CAPSTONE PROJECT PHASE- 1 REVIEW-1

Optimization of Heart Disease Prediction using Machine Learning Model

Team Members:

- 21BCE10225 ARCHITA GUPTA
- 21BCE10406 SONALI RAGHUWANSHI
- 21BCE10439 PRIYANSHI YADAV
- 21BCE10669 TINA CHELWANI
- 21BCE10708 ABHINAV SHRIVASTAVA

Supervisor
Dr.J. Manikandan

Reviewer 1
Dr.Sasmita Padhy

Reviewer 2
Dr. Antima Jain

OBJECTIVE

Heart Disease:

- Affects heart structure and function, involving blood vessels, rhythm, or muscle issues.
- Leading cause of death globally

Common Types:

- Coronary Artery Disease (CAD): Narrowing/blockage of blood vessels.
- Arrhythmias: Irregular heartbeats.
- Heart Failure: Ineffective blood pumping.
- Congenital Heart Defects: Structural abnormalities from birth.
- Cardiomyopathy: Heart muscle diseases.
- Heart Valve Diseases: Valve dysfunction.

OBJECTIVE

- Machine Learning for Prediction:
  - Processes medical datasets to detect hidden patterns and risk factors.
  - Enables early, accurate heart disease prediction.
- Advantages of Machine Learning:
  - Non-invasive, scalable, cost-effective, and precise.
  - Empowers healthcare professionals with data-driven decision-making.
  - Supports timely interventions, improving patient outcomes.

PROBLEM STATEMENT

- Critical Health Challenge: Heart disease requires early and accurate diagnosis for effective treatment.
- Existing Models:
  - Random Forest, Logistic regression and SVM having accuracies 80%, 61%, 64% respectively.
  - Trained on smaller dataset
- Proposed Solution:
  - Implement XGBoost and Ada Boost Algorithm
  - 10,000-patient’s dataset

MOTIVATION OF STUDY

- Challenges in Existing Approaches to Heart Disease Prediction:

- Early Models:
- Naive Bayes achieved 85.25% accuracy.
- Decision Trees reached 81.97% accuracy (Fadnavis et al., 2021).

- Traditional Algorithms:
- Logistic Regression and K-Nearest Neighbors (KNN) showed competitive performance.
- KNN achieved 88.52% accuracy but struggled with feature scaling and capturing complex data relationships (Jindal et al., 2020).

- Advanced Deep Learning Models:

- Hybrid models and Artificial Neural Networks (ANNs) achieved up to 98.6876% accuracy (Krishnan et al., 2021; Rindhe et al., 2021).
- These models required high computational resources and were often specific to particular datasets, limiting real-world scalability.

- The proposed system integrates Adaboost and XGBoost algorithms to create an accurate and efficient heart disease prediction model.

- Adaboost Algorithm

- Description: Adaboost combines multiple weak learners (e.g., decision stumps) to create a strong classifier. It focuses on misclassified samples by assigning them higher weights in subsequent iterations.

- Advantages:
- Effective on moderate-sized datasets.
- Improves model accuracy by iteratively refining weak learners.

- XGBoost Algorithm

- Description: XGBoost is an optimized gradient-boosting algorithm known for its speed and accuracy. It incorporates regularization techniques (L1 and L2) and parallel processing to enhance performance.

- Advantages:

- Handles large and high-dimensional datasets.
- Reduces overfitting with advanced regularization.
- Provides feature importance insights.

- Advanced ML techniques like XGBoost and Multilayer Perceptrons (MLPs) demonstrate high effectiveness, achieving accuracies of 86.87% and 87.28% (Bhatt et al., 2023). While ensemble models like Random Forest perform consistently, they face challenges in scalability and interpretability for large datasets.

- Motivation for Using AdaBoost and XGBoost:

- AdaBoost and XGBoost address the limitations of traditional and ensemble methods by iteratively improving predictions on misclassified samples. XGBoost excels with its gradient boosting framework, regularization, and efficient handling of missing data. Compared to deep learning models, these boosting algorithms offer high accuracy with lower computational requirements, making them ideal for real-world applications.

- Expected Contributions:

- This study highlights AdaBoost and XGBoost as efficient models for heart disease prediction, balancing accuracy, interpretability, and computational efficiency. By incorporating preprocessing techniques like SMOTE for class imbalance, these algorithms demonstrate superior performance over traditional ML and deep learning models, ensuring broader applicability and generalizability.

DIFFERENCE BETWEEN PROPOSED MODEL AND EXISITNG MODEL

Existing Model

1. Comparison with Random Forest, KNN,CNN, RNN, and SVM.
2. Accuracy-> 69%
3. Area under curve: 0.74
4. precision: 0.70
5. F1 Score: 0.68
6. Recall : 0.67
7. Support: 191

Proposed Model

1. Comparison with AdaBoost algorithm and XGboost algorithm
2. Accuracy-> 72.86%
3. Area under curve: 0.77
4. precision: 0.77
5. F1 Score: 0.70
6. Recall : 0.64
7. Support: 1448

Year	Proposed techniques	Tools	Accuracy
2021[1]	logistic regression, Random Forest Classifier and KNN	Jupyter Notebook	87.5%
2019[2]	Support Vector Machine (SVM) Logistic Regression Naïve Bayes Algorithm	Jupyter Notebook, Web Framework	64.4% 61.45% 60%
2021[3]	Support Vector Classifier Neural Network Random Forest Classifier	MS excel, Python	84.0 % 83.5 % 80.0 %
2023[4]	Random forest Decision tree Multilayer perception XGBoost classifier.	Python, Jupyter Notebook	87.05% 86.37% 87.28% 86.87%
2021[5]	Recurrent Neural Network (RNN)	Python 3.7	98.6876%
2018[6]	Recurrent Fuzzy Neural Network (RFNN)	MATLAB	96.63%
2012[7]	Naive Bayes Decision Trees Neural Networks	Jupyter Notebook Python	90.74% 96.66% 99.25%
2021[8]	Naive Bayes Decision Trees	Jupyter Notebook Python	85.25% 81.97%
2024 [9]	Random forest Ada Boost Gradient Boosting Naive Bayes Logistic Regression	Python, Jupyter notebook	98.71% 88% 93% 80% 80%
2024[10]	Bat Algorithm Particle Swarm Optimization Random Forest	Python, Jupyter notebook	96.88 97.53 94.79

COMPARISON OF ALGORITHMS

1.Logistic Regression

Pros:

-
- Simple and interpretable.
- Fast training and prediction.
- Works well with linearly separable data.

Cons:

- Poor performance with non-linear data.
- Limited capability in handling feature interactions.

2.KNN

Pros:

- Simple to implement and understand.
- Performs well with smaller datasets.

Cons:

- Computationally expensive for large datasets.
- Sensitive to irrelevant features and noise.

