Stacked Generalization Ensemble-Based Hybrid Gradient Boosted Model for Predicting Diabetes

Priyabrata Sahu[1], Jibendu Kumar Mantri[2]
[1]Department of CSE and IT, Indira Gandhi Institute of Technology, Sarang, India
[2]Department of CSE and IT, Indira Gandhi Institute of Technology, Sarang, India
sahu.priyabarta@gmail.com

Abstract : In the modern world, diabetes is a very scary problem. It is a long-term condition that can lead to a number of health problems. It is a grouping of illnesses that cause blood sugar to be too high. Machine learning is being used more and more in the field of health care because of how quickly it is improving. The goal of this study is to find the most accurate way to predict how likely it is that a patient will get diabetes.

This article shows how to make the A Hybrid Stacked Generalization LGBM-XGB Model Based on Ensembles for Diabetes Prediction work well with computers. The suggested method predicts the start of diabetes early on by using a strategy based on stacking generalisation. This method uses the LGBM -Light Gradient Boosting Machine and the EGBM- Extreme Gradient Boosting Machine together (XGB). The Hybrid XGB-LGBM model works by making meta-data from the XGB and LGBM models so that the SMOTE technique for balancing data can be used to figure out the final predictions. Using two datasets from PIDD, the Stacked XGB-LGBM-SMOTE model is tested to see how well it works. The most important things that this study findings are: 1) An enhanced new hybrid ensemble-based approach is made; 2) The data balancing method is used successfully; and 3) A comparison of how well PIDD datasets work with and without data balancing models is done. Case studies have been done to show that the proposed enhanced model works superior to both the current benchmark methods and the hybrid stacked models.

Keywords: Light Gradient Boosting Machine (LGBM), data balancing, Staking approach, Ensemble , Extreme Gradient Boosting machine (XGB).

I. INTRODUCTION

Disease analysis is a difficult aspect of medicine. To determine whether someone has diabetes, just compare their blood sugar level to the desired range. A chronic sickness or ailment is one that lasts a long period or has long-term repercussions. People suffering from these illnesses have an extremely bad quality of life. A global health crisis that affects millions of people every year is diabetes which is prevalent and serious health issues. This long-term the majority of human lives are cut short by illness for people all around the globe. Having a chronic condition comes with a cost. Chronic illnesses are expensive for both the government and the citizens [1,2]. According to global diabetes statistics [3,] over 382 million individuals worldwide were diagnosed with diabetes in 2013. Diabetes is more prevalent in areas where individuals earn more money per capita [4]. Diabetes was discovered and treated in over 451 million persons in 2017. According to estimates, over half of the world's 693 million diabetics would be unaware of their condition by 2045. In 2017, $850 million more was spent on diabetics [5]. Although there hasn't been much research into biological data, technical improvements have made it feasible to examine the data using computers and statistical models. Organizations in the health care industry also gather a large amount of data. New insights may be discovered when data mining technologies are utilised to create models that can learn from what is observed. Data mining, the technique of extracting usable information from massive databases, may aid clinicians in making better judgements [6]. Because so many people utilise information and communication technology, and there is so much digital information and data, a number of initiatives have lately shifted their aims to enhance and update the old method of diabetes prediction.

In this paper, we first investigate whether it is viable to apply a Stacked Generalization method to forecast diabetes by merging XGB and LGBM models with the SMOTE data balancing technique. The accuracy, performance, and efficiency of the Stacked XGB-LGBM-SMOTE model are all excellent. It is also simple to use. Using an open data source

R. Murugan et al. (eds.), *Proceedings of the International Conference on Signal Processing and Computer Vision (SIPCOV 2023)*, Advances in Engineering Research 239,
https://doi.org/10.2991/978-94-6463-529-4_23

to perform simulations revealed that the Stacked XGB-LGBM-SMOTE model predicted diabetes better than other models.

The following are the most notable additions made by the paper:

a. A new hybrid XGB-LGBM-SMOTE model improves general regression performance. Despite learning and being founded on strong Concepts in Mathematics, XGB and LGBM modelling, only tree models will do from the same group. can make fixing built-in mistakes difficult. Meta-data may improve SMOTE training.
b. We improve a unique diabetes prediction algorithm in this study. Most diabetes detection approaches use neural networks or ensemble models. Previous research has ignored the benefits of integrating ensembles with data-balancing models.
c. Five diabetes prediction systems were analysed. Much prior work has focused on improving machine learning models. Most ML users haven't considered approach selection.
d. PIDD datasets test the suggested technique. .
e. Cutting-edge benchmarking methods were compared. The suggested stacked XGB-LGBM-SMOTE model outperformed 11 benchmarks in a rigorous comparison assessment.
The following is how the paper is put together: In Section II of this work, we discuss models that may be used to predict diabetes complications. Section III lays the groundwork for the proposed technique. Case studies are included in Chapter IV. The suggested approach is evaluated using a number of different Machine Learning (ML) models that are already in existence. Research was also conducted for comparing the two approaches utilising the same data set. Section V contains a summary of what was discovered and what was stated about it, while Section VI has a list of conclusions.

