

Improving Heart Disease Prediction with Stacked Ensemble Learning: A Comparison of Binary and Multiclass

Amirthaa M
School of Electronics Engineering
Vellore Institute of Technology
Vellore, India
amirthaa.m2021@vitstudent.ac.in

Anitha Julian
Computer Science and Engineering
Saveetha Engineering College
Chennai, Tamilnadu, India
cse.anithajulian@gmail.com

*Gerardine Immaculate Mary
School of Electronics Engineering
Vellore Institute of Technology
Vellore, India
gerardine@vit.ac.in

Anto Lourdu Xavier Raj Selvarathinam
Data Science and Analytics
Grand Valley State University
Michigan, USA
arockiaa@mail.gvsu.edu

Abstract— Heart disease prediction remains a critical area of research due to its significant impact on public health. This paper, titled "Improving Heart Disease Prediction with Stacked Ensemble Learning: A Comparison of Binary and Multiclass," presents a comprehensive study comparing the performance of binary and multiclass classification models for heart disease prediction. Utilizing two distinct datasets, one for binary classification and one for multiclass classification. The study evaluates various machine learning and deep learning models, with a focus on stacked ensemble methods. The Binary Classification Results leveraging the Preprocessing-and-PCA-on-Heart-Disease-Dataset, the study evaluated several models including Random Forest, Support Vector Machine (SVM), XGBoost, Gradient Boosting, Deep Learning model, and Logistic Regression. The results indicate that the Stacking Ensemble model achieved the highest accuracy of 90.2%, outperforming individual models where Random Forest (88.5%), SVM (89.1%), XGBoost (89.7%), Gradient Boosting (88.0%) and Deep Learning (89.1%). The Multiclass Classification study using the UCI Heart Disease Dataset, when applied the same models, the Stacking Ensemble model achieved an accuracy of 64.1%, which, although superior to other individual models such as Random Forest (58.1%), SVM (54.3%), Gradient Boosting (59.2%) and Deep Learning model (57.6%) fell short compared to XGBoost, which attained an accuracy of 65.8%. This outcome highlights the challenges associated with multiclass classification in heart disease prediction.

Keywords— heart disease prediction, machine learning, deep learning, binary classification, multiclass classification, stacked ensemble learning.

I. INTRODUCTION

Heart disease remains a leading cause of mortality worldwide, with cardiovascular diseases accounting for approximately 17.9 million deaths annually, as reported by the World Health Organization. Early and accurate prediction of heart disease is crucial for effective prevention and treatment. Traditional diagnostic methods, which rely on clinical

assessments and basic statistical models, have limitations in handling complex datasets and providing precise predictions.

In recent years, advancements in machine learning (ML) and deep learning (DL) have demonstrated significant potential in improving heart disease prediction. Several studies have employed various ML and DL techniques to enhance predictive accuracy. For instance, deep learning models have achieved notable success in binary classification tasks, where the objective is to differentiate between the presence and absence of heart disease. However, multi-class classification, which involves predicting multiple categories of heart disease severity or classifying patients into several distinct groups, presents additional challenges. The increased complexity of distinguishing between multiple classes can impact model performance and accuracy.

A significant body of research has explored different approaches to address these challenges. For example, recent works have utilized ensemble methods and dimensionality reduction techniques to improve prediction outcomes. Despite these advancements, multiclass classification for heart disease remains less explored and presents unique challenges compared to binary classification.

This paper aims to address these challenges by comparing the performance of binary and multiclass classification models using the Preprocessing-and-PCA-on-Heart-Disease-Dataset and UCI Heart Disease Dataset. The study evaluates several ML and DL models, including Random Forest, Support Vector Machine (SVM), XGBoost, Gradient Boosting, and deep learning techniques, with a focus on stacked ensemble methods. The objective is to assess the effectiveness of these models in predicting heart disease and to provide insights into the performance differences between binary and multiclass classification approaches.

II. LITERATURE SURVEY

The study uses Naive Bayes, KNN, Decision Tree, Genetic Algorithm, Neural Networks, and SVM for predicting heart disease. For improved accuracy, a hybrid model has been used

called the HRFLM model achieved an accuracy of 88.7%. Which combines the features of a linear model and a random forest [1].

The study provides improved heart disease prediction by combining deep learning and machine learning approaches to create a proposed model that outperforms existing models with an accuracy of 94.14%. Through which they can detect early heart disease [3].

This study uses the ensemble method of machine learning to improve the prediction of cardiovascular disease. To maximize the model's performance, they have used the SHAP approach. The final model achieves a better accuracy of 96 percent than the traditional method [4].

The study focuses on developing a machine-learning model to predict heart failure. They have created a stacking model, which provides improved accuracy compared to individual models. The accuracy percentages provided for these models vary, with RF (Random Forests) achieving the highest accuracy of 87.03%, followed closely by LGBM and GBC [5].

An extensive data set of 70,000 patients has been used to predict heart disease using random forest, SVM, and decision tree techniques. Out of which the decision tree provided the highest accuracy of 87.28% [6].

This study enhances heart disease prediction using various machine learning algorithms. After data preprocessing and feature selection, the models were evaluated, with SVM achieving the highest accuracy at 86% [8].

The study included a dataset with 15 attributes, 13 traditional medical attributes plus obesity and smoking. The results indicated that Neural Networks achieved the highest accuracy at 100%, followed by Decision Trees at 99.62%, and Naive Bayes at 90.74% [9].

They employed a 10-fold cross-validation approach to evaluate model performance. The study found that the RF model achieved the highest accuracy of 91.7%, outperforming GBM and SVM models, which had accuracies of 89.8% and 87.2%, respectively [10].