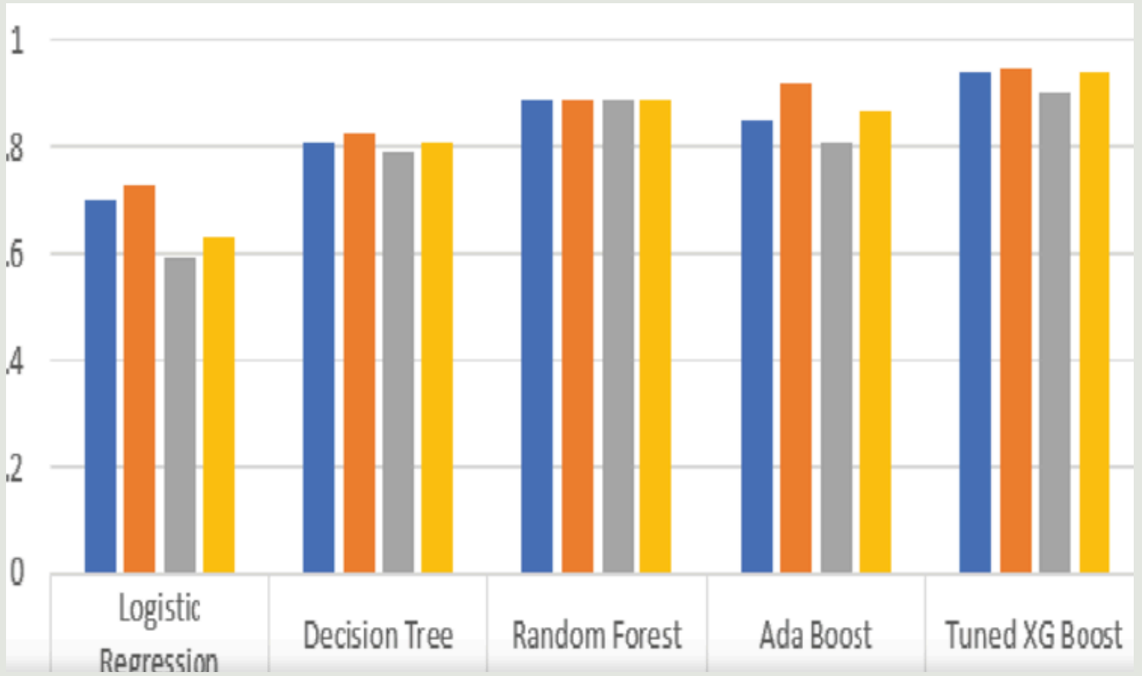
3.Random Forest

Pros:

- Robust to overfitting.
- Provides feature importance insights.
- Handles non-linear relationships effectively.

Cons:

- Computationally intensive for large datasets.
- Less interpretable than simpler models.

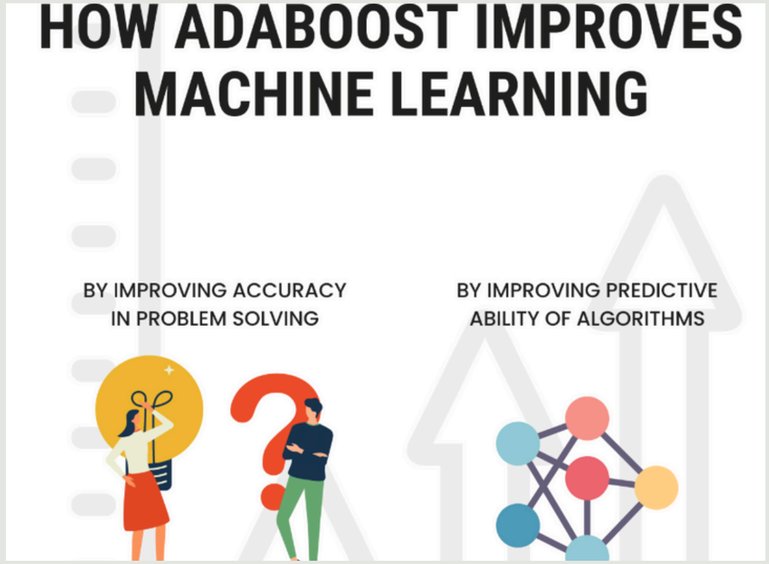


RESEARCH GAP IN EXISTING WORK VS PROPOSED WORK

- Traditional algorithms like Logistic Regression and KNN have limited accuracy with non-linear data, while XGBoost and Adaboost excel.
- Scalability issues plague traditional models with large datasets, but XGBoost handles them efficiently, and Adaboost performs well with more resources.
- Random Forest often overfits, whereas XGBoost reduces overfitting with regularization, and Adaboost can do so with proper tuning.
- Traditional algorithms lack boosting, limiting performance, while XGBoost and Adaboost enhance accuracy and robustness through advanced boosting techniques.

WHY CHOOSE XGBOOST AND ADABOOST?

The existing approaches, while effective to some degree, fall short in handling complex, high dimensional datasets.



Adaboost and XGBoost stand out due to their ability to:

- Manage complex data interactions with high accuracy.
- Reduce overfitting through regularization (XGBoost) and iterative learning (Adaboost).
- Provide feature importance, aiding in better interpretability and insights.
- These algorithms also excel in handling imbalanced datasets and noise, making them highly suitable for medical predictions where data quality and class imbalance are common concerns.

Major Parameters Considered:

1) ap_lo

2) ap_hi

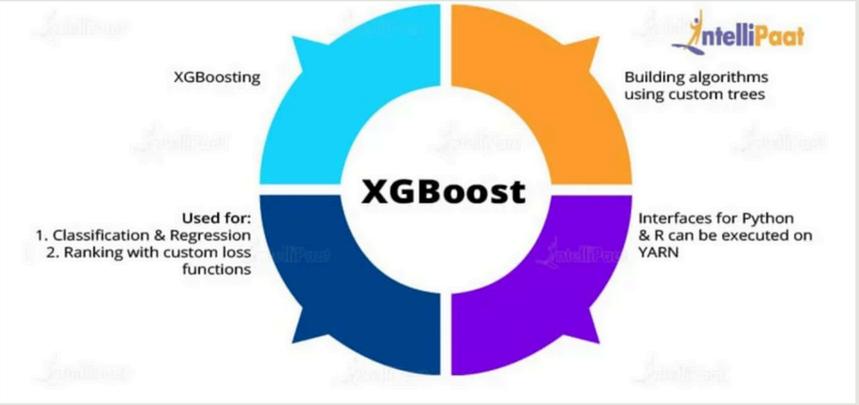
3) cholestrol

4) gender

5) activity levels

6)Alcohol consumption

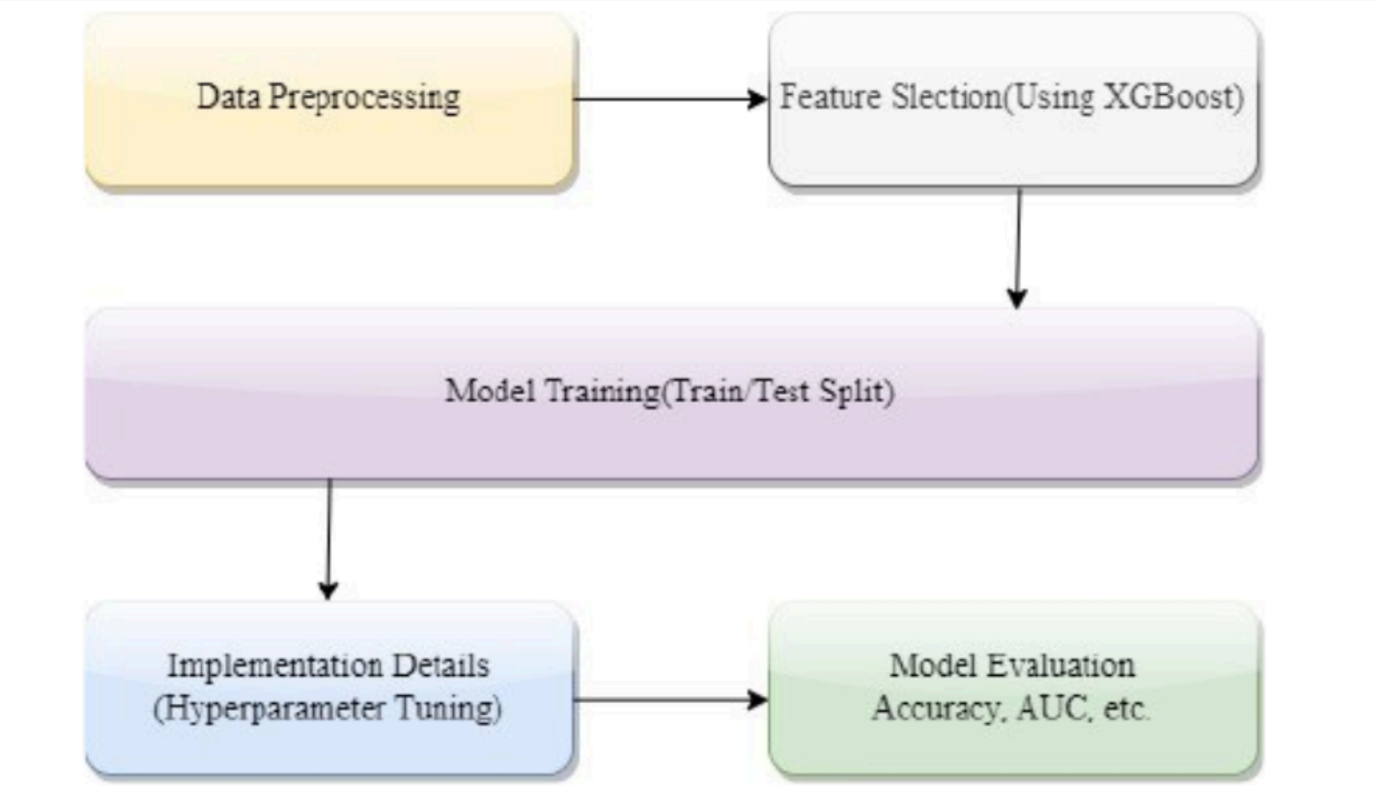
7) Smoke consumption

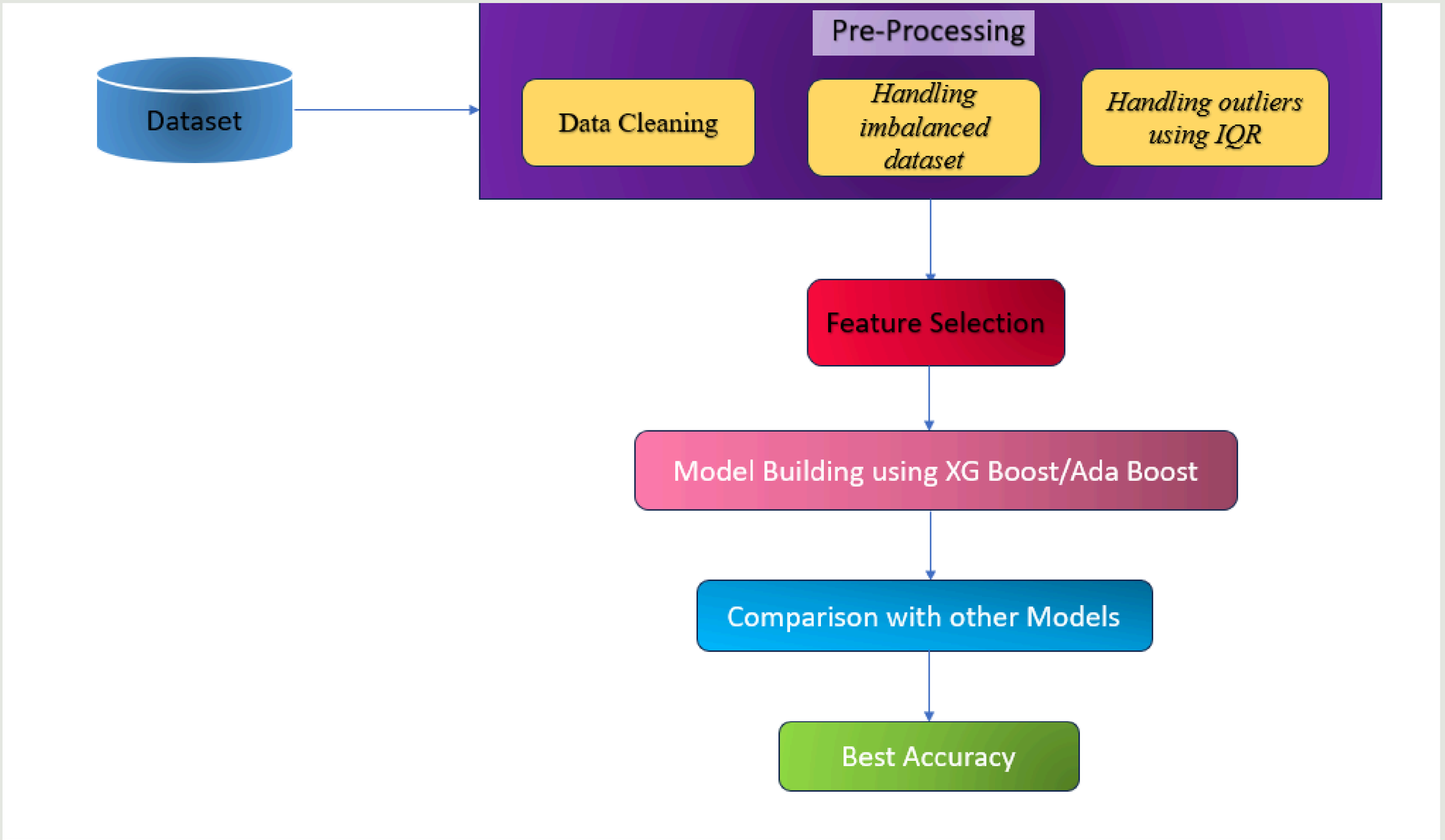


SCOPE OF THE PROJECT

THE PROJECT AIMS TO LEVERAGE MACHINE LEARNING TECHNIQUES TO DEVELOP A PREDICTIVE MODEL THAT CAN ASSESS THE LIKELIHOOD OF HEART DISEASE IN INDIVIDUALS BASED ON CLINICAL DATA. THE SCOPE ENCOMPASSES VARIOUS STAGES, FROM DATA PREPROCESSING, FEATURE SELECTION AND MODEL DEVELOPMENT TO TUNING OF THE USED MODEL AND COMPARASION WITH OTHER MODELS.

ARCHITECHTURE/ WORK FLOW




Dataset
Pre-Processing
Data Cleaning
Handling imbalanced dataset
Handling outliers using IQR
Feature Selection
Model Building using XG Boost/Ada Boost
Comparison with other Models
Best Accuracy

PROPOSED WORK

1.Data Collection

.