II. LITERATURE SURVEY

In the most recent few years, several models for diabetes prediction have been proposed and published. An ML-based framework was proposed in [7], and within that framework, Linear Discriminant Analysis (LDA) [8, 9], AdaBoost (AB) , Decision Tree (DT) [16] ,Quadratic Discriminant Analysis (QDA) [10, 11], Naive Bayes (NB) [12, 13], [14], Logistic Regression (LR) [15], , and Random Forest (RF) [17] were implemented with various techniques for feature extraction and cross-validation. To improve the ML model, they ran several experiments on discarding outliers and filling in missing data. The author in [18] used three different Machine Learning (ML) classifiers, namely State vector machine (SVM), Decision Tree (DT), and Naive Bayes (NB), in order to make the most accurate prediction possible on the likelihood of developing diabetes: They showed that the Naive Bayes (NB) model is the most accurate by giving it an AUC- area under the curve score of 0.819. [19] seeks to categorise diabetes mellitus. It investigates and employs the AB and Ensemble bagging methods in conjunction with J48 (c4.5)-DT as a base learner and stand-alone data mining approach (J48). Their findings suggest that the AB ensemble approach outperforms bagging and solo J48-DT. In [20], genetic programming was applied to predict diabetes, and the resultant framework performed much better than the other approaches tested.

Polat et al. [21] Suggested a Least Square Support Vector Machine (LS-SVM) technique with a 79.16% accuracy as a better way to categorise items than what had previously been done. We employed Generalized Discriminant Analysis (GDA) as a pre-processing step to isolate the disorder's characteristics before using the LS-SVM approach to these variables to identify PIDD. In [22], M. F. Ganji and M. S. Abadeh propose utilising the fuzzy classification using ant colony optimization to determine whether someone has diabetes. The Regression Tree ,Fuzzy Min-Max neural network, , and Random Forest (FMM-CARTRF) integrated hybrid classification algorithm presented by Seera and Lim [23]. Sa'di et al. [24] employed a variety of data mining approaches to categorise PIDD patients. The RBF network and J48 were found to be less accurate than Nave Bayes (76.95%). Bansal et al. [25] developed an evolutionary technique that use the Particle

Swarm Optimization (PSO) methodology to apply the k-Nearest Neighbour (KNN) classification algorithm on the PIDD features. As a result, the solution was 77% correct. Choubey et al. [26] used GA to choose variables for PIDD models. We then utilised these characteristics to train a diabetic Naive Bayes (NB) classifier, which was correct 78.69% of the time.

III. METHODOLOGY

Differentiated by their unique structure, LGBM, XGB, and SMOTE Network are presented as the three ML models covered in this part. The suggested method is also extensively studied.

3.1. LIGHT GRADIENT BOOSTING MODEL

Light Gradient Boosting Model (LGBM) is an ensemble booster model that can strengthen a collection of poorly connected learners [27]. In 2017, Microsoft provided the source code for this algorithm [28]. LGBM basically improves the performance of GBDT-Gradient Boosted Decision Trees classifier without compromising accuracy [29] by lowering memory needs and speeding up calculation. When confronted with a huge data collection, traditional GBDT-based classifier lose accuracy and exhibit significant slowing in forecasting performance. The LGBM model employs a basic principle of histogram-based technique to lessen the influence of data having high dimensions, speeding up computation, and protect the forecasting system from overfitting. The eigenvalues of a continuous random variable are transformed to l integers and used to build a histogram with a limited depth and breadth k. In contrast to XGB, LGBM employs a Decision Trees (DT) technique that is based on pre-sorted data. Parallel learning with the help of a parallel voting DT is also utilised for LGBM training. Because of this, the model may now learn in parallel. The top-k samples are chosen by dispersing the initial samples over many trees and then using Local Voting Decision (LVD). Further it is shown that the result of global voting considers the top k LVD traits to determine the top 2k characteristics for k iterations. LGBM employs the Leaf-wise technique to find optimum leaves as part of its optimization process. It is to be noted that the objective function of LGBM is given by [30]:

$$Obj(t) = L(t) + \Omega(t) + c(1)$$

In this equation (t) represents the regular function, L(t) is basically the loss function, c indicates the extra parameter, and t represents sampling interval. By altering the depth of the tree, over fitting may be avoided with the use of the supplementary parameter c. The model's regular function demonstrates its complexity. LGBM is distinguished from the remainder of GBDT with regard to the acceleration of computational effort and the model's practicability based on the parameter L (t). Loss is a quantification of the model's predictive power by comparing the observed value of yi to its expected.

output yi given N samples, as explained in [30]:

$$L(t) = \sum_{n=1}^{n} (y_i(t) - (\hat{y})_i(t))^2 \qquad (2)$$

By linking the regression trees in a series, the residual information from the previous learners is passed on to the next in the chain. The final output yi is produced by the aggregate of these remaining trees.

3.2. Extreme Gradient Boosting Model (XGB)

Basically, the XGB classifier is a popular method for dealing with prediction difficulties [31]. The XGB constructs a powerful regressor using an ensemble of DT. This massively parallel ML technique is already configured to leverage parallelism with numerous threads, which reduces runtime [31]. To calculate loss, GBDT models utilise the first-order Taylor expansion, while XGB models use the second-order Taylor expansion. The objective function of XGB additionally takes regularisation factors into consideration, such as tree depth and leaf node weights. As a result, the number of iterations may decrease while the efficiency of generating trees may increase. To make the model more understandable, we use a technique known as "decision tree development," which operates on many levels. The target function is the same in both XGB and LGBM. [31] provides the XGB model's loss function:

$$L(t) \approx \sum_{n=1}^{n} (L(y_i, \hat{y}_{i-1}) + g_i f_i(x_i) + \frac{1}{2} h_i f_i^2(x_i \quad (3)$$

In this equation basically , g_i and h_i are related functions given by equations (4) and (5) respectively [31]:

$$g_i = f^{'}(t) = \frac{\partial L(y_i, \hat{y}^{t-1})}{\partial \hat{y}^{t-1}} \quad (4)$$

$$h_i = f^{"}(x) = \frac{\partial^2 L(y_i, \hat{y}^{t-1})}{\partial \hat{y}^{(t-1)}} \quad (5)$$

3.3 SMOTE Algorithm Working Procedure*:*

SMOTE [32] is a statistical approach for boosting the sample size of underrepresented populations systematically. The system generates new instances based on already existing minority groups.

