[1]Ujwala Ghodeswar

[2]Minal Keote

Analysis of Diabetic Prediction & Regression System using Machine Learning Algorithms

Abstract: - This paper analyses linear regression classifier machine learning algorithms for prediction of diabetes. Classification & Regression analysis is done by using linear classifier, logistic regression, Naïve bayes, Random Forest, XGBoost, Decision classifier machine learning algorithm. Datasets of diabetes is taken from Kaggle. There are total 768 patients data with 9 features. Feature Outcome indicates output dependent feature. 20% Test size and 80% training size are considered for calculation of training, testing accuracy. Model is iterated number of times. Highest accuracy is calculated after doing iterations. Naïve bays algorithm gives highest accuracy as compared to Decision tree, linear regression and logical regression algorithms. Training and testing accuracy is calculated for the mentioned machine learning algorithms. K fold cross validation method is used to remove the overfitting problem. further performance of the algorithm is also calculated based on precision, recall, F1 score, support parameters. AUC curve and confidence matrix terms are also used for validation of results. These parameters are derived using the confidence matrix. The results show that naïve bays, XGBoost, and random forest algorithms outperform other algorithms in terms of precision and accuracy. As a result, for the diabetic data set, these three algorithms are utilized to predict people with diabetes disease using AUC and Precision Recall analysis.

Keywords: Machine learning, Naïve bays, Decision tree, Regression algorithms, XGBoost, Random Forest

I. INTRODUCTION

Peoples are affected by diabetes if they are not taking healthy diet, green vegetables and not doing regular exercises. Most of the peoples are eating junk food because of busy working schedule. This causes diabetes to be detected at the age of 35 to 45 years. The person suffering from diabetes is not able to prepare sufficient insulin in his body. Diabetes is the chronic diseases like slow poison. Many peoples in India are suffering from diabetes disease. This disease is required to be controlled in early stage otherwise it leads to several complications on health and eye sight. It should be detected and diagnosed in early stage of the disease. There are two types of diabetics. First one is Type I and second is Type II diabetes. Type I diabetes is due to genetics and Type II diabetes occurs at the age of approximately 40 when the person's body is not able to form insulin. Reason for Type II is mostly due to person's obesity, fatness, and consuming large amount of fast food. A person's diabetes diseases can be identified using a variety of measures. The following section discusses the work of many researchers in the diabetes prediction and analysis using machine learning.

Machine learning model gives accuracy from 2.71% to 13.13% and implemented web-based application. Authors [3,5] done analysis of various available methods and implemented ontology-based model for prediction. They also compared implementation with accuracy of other models. PIMA Indian and curated dataset is used to predict diabetes mellitus [2]. India is the biggest country having patients suffering from diabetes.

This disease can also be due to genetics i.e. if father or mother is detected with the disease then it is most likely that child can also be detected at the age of 40 with diabetes. Good amount of accuracy is resulted by using SVM algorithm as compared with other algorithms [4]. Light gradient Boosting algorithm resulted in 98.99% accuracy [7]. Various machine learning algorithms are analyzed for accuracy and prediction results are obtained using five machine learning algorithms i.e. k-nearest neighbor, logistic regression, decision tree, random forest and SVM [8].

Various machine algorithms for predicting diabetes disease are analyzed for training and testing accuracy. This gives the result based on analyzing various parameters related to diabetes [6].

[1] *Associate Professor: Yeshwantrao Chavan College of Engineering, Nagpur, India

[2] Assistant Professor, Yeshwantrao Chavan College of Engineering, Nagpur, India

The organization of this paper is as follows. Section 2 describes datasets information and explanation of dataset. Section 3 describes performance metrics required for prediction. Section 4,5,6 gives machine learning algorithms used for classification and prediction. Section 7 gives results and discussion.

II. DIABETES DATASET INFORMATION

Diabetes dataset is taken from Kaggle [10]. As per clinical terms a person is said to be at risk or prediabetes if HBA1C (3-month glucose load) percentage is in between 5.7 to 6.4. For prediction of diabetes PIMA database is taken from Kaggle website. This dataset consists of 8 features. The features are as given in Table 1 below. This dataset consists of 8 independent variables i.e. pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, Body Mass Index (BMI), DiabetesPedigreeFunction, Age and one dependent variable as Outcome. These features are shown with lower and higher range values. Histogram plot of these variables given in fig.1 for visualizing the range of values.