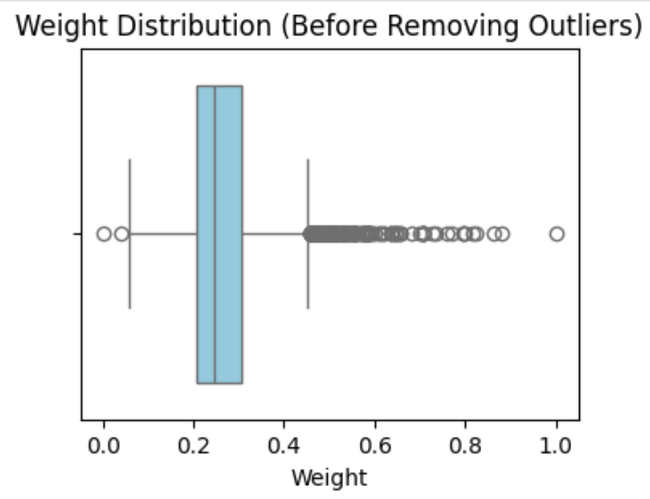
- 10,000 records with 13 parameters.
- 12 Features-> id, age , gender, height, weight, systolic blood pressure, diastolic blood pressure, cholesterol, glucose level, smoking, alcohol consumption, physical activity.
- 1 Feature-> Result, Heart Disease or not
- Numeric data -> id, age, height, weight, systolic blood pressure, diastolic blood pressure
- Binary data -> smoking, alcohol consumption, physical activity
- Category data-> cholesterol, glucose level, gender

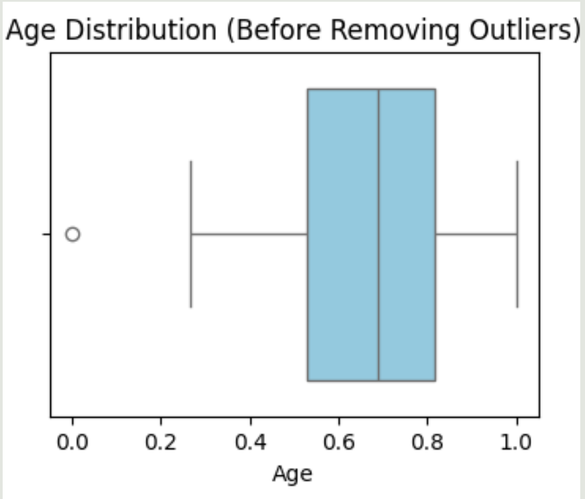
2. Data Preprocessing

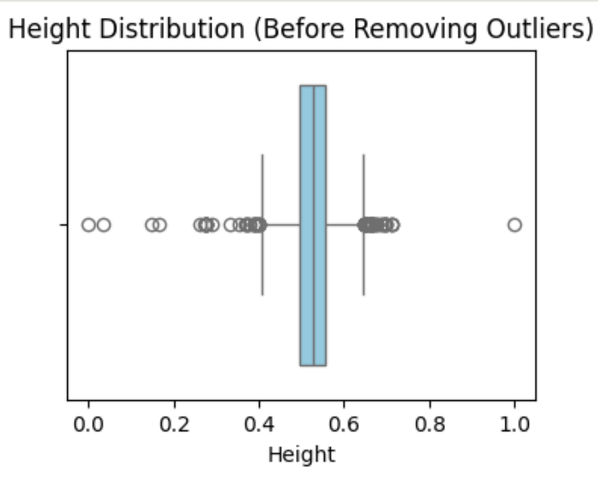
- Handling Imbalanced dataset
  - Already balanced, 5,030 negative and 4,969 positive cases
  - Model learns effectively from both classes
- Data Cleaning
  - Heatmap to Find missing values
  - No missing values found
  - No risk of biased results.

- **Handling Outliers using IQR**
  - Outlier identification and removal
  - IQR method in Age , Weight and Height feature
  - Outliers were defined as data points outside Q3 + 1.5 * IQR or Q1 – 1.5 * IQR.
  - Reduced bias and enhanced model accuracy.

