Acta Scientiarum
http://periodicos.uem.br/ojs
ISSN on-line: 1807-8664
Doi: 10.4025/actascitechnol.v46i1.64783

The diabacare cloud: predicting diabetes using machine learning

Mehtab Alam*, Ihtiram Raza Khan, Mohammad Afshar Alam, Farheen Siddiqui and Safdar Tanweer

Department of Computer Science and Engineering, School of Engineering Sciences & Technology, Jamia Hamdard, New Delhi, India. *Author for correspondence. E-mail: mahiealam@gmail.com

ABSTRACT. Machine learning (ML) is the buzz all around the technology industry and is illuminating each and every sector of human lives, be it, healthcare, finance, bioinformatics, data science, mechanical engineering, agriculture or even smart cities nowadays. ML consists of supervised and unsupervised techniques. Due to the availability of data in abundance, supervised ML has been the most preferred method in the field of data mining. In this research paper, a publicly available dataset for diabetes detection is tested to understand the efficiency of classification of a number of supervised ML algorithms to find the most accurate model. The dataset consisted of data of 768 persons out of which 500 were control and 268 were patients we found that the Random Forest algorithm outperformed the other 6 classification algorithm. In the first iteration, the Random Forest algorithm reached 78.44% accuracy. The tweaks performed in the paper outclassed the original random forest algorithm with a difference of 1.08% reaching a score of 79.52%. Further, iteration I gave 171 whilst iteration II gave 173 correct predictions out of the total 218 test data.

Keywords: machine learning; artificial intelligence; diabetes; ML; AI; random forest.

Received on August 25, 2022.
Accepted on March 14, 2023.

Introduction

Diabetes is one of the most common chronic diseases and can sometimes prove to be life-threatening if not treated in time. The prime attribute of the disease is the high level of glucose in the blood of a person. The increased level of glucose is due to some inadequacy in insulin secretion by the pancreas and/or its diminished biological effects (Lonappan et al., 2007). Diabetes, in its extreme, can lead to the death of the patient, but in less severe scenarios it can lead to severe chronic damage and flawed functioning of vital organs such as the blood vessels, heart, eyes, kidneys, and nerves (Krasteva, Panov, Krasteva, Kisselova, & Krastev, 2011; Parveen, Sehar, Bajpai, & Agarwal, 2020).

Diabetes is characterized into 2 distinct types, Type 1 Diabetes and type 2 diabetes. In type 1 diabetes, the pancreas produces very little to no insulin which is responsible for helping the blood sugar enter the cells where it is used to produce energy. It is usually diagnosed in young people below the age of 30 years, but it can prosper at any stage of life. Some of the symptoms of type 1 diabetes are an increase in thirst and an increase in the frequency of urination (Dua, Doyle, & Pistikopoulos, 2006). Type 2 diabetes is seen very commonly in middle-aged and elderly people. Obesity, dyslipidemia, hypertension and arteriosclerosis are actively associated with the onset of chronic disease (Islam, Qaraqe, Belhaouari, & Abdul-Ghani, 2020). Type 1 is less common when compared to type 2 diabetes and approximately 5-10% of diabetes patients are type 1. A cure for diabetes is still not found. It can only be controlled and regulated with healthy health habits.

Type 1 Diabetes Mellitus (T1DM), is also known as insulin-dependent diabetes mellitus which consists of only 5-10% of all diabetes mellitus cases. T1DM is an autoimmune disorder that results in the deficiency of insulin in the body and in due time developing hyperglycemia. T1DM is also greatly influenced by environmental as well as genetic factors (Banday, Sameer, & Nissar, 2020). While Type 2 Diabetes Mellitus (T2DM), also called non-insulin-dependent diabetes mellitus constitutes around 90-95% of all diabetes mellitus cases. It is characterized by insulin resistance and β-cell dysfunction. T2DM is linked to increasing age, family history of diabetes, physical inactivity, obesity, adoption of modern lifestyles, and with conditions such as hypertension and dyslipidemia (Genuth, Palmer, & Nathan, 2018).

ML and Artificial Intelligence (AI) have been around for a while now. ML is a very efficient approach for analyzing data for scientific as well as clinical studies. ML techniques are being used to classify individuals

with and without a required health condition. ML techniques are very likely to provide high accuracy in the classification of data. The predicted results can be highly accurate (Alam, Khan, Siddiqui, Wiquar, & Anwar, 2021).

The main aims of the study are listed below

- Q1: Which is the most accurate and suitable ML classification model for classifying the presence of diabetes in the patient, with a significant and reasonable accuracy rate?
- Q2: What are the key indicators that can help in designing the classification model for predicting diabetes in patients?
- Q3: Can the presence or absence of diabetes be predicted with a significant and reasonable accuracy rate using a few key details about the lifestyle of the patients?
- Q4: Can the model predict the results for the male population as accurately as it does for the female population?
- Q5: Can we build a smartphone app that can predict the presence or absence of diabetes after keying in some vital information about the patients?

The research paper is organized as follows. Section 2 presents the related work on the subject. Section 3 presents the research methodology. Section 4 presents the results and discussion. Section 5 concludes the paper and gives future directions. The link to the code is given before the References which are listed at the end.

Related work

In 2006, Giardina et al. applied a genetic algorithm and weighted k-nearest neighbors to analyze type 2 diabetes patients by means of the presence of coronary heart disease or its absence (Giardina, Azuaje, McCullagh, & Harper, 2006).

In 2013, Bennetts et al, used k-means clustering to find out the regional peak plantar pressure distributions in 819 diabetic feet (Bennetts, Owings, Erdemir, Botek, & Cavanagh, 2013). Khanna et al. performed classification on the diabetes dataset. Weights were assigned to various variables. They tried to find out if the class label for the data was low, medium or high risk (Khanna & Agarwal, 2013).

In 2016, Zheng et al. developed a semi-automatic framework using ML to improve the recall rate and keep the false positive rate low. They applied k-NN, Naïve Bayes, Decision Tree, Random Forest, SVM and LR (Zheng et al., 2017). Kanchan et al. studied PCA, which determines the least number of variables needed to optimize the precision of various ML algorithms (Kanchan & Kishor, 2016). Huang et al. used chi-squared automatic interaction detection (CHAID), Decision tree, KNN, recursive partitioning and regression tree and SVM to represent medical knowledge (Huang & Nashrullah, 2016).

In 2017, Osman et al. proposed and approach integrating support vector machine and K-means clustering techniques to predict diabetes (Osman & Aljahdali, 2017). Hashi et al. used Decision tree and KNN and proposed a system that calculates the accuracy of C4.5 which provided better accuracy for diabetes diagnosis (Hashi, Zaman, & Hasan, 2017). Khalil et al. used machine learning techniques to predict depression in diabetes patients (Khalil & Al-Jumaily, 2017).

In 2019, Cedeno-Moreno et al. used several machine learning techniques to find and generate models that can help detect and diagnose health problems like diabetes (Cedeno-Moreno & Vargas-Lombardo, 2019).

Research methodology

ML algorithms utilize a wide range of statistical, optimization and probabilistic methods to learn from their past experiences and determine useful and valid insights from big data. The data can be structured, unstructured as well as complex. ML algorithms have numerous usages in healthcare, finance, bioinformatics, data science, mechanical engineering, agriculture, smart cities, text categorization, intrusion detection, and junk e-mail filtering. A large variety of these uses are being implemented using supervised ML algorithms rather than unsupervised ML. We have also used the supervised ML algorithms in our work so we will focus on this variant only.