Table1: Dataset information

S.No.	features	Lower Range	Higher Range
1	Pregnancies	0	17
2	Glucose	0	199
3	Blood Pressure	0	122
4	Skin Thickness	0	99
5	Insulin	0	846
6	BMI: Body Mass Index	0	67.1
7	DiabetesPedigreeFunction	0	2.42
8	Age	21	81
9	Outcome	0	1

The graphical representation of eight attributes using a histogram distribution is shown in figure 1. This graph exhibits the distribution of features across a range. This dataset consists of total 768 rows with 8 columns as features. The available machine learning algorithm used for prediction are Linear regression, Logistic regression, Decision tree, Naïve Bayes, Random Forest and XGBoost algorithm.

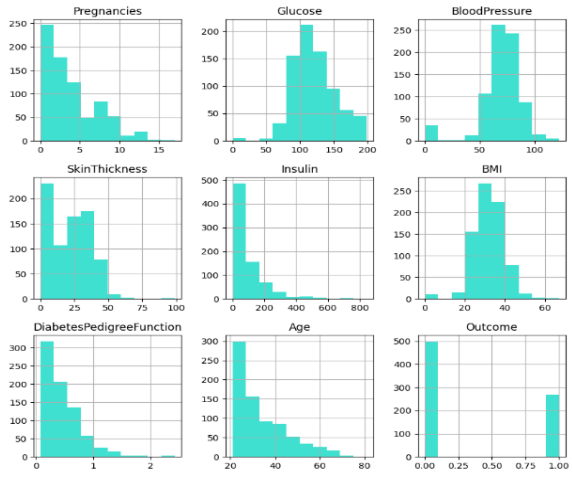

Fig.1 Feature Histogram.

The feature distribution graphic shows 549 women are more likely to get suffered from diabetes in their 0 to 5 months of pregnancy. A greater number of women have glucose levels higher than 50. A total 698 peoples with Blood pressure levels ranging from 48.80 to 97.60mm Hg. 768 women have skin thickness levels ranging from 0 to 49.50 mm. Insulin level is varying from 0 to 253.80 U/ml for 712 women's. Body mass index is varying from 20.13 to 46.97 for total 722 women's. A lower BMI implies being underweight, whereas a higher BMI denotes

obesity. Diabetic pedigree function is ranging from 0.08 to 1.01 for 718 data sets. 687 data for age feature is ranging from 21 to 51. These age peoples are more likely to be affected by diabetes disease.

This paper analyses performance of six machine learning algorithms i.e. Linear regression, Logistic regression, Decision tree, Naïve Bayes, Random Forest and XGBoost algorithm. Performance of algorithm is calculated based on accuracy, precision and F1 score. The k-fold cross validation score is used to improve performance metric parameters and address the overfitting problem

III Execution steps for classification of Diabetes

Figure 2 describes the step-by-step execution of prediction algorithms. These steps are described below.

Step1: All the required libraries are imported in python. Diabetes data set is loaded

Step 2: Find null values in the data, Analyze the data,

Step 3: Divide the data into training and testing datasets

Step 4: Apply classification model and calculate accuracy.

Step 5: Apply hyperparameter tuning using XGboost algorithm and calculate accuracy.

Step 6: Compute training and testing accuracy obtained using k fold cross validation score.

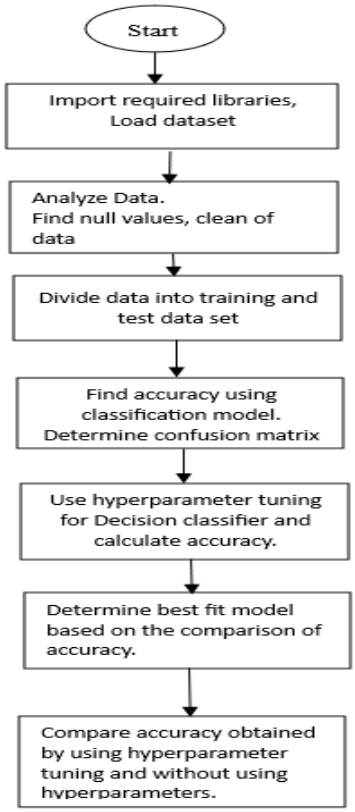

Fig. 2. Execution steps for classification of Diabetes prediction system

IV Linear Regression machine learning algorithm

In this machine learning method first correlation between all the features is calculated. Figure 3 shows correlation between 8 features and one outcome using heatmap command. This shows relation between all the features using 8 by 8 matrix. Darker portion in column shows that corresponding feature is less correlated with relevant row. For example, by considering Bloodpressure row and outcome as column, Bloodpressure is having less correlation with target outcome as the correlation value is 0.065. This value is lower as compared to the other features. Outcome is output dependent variable which gives information about patient having diabetics or not. It is given in terms of

binary values. If outcome is equal to binary value ‘1’ then the person is having diabetes and binary value ‘0’ indicates that the person is not having diabetics.