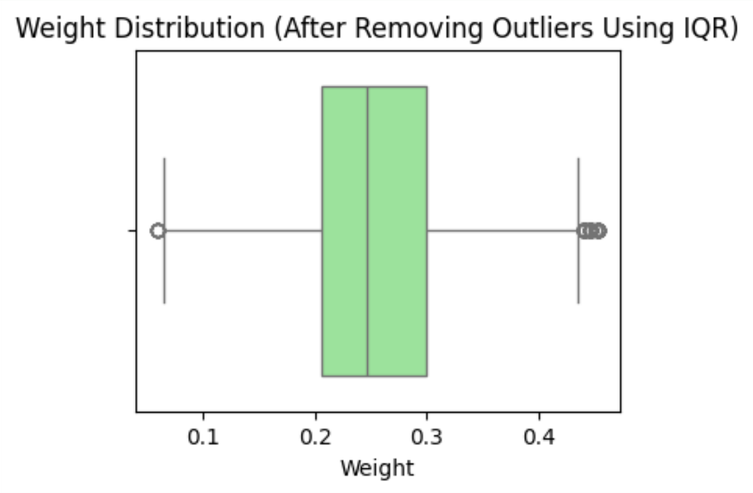
Before Handling Outliers

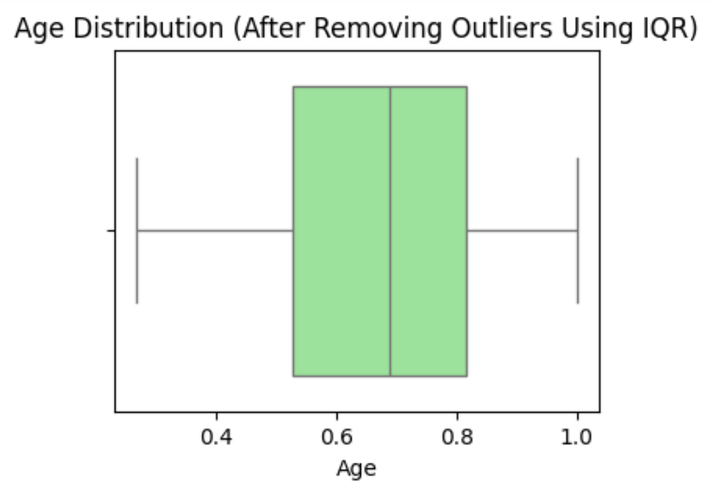


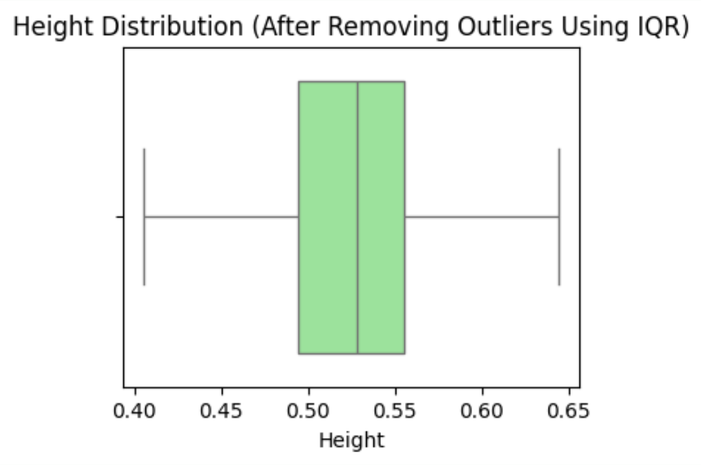




After Handling Outliers

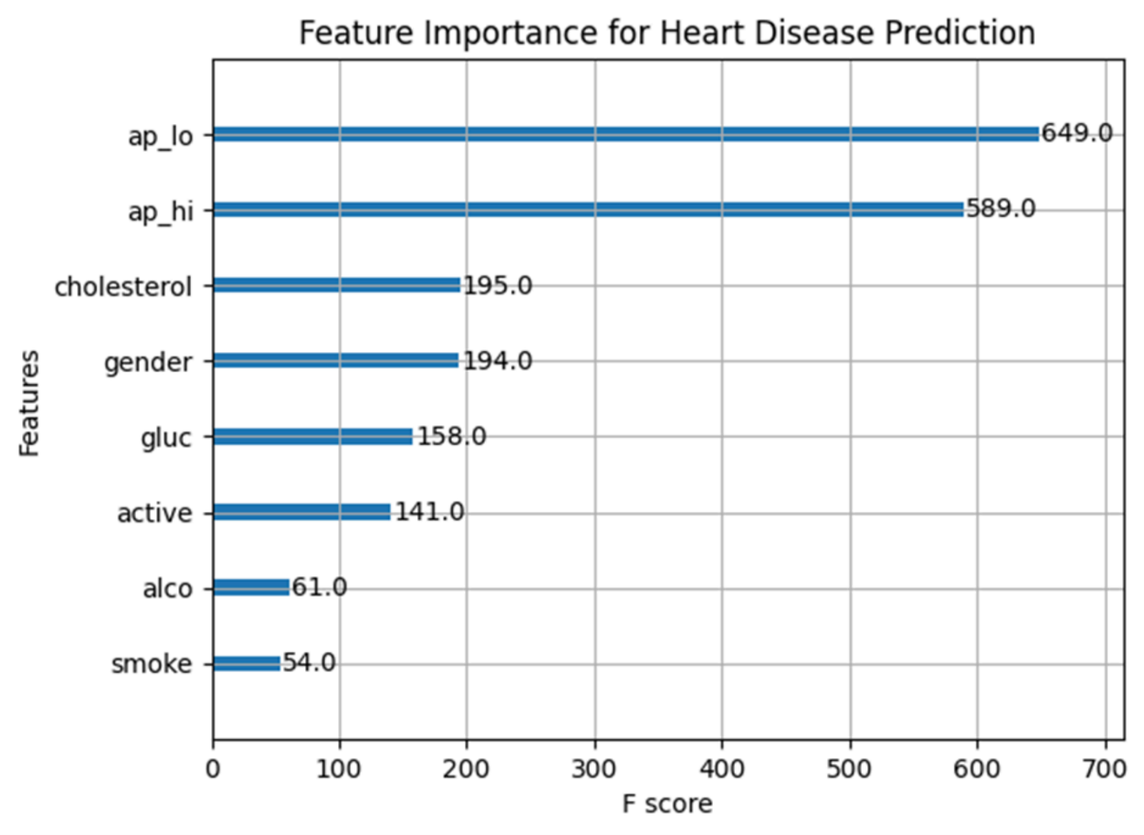




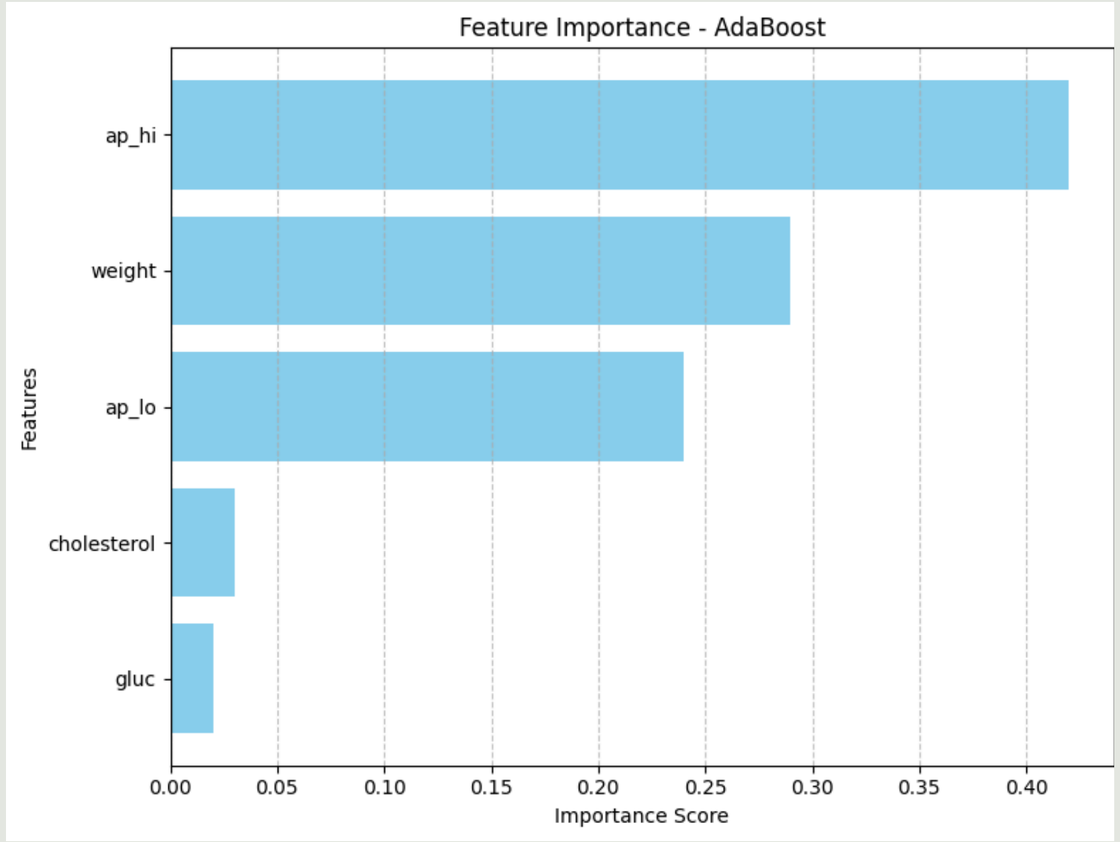


3.Feature Selection

- Selected most relevant features for Both the Algorithms
- Feature Importance Graph
- XG Boost-> systolic blood pressure, diastolic blood pressure, cholesterol, gender, glucose and activity
- ADA Boost-> systolic blood pressure, diastolic blood pressure, cholesterol, glucose and weight



Important Features in XG Boost Algorithm



Important Features in Ada Boost Algorithm

4. Model Building

- We employed AdaBoost and XGBoost, combining multiple weak learners to predict heart disease.
- Both models improve accuracy by iteratively adjusting the weights of misclassified data points, focusing on difficult cases.

4.1 AdaBoost

- AdaBoost combines weak classifiers, focusing on misclassified instances by adjusting their weights iteratively.

$$\hat{y} = \sum_{i=1}^{N} \alpha_i h_i(x)$$

- The formula predicts ŷ as the weighted sum of hi(x), where αi reflects each classifier's accuracy.

- AdaBoost’s final prediction is a weighted sum of the predictions from all weak classifiers, with weights based on each classifier's performance.

4.2 XGBoost

- XGBoost is an optimized version of gradient boosting that builds trees sequentially to minimize errors from previous ones.

- It incorporates regularization techniques to prevent overfitting and efficiently handles missing data.

$$\hat{y} = \sum_{i=1}^{N} \alpha_i h_i(x)$$

- The formula ŷ is the predicted value, N is the number of trees, hi(x) is the prediction of the i-th tree, and αi is the weight assigned to each tree.

- Its advanced optimizations result in faster training and superior predictive performance.

