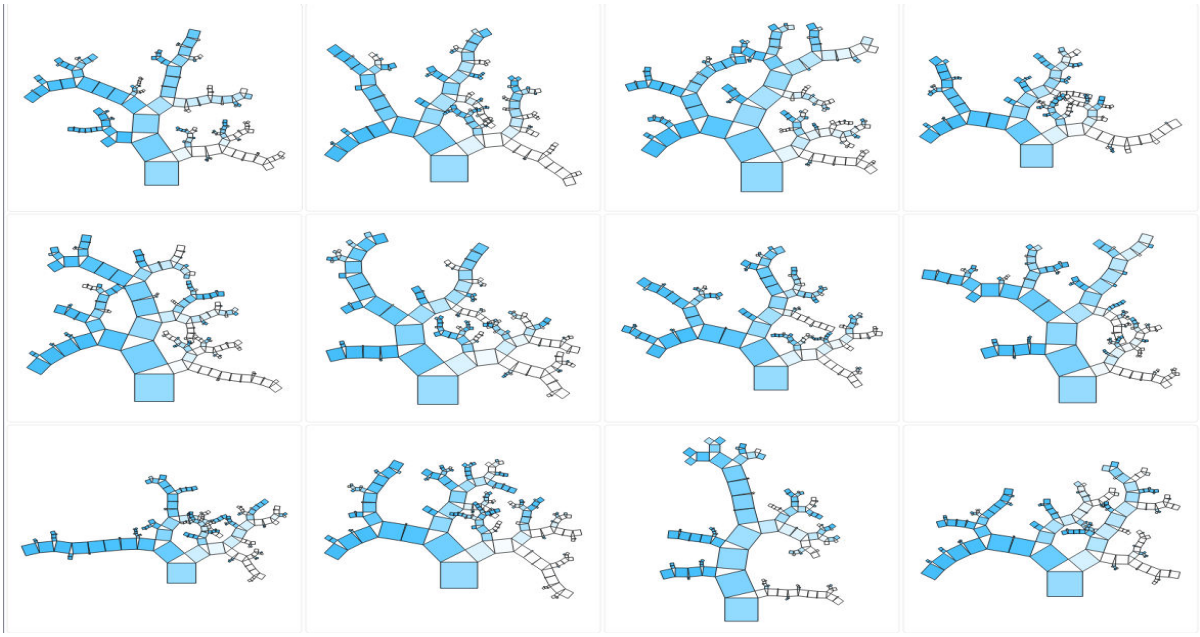


Figure 6. Random Forest Trees for Class 0.

Figure 7 illustrated the all possibility independent trees for class1.

