A COMPARATIVE STUDY OF HEART DISEASE PREDICTION USING HISTGRADIENT BOOSTING ALGORITHM

Project Report submitted in partial fulfillment of the Requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

by

Ch. ASHOK TEJA	**(1012102002)**
V. DIVYA	**(1012102009)**
G.D.S. VIGNESH	**(1012102010)**
B. HARSHA VARDHAN REDDY	**(1012102015)**

Under the Esteemed Guidance of

Sri. T. Mukthar Ahamed,

Academic Consultant

Department of Computer Science and Engineering

Y.S.R ENGINEERING COLLEGE OF YOGI VEMANA UNIVERSITY

PRODDATUR-516360, Y.S.R (Dt.), A.P.

(2024-2025)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Y.S.R. ENGINEERING COLLEGE OF YOGI VEMANAUNIVERSITY

PRODDATUR – 516360, Y.S.R. (Dt.), A.P.

CERTIFICATE

This is to certify that the project report entitled "A COMPARATIVE STUDY OF HEART DISEASE PREDICTIOPN USING HISTGRADIENT BOOSTING ALGORITHM" is submitted by Ch. Ashok Teja, V. Divya, G.D.S. Vignesh, B. Harsha Vardhan Reddy, in partial fulfillment of the requirement for the award of the Degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING, Y.S.R. ENGINEERING COLLEGE OF YOGI VEMANA UNIVERSITY, PRODDATUR, is a record of Bonafide work carried out by them under my guidance and supervision.

PROJECT GUIDE PROJECT CO-ORDINATOR HEAD OF THE DEPARTMENT

Examiners:

1. **2.**

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without mention of the people, who made it possible, whose constant guidance and encouragement crowned our efforts with success. We take this opportunity to express my deepest gratitude and appreciation to all those who have helped us directly or indirectly towards the successful completion of this project.

It is a great pleasure to express a deep sense of gratitude and veneration to our guide **Mr. T. Mukthar Ahamed**, Academic Consultant in the Department of Computer Science and Engineering for his valuable guidance and thought-provoking discussion throughout the course of project work.

We extend our profound gratefulness to our professor **Dr. S. KIRAN**, Associate professor and Head of the Computer Science and Engineering department for his encouragement and support throughout the project and also our project coordinator **Dr. R. PRADEEP KUMAR REDDY**, Associate Professor, esteemed faculty at Department of Computer Science & Engineering, who helped us a lot in achieving success of the project.

We take this opportunity to offer gratefulness to our Honourable Vice-Chancellor **Prof. ALLAM SRINIVASA RAO Garu**, our Dean of Engineering Prof. **K. VENKATA RAMANAIAH**, our Principal **Prof. B. JYARAMA REDDY** for providing all sorts of environment during the project work.

We express our gratitude to **Dr. ASHOK KUMAR**, Associate Professor in the department of Science and Humanities, for support and putting up the extra time to reach our targets, which was greatly helpful in the progression and smoothness of the entire project work.

We express our thanks to all our college teaching and non-teaching staff members who encouraged and helped us in some way or other throughout the project work.

Ch. Ashok Teja (1012102002)

V. Divya (1012102009)

G.D.S.Vignesh (1012102010)

B. Harsha Vardhan Reddy (1012102015)

LIST OF CONTENTS **PAGENO**

LIST OF FIGURES

LIST OF TABLES

A COMPARATIVE STUDY OF HEART DISEASE PREDICTION USING HISTGRADIENT BOOSTING ALGORITHM

ABSTRACT

These days, more people seem to be developing heart or blood related problems. The term cardiovascular disease encompasses any disease affiliated with the heart and the entire network of arteries, veins, and capillaries in the organism. Timely diagnosis, lifestyle alterations, and medical science significantly aid in the prevention and treatment of heart diseases. Conventional diagnostic methods depend on clinical testing and expert analysis, both of which can be time-consuming and prone to human error. Machine learning (ML) models do not require any expensive and time-consuming process. Once trained, they produce predictions quickly. Machine Learning offers a data-driven approach to accurately predict heart disease by analysing key health indicators such as age, cholesterol levels, blood pressure, diabetes status, and lifestyle factors. Machine learning has significantly improved the capability to predict heart disease by analysing medical data. In the existing methods, Support Vector Machines (SVM) is effective in identifying risk factors by finding patterns in structured and smaller datasets. On the other hand, Artificial Neural Networks (ANN) use deep learning to recognize difficult relationships in large datasets, making them useful for diagnosing heart conditions from various types of data, including medical images. The Hist Gradient Boosting (HGB) algorithm is a machine learning algorithm that uses a gradient boosting approach to build a strong predictive model; it has high speed and better performance than SVM; it is known for its accuracy and efficiency in handling various datasets; and it is used in the proposed study to compare the strengths and limitations of these methods in heart disease prediction. While both approaches increase prediction accuracy, ANN requires large amounts of training data, while SVM is more efficient for smaller datasets.

Keywords*: SVM, ANN, HistGradientBoosting, Linear Kernel.*

CHAPTER 1
INTRODUCTION

1.1 OVERVIEW

Heart disease is a major health problem that affects people all over the world. In recent years, more young people have been developing heart-related issues due to lack of exercise, unhealthy food habits, and high levels of stress. With modern lifestyles becoming more fast-paced, many individuals rely on processed foods and spend long hours sitting, which increases the risk of heart disease. Health conditions like high blood pressure, obesity, and diabetes also contribute to heart problems.

Since heart disease often develops slowly and may not show symptoms in the early stages, regular health checkups are important for early detection and prevention. Despite increased awareness, heart disease remains one of the leading causes of illness and death globally.

To prevent heart disease, making small but effective lifestyle changes is important. Eating a balanced diet, staying active, and managing stress can help keep the heart healthy. Avoiding smoking, limiting alcohol, and getting enough sleep also reduce the risk. Medical advancements, such as smart devices that track heart health and early detection tools, are helping doctors diagnose and treat heart conditions more effectively.

Governments and health organizations are also promoting awareness campaigns to encourage healthier living. However, the best way to stay safe is through prevention, and taking care of heart health should be a priority for everyone.