First, Minority status in its early stages , the k-closest neighbours of x are determined by finding the Euclidean distance between x and each example in set A.
Second, the degree of imbalance, given by N, dictates the frequency of testing. N models (x1, x2,... xn) are chosen at random from each individual's k closest neighbours, and these N models are utilised to form the set.
Third, by entering each model into the appropriate equation (k=1, 2, 3,...., N), a new model is formed. The rand(0, 1) function returns an irrational integer between 0 and 1.

3.4. Hybrid Gradient Boosted Model

The proposed model in this study is a powerful meta-learner built from XGB, LGBM, and SMOTE networks. The phrase "stacked generalisation" (Stacking) refers to a more sophisticated nonlinear method of associating models [33]. We must enhance accuracy so that the forecasting system, this combination technique uses non-linear weightings for the early predictors. In many cases, computer simulations demonstrate that the Stacking method outperforms other kinds of base learners [34, 35]. The Stacked design has two layers, as seen in Fig. 1b. (Level 0 and level 1).

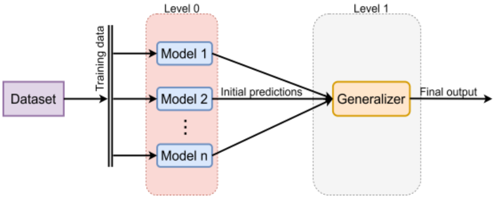


Figure 1: Graphical Model of the stacking Hybrid approach.

The foundation layer [33] of the architecture is in charge of providing temporal predictions via a set of learners. At level 0, all biases in the system are expected to be present. Predictions are sent into a Meta-learner in the first layer, which then employs a Cross-Validation (CV) approach to generate prediction outputs. Using this method, the output results of the first level of generalisation will be filtered away. The forecasting system is trained in the following manner: The authors assume N training set samples (Si,yi), with 1 I N representing the sampling time. The training data is divided at random. into r folds of nearly similar size to create (Si,yi)k, where k = (1,...,r) is the fold number. The $(S_i, y_i)_k$ satisfy the condition as mentioned as follows.

$$\begin{cases} (S, y)_k \cup \overline{(S, y)_k} = (S, y) \\ (S, y)_k \cap \overline{(S, y)_k} = \oslash \end{cases} \quad (7)$$

Level 0 utilises the N/r part to complete the dataset (Si,yi)k. The novice learners (L1,...,LN) construct predictions about Y I using S/Si feature vectors. At this step, weak learners are employed to compute the test component Si. The output with the real dataset yi causes the meta-level dataset MSi to be reshaped with a new feature vector as a consequence of the interaction. Meta-learning is used to generate a meta-level vector from the MSi dataset's fundamental predictors. In this case , the Stacking concept employed in level-0, as shown in Fig. 2, consists of two training models, each with five folds.

Figure 2 shows XGB and LGBM being used to represent level-0 learners, while an MLP model is being used to represent the conceptual. The basic concept behind the proposed approach is to develop a multimodal diabetes prediction system via the hierarchical coupling of ensemble methods and data balancing through the SMOTE model. Because of their excellent track records in a wide variety of applications requiring forecasting, XGB and LGBM are an obvious option to serve as foundation models. We hypothesise that as the prediction skill of each heterogeneous base model for diabetes improves, so will the overall performance of the stacking ensemble. The objective of this research is to build a specialised prediction system, in this instance for diabetes prediction, that can manage the underlying system's intrinsic nonlinearity. When selecting how to combine XGB and LGBM at level 0, consider the input parameters like glucose_level, BMI, blood_pressure (BP), and skin_thickness. Both of these methods use an Out-Of-Fold (OOF) methodology to create test set predictions. The training component uses a CV sub-fold for fashion training in order to reduce overfitting and different learnings. Next-level parameters are inputs used in forecasting. Temporary predictions from the training set are fit using the SMOTE model at level 1.

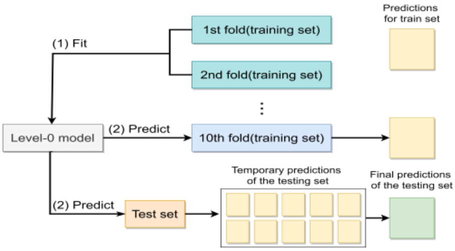


Figure 2: Schematic of layered architectural framework level-0 made from basic learners: XGB and LGBM.

After training, the forecasting approach just assesses model output data. To accommodate all sub-folds with separate base learners and meta-learners, the training approach needs more time. Total stacking and therefore, using more complex predictors does not always provide more accurate outcomes. The detailed model synoptic is shown in Fig. 3.

