Turkish Journal of Computer and Mathematics Education Vol.12 No.13 (2021), 177-184

Research Article

Diabetes Diagnosis using Ensemble Models in Machine Learning

Ashok B[a], Mr. Manoj Wairiya [b], Dr. Divya Kumar[c]

[a,b,c] Department of Computer Science Engineering, Motilal Nehru National Institute of Technology, Prayagraj, IndIa
[a] ashok261994@gmail.com, [b] wairya@mnnit.ac.in, [c]divyak@mnnit.ac.in

Article History: Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 4 June 2021

Abstract: Diabetes is one of the diseases where early detection is must given the fact that it is not possible to cure the disease once the patient gets the diabetes disease. As the number of patients is increasing on a day to day basis, it is difficult for the doctors to perform manual detection. With the technologies like Machine Learning in hand, we can perform automative detection to some extent. Lot of research has been performed till now on this diabetes diagnosis problem. This paper discusses predictive analysis using two ensemble machine Learning Algorithms such as Random Forest and GBDT. In this paper, we have performed various Experiments on Pima Indians Diabetes Dataset which contains Diabetes patients record and results are discussed. This paper additionally discusses the importance of Interpretability of output in the healthcare domain and explains how it will help the doctors in real time if we could provide interpretability of the output along with the output of the patient record given by machine learning model.

Keywords: Diabetes, GBDT, Healthcare, Interpretability, Machine Learning,Random Forest

1. Introduction

Diabetes is a disease that occurs when the body doesn't make enough insulin. Diabetes is one of the interesting healthcare problems to work with. The reason is that, lots of people are affected with diabetes and at the same time, the early detection of this disease is must otherwise the patient can have various issues including loss of eyesight permanently. Once the patient is diagnosed with Diabetes, it can't be cured. Majorly, Diabetes patients have two types of disease such as type 1 or type 2. Body of the Type 1 diabetes patients doesn't produce enough insulin whereas the body of Type 2 diabetes patients doesn't respond to insulin when compared to a normal person. As the number of Diabetes patients are increasing rapidly, the automated Machine Learning models will definitely help the doctors in reducing their workload. In this paper, we have discussed some of the methods of doing diabetic diagnosis using Machine Learning. With the availability of the data, we can use machine learning models to predict whether the patient has diabetes or not. In this paper, we have used Pima Indians Diabetes Dataset. This dataset contains the records of 768 female diabetes patients. This dataset contains two classes such as class 0 and class 1. class 0 represents that corresponding patient don't have diabetes whereas class 1 represents the patient has diabetes.This dataset contains 8 features such as Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, DiabetesPedigreeFunction, Age, Outcome along with class label feature named 'Outcome'.

2. Literature Review

Paper[1] has applied the models such as SVM, Naive Bayes and Decision Tree and concluded that each of these models will work well for different scenarios. In Paper[2], comparison of models such as Support Vector Machine and Naive Bayes is performed and these comparison are performed using the metrics such as precision, specificity, sensitivity and accuracy.In paper[3], the authors discussed about the method of using fuzzy interface system in order to perform diagnosis of diabetes. In paper[4], authors have discussed the various supervised machine learning algorithms and also discussed the strengths and weakness of algorithms. Paper[5] have performed outlier detection and applied Auto MultiLayer Perceptron to provide good accuracy. Paper[6] have discussed the sensors for diabetes diagnosis. Paper[7] discussed the methodology of applying Naive Bayes for Diabetes Diagnosis. In Paper[8], the authors have discussed the application of Deep Learning in the context of diabetes. Paper[9] discussed the problem in the form of two phases. First phase is for performing various data pre-processing techniques and in the second phase, Decision Tree model is applied and showcases the results. Paper[10] discusses the importance of data pre-processing before applying the actual data analysis and model building and it shows the comparison between accuracies of models when a model is built with data preprocessing and without data preprocessing. Paper[11] proposed a concept called Diabetes Diagnosis Expert System and discusses how it helps the diagnosis of diabetes. In Paper[12], the authors applied the ensemble boosting with perceptron algorithm and applied these datasets on three different datasets. In Paper[13], the authors have proposed the device which helps in early diagnosis of diabetes by monitoring the importance of diagnosis of diabetes. Paper[14] applied 6 different models and compared the performance metrics such as performance and accuracy and found the best model out of the 6 models in order to detect the diabetes with the help of patient record. Paper[15] discussed both the machine learning model like Support Vector Machine and Deep learning models like CNN and applied these models on Diabetes Dataset and produced the accuracy