1.2 INTRODUCTION TO MACHINE LEARNING

One of the most important topics nowadays is Artificial Intelligence (AI). It is used in many different applications. Machine Learning (ML) is a branch of Artificial Intelligence that focuses on creating algorithms depending on the information presented. It is a novel strategy that adjusts the algorithm itself in accordance with the data to increase accuracy [1]. Based on the information given, the machine learning algorithms built a model that recognizes new developments. The following points justify why machine learning is so well-liked in modern times.

- Huge amount of data
- Enhancement of computing power
- Design of effective model with respect to data.

Machine Learning is a sub-part of Artificial Intelligence, that makes the work much easier because of the potential compilation of a large amount of computational data and it is referred to as a commercial revolution technique. Prior to this industrial revolution managed physical and mechanical strength, but this revolution concentrated on enhancing intellectual and cognitive ability. In this case, the system is treated as intellectual labour.

1.3 Concept of Machine Learning

Machine Learning has the ability to replicate and adjust according to the hypothesis. In each step, whatever the action generated is turned into something new to the system, which can be easily learned and understood to make it better.

In the development of Machine Learning algorithms, explicit programming is not involved, as zeta bytes of data are produced by the industries because of the automated digitization process and time-consuming tasks developed by the Machine Learning algorithms with appropriate predictive models. Machine learning algorithms entirely differ from traditional algorithms. In the case of a traditional process, logic, and input parameters are provided to the algorithm, which generates an appropriate output. But in the case of a Machine learning algorithm, along with input parameters, a hypothesis is given to an algorithm finally, which provides logic with the given information.

1.4 Machine Learning Life Cycle

When instruction is executed in the CPU, a list of steps is undergone in the generation of the result. The same kind of sequence has been used in the case of Machine Learning algorithms.

The following steps are involved in the case of the Machine Learning life cycle.

- Data gathering
- Data pre-processing
- Feature engineering
- Algorithm selection and training
- Making predictions

1.4.1 Data gathering: -

In this step, labeling plays a key role in case of identifying the data. In general, labeling has been done by humans who have domain knowledge; sometimes, they may refer to them as experts. When an algorithm needs a huge amount of information, the quality and quantity of information dictate the accuracy of the model.

1.4.2 Data pre-processing: -

Pre-processing is an essential step in computer vision, which identifies noise and eliminates it by using noise removal methods; likewise, in Machine Learning algorithms, data pre-processing ensures to avoidance of the following kinds of information.

- Missing values
- Outliers

- Bad encoding
- Wrong labeling
- Biased information

1.4.3 Feature Engineering: -

For any kind of data, the individual measurable property is the feature. Generally, the data which is given as an input consists of a set of attributes. Further, these are referred to as features of objects. For example, when mail is encountered to an e-mail id, it is necessary to categorize the mail as spam, promotion, social and primary. To categorize this incoming mail, it is essential to identify the content encountered. Depending on the content and with the previous learning mechanism, the mail has been categorized.

1.4.4 Algorithm selection and training: -

Machine Learning is categorized into three types supervised, unsupervised, and reinforcement. The appropriate category of the algorithm is selected and implemented depending on the type of available information.

After the selection of the algorithm, training has been done in two phases.

- Predict
- Adjust

Training is an incremental improvement process that evaluates different metrics according to the application and compares the solution. If the solution is not satisfied, appropriate adjustments are made within the algorithm and further evaluated with the given data. This two-phase implementation algorithm selection and training are represented the in the given figure 1.1.

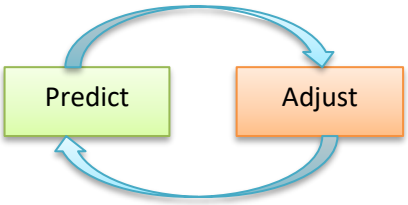


Figure 1.1: Phases of training

1.4.5 Making predictions: -

Predictions are identified in two phases. Training and Prediction phase. In the case of the training phase, samples along with labels are supplied to the learning model. From the given samples, appropriate features are extracted and executed with the learning model. Depending on the hypothesis defined, the learning model adjusts its parameters and finds a better solution. Whereas in the prediction phase, the features to be predicted are given as input for the trained

classifier, and further label is identified by the trained classifier. The entire process of making predictions is represented in figure 1.2.

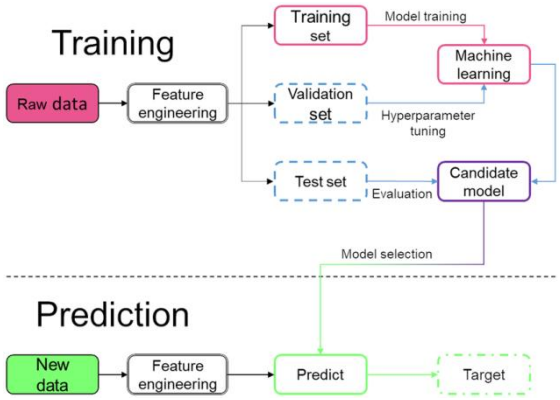


Figure 1.2: Training and Prediction process

1.5 Classification of Machine Learning

Machine Learning (ML) is classified into different types based on how models learn from data. The main classifications are:

- Supervised learning
- Unsupervised learning
- Reinforcement learning

1.5.1 Supervised Learning

In this model's case, data to be considered for evaluation with a model must consist of a label.

In this type of learning algorithm, two kinds of tasks are performed.

- Classification
- Regression

1.5.1.1 Classification: -

Classification is a task that categorizes the information depending on a categorical variable. A simple example of this is e-mail, which is already discussed. According to the classification, some important models are given below.

- K-Nearest Neighbor (KNN)
- Naïve Basie

- Logistic Regression
- Support Vector Machine
- Decision Trees
- Ensemble models

The ensemble models are multiple Machine Learning models gathered together to generate a good result.

1.5.1.2 Regression: -

When there is a necessity to predict the continuous values in a given problem, regression is applied. It is a kind of model practiced from previous information and generates feature information.

In this, some important models are given below.

- Logistic regression
- Lasso regression
- Ridge regression
- Support Vector Machine
- Decision trees

1.5.2 Unsupervised Learning

Clustering is one of the models in Machine Learning which collects similar objects and makes them as a group. Effective supervised models are not available to perform clustering. To do such unsupervised models play a key role. The most widely used unsupervised models are K-Means, K- Means++, K-Medoids, Agglomerative clustering, and clustering with large dataset algorithms. Unsupervised algorithms are represented without any labeled data. Based on the similarity in the input information, information is gathered and represented as groups.