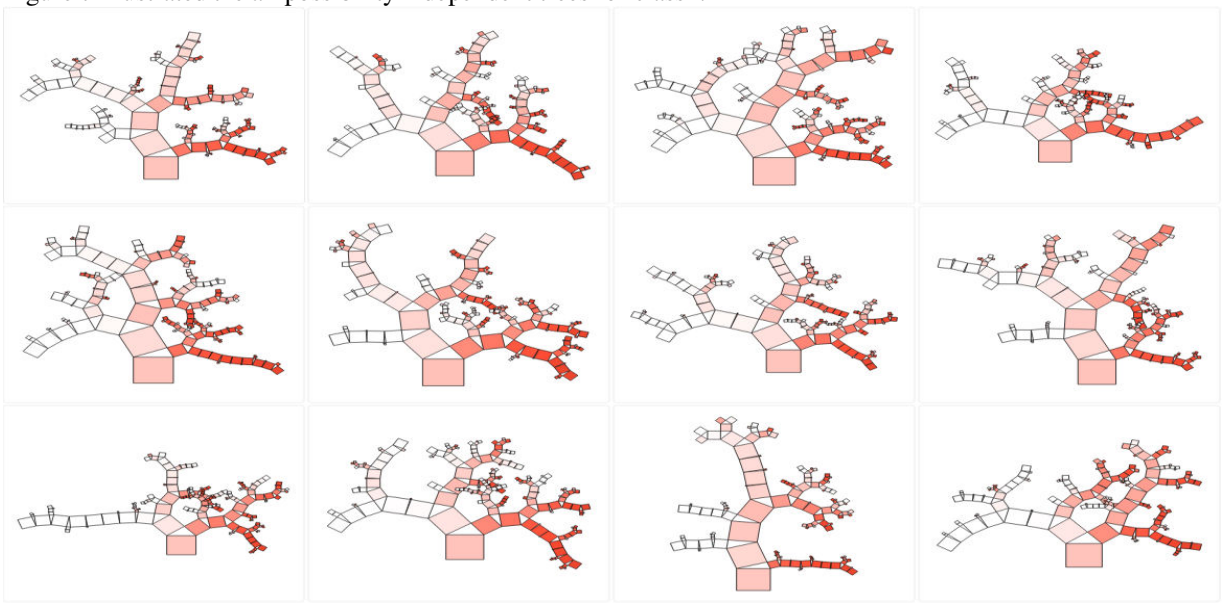


Figure 7 Random Forest Trees for Class 1.

Based on the probabilities of all the features in the data set computed using the Random Forest algorithm as shown in Figure 5 ,6 and 7 willtaken the highest values obtained for each Feature and then rearrange the feature from highest to lowest as shown in figure 8.

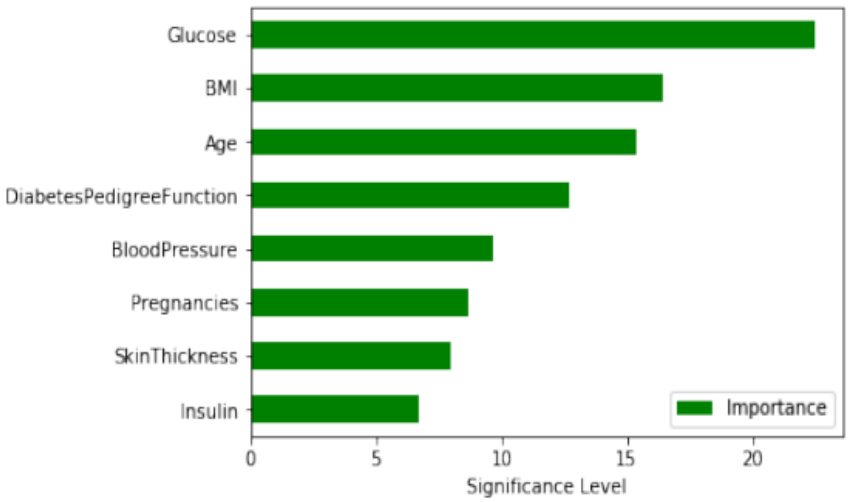


Figure 8 Levels Importance Features.

Then select 5 features which importance is greater than threshold 80% ,the value of threshold in this work will be proposed based on values of values. The important features that selected by the Random Forest Algorithm are GL, AGE, DPF, BP, and BMI, see distribution of data feature selection illustrated in figure 9.