Feature engineering, forecasting, object recognition, and evaluation are the four processes shown in Fig. 3. Feature engineering involves cleansing, extracting, and selecting data. Among the factors used to determine an item are glucose, BMI, blood pressure, and skin thickness. Here, problems are conceptualised, and data is split into training and testing. For each learner and meta-learner, ML model hyperparameters are adjusted. When anticipating, level-0 stacking teaches XGB and LGBM. The SMOTE model balances the outputs of meta-data. The output is determined from the attribute inputs by the meta-learner. The process flowchart is shown in Fig. 4. In order to assess the effectiveness of the forecasting system, K-fold cross-validation, point forecasting, , and visualisation graphs are used.

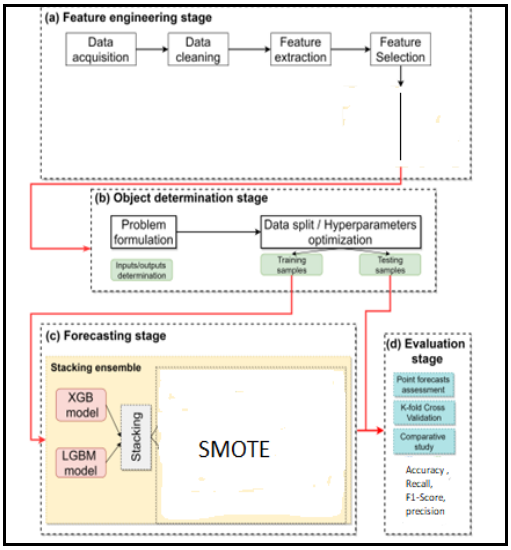


Figure 3: Stack ensemble forecasting framework block diagram.

IV. CASE STUDIES AND PERFORMANCE ASSESSMENT

Real-world case studies were run using PIDD datasets to evaluate the suggested method and showcase the hybrid model's prediction capabilities.

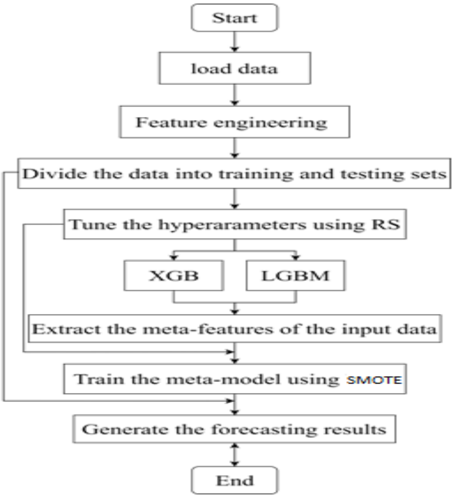

Start
load data
Feature engineering
Divide the data into training and testing sets
Tune the hyperarameters using RS
XGB
LGBM
Extract the meta-features of the input data
Train the meta-model using SMOTE
Generate the forecasting results
End

Figure 4: Stacked XGB-LGBM-MLP method flowchart.

The most up-to-date standards are also used for comparison. Furthermore, the suggested method's excellent accuracy with the existing methodology for the same dataset has been confirmed.

4.1. Data Processing and its Analysis: Both the accuracy of the input data and the sophistication of the underlying forecasting engine contribute significantly to the success of a given forecasting system. Therefore, improving the quality of the system's data necessitates the use of data analysis and feature engineering. PIDD datasets are used to test the effectiveness of the suggested method. The information used in this case study was obtained from PIDD's publicly available database.

4.2. Hyperparameter optimization

This study examines the best diabetes prediction technique, Randomized Search (RS). Trial and error determined hyperparameter values. To achieve this, we review the model outcomes and optimise its parameters. Manual tuning takes time and typically doesn't work well unless the tuner is experienced. RS is an accelerated Grid search technique for choosing hyperparameters. . RS simplifies and parallelizes complicated models [36] . We have this difficulty since gradient optimization doesn't affect each iteration. RS uses discontinuous functions. Unfortunately, RS may miss ideal values during random iteration. A larger RS sampling method value range may solve this problem.

4.3. Evaluation Criteria

Due to the expanding number of Machine Learning classifiers for well-defined prediction tasks, choosing the right strategy to manage uncertainty and seasonality of weather parameters is challenging. While many ML algorithms are optimised for certain types of structured data, they are not able to be generalized to unrelated tasks or datasets. Model selection requires a lot of computational labour, and it's hard to decide what criteria should be used to evaluate a model's performance. . ML approaches' accuracy, computational ease, and speed must continually be tested. This study uses graphical representations, point forecast analysis, and K-fold cross-validation to evaluate criteria objectively and consistently.

4.5. Numerical Evaluation for case study

We examine the recommended model's performance over many future horizons. . The model was built using Python modules like Scikit-learn, LightLBM, and XGBoost [36, 37]. Layered generalisation was built using Vecstack [38]. RS's Hyper parameter optimization exhaustively selected the ideal parameters to increase forecasting accuracy. Model training and comparisons to standalone models (XGB, SMOTE, and LGBM) assess the technique [39]. The novel approach was initially compared against these methods. . Ten simulations were done to verify the forecasting system. To compare model performance, a single-step intuitive ten-fold CV (10-CV) was done. Stacked XGB-LGBM-SMOTE exceeded benchmarks in F1 score, recall, confusion matrix, and accuracy.

V. RESULTS AND DISCUSSIONS:

The basic aim of this research work is to classify / predict whether a patient is prone to diabetes depending on multiple features. It is a binary classification with multiple numerical features. Binary Classification is a discussion post that describes the approach required to deal with Binary Classification problems explained that contain work done on datasets with easy and understandable code explanation.

