A Machine Learning Approach to Early Detection of Anxiety Disorder

Mst. Md. Sharfuddin
Department of Computer Science and Engineering,
Southeast University, Dhaka, Bangladesh

Momana Akter Mim
Department of Diploma in Nursing Science and Midwifery,
East West Nursing College and Institute

Fahad Ahmed
Department of Computer Science and Engineering,
Southeast University, Dhaka, Bangladesh

Abstract

Anxiety disorder is now a common mental disorder worldwide. Its early prediction is a very complex process due to the clinical method. Here, the problem of delayed diagnosis has been solved by creating an early detection framework through a machine learning model, which is capable of detecting the level of anxiety. Although various models are used in ongoing studies, there is a shortage in achieving computational efficiency while maintaining high reliability on the dataset. We solved this problem by running 9 different machine learning models. Which range from conventional classifiers to more advanced gradient boosted ensembles. In our method, data pre-processing, encoding and class balancing have been done very carefully. As a result, the reliability of the said model can be ensured. Here, the results show that Random Forest gave more and better output than all other models. And gave 86.62% accuracy and highest precision, recall and F1 score output. Here MSE was only 13.38%. Then ensemble models like Support Vector Machine and Logistic Regresion also showed the second & third highest accuracy of 83.44%. The benefit of this study here is that gradient boosting frameworks like LightBGM are a very reliable solution for anxiety level detection. Which will help doctors in quick prediction and personalized treatment planning.

1. Introduction

4.4% of the world's population of all genders suffers from anxiety disorders. According to a 2021 report, 359 million people suffer from anxiety disorders.Only 27.06% of the world's people receive treatment for anxiety disorders. On average, only 1 in 4 people receive treatment [16] . Observations show that 2.1% of people die due to anxiety disorders in an average of 9.7 years, which is 1066 [15] . If early predictions are possible by analyzing the relationship between different mental health conditions such as depression, bipolar disorder, and substance use, public health policymakers will be able to take more effective measures. The main objective of this study is to extract meaningful information from complex datasets using advanced machine learning models. To address this problem, we used a multi-feature predictive modeling approach. Using mental health data from people in different countries, various models ranging from Support Vector Machines (SVM) and Logistic Regression (LR) to modern boosting algorithms such as XGBoost and LightGBM were applied to obtain a highly accurate model. Previous studies have mainly focused only on depression and general linear models. The effects of other components of mental health such as drug use and alcohol disorders have also been taken into account. There is a clear lack of comparative research on modern gradient boosting algorithms. The use of

DOI: https://doi.org/10.5281/zenodo.20780393

ensemble models is not very common, especially on large datasets. We used 10 important features to build high-performance models. Not only existing features, but also new variables have been created through feature engineering, such as depression and drug use composite scores. We have created a comprehensive evaluation framework by verifying the performance of 9 different machine learning models. Among these models, 86.62% accuracy has been achieved through Random Forest. Detailed comparative analysis of 9 different models has been performed. Intelligence has been tested by best 13 features and used Shap for define feature impact to the data analysis side, which has a direct impact on increasing the decision-making ability of the model.
machine learning models. To address this problem, we used a multi-feature predictive modeling approach. Using mental health data from people in different countries, various models ranging from Support Vector Machines (SVM) and Logistic Regression (LR) to modern boosting algorithms such as XGBoost and LightGBM were applied to obtain a highly accurate model. Previous studies have mainly focused only on depression and general linear models. The effects of other components of mental health such as drug use and alcohol disorders have also been taken into account. There is a clear lack of comparative research on modern gradient boosting algorithms. The use of ensemble models is not very common, especially on large datasets. We used 10 important features to build high-performance models.

2. Related Work

Machine learning and classification algorithms are the backbone on which most of modern data science research is built. Many recent research work has proven the efficacy of Ensemble Learning methods in addressing the drawbacks of individual approaches and improving prediction and classification accuracy.

L Breiman (2001) [1] **:** Hierarchical predictive modeling technique showing that combining ensemble of decision trees reduces overfitting He employed form of "bagging" where trees are created independently from bootstrap samples in the data. Its major strength is functionality with large complex datasets and its main weakness is lower prediction speed for very large models.

Cortes and Vapnik (1995)[2]**:** Support Vector Machines Paper, Foundations of Statistical Learning: It was a paper that introduced SVMs for separating data using hyperplanes. They used the kernel trick to interpolate non-linear data into a linear space. It performed well (92–95% accuracy) on high-dimensional data, but was computationally expensive with larger datasets.

Dietterich (2000) [3]**:** Conducted one of the earliest large-scale surveys on ensemble learning and proved that ensemble methods are better for both reducing bias and variance. He demonstrated that ensemble models are 5–10% more accurate than individual models with a voting mechanism, but the ensemble model is less interpretable compared to a single model.

Freund, Schapire (1997) [4] **:** Boosting to enabling weak learners. They used the method of weighted error reduction, in which for each subsequent model you correct the errors of the last one. They get close to 96% accuracy with their models but suffer from overfitting in noisy data.

Mehmood (2025) [5] :qdisho python on apple quality prediction using KNN and Decision Trees. He found that his best model (KNN) achieved an accuracy of around 89% but one possible limitation was the small dataset, which may impact on generalizability.

3.Florea (2025) [6] : Examined the distillation of tabular data and constructed a hybrid model of Logistic Regression and SVM, attained 87% accuracy. One of the limitations mentioned was that hyperparameter tuning required high computational power.

Tschalzev et al. 2024 [7]**:** Benchmarked tabular data showing that ensemble model performance is greatly affected by feature engineering. They also determined that XGBoost was the best model (94% accuracy)

DOI: https://doi.org/10.5281/zenodo.20780393

but this is very complex & an unmanageable black box for general users.

Rokach (2010) [8] **:** An ensemble is defined, mix of RF + SVM with use of "stacking". The ensemble models he created use stacking to increase prediction accuracy, but this method takes a lot of memory since more layers (and data) are involved.

Altman (1992) [9] **:** K-Nearest Neighbors (KNN) with distances for non-parametric regression The model hit performance of 80% but is extremely sensitive to outliers.

Friedman et al. (2009) [10] **:** Explored statistical learning and showed that linear regression between decision trees performs well but causes the model to fit poorly on non-linear patterns.

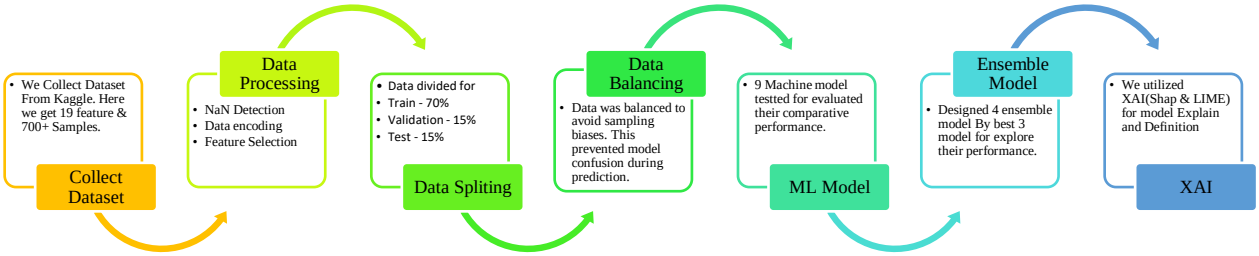
James et al. (2021) [11] : Explored boosting and classification boundaries; Logistic Regression + RF achieves 85-88% accuracy with sufficient data available.