5. Parameter Comparision

1.The dataset for heart disease prediction is divided into input features (blood pressure, cholesterol, weight) and the target variable (heart disease).

2.The data is split into 70% training and 30% testing sets with balanced class distribution using the stratify=y parameter.

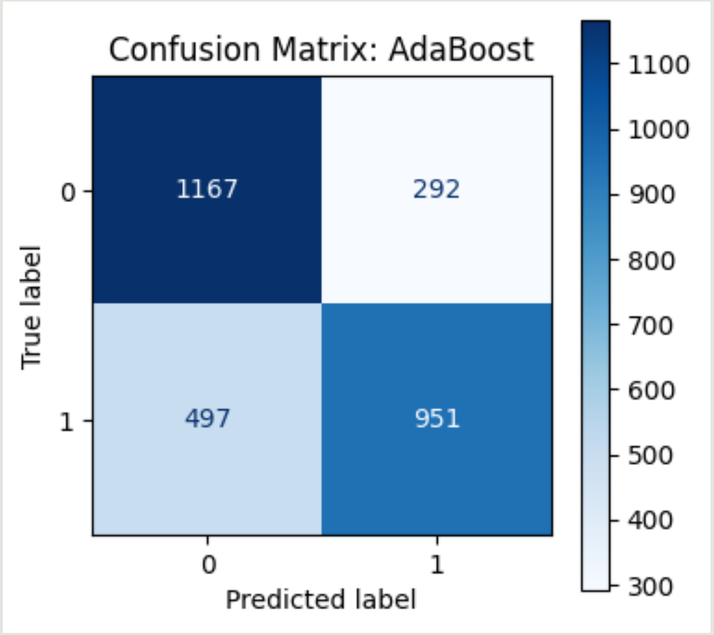
3.This allows AdaBoost and XGBoost models to learn from the training data and evaluate performance on unseen test data.

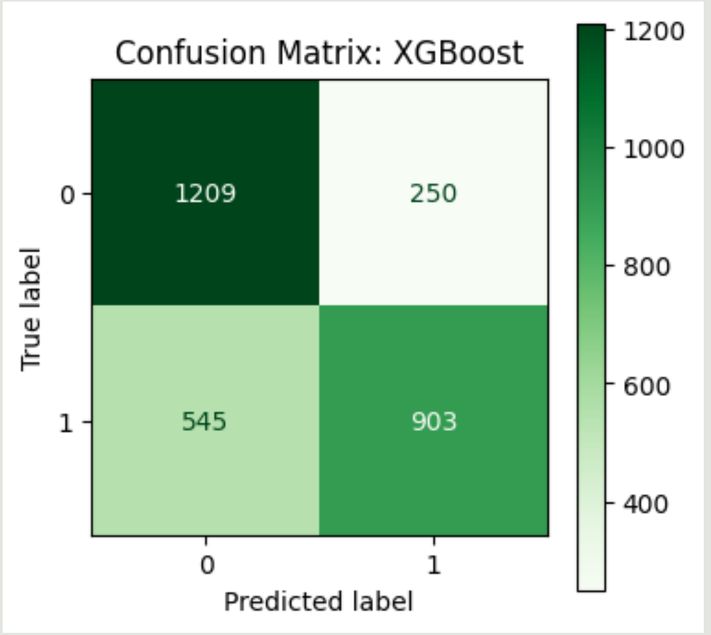
4.AdaBoost and XGBoost classifiers are initialized with 100 estimators, and XGBoost is configured with a learning rate of 0.05 and maximum tree depth of 4.

5.Both models are trained and evaluated on test data using the classification_report function, providing metrics like precision, recall, and F1-score for performance comparison.

- The confusion matrices for AdaBoost and XGBoost models are visualized to compare their performance by showing true positives, false positives, and other classification outcomes.

- The matrices are displayed in blue (AdaBoost) and green (XGBoost), helping to visually assess and compare the models' accuracy in classifying the test data.





6.Hypertuning

1.Purpose: Optimize model performance by finding the best hyperparameter combinations.

2.Methods: Use Grid Search, Randomized Search, or advanced methods like Bayesian Optimization.

3.Tools: Libraries like GridSearchCV (sklearn) or xgb.cv (XGBoost).

4.Key Parameters

- AdaBoost: n_estimators, learning_rate, base_estimator.
- XGBoost: learning_rate, max_depth, n_estimators, subsample, etc.

5.Process: Define search space, apply cross-validation, and select the best-performing hyperparameters for final training.