3. Preprocessing Of The Data

Before looking at the importance of pre-processing in machine learning conference, let's look at the steps that's followed in this research paper to produce very good accuracy for this problem using the flowchart mentioned in Fig 1.

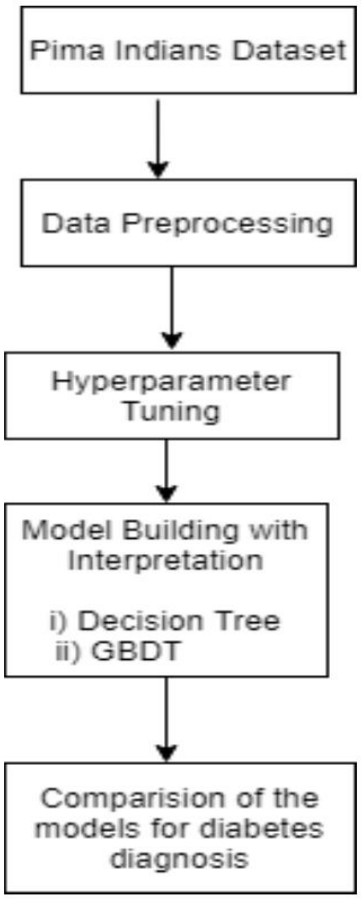

Fig 1. Steps involved in this process

As we have seen in the Introduction Section, there are 768 data points in the data set. Out of the 768 data points, 268 data points belong to Class 1 and 500 data points belong to Class 0. Generally, while handling Machine Learning Problems, there are two types of data set such as Balanced and Imbalanced Data set. Balanced Data sets are the data sets that we have almost equal number of points in class 0 and class 1. Imbalanced Data sets are the data sets that have fair amount of difference between the number of data points between the classes. This data set is an Imbalanced Data set because Class 0 has approximately twice the amount of data points that belong to class 1. In general, if we build models on this Imbalanced data set, our models won't perform well because output will be more favoured towards class 0 as it has more data points. So, we have performed upsampling. Upsampling is the process of increasing the number of datapoints of minor classes to be equal to maximal class. In this example, After performing upsampling on class 0, we have 500 data points for class 0 and 500 data points for class1. We have converted an Imbalanced data set to Balanced Data set.

Data set contains lots of missing or erroneous values. In our dataset, we have taken our features one by one and then performed imputation. Generally, median based imputation is preferred over mean based imputation because the mean based imputation is prone to outliers. For example, Blood Pressure feature had 16 missing values when class is 0 and 19 missing values when class is 1. So, once we have retrieved those missing values for both the classes seperately, we have performed class-based median level imputation. Similarly, we have applied imputation techniques for other features.

4. Data Splitting

Data set is splitted into Train and Test data. Train data is used to train the model and find the best hyper parameters of the models that we are going to train. Test dataset should be future unseen data and this dataset is used to find the accuracy of built models such as Random Forest and GBDT on this problem. We have discussed the models used and Hyper parameter tuning in the later section of this paper. As discussed above, after performing upsampling, we have 1000 data points. We have divided this upsampled dataset into train and test data such that train data set contains 67% of datapoints and test datset contains 33% of datapoints.

5. Hyperparameter Tuning

Hyper parameter Tuning is one of the most important step in Machine Learning. Hyperparameter Tuning is useful to avoid Overfitting and Underfitting while building the model. This process has to be carried out before building every model else model might be useless. Overfitting is the process of training the model so much to train data in such a way that train error is very less and test error is very high. In other words, train and test error should be as close as possible to avoid Overfitting. The machine learning model is said to be an underfitting model when both train and test error is very high. That is, if model does not work well even for training data, then the model is said to be Underfitting.