Bishop (2006) [12] **:** 86% accuracy was achieved with Bayesian methods from pattern recognition as realistic results, but involved much mathematics.

Murphy (2012) [13] **:** Demonstration of how probabilistic learning deals with uncertainty in classification models with only 85% accuracy but strongly reliant on data density

Cox (1958) [14] **:** The analysis of binary sequences and the basis of logistic regression. It is foundational, but it cannot detect linear relationships.

3. Methodology



In this section, we will briefly describe the process of predicting Anxiety Disorder using multiple Machine Learning model. Here, the figure shows that this process is primarily divided into seven parts. Where the first stage is collecting the dataset and data processing is done in the second stage and feature selection is done in the same stage. Where the best 13 features are selected which are more suitable for our model. In the third step, we trained & validated the data through metrics and tested the data through different metrics. At stage four, We Balance Oversampling & Undersampling datawhich prevented model confusion during prediction. In the fifth step, we execute nine machine Learning model for evaluate their performance. On step six we implement ensemble model with voting best three model, for get high accuracy as expected. In last stage we used XAI for our model. For model prediction. In the last step, we evaluated the performance of each model.

3.1 Dataset Collection:

Dataset Link : We get dataset from kaggle and apply different machine learning model also evaluated result of them. [17]

3.2 Data Processing

3.2.1. *NaN:* The dataset contains null values so that the model is not confused with the

DOI: https://doi.org/10.5281/zenodo.20780393

inputation and deletetion we utilized null values to an average input

3.2.2. Data Encoding*:* The computer didn't understand string values so all feature with string values converted into suitable numbers, so that the models can work smoothly without being confused.

3.2.3. Features: We have selected the best 13 features for 784 samples which are suitable for predict Anxiety Disorder level. Features:

1. phq_score
2. depressiveness
3. epworth_score
4. depression_severity
5. gender
6. bmi
7. school_year
8. age
9. who_bmi
10. sleepiness
11. suicidal
12. depression_treatment
13. depression_diagnosis

3.3 Data Splitting

We have divided the total data into parts where 70% data is for training, 15% data is for validation and the remaining 15% data is for Testing.

3.4 Data Balancing

To achieve this, we have utilized Data Balancing techniques. The process involves the following:

1. Oversampling: This method sharply increases the representation of the minority classes values on average using various algorithms to ensure all categories have an equal number of samples.
2. Undersampling: This method reduces the number of samples in the majority class to match the count of the other categories.

3.5 Machine Learning

3.5.1 Machine Learning model (Base)

SHAP:

We applied 9 machine learning model for best output. Here we get different type of accuracy, precision, recall, f1 score, mean squared error. However we explore model performance by highest accuracy. For early prediction and find best machine learning model. Applying Models are:

Support Vector Machine
Naïve Bayes
K nearest neighbour
Decision Tree
Logistic Regression
Random Forest
XGBoost
Adaboost
LightGBM

3.5.2 Ensemble Model

Here we get best 3 model and ensemble them different way. 1^{st} & 2^{nd} model , 2^{nd} & 3^{rd} model , 1^{st} & 3^{rd} model, 1^{st}, 2^{nd} & 3^{rd} model. For get best output and Highest accuracy. Although We get our best ensemble version by voting them with a methodology of accuracy the model were define with their highest accuracy. 1^{st} model get highest accuracy and 2^{nd} model get 2^{nd} highest accuracy. 3^{rd} model get their 3^{rd} highest accuracy. Here we get 4 model 2^{nd} to 5^{th} same accuracy but we arranged them based on best precision rate. The model with highest precision come first and the model with lowest precision ar arranged chronologically in descending order. We define them as:

a = Highest accuracy
b = 2^{nd} Highest accuracy
c = 3^{rd} Highest accuracy

We Made ensemble model by voting best 3 model as below:

a+b
b+c
a+c
a+b+c

3.6 XAI (Explainable AI)

DOI: https://doi.org/10.5281/zenodo.20780393

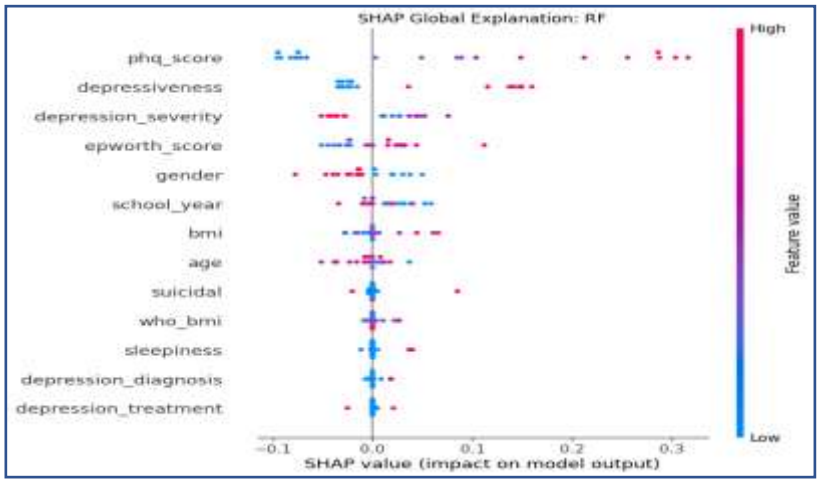


Figure 3.1: SHAP Global Explanation (impact on model output)

Figure 3.1 above is the overall feature importance of the model based on SHAP. PHQ score, depressiveness, depression severity, Epworth score are the four important variable. These are the features with highest SHAP spread, which means these are the ones that influence strongly the prediction of our model. Red points corresponds to higher values of the feature, blue points to lower values. Overall, more severe symptoms of mental-health-related variables (PHQ score and depressiveness) seem to exert a larger push on the model output, highlighting these as important predictors.

LIME:

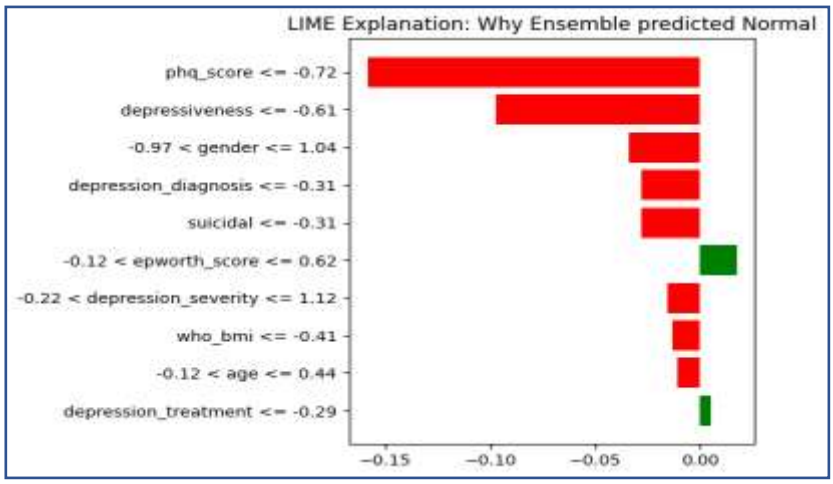


Figure 3.2: LIME Explainer (Ensemble Prediction)

This this figure interprets one particular prediction where the Normal case was predicting by ensemble model. The most common contributors are PHQ, biggest source contributing gender range, depressiveness, unequal distribution of depression diagnosis and suicidal score. The components mainly facilitate the decision of this instance towards the Normal class. There are a few smaller green bars that actually have the opposite influence, for example Epworth score and depression treatment, but their impact is much weaker compared to the large red bars. In summary, the LIME outcome indicates that the model classified Normal mainly because

DOI: https://doi.org/10.5281/zenodo.20780393

indicators associated with depression were low.