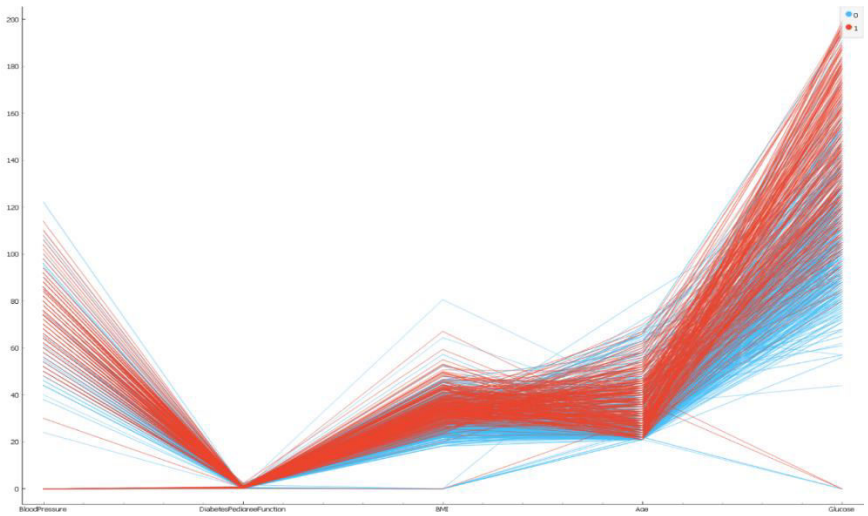


Figure 9 Distribution Data for GL, AGE, DPF, BP, and BMI Features After applied Random Forest Algorithm.

Standardize features by extracting the mean and scaling to unit variance after choosing a function using the RF algorithm. As shown in equation (27) [11], the standard score of sample x is estimated:

$$s = z = (x - u) / s \quad (27)$$

Where u is the training samples' mean, or zero if with mean=False, and S is the training samples' standard deviation, or one if with std=True.

NB,RF,KNN,SVM, and LR Classification

To classify diabetes and Non-diabetes people, the proposed model using five ML algorithms Gaussian Naïve Bayes (NB), Random Forest (RF), K-Neighbor Classification algorithms (KNN), support vector machine (SVM), and Logistic Regression (LR).

Results Discussion

The results of each step of the proposed system are illustrated in this section, where the proposed model implementation under Windows 10 Professional operating system, Intel(R) Core(TM) i5-2450M CPU @ 2.50GHz, 8 GB random access memory, and 64-bit system type and to run the proposed system within an environment of python language.

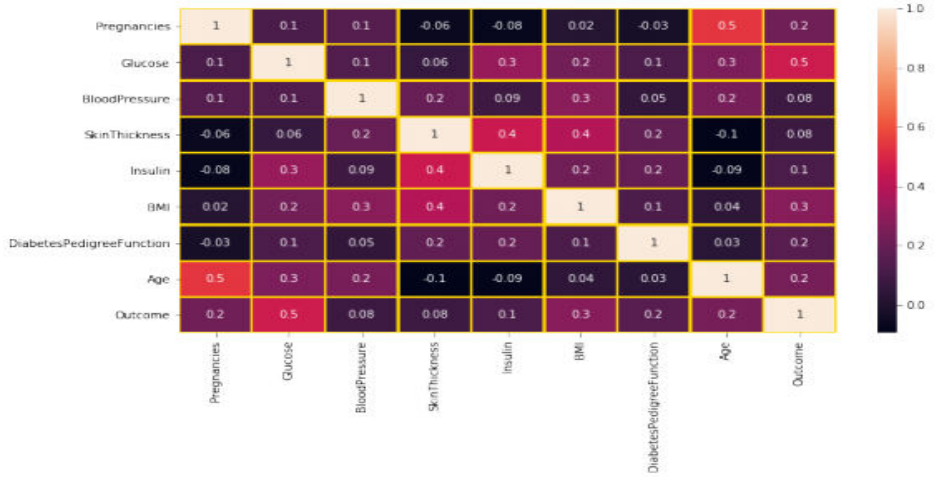


Table 1 list the diabetes dataset instances and attributes ,as well as some statistical data .A correlation between the data set features is also visualized in Figure 10 using correlation matrix .

Figure 10 Correlation Matrix Analysis for all features in Diabetes Dataset.