RESULT AND DISCUSSIONS

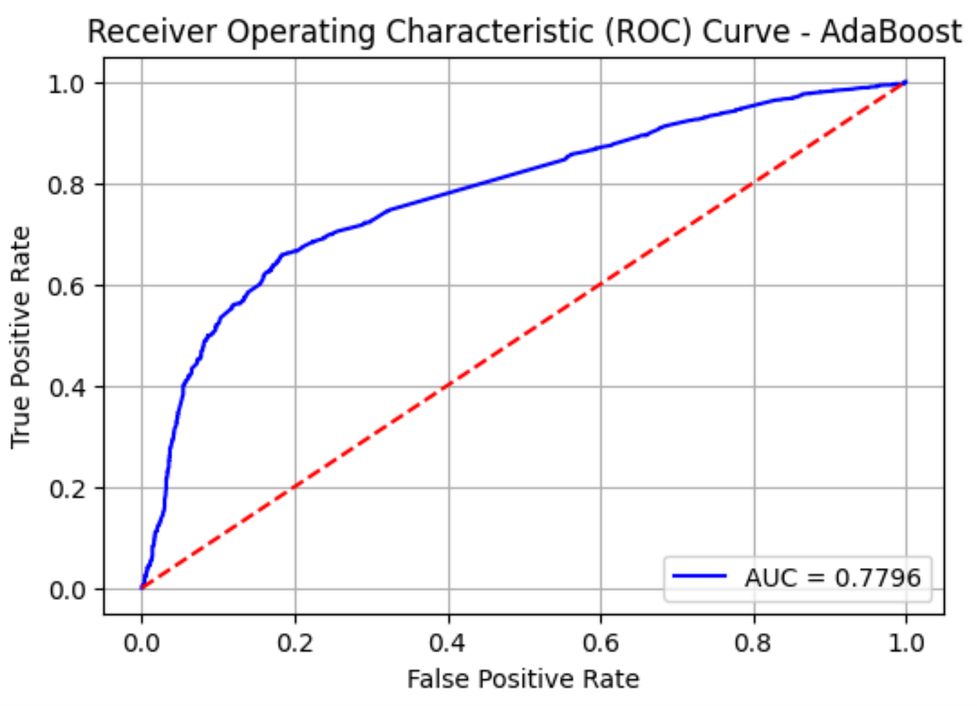
1.Performance Assessments

1.1Feature Selection

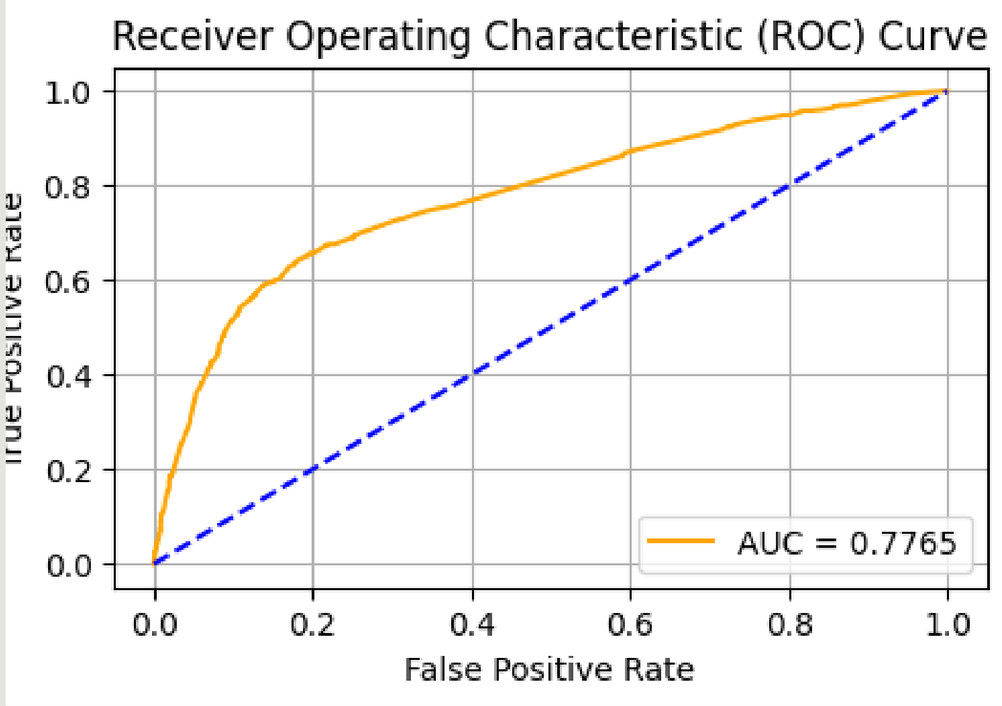
- AdaBoost and XGBoost identify key features like blood pressure, cholesterol, and glucose, with AdaBoost emphasizing body weight and XGBoost highlighting physical activity.

- The combination of both models ensures a robust feature selection, balancing accuracy, interpretability, and generalization.

- The model's strong performance, measured by accuracy and ROC-AUC, demonstrates the effectiveness of the selected features, suggesting potential for real-world applications.



ROC Curve of Ada Boost



ROC Curve of XG Boost

1.2 Model Building using XG Boost and Ada Boost

- AdaBoost enhances weak classifiers by focusing on misclassified samples, iteratively adjusting their weights to improve accuracy, making it effective for complex and imbalanced datasets.

- XGBoost builds decision trees sequentially, correcting errors from previous trees, and incorporates regularization and early stopping to reduce overfitting, making it highly effective for complex datasets with feature interactions.

- Both models enhance predictive accuracy through iterative learning, but XGBoost offers additional strategies like regularization to ensure robustness in high-dimensional datasets.

XGBoost Model Performance Metrics Table

Performance Metrics	
Metrics	**Values**
Accuracy	0.7282
Precision	0.79
Recall	0.62
F1 score	0.69

AdaBoost Model Performance Metrics Table

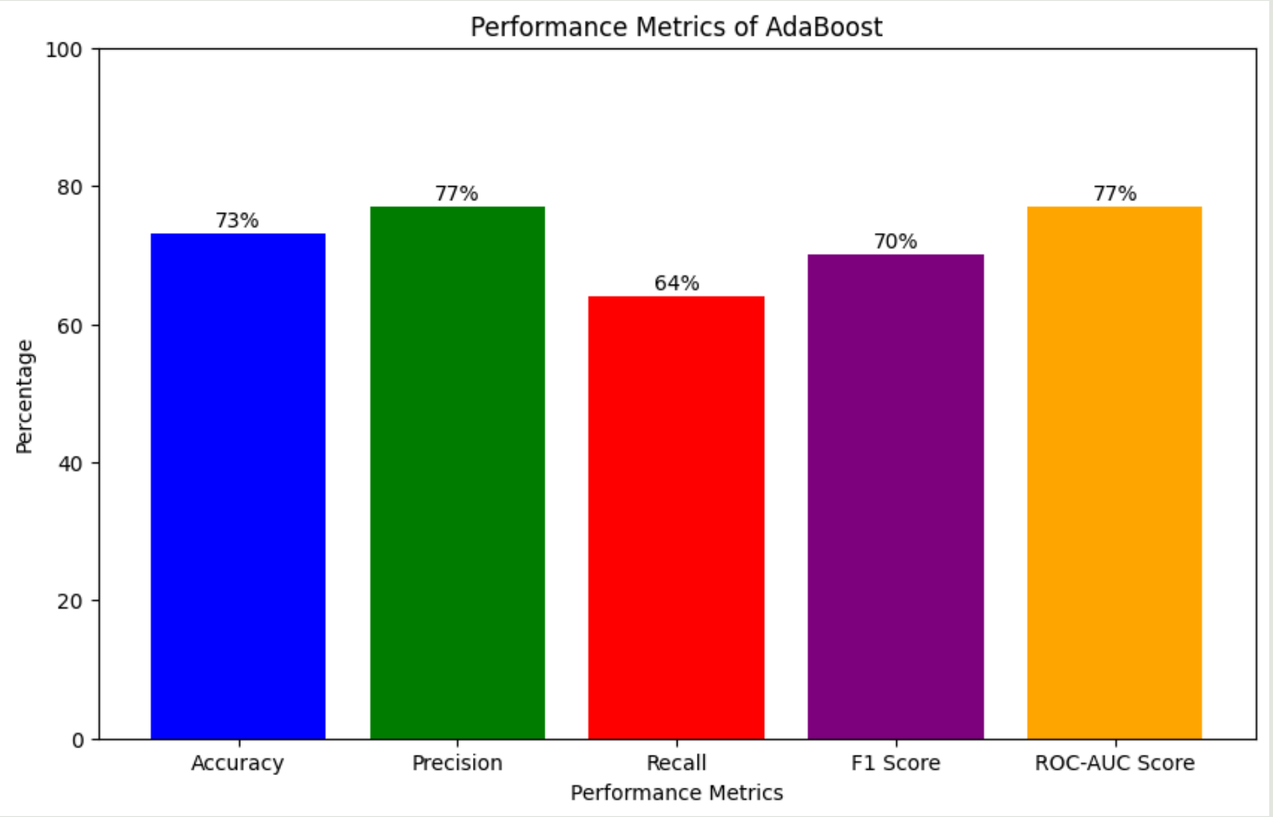
Performance Metrics	
Metrics	**Values**
Accuracy	0.7286
Precision	0.77
Recall	0.64
F1 score	0.70

- AdaBoost Performance: AdaBoost shows solid results, with 73% accuracy, 77% precision, 64% recall, and an F1 score of 0.70, highlighting its capability in heart disease prediction.