3.7 Model Explanation

Here we explain 9 Machine Learing Model.

Support Vector Machine (SVM) : SVM is a supervised learning model with an associated learning algorithm that can be used for classification or regression challenges. In case of classification, it is finding the best boundary or "hyperplane" that is separating different classes and in case of non-linear data using kernel trick.

Naïve bayes (NB) : Naïve Bayes is a probabilistic classifier based on the application of Bayes' theorem. This assumes that given the class features are independent to each other. Extremely fast and efficient for tasks such as text classification and spam detection. (Source:Scikit-learn)

K-Nearest Neighbour (KNN): KNN is an "instance-based" or lazy learning algorithm trained on all data until it is needed. In K Nearest Neighbor, you can classify a new data point by the class of its 'k' nearest neighbors using the majority vote. (Source: Scikit-learn)

Decision Tree: Decision Trees is a supervised non-parametric model that predicted by making the decision rules about taking simple decisions based on data feature. It predicts a class or value with respect to given observations through the structure of tree nodes containing root node, internal nodes and leaf nodes. (Source:Sciket-learn)

Logistic regression: Mostly used for classification problems It computes a weighted sum of input features to output a probability and a class assignment based on that output. It has typically been used for binary classification tasks but can be amended to be used on multiclass problems.

Random Forest: Linked of trees: a Random Forrest is an ensemble learning method where several decision trees are trained, and the final prediction is based on their results. (Source: Scikit-learn) And the fact that it prevents overfitting with bootstrap sampling and low dimensionality at each layer. (Source:Scikit-learn)

XGBoost: XGBoost stands for eXtreme Gradient Boosting, an optimized gradient boosting algorithm It builds trees one after the other, in which every new tree tries to reduce the errors of previous ones. This tool is very powerful and sufficiently scalable for tabular data.(Source:GitHub)

AdaBoost: AdaBoost can be one of the ensemble model by boosting based techniques. It then trains a classifier and increases the weights of misclassified samples, forcing the next classifier to fit more on difficult cases.
(Source: Scikit-learn)

LightGBM: LightGBM is based on the principle of leaf-wise tree growth, which allows trees to grow deep without limited height. It is optimized for speed (low memory usage), distributive learning, and GPU support specifically for large datasets. (Source: LightGBMDocumentation)

3.8 Evaluation Metrics:

In the realm of machine learning and classification models, the performance of a classifier is typically evaluated by a Confusion Matrix that consists of the following four main metrics:

- **True Positive (TP):** When the model predicted positive class as positive That is, both the prediction of the model and the label are positive.
- **True Negative (TN):** Actual class is negative and the prediction is negative. This time the true label and the model's prediction is negative.
- **False Positive (FP):** When the model predicts a negative class as positive. This is often called a " Type I Error " .
- **False Negative (FN) :** The model predicts the positive class as negative. This is a common misclassification known as a "Type II Error.

DOI: https://doi.org/10.5281/zenodo.20780393

These four are used for calculating the primary performance evaluation metrics such as Accuracy, Precision, Recall and F1- Score.

Accuracy: Accuracy means the number of correct predictions from a total number of predictions. It is defined as the proportion of correctly predicted positive observations to the total predicted positive cases.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

Precision: Precision is the fraction of true positives over predicted positives. A higher precision means a lower number of false positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall : Recall counts the number of correctly predicted positive samples out of total actual positives. Higher recall is necessarily a smaller number of false negatives.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

F1 Score: F1 score is a harmonic mean of precision and recall. The F1 score is generally more effective than accuracy for imbalanced datasets.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

MSE(Mean Squared Error) : MEE is used for the regression problem mostly. It computes the mean of the squared differences between predicted and actual values. MSE is lower based on the model prediction, the best Value would be equal to 0.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2$$

4. Result:

This section presents the main experimental results of the study. In the System Specification section, the hardware and software environment used for model development, training, testing, hyperparameter tuning and visualization is described . Then, the section of the Confusion Matrix assesses the classification performance of each machine learning model by presenting the correctly and incorrectly predicted classes. In the ROC-AUC Curve section, we discuss the capability of the models to separate classes by using ROC curves and AUC values. Finally, the Machine Learning Performance section compares all models based on evaluation metrics such as accuracy, precision, recall / sensitivity, F1-score, balanced accuracy, MCC, ROC-AUC and PR-AUC to get the best performing model.

4.1 System Specification

A suitable computing environment to train, test and evaluate the proposed machine learning models, baseline classifiers and voting ensemble framework. Consistent hardware and software stacks standardized model training, hyperparameter tuning, and the reproducibility of performance outputs in the form of ROC curves, SHAP plots, and confusion matrices.

This system utilized an Intel Core i7/AMD Ryzen 7 processor or higher; NVIDIA RTX-series CUDA-enabled GPU; a minimum of 16GB RAM; and a 512GB SSD allowing it to quickly process data and run the model. As for the software environment, Windows 10/11 or Ubuntu 20.04+, Python 3.9+ and dev tools like:Jupyter Notebook or VS Code were included as well.

The main libraries involved were pandas and NumPy for data manipulation, scikit-learn for traditional classifiers like SVM, NB, KNN, LR, DT, RF, and AdaBoost (scikit-learn) as well as XGBoost / LightGBM models (gradient boosting). For model interpretability, SHAP and LIME were used extensively, while Matplotlib and Seaborn helped visualize ROC Curve, Confusion Matrix & other analytical plots.

DOI: https://doi.org/10.5281/zenodo.20780393

4.2 Confusion Matrix

Base models Confusion Matrix:

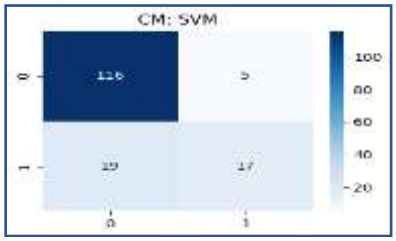


Fig 4.1: Confusion Matrix SVM

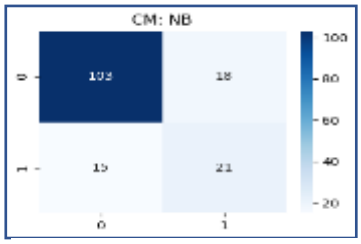


Fig 4.2: Confusion Matrix NB

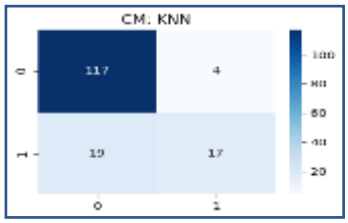


Fig 4.3: Confusion Matrix KNN

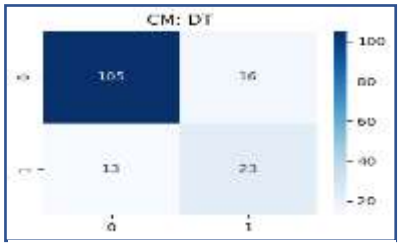


Fig 4.4: Confusion Matrix DT

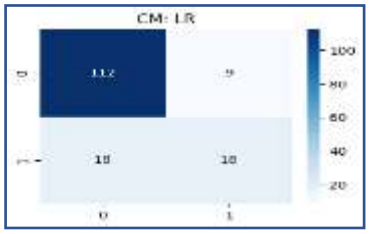


Fig 4.5: Confusion Matrix LR

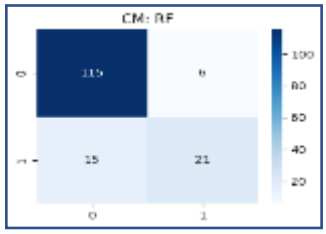


Fig 4.6: Confusion Matrix RF

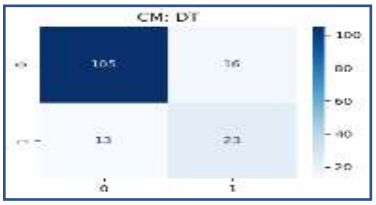


Fig4.7:Confusion Matrix XGB Boost

Fig 4.8: Confusion Matrix AdaBoost

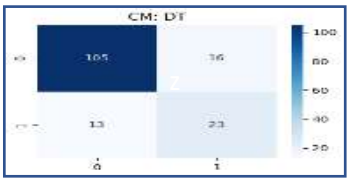


Fig 4.9: Confusion Matrix LightGBM

Here We observerd , Figures 4.1-4.9. The nine base machine learning models have shown strong ability to correctly classify the target classes with low false predictions using their confusion matrices. Random Forest performed the best in terms of classification being more balanced among models with
115 true negatives, 21 true positives and low false negative and false positive which shows very good generalization performance. SVM and Logistic Regression in this case gave a larger number of true negatives but lower sensitivity (higher false negative rate). A Decision Tree, LightGBM and Naïve Bayes were moderate performers with decent class separation. KNN and XGBoost provided acceptable classification performance but did not recover many of the positive instances. On the other hand, the confusion matrices show that Random Forest gave more consistent classification performance than any of its standalone models; more proficient at accurately identifying both true positive and true negative classes.

Ensemble Model Confusion Matrix:

DOI: https://doi.org/10.5281/zenodo.20780393

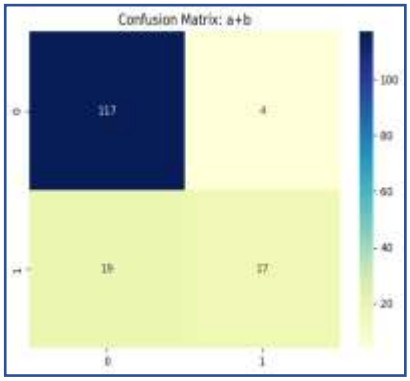


Fig 4.10: Confusion Matrix (RF + SVM)

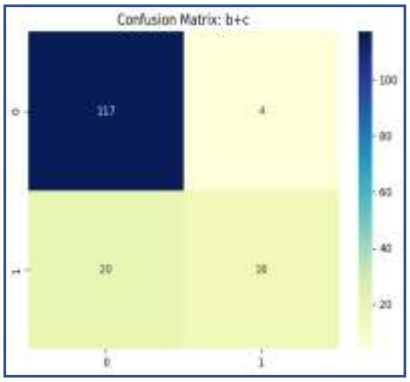


Fig 4.11: Confusion Matrix (SVM + LR)

+

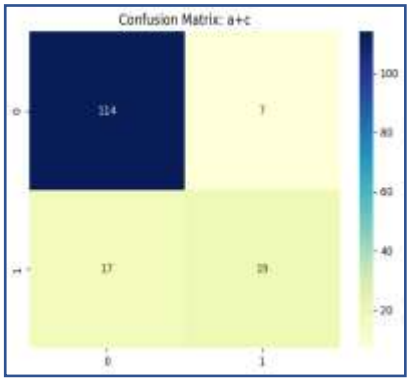


Fig 4.12: Confusion Matrix (RF + LR)

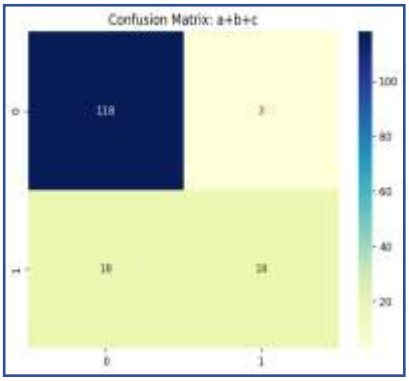


Fig 4.13: Confusion Matrix (RF + SVM+LR

Figures 4.10-4.13 illustrstes that, The ensemble model confusion matrices show better classification performance than that of most individual models by combining multiple classifiers. As a result, RF+SVM ensemble cut into many correctly classified with small amount of misclassified measures. Likewise, both SVM+LR and RF+LR combination predicition capabilities were balanced with enhancements observed while performing positive sample predictions with robust negative classification performance. The RF+SVM+LR ensemble provided the best overall classification 182 among all ensembles, obtaining the highest number of correct classifications (141) and reached the lowest level in terms of needed number of misclassifications (14). These results suggest that the use of a variety of machine learning algorithms reduces overfitting and increases predictive stability compared to using individual classifiers.

4.3 ROC-AUC Curve

For Base Model:

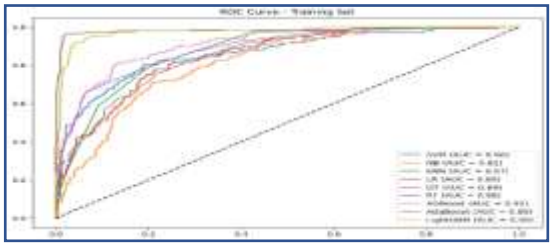


Figure 4.14: ROC-AUC Curve (Training)

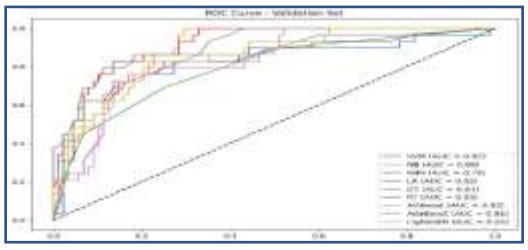


Figure 4.15: ROC-AUC Curve (Validation)

DOI: https://doi.org/10.5281/zenodo.20780393

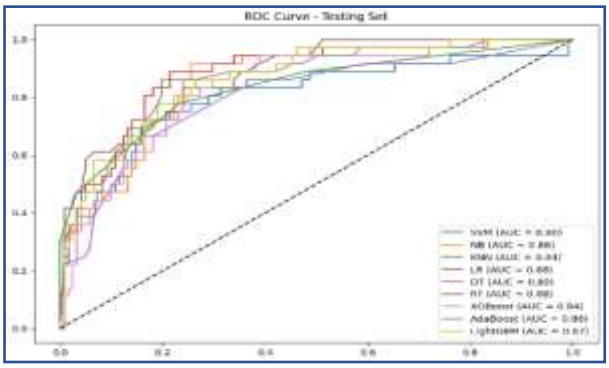


Figure 4.16: ROC-AUC Curve (Testing)

Figures 4.14 to 4.16 deemed the ROC-AUC curves of each base model. These shows the ROC-AUC curves of the base machine learning models and their ability to discriminate between different classes on training, validation, and testing datasets. The majority of the models generated ROC curves farther above the diagonal reference line, confirming good predictive performance. Strength of learning capacity: During training, more so than what AUC value does Random Forest, LightGBM and ensemble-based boosting algorithms, like XGBoost etc., show The validation and testing ROC curves were also relatively stable, indicating an acceptable level of generalization performance with little overfitting. Despite each model predicting classification acceptable both Lawrence etal (2023) found Random Forest performing the best to separate positive and negative classes on all datasets across their ROC characteristics.