Pregnancies is strongly correlated to feature ‘age’. Outcome is positively correlated with Glucose as it is having highest value i.e.0.47. Feature ‘BMI’ and Insulin are having strong relation with SkinThickness attribute.

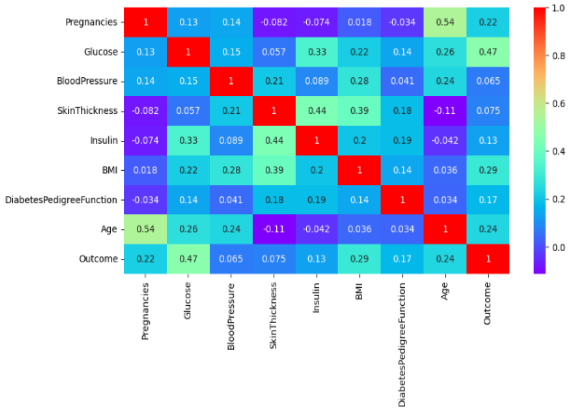

Fig.3: Correlation between dependent and independent variable

Out of 768 samples 80% of the samples are taken as training samples i.e.614 samples are taken as train data set and 20% of the data is given as testing data sets i.e. 154 samples.

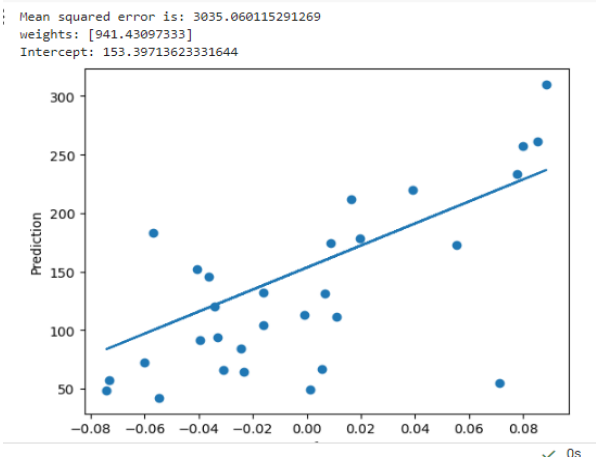

Fig.4: Prediction using Linear classifier

Figure 4 shows linear classification of diabetes prediction. Basic equation for linear regression is y=mx+c. Using this algorithm modified equation is y=941x+153.

This is the linear regression equation calculated by considering single feature. Mean squared error for single feature is 3035 which is very large. Weight value obtained is 941 and intercept value calculated is 153. After this all the features are given as input to the model and mean squared error is calculated. For this Training size is 70% and testing size is given as 30% of the features. Mean squared error obtained by considering all features is 1826.484 which is very small as compared to the single feature.

Calculated weights are: [-1.166, -237.181, 518.312, 309.042, -763.108, 458.883, 80.611, 174.317, 721.480, 79.195] and Intercept: 153.058

Linear regression gives train accuracy of 50 % and test accuracy of 64%. As the training accuracy is less than testing accuracy this model is underfitted for this type of data. This classifier is used for prediction and classification of diabetes datasets. Using following command train and test accuracy is calculated in python.

print('train accuracy:', model.score (diabetes_x_train, diabetes_y_train))

print('test accuracy:', model.score(diabetes_x_test, diabetes_y_test))

train accuracy: 0.5070727746230751

test accuracy: 0.6454950067486565

Using confidence matrix, Performance parameters like accuracy, misclassification, recall, precision, F1 score are calculated. Equations of these parameters are given below.

$$Accuracy = \frac{TN + TP}{TN + FP + FN + TP} \quad (1)$$

$$Misclassification = \frac{FN + FP}{TN + FP + FN + TP} \quad (2)$$

$$\text{Re}call = \frac{TP}{FN + TP} \quad (3)$$

$$\text{Pr}ecision = \frac{TP}{FP + TP} \quad (4)$$

$$Specificity = \frac{TN}{TN + FP} \quad (5)$$

$$\text{F1 score} = 2 \times \frac{\text{Pr}ecision \times \text{Re}call}{\text{Pr}ecision + \text{Re}call} \quad (6)$$