It found that machine learning algorithms, particularly SVM, performed well with an accuracy up to 84.09% without feature selection or outlier detection, while Random Forest achieved 88% accuracy when feature selection was applied. However, deep learning models significantly outperformed machine learning approaches, achieving an accuracy of 94.2% when both feature selection and outlier detection were utilized [11].

This study presents a new approach for detecting cardiovascular disease based on Chi-square feature selection and voting assembly. They have found five important features by using Chi-Square from their dataset, which decreases the computation time and provides an increased accuracy of 92.11% [2].

For the Cleveland dataset, the soft voting ensemble classifier achieved an accuracy of 93.44%, and for the IEEE Dataport dataset, it achieved an accuracy of 95%. The study highlights that the ensemble approach, combined with GridSearchCV and five-fold cross-validation for hyperparameter optimization, effectively enhances model performance over individual algorithms [7].

The models were trained and combined using a weighted sum rule, resulting in three fusion models. The approach was tested on both binary and multi-class datasets, achieving a maximum accuracy of 95% for binary classification and 75% for multi-class classification [12].

III. MATERIALS AND METHODS

A. Binary classification

The dataset utilized for this study is sourced from the GitHub repository titled "Preprocessing-and-PCA-on-Heart-Disease-Dataset". The binary classification dataset used in this study consists of twelve features. The Age feature represents the patient's age in years. Sex indicates the patient's gender, with values of 1 for male and 0 for female. ChestPainType describes the four types of chest pain, namely typical angina, atypical angina, non-anginal pain, and asymptomatic. RestingBP measures the resting blood pressure in mm Hg upon admission. Cholesterol denotes the serum cholesterol level in mg/dl. The FastingBS feature indicates whether the fasting blood sugar level is greater than 120 mg/dl (1 = true, 0 = false). RestingECG shows the resting electrocardiographic results, including categories for normal results, ST-T wave abnormalities, and left ventricular hypertrophy. MaxHR indicates the maximum heart rate achieved during exercise. ExerciseAngina denotes exercise-induced angina (1 = yes, 0 = no). Oldpeak indicates the depression induced by exercise relative to rest. ST_Slope shows the slope of the peak exercise ST segment, categorized as upsloping, flat, or downsloping. The HeartDisease feature is the target variable, indicating the presence (1) or absence (0) of heart disease [9].

Data preprocessing is critical to prepare the dataset for machine learning models. Initially, the dataset was loaded into a Pandas DataFrame for preliminary exploration. The `'info()'` and `'describe()'` functions revealed no missing values, which streamlined the preprocessing process by eliminating the need for imputation. Outliers shown in Fig. 1 were detected using visual methods such as boxplots and statistical methods like Z-scores. Data points with Z-scores exceeding 3 were removed to prevent model distortion.

The next step involved feature engineering, where categorical variables were converted into a numerical format using one-hot encoding. Compatibility with machine learning techniques is assured by this adaptation. The correlation matrix shown in Fig. 2 was analyzed to identify multicollinear features, aiding in feature selection and ensuring that the models could operate effectively without redundant information.

Subsequently, the dataset was split into training (80%) and testing (20%) sets to evaluate model performance. Standardization of features was performed, which scales the data to have a mean of 0 and a standard deviation of 1. Standardization is essential as it enhances model performance by bringing all features to a similar scale, improving convergence and accuracy, (Fig. 3).

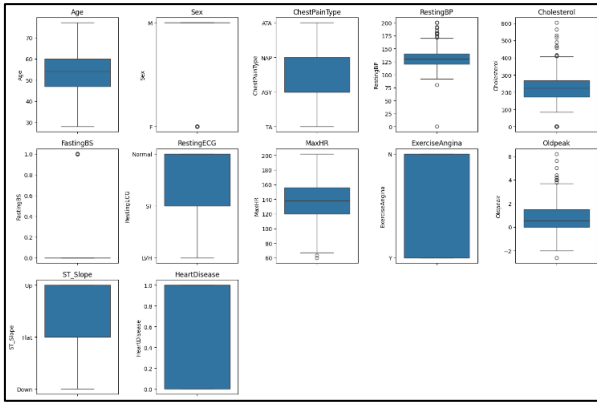


Fig. 1. Outliers of binary classification

B. Multiclass classification

The dataset employed for this multiclass classification task was sourced from the UCI Heart Disease Dataset available via Kaggle. The multiclass classification dataset encompasses sixteen features, each providing critical information about the patient's health status.