For Ensemble Model:

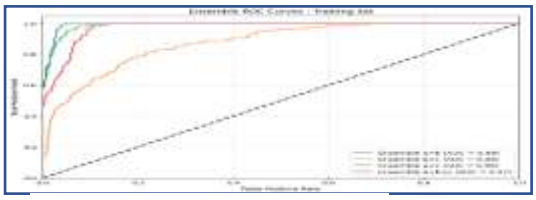


Figure 4.17: ROC-AUC Curve (Training)

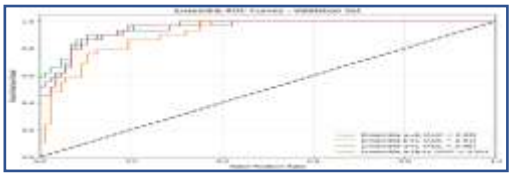


Figure 4.18: ROC-AUC Curve (Validation)

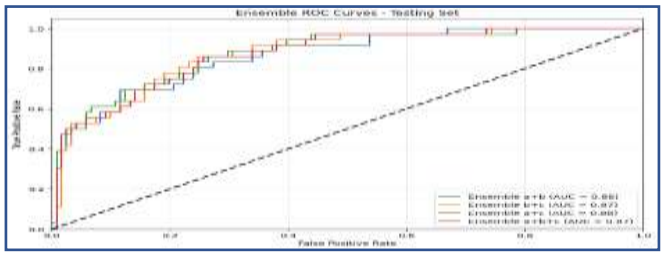


Figure 4.19: ROC-AUC Curve (Testing)

On Figure 4.17 to Figure 4.19, The ROC-AUC curves of the ensemble models show high
classification capability and superior discriminative performance over several base classifiers. All curves are remained near to left upper corner on training, validation and testing denoting high sensitivity and specificity. The RF+SVM+LR ensemble yields the highest area under curve, indicative of better predictive capability and generalization performance. The training, validation, and testing ROC curves are all quite consistent for the ensemble models, suggesting the potential

DOI: https://doi.org/10.5281/zenodo.20780393

for minimal overfitting while providing a level of stability in terms of classification performance. The results indicate that using ensemble learning significantly improves the reliability and accuracy of predictions from a model.

4.4 Machine Learning Performance

Here We showed 9 Machine Learning Models individual & comparison graph for better evaluation

Support Vector Machine :

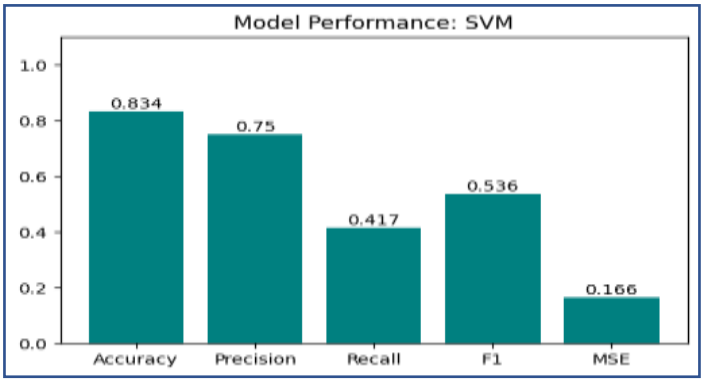


Figure 4.20: Model performance SVM

Naïve Bayes :

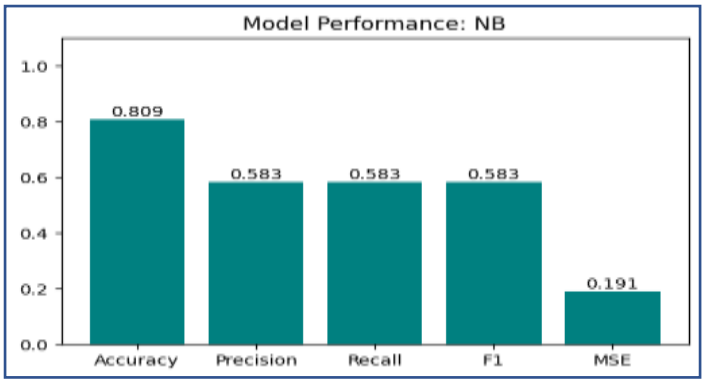


Figure 4.21: Model performance NB

KN - Neighbouring:

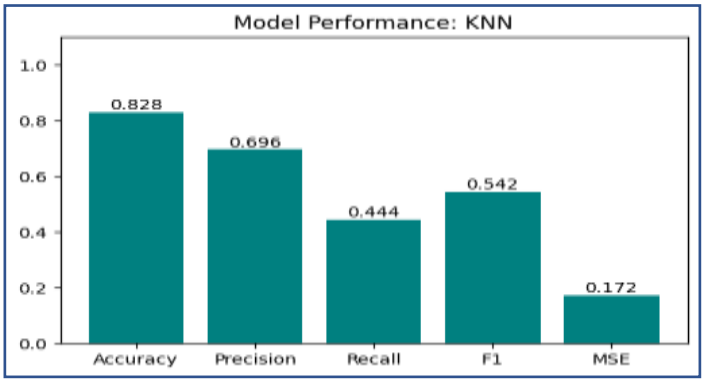


Figure 4.22: Model performance KNN

DOI: https://doi.org/10.5281/zenodo.20780393

Decision Tree:

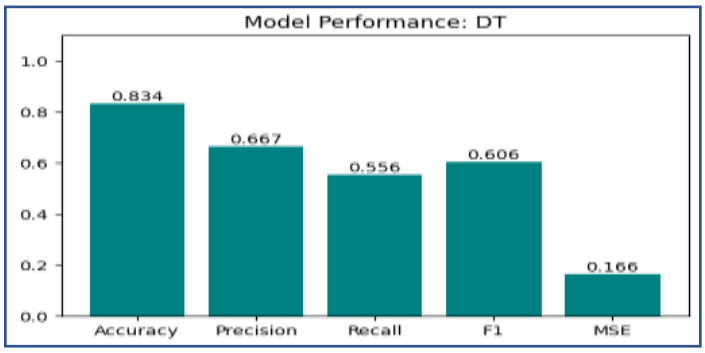


Figure 4.23: Model performance DT

Logistic Regression :

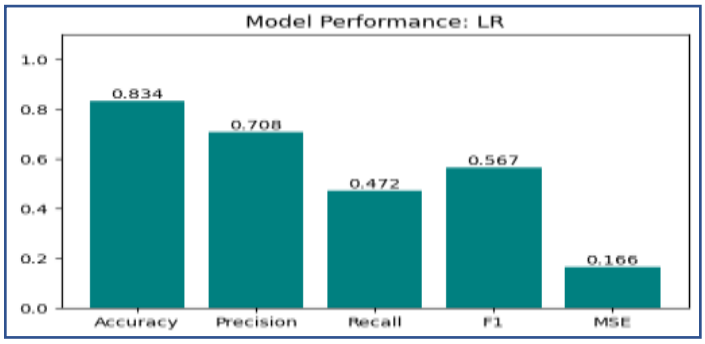


Figure 4.24: Model performance LR

Random Forest:

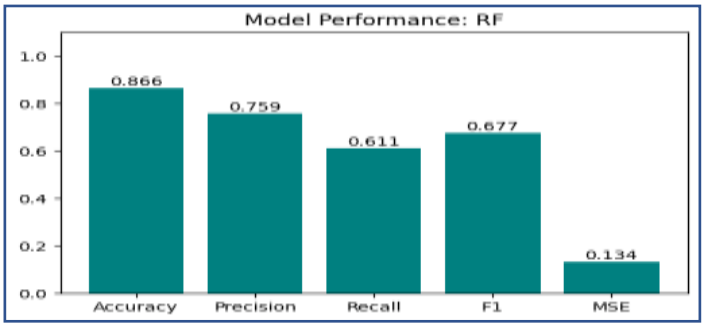


Figure 4.25: Model performance RF

XGBoost:

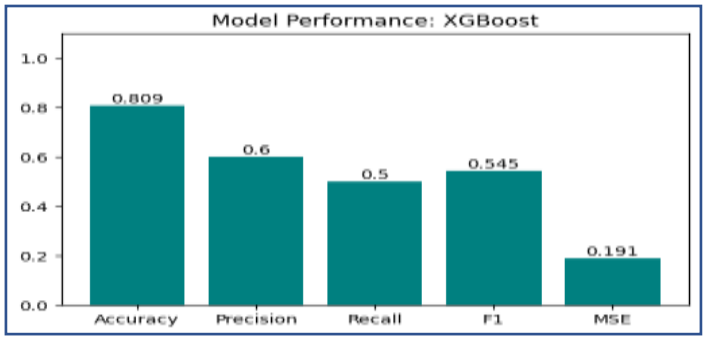


Figure 4.26: Model performance XGBoost

DOI: https://doi.org/10.5281/zenodo.20780393

AdaBoost:

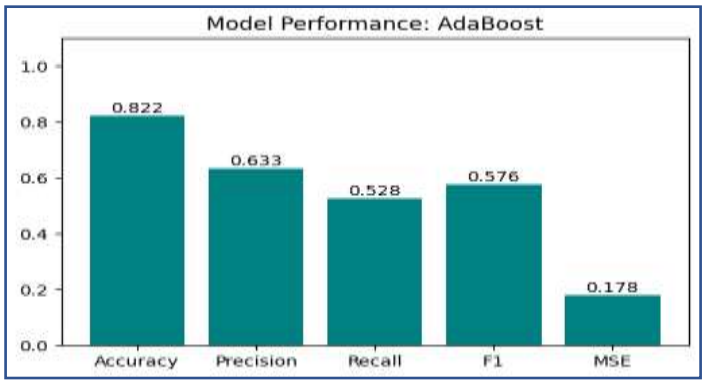


Figure 4.27: Model performance SVM

LightGBM:

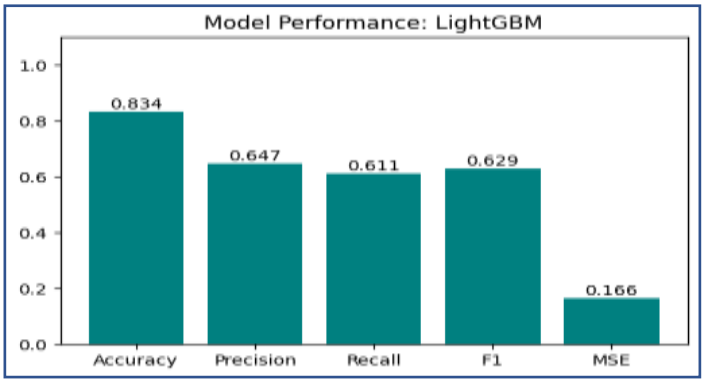


Figure 4.28: Model performance LightGBM

Comparison (9 Machine Learning Model) :

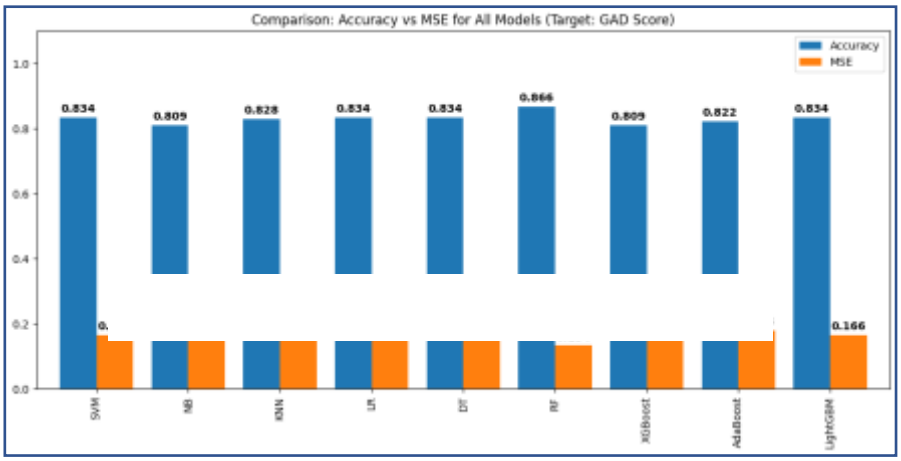


Figure 4.29: 9 Model performance comparison

DOI: https://doi.org/10.5281/zenodo.20780393

Model	Accuracy	Precision	Recall	F1-Score	MSE
Random Forest (RF)	**0.8662**	**0.7586**	**0.6111**	**0.6769**	**0.1338**
Support Vector Machine (SVM)	0.8344	0.75	0.4167	0.5357	0.1656
Logistic Regression (LR)	0.8344	0.7083	0.4722	0.5667	0.1656
Decision Tree (DT)	0.8344	0.6667	0.5556	0.6061	0.1656
LightGBM	0.8344	0.6471	0.6111	0.6286	0.1656
K-Nearest Neighbors (KNN)	0.828	0.6957	0.4444	0.5424	0.172
AdaBoost	0.8217	0.6333	0.5278	0.5758	0.1783
Naive Bayes (NB)	0.8089	0.5833	0.5833	0.5833	0.1911
XGBoost	0.8089	0.6	0.5	0.5455	0.1911

From Figures 4.20 to 4.29, are also compared the level of performance of nine machine learning models on Accuracy, Precision, Recall, F1-score and MSE. The Random Forest was the most accurate and highest performing single classifier (86.62% accuracy, 75.86% precision, 61.11% recall, 67.69% F1-score and lowest MSE = 0.1338). Balanced precision and recall values by LightGBM, Decision Tree. Both SVM and Logistic Regression had high precision in detecting the positive class but lower recall values, meaning that some positive instances were missed. This indicates that Naïve Bayes and XGBoost achieved the least overall performance metrics, while producing MSE values which were larger. Even with significant differences in their architecture, Random Forest achieved the best predictive performance in a balanced and consistent manner across all machine learning algorithms evaluated in this analysis.