1.5.3 Reinforcement Learning

Reinforcement learning is one of the areas in the Machine Learning technique which is used to identify possible behavior for a given model in a specific situation. It is differentiated from supervised learning in a way that supervised learning uses training data as an answer key for the model, whereas in reinforcement, there is no answer key, but the reinforcement agent decides how to execute the task in the absence of training data.

Key points: -

Input

- Output
- Training

In the case of input, it is to be referred to as the initial state from where the model will start. In the case of output variety of solutions may be generated for a given problem. In the training phase, depending on the input, the model will return an output. Here, the user needs to decide whether to reward or punish the model. The best response is decided depending on the max reward.

1.5.3.1 Types of Reinforcement: -

Reinforcement learning is divided into two types

- Positive Reinforcement learning
- Negative Reinforcement learning

Positive Reinforcement learning is defined as when an incident occurs due to specific behavior model strength is increased. In other words, it is to be referred to as a positive effect on learning. In negative reinforcement learning, in this case, the strength of the model is stopped or avoided.

1.6 Introduction to Heart Disease Prediction

Heart disease is one of the leading causes of mortality worldwide, affecting millions of people each year. Early detection and accurate diagnosis play a crucial role in preventing severe complications and improving patient outcomes. Traditional diagnostic methods, such as clinical assessments and medical tests, often require expert interpretation and can be time-consuming. With advancements in technology, machine learning and artificial intelligence (AI) have emerged as powerful tools to enhance heart disease prediction by analyzing large datasets and identifying patterns that may not be easily detectable by humans.

Machine learning models can process various risk factors, including age, blood pressure, cholesterol levels, blood sugar, and lifestyle habits, to predict the likelihood of heart disease. By leveraging historical patient data and advanced algorithms, these models can assist healthcare professionals in making more informed decisions. Techniques such as logistic regression, support vector machines (SVM), artificial neural networks (ANN), and ensemble learning methods have shown promising results in improving diagnostic accuracy.

The implementation of predictive models in heart disease diagnosis can lead to early intervention, personalized treatment plans, and better patient care. However, challenges such as data quality, feature selection, and model interpretability remain critical aspects that need to be addressed. As research in this field continues to evolve, integrating machine learning into healthcare systems has the potential to revolutionize the way heart disease is detected and managed, ultimately saving lives and reducing healthcare costs.

1.7 Problem Definition

Heart disease is a major public health concern, accounting for a significant number of deaths worldwide. Early detection and accurate prediction of heart disease are crucial for timely medical intervention and reducing mortality rates. Traditional diagnostic methods rely on clinical assessments, medical history, and laboratory tests, which can be time-consuming, expensive, and prone to human error. As a result, there is a need for an automated, data-driven approach that can assist healthcare professionals in identifying individuals at high risk of developing heart disease.

Machine learning techniques provide a potential solution by analyzing large datasets containing medical and lifestyle-related factors to predict heart disease with high accuracy. However, selecting the best-performing algorithm remains a challenge due to variations in model efficiency, interpretability, and computational complexity. While traditional models like Support Vector Machines (SVM) and Artificial Neural Networks (ANN) have been widely used, newer approaches, such as Histogram-Based Gradient Boosting (HGB), offer improved accuracy and efficiency by handling large datasets effectively.

The objective of this study is to develop and compare machine learning models, including HGB, SVM, and ANN, for heart disease prediction. By evaluating their performance using key metrics such as accuracy, precision, recall, and F1-score, the study aims to determine the most effective approach for early heart disease detection. The ultimate goal is to contribute to the development of a reliable and efficient predictive model that can assist in clinical decision-making, reduce diagnostic delays, and improve patient outcomes.

1.8 Project Purpose

The primary purpose of this project is to develop an accurate and efficient machine learning-based model for heart disease prediction, aiding in early diagnosis and timely medical intervention. By leveraging advanced machine learning techniques, including Histogram-Based Gradient Boosting (HGB), Support Vector Machines (SVM), and Artificial Neural Networks (ANN), this study aims to enhance prediction accuracy and provide a data-driven approach to healthcare decision-making.

1.8.1 Key Objectives

Heart disease is one of the leading causes of health problems worldwide. Detecting it early can help prevent serious complications. This project focuses on using machine learning to analyze health data and predict the risk of heart disease, helping doctors provide timely care and better treatment options.

- Early Detection of Heart Disease

Many people do not realize they have heart problems until the condition becomes severe. This project aims to identify individuals at risk before symptoms worsen. By analyzing factors like age, blood pressure, cholesterol levels, and other health indicators, the system can

predict the likelihood of developing heart disease. Early detection allows doctors to take preventive measures and guide patients toward healthier lifestyles.

- Comparing Different Machine Learning Methods

  There are several machine learning techniques for predicting heart disease, but some are more effective than others. This project compares three methods:

  Histogram-Based Gradient Boosting (HGB)

  Support Vector Machine (SVM)

  Artificial Neural Network (ANN)

By testing these models, we can determine which one provides the most accurate and reliable predictions. The goal is to find the best approach for identifying heart disease risk.

1.9 Project Features

This project focuses on predicting heart failure outcomes using machine learning by analyzing important clinical and demographic features. It utilizes models such as Histogram-Based Gradient Boosting (HGB), Support Vector Machines (SVM), and Artificial Neural Networks (ANN) to assess patient health based on factors like age, diabetes, high blood pressure, smoking, and key lab measurements (e.g., ejection fraction, serum creatinine, and platelets). By identifying patients at higher risk of heart failure-related fatalities, the system provides an early warning, enabling doctors to take preventive and timely medical actions.

To ensure the best predictive performance, the project compares different machine learning models using evaluation metrics such as accuracy, precision, recall, F1-score, and AUC-ROC. The HGB algorithm is particularly useful for handling complex patterns and large datasets, making it an efficient choice for processing medical records. The system also categorizes patients into low, moderate, and high-risk groups, allowing doctors to prioritize critical cases and provide better-targeted treatments. Additionally, the model is designed to be scalable, ensuring that it can be implemented in hospitals and healthcare institutions without excessive computational requirements.