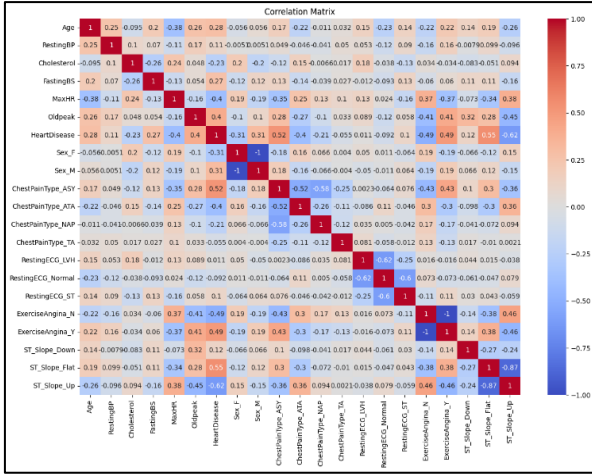


Fig. 2. Correlation matrix of binary classification

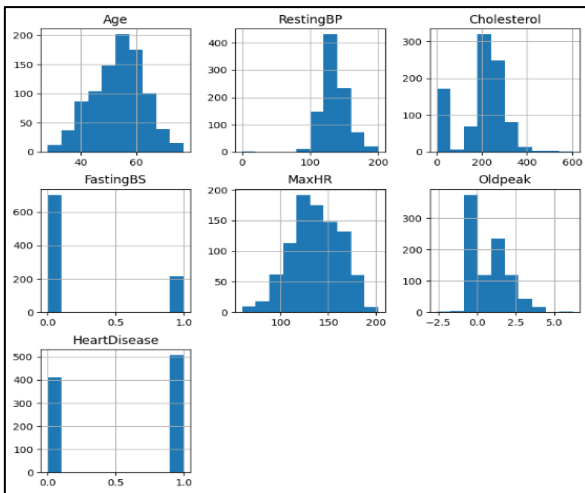


Fig. 3. Histogram of binary classification

ID is a unique identifier for each patient. Age represents the patient's age in years. Sex denotes gender, with 1 for male and

0 for female. Dataset identifies the origin of the data. Cp indicates chest pain type, ranging from 1 for typical angina to 4 for asymptomatic. Trestbps measures resting blood pressure in mm Hg on admission. Chol signifies serum cholesterol level in mg/dl. The Fbs indicates whether fasting blood sugar is above 120 mg/dl (1 = true, 0 = false). Restecg shows resting electrocardiographic results, with 0 for normal, 1 for ST-T wave abnormality, and 2 for left ventricular hypertrophy. Thalch measures the maximum heart rate achieved during exercise. Exang represents exercise-induced angina (1 = yes, 0 = no). Oldpeak indicates depression induced by exercise relative to rest. Slope describes the slope of the peak exercise ST segment, categorized as 1 for upsloping, 2 for flat, and 3 for downsloping. Ca denotes the count of major vessels (ranging from 0 to 3) that are visualized through fluoroscopy.

Thal refers to thalassemia, with values of 3 indicating normal, 6 representing a fixed defect, and 7 indicating a reversible defect. The Num feature categorizes the severity of heart disease into several levels: 0 for no disease, 1 for mild disease, 2 for moderate disease, 3 for severe disease, and 4 for very severe disease.

Data preprocessing was performed like binary classification to ensure that the dataset is in a suitable format for modelling. Specifically, categorical variables with missing values were filled with their mode, while numerical variables were imputed with their mean values.

Data cleaning also involved the removal of the 'id' column, which was deemed unnecessary for the prediction task. Outliers shown in Fig. 4 were analyzed using boxplots, and extreme values beyond a Z-score threshold of 3 were removed to enhance model performance.

Feature engineering was carried out by converting categorical variables into numerical values through one-hot encoding, thereby making them compatible with machine learning algorithms. The correlation matrix shown in Fig. 5 was analyzed to assess the relationships between features and to eliminate any redundant features, ensuring that the models could operate efficiently without multicollinearity issues.

Subsequently, the dataset was divided into training (80%) and testing (20%) sets to evaluate model performance. Features were standardized using 'StandardScaler' to ensure that all variables were on a similar scale, which is crucial for algorithms sensitive to feature scaling, (Fig.6).