Supervised machine learning algorithms

In supervised ML, the machines are trained using labeled training data. Based on this learning, the machine makes a prediction for new or abstract data. Labeled data is the data that already consists of the correct output or result. The training data furnished to the machine is supposed to be working as the teacher,

which teaches the machine to make predictions accurately. It can be considered analogous to student learning under the supervision of the teacher. A structured definition of supervised machine learning can be 'A supervised machine learning is a process of furnishing the input data along with the accurate output result for the given data to the machine'. A supervised machine learning model aims to find a mapping function to map the provided input with the desired/needed output (Gramajo, Ballejos, & Ale, 2020).

Supervised ML is further classified into two types, Regression and classification. In our paper, we have used classification techniques, so we named the classification techniques in the next sections and illustrated them in later sections.

- Decision Tree
- K-Nearest Neighbor (KNN)
- Logistic Regression (LR)
- Naïve Bayes
- Random Forest
- Support Vector Machine (SVM)
- XGBoost Classifier

Flow diagram

Figure 1 depicts the steps taken to complete the study. The very first step is to load the dataset into the system. The next step is data processing. The data can be structured as well as unstructured. In this step, the data is transformed into a usable, easy to understand and desired form. The data is made more meaningful and informative. We will discuss this step in great detail in upcoming sections.

Once the data is brought into a standard and machine-readable format, we have divided the dataset into two parts, the training data and the test data. As the name suggests, the training data is used to give training to the system and the test data is used to test the accuracy of the system. The training algorithm is implemented and runs on Google Colaboratory (Carneiro et al., 2018). Once the training is completed, evaluation and validation are carried out for the model. After these steps, the model is ready to make predictions on the basis of the data provided. Then the test data is used to test the model. Further, the accuracy score can be calculated on the basis of the output of the test data as the outcomes are already known for the test data.

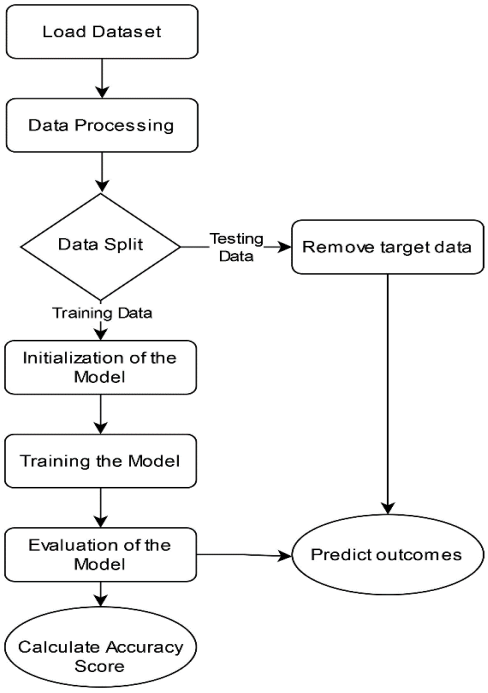


Figure 1. Flow Chart of the proposed methodology.

Data source and data extraction

The dataset used for the work was downloaded from the Kaggle website and is from the National Institute of Diabetes and Digestive and Kidney Diseases (Kaggle, 2018). The database helps in diagnostically predicting

whether a person is diabetic or not diabetic, based on the parameter included in the dataset. The database consists of a few constraints applied to it, some of them include, all the patients included in the dataset who are over the age of 21 years and are females and belong to the Pima Indian heritage. The diabetes dataset includes a number of different independent medical predictor variables and a single target variable. The variables are number of pregnancies, BMI, insulin level, blood pressure, skin thickness, glucose level, age and Diabetes Pedigree Function (DPF). And the target variable is OUTCOME. There was a total of 768 observations, of which 268 were diabetic (1) and 500 were non-diabetic (0).

Dataset analysis

In the analysis of the dataset, preprocessing steps were taken to prepare the data to be passed through the visualization and clustering techniques. The pre-processing stage consisted of four steps as shown in Figure 2.

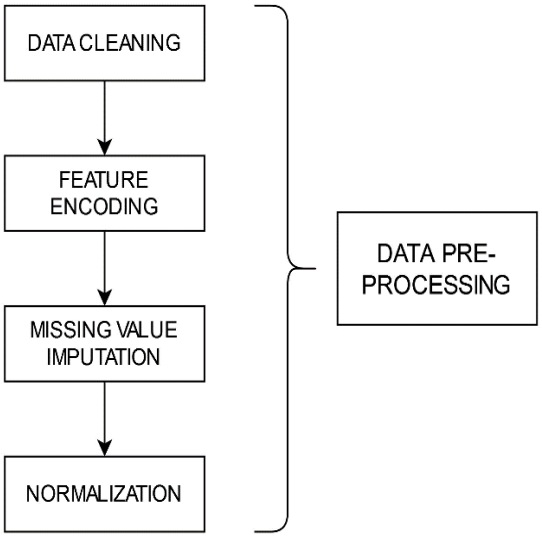


Figure 2. Data pre-processing steps.

In the Data cleaning step, all the irrelevant attributes of the dataset that are not required and do not have a significant impact on the output are removed. In our study, we have applied each supervised ML algorithm two times, once on the complete dataset, and the next in which the variables used were the number of pregnancies, BMI, blood pressure, glucose level and age. We aim to find the most accurate and suitable ML algorithm for the detection of diabetes in patients. Apart from this, we aim to build a model, in the form of a smartphone app, that can be used by anyone to predict his/her diabetic status after keying in some vital information. For that reason, in the second iteration, we have removed the variables, skin thickness, insulin and DPF, since the information is not easily available to the people at home. The data in the database can be of varied formats since they come from various tests and observations. To deal with this, we have the features encoding step. The data is transformed into a format in which the ML algorithm can be applied. Algorithms such as SVM are designed to work with numerical data and not on any other format of data. In the next step, missing value imputation, we try to identify missing values if any. We did not find any missing instances in the dataset. The next step is the normalization of the data. In this step, we organize the data to appear similar across all the records and fields (Pendharkar, 2005).

Diabetes dataset

The dataset used in our study consists of 768 distinct data tuples and each tuple has 9 variables. The variables include pregnancies, glucose, blood pressure, skin thickness, insulin, BMI, diabetes pedigree function, age and outcome. We have displayed a part of the dataset in Table 1.

Variables

In this section, we have discussed the variables of the dataset:

- Pregnancies;

This field talks about the number of times the patient was pregnant;

- Glucose.

The glucose test was performed on the patients using a glucometer. The standard glucose level for a person above the age of 18 years is depicted in Table 2.

- Blood Pressure (BP)

BP is measured as the pressure developed in the arteries when the heart is at rest between two consecutive heartbeats. This is the small amount of time when the heart is filled with blood and oxygen is added to it. Table 3 depicts the various BP levels for a person over the age of 18 years.

- Skin Thickness

Skin thickness is measured from the triceps skin fold thickness in (mm). It gives information about the total amount of body fat present. This variable also talks about the fat reserves of the body and protein reserves. Table 4 depicts the standard skin thickness measurements.

- Insulin

Insulin is a naturally produced hormone secreted by the pancreas that helps in moving the sugar present in the blood, also called blood glucose, from the blood in the arteries to the cells. It is the 2-hour serum insulin measured in (mu U mL-1). Insulin therapy is used when the pancreas is not able to produce any or enough insulin which helps in keeping the blood glucose level under control.