Steps of results analysis :

- Dataset Information
- Exploratory Data Analysis (EDA)
- Summary of EDA
- Feature Engineering
- Modelling
- Conclusion

The processes used in EDA are :

- Data Visualization
- Data Scaling
- Statistical Tests for Feature Selection
- Modelling and visualization of results for algorithms
- Difference in model performances when trained on balanced and unbalanced data.
- Stacking of classifiers

5.1 Dataset information **:**
Dataset Attributes are basically- “Pregnancies, Glucose, Blood Pressure (BP) ,Skin_Thickness, Insulin, Body mass Index (BMI), Diabetes_Pedigree_Function, Age, and Outcome” .

5.2 Exploratory Data Analysis
Initial steps for Exploratory Data Analysis is this Dividing features into Numerical and Categorical
Numerical Features: Here, categorical features are defined if the the attribute has less than 6 unique elements else it is a numerical feature. The datatypes of the pieces that make up each characteristic may also serve as a basis for a strategy that is often used for this split of features. For this dataset, as the number of features is less, we can manually check the dataset as well. Making a complete duplicate of the dataset and label encoding the text data for the categorical characteristics. The deep copy will not reflect changes made to the source dataset. Therefore, we use this numerically accurate deep copy of the information for both visualisation and modelling.

5.3 Summary of EDA : Here , basically Summary of exploratory data Analysis (EDA) related to Order / Values of features for positive cases of disease under indication is briefly explained :
Domain Information:

- “Pregnancies, Glucose, Blood_Pressure , Skin_Thickness, Insulin, BMI, Diabetes_Pedigree_Function, Age , and Outcome”
- All the information mentioned is gathered from websites and research papers. We will use this information for cross checking the summary of EDA and feature selection.
- Range of values obtained from the EDA slightly miss match the Domain Information for the features : Glucose, Blood Pressure, and Insulin.
- Thus, we will carry out the feature engineering process, balance the dataset using SMOTE analysis and record the difference between the model performances when trained on balanced and unbalanced datasets.

5.4 Feature Engineering
5.4.1 Data Scaling **:**
Machine learning model does not understand the units of the values of the features. It treats the input just as a simple number but does not understand the true meaning of that value. Thus, it becomes necessary to scale the data. We have 2 options for data scaling : 1) Normalization 2) Standardization. As most of the methodology assume the data to be normally (Gaussian) distributed, Normalization is done for features whose data does not display normal distribution and standardization is carried out for features that are normally distributed where their values are huge or very small as compared to other features.

- Normalization : Pregnancies, Insulin, DiabetesPedigreeFunction and Age feat ures are normalized as they displayed a right skewed data distribution. Blood Pressure, Skin Thickness, Glucose & BMI highlight a bimodal data distribution.
- Standardization : None of the features are standardized for the above data.

5.4.2 Data Balancing using SMOTE :
In order to cope with unbalanced data, there are 2 options a. Under sampling : Trim down the majority samples of the target variable.
b. Oversampling : Increase the minority samples of the target variable to the majority samples. In this case, we will oversample the minority class. Due to the use of synthetic data, we cannot evaluate the model’s using accuracy. We have duplicated the data, thus

using accuracy would be misleading for evaluating the model. We will use the confusion matrix, ROC-AUC graph-score for model evaluation. ROC-AUC gives us the relation between True Positive and False Positive rate.

5.5 Modelling

The tests were broken down into 75 percent train data and 25 percent test data. The results obtained for Confusion Matrix and Classification Report is shown as under. *Here three different classifiers are used i.e XGBOOST* . lightgbm and stacking of these two classifiers. With and without Data Balancing.

5.5.1 XGBoost Classifier :

Unbalanced Dataset **:** As per figure 12 and table 5.

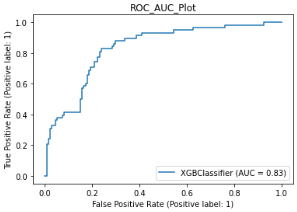
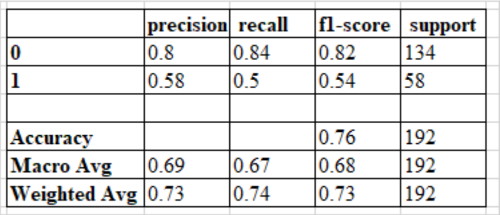


Figure 12: XGBoost- Unbalanced data set

Table 5: XGBoost- Unbalanced data set



	precision	recall	f1-score	support
0	0.8	0.84	0.82	134
1	0.58	0.5	0.54	58
Accuracy			0.76	192
Macro Avg	0.69	0.67	0.68	192
Weighted Avg	0.73	0.74	0.73	192

Balanced Dataset **:** As per figure 6 and Table 6

Cross Validation Score : 86.35% ,ROC_AUC Score : 75.66% .