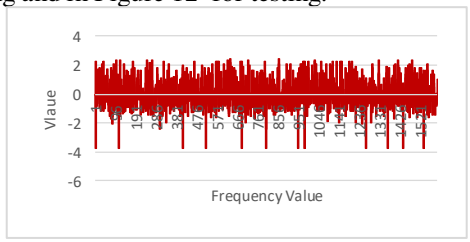
As shown in figure 4, all the diagonal elements of the correlation matrix (c) must be 1 because the correlation of a variable with itself is always perfect, $c_{ij}=1$ and It should be symmetric $c_{ij}=c_{ji}$. The diagonal element divided a correlation Coefficient matrix into two parts (Top triangle and bottom triangle) and bout they have the same

values which are in the range between [1 to -1].Table 4. shows the results of the AED technique to descriptive statistics for all features in Diabetes Dataset by calculating the mean using equation (25), standard deviation using equation (26), count, min, max, and Quantile (25%,50%,75%).

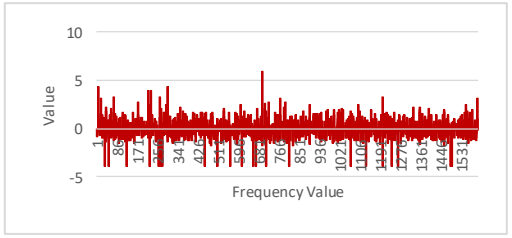
Table 4. Results of EDA Technique on Diabetes Dataset.

Feature ID	count	mean	SD	Min	max	25%	50%	75%
PG	2000.0	3.70350	3.306063	0.000	17.00	1.000	3.000	6.000
GL	2000.0	121.18250	32.068636	0.000	199.00	99.000	117.000	141.0000
BP	2000.0	69.14550	19.188315	0.000	122.00	63.500	72.000	80.000
ST	2000.0	20.93500	16.103243	0.000	110.00	0.000	23.000	32.000
IS	2000.0	80.25400	111.180534	0.000	744.00	0.000	40.000	130.000
BMI	2000.0	32.19300	8.149901	0.000	80.60	27.375	32.300	36.800
DPF	2000.0	0.47093	0.323553	0.078	2.42	0.244	0.376	0.624
AGE	2000.0	33.09050	11.786423	21.000	81.00	24.000	29.000	40.000
outcome	2000.0	0.34200	0.474498	0.000	1.00	0.000	0.000	1.000

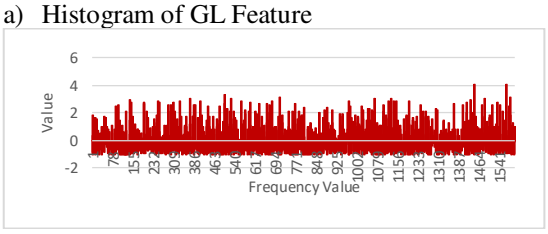
Table 4 can find the similarity between all features, and when finding two features or more equal, then will take one of them to increase the accuracy of the system in the classification stage. After selecting the important features using the Random Forest Algorithm as shown in section (4.3). In this stage, the data of these Features will be scaling using equation (27). The data distribution after applied the scaling algorithm for the five features (Glucose (GL), BMI , Age, Diabetes Pedigree Function (PDF) , Blood Pressure (BP)) is illustrated in Figure 11 for training and in Figure 12 for testing.