- BMI

BMI is the Body mass index which is calculated as

$$BMI = \frac{weight\ in\ kg}{(height\ in\ m)^2}$$

Table 1. A part of the diabetes dataset.

	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	DPF	Age	Outcome
0	5	116	74	0	0	25.6	0.201	30	0
1	3	78	50	32	88	31	0.248	26	1
2	10	115	0	0	0	35.3	0.134	29	0
3	2	197	70	45	543	30.5	0.158	53	1
4	8	125	96	0	0	0	0.232	54	1
5	4	110	92	0	0	37.6	0.191	30	0
6	10	168	74	0	0	38	0.537	34	1
7	10	139	80	0	0	27.1	1.441	57	0
8	1	189	60	23	846	30.1	0.398	59	1
9	5	166	72	19	175	25.8	0.587	51	1

Table 2. Blood glucose chart.

BP Category	Diastolic	Systolic
Normal	< 80	< 120
Elevated	< 80	120-129
High BP-Hypertension Stage 1	80-89	130-139
High BP-Hypertension Stage 2	90 and above	140 and above
High BP-Hypertension Stage 3	120 and above	180 and above

Table 3. BP level chart.

BP Category	Diastolic	Systolic
Normal	< 80	< 120
Elevated	< 80	120-129
High BP-Hypertension Stage 1	80-89	130-139
High BP-Hypertension Stage 2	90 and above	140 and above
High BP-Hypertension Stage 3	120 and above	180 and above

Table 4. Skin thickness measurements.

	Normal	Obese
Men	2.5 or 20% fat	>20mm
Women	18 or 30% fat	>30mm

The BMI index of people over the age of 18 is depicted in Table 5.

- Diabetic Pedigree Function

It is the function that calculates the likelihood of diabetes in a patient based on his/her family history. The amount of genetic influence provides an idea of the patients' hereditary risks for the onset of diabetes.

- Age

Age was calculated in years.

- Outcome

In our data set, the outcome variable consisted of 2 classes, class 1 and class 0. It symbolizes whether the patient is diabetic or not. '1' resembles the patient being 'diabetic' and '0' symbolizes that the patient is 'not diabetic'. As depicted in Table 6.

We have summarized the dataset variable in Table 7 below.

Table 5. BMI index.

BMI	Classification
< 18	Highly underweight
18-20	Underweight
20-25	Healthy weight
25-30	Overweight
30-35	Obese (Class I)
35-40	Obese (Class II)

Table 6. Outcome.

Outcome	Inference
0	Diabetic
1	Non-Diabetic

Table 7. Features of the Diabetic dataset.

Variable	Details
Pregnancies	Total number of times pregnant
Glucose	Blood glucose concentration, oral glucose tolerance test after 2 hours
Blood pressure	Diastolic blood pressure (mm Hg)
Skin thickness	Skin fold thickness of triceps (mm)
Insulin	2 hours serum insulin (mu U mL^{-1})
BMI	Body mass index calculated as weight in kg (height in m)^2
Diabetes pedigree function	Calculates the likelihood of diabetes in a patient based on his/her family history
Age	Age of the patient in years
Outcome	Whether the patient is diabetic or not

The descriptive analysis of the dataset is displayed in Table 8 and Table 9. In Table 8 we have displayed the mean, standard deviation (SD), mean ± SD, Minimum, Maximum, and median values of the dataset for each variable. The average for pregnancies was found to be 3.84 ± 3.70, for glucose 120.89 ± 31.97, for blood pressure 69.1 ± 19.35, for skin thickness 20.54 ± 15.95, for insulin 79.80 ± 115.24, for BMI 32.0 ± 7.88, for diabetes pedigree function 0.47 ± 0.33 and the average for age was 33.24 ± 11.76. The median was also calculated as $'Median = Max - Min'$ for the dataset. The median for pregnancies was calculated as 17, glucose as 199, blood pressure as 122, skin thickness as 99, insulin as 846, BMI as 67.1, diabetes pedigree function as 2.342 and age as 60.

The quartile function was also calculated for the dataset. It helps in dividing the data into quarters. First, the data is sorted into ascending order and then divided into quarters. Quartiles divide the data into 4 equal parts. There are four quartiles. 1st quartile is 0.25, 2nd is the 0.50 and the 3rd is the 0.75 and the 4th is the 1.0 ie the complete dataset.

The first quartile (0.25 or Q1) is calculated as $Q1 = \left(\frac{n+1}{4}\right)^{th}$ term. The second quartile (0.50 or Q2) is calculated as $Q2 = \left(\frac{n+1}{2}\right)^{th}$ term. The third quartile (0.75 or Q3) is calculated as $Q3 = 3\left(\frac{n+1}{4}\right)^{th}$ term. The results are summarized in Table 9.

The correlation between each pair of the variables was calculated and displayed in a correlation matrix in Table 10 and depicted graphically in Figure 3a. A correlation matrix is a table depicting coefficients of correlation between two variables. Each cell of the table displays the correlation between the two variables. It is used to summarize the data, in the form of input for an advanced analysis or as a diagnostic for the same. The more the value is closer to 1, the higher the correlation between the two variables.

For the second iteration 'skin thickness', 'insulin' and 'diabetes pedigree function' were removed from the dataset. The correlation matrix for the same is displayed in Table 11 and graphically in Figure 4.

Heatmaps are a tool for representing the data in two dimensions using colors. It provides colored visualizations of the data and shows the magnitude of the phenomenon with various colors.

Table 8. Descriptive statistics for quantitative variables (mathematics).

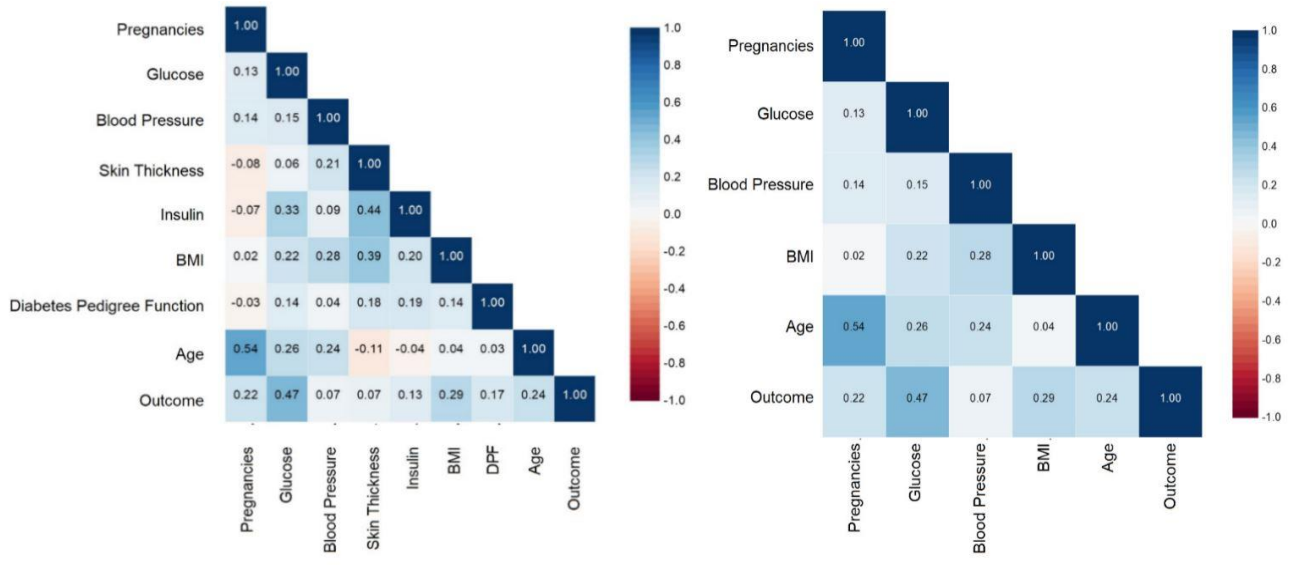
	Mean	SD	Mean ± SD	Min	Max	Median (Max-Min)
Pregnancies	3.84505	3.36958	3.84 ± 3.70	0	17	17
Glucose	120.895	31.9726	120.89 ± 31.97	0	199	199
Blood Pressure	69.1055	19.3558	69.1 ± 19.35	0	122	122
Skin Thickness	20.5365	15.9522	20.54 ± 15.95	0	99	99
Insulin	79.7995	115.244	79.80 ± 115.24	0	846	846
BMI	31.9926	7.88416	32.0 ± 7.88	0	67.1	67.1
DPF	0.47188	0.33133	0.47 ± 0.33	0.078	2.42	2.342
Age	33.2409	11.7602	33.24 ± 11.76	21	81	60