False Positive Rate(FPR)=1 -Specificity (7)

$$FPR = \frac{FP}{TN + FP} \quad (8)$$

Table II Confusion Matrix

Confusion matrix	Predicted negative value	Predicted positive value
Actual negative value	TN	FP
Actual Positive value	FN	TP

Table II gives the basic format of confusion matrix. Performance parameters given in equation 1,2,3,4,5,6,7,8 are calculated from confusion matrix. Parameters of confusion matrix are True Negative (TN), False Negative (FN), False Positive (FP), True Positive (TP). If the actual value is negative and predicted value is also negative then it is indicated as TN. If the actual value is positive and Predicted value is also positive then it is indicated as TP. Similarly, if the actual value is positive and predicted value is negative then it is FN. FP is the predicted positive value but actual value is negative.

For the remaining five machine learning algorithms classification report & confusion matrix is calculated. This classification report gives the performance matrix values of accuracy, precision, F1 score, Recall as given in eq.1,2,3,4, and 5.

For increasing accuracy of the system FPR (false positive rate) should be decreased and TPR (True positive rate) should be increased. AUC is area under the ROC curve. It is used to find the performance of the model for all threshold values. If the value of AUC score is more then its predictions are correct so it is able to separate between two classes. An excellent model has AUC near to 1. And poor model has AUC near to 0 means it has worst separability. i.e. it is predicting 0s as 1s and 1s as 0s. If AUC is 0.5 it means model has no class separation capacity. TPR =1 and FPR=0 then it is perfect model which perfectly separating two classes. AUC=0 means wrong classification for all the values is done.

Precision Recall (PR) curve is used when there is a class imbalance data set. High precision means low false positive rate and high recall means low false negative rate. In PR curve y axis is equal to Precision, and x axis is

recall. No skill means there is no basic classifier i.e. base line shown in dashed line. For each regression model if the probability is greater than 0.5 then it is considered as belonging to class 1 else it is considered as class 0. AUC score is calculated by considering 0.5 threshold by default and then threshold is changed to 0.3

V. Logistic Regression

Logistic regression is used for binary classification having two classes. The two classes are patient is having diabetes disease or not having diabetes disease.

F(x)=1/(1+e-x) It is a sigmoid function which is S shaped curved. It gives value either 0 or 1. Probability of target variable is converted to 0 and 1 depending on probability is greater than threshold then it is considered as class 1 and for less than 0.5 it is considered as class 0

The intercept b0= [-2.48769611]

The coefficient b1= [[2.282e-02, -4.5712e-03, -7.740e-03, -2.886e-03,8.2848e-04, -1.739e-02,-1.513e-02,

-2.468e-03, 7.832e+00]]

VI Decision Tree Classifier

In decision tree classifier 20% test size and 80% train size is used. Maximum depth of the tree is considered as 3. Using this output 'y' is predicted and test accuracy score obtained is 74.02% and train accuracy is 78.01%. In decision tree classifier 20% test size and 80% train size is used. Using this output 'y' is predicted and accuracy score obtained is 75.97% with train accuracy of 76.22% and test accuracy is 75.97%. AUC score is 70.15.TPR i.e. Recall or sensitivity measures the proportion of the actual positives that are correctly specified.

True posistive: 27

false posistive: 13

True negative: 90

false negative: 24

$$\mathrm{Re}call = \frac{32}{22+32}$$

Recall=0.59

FPR i.e. False positive rate measures the ratio between false positives and total number of actual negatives regardless of classification.

FPR=FP/FP+TN= 22/21+79=0.22

VII Results and Discussion

This section is divided into five parts. These includes results of Train and test accuracy, calculation of train and test accuracy using k fold cross validation, AUC score, Precision recall and confidence matrix. Results are computed by using two different threshold value.

I. Train and Test Accuracy:

Training size and testing data size of all the machine learning algorithm is selected as given in table III below. Machine is trained and accuracy is calculated. It can be observed that among the given algorithms linear regression is having lowest accuracy hence this algorithm is not considered for this regression analysis. This is considered in this paper for classification purpose.

Random forest, Decision tree and XGBoost algorithms are having train accuracy equal to 100% and approximately 30% less testing accuracy is obtained. This is the overfitting problem as train accuracy is greater than test accuracy.

For removing this big difference between train and test accuracy k fold cross validation score method is applied. Table IV shows the results of k fold validation method. Using this method overfitting problem is removed.