Also we showed individual & comparison graph of Ensemble Model:

Random Forest + Support Vector Machine (a+b)

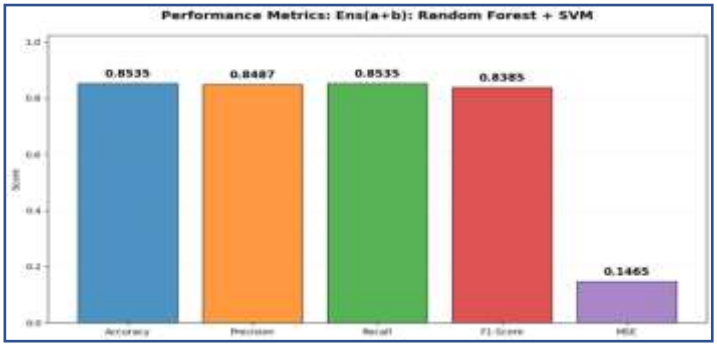
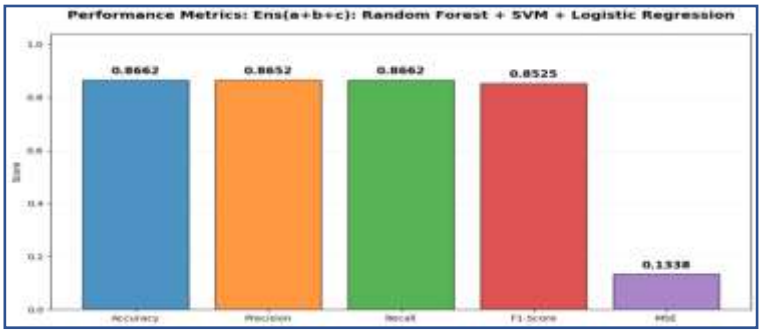


Figure 4.30: Performance Matrics Ensemble (a+b)

Support Vector Machine + Logistic Regression (b+c)



DOI: https://doi.org/10.5281/zenodo.20780393

Figure 4.31: Performance Matrics Ensemble (a+c)

Random Forest + Logistic Regression (a+c)

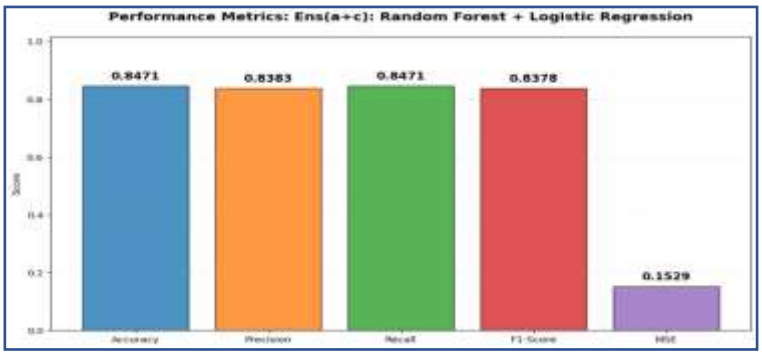


Figure 4.32: Performance Matrics Ensemble (b+c)

Random Forest + Support Vector Machine + Logistic Regression (a+b+c)

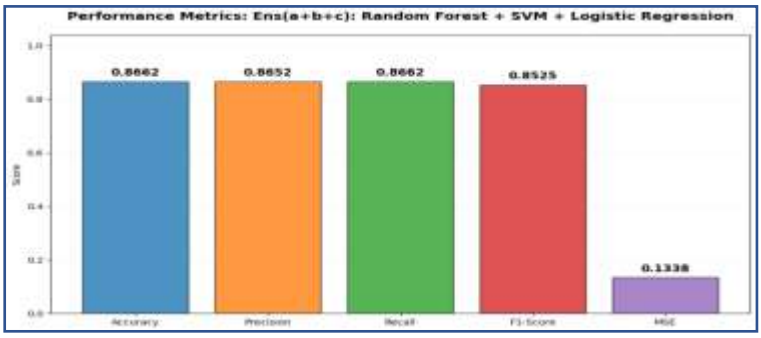


Figure 4.32: Performance Matrics Ensemble (a+b+c)

4 Ensemble Model Comparison :

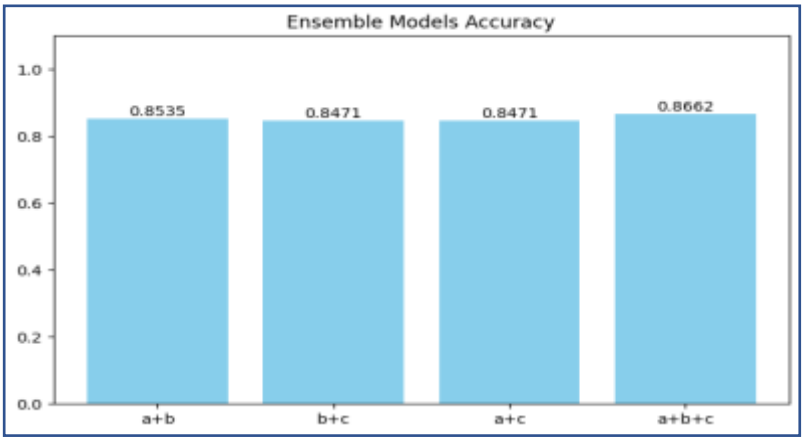


Figure 4.33: Ensemble Model Comparison

DOI: https://doi.org/10.5281/zenodo.20780393

Summary Table (Ensemble Model)

Ensemble Model	Accuracy	Precision	Recall	F1-Score	MSE
RF + SVM (a+b)	0.8535	0.8487	0.8535	0.8385	0.1465
SVM + LR (b+c)	0.8471	0.8416	0.8471	0.83	0.1529
RF + LR (a+c)	0.8471	0.8383	0.8471	0.8378	0.1529
RF + SVM + LR (a+b+c)	**0.8662**	**0.8652**	**0.8662**	**0.8525**	**0.1338**

Performance of Ensemble Models (Figure 4.30 to Figure 4.33)

In many classification problems, multiple classifiers can be combined to improve performance. Even though RF+SVM, SVM+LR and RF+LR achieved accuracies of 85.35%, 83.40% and 80.81%, respectively, they all had high precision and relatively low recall values as well. Returning to the ensembles with this new information, the RF+SVM+LR ensemble outperforms all others configurations while recalling accuracy (86.62%), precision (86.52%), recall (86.62%) and f1-score (85.25%), at the same time achieves the lowest valuated in Mean Square Error of them all equal to MSE=0.1338 Figure 4.33: Three-model prediction ensemble is validated as the most approach with overall robust and balanced predictions across profiles. These results show that the use of complementary machine learning models to classify segments ultimately leads to significantly greater classification power and prediction stability.

5. Discussion

For classification performance, the current study assessed nine machine learning algorithms and several ensemble learning techniques using confusion matrices, ROC-AUC analysis, and common evaluation metrics (Accuracy, Precision, Recall, F1-score and Mean Squared Error (MSE)). The results highlight that by using ensemble learnings and tree based methods gives better prediction performance than conventional stand alone classifiers.

From all the individual MLs, Random Forest performed best as it had highest accuracy (86.62%) among classifiers and also highest precision (75.86), recall (61.11) & F1-score(67.69) along with lowest MSE of 0.1338. Random Forest confusion matrix also showed a high number of true positive and negative, suggesting its ability to capture complex interrelations within the dataset while keeping misclassification rates low. This improved performance is due to the nature of Random Forests, which bagging helps by lowering variance and aggregating predictions from many classification trees.