Table 9. Descriptive statistics for variables: Quartile function (mathematics).

	0.25	0.5	0.75
Pregnancies	1	3	6
Glucose	99	117	140.25
Blood Pressure	62	72	80
Skin Thickness	0	23	32
Insulin	0	30.5	127.25
BMI	27.3	32	36.6
DPF	0.24375	0.3725	0.62625
Age	24	29	41

Table 10. Correlation matrix of all variables.

	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	DPF	Age	Outcome
Pregnancies	1	0.12946	0.14128	-0.0817	-0.0735	0.01768	-0.0335	0.54434	0.2219
Glucose	0.12946	1	0.15259	0.05733	0.33136	0.22107	0.13734	0.26351	0.46658
Blood Pressure	0.14128	0.15259	1	0.20737	0.08893	0.28181	0.04127	0.23953	0.06507
Skin Thickness	-0.0817	0.05733	0.20737	1	0.43678	0.39257	0.18393	-0.114	0.07475
Insulin	-0.0735	0.33136	0.08893	0.43678	1	0.19786	0.18507	-0.0422	0.13055
BMI	0.01768	0.22107	0.28181	0.39257	0.19786	1	0.14065	0.03624	0.2927
DPF	-0.0335	0.13734	0.04127	0.18393	0.18507	0.14065	1	0.03356	0.17384
Age	0.54434	0.26351	0.23953	-0.114	0.04216	0.03624	0.03356	1	0.23836
Outcome	0.2219	0.46658	0.06507	0.07475	0.13055	0.2927	0.17384	0.23836	1



a. For all variables b. For 2nd iteration

Figure 3. Heatmap of correlation.

Table 11. Correlation matrix for 2nd iteration.

	Pregnancies	Glucose	Blood Pressure	BMI	Age	Outcome
Pregnancies	1	0.12946	0.14128	0.01768	0.54434	0.2219
Glucose	0.12946	1	0.15259	0.22107	0.26351	0.46658
Blood Pressure	0.14128	0.15259	1	0.28181	0.23953	0.06507
BMI	0.01768	0.22107	0.28181	1	0.03624	0.2927
Age	0.54434	0.26351	0.23953	0.03624	1	0.23836
Outcome	0.2219	0.46658	0.06507	0.2927	0.23836	1

```
print(X_train.shape)
print(X_test.shape)
print(y_train.shape)
print(y_test.shape
(506, 8)
(218, 8)
(506,)
(218,)
```

Figure 4. Code for the shape of the train and test dataset.

Results and discussion

The dataset which was used for the study was a structured data set. We have applied the classification techniques that categorize the data into 2 or more classes, as required or defined by the user. The prime objective of a classifier is to recognize the given/input data to the class that most suitably describes the new data.

70% of the data was used for training the classifier model. The left 30% data was utilized to test the classifier model (506, 218). Next, we have discussed the main steps that were taken to build the classification model.

Step I: Initialization of the Model: The first and foremost step is to initialize the classifier model to be used.

Step II: Training the Model: The next step is to train the model. All the classifiers used in the study use a fitness function $fit(X, y)$ to train the model on the given training data X and training label y. A confusion matrix is plotted for the same.

Step III: Evaluating the Model: The model is then evaluated by giving the test data whose outcome is already known. The accuracy score is calculated for the same, for two iterations, using the below formula.

$$Accuracy = \frac{(True\ Positive + True\ Negative)}{Total\ Population}$$

Average Precision is also calculated for each model for both iterations:

$$Average\ Precision = \sum_{i=1}^{n} \frac{Precision_i}{n}$$

where:

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Recall is also calculated for each model for both iterations:
where:

$$Recall = \frac{True\ Positive}{(True\ Positive + False\ Negative)}$$

Further, the F1 score is also calculated. F1 score is the weighted average of Precision and Recall:

$$F1\ score = \frac{2\ \boldsymbol{x}\ Precision\ \boldsymbol{x}\ Recall}{Precision + Recall}$$

Step IV: Predicting the Target: The classifier model is then given unlabeled data and it returns the prediction label for the data.

Legend:

True Positive: Number of Accurate Predictions Labeling the Occurrence as Positive;

True Negative: Number of Accurate Predictions Labeling the Occurrence As Negative;

Total Population: Total Number of Predictions Made;

Recall: How often is the prediction correct when the occurrence is positive?

Below we have discussed the classifier models and evaluated their results. We have used four matrices to evaluate the same, Confusion matrix, Accuracy, Precision, Recall and F1 score. The results are displayed in two columns below, the left column depicts the 1st iteration with all the variables, whilst the right column depicts the 2nd iteration with commonly known variables.

Decision tree

Definition: A decision tree produces a sequence of rules from the given set of labeled data that are further used to classify the data. The code for the Decision tree classifier and its output is given in Figure 5

Advantage: It is easy to understand and visualize. It needs zero to less preparation of the data and can handle categorical as well as numerical data.

Disadvantage: It can build complex trees that do not predict very well. It can be very unstable since even a small variable can lead to the generation of a completely different tree (Cruz & Wishart, 2007).

K-Nearest neighbor

Definition: KNN is a lazy learning method. It does not construct an input model but saves the training data and computes the best result by a majority vote of the nearest k neighbors. The code for K-Nearest Neighbor and its output is given in Figure 6.