Table III Train and test accuracy

Algorithms	Train accuracy	Test accuracy	Test size	Train size
Decision tree classifier	100%	70.12%	20	80
Naïve Bayes	76%	75%	20	80
Linear regression	32.81%	17.09%	30	70
Logistic regression(LR)	78.99%	74.67%	20	80
Random Forest	100%	73.37%	20	80
XGBoost	100%	77.92%	20	80

II. Train and Test Accuracy by using K fold Cross validation score:

K fold cross validation score removes overfitting problem as given in table IV. Decision tree, Random Forest and XGboost algorithm train accuracy is nearabout same as test accuracy as observed from table IV.

Table IV Train and test accuracy using k fold cross validation score

Algorithms	Train accuracy after K fold cross validation	Test accuracy after K fold cross validation	Test size	Train size
Decision tree classifier	71.47%	71.33%	20	80
Naïve Bayes	76%	75%	20	80
Linear regression	32.81%	17.09%	30	70
Logistic regression(LR)	78.99%	74.67%	20	80
Random Forest	76.71%	77.45%	20	80
XGBoost	75.25%	73.91%	20	80

III. Area Under the Curve score (AUC)

It can be observed from the table V of AUC score that AUC score of XGBoost algorithm is more as compared to the other algorithms for threshold =0.5 and also for threshold=0.3. This area is also shown in the ROC curve for all the five algorithms. Area covered by XGBoost algorithm is more. Accuracy for XGBoost algorithm is approximately 76% which is more as compared to the other algorithms. Hence XGBoost algorithm is used for diabetes prediction datasets.

ROC curve is used to decide best threshold value. ROC curve is used when there are equal number of observations for each class. i.e it gives balanced data. It is plotted with TPR on y axis and FPR on x axis. TPR is true positive rate and FPR is false positive rate. Threshold value is selected between 0 .0 to 1.0 TPR is recall/sensitivity.

Table V AUC Score

Algorithms	AUC score	Threshold	Accuracy
Decision tree	67.39	0.5	70.12
	67.39	0.3	70.12
Naïve Bayes	70.30	0.5	75.3
	72.02	0.3	73.37
Logistic regression(LR)	70.13	0.5	74.67
	72.56	0.3	72.72
Random forest	69.75	0.5	73.37
	74.47	0.3	72.07

XGBoost	73.70	0.5	75.97
	73.42	0.3	76

Figure 5,6,7,8,9 gives the AUC score of Decision tree, Naïve bayes, logistic regression, Random forest, XGboost algorithms. It can be observed from Receiver operating Curve(ROC) that the maximum area is covered in Random forest, XGboost and Naïve bayes machine learning algorithms. Lower area is covered in decision tree and logistic regression algorithms.