The model provides visual reports, risk scores, and graphical representations to make predictions more understandable for both doctors and patients. Since patient data security is a top priority, the project follows strict privacy protocols and encryption measures to maintain confidentiality.

By ensuring high accuracy, efficiency, and ease of use, this project provides a reliable and effective solution for heart disease prediction. It aids in early detection, helping healthcare professionals make informed decisions and improve patient outcomes through timely medical interventions.

1.10 MODULES DESCRIPTION

1.10.1 DATASET

This module is the first and most important step in the project. It focuses on collecting and organizing patient health data that can help predict heart disease. The dataset includes important health details like age, blood pressure, diabetes, smoking habits, heart function, and key blood test results. The data is gathered from hospitals, medical research studies, and healthcare records to make sure it is accurate and reliable.

A well-organized dataset helps in finding patterns and understanding risk factors for heart disease. Good quality data is essential for building a strong prediction model. This step ensures that the project starts with the right information for effective results.

1.10.2 DATA PRE-PROCESSING

Once the data is collected, it needs to be cleaned and prepared before using it for predictions. This process includes removing missing or incorrect values, getting rid of duplicate records, and making sure all data is in a standard format. Some values, like blood pressure and cholesterol, need to be adjusted to match a common scale, while other details, like smoking status, need to be converted into numbers so that machine learning models can understand them. The dataset is then split into training and testing parts, so the model can learn from one part and be tested on another. This step ensures that the data is well-organized, error-free, and ready for accurate predictions.

1.10.3 TRAINING DATASET

The Training Dataset is a set of patient records used to teach the machine learning model how to predict heart disease. It contains different health factors that help the model understand patterns and connections between risk factors and disease outcomes. Before training, the dataset is cleaned and prepared to remove any issues that could affect accuracy. Different machine learning methods, such as Histogram-Based Gradient Boosting (HGB), Support Vector Machines (SVM), and Artificial Neural Networks (ANN), are used to identify important patterns. Adjustments are also made to improve performance. This step helps the model learn from past data so it can make better predictions in the future.

1.10.4 TEST DATASET

After the model is trained, it needs to be tested to see how well it performs. The Test Dataset is a separate set of patient records that the model has never seen before. It is used to check if the model can predict heart disease correctly. The system compares the predicted results with actual patient diagnoses to measure accuracy. This step ensures that the model works well with new data and does not just memorize old data. If the test results show any problems, the model is improved further. This part of the project is important to make sure the prediction system is reliable and useful in real-world situations.

1.10.5 RESULT EVALUATION

The last step is to check how well the heart disease prediction model is performing. Different measures are used to see if the predictions are correct. Accuracy tells us how often the model is right, while Precision and Recall help us understand how well the model detects actual heart disease cases without making mistakes. The F1-Score combines these measures for a balanced evaluation, and AUC-ROC helps determine how well the model can separate healthy patients from those at risk. The results are displayed in graphs**,** charts, and tables to make it easier to understand. The goal is to develop a reliable and accurate heart disease prediction system that can help doctors and patients make informed decisions about heart health.

CHAPTER 2
LITERATURE SURVEY

2. LITERATURE SURVEY

El-Sofany, Bouallegue, and Abd El-Latif (2024) [1] proposed a machine learning-based approach for heart disease prediction, integrating feature selection techniques such as chi-square, ANOVA, and mutual information to enhance model performance. They evaluated multiple algorithms, including SVM, XGBoost, bagging, decision trees, and random forests, using both private and public datasets. To address class imbalances, the study applied the SMOTE technique. The XGBoost model, combined with a selected feature subset, showed strong predictive capabilities. Additionally, the researchers developed an explainable AI method using SHAP to provide transparency in model decisions. A mobile application was also created to facilitate instant heart disease prediction based on user inputs, making early detection more accessible [1].

In their 2018 study, Nashif, Raihan, Islam, and Imam [2] proposed a cloud-based system for heart disease detection, combining machine learning algorithms with real-time cardiovascular health monitoring. They evaluated various machine learning models using the WEKA platform to identify the most accurate algorithm for heart disease prediction. The selected model was integrated into a mobile application, enabling users to input health parameters and receive immediate assessments of their heart disease risk. Additionally, they developed an Arduino-based monitoring system equipped with sensors to measure real-time physiological data such as heart rate, temperature, and humidity. This system transmits data to a central server every 10 seconds, allowing doctors to remotely monitor patients' health status. Alerts are triggered when sensor readings exceed predefined thresholds, facilitating timely medical intervention. The integration of machine learning with real-time data collection aims to enhance early detection and management of heart disease, providing a comprehensive approach to cardiovascular health monitoring.

Srinivasan et al. (2023) [3] investigated machine learning techniques for predicting cardiovascular heart disease using the UCI repository dataset. They tested eight ML classifiers to identify key features improving prediction accuracy. Learning Vector Quantization emerged as the most effective model. Other models, such as Naïve Bayes and Radial Basis Function networks, also performed well. The study emphasized the importance of feature selection in enhancing predictive performance. Data preprocessing techniques, including normalization and handling missing values, were applied to refine the dataset. Results highlighted the role of ML in early diagnosis and risk assessment. The authors advocated for AI-driven healthcare solutions, emphasizing their potential to reduce diagnostic errors. Their findings suggest ML can assist in timely medical interventions, leading to improved patient outcomes. A comparative analysis of the classifiers provided insights into model efficiency and reliability. The study contributes to advancing intelligent heart disease prediction models, helping in early detection and prevention. Future work may focus on refining models with larger, real-world datasets. The authors also suggested integrating deep learning techniques for enhanced accuracy. Incorporating real-time patient monitoring systems could further improve the effectiveness of ML models.

In their 2023 study published in *Diagnostics*, Ahmad Ayid Ahmad and Huseyin Polat [4] developed a machine learning model for heart disease prediction utilizing the Cleveland Heart Disease dataset. To enhance model performance and mitigate overfitting, they employed the Jellyfish Optimization Algorithm for feature selection, effectively reducing the dataset's dimensionality. This optimization technique is noted for its high convergence speed and flexibility in identifying optimal features. The refined dataset was then used to train various machine learning classifiers, including Support Vector Machine (SVM), Decision Tree (DT), Artificial Neural Network (ANN), and AdaBoost. Among these, the SVM classifier demonstrated superior performance, achieving a sensitivity of 98.56%, specificity of 98.37%, accuracy of 98.47%, and an area under the curve (AUC) of 94.48%. These results indicate that the combination of the Jellyfish Optimization Algorithm and SVM classifier offers a highly effective approach for heart disease prediction, potentially aiding in early diagnosis and improved patient outcomes.