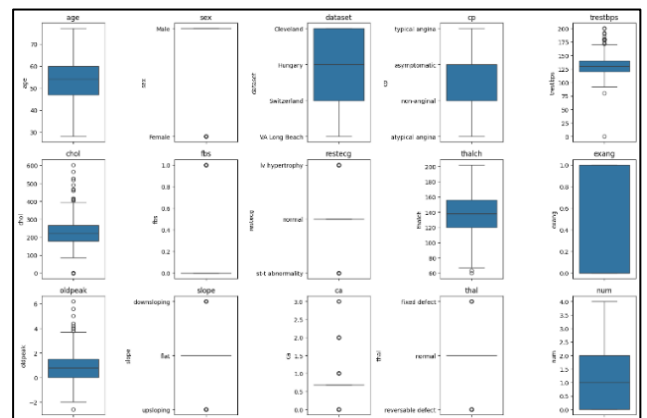


Fig. 4. Outliers of multiclass classification

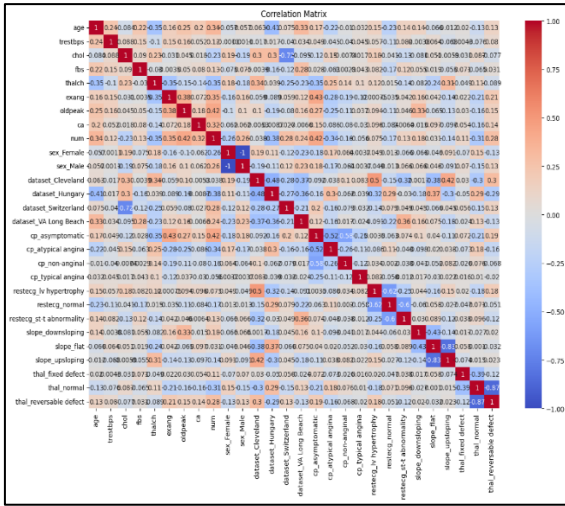


Fig. 5. Correlation matrix of multiclass classification

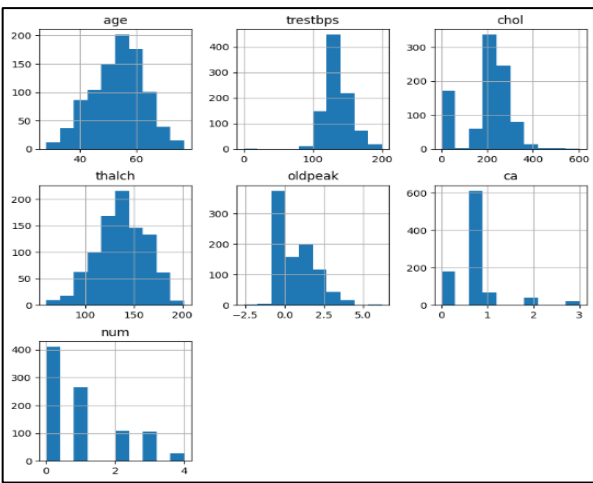


Fig. 6. Histogram of multiclass classification

IV. METHODOLOGY

A. Modal training and evaluation of binary classification

Random Forest Classifier: The Random Forest algorithm, an ensemble method composed of multiple decision trees, was selected for its robustness and efficacy in handling high-dimensional data without overfitting. Each decision tree in the forest independently votes for a class, and the majority vote is used for prediction. The Random Forest model demonstrated an accuracy of 88.6% on the test set, indicating its effectiveness in distinguishing between the presence and absence of heart disease.

Support Vector Machine (SVM): The SVM model was employed to determine the optimal hyperplane that separates classes in high-dimensional spaces. SVM is particularly effective for classification tasks with complex boundaries. The SVM achieved an accuracy of 89.1%, showcasing its capability in handling complex datasets and providing high precision in classification.

XGBoost: Due to its effectiveness and increased performance when processing structured data, the gradient boosting algorithm was included. Predictive accuracy is

improved by building trees one after the other, each one fixing mistakes from earlier trees. The XGBoost model achieved an accuracy of 89.7%, underscoring its strength in boosting performance and handling intricate patterns in data.

Gradient Boosting Classifier: This model was chosen to leverage its ability to build a series of weak learners (decision trees) sequentially, reducing bias and variance. The Gradient Boosting Classifier provided an accuracy of 88.0%, contributing valuable insights into the boosting techniques used in the study.

Deep Learning Model: A neural network with multiple dense layers and a sigmoid activation function was employed to capture complex data patterns. Deep learning models are known for their capability to model intricate relationships and interactions. Despite the computational intensity, the deep learning model achieved an accuracy of 88.0%, demonstrating its effectiveness in handling complex data patterns.

Stacking Ensemble Model: The central approach of this study is the stacking ensemble model, which integrates predictions from multiple base models—Random Forest, SVM, XGBoost, Gradient Boosting, and Deep Learning into a single meta-model. The stacking model uses a Logistic Regression meta-learner to combine base model predictions. This ensemble method achieved the highest accuracy of 90.2%, outperforming individual models. By utilizing each base model's advantages, the stacking model raises total prediction accuracy and performance [10], [11].

B. Results and Comparison of Binary Classification

The stacking ensemble model significantly outperformed the individual base models. With an accuracy of 90.2%, it demonstrated superior performance in classification tasks compared to Random Forest (88.6%), SVM (89.1%), XGBoost (89.7%), Gradient Boosting (88.0%), and Deep Learning (88.0%). The stacking model's enhanced performance is attributed to its ability to integrate the diverse strengths of the base models, resulting in improved precision, recall, F1-score, and AUC across both classes. The stacking ensemble model achieved an impressive accuracy of 90.2% in the binary classification task, outperforming all individual models. This outcome demonstrates the model's high recall and precision in predicting the presence or absence of heart disease.

1) The classification report reveals the following metrics for the stacking model:

Class 0: The precision and recall for Class 0, representing the absence of heart disease, are both 0.88, resulting in an F1-score of 0.88. This indicates that the model was highly effective in correctly identifying instances where heart disease was not present, maintaining a good balance between precision (the proportion of true negatives among all predicted negatives) and recall (the proportion of true negatives among all actual negatives).

Class 1: For Class 1, which represents the presence of heart disease, the model achieved a precision of 0.92, a recall of 0.92, and an F1-score of 0.92. These metrics highlight the model's strong performance in correctly identifying patients with heart disease, showing a higher precision and recall compared to Class 0. This suggests that the model was particularly effective at detecting positive cases of heart disease. The macro average and

weighted average scores, both at 0.90, further indicate balanced performance across both classes, underscoring the model's consistency in prediction accuracy.

2) *The confusion matrix for the stacking model provides additional insights (Fig.7):*

True Positives (TP): The model correctly identified 98 instances of Class 1 (heart disease) and 68 instances of Class 0 (no heart disease). The high number of true positives for Class 1 reflects the model's capability to accurately identify patients with heart disease.

False Negatives (FN): The model misclassified 9 instances of Class 1 as Class 0. While this represents a relatively small number of false negatives, it highlights the occasional misclassification of positive cases.

False Positives (FP): The model incorrectly identified 9 instances of Class 0 as Class 1. This indicates that some patients without heart disease were mistakenly classified as having it.

True Negatives (TN): The model correctly identified 68 instances of Class 0, demonstrating its ability to accurately predict cases where heart disease was absent.

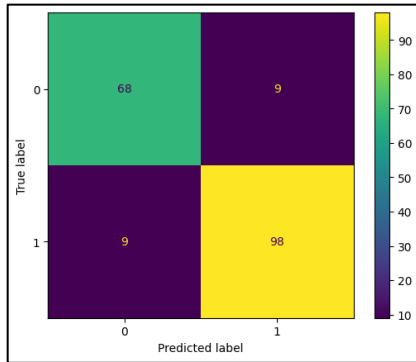


Fig. 7. Confusion matrix of binary classification

3) *Area Under the curve:*

The performance of the heart disease prediction model was assessed using the Area Under the Receiver Operating Characteristic Curve (AUC). The model demonstrated a high level of accuracy in distinguishing between the positive and negative classes, achieving an AUC score of 0.899. This result indicates the robust ability of the model to correctly classify patients with heart disease, reflecting its effectiveness in the prediction task, (Fig.8).

