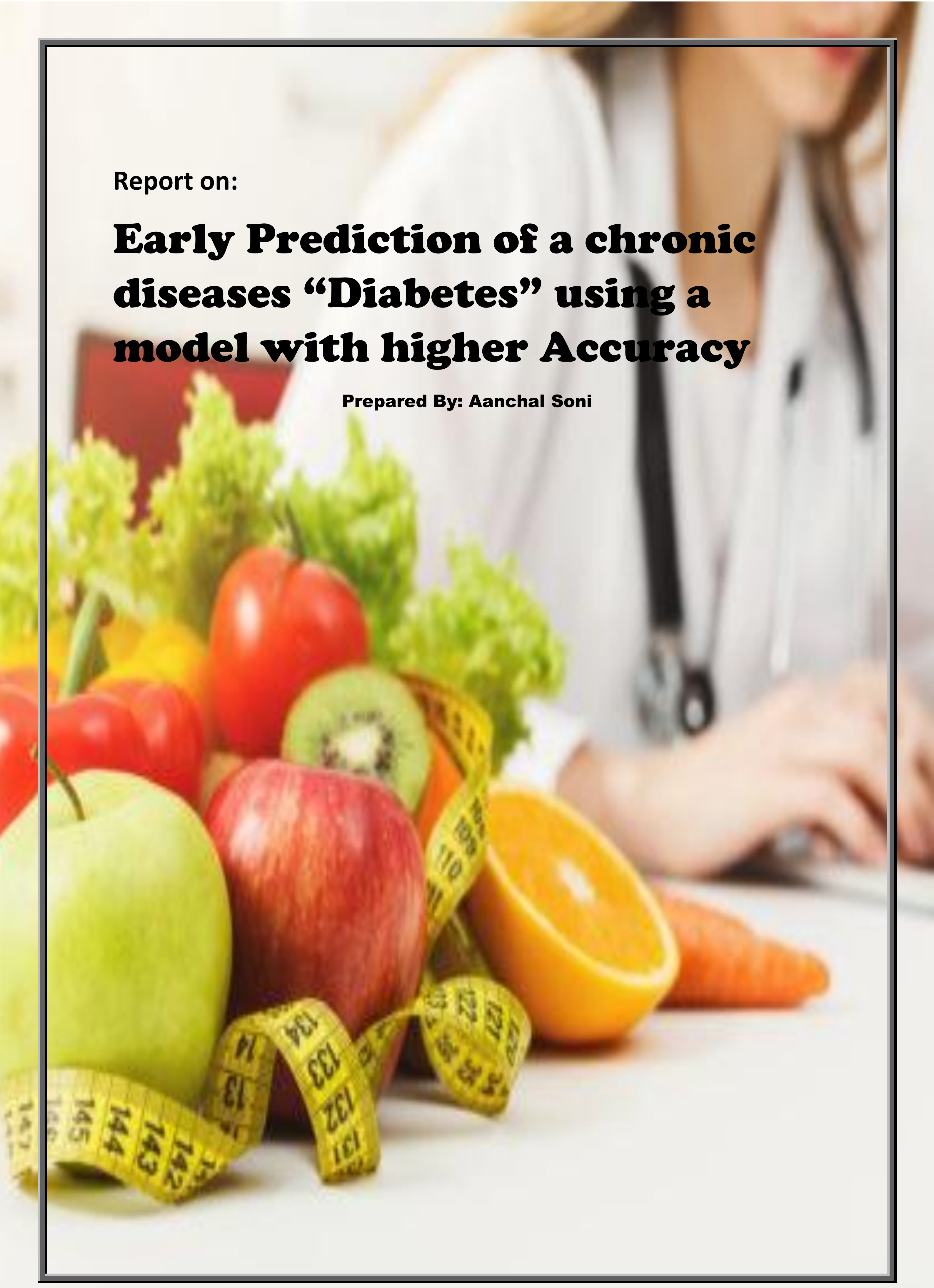
Report on:

Early Prediction of a chronic diseases “Diabetes” using a model with higher Accuracy

Prepared By: Aanchal Soni



ABSTRACT:

Diabetes is an illness caused because of high glucose level in a human body. Diabetes should not be ignored if it is untreated then Diabetes may cause some major issues in a person like: heart related problems, kidney problem, blood pressure, eye damage and it can also affect other organs of human body. Diabetes can be controlled if it is predicted earlier. To achieve this goal this project work we will do early prediction of Diabetes in a human body or a patient for a higher accuracy through applying, Various Machine Learning Techniques. Machine learning techniques Provide better result for prediction by constructing models from datasets collected from patients. In this work we will use Machine Learning Classification and ensemble techniques on a dataset to predict diabetes. Which is K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Naïve Bayes (NB) and Random Forest (RF). The accuracy is different for every model when compared to other models. The Project work gives the higher accuracy model shows that the model is capable of predicting diabetes effectively.