For this problem, I have built two models such as Random Forest and GBDT. Both these models are ensembling models and its underlying base models are Decision Trees.

- min samples leaf - minimum number of data samples allowed in a leaf node of the tree.
- n estimators - Number of base estimators that can be trained
- min samples split - Minimum number of data samples allowed in all the nodes of the tree.

In this problem, we have used GridSearchCV to find out the best value for each of the hyperparameters that is mentioned above.

GridSearchCV

When there is a chance for a hyperparameter to take values from the grid of values, then we can use grid search cv. When we apply grid search cv on grid of values, it will calculate auc score on each and every value of grid and whichever value has the best auc score, it will be selected as the best hyper-parameter and model will be trained on the value.

K-Fold Cross validation

While doing hyperparameter tuning, we want to test the hyperparameter accuracy on some data to select the best hyper-parameter. For this purpose, we are taking some part of data as cross validation data. If we don't have cross validation data, then we might need to check the hyperparameter accuracy on test data. But the main motive of machine learning is to keep the test data as future, unseen data. So, we will take part of the data as cross validation data. Instead of allocating separate dedicated parts to cross validation data, we will divide data into train and test data. Now train data is divided into K parts and in which one part will be taken as cross validation data and other K-1 parts will be taken as train data. In this example, K value is taken as 3.

6. Building Models

In this paper, we have discussed two models such as Ran- dom Forest and GBDT which are used.

Random Forest

Random Forest is an ensemble model and it uses bagging technique. In Random Forest, the base learners are Decision Trees which means its final output depends on the output of various Decision Trees. Decision Trees are basically built by performing node split at each level of a tree. Node splits can be done based on the value of Information Gain of the feature. If there are n base learners, then there are n Decision Trees will be built where n is numerical value. Each of n base learners will be built on 'n' different data set after performing row sampling and column sampling on the main data set. The row and column sampling are performed randomly by filtering the number of rows and number of columns from the original dataset. As it is a classification problem, in Random Forest, its output is decided based on the concept of Majority Voting. In this problem, there are 2 classes such as class labels 0 and 1 which represent whether the patient has diabetes or not. So, out of 'n' Decision Trees, it will count the number of Decision Trees produces the output as 0 and let it be n1. It will also count the number of Decision Trees which produces the output as 1 and let it be n2. The output of the random Forest model will be the maximum count of n1 and n2. The following figure demonstrates how the Random Forest algorithm works.

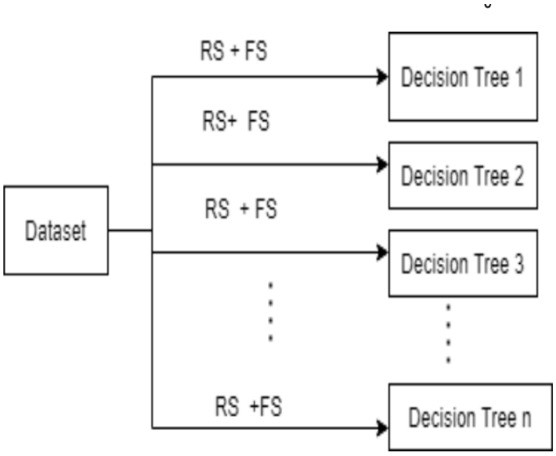

Fig 2. Working of Random Forest

n in the above diagram denotes the number of estima- tors. RS and FS denotes the Row Sampling and Feature Sampling. While building the Random Forest, Row Sampling with replacement and Feature Sampling will be done on the dataset and on top of that, we will build a decision Tree which is our base learners in the Random Forest.