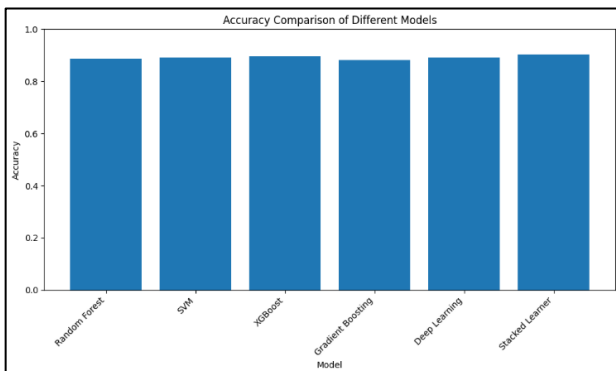


Fig. 8. Comparison of model accuracies of binary classification

C. Modal training and evaluation of multiclass classification

Random Forest Classifier: The Random Forest Classifier was employed due to its robustness and its ability to handle high-dimensional data effectively. The model demonstrated a classification accuracy of 58.2% on the test set, illustrating its competence in predicting the severity of heart disease.

Support Vector Machine (SVM): SVM is used to identify the best hyperplane in the feature space for class separation. This algorithm achieved an accuracy of 54.3%, reflecting its effectiveness in high-dimensional spaces but with slightly lower performance compared to other models.

XGBoost: The XGBoost algorithm was chosen for its boosting capabilities, which sequentially builds trees to correct errors from previous models. It achieved an accuracy of 65.8%, highlighting its superior performance in handling structured data and capturing complex patterns.

Gradient Boosting Classifier: This model, which builds a series of weak learners sequentially, achieved an accuracy of 59.2%. It was effective in reducing bias and variance but did not surpass the accuracy of XGBoost.

Deep Learning Model: A neural network model was implemented with multiple dense layers and a softmax activation function. The deep learning model achieved an accuracy of 55.4%, demonstrating its ability to model intricate data patterns despite its computational complexity.

Stacking Ensemble Model: The core approach of this study was to use a stacking ensemble model that combines predictions from the base models—Random Forest, SVM, XGBoost, Gradient Boosting, and Deep Learning into a single meta-model. The stacking ensemble utilized Logistic Regression as the meta-learner to integrate the diverse base model predictions. This ensemble method achieved the highest accuracy of 64.1%, outperforming the individual models except for XGBoost by 1.7% [12].

D. Results and Comparison of multiclass classification

The stacking ensemble model provided a notable improvement over individual models, achieving an accuracy of 64.1%. In comparison, the Random Forest model achieved an accuracy of 58.2%, SVM achieved 54.3%, XGBoost achieved 65.8%, Gradient Boosting achieved 59.2%, and the Deep Learning model achieved 55.4%. The stacking ensemble model's enhanced performance is attributed to its ability to leverage the strengths of each base model, except for XGBoost which outperforms by 1.7%. The stacking ensemble model outperformed individual models, particularly XGBoost with an accuracy of 65.8%. Despite the stacking model's performance being lower than anticipated, it demonstrates the value of ensemble methods in improving prediction accuracy. The stacking ensemble model's accuracy of 64.1% reflects its effectiveness in predicting the severity of heart disease with improved precision and recall across classes.

1) *The classification report for the stacking model offers a comprehensive evaluation:*

Class 0: Precision of 0.73, recall of 0.92, and an F1-score of 0.82, indicating strong performance in identifying instances of Class 0 (no heart disease).

Class 1: Precision of 0.56, recall of 0.59, and an F1-score of 0.58.

Class 2: Precision of 0.50, recall of 0.38, and an F1-score of 0.36.

Class 3: Precision of 0.59, recall of 0.38, and an F1-score of 0.47.

Class 4: Precision of 0.00, recall of 0.00, and an F1-score of 0.00, demonstrating the challenges in predicting Class 4.

The macro average and weighted average scores were 0.48 and 0.62, respectively, indicating balanced performance across classes but highlighting areas for improvement.

2) The confusion matrix reveals (Fig.9):

True Positives (TP): High accuracy in identifying true positives for Class 0, but lower performance for other classes.

False Negatives (FN): Misclassification of several instances of Class 1, indicating some challenges in predicting positive cases.

False Positives (FP): Incorrect predictions of Class 0 as Class 1.

True Negatives (TN): Accurate identification of instances where heart disease is absent.

3) Area Under the curve:

The performance of the multiclass classification model was evaluated using the Area Under the Receiver Operating Characteristic Curve (AUC). The model achieved a macro-averaged AUC score of 0.827. This score reflects the model's effectiveness in distinguishing between the five classes (0 through 4) in the multiclass classification problem, (Fig.10).