Boukhatem, Youssef, and Nassif (2022) [5] explored the use of machine learning algorithms for heart disease prediction, evaluating multiple classifiers, including Support Vector Machine (SVM), Random Forest (RF), Multilayer Perceptron (MLP), and Naïve Bayes (NB). The study focused on preprocessing techniques such as feature selection and data balancing to improve predictive accuracy. Performance evaluation was conducted using metrics like precision, recall, and F1-score. Among the tested models, SVM exhibited the highest predictive performance. The findings highlighted the significance of machine learning in early heart disease detection and emphasized the importance of feature selection and model optimization. The research contributes to the development of AI-driven diagnostic tools, aiding in timely and accurate heart disease prediction.

Bhatt et al. (2023) [6] proposed a machine learning-based approach for heart disease prediction, aiming to improve diagnostic accuracy. They integrated k-modes clustering with Huang initialization to enhance classification performance. The study evaluated various models, including Random Forest (RF), Decision Tree (DT), Multilayer Perceptron (MP), and XGBoost (XGB). GridSearchCV was used for hyperparameter tuning to optimize model efficiency. Feature selection techniques were applied to identify key predictors of heart disease. The proposed method improved classification accuracy compared to traditional models. Results highlighted the potential of AI-driven approaches in early diagnosis. The study emphasized the importance of data preprocessing in achieving reliable predictions. Findings support the integration of machine learning in healthcare for better risk assessment. Future research could explore deep learning models for further improvements.

In their 2021 study, Rindhe et al. [7] addressed the critical issue of heart disease, a leading cause of mortality globally, by exploring the application of machine learning techniques for its prediction. They emphasized the necessity for reliable and accurate diagnostic systems to facilitate timely treatment. The research focused on utilizing data analytics to predict heart disease occurrences, leveraging patient data maintained over time. The authors discussed the implementation of machine learning algorithms, including Artificial Neural Networks (ANN), Random Forest (RF), and Support Vector Machines (SVM), highlighting their potential in automating the analysis of complex medical datasets. The study underscored the role of these

algorithms in enhancing diagnostic accuracy and supporting healthcare professionals in the early detection and management of heart-related diseases.

In their 2019 study, [8] Mohan, Thirumalai, and Srivastava introduced a novel hybrid machine learning model to enhance the prediction of heart disease. They employed the Hybrid Random Forest with Linear Model (HRFLM), which combines the strengths of Random Forest and linear methods, to improve predictive accuracy. The model was tested on the Cleveland Heart Disease dataset, focusing on identifying significant features through advanced feature selection techniques. The HRFLM model achieved an accuracy of 88.7%, outperforming traditional classification algorithms. This research highlights the potential of hybrid machine learning approaches in developing reliable diagnostic tools for early detection of cardiovascular diseases.

In their 2019 study, Ali et al. [9] introduced an expert system for heart failure prediction by stacking two Support Vector Machine (SVM) models. The first model employs L1 regularization to eliminate irrelevant features, while the second uses L2 regularization for prediction. A Hybrid Grid Search Algorithm (HGSA) was developed to optimize both models simultaneously. Evaluated using metrics like accuracy, sensitivity, specificity, and AUC, the proposed system demonstrated a 3.3% performance improvement over conventional SVM models and outperformed ten previously proposed methods.

Jabbar, Deekshatulu, and Chandra (2013) [10] proposed a heart disease classification model combining K-Nearest Neighbor (KNN) and Genetic Algorithm (GA). KNN classifies patients based on the similarity of their nearest neighbors, while GA optimizes feature selection. The hybrid approach enhances predictive accuracy by reducing irrelevant features. The study used a heart disease dataset to evaluate the model's effectiveness. Results showed improved classification performance compared to traditional KNN. The model helps in early detection of heart disease, aiding clinical decision-making. Feature selection plays a key role in improving diagnosis. The approach minimizes computational complexity while maintaining accuracy. The study highlights the significance of AI-driven techniques in healthcare. Future work could explore deep learning for further improvements.

In their 2017 study, [11] researchers developed a heart disease prediction system utilizing the Random Forest algorithm to analyze patient data for early detection of cardiovascular conditions. By training the model on 303 instances from the Cleveland Heart Disease dataset, the system achieved an accuracy of 85.81%, outperforming several other machine learning techniques. The study highlighted the effectiveness of Random Forest in handling the dataset's non-linear characteristics and emphasized the importance of selecting relevant attributes to enhance predictive performance. The authors concluded that implementing such a system could significantly aid in the timely diagnosis and treatment of heart disease, potentially saving lives.

In his 2024 comprehensive narrative review, Hajiarbabi [12] examines the application of machine learning techniques in heart disease detection. The study categorizes research into three primary areas: detection based on standard clinical information, electrocardiogram (ECG) and phonocardiogram (PCG) signals, and X-ray images. Findings indicate that methods such

as Extreme Gradient Boosting (XGBoost), Random Forest, Ensemble Learning, and Neural Networks outperform traditional machine learning approaches when utilizing standard clinical data. Additionally, the integration of dimensionality reduction techniques like Principal Component Analysis (PCA) enhances predictive performance. For ECG-based detection, Convolutional Neural Networks (CNNs) demonstrate superior efficacy. The review underscores the potential of machine learning in facilitating early diagnosis and improving patient outcomes in cardiovascular care.

Bani Hani and Ahmad (2023) [13] conducted a systematic review on machine learning algorithms for predicting ischemic heart disease (IHD). The study followed PRISMA guidelines and analyzed research from databases such as PubMed, ScienceDirect, CINAHL, and IEEE Explore. It focused on studies published between 2017 and 2021 that evaluated machine learning models for IHD prediction. The review assessed the performance of different algorithms, including Support Vector Machines (SVM), Random Forest (RF), Neural Networks (NN), and Ensemble Learning techniques. Feature selection methods like Principal Component Analysis (PCA) were also discussed for improving model performance. The study emphasized the role of data preprocessing in achieving reliable predictions. Results highlighted that hybrid and deep learning models often outperform traditional methods. The authors noted the importance of large, high-quality datasets for training robust models. The study pointed out challenges such as data imbalance and interpretability of ML models. The potential integration of AI in clinical decision-making was also discussed. Machine learning was identified as a promising tool for early diagnosis and risk stratification. The review underscored the need for further research on real-world implementation. Ethical considerations and patient privacy concerns were also highlighted. The study concluded that ML-based systems could significantly aid healthcare professionals in detecting IHD. Future research should focus on optimizing models with larger, diverse datasets.