GBDT

Gradient Boosting is a boosting technique. It concentrates on loss function and base learners. In this model, there are M base learners where M is numerical value. In this model, we will compute pseudo-residuals in every iteration and it has to optimize the loss function. Loss functions have to be differentiable. The base learners used are decision trees. So, we can perform the node splits based on the values such as Gini Impurity or Information Gain. So, In this model also, we have to perform Hyper parameter tuning for various parameters such as min samples split, min samples leaf, n_estimators etc., After performing hyperparameter tuning, the best values that we got for hyper-paramters max depth as 8, max features as 3, min samples leaf as 4, min samples split as 5 and n estimators as 750. In this model, basically trees will be added at the end of each iteration in order to minimize the loss function.

7. Performance Metrics

Performance Metrics are used to evaluate the performance of a model by providing some numerical measure as an output. Using the numerical measure output, we can find how well our model works. There are various performance metrics that are proposed for classification and Regression problems. In this paper, we have discussed the performance metrics that we have used in this problem.

Confusion Matrix

This is a binary class classification problem. So, confusion Matrix is a 2*2 matrix. Basically this matrix displays four type of values such as True Positives, True Negatives, False Positive, False Negative.

True Positives denotes the number of data points that are actually positives and also predicted as positives. True Negatives denotes the number of data points that are actually negatives and also predicted as negatives.

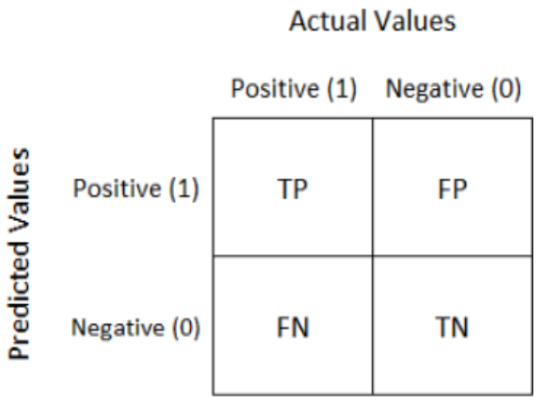

Fig 3. Confusion Matrix

False Positives denotes the number of data points that are predicted as positive but it's actually negative. False Negatives denotes the number of data points that are predicted as negative but it's actually positive.

Ideally False Positives and False Negatives should be very close to 0 for the model to be classified as good model.

Precision

Precision finds "Out of the points that are predicted as positives, how many percent of values are actually positives". The predicted positives can be True Positives or False Positives. Hence we can write Precision formulation as

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (1)$$

Recall

Recall finds "Out of the points that are actually positive points, how many percent of values are predicted as positives". The actual positives can be True Positives or False Negatives.Hence we can write Recall formulation as

$$Recall = \frac{True\ Posititves}{True\ Positives + False\ Negatives} \quad (2)$$

F1-Score

F1-Score is another performance metrics which is used to analyse the performance of the model. The formulation of F1-score is called as

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (3)$$

8. Probabilistic Interpretation

When we predict outputs using Machine Learning models, there is a chance that the patient record might be predicted wrongly. In such cases, the doctor won't be very sure because doctors have to explain to patients that they have this disease because of the particular reason. Since doctors cannot believe the output of the model, the proposed model won't be very much useful. So, I am going to give a probabilistic interpretation of the output. My model will take an image as input and will do all the internal complex steps and after that it will predict the patient record to which class out of the 2 available classes. Also my model will output the probabilistic interpretation of the class. For example, after taking the patient record, if the model predicts that the patient record belongs to class 0 and if it outputs the probability that the patient record belongs to class 0 and class 1 are 0.93 and 0.07. In this example, the model is pretty sure that it belongs to class 0. Let's take another example. if the model predicts the patient record to be class 1 and if it outputs the probability of this patient record belonging to class 0 and class 1 is 0.47 and 0.53. In this case, the doctor after looking at the probabilities can tell that model is not very sure about the output because probabilities are very close. So, after knowing this, doctors can perform manual testing to confirm whether the patient has diabetes or not. So, as we have seen in these examples, probabilities will add a certain value to the output.