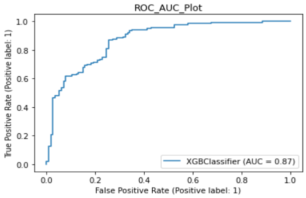
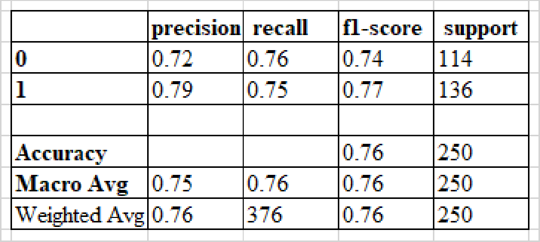


Figure 6: XGBoost- Balanced data set

Table 6: XGBoost- Balanced data set



	precision	recall	f1-score	support
0	0.72	0.76	0.74	114
1	0.79	0.75	0.77	136
Accuracy			0.76	250
Macro Avg	0.75	0.76	0.76	250
Weighted Avg	0.76	376	0.76	250

5.5.2 LightGBM Classifier :

Unbalanced Dataset : Cross Validation Score : 82.36% , ROC_AUC Score : 64.20% (Figure 7 and Table 7).

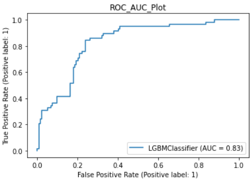
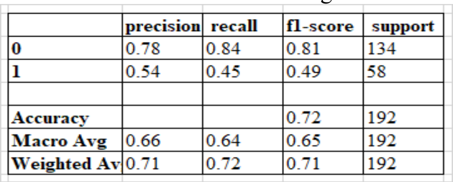


Figure 7: LightGBM- Unbalanced data set

Table 7: LightGBM- Unbalanced data set



	precision	recall	f1-score	support
0	0.78	0.84	0.81	134
1	0.54	0.45	0.49	58
Accuracy			0.72	192
Macro Avg	0.66	0.64	0.65	192
Weighted Av	0.71	0.72	0.71	192

Balanced Dataset :

Cross Validation Score : 86.02% ,ROC_AUC Score : 76.97% (Figure 8 and Table 8).

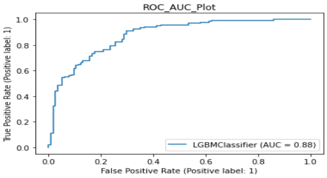
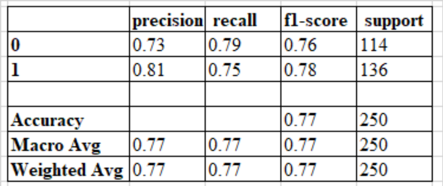


Figure 8: LightGBM- Balanced data set

Table 8: LightGBM- Balanced data set



	precision	recall	f1-score	support
0	0.73	0.79	0.76	114
1	0.81	0.75	0.78	136
Accuracy			0.77	250
Macro Avg	0.77	0.77	0.77	250
Weighted Avg	0.77	0.77	0.77	250

5.5.3 Stack of XGB Classifier and LightGBM Classifier **:** For stacking of classifiers, we stack the above 2 classifiers i.e XGB Classifier and LightGBM Classifier . It has an important hyperparameter known as final estimator. It is the final classifier that makes the final prediction by using the predicted classes by the various classifier and predicts the final output.

Unbalanced Dataset : Cross Validation Score : 80.15%

ROC_AUC Score : 65.70% (Figure 9 and Table 9).

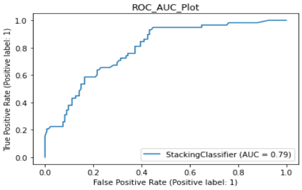
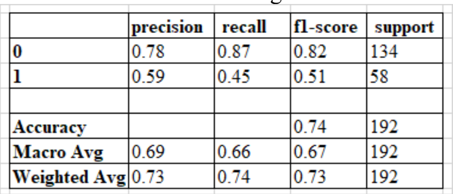


Figure 9: Stack of XGBClassifier and LightGBM- Unbalanced data set

Table 9: Stack of XGBClassifier and LightGBM- Unbalanced data set



	precision	recall	f1-score	support
0	0.78	0.87	0.82	134
1	0.59	0.45	0.51	58
Accuracy			0.74	192
Macro Avg	0.69	0.66	0.67	192
Weighted Avg	0.73	0.74	0.73	192

Balanced Dataset : Cross Validation Score : 84.19% ,ROC_AUC Score : 79.25% (Figure 10 and Table 10).

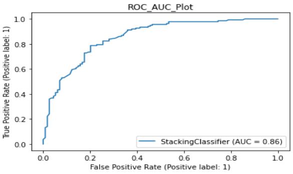
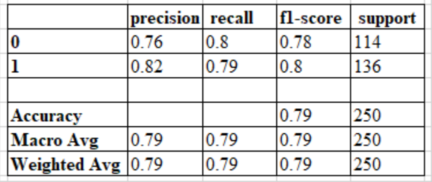


Figure10: Stack of XGBClassifier and LightGBM- balanced data set

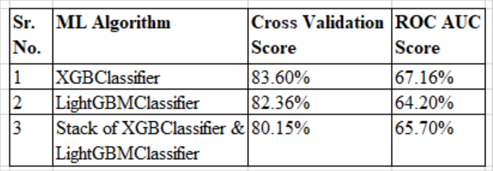
Table 10: Stack of XGBClassifier and LightGBM- balanced data set



	precision	recall	f1-score	support
0	0.76	0.8	0.78	114
1	0.82	0.79	0.8	136
Accuracy			0.79	250
Macro Avg	0.79	0.79	0.79	250
Weighted Avg	0.79	0.79	0.79	250

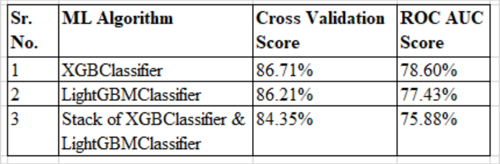
5.5.5 Algorithm Results Table :

Table 11a: overall results Unbalanced Dataset



Sr. No.	ML Algorithm	Cross Validation Score	ROC AUC Score
1	XGBClassifier	83.60%	67.16%
2	LightGBMClassifier	82.36%	64.20%
3	Stack of XGBClassifier & LightGBMClassifier	80.15%	65.70%

Table 11b: overall results balanced Dataset.