In their 2023 study [14] presented at the IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), Das et al. explored the application of machine learning algorithms for heart disease detection. They evaluated six different machine learning models across four distinct datasets containing patient medical data related to heart disease. A key contribution of their research was the identification of the most significant features within the raw datasets, aiming to enhance predictive accuracy. The study's primary goal was to predict cardiovascular diseases (CVDs) using machine learning techniques, providing a comparative analysis with previously published results. The authors concluded that their findings could assist physicians in promptly and accurately identifying potential risk factors for heart disease, facilitating early intervention and prevention of CVDs.

In their 2023 study [15] presented at the 7th International Conference on Computing Methodologies and Communication (ICCMC), Gangadhar et al. investigated the application of machine learning (ML) and deep learning (DL) algorithms for accurately predicting the risk of coronary heart disease (CHD). Utilizing the Cleveland Heart Disease dataset, they implemented models including Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Tree (DT), Random Forest (RF), and Artificial Neural Networks (ANN). Their analysis focused on comparing the predictive performance of these algorithms, highlighting

the effectiveness of ANN in achieving higher accuracy after data preprocessing. The study underscores the potential of integrating ML and DL techniques to enhance early detection and risk assessment of CHD, contributing to improved patient outcomes.

In their 2023 study, Khader Basha S, Roja D, Santhj Priya S, et al. [16] proposed a hybrid machine learning approach for predicting and classifying coronary heart disease (CHD). They combined Decision Tree (DT) and Adaptive Boosting (AdaBoost) algorithms to enhance predictive accuracy. Utilizing the Framingham Heart Study dataset, which includes 16 features, they allocated 70% of the data for training and the remainder for testing. Performance metrics such as accuracy, True Positive Rate (TPR), and specificity were employed to evaluate the model's effectiveness. The study demonstrated that the hybrid DT-AdaBoost model outperformed individual algorithms, highlighting the potential of ensemble methods in improving CHD prediction and classification.

In their 2023 study, Jahed, Asser, and Al-Mousa [17] explored the use of personal key indicators alongside machine learning classifiers to predict heart disease. They analyzed various machine learning models to determine their effectiveness in identifying individuals at risk. The research highlights the importance of selecting relevant personal health indicators to enhance predictive accuracy. Their findings suggest that integrating these indicators with machine learning techniques can improve early detection and prevention strategies for heart disease. This approach underscores the potential of personalized data in developing more accurate and individualized healthcare solutions.

In their 2023 study [18] presented at the International Conference on Innovative Data Communication Technologies and Application (ICIDCA), Chopra, Karla, and Rani explored the application of machine learning and ensemble learning techniques for the identification of cardiovascular disease (CVD). Utilizing a dataset comprising various patient health indicators, they implemented multiple machine learning algorithms, including ensemble methods, to enhance predictive accuracy. The study demonstrated that ensemble learning approaches, which combine multiple models to improve performance, yielded superior results in detecting CVD compared to individual models. The authors concluded that integrating ensemble learning techniques can significantly aid in the early detection and diagnosis of cardiovascular diseases, thereby improving patient outcomes.

In their 2023 study [19] presented at the IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing Workshops (CCGridW), Gola and Arya introduced a classification model for heart disease prediction that integrates deep learning with the Satin Bowerbird Optimization (SBO) algorithm. The SBO algorithm, inspired by the mating behavior of the satin bowerbird, was employed for feature selection to identify the most relevant attributes from patient data. Subsequently, a deep learning model utilized these selected features to predict the risk of heart disease. The proposed approach aimed to enhance predictive accuracy and reduce overfitting issues commonly encountered in medical diagnostics. The study's results demonstrated improved performance over existing methods, highlighting the potential of combining nature-inspired optimization techniques with deep learning in e-healthcare applications.

In their 2023 study [20] presented at the International Conference on Innovative Data Communication Technologies and Application (ICIDCA), Shaik, Sreeja, Zainab, and colleagues investigated the application of various machine learning algorithms to enhance the accuracy of heart disease prediction. They evaluated multiple models, including logistic regression and K-nearest neighbors (KNN), to identify the most effective approach for predicting heart disease. The study demonstrated that logistic regression and KNN provided better prediction accuracy in a shorter amount of time compared to other machine learning techniques. The authors concluded that implementing these algorithms could significantly aid in the early detection and prevention of heart disease.

Shukla et al. (2023) [21] proposed a novel heart disease prediction system using machine learning, presented at the International Conference on Inventive Computation Technologies (ICICT) in Lalitpur. The study aimed to enhance diagnostic accuracy by analyzing patient health data through advanced computational techniques. The researchers explored various machine learning models to identify key risk factors and improve predictive performance. Their system integrates multiple classifiers to optimize the detection process, ensuring better reliability. Feature selection techniques were employed to refine the dataset and remove irrelevant variables. The study demonstrated that machine learning can assist healthcare professionals in making timely and informed decisions. The proposed system showed promising results in detecting heart disease at an early stage. The authors highlighted the importance of data preprocessing and model tuning to achieve optimal outcomes. Future research may focus on expanding datasets and incorporating deep learning for further improvements. Their findings emphasize the role of AI in modern healthcare, particularly in disease prevention and early diagnosis.

In their 2023 study [22] presented at the 4th International Conference for Emerging Technology (INCET), Kaustav Sen and Bindu Verma introduced a heart disease prediction model that employs a soft voting ensemble of Gradient Boosting Models (GBM), Random Forest (RF), and Gaussian Naive Bayes (GNB). This ensemble approach combines the strengths of these individual classifiers to enhance predictive accuracy. The researchers evaluated the model's performance using standard metrics and compared it against standalone classifiers. The results demonstrated that the ensemble method outperformed individual models, highlighting its potential in improving heart disease diagnosis. The study underscores the effectiveness of ensemble learning techniques in medical diagnostics, particularly for complex conditions like heart disease.