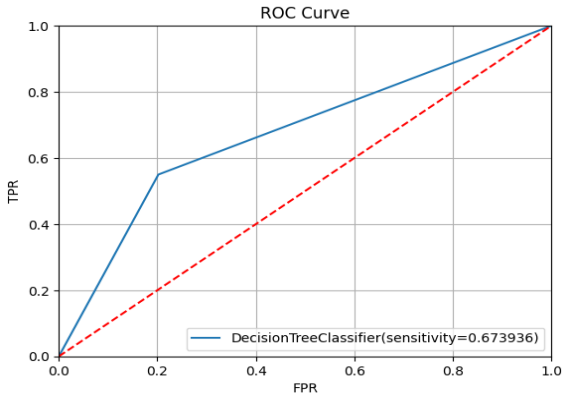

Fig.5 AUC curve of Decision Tree Classifier

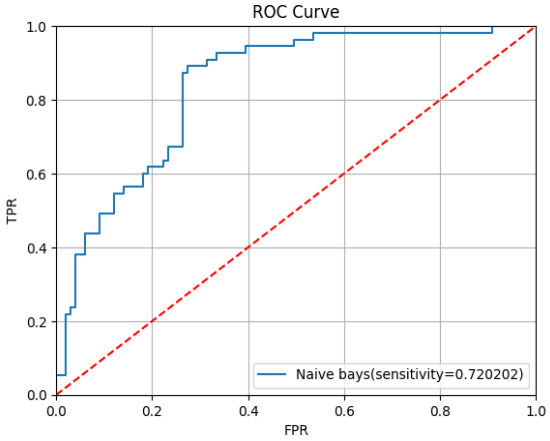

Fig.6 AUC curve of Naïve Bays

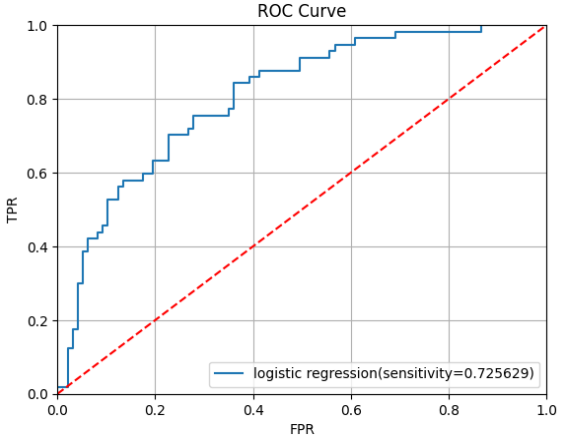

Fig.7 AUC curve of logistic regression

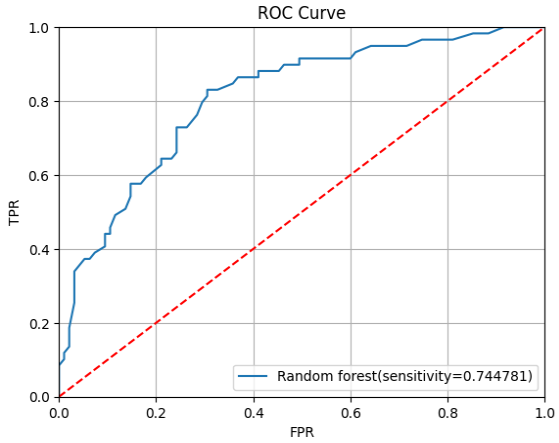

Fig.8 AUC curve of Random Forest

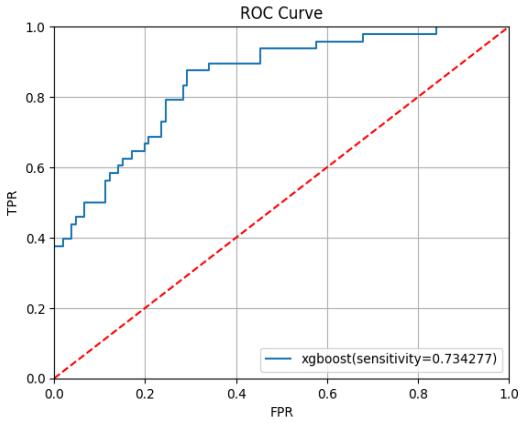

Fig.9 AUC curve of XGBoost

IV. Precision and Recall Curve

Results are further validated by calculating Precision, Recall, F1 score by considering two threshold values i.e.0.5 and 0.3. Table VI shows results by considering 0.5 as threshold and table VII shows calculations using 0.3 as threshold. It can be observed that as the threshold value is decreased from 0.5 to 0.3, precision value is increased for Naïve bays, logistic regression, random forest and XGboost algorithms whereas there is no improvement in the precision value of decision tree algorithm. But the values of F1 score and Recall are decreased as the threshold value is decreased from 0.5 to 0.3. Less values are obtained because in the formula of Recall the false negative term is added in denominator with true positive term.

Table VI Precision, recall, f1 score for threshold=0.5

Algorithms	Class	Precision	Recall	F1 score
Decision tree	0	0.74	0.80	0.77
	1	0.63	0.55	0.59
Naïve Bayes	0	0.77	0.88	0.82
	1	0.71	0.53	0.60
Logistic regression	0	0.76	0.88	0.81
	1	0.71	0.53	0.61
Random forest	0	0.75	0.85	0.80
	1	0.70	0.54	0.61
XGBoost	0	0.83	0.85	0.84
	1	0.65	0.62	0.64

Table VII Precision, recall, f1 score for threshold=0.3

Algorithms	Class	Precision	Recall	F1 score
Decision tree	0	0.74	0.80	0.77
	1	0.63	0.55	0.59
Naïve Bayes	0	0.81	0.77	0.79
	1	0.62	0.67	0.64
Logistic regression	0	0.82	0.73	0.77
	1	0.61	0.72	0.66
Random forest	0	0.87	0.64	0.74
	1	0.60	0.85	0.70
XGBoost	0	0.84	0.80	0.82
	1	0.60	0.67	0.63

Precision and Recall curve are shown in figure 10,11,12,13,14. Numerical values for class 0 and 1 are summarized in table VI and VII. It can be observed from P-R curve that when the precision value is more recall rate is lower and lower value of precision rate gives higher recall rate. Class 0 means person is not having diabetics and class 1 means person detected with diabetics' disease.