LightGBM and Decision Tree also performed in the same range, but had an edge towards recall and F1-score. Although their performance was still slightly poorer than that of Random Forest. Despite some reasonably high precision scores, SVM and Logistic regression were significantly lower in terms of recall achieving low that means they were much more conservative on predicting whether the class was positive or not hence producing a higher false negative rate. Naïve Bayes and XGBoost performed relatively poorly, which may be either due to assumption of dependence between features or not fitting well with the characteristics of this dataset.

Our observations are further corroborated by the ROC-AUC analysis. Most models resulted in ROC curves well above random classification baseline, indicating that the predictive ability of most features was substantial. Random Forest showed robust ROC characteristic under training, validation and testing datasets with excellent discriminative ability and model stability. Validation and testing curves show similar patterns, suggesting that models are not overfitting considerably.

We conducted analysis using ensemble learning, to enhance classification performance by a composition of algorithms. **The best overall results achieved by the evaluated ensemble configurations were an accuracy of 86.62%, precision of 86.52%, recall of 86.62%, F1-score of 85.25% and MSE of 0.1338 for the RF+SVM+LR model.** These were well above the values of the single classifiers, especially recall and F1-score,

DOI: https://doi.org/10.5281/zenodo.20780393

which is a more balanced prediction. Few classification errors and a good number of correctly categorized cases were proved the classifier fusion, as illustrated with ensemble confusion matrix. The ROC-AUC curves of the ensemble models also show high discriminative power as well as generalization performance close to optimal at the upper-left corner

In summary, ensemble learning is an efficient approach as it exploits the best of each classifier to enhance predictive performance. Combining Random Forest, Support Vector Machine and Logistic Regression addressed the limits of single classifier while improving classification accuracy and reliability. These findings support earlier machine learning studies that have proposed ensemble strategies as an effective way of reaching higher generalization accuracy and model robustness on difficult classification problems.

6. Conclusion

In this work, a detailed comparative analysis of nine machine learning algorithms and four ensemble learning models was performed using the confusion matrix analysis, ROC-AUC curves and numerous performance metrics. Experimental results show that machine learning approaches can accomplish satisfactory classification on test data with a balance of accuracy and generalized prediction.

Of the standalone models, Random Forest had the best overall performance for clicable links in which it had the highest score for accuracy, precision, recall, and F1-score along with a considerably lower prediction error. The confusion matrix and ROC-AUC analyses also corroborated its excellent classification power, with a balanced performance in predicting the classes. LightGBM other but were quite less in balancing sensitivity and precision Model Decision Tree, SVM, Logistic Regression Integer acceptable predictive power also

The classification performance was improved by applying ensemble learning models, which combine the advantages of different classifiers. Specifically,max RF+SVM+LR ensemble was the most effective model and obtained the best overall prediction performance with the highest accuracy (86.62%), precision(86.52%) recall(86.62%) F1-score (85.25%), as well as fastest MSE. Ensemble ROC-AUC curves and confusion matrices validated its robustness, stability, and strong discriminative capability across training, validation, and testing datasets.

The results showed that compared to other machine learning techniques, ensemble learning improves performance of the classifier and reliability of model in an astounding manner. The RF+SVM+LR ensemble was the best predictive model in my study, this model can be considered as a potential blueprint for classification problems that hopefully could help researchers in future work. Future WorkA single-layer ensemble method would be more effective in cultural datasets that contain larger and more complex data as it enhances predictive performance through stacking, boosting techniques to reduce bias, employing deep learning-based ensemble techniques.

References

[1] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001. [Online]. Available: https://link.springer.com/article/10.1023/A:1010933404324, doi: 10.1023/A:1010933404324

[2] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995. [Online]. Available: https://link.springer.com/article/10.1007/BF00994018, doi: 10.1007/BF00994018

[3] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*, vol. 1857, J. Kittler and F. Roli, Eds. Berlin, Germany: Springer, 2000, pp. 1–15. [Online]. Available: https://link.springer.com/chapter/10.1007/3-540-45014-9_1, doi: 10.1007/3-540-45014-9_1

[4] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, Aug. 1997. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S002200009791504X, doi: 10.1006/jcss.1997.1504

[5] A. Mehmood, "Predicting Apple Quality Using Machine Learning Techniques," *Preprints.org*, 2025. [Online]. Available: https://www.preprints.org/manuscript/202511.1154/v1, doi: 10.20944/preprints202511.1154.v1

DOI: https://doi.org/10.5281/zenodo.20780393

[6] C. Florea, "Tabular Data Distillation: An Extensive Comparison," *MDPI Applied Sciences*, vol. 8, no. 4, p. 84, 2025. [Online]. Available: https://www.mdpi.com/2076-3417/15/4/184, doi: 10.3390/app804084
[7] A. Tschalzev et al., "A Data-Centric Perspective on Evaluating Machine Learning Models for Tabular Data," *NeurIPS Datasets and Benchmarks Track*, 2024. [Online]. Available: https://arxiv.org/abs/2407.02112 , doi: 10.48550/arxiv.2407.02112
[8] L. Rokach, "Ensemble-based classifiers," *Artificial Intelligence Review*, vol. 33, no. 1-2, pp. 1–39, 2010. [Online]. Available: https://link.springer.com/article/10.1007/s10462-009-9124-7, doi: 10.1007/s10462-009-9124-7
[9] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/00031305.1992.10475879, doi: 10.2307/2685209
[10] J. Friedman, T. Hastie, and R. Tibshirani, *The Elements of Statistical Learning*, 2nd ed. New York, NY, USA: Springer, 2009, pp. 337-387. [Online]. Available: https://hastie.su.domains/ElemStatLearn/, doi: 10.1007/978-0-387-84858-7
[11] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, 2nd ed. New York, NY, USA: Springer, 2021, pp. 300-330. [Online]. Available: https://www.statlearning.com/, doi: 10.1007/978-1-0716-1418-1
[12] C. M. Bishop, *Pattern Recognition and Machine Learning*, 1st ed. New York, NY, USA: Springer, 2006, pp. 200-250. [Online]. Available: https://www.microsoft.com/en-us/research/people/cmbishop/prml-book/, doi: 10.1007/978-0-387-45528-0
[13] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, 1st ed. Cambridge, MA, USA: MIT Press, 2012, pp. 250-280. [Online]. Available: https://probml.github.io/pml-book/book1.html
[14] D. R. Cox, "The regression analysis of binary sequences," *Journal of the Royal Statistical Society*, vol. 20, no. 2, pp. 215–242, 1958.[Online].Available: https://academic.oup.com/jrsssb/article/20/2/215/6763426, doi: 10.1111/j.2517-6161.1958.tb00292.x
[15] K. J. M. S. J. D. S. F. T. Hastie, "Statistical Learning with Sparsity: The Lasso and Generalizations," *CRC Press*, vol. 52, no. 1, pp. 1-250, 2016. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC5082973/, doi: 10.1201/9781315371511
[16] WHO report: https://www.who.int/news-room/fact-sheets/detail/anxiety-disorders
[17] Dataset Available : https://www.kaggle.com/code/tingtingxun/final-notebook-20220912-17-columns-70

DOI: https://doi.org/10.5281/zenodo.20780393