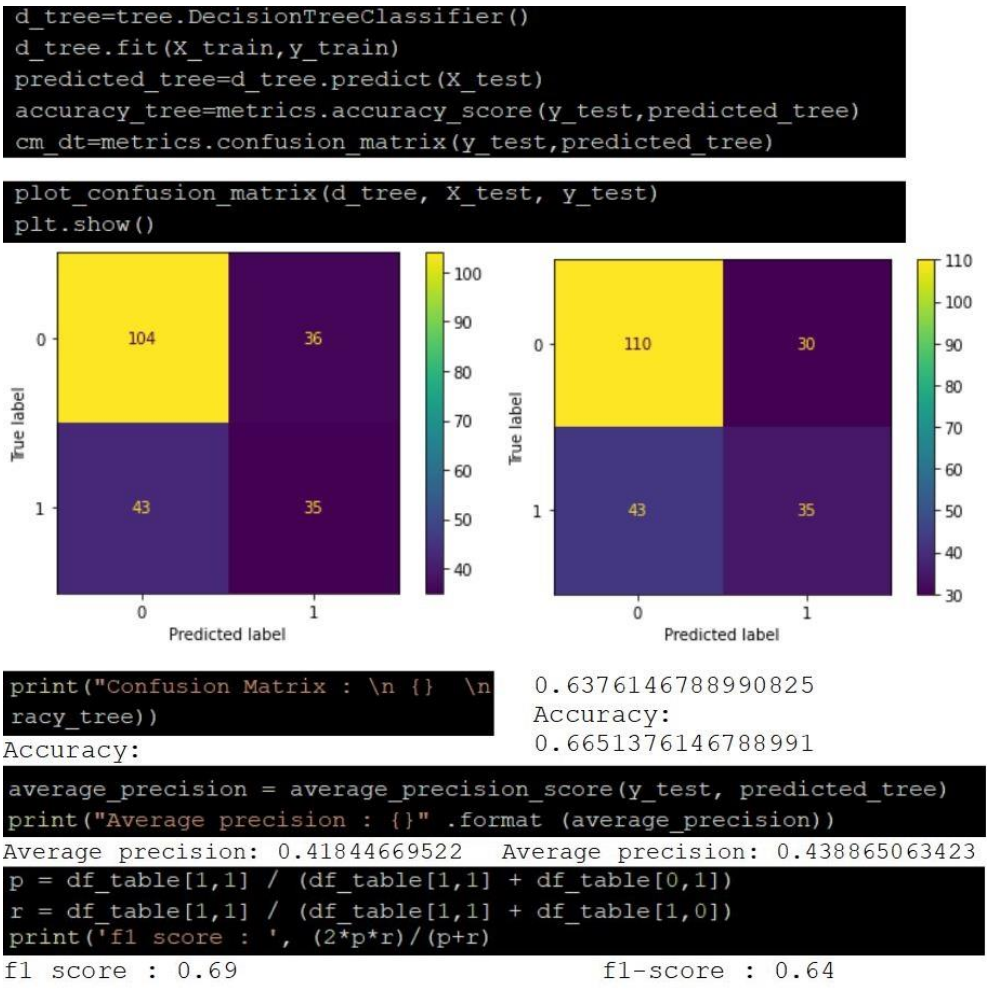


Figure 5. Code for Decision tree classifier and its output.

Advantage: It is very simple, resistant to noisy data and very effective if training data is large.

Disadvantage: Determination of the value of K is very costly (Cover & Hart, 1967).

Logistic regression

Definition: A logistic function is used to model the probabilities describing all the possible outcomes. The code for Logistic Regression and its output is given in Figure 7.

Advantage: It is very useful in finding the impact of a variable on the outcome variable.

Disadvantage: Works only on binary data. Missing values can cause problems (Hosmer, Lemeshow, & Sturdivant, 2013).

Naïve bayes

Definition: It is based on Bayes' theorem. An assumption is made that each variable is independent of every other variable. The code for Naïve Bayes and its output is given in Figure 8.

Advantage: Small training data also make correct predictions. It is very fast.

Disadvantage: It is a bad estimator (Rish, 2001).

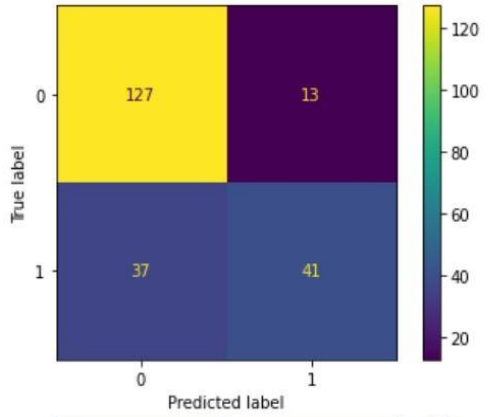
Random forest

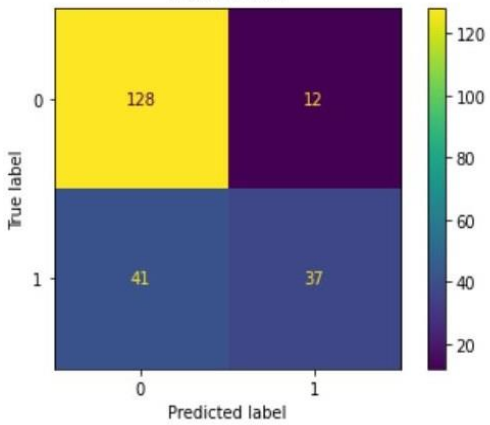
Definition: Random forests use a decision tree and creates multiple trees on a variety of sub-samples of the dataset and uses the average to increase the prediction accuracy and reduce over-fitting. The code for Random Forest and its output is given in Figure 9.

Advantage: Over-fitting reduction is more accurate than a decision tree.

Disadvantage: It is a very complex algorithm and difficult to implement. It is slow also (Breiman, 2001).

```
knn=KNeighborsClassifier(n_neighbors=11)
knn.fit(X_train,y_train)
predicted_knn=knn.predict(X_test)
cm_knn=metrics.confusion_matrix(y_test,predicted_knn)
accuracy_knn=metrics.accuracy_score(y_test,predicted_knn)
```






```
Accuracy: 0.7706422018348624
Accuracy: 0.7568807339449541
```

```
Average Precision: 0.5688225864067 Average Precision: 0.546262824113
f1 score: 0.77                     f1-score: 0.76
```

Figure 6. Code for K-Nearest Neighbor and its output.

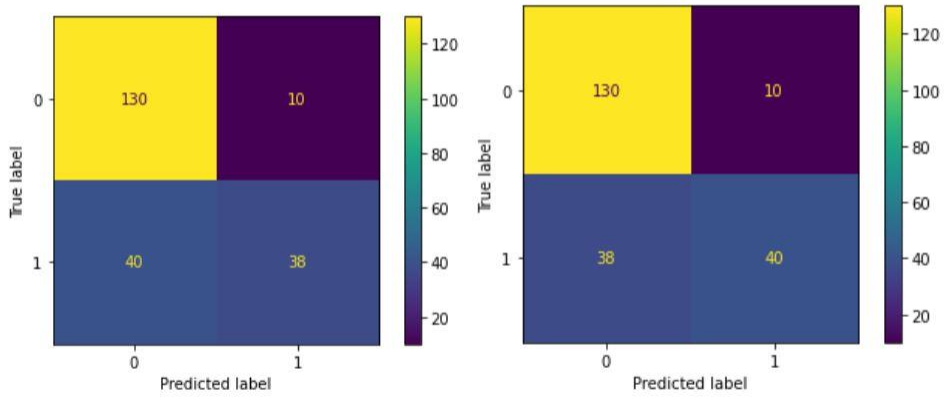
Support vector machine

Definition: SVM represents the training data as points in categories with the maximum gaps possible. New data is then mapped into the categories and classified into the category nearest to the data. The code for the Support vector machine and its output is given in Figure 10.

Advantage: Effective in multidimensional spaces and is memory efficient.

Disadvantage: Five-fold cross-validation is very expensive (Joachims, 1998).