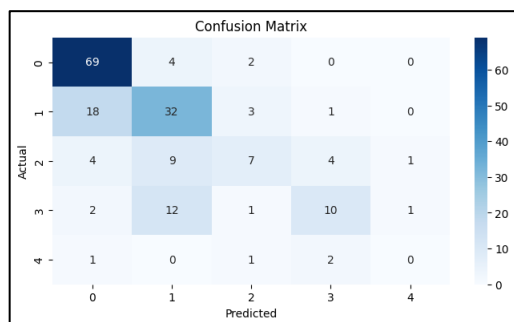


Fig. 9. Comparison of model accuracies of multiclass classification

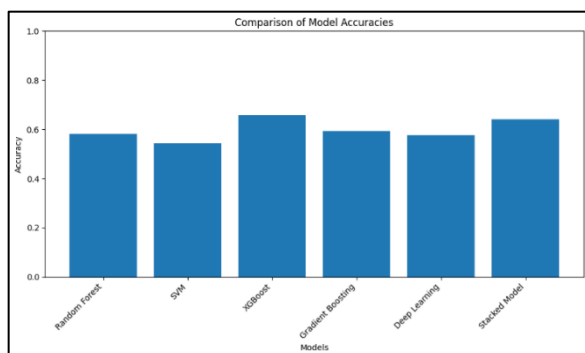


Fig. 10. Comparison of model accuracies of multiclass classification

E. Common procedures in both binary and multiclass classification

The dataset utilized in both binary and multiclass classification was initially loaded into a Pandas DataFrame, and preliminary exploration was conducted. The shape of the dataset was noted, and duplicate entries were removed. Missing values were assessed visually using heatmaps and bar plots generated and were handled through imputation, numerical columns were filled with their mean values, while categorical columns were imputed with the mode.

To address outliers, the dataset was examined using boxplots and Z-scores. Outliers were identified through Z-scores exceeding a threshold of 3 and were subsequently removed to ensure the integrity of the model training process. Feature engineering included the conversion of categorical variables into numerical format through one-hot encoding, and a correlation matrix was analyzed to avoid multicollinearity.

The dataset was split into training and testing sets (80%/20%) using `'train_test_split'`. Feature scaling was performed using `StandardScaler`, which normalizes the data by setting the mean to 0 and the standard deviation to 1. This process was essential to ensure that all features had an equal impact on the model's performance.

1) Model Training and Evaluation:

Base Models: Several base models were trained and evaluated for their performance. These models included a Random Forest Classifier, a Support Vector Machine (SVM), XGBoost, a Gradient Boosting Classifier, and a Deep Learning model. Each model was trained on the training set and evaluated on the testing set using accuracy, precision, recall, F1-score, and confusion matrix metrics.

The Deep Learning model utilized a neural network architecture consisting of dense layers with ReLU activation functions and a final output layer with a sigmoid or softmax activation function. The model was compiled with the Adam optimizer and binary cross-entropy or categorical cross-entropy loss function, depending on the classification task, and trained using a specified number of epochs and batch size.

Stacking Ensemble Model: For an advanced ensemble approach, a stacking model was employed. The predictions from base models (Random Forest, SVM, XGBoost, Gradient Boosting, and Deep Learning) were used as input features for a meta-learner, specifically Logistic Regression. The meta-learner was trained on the predictions from the base models to generate the final classification results.

Performance Metrics: Model performance was evaluated using accuracy scores, and detailed classification reports including precision, recall, F1-score, and AUC. Confusion matrices were generated to analyze the distribution of true positives, false positives, true negatives, and false negatives. A comparison of performance metrics was conducted to assess the effectiveness of the stacking ensemble model against individual base models. The AUC (Area Under the Curve) measures the model's ability to distinguish between different classes and provides a single-value summary of the model's overall discriminative power.

Feature Importance: For models such as Random Forest, feature importance was analyzed to identify and visualize the most significant features affecting the classification outcomes. Feature importance was plotted to provide the relevance of each feature in the predictive model.

F. Differences in Multiclass Classification Compared to Binary Classification

In the binary and multiclass classification approaches, several key differences emerge due to the distinct nature of the classification problems and the models used. Here's a detailed comparison of the code and methodologies applied:

1) Output Layer and Loss Function:

Binary Classification: The deep learning model uses a single output node with a sigmoid activation function, appropriate for binary classification tasks. The loss function employed is binary cross-entropy, which is suitable for scenarios with two class labels.

Multiclass Classification: In the multiclass classification model, the neural network's output layer includes several nodes, corresponding to the number of classes, and utilizes a SoftMax activation function for probability distribution across these classes. This allows the model to predict probabilities across multiple classes. The loss function used is categorical cross-entropy, aligning with the need to classify into more than two categories.

2) Prediction Output:

Binary Classification: Predictions from the base models and the neural network are binary, with probabilities indicating the likelihood of the positive class. For model evaluation, the binary predictions are used directly.

Multiclass Classification: The base models provide probability estimates for each class, and these probabilities are utilized for the stacking ensemble. The neural network produces a probability distribution across all classes. The predictions for each model are saved in a 2D array, where each element reflects the probability scores corresponding to each class.

3) Meta-Learner:

Binary Classification: The meta-learner is trained using binary predictions from the base models and combines these predictions using a logistic regression model.

Multiclass Classification: The meta-learner in the multiclass scenario, also a logistic regression model, is adapted to handle multiple classes using a one-vs-rest (OvR) strategy. This requires reshaping the predictions into a format suitable for multiclass classification.

4) Evaluation Metrics:

Binary Classification: Evaluation includes accuracy, classification report (precision, recall, F1-score), confusion matrix, and AUC. These metrics provide insights into the model's performance in distinguishing between two classes.

Multiclass Classification: Evaluation involves similar metrics, but the classification report, confusion matrix, and AUC are extended to handle multiple classes. Metrics such as macro average and weighted average are used to provide a comprehensive assessment of the model's performance across all classes.