Table of Contents:

- Introduction
- Existing Methods
- Proposed Method with architecture
- Methodology
- Implementation
- Result & Conclusion

INTRODUCTION:

Health care systems are merely designed to meet the needs of increasing population globally. People around the globe are affected with different types of deadliest diseases. Among the different types of commonly existing diseases, diabetes is a major cause of blindness, kidney failure, heart attacks, etc. Health care monitoring systems for different diseases and symptoms are available all around the world. The rapid development in the fields of Information and Communication Technologies made remarkable improvements in health care systems. Various Machine Learning algorithms are proposed which automates the working model of health care systems and enhances the accuracy of disease prediction. This work proposes the novel implementation of machine learning algorithms in for diabetes prediction. The results show that the machine learning algorithms can able to produce highly accurate diabetes predictive healthcare systems. Pima Indians Diabetes Database from Kaggle is used to evaluate the working of algorithm.

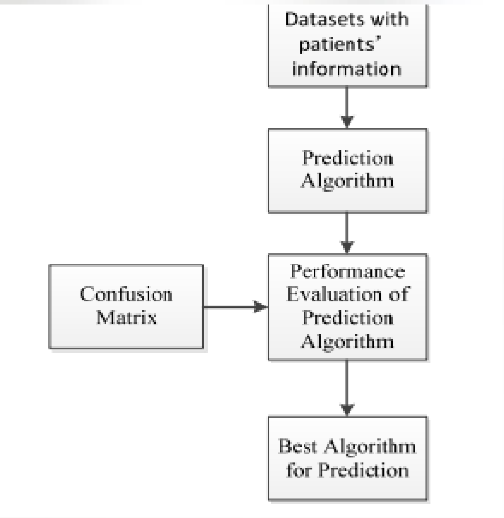
EXISTING METHODS:

There are many methods exist for the prediction of diabetes and the names of few from them are listed below:

- Decision Tree
- Gradient Boosting
- Support Vector Machine
- K-Nearest Neighbor
- Logistic Regression
- Random Forest

PROPOSED METHODOLOGY:

The algorithm process proposed in the project is represented in figure attached.

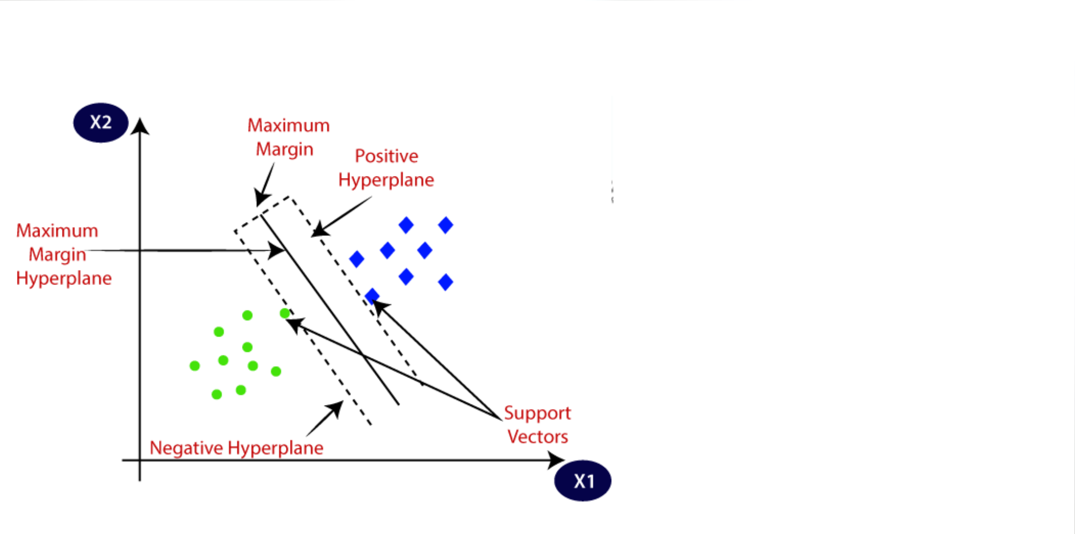

Datasets with patients' information
Prediction Algorithm
Confusion Matrix
Performance Evaluation of Prediction Algorithm
Best Algorithm for Prediction

First, the data set as input to the prediction algorithm, and then, though the evaluation model which is the method of introducing a confusion matrix to verify the classification accuracy of the algorithm. Finally, we get the algorithm with the highest accuracy in predicting diabetes.