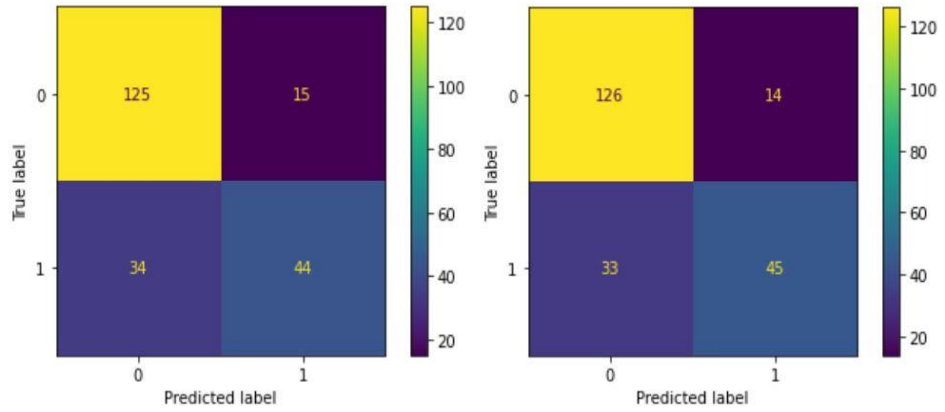
```
logisticRegr= LogisticRegression()
logisticRegr.fit(X_train,y_train)
predict_lg = logisticRegr.predict(X_test)
accuarcy_lg=metrics.accuracy_score(y_test,predict_lg)
cm_lg=metrics.confusion_matrix(y_test,predict_lg)
```




```
Accuracy: 0.771512201834862        0.7798165137614679
Accuracy:
Average     Precision:     0.5691699992159     Average     Precision:
0.5845683368619
f1 score: 0.77                              f1-score: 0.78
```

Figure 7. Code for Logistic Regression and its output.

```
naive = GaussianNB()
naive.fit(X_train,y_train)
predicted_naive=naive.predict(X_test)
cm_naive=metrics.confusion_matrix(y_test,predicted_naive)
accuracy_naive=metrics.accuracy_score(y_test,predicted_naive)
```




```
Accuracy: 0.77522935779816         Accuracy: 0.78440366972477
Average     Precision:     0.5766499607271     Average     Precision:
0.5914022224083
f1 score: 0.78                              f1-score: 0.78
```

Figure 8. Code for Naïve Bayes and its output.

XGBoost classifier

Definition: It uses the Gradient Boosting framework. It provides a parallel tree boosting (also known as GBDT, GBM) that can solve a variety of prediction problems accurately and in very little time. The code for the XGBoost Classifier and its output is given in Figure 11.

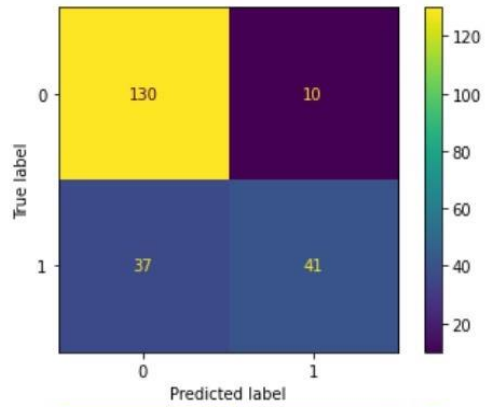
Advantage: The same code running on a system is very likely to run on every other environment (Hadoop, SGE, MPI). It can handle large volumes of data easily.

Disadvantage: The performance of the XGBoost Classifier decreases on sparse and unstructured data. The overall method is not scalable.

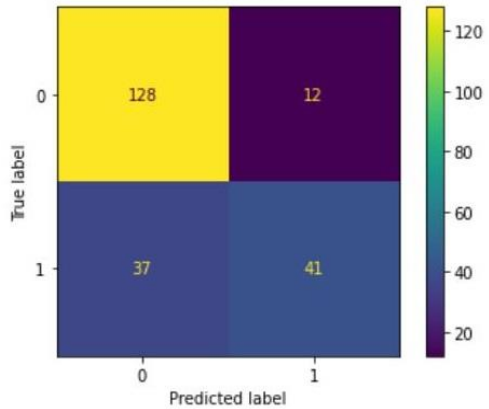
Evaluation of the classification models

In this section, we have summarized all the observations and results in an easy-to-understand layout. Table 12 summarizes and compares the accuracy and precision of iteration I and iteration II. It is evident from the table that the Random Forest classifier is the best classifier for the detection of diabetes in patients. However, the data set used in iteration II brings out a better accuracy for the random forest classifier. The RF classifier of iteration II outperformed that of iteration I by a difference of 0.010825688. Figure 12 depicts the comparison of the accuracy and precision of iteration I and iteration II. It is evident that iteration II outperforms iteration I in almost every classification algorithm.

```
rfm = RandomForestClassifier(n_estimators=
70, oob_score=True, n_jobs=-1, random_state=
101, max_features = None, min_samples_leaf = 30)
rfm = rfm.fit(X_train, y_train)
pred_forest=rfm.predict(X_test)
accuracy_forest=metrics.accuracy_score(y_test,pred_forest)
cm_rf=metrics.confusion_matrix(y_test,pred_forest)
```



Accuracy: 0.7844036697247706



Accuracy: 0.7952293577981652

Average Precision: 0.5922989285105 Average Precision: 0.5763527338739

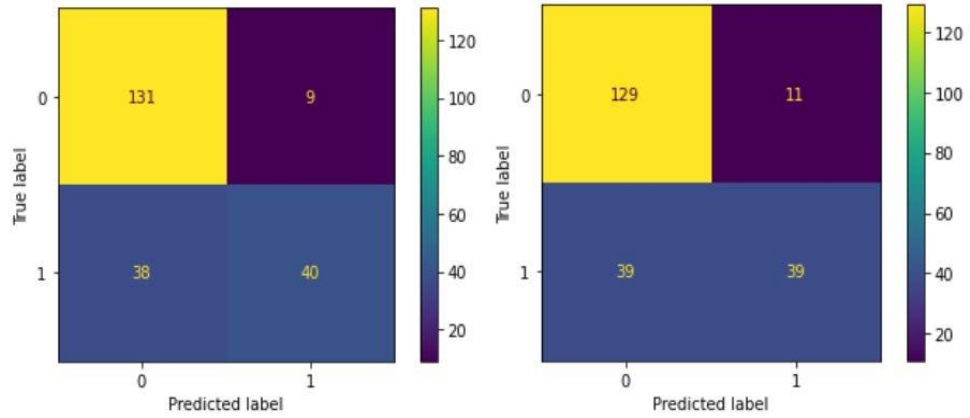
f1 score: 0.78 f1-score: 0.78

Figure 9. Code for Random Forest and its output.

In Table 8 we have summarized the accuracy, recall and F1 score data for both iterations. Figure 13 depicts the data from Table 13 in a graphical form. Again, it is evident that iteration II outperformed iteration I in all the classification algorithms except the Decision tree algorithm.

Table 14 shows the instances classified correctly and incorrectly by the various classification algorithms. The test data was taken to be 30% of the total data which was 218 unique data items. The table describes the trained performance of the data model.

```
scaler = StandardScaler()
scaler.fit(X)
standardized_data = scaler.transform(X)
X = standardized_data
Y = df['Outcome']
SupportV = svm.SVC(kernel='linear')
SupportV = SupportV.fit(X_train, y_train)
SupportV_pred = SupportV.predict(X_train)
accuracy_SupportV = accuracy_score(SupportV_pred, y_train)
cm_SupportV=metrics.confusion_matrix(y_train, SupportV_pred)
```




```
Accuracy:                          Accuracy:
0.7608695652173914                 0.7707509881422925
Average    Precision:    0.52605422239626    Average    Precision:
0.542593190945
f1 score: 0.76                     f1-score: 0.77
```

Figure 10. Code for Support vector machine and its output.