In their 2023 study presented at the Second International Conference on Electronics and Renewable Systems (ICEARS) in Tuticorin, Varshini G., Ramya A., Sravya C.L., and colleagues explored [23] enhancing heart disease prediction models through data transformation techniques. They employed Principal Component Analysis (PCA) and Relief Feature Selection to refine the dataset by identifying and retaining the most relevant features. By integrating these feature selection methods with various classifiers, the study aimed to improve predictive accuracy in diagnosing heart disease. The results indicated that models utilizing PCA and Relief demonstrated superior performance compared to those without such

data transformation, highlighting the importance of effective feature selection in medical diagnostics.

In their 2023 study presented at the International Conference on Electrical, Computer and Communication Engineering (ECCE) in Chittagong, Mahmud, Barua, Begum, and colleagues [24] introduced an enhanced framework for cardiovascular disease (CVD) prediction utilizing hybrid ensemble learning techniques. Their approach combines multiple machine learning classifiers to improve predictive accuracy and reliability in diagnosing CVD. The researchers employed advanced feature selection methods to identify the most relevant clinical attributes, thereby reducing computational complexity and enhancing model performance. The proposed hybrid ensemble model demonstrated superior accuracy compared to individual classifiers, highlighting its potential as a valuable tool for early detection of cardiovascular conditions. The study underscores the effectiveness of ensemble learning in medical diagnostics and suggests that integrating such models into healthcare systems could significantly aid in the timely identification and management of heart disease.

In their 2023 study presented at the IEEE 8th International Conference for Convergence in Technology (I2CT) in Lonavla, Ramesh HV and Pathinarupothi RK [25] conducted a performance analysis of various machine learning algorithms for predicting cardiovascular disease (CVD). The researchers evaluated multiple classifiers, including Naive Bayes (NB), Random Forest (RF), and Support Vector Machine (SVM), to determine their effectiveness in diagnosing heart conditions. Their findings indicated that the Random Forest model achieved the highest accuracy, suggesting its potential as a reliable tool for early detection of cardiovascular diseases. The study underscores the importance of selecting appropriate machine learning techniques to enhance predictive accuracy in medical diagnostics.

In their 2022 study published in *IEEE Access*, Abdellatif et al. [26] introduced a machine learning-based model for detecting heart disease and classifying its severity. To address the challenge of imbalanced datasets, they employed the Synthetic Minority Oversampling Technique (SMOTE), ensuring balanced class representation. The researchers evaluated six different machine learning classifiers, applying hyperparameter optimization using the Hyperband method to enhance performance. Utilizing two public datasets, their optimized Extra Trees (ET) classifier achieved detection accuracies of 99.2% and 98.52%, respectively. Additionally, the model attained a 95.73% accuracy rate in severity classification using the Cleveland dataset. These results suggest that the proposed approach could assist healthcare professionals in accurately determining a patient's heart disease status, facilitating early intervention and potentially reducing mortality rates.

The 2019 *IEEE CSCI* conference paper by Aram et al. [27] explores the use of deep learning to diagnose different types of heart disease from chest X-ray images. The authors employ convolutional neural networks (CNNs) to automatically extract features and classify heart conditions. Their approach leverages a large dataset of labeled chest X-rays to train and evaluate the model. The study demonstrates the effectiveness of deep learning in detecting heart abnormalities, reducing the need for manual interpretation. The results indicate that deep

learning techniques can enhance diagnostic accuracy and assist radiologists in identifying heart disease more efficiently.

The study by Patro and Padhy [28] proposes an ensemble approach for predicting cardiovascular disease using meta-classifier boosting algorithms. The authors utilize multiple machine learning classifiers and enhance their performance through boosting techniques to improve prediction accuracy. Their approach integrates models like Decision Trees, Random Forest, and Gradient Boosting to optimize classification performance. By leveraging ensemble learning, the study aims to minimize misclassification and enhance reliability in diagnosing cardiovascular disease. The results indicate that boosting algorithms effectively improve prediction accuracy compared to individual classifiers. The research highlights the potential of ensemble methods in advancing heart disease diagnosis using machine learning.

CHAPTER 3
EXISTING METHOD

3. EXISTING METHOD

3.1 Heart Disease Prediction Using Support Vector Machine and Artificial Neural Network

3.1.1 Overview

Heart disease refers to a range of conditions that affect the heart. It's a broad term, but it usually means problems with the heart's blood vessels, rhythm, or structure. The most common type is coronary artery disease, where arteries get clogged with plaque, reducing blood flow and possibly leading to chest pain (angina) or a heart attack. Other types include heart failure (when the heart can't pump well), arrhythmias (irregular heartbeats), and valve issues.

Heart disease is one of the top causes of death worldwide. It often builds up over time due to things like high cholesterol, high blood pressure, smoking, unhealthy eating, or not moving enough. Stress, diabetes, and family history can also play a part. Symptoms vary—some people feel chest pain or shortness of breath, while others might not notice anything until it's serious.

This paper presents a machine learning-based approach for heart disease prediction using Support Vector Machine (SVM) and Artificial Neural Network (ANN) models. It highlights the importance of early diagnosis due to heart disease's high mortality rate and the challenges posed by insufficient medical facilities. The dataset was collected from Kaggle and preprocessed, including feature selection using correlation heat maps. ANN achieved 86.6% accuracy, outperforming SVM's 81.6%. SVM constructs an optimal decision boundary, while ANN processes input through multiple hidden layers using ReLU and sigmoid activation functions. ANN outperforms SVM in precision, sensitivity, and specificity, making it a more reliable predictor. The study compares its models with existing machine learning-based systems, demonstrating competitive results. The proposed model can aid in early heart disease detection, reducing healthcare burdens. Future work may involve ensemble learning and deep learning for better accuracy.

3.1.2 ALGORITHMS

3.1.2.1 Support Vector Machine (SVM) Algorithm:

Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification and regression tasks. While it can handle regression problems, SVM is particularly well-suited for classification tasks.

SVM aims to find the optimal hyperplane in an N-dimensional space to separate data points into different classes. The algorithm maximizes the margin between the closest points of different classes.