Sr. No.	ML Algorithm	Cross Validation Score	ROC AUC Score
1	XGBClassifier	86.71%	78.60%
2	LightGBMClassifier	86.21%	77.43%
3	Stack of XGBClassifier & LightGBMClassifier	84.35%	75.88%

5.6 Overall results **:**

The most important takeaways from this study are outlined and summarised below. As shown in table 11 , Without data balancing XGB Classifier gives cross validation score of 83.6 % and ROC AUC Score of 67.16% LGBM Classifier gives cross validation score of 82.36 % and ROC AUC Score of 64.2% .Stacked XGB-LXGB Classifier gives cross validation score of 80.15 % and ROC AUC Score of 61.20% and accuracy of 74% With data balancing XGB Classifier gives cross validation score of 86.71 % and ROC AUC Score of 78.6% . LGBM Classifier gives cross validation score of 86.21 % and ROC AUC Score of 77.4% .Stacked XGB-LXGB Classifier gives cross validation score of 84.35 % and ROC AUC Score of 74.88 and accuracy of 79%.

VI. CONCLUSION

This research work presents a unique computational framework for Diabetes prediction based on the stacking generalisation method. The suggested method incorporates three powerful ways to boost the performance of individual techniques: Light Extreme Gradient Boosting (LGBM), Extreme_Gradient_Boosting(XGB), , and SMOTE models. Individual performance, training duration, and implementation simplicity are all considered while choosing the model's components. Extensive experimental evidence is shown proving the efficacy of the Stacked XGB-LGBM-SMOTE model on Standard PIDD datasets. The key findings of this paper are :Without data balancing, Stacked XGB-LXGB Classifier gives cross validation score of 80.15 % and ROC AUC Score of 61.20% and accuracy of 74%

With data balancing, Stacked XGB-LXGB Classifier gives cross validation score of 84.35 % and ROC AUC Score of 74.88 and accuracy of 79%. SMOTE analysis is used for data balancing. In order to highlight the performance difference when trained on unbalanced and balanced data, tree based models are trained. Performance of LGBM and XGB is neck to neck however the stack of the 2 classifiers did not outperform the other classifiers! When models are trained with a balanced dataset, a significant performance boost is seen in general.