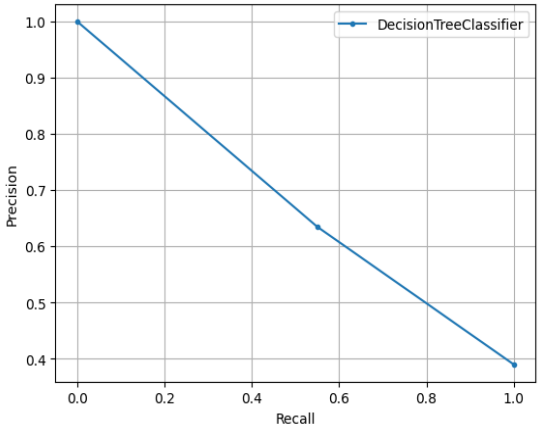

Fig.10 Precision Recall curve for decision tree classifier

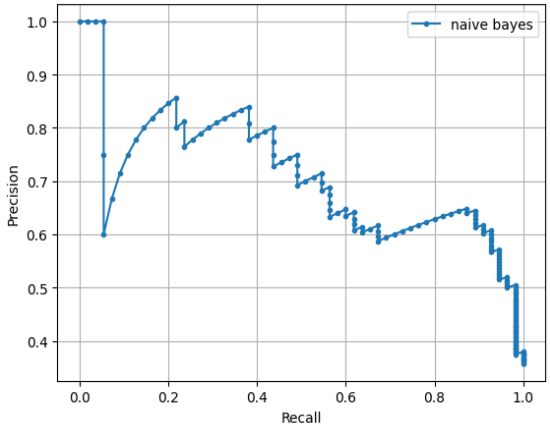

Fig.11 Precision Recall curve for naïve bays

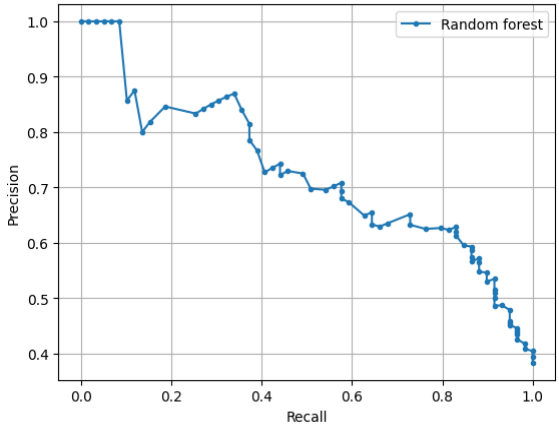

Fig.12 Precision Recall curve for Random Forest

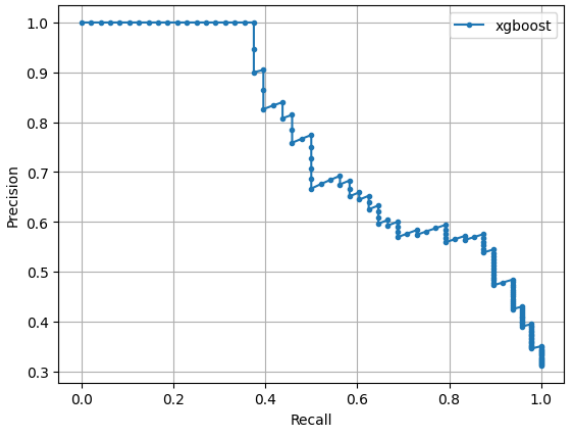

Fig. 13 Precision Recall curve for XGboost

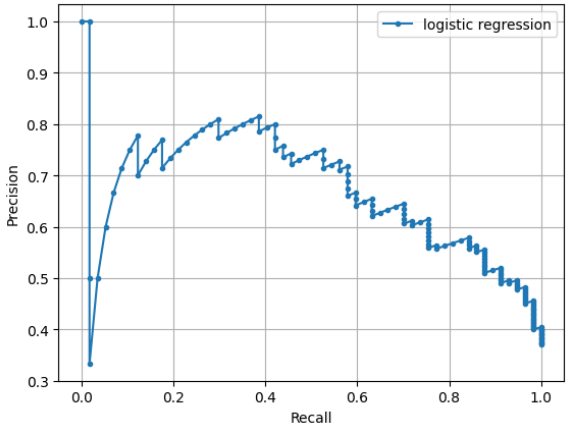

Fig.14 Precision Recall curve for logistic regression

V. Confusion matrix parameters

Values of Precision, Recall and F1 score are calculated based on Confidence matrix. Table VIII summarizes confidence matrix for five machine learning algorithms using two values of threshold. It can be observed that True positive rate is increased for naïve bays, random forest and XGBoost algorithms as the threshold is decreased from 0.5 to 0.3. There is no improvement in the Decision tree classifier values as observed from table IX.