TABLE I. COMPARISON BETWEEN BINARY AND MULTICLASS CLASSIFICATION

Metric	Binary Classification	Multiclass Classification
Stacked Model Accuracy	0.9022	0.6413
AUC	0.8995	0.8278
Random Forest Accuracy	0.8859	0.5815
SVM Accuracy	0.8913	0.5435
XGBoost Accuracy	0.8967	0.6576
Gradient Boosting Accuracy	0.8804	0.5924
Deep Learning Accuracy	0.8913	0.5435
Precision (Class 0)	0.88	0.73
Recall (Class 0)	0.88	0.92
F1-Score (Class 0)	0.88	0.82
Precision (Class 1)	0.92	0.56
Recall (Class 1)	0.92	0.59
F1-Score (Class 1)	0.92	0.58
Precision (Class 2)	-	0.5
Recall (Class 2)	-	0.28
F1-Score (Class 2)	-	0.36
Precision (Class 3)	-	0.59
Recall (Class 3)	-	0.38
F1-Score (Class 3)	-	0.47
Precision (Class 4)	-	0
Recall (Class 4)	-	0
F1-Score (Class 4)	-	0
Confusion Matrix (Class 0)	[[68, 9], [9, 98]]	[[69, 4, 2, 0, 0]]
Confusion Matrix (Class 1)	-	[[18, 32, 3, 1, 0]]
Confusion Matrix (Class 2)	-	[[4, 9, 7, 4, 1]]
Confusion Matrix (Class 3)	-	[[2, 12, 1, 10, 1]]
Confusion Matrix (Class 4)	-	[[1, 0, 1, 2, 0]]

5) Prediction Handling:

Binary Classification: Predictions are handled as single-dimensional arrays representing class labels or probabilities.

Multiclass Classification: Predictions are initially handled as 2D arrays containing probabilities for each class. These are flattened or reshaped appropriately for the meta-learner and evaluation.

6) Training and Testing Data Preparation:

Binary Classification - Base models and meta-learner training use binary labels. The target is to classify instances into one of two categories.

Multiclass Classification - Training involves multiclass labels, with special attention to how probabilities are processed and utilized. The models must correctly classify instances into one of several categories, and the meta-learner must be able to integrate these predictions effectively.

In summary, the transition from binary to multiclass classification necessitates adjustments in model architecture, output handling, and evaluation metrics. The approach to combining predictions and training the meta-learner must be adapted to handle the complexity of multiple classes, ensuring that the final model accurately reflects the nature of the classification problem.

V. RESULTS AND DISCUSSION

For the *binary classification*, the stacked ensemble model achieved an accuracy of 90.22%. The precision was 0.88 for Class 0 and 0.92 for Class 1, with corresponding recall values of 0.88 and 0.92, resulting in F1 scores of 0.88 and 0.92, respectively. The confusion matrix demonstrated 68 true negatives, 98 true positives, and 18 misclassifications. The area under the curve (AUC) was 0.899. Comparatively, other models produced slightly lower accuracies: Random Forest at 88.59%, SVM at 89.13%, XGBoost at 89.67%, Gradient Boosting at 88.04%, and Deep Learning at 89.13%. (Fig.11 and Fig.12).

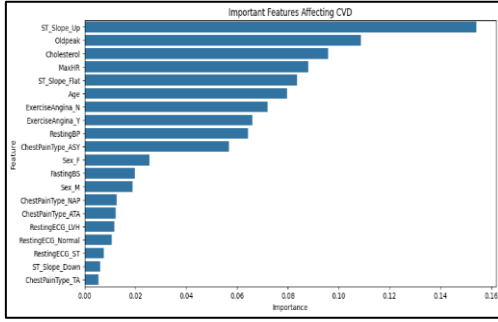


Fig. 11. Important features affecting CVD in binary classification

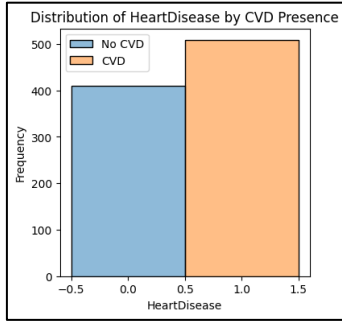


Fig. 12. Distribution of heart disease in binary classification

For the *multiclass classification*, the stacked ensemble model achieved an accuracy of 64.13%. Precision values varied across the five classes, with the highest precision of 0.73 for Class 0 and the lowest at 0.00 for Class 4. Recall values ranged from 0.92 for Class 0 to 0.00 for Class 4, resulting in F1 scores from 0.82 for Class 0 to 0.00 for Class 4. The confusion matrix displayed a spread of correct and incorrect classifications across all classes. The AUC for the multiclass classification was 0.8278. Other models reported lower accuracies: Random Forest at 58.15%, SVM at 54.35%, XGBoost at 65.76%, Gradient Boosting at 59.24%, and Deep Learning at 54.35%. (Fig.13 and Fig.14).

The stacked ensemble model showed a better performance across most metrics, highlighting its effectiveness in both binary and multiclass settings, (Table I).