9. Results

We have built two models such as Random Forest and GBDT. Out of 330 test data points, in the random forest model, the number of false negatives and false positives are 21 and 5 and the remaining 304 data points were correctly predicted by the model. The confusion matrix is mentioned in Fig 4.

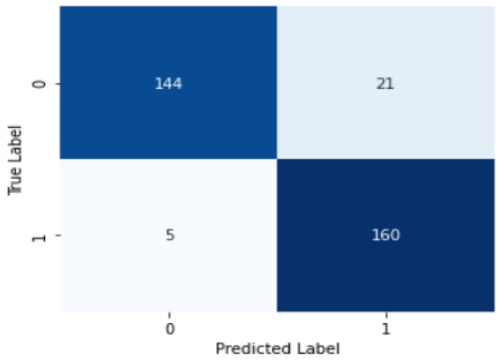

Fig 4. Confusion Matrix for Random Forest Model

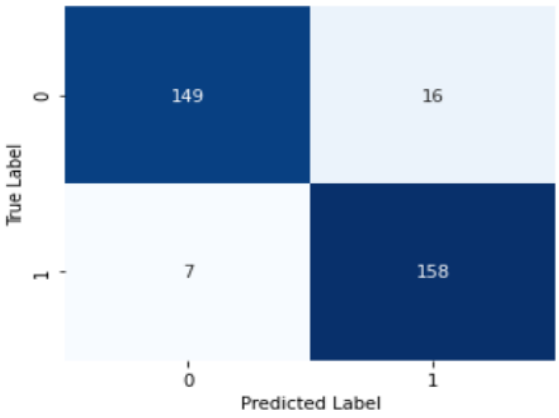

Fig 5. Confusion Matrix for GBDT Model

In the case of GBDT, Out of 330 test data points, False Negatives and False Positives are 16 and 7 and the remaining 307 data points were correctly predicted by the model. With the confusion matrix of both the models, we can say that both the models are good because the false positives and False Negatives are less. If we want to pick one out of these two models, we can say that GBDT is slightly better than Random Forest in terms of the number of confusion matrix of both the models such as Random Forest and GBDT. The confusion matrix of GBDT is mentioned in Fig 5,

The ROC Curve of models such as Random Forest and GBDT is mentioned in Fig 6. ROC Curve again proves that how good both the models are. AUC Scores can range from 0 to 1. In this experiment, we got corresponding AUC scores of Random Forest and GBDT as 0.976 and 0.989. If the AUC Scores of the models are very close to 1, then those models are considered as very good models. With the help of AUC Scores, we can reiterate that GBDT model is slightly outperforming the Random Forest Model.

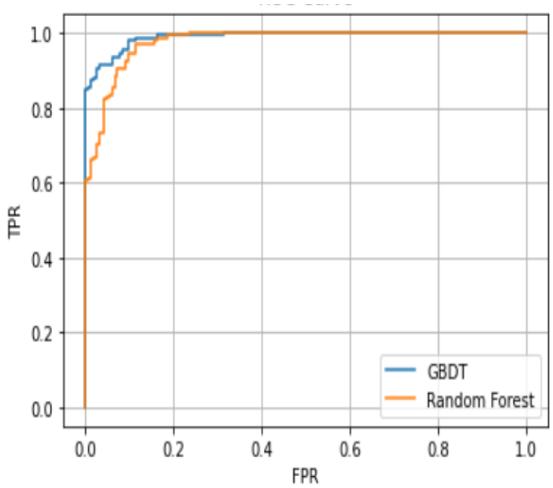

Fig 6. ROC Curve

Table 1. Results of Various Built Models

Models	Class	Precision	Recall	F1-Score
Random Forest	0	0.966	0.872	0.917
Random Forest	1	0.883	0.969	0.924
GBDT	0	0.955	0.903	0.928
GBDT	1	0.908	0.957	0.932

Also, we have produced performance metric output values such as Precision, Recall and F1-Score of two different models such as Random Forest and GBDT in the tabular form. We know that automated systems for healthcare domain should produce good performances and in our experiments, the outputs of all the performance metrics are very good because we have performed many pre-processing techniques before building the actual model on the data.