Table 12. Comparison Matrix of accuracy and precision of iteration I and iteration II.

Classification Algo	Iteration 1		Iteration II	
	Accuracy	Precision	Accuracy	Precision
Decision Tree	0.637614679	0.418446695	0.665137615	0.438865063
K-Nearest Neighbour	0.770642202	0.568822586	0.756880734	0.546262824
Logistic Regression	0.770642202	0.569169999	0.779816514	0.584568337
Naïve Bayes	0.775229358	0.576649961	0.78440367	0.591402222
Random Forest	0.78440367	0.592298929	0.795229358	0.576352734
Support Vector Machine	0.760869565	0.526054222	0.770750988	0.542593191
XGBoost	0.75974026	0.543441055	0.733766234	0.5064711344

Table 13. Comparison Matrix of accuracy, recall and F1 score of iteration I and iteration II.

Classification Algo	Iteration 1			Iteration II		
	Accuracy	Recall	F1 Score	Accuracy	Recall	F1 Score
Decision Tree	63.76%	0.65	0.69	66.51%	0.63	0.64
K-Nearest Neighbour	77.06%	0.73	0.77	75.69%	0.74	0.76
Logistic Regression	77.15%	0.71	0.77	77.98%	0.75	0.78
Naïve Bayes	77.52%	0.73	0.78	78.44%	0.76	0.78
Random Forest	78.44%	0.72	0.78	79.52%	0.75	0.79
Support Vector Machine	76.09%	0.70	0.76	77.08%	0.74	0.77
XGBoost	75.97%	0.71	0.76	73.37%	0.68	0.73

Table 14. Random sampling results.

Classification Algo	Iteration 1		Iteration II	
	Instances classified correctly	Instances classified incorrectly	Instances classified correctly	Instances classified incorrectly
Decision Tree	139	79	145	73
K-Nearest Neighbour	168	50	165	53
Logistic Regression	168	50	170	48
Naïve Bayes	169	49	171	47
Random Forest	171	47	173	45
Support Vector Machine	163	52	167	49
XGBoost	165	53	160	58

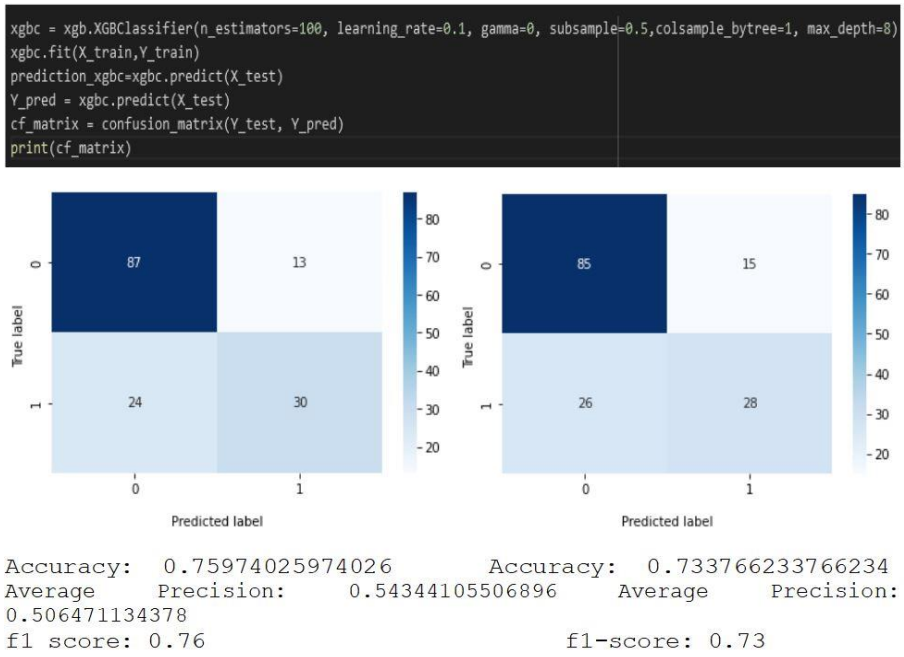


Figure 11. Code for XGBoost Classifier and its output.

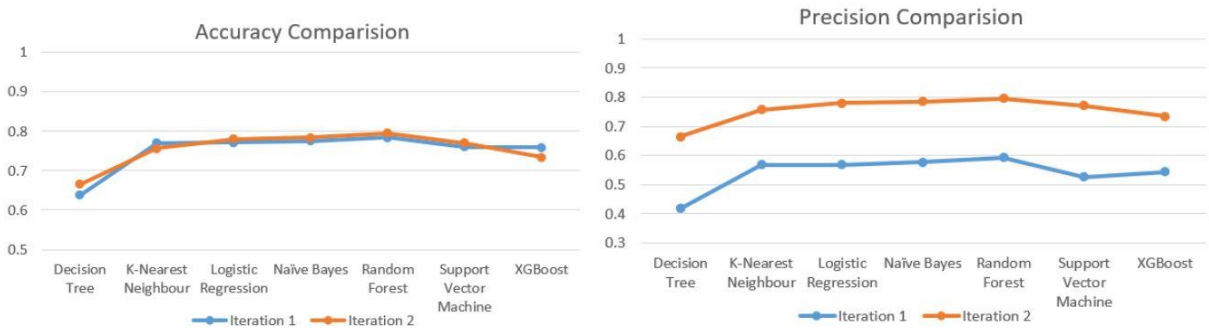


Figure 12. Accuracy and precision comparison for iteration I and iteration II.

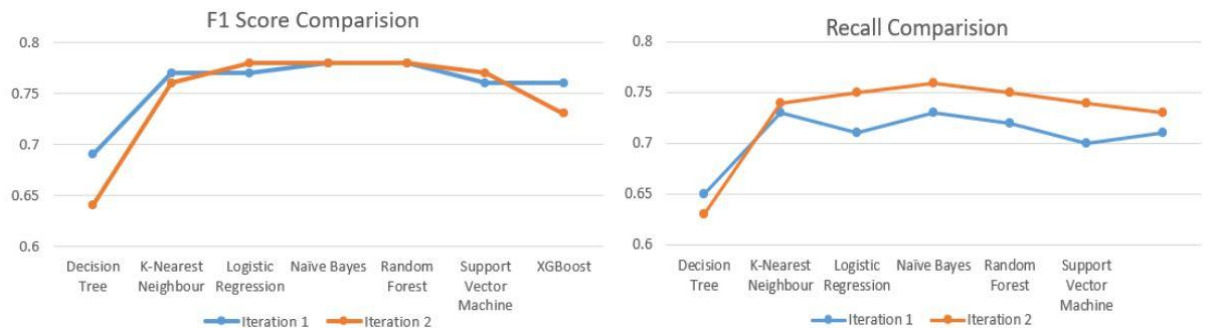


Figure 13. F1 score and recall comparison for iteration I and iteration II.

Conclusion and future work

Machine learning is penetrating almost every sector of civilization. It is helping the healthcare sector in a number of ways now. The paper provides the performance overview of the 7 distinct supervised ML classification algorithms for the prediction of diabetes in females. The research work can be used as an aid in further research in the selection of the most appropriate supervised machine learning algorithm for their future endeavors. We found out that the Random Forest algorithm was the best-fit algorithm in the detection of diabetes in patients. It made 171 correct predictions out of a total of 218 instances. Out tweaks and modifications increased the number to 173 correct predictions which counted to be 1.08% better than its counterpart.