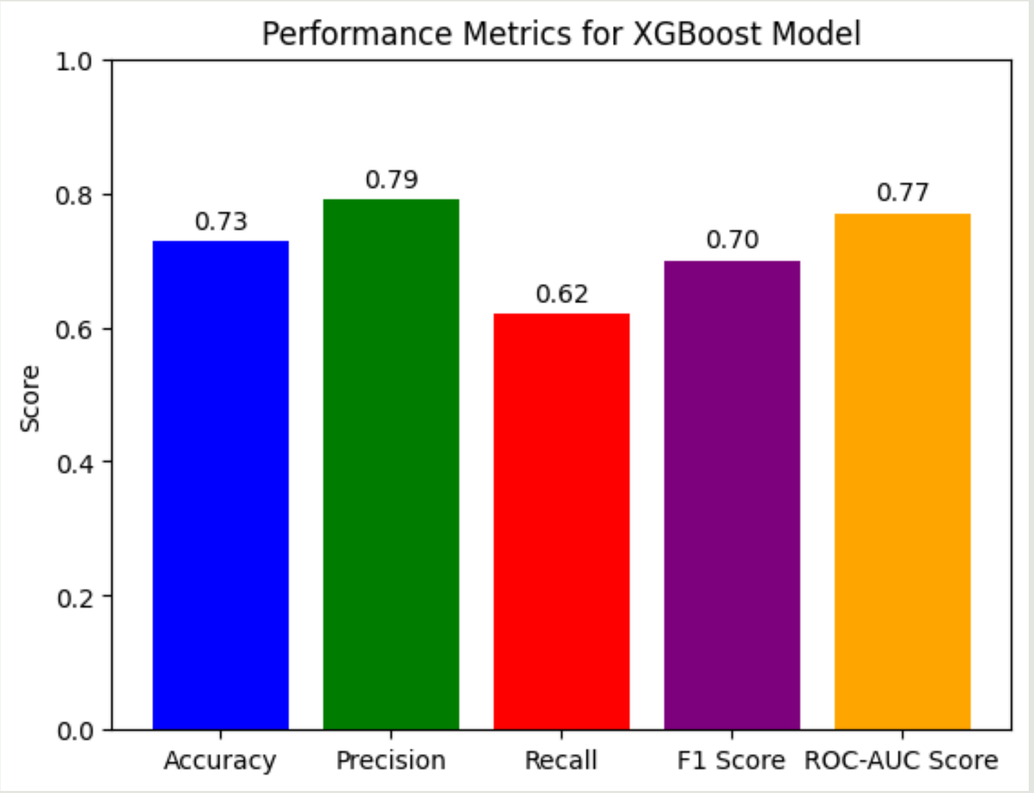
- XGBoost Performance: XGBoost outperforms AdaBoost in terms of precision (79%) and ROC-AUC score (77%), with an accuracy of 72.82% and recall of 62%.

- Key Metrics Comparison: XGBoost excels in precision and ROC-AUC, whereas AdaBoost balances precision and recall, achieving a good F1 score.

- Model Strengths: XGBoost's higher ROC-AUC score suggests better classification ability, while AdaBoost demonstrates reliable performance with a strong precision-recall balance.



A bar graph for performance metrics for Ada Boost



A bar graph for performance metrics for XG Boost

2.Comparison of Proposed Method and Other methods on Heart Disease Prediction

Author	Method Used	Accuracy
Baban Uttamrao et al. (2021)	Random Forest	80.0%.
R. Fadnavis et al. (2021)	Naive Bayes and Decision Trees	81.97%
Harshit Jindal et al (2020)	K-Nearest Neighbors (KNN), and Random Forest Classifier.	87.5%
Abhijeet Jagtap et al (2019)	SVM, Logistic Regression, and Naïve Bayes	60%
Our Study	XG Boost	72.82%

SUMMARY

In Phase 2 of our heart disease prediction project, we plan to expand our research by exploring and implementing new algorithms to enhance the model's predictive accuracy and reliability. This phase will focus on testing advanced machine learning techniques and optimizing the current models to ensure they perform effectively across diverse datasets. Additionally, we will develop a user-friendly frontend interface where users can input their health parameters, such as age, blood pressure, cholesterol levels, and more. This interface will be providing instant predictions on the likelihood of heart disease. This combination of algorithmic innovation and user-centered design will make our system both accessible and impactful.

INDIVIDUAL CONTRIBUTIONS

ARCHITA GUPTA 21BCE10225

I took responsibility for sourcing the dataset from Kaggle, which played a crucial role in the project. After preprocessing the data, I conducted feature selection and model training using the XGBoost algorithm. I identified key features such as Cholesterol, Systolic Blood Pressure, diastolic blood pressure, active and gender which significantly contributed to the model’s performance. Following the training with 30%dataset used for testing and 70% used for training, I achieved an accuracy of 72.82% and an AUC of 0.7765, highlighting the model's strong predictive capability. Additionally, I studied a 2024 paperheart disease prediction which provided valuable insights for refining my approach.

ABHINAV SHRIVASTAVA 21BCE10708

I identified the base research paper from 2024 on the XGBoost algorithm for the heart disease prediction project and handled the entire data preparation and analysis process. After sourcing the dataset from Kaggle, I performed exploratory data analysis (EDA) to understand feature distributions and checked for missing values using a heat map, confirming none were present. I used box plots for visualizing outlier removal, normalized key features like age, height, weight, systolic blood pressure and diastolic blood pressure MinMaxScaler, removed outliers with IQR technique, and filtered the dataset to ensure it was ready for training. This thorough preprocessing laid a strong foundation for the XGBoost model.

SONALI RAGHUWANSHI 21BCE10406

I played a crucial role in optimizing the AdaBoost algorithm using Python libraries to enhance its effectiveness for heart disease prediction. My focus was on meticulously tuning key hyperparameters, including learning rate, maximum depth, and gamma, which led to significant improvements in the model’s performance. I applied systematic techniques such as cross-validation to fine-tune these parameters effectively.

For the prediction model, I focused on five important features: systolic blood pressure (ap_hi), diastolic blood pressure (ap_lo), glucose (gluc), cholesterol, and weight. These features were crucial in improving the accuracy of the model.

These efforts culminated in achieving an impressive accuracy of 0.7286, highlighting the model’s reliability and generalizability. The model's performance was further validated with an outstanding ROC AUC score of 0.77, which was visualized using ROC curves, showcasing its exceptional ability to distinguish between patients with and without heart disease.

PRIYANSHI YADAV 21BCE10439

My contribution to this project involves detailed comparison of XGBoost and AdaBoost, identifying their strengths, weaknesses, and suitability for heart disease prediction based on performance metrics and interpretability. Through this comparative analysis, I aim to contribute valuable insights into the effectiveness of XGBoost and AdaBoost for heart disease prediction, informing the selection of appropriate machine learning algorithms for clinical applications and advancing the field of cardiovascular disease research. In additon to this I also helped in implementation of adaboost algorithm.

TINA CHELWANI 21BCE10669

I focused on the evaluation and implementation of the AdaBoost algorithm for heart disease prediction. I coded the AdaBoost algorithm and conducted extensive research to gain insights from academic papers, enhancing my understanding of the model's behaviour. My work involved analyzing performance metrics like accuracy, precision, recall, and F1-score, where AdaBoost achieved an accuracy of 69%. My classification analysis highlighted how AdaBoost performed across precision and recall metrics for different classes. Lastly, I ensured that the report included comprehensive references to existing literature, showcasing the value of machine learning in healthcare, especially for heart disease prediction.

REFERENCES:

https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset/data

https://www.sciencedirect.com/science/article/pii/S1746809421010533?casa_token=FGzE-UUrEJkAAAAA:RFB5rjq8y5ruzjeB2Z_YrLd10zMrunqBSiGrVUozIajleWLnY9jxJdF2GEpi1A8j-w9CyHI

https://www.jeeemi.org/index.php/jeeemi/article/view/440

https://www.sciencedirect.com/science/article/pii/S1319157820304936

https://www.mdpi.com/2078-2489/15/7/394

https://ieeexplore.ieee.org/abstract/document/10620208

https://www.sciencedirect.com/science/article/pii/S235291481830217X

Thank You