Table VIII Confusion matrix for threshold=0.5

Algorithm	TP	FP	TN	FN
Decision tree classifier	33	19	75	27
Naïve bayes	29	12	87	26
Logistic Regression	30	12	85	27
Random forest	32	14	81	27
XGBoost	30	16	90	18

Table IX Confusion matrix for threshold=0.3

Algorithm	TP	FP	TN	FN
Decision tree classifier	33	19	75	27
Naïve bayes	37	23	76	18
Logistic Regression	41	26	71	16
Random forest	50	34	61	9
XGBoost	32	21	85	16

Threshold in Naïve bayes is considered as 0.3 means if the predicted probability is greater than 0.3 then it is considered as of class 1 and if it is less than the 0.3 then it is considered as belonging to class 0. Class 0 is considered as when the probability is between 0 to 0.5 and class 1 is considered when the probability is between 0.5 to 1.

VIII Conclusion

From table I of accuracy, it can be observed that decision tree classifier, random forest and XGBoost algorithms are having training accuracy more than testing accuracy. This is problem of overfitting. In these three models train accuracy is 100% and testing accuracy is 30% lower than train accuracy. This overfitting problem is solved by using k fold cross validation method. In this method different experiments are carried on different slices of test data and train data. Test data is taken as 10% and train data is taken as 80%. Using cross validation overfitting problem is removed. After cross validation train accuracy and test accuracy are calculated for these three models. It can be observed from table that XGBoost and random forest model performs well for train data as it is having good train and test accuracy as compared to other models after applying k fold cross validation.

Results are further validated by using other metrics such as plotting the AUC curve, calculating precision, recall and f1 score. These parameters are calculated based on confidence matrix. Using these results it is verified that naïve bays, XGboost and random forest gives good precision and accuracy of these algorithm is more as compared to other algorithms. Hence for diabetics data set these three algorithms are used for predicting the person with diabetes disease based on AUC and Precision Recall analysis.

References

[1] Nazin Ahmed, Rayhan Ahammed, Md. Manowarul Islam, Md. Ashraf Uddin, Arnisha Akhter, Md. Alamin Talukder, Bikash Kumar Paul, Machine learning based diabetes prediction and development of smart web application, International Journal of Cognitive Computing in Engineering, Volume 2,2021,Pages 229-241, ISSN 2666-3074, https://doi.org/10.1016/j.ijcce.2021.12.001.

[2] Wee, B.F., Sivakumar, S., Lim, K.H. et al. Diabetes detection based on machine learning and deep learning approaches. Multimed Tools Appl (2023). https://doi.org/10.1007/s11042-023-16407-5

[3] Neha Thakur,Virendra Singh, Diabetes prediction using machine learning, International journal of Emerging Technologies and innovative research,Vol.8, Issue 6, pp.1490-1499,June-2021

[4] Hakim EI Massari,Zineb Sabouri,Sajida Mhammedi,and Noreddine Gherabi, Diabetes prediction using machine learning algorithms and ontology, Journal of ICT standardization,vol.10.2. pg.no.319-338

[5] Tasin I, Nabil TU, Islam S, Khan R. Diabetes prediction using machine learning and explainable AI techniques. Healthcare Technol Lett. 2022 Dec 14;10(1-2):1-10. doi: 10.1049/htl2.12039.

[6] Aishwarya Mujumdar, V Vaidehi, Diabetes Prediction using Machine Learning Algorithms, Procedia Computer Science, Volume 165, 2019, Pages 292-299

[7] B. Shamreen Ahamad, Meeakshi S.Arya, Auxilia Osvin V. Nancy, Diabetes Mellitus Disease Prediction using machine learning Classifiers with oversampling and feature augmentation", Advances in Human- computer Interaction, Volume 2022,pp. 1-14.

[8] Shimoo Firdous,Gowher A Wagai, Kalpana Sharma, A survey on diabetes risk prediction using machine learning approaches, Journal of family medicine and primary care, pp.6929-6934, 2022.

[9] K.M.Jyoti Rani,"Diabetes prediction using machine learning", International journal of scientific research in computer science, engineering and information technology,volume 6,Issue 4,pp . 294-305.

[10] PIMA Indian Daibetes Database, Kaggle, https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database