Predicted Algorithm for performance evalution with algorithms are:

1. Support Vector Method

SVM is a generalized linear classifier that performs binary classification of data according to supervised learning. Its decision boundary is the maximum-margin hyperplane for solving learning samples. This hyper plane can be used for classification or regression also. SVM differentiates instances in specific classes and can also classify the entities which are not sup- ported by data. Separation is done through hyper plane performs the separation to the closest training point of any class.

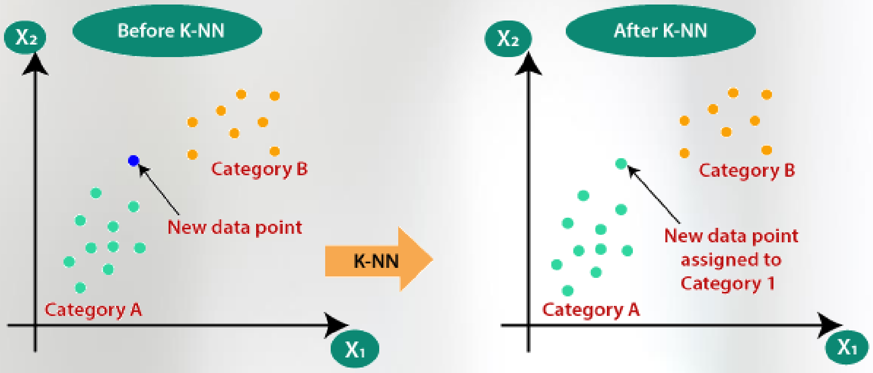


Algorithm

- Select the hyper plane which divides the class better.
- To find the better hyper plane you have to calculate the distance between the planes and the data which is called Margin.
- If the distance between the classes is low then the chance of miss conception is high and vice versa. So, we need to Select the class which has the high margin.
- Margin = distance to positive point + Distance to negative point.

2. K-Nearest Neighbor (KNN)

KNN is a supervised machine learning algorithm which uses Euclidean distance formula for finding the data nearest to the available categories.

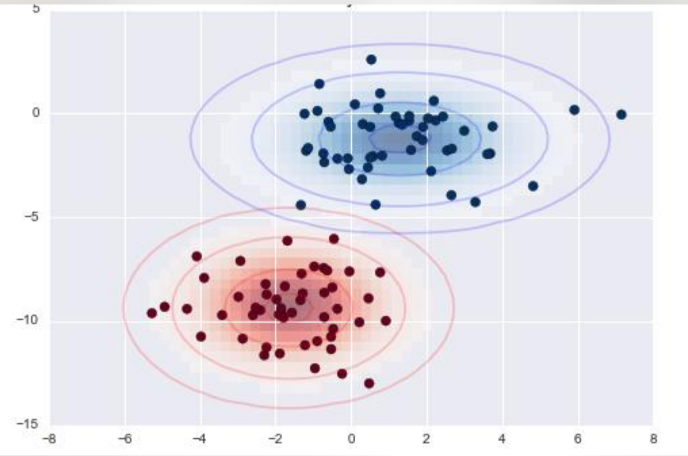


Algorithm

- Take a test dataset of attributes and rows.
- Find the Euclidean distance by the help of formula.
- Then, decide a random value of K is the no. of nearest neighbors.
- Then with the help of these minimum distance and Euclidean distance find out the nth column of each.
- Find out the same output values.
- If the values are same, then the patient is diabetic, otherwise not.