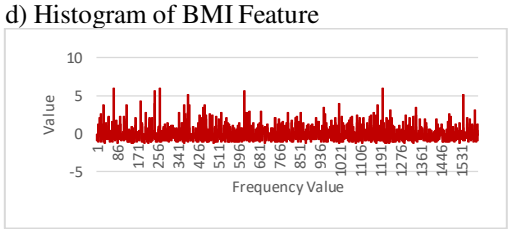
a) Histogram of GL Feature



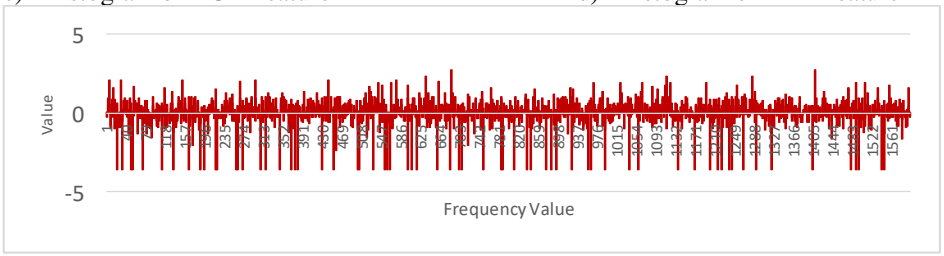
d) Histogram of BMI Feature



c) Histogram of AGE Feature



d) Histogram of PDF Feature



e) Histogram of BP Feature

Figure 11 Training Feature After applied Scaling Data

a) Histogram of GL Feature

d) Histogram of BMI Feature

c) Histogram of AGE Feature

b) Histogram of PDF Feature

e) Histogram of BP Feature

Figure 12 Testing Features After applied Scaling Data

Figure 11 and 12 explain the effect of the scaling algorithm on the distribution of training and testing data, where the effected means the values of a single feature are limited to a certain range by it gives equal weight to very small values (which could only be noise) and large values. Scaling up the small variables (may be no relevant also) could change the results profoundly.

The comparative performance of NB, RF, KNN, SVM, and LR Classification algorithms based on accuracy rate using equation (20) are shown in Figure 13. which is illustrated that the RF obtains the best accuracy rate in comparison with other classification algorithms used in the proposed system, where the accuracy of RF=99% with the accuracy of the KNN=98.75% , the accuracy of SVM=81%, the accuracy of LR=77.5%, and the accuracy NB=77.25.

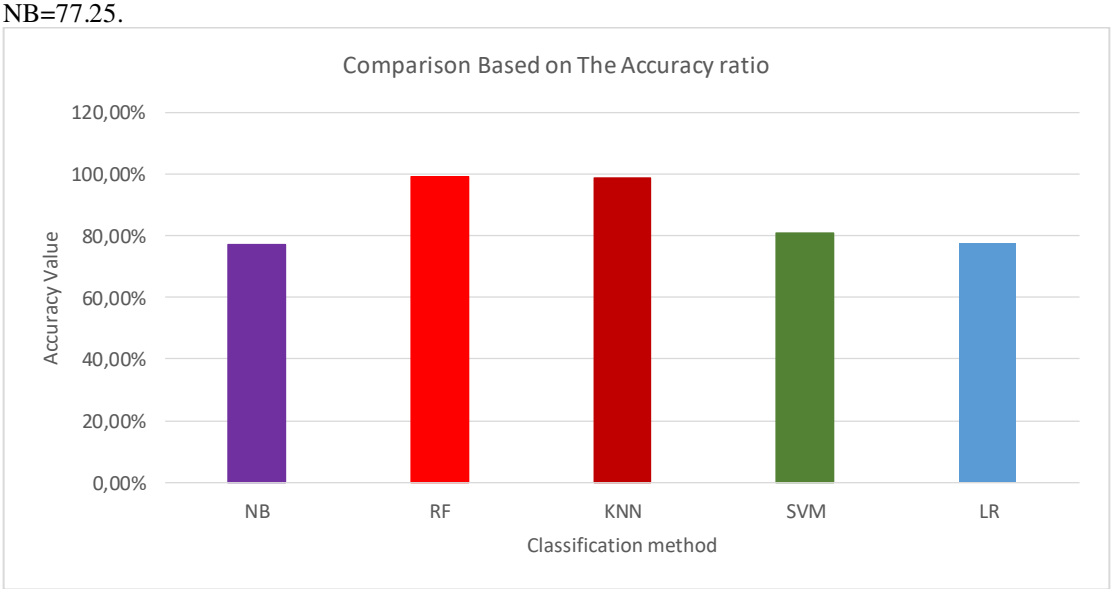


Figure 8. Comparison Between Five Classification Algorithm that using in the Proposed System Based on Accuracy Ratio.