We plan to improve our work in several directions, we can add labels and variables which will be able to differentiate between T1DM and T2DM. Expanding or reducing the measure and impact of the variables may improve the accuracy of predicting diabetes. The current model is only suitable for the female population. We will further work on developing a smartphone-based app that will take some inputs regarding the disease and predict the outcomes which will help the population immensely.

Acknowledgment

The authors would like to thank the Department of Science and Technology- Promotion of University Research and Scientific Excellence (DST-PURSE) and Jamia Hamdard, New Delhi, India for the research fellowship.

References

Alam, M., Khan, I. R., Siddiqui, S. A., Wiquar, R., & Anwar, H. (2021). IoT and AI as key enabler of growth of smart cities. In *Proceedings of the 2nd International Conference on ICT for Digital, Smart, and Sustainable Development*. New Delhi, IN. DOI: https://doi.org/10.4108/eai.27-2-2020.2303467

Banday, M. Z., Sameer, A. S., & Nissar, S. (2020). Pathophysiology of diabetes: an overview. *Avicenna Journal of Medicine, 10*(4), 174-188. DOI: https://doi.org/10.4103/ajm.ajm_53_20

Bennetts, C. J., Owings, T. M., Erdemir, A., Botek, G., & Cavanagh, P. R. (2013). Clustering and classification of regional peak plantar pressures of diabetic feet. *Journal of Biomechanics, 46*(1), 19-25. DOI: https://doi.org/10.1016/j.jbiomech.2012.09.007

Breiman, L. (2001). Random forests. *Machine Learning, 45*, 5-32. DOI: https://doi.org/10.1023/A:1010933404324

Carneiro, T., Nobrega, R. V. M., Nepomuceno, T., Bian, G.-B., Albuquerque, V. H., & Rebouças Filho, P. P. (2018). Performance analysis of google colaboratory as a tool for accelerating deep learning applications. *IEEE Access, 6*, 61677-61685. DOI: https://doi.org/10.1109/access.2018.2874767

Cedeno-Moreno, D., & Vargas-Lombardo, M. (2019). Application of machine learning with supervised classification algorithms: in the context of health. In *7th International Engineering, Sciences and Technology Conference* (p. 613-618). Cidade do Panamá, PA: IEEE.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory, 13*(1), 21-27. DOI: https://doi.org/10.1109/TIT.1967.1053964

Cruz, J. A., & Wishart, D. S. (2007). Applications of machine learning in cancer prediction and prognosis. *Cancer Informatics, 11*(2), 59-77.

Dua, P., Doyle, F. J., & Pistikopoulos, E. N. (2006). Model-based blood glucose control for type 1 diabetes via parametric programming. *IEEE Transactions on Biomedical Engineering, 53*(8), 1478-1491. DOI: https://doi.org/10.1109/TBME.2006.878075

Genuth, S. M., Palmer, J. P., & Nathan, D. M. (2018). Classification and diagnosis of diabetes. In *Diabetes in America* (3rd ed.). Bethesda, MD: National Institute of Diabetes and Digestive and Kidney Diseases.

Giardina, M., Azuaje, F., McCullagh, P., & Harper, R. (2006). A supervised learning approach to predicting coronary heart disease complications in type 2 diabetes mellitus patients. In *IXth IEEE Symposium on BioInformatics and BioEngineering* (p. 325-331). Arlington, VA: IEEE.

Gramajo, M., Ballejos, L., & Ale, M. (2020). Seizing requirements engineering issues through supervised learning techniques. *IEEE Latin America Transactions, 18*(7), 1164-1184. DOI: https://doi.org/10.1109/TLA.2020.9099757

Hashi, E. K., Zaman, M. S. U., & Hasan, M. R. (2017). An expert clinical decision support system to predict disease using classification techniques. In *International Conference on Electrical, Computer and Communication Engineering* (p. 396-400).. Cox's Bazar, BD: IEEE.

Hosmer Jr., D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression*. New Yor, USA: John Wiley & Sons.

Huang, Y.-P., & Nashrullah, M. (2016). SVM-based decision tree for medical knowledge representation. In *International Conference on Fuzzy Theory and Its Applications* (p. 1-6). Taichung, TW: IEEE.

Islam, M. S., Qaraqe, M. K., Belhaouari, S. B., & Abdul-Ghani, M. A. (2020). Advanced techniques for predicting the future progression of type 2 diabetes. *IEEE Access, 8*, 120537-120547. DOI: https://doi.org/10.1109/ACCESS.2020.3005540

Joachims, T. (1998). *Making large-scale support vector machine learning practical*. Dortmund, DE: Universität Dortmund.

Kaggle. (2018). *Pima Indians diabetes database*. Retrieved from https://www.kaggle.com/uciml/pima-indians-diabetes-database

Kanchan, B. D., & Kishor, M. M. (2016). Study of machine learning algorithms for special disease prediction using principal of component analysis. In *International Conference on Global Trends in Signal Processing, Information Computing and Communication* (p. 5-10). Jalgaon, India: IEEE.

Khalil, R. M., & Al-Jumaily, A. (2017). Machine learning based prediction of depression among type 2 diabetic patients. In *12th International Conference on Intelligent Systems and Knowledge Engineering* (p. 1-5). Nanjing, CN: IEEE.

Khanna, S., & Agarwal, S. (2013). An integrated approach towards the prediction of likelihood of diabetes. In *International Conference on Machine Intelligence and Research Advancement* (p. 294-198). Katra, IN: IEEE.

Krasteva, A., Panov, V., Krasteva, A., Kisselova, A., & Krastev, Z. (2011). Oral cavity and systemic diseases—diabetes mellitus. *Biotechnology & Biotechnological Equipment, 25*(1), 2183-2186. DOI: https://doi.org/10.5504/BBEQ.2011.0022

Lonappan, A., Bindu, G., Thomas, V., Jacob, J., Rajasekaran, C., & Mathew, K. T. (2007). Diagnosis of diabetes mellitus using microwaves. *Journal of Electromagnetic Waves and Applications, 21*(10), 1393-1401. DOI: https://doi.org/10.1163/156939307783239429

Osman, A. H., & Aljahdali, H. M. (2017). Diabetes Disease Diagnosis Method based on Feature Extraction using K-SVM. *International Journal of Advanced Computer Science and Applications, 8*(1), 236-244. DOI: https://doi.org/10.14569/IJACSA.2017.080130

Parveen, R., Sehar, N., Bajpai, R., & Agarwal, N. B. (2020). Association of diabetes and hypertension with disease severity in COVID-19 patients: a systematic literature review and exploratory meta-analysis. *Diabetes Research and Clinical Practice, 166*, 108295. DOI: https://doi.org/10.1016/j.diabres.2020.108295

Pendharkar, P. C. (2005). A data envelopment analysis-based approach for data preprocessing. *IEEE Transactions on Knowledge and Data Engineering, 17*(10), 1379-1388. DOI: https://doi.org/10.1109/TKDE.2005.155

Rish, I. (2001). An empirical study of the naive bayes classifier.In *IJCAI 2001 workshop on empirical methods in artificial intelligence* (p. 41-46). New York, NY: IBM.

Zheng, T., Xie, W., Xu, L., He, X., Zhang, Y., You, M., ... Chen, Y. (2017). A machine learning-based framework to identify type 2 diabetes through electronic health records. *International Journal of Medical Informatics, 97*, 120-127. DOI: https://doi.org/10.1016/j.ijmedinf.2016.09.014