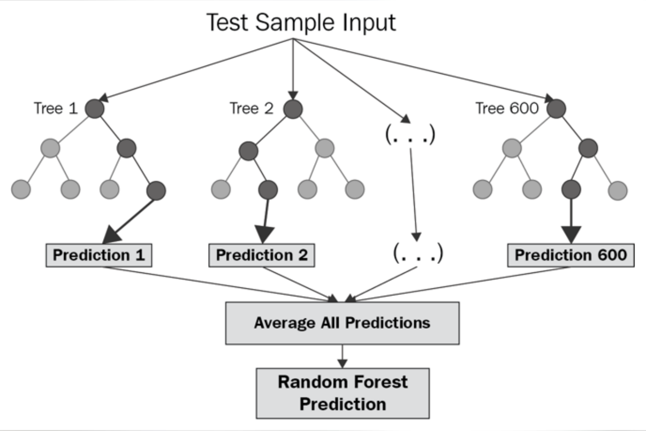
3. Naïve Bayes Classifier

Naive Bayes classifier is a series of simple probability classifiers based on the use of Bayes' theorem under the assumption of strong (naive) independence between features. The classifier model assigns class labels represented by feature values to problem instances, and class labels are taken from a limited set. For the given item to be classified, the probability of each category appearing under the condition of the occurrence of the item is solved, whichever is the largest, and the category to be classified is considered.



4. Random Forest Algorithm

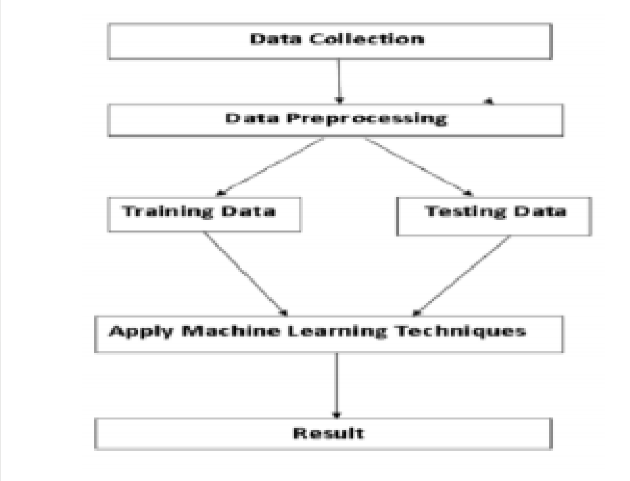
The random forest is a classification algorithm consisting of many decision trees. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.



Algorithm

- The first step is to select the R features from the total features m where R<<M.
- Among the R features, the node using the best split point.
- Split the node into sub nodes using the best split.
- Repeat a to c steps until l number of nodes has been reached.
- Built forest by repeating steps a to do for a number of times to create n number of trees.

METHODOLOGY AND IMPLEMENTATION:



- For the data collection the data set, Pima Indians Diabetes Database from Kaggle is used to evaluate the working of algorithm which has data of 768 patients and Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function and Age are the mentioned parameters on which diabetes depends.

- For Data Processing, we will perform two steps:

1. **Missing value removal:**
   Remove all the instances that have zero (0) as worth. Having zero as worth is not possible. Therefore, this instance is eliminated. Through eliminating irrelevant features/instances we make feature subset and this process is called features subset selection, which reduces dimensionality of data and help to work faster.

2. **Splitting of data:**
   After cleaning the data, data is normalized in training and testing the model. When data is spitted then we train algorithm on the training data set and keep test data set aside. This training process will produce the training model based on logic and algorithms and values of the feature in training data. Basically, aim of normalization is to bring all the attributes under same scale.

- Apply Machine Learning

When data has been ready, we apply Machine Learning Technique. We use different classification and ensemble techniques, to predict diabetes. The methods applied on Pima Indians diabetes dataset. Main objective to apply Machine Learning Techniques to analyse the performance of these methods and find accuracy of them, and also been able to figure out the responsible / important feature which play a major role in prediction.

The Techniques are follows:

1. Support vector method
2. K Nearest Neighbor
3. Naïve Bayes
4. Random Forest

IMPLEMENTATION OR STEPS FOR MODEL BUILDING:

Step 1: Import required libraries, Import diabetes dataset.

Step 2: Pre-process data to remove missing data.

Step 3: Perform percentage split of 80% to divide dataset as Training set and 20% to Test set.

Step 4: Select the machine learning algorithm i.e., K Nearest Neighbor, Support Vector Machine, Naïve bays Random Forest algorithm.

Step 5: Build the classifier model for the mentioned machine learning algorithm based on training set.