Table 5 calculates the value of Precision metric using equation(21) ,Sensitivity or Recall metric using equation (22) and Specificity metric using equation (23) of five classification algorithms(Gaussian Naïve Bayes (NB) algorithm, Random Forest (RF) algorithm, k-Nearest Neighbor (KNN) algorithm, Support Vector Machin (SVM) algorithm ,and finally Logistic Regression (LR) algorithm).

Table (5) Classification performance based on Precision , Sensitivity, and Specificity ratio.

No	Methods	Precision	Sensitivity	Specificity
1	NB	97.9591 %	99.2094%	98.6301%
2	RF	60.5442 %	87.3517%	73.5537%
3	KNN	98.6394 %	98.8142%	97.9729%
4	SVM	48.2993 %	99.2094%	97.2602%
5	LR	57.1428 %	88.9328%	75.0%

Table 5 compares the proposed approach to the previous methods in terms of precision. With the RF classification algorithm, the proposed method performed well in terms of accuracy, achieving 99 % accuracy. This proves that the proposed method can effectively diagnose diabetes.

Table 5. On the diabetes dataset, the proposed approach is compared to previous methods in terms of accuracy.

No.	Reference	Method	Accuracy
1	Miao L., et al. [6]	SVM	96.5%
2	CherradiB.,et. Al[7]	KNN	97.53%.
3	Anwar N.K &Saian R. [8]	Ant-Miner	73.64%
4	The proposed System	RF	**99%**

Conclusion

One of the biggest problems in the healthcare sector is detecting diabetes early. In the proposed model, build a model that can accurately predict diabetes. The proposed model uses a diabetes dataset that has 2000 instances with 9 features and passes this data through several stages are: Load diabetes data set, Analysis dataset using correlation matrix, preprocessing using data statistically, split dataset to 80%Train and 20%test, feature selection using Random Forest Feature analysis to Scaling selected important Features, and classification stage (Naive Bayes, Random Forest, Support Vector Machine, K-Near Neighbor and Logistic Regression). The results of the performance of the proposed model based on accuracy rate are show KNN obtains the best accuracy rate in comparison with other classification algorithms used in the proposed system, where the accuracy of KNN=98.75% while the accuracy of NB=77.25%, the accuracy of RF=99%, the accuracy of SVM=81%, and the accuracy of LR=77.5%. Besides, compared to previous methods, the proposed approach is more accurate 99 % accuracy was achieved, which was a great result with the RF classification algorithm. This proves that the proposed method can effectively diagnose diabetes.

Reference

[1] Srivastava, S., Sharma, L., Sharma, V., Kumar, A., & Darbari, H. (2019). Prediction of diabetes using artificial neural network approach. In *Engineering Vibration, Communication and Information Processing* (pp. 679-687). Springer, Singapore.

[2] Tama, B. A., & Rhee, K. H. (2019). Tree-based classifier ensembles for early detection method of diabetes: an exploratory study. Artificial Intelligence Review, 51(3), 355-370.

[3] Battineni, G., Sagaro, G. G., Chinatalapudi, N., &Amenta, F. (2020). Applications of machine learning predictive models in the chronic disease diagnosis. Journal of personalized medicine, 10(2), 21.