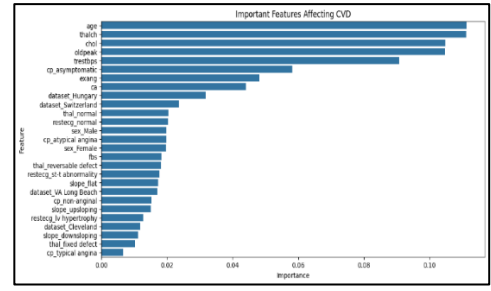


Fig. 13. Important features affecting CVD in multiclass classification

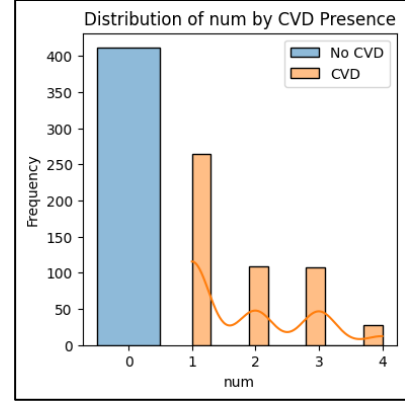


Fig. 14. Distribution of heart disease in multiclass classification

VI. CONCLUSION

This study, explored the effectiveness of stacked ensemble learning for heart disease prediction, comparing it with traditional machine learning models across both binary and multiclass classification tasks. The results indicate that the stacked ensemble model consistently outperformed individual models in terms of accuracy, precision, recall, F1-score, and AUC, especially in the binary classification setting.

The confusion matrix of the multiclass classification revealed that most misclassifications occurred in the middle severity classes, suggesting that the model finds it difficult to distinguish between intermediate stages of heart disease. The findings of this study confirm that stacked ensemble learning is a powerful approach for heart disease prediction, particularly in binary classification scenarios. The model's superior accuracy and balanced performance metrics underscore its potential as a reliable tool for clinical decision support systems. However, in multiclass classification, while the stacked ensemble still outperformed other models, there is room for improvement, particularly in handling imbalanced classes and distinguishing between various severity levels of heart disease.

Future work could involve exploring advanced techniques such as deep ensemble learning, data augmentation, and the use of domain-specific features to further enhance model performance, especially in multiclass settings. Additionally, addressing class imbalance more effectively, possibly through techniques like SMOTE (Synthetic Minority Over-sampling Technique) or cost-sensitive learning, could help improve the model's accuracy in predicting the severity of heart disease. Integrating clinical insights and expert feedback into the model

development process could also lead to more refined and clinically relevant predictions.

REFERENCES

- [1] S. Mohan, C. Thirumalai, and G. Srivastava, "Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques," *IEEE Access*, vol. 7, pp. 81542–81554, 2019, doi: <https://doi.org/10.1109/access.2019.2923707>.
- [2] A. E. Korial, Ivan Isho Gorial, and A. J. Humaidi, "An Improved Ensemble-Based Cardiovascular Disease Detection System with Chi-Square Feature Selection," *Computers*, vol. 13, no. 6, pp. 126–126, May 2024, doi: <https://doi.org/10.3390/computers13060126>.
- [3] S. Abbas, Gabriel Avelino Sampedro, Shtwai Alsubai, A. Almadhor, and T. Kim, "An Efficient Stacked Ensemble Model for Heart Disease Detection and Classification," *Computers, materials & continua/Computers, materials & continua (Print)*, vol. 77, no. 1, pp. 665–680, Jan. 2023, doi: <https://doi.org/10.32604/cmc.2023.041031>.
- [4] S. Subramani *et al.*, "Cardiovascular diseases prediction by machine learning incorporation with deep learning," *Frontiers in Medicine*, vol. 10, Apr. 2023, doi: [10.3389/fmed.2023.1150933](https://doi.org/10.3389/fmed.2023.1150933).
- [5] S. Ahmed *et al.*, "Prediction of Cardiovascular Disease on Self-Augmented Datasets of Heart Patients Using Multiple Machine Learning Models," *Journal of Sensors*, vol. 2022, p. e3730303, Dec. 2022, doi: <https://doi.org/10.1155/2022/3730303>.
- [6] C. M. Bhatt, P. Patel, T. Ghetia, and P. L. Mazzeo, "Effective Heart Disease Prediction Using Machine Learning Techniques," *Algorithms*, vol. 16, no. 2, p. 88, Feb. 2023, doi: <https://doi.org/10.3390/a16020088>.
- [7] N. Chandrasekhar and Samineni Peddakrishna, "Enhancing Heart Disease Prediction Accuracy through Machine Learning Techniques and Optimization," vol. 11, no. 4, pp. 1210–1210, Apr. 2023, doi: <https://doi.org/10.3390/pr11041210>.
- [8] R. Goel, "Heart Disease Prediction Using Various Algorithms of Machine Learning," *papers.ssrn.com*, Jul. 12, 2021, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3884968.
- [9] C. Dangare, S. Apte, and M. Student, "Improved Study of Heart Disease Prediction System using Data Mining Classification Techniques," *International Journal of Computer Applications*, vol. 47, no. 10, 2012, pp. 975–888.
- [10] S. Patel, J. Patel and S. Tejalupadhyay, "Cardiovascular Disease prediction using Machine learning and Data Mining Technique", *Journal - IJCSC*, vol. 7, 2022. Available: [10.090592/IJCSC.2016.018](https://doi.org/10.090592/IJCSC.2016.018)
- [11] R. Bharti, A. Khamparia, M. Shabaz, G. Dhiman, S. Pande, and P. Singh, "Prediction of Heart Disease Using a Combination of Machine Learning and Deep Learning," *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–11, Jul. 2021, doi: <https://doi.org/10.1155/2021/8387680>.
- [12] H. B. Kibria and A. Matin, "The severity prediction of the binary and multi-class cardiovascular disease – A machine learning-based fusion approach," *Computational Biology and Chemistry*, vol. 98, p. 107672, Jun. 2022, doi: [10.1016/j.compbiolchem.2022.107672](https://doi.org/10.1016/j.compbiolchem.2022.107672)