REFERENCES

1. Falvo D, Holland BE. “Medical and psychosocial aspects of chronic illness and disability”. Jones & Bartlett Learning; 2017.
2. Skyler JS, Bakris GL, Bonifacio E, Darsow T, Eckel RH, Groop L, et al. “Differentiation of diabetes by pathophysiology, natural history, and prognosis”. Diabetes 2017;66:241–55.
3. Tao Z, Shi A, Zhao J. “Epidemiological perspectives of diabetes. Cell “ Biochem Biophys 2015;73:181–5.
4. Organization WH. World health statistics 2016:”Monitoring health for the SDGs sustainable development goals”. World Health Organization; 2016.
5. Cho N, Shaw J, Karuranga S, Huang Y, da Rocha Fernandes J, Ohlrogge A, et al. “IDF Diabetes Atlas: global estimates of diabetes prevalence for 2017 and projections for 2045”. Diabetes Res Clin Pract 2018;138:271–81.
6. Diwani S, Mishol S, Kayange DS, Machuve D, Sam A. “Overview applications of data mining in health care: the case study of Arusha region‖”. Int J Comput Eng Res 2013;3:73–7.
7. M. Maniruzzaman, M. J. Rahman, M. A. M. Hasan, H. S. Suri, M. M. Abedin, A. El-Baz, and J. S. Suri, “Accurate diabetes risk stratification using machine learning: role of missing value and outliers,” Journal of Medical Systems, vol. 42, no. 5, pp. 92, May 2018.
8. G. J. McLachlan, “Discriminant analysis and statistical pattern recognition,” Journal of the Royal Statistical Society: Series A (Statistics in Society), vol. 168, no. 3, pp. 635-636, Jun. 2005.
9. T. M. Cover, “Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition,” IEEE Transactions on Electronic Computers, vol. 14, no. 3, pp. 326-334, Jun. 1965.
10. G. I. Webb, J. R. Boughton, and Zhihai Wang, “Not So Naive Bayes: Aggregating one-dependence estimators,” Machine learning, vol. 58, no. 1, pp. 5-24, Jan. 2005.
11. S. B. Belhouari and A. Bermak, “Gaussian process for nonstationary time series prediction,” Computational Statistics & Data Analysis, vol. 47, no. 4, pp. 705-712, Feb. 2004.
12. C. Cortes and V. Vapnik , “Support-vector networks,” Machine Learning, vol. 20, pp. 237-297, Sep. 1995.
13. A. Reinhardt and T. Hubbard, “Using neural networks for prediction of the subcellular location of proteins,” Nucleic Acids Research, vol. 26, no. 9, pp. 2230-2236, Mar. 1998.
14. B. Kégl, “The return of AdaBoost. MH: Multi-class Hamming trees,” arXiv:1312.6086, Dec. 2013.
15. T. BP and H. WH, “A multivariate logistic regression equation to screen for diabetes: development and validation,” Diabetes Care, vol. 25, no. 11, pp. 1999-2003, Nov. 2002.
16. I. Jenhani, N. B. Amor, and Z. Elouedi, “Decision trees as possibilistic classifiers,” International Journal of Approximate Reasoning, vol. 48, no. 3, pp. 784-807, Aug. 2008.
17. L. Breiman, “Random forests,” Machine Learning, vol. 45, no. 1, pp. 5-32, Oct. 2001.
18. D. Sisodia and D. S. Sisodia, “Prediction of diabetes using classification algorithms,” Procedia Computer Science, vol. 132, pp. 1578-1585, Jan. 2018.
19. S. Perveen, M. Shahbaz, A. Guergachi, and K. Keshavjee, “Performance analysis of data mining classification techniques to predict diabetes,” Procedia Computer Science, vol. 82, pp. 115-121, Mar. 2016.
20. M. Pradhan and G. R. Bamnote, “Design of Classifier for Detection of Diabetes Mellitus Using Genetic Programming,” in Proc. third International Conference on Frontiers of Intelligent Computing: Theory and Applications, Nov. 2015, pp. 763-770.
21. K. Polat, S. Güneş, and A. Arslan, ‘‘A cascade learning system for classification of diabetes disease: Generalized discriminant analysis and least square support vector machine,’’ Expert Syst. Appl., vol. 34, no. 1, pp. 482–487, 2008.
22. M. F. Ganji and M. S. Abadeh, ‘‘A fuzzy classification system based on ant colony optimization for diabetes disease diagnosis,’’ Expert Syst. Appl., vol. 38, no. 12, pp. 14650–14659, 2011.
23. M. Seera and C. P. Lim, ‘‘A hybrid intelligent system for medical data classification,’’ Expert Syst. Appl., vol. 41, no. 5, pp. 2239–2249, 2014.
24. S. Sa’di, A. Maleki, R. Hashemi, Z. Panbechi, and K. Chalabi, ‘‘Comparison of data mining algorithms in the diagnosis of type II diabetes,’’ Int. J. Comput. Sci. Appl., vol. 5, no. 5, pp. 1–12, 2015.
25. R. Bansal, S. Kumar, and A. Mahajan, ‘‘Diagnosis of diabetes mellitus using PSO and KNN classifier,’’ in Proc. Int. Conf. Comput. Commun. Technol. Smart Nation (IC3TSN), Oct. 2017, pp. 32–38.
26. D. K. Choubey, S. Paul, S. Kumar, and S. Kumar, ‘‘Classification of Pima indian diabetes dataset using naive Bayes with genetic algorithm as an attribute selection,’’ in Proc. Int. Conf. Commun. Comput. Syst. (ICCCS), Feb. 2017, pp. 451 455.

27. Y. Ju, G. Sun, Q. Chen, M. Zhang, H. Zhu, M. U. Rehman," A model combining convolutional neural network and lightgbm algorithm for ultra-short-term wind power forecasting", IEEE Access7(2019)28309–28318. doi:10.1109/ACCESS.2019.2901920.
28. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu,"Lightgbm: A highly efficient gradient boosting decision tree", in: Advances in neural information processing systems, 2017, pp. 3146–3154.
29. Y. Ju, G. Sun, Q. Chen, M. Zhang, H. Zhu, M. U. Rehman," A model combining convolutional neural network and lightgbm algorithm for ultra-short-term wind power forecasting", IEEE Access7(2019)28309–28318.doi:10.1109/ACCESS.2019.2901920.
30. Q. Meng, G. Ke, T. Wang, W. Chen, Q. Ye, Z. M. Ma, T. Y. Liu, "A communication efficient parallel algorithm for decision tree", Advances in Neural Information Processing Systems (2016) 1279–1287. arXiv:1611.01276.
31. T. Chen, C. Guestrin, "Xgboost: A scalable tree boosting system", in: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794.
32. Nitesh V. Chawla ,et el ,"SMOTE: synthetic minority over-sampling technique" , Journal of Artificial Intelligence Research Volume 16 Issue 1 January 2002 pp 321–357
33. D. H. Wolpert, "Stacked generalization", Neural Networks 5 (1992) 241–259. doi:10.1016/S0893-6080(05)80023-1.
34. Z. Ma, Q. Dai, "Selected an Stacking ELMs for Time Series Prediction", Neural Processing Letters 44 (2016) 831–856. doi:10.1007/s11063-016-9499-9.
35. X. Luo, J. Sun, L. Wang, W. Wang, W. Zhao, J. Wu, J. H. Wang, Z. Zhang, "Short-term wind speed forecasting via stacked extreme learning machine with generalized correntropy", IEEE Transactions on Industrial Informatics 14 (2018) 4963–4971. doi:10.1109/TII.2018.2854549.
36. Simon Blanke, "Hyperactive: A hyperparameter optimization and meta-learning toolbox for machine /deep-learning models"., https://github.com/SimonBlanke, since 2019.
37. Moez Ali, Home - PyCaret, 2020. URL: https://pycaret.org/.
38. I. Ivanov,"vecstack: Python package for stacking (machine learning technique)", URL: https://github.com/vecxoz/vecstack.
39. S. Seabold, J. Perktold, "Statsmodels: Econometric and Statistical Modeling with Python", Proceedings of the 9th python in science conference (2010).