[4] Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., &Chouvarda, I. (2017). Machine learning and data mining methods in diabetes research. Computational and structural biotechnology journal, 15, 104-116.
[5] Kaware, S. R., &Wadne, V. S. (2020). Improve the performance of cancer and diabetes detection using novel technique of machine learning (No. 2332). EasyChair.
[6] Miao, L., Guo, X., Abbas, H. T., Qaraqe, K. A., & Abbasi, Q. H. (2020, August). Using Machine Learning to Predict the Future Development of Disease. In 2020 International Conference on UK-China Emerging Technologies (UCET) (pp. 1-4). IEEE.
[7] Daanouni, O., Cherradi, B., &Tmiri, A. (2019, October). Predicting diabetes diseases using mixed data and supervised machine learning algorithms. In Proceedings of the 4th International Conference on Smart City Applications (pp. 1-6).
[8] Anwar, N. H. K., &Saian, R. (2020). Predictive accuracy for two diabetes datasets using ant-miner algorithm. International Journal of Scientific and Technology Research, 9(4), 239-242.
[9] Maniruzzaman, M., Rahman, M. J., Ahammed, B., & Abedin, M. M. (2020). Classification and prediction of diabetes disease using machine learning paradigm. Health information science and systems, 8(1), 1-14.
[10] https://www.kaggle.com/emrzcn/diabetes-prediction-with-lr-knn-nb-svm-rf-gbm.
[11] Haq, A. U., Li, J. P., Khan, J., Memon, M. H., Nazir, S., Ahmad, S., ... & Ali, A. (2020). Intelligent Machine Learning Approach for Effective Recognition of Diabetes in E-Healthcare Using Clinical Data. Sensors, 20(9), 2649.
[12] Yin, H., & Chaoyang, Z. (2011, October). An improved bayesian algorithm for filtering spam e-mail. In 2011 2nd International Symposium on Intelligence Information Processing and Trusted Computing (pp. 87-90). IEEE.
[13] Kamel, H., Abdulah, D., & Al-Tuwaijari, J. M. (2019, June). Cancer Classification Using Gaussian Naive Bayes Algorithm. In 2019 International Engineering Conference (IEC) (pp. 165-170). IEEE.
[14] S-B. Kim, K-S. Han, H-C. Rim and S. H. Myaeng, "Some Effective Techniques for Naive Bayes Text Classification", IEEE Transactions on Knowledge and Data Engineering, vol.18, no.11, pp.1457-1466, Nov. 2006.
[15] Arora, J., & Agrawal, U. (2020). Classification of Maize leaf diseases from healthy leaves using Deep Forest. Journal of Artificial Intelligence and Systems, 2(1), 14-26.
[16] Gobalakrishnan, N., Pradeep, K., Raman, C. J., Ali, L. J., & Gopinath, M. P. (2020, July). A Systematic Review on Image Processing and Machine Learning Techniques for Detecting Plant Diseases. In 2020 International Conference on Communication and Signal Processing (ICCSP) (pp. 0465-0468), IEEE.
[17] Mecheter, I., Alic, L., Abbod, M., Amira, A., & Ji, J. (2020). MR Image-Based Attenuation Correction of Brain PET Imaging: Review of Literature on Machine Learning Approaches for Segmentation. Journal of Digital Imaging, 1-18.
[18] Tigga, N. P., & Garg, S. (2020). Prediction of type 2 diabetes using machine learning classification methods. Procedia Computer Science, 167, 706-716.
[19] Chow, L. S., &Paramesran, R. (2016). Review of medical image quality assessment. Biomedical signal processing and control, 27, 145-154.
[20] Tambade, S., Somvanshi, M., Chavan, P., & Shinde, S. (2017). SVM-based diabetic classification and hospital recommendation. International Journal of Computer Applications, 167(1), 40-43.
[21] Nguyen, L. (2017). Tutorial on support vector machine. Applied and Computational Mathematics, 6(4-1), 1-15.
[22] Zhu, C., Idemudia, C. U., & Feng, W. (2019). Improved logistic regression model for diabetes prediction by integrating PCA and K-means techniques. Informatics in Medicine Unlocked, 17, 100179.
[23] Doreswamy, H. K. (2012). Performance evaluation of predictive classifiers for knowledge discovery from engineering materials data sets. arXiv preprint arXiv:1209.2501.
[24] Indrakumari, R., Poongodi, T., & Jena, S. R. (2020). Heart Disease Prediction using Exploratory Data Analysis. Procedia Computer Science, 173, 130-139.
[25] Vakili, M., Ghamsari, M., & Rezaei, M. (2020). Performance Analysis and Comparison of Machine and Deep Learning Algorithms for IoT Data Classification. arXiv preprint arXiv:2001.09636.