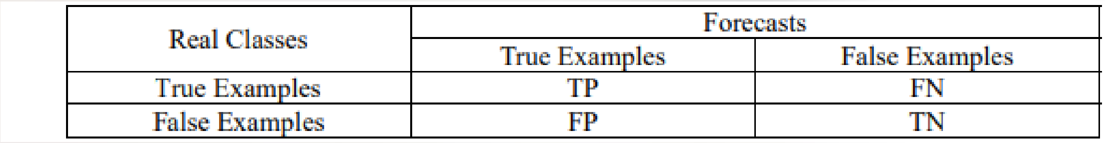
Step 6: Test the Classifier model for the mentioned machine learning algorithm based on test set.

Step 7: Perform Comparison Evaluation of the experimental performance results obtained for each classifier.

Step 8: After analysing based on various measures conclude the best performing algorithm.

RESULT & CONCLUSION:

In order to compare the pros and cons of the classification models, it is necessary to provide metrics to evaluate the performance of the models. Here we divide the sample into four classes like true examples (True Positive, TP), false positive (FP), true negative examples (True Negative, TN), and false negative examples (False Negative, FN). Let TP, FP, TN, and FN respectively denote the corresponding number of samples, TP+FP+TN+FN=n, n is the sample size, and the confusion matrix of the classification result is shown in the following table:



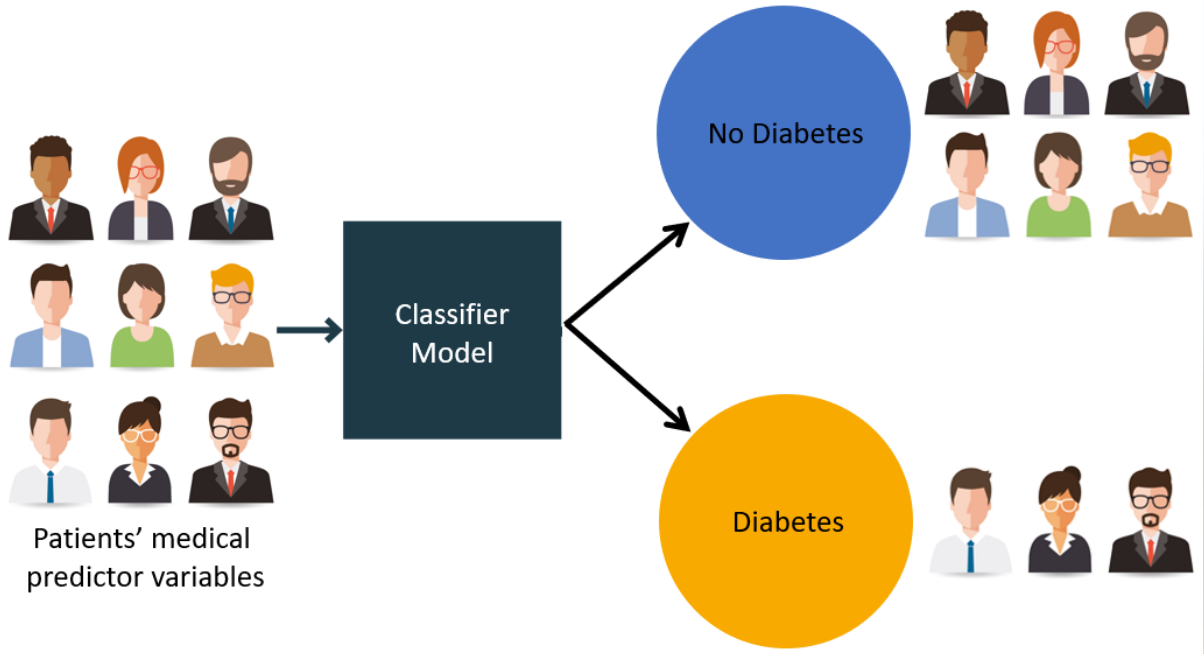
Real Classes	Forecasts	
	True Examples	False Examples
True Examples	TP	FN
False Examples	FP	TN

The characteristic results into two categories, using "1" for positive results and "0" for negative results. First, we split the data into two parts. In this experiment, the ratio of training set to prediction set is 80:20. Using the training set data for model to train, and then use the trained model and prediction set as input in the prediction component.

We summarize the results of the above four algorithms. The ML model KNN was able to classify patients as diabetic or not with an accuracy of 72.078%. The ML model RF was able to classify patients as diabetic or not with an accuracy of 74.025% and the ML model SVM and Naïve Bayes were able to classify patients as diabetic or not with an accuracy of 77.272%.

The main aim of this project was to design and implement Diabetes Prediction Using Machine Learning Methods and Performance Analysis of that methods and it has been achieved successfully!

.



Thank You!