10. Conclusion

In this paper, we have built the two models such as Random Forest and GBDT. We were able to produce the high level accuracy because of performing various pre-processing techniques and also we have discussed the need of various processes such as Hyperparameter Tuning and K-fold Cross validation. Performance of the models is measured using the various performance metrics values such as Precision, Recall F1-Score. Also, we have highlighted the importance of interpretability of the output and explained how it will be useful to the doctors while analysing the output of machine learning models that are especially built for healthcare problems like Diabetes diagnosis. Even though, both the models such as Random Forest and GBDT produced high accuracy, we could say that the best model out of these two models for this problem is GBDT which we could say based on a number of misclassifications and also with the help of above results..

References

[1] Priyanka Sonar, Prof. K. JayaMalini, ”Diabetes Prediction Using Different Machine Learning Approaches”, Proceedings of the Third International Conference on Computing Methodologies and Communication (ICCMC 2019).

[2] Dominikus Boli Watomakin, Andi Wahju Rahardjo Emanuel, ”Comparision of Performance Support Vector Machine Algorithm and Naive Bayes for Diabetes Diagnosis”, 5th International Con- ference on Science in Information Technology (ICSITech), 2019

[3] Nilam Chandgude, Prof. Suvarna Pawar, ”Diagnosis of Diabetes using Fuzzy Interface System”

[4] MelkyRadja, Andi Wahju Rahardjo Emanuel, ”Performance Evaluation of Supervised Machine Learning Algorithms using Different Data sets sizes for Diabetes Prediction”, 5th International Confer- ence on Science in Information Technology (ICSITech),2019

[5] Maham Jahangir, Hammad Afzal, Mehreen Ahmed, Khawar Khurshid, Raheel Nawaz, ”An Expert System for Diabetes Prediction using Auto Tuned Multi-Layer Perceptron”,Intelligent Systems Conference 2017
[6] Ke Yan, David Zhang, ”A Novel Breadth Analysis for Diabetes Diagnosis”
[7] K. Lakshmi Priya, Mourya Sai Charan Reddy Kypa, Muchu- marri Madhu Sudhan Reddy, G. Ram Mohan Reddy, ”A Novel Approach to predict Diabetes using Naive Bayes Classififer”, Proceedings of the Fourth International Conference on Trends in Electronics and Informatics (ICOEI 2020)
[8] Taiyu Zhu, Pau Herrero, ”Deep Learning for Diabetes : A systematic Review”
[9] Asma A. AlJarullah, ”Decision Tree Discovery for the Diagnosis of Type II Diaebetes”, International Conferences on Innovations in Information Technology 2011.
[10] Dr. S. N. Singh, Komal Kathuria, ”Diabetes Diagnosis using Differ- ent Data Pre-processing Technique”, 24th International Conference on Computing Communication and Automation (ICCCA), 2018
[11] Bo Hang, ”The Research and Implement of Diabetes Diagnosis Expert System”, International Conference on Computer and Communication Technologies in Agriculture Engineering,2010
[12] Roxana Mirshahvalad, Nastaran Asadi Zanjani, ”Diabetes Prediction using Ensemble Perceptron Algorithm”, 9th International Conference on Computational Intelligence and Communication Networks, 2017.
[13] Lina Nachabe, Bachar ElHassan, Dima AlMouhammad, Marc Girod Genet, ”Intelligent System for Diabetes Patients monitoring and assitance”, Fourth International Conference on Advances in Biomedical Engineering (ICABME), 2017.
[14] Muhammad Azeem Sarwar, Nasir Kamal, Wajeeha Hamid, Munam Ali Shah, ”Prediction of Diabetes using Machine Learning Algorithms in Health Care”, Proceedings of the 24th International Conference on Automation Computing, Newcastle Uni- versity,2018.
[15] Amani Yahyaoui, Akhtar Jamil, Jawad Rasheed, Mirsat Yesiltepe, ”A Decision Support System for Diabetes Prediction using Machine Learning and Deep Learning Techniques”