Support Vector Machine (SVM) Terminology

- **Hyperplane**: A decision boundary separating different classes in feature space, represented by the equation **wx + b = 0** in linear classification.
- **Support Vectors**: The closest data points to the hyperplane, crucial for determining the hyperplane and margin in SVM.
- **Margin**: The distance between the hyperplane and the support vectors. SVM aims to maximize this margin for better classification performance.
- **Kernel**: A function that maps data to a higher-dimensional space, enabling SVM to handle non-linearly separable data.
- **Hard Margin**: A maximum-margin hyperplane that perfectly separates the data without misclassifications.
- **Soft Margin**: Allows some misclassifications by introducing slack variables, balancing margin maximization and misclassification penalties when data is not perfectly separable.
- **C**: A regularization term balancing margin maximization and misclassification penalties. A higher C value enforces a stricter penalty for misclassifications.
- **Hinge Loss**: A loss function penalizing misclassified points or margin violations, combined with regularization in SVM.
- **Dual Problem**: Involves solving for Lagrange multipliers associated with support vectors, facilitating the kernel trick and efficient computation.

3.1.2.2 How does Support Vector Machine Algorithm Work?

The key idea behind the SVM algorithm is to find the hyperplane that best separates two classes by maximizing the margin between them. This margin is the distance from the hyperplane to the nearest data points (**support vectors**) on each side.

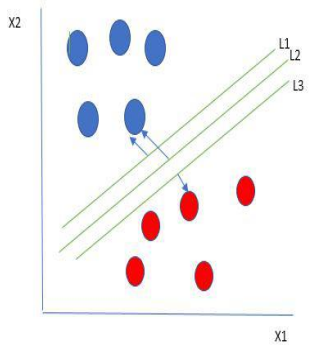


Figure 3.1: Multiple hyperplanes separate the data from two classes

The best hyperplane, also known as the **"hard margin,"** is the one that maximizes the distance between the hyperplane and the nearest data points from both classes. This ensures a clear separation between the classes. So, from the above figure, we choose L2 as hard margin.

Let's consider a scenario like shown below:

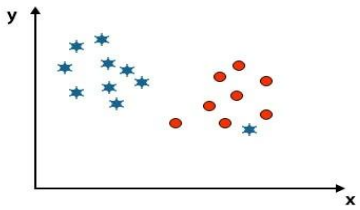


Figure 3.2: Selecting hyperplane for data with outlier

Here, we have one blue ball in the boundary of the red ball.

3.1.2.3 How does SVM classify the data?

It's simple! The blue ball in the boundary of red ones is an outlier of blue balls. The SVM algorithm has the characteristics to ignore the outlier and finds the best hyperplane that maximizes the margin. SVM is robust to outliers.

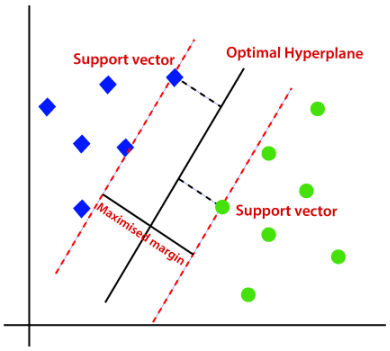


Figure 3.3: Hyperplane which is the most optimized one

A soft margin allows for some misclassifications or violations of the margin to improve generalization. The SVM optimizes the following equation to balance margin maximization and penalty minimization:

$$\text{Objective Function} = \left(\frac{1}{margin}\right) + \lambda \sum penalty$$

The penalty used for violations is often **hinge loss**, which has the following behaviour:

- If a data point is correctly classified and within the margin, there is no penalty (loss = 0).
- If a point is incorrectly classified or violates the margin, the hinge loss increases proportionally to the distance of the violation.

Till now, we were talking about linearly separable data (the group of blue balls and red balls are separable by a straight line/linear line).

3.1.2.4 What to do if data are not linearly separable?

When data is not linearly separable (i.e., it can't be divided by a straight line), SVM uses a technique called **kernels** to map the data into a higher-dimensional space where it becomes separable. This transformation helps SVM find a decision boundary even for non-linear data.

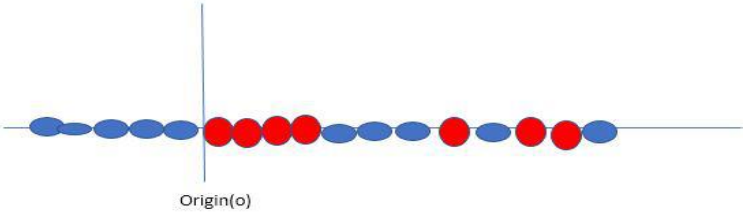


Figure 3.4: Original 1D dataset for classification

A **kernel** is a function that maps data points into a higher-dimensional space without explicitly computing the coordinates in that space. This allows SVM to work efficiently with non-linear data by implicitly performing the mapping.

For example, consider data points that are not linearly separable. By applying a kernel function, SVM transforms the data points into a higher-dimensional space where they become linearly separable.

- **Linear Kernel**: For linear separability.
- **Polynomial Kernel**: Maps data into a polynomial space.
- **Radial Basis Function (RBF) Kernel**: Transforms data into a space based on distances between data points.

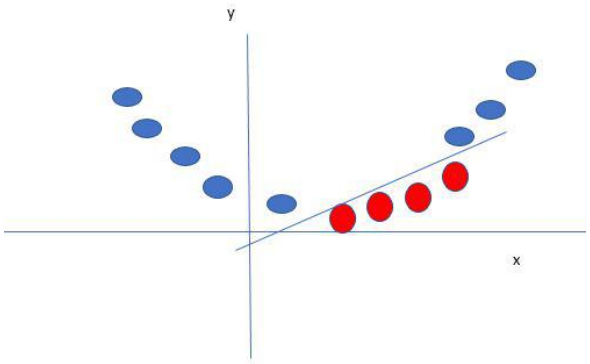


Figure 3.5: Mapping 1D data to 2D to become able to separate the two classes

In this case, the new variable y is created as a function of distance from the origin.

3.1.2.2 Artificial Neural Network (ANN) Algorithm:

The term "Artificial Neural Network" is derived from Biological neural networks that develop the structure of a human brain. Similar to the human brain that has neurons interconnected to one another, artificial neural networks also have neurons that are interconnected to one another in various layers of the networks. These neurons are known as nodes.

The schematic Biological Neural Network Diagram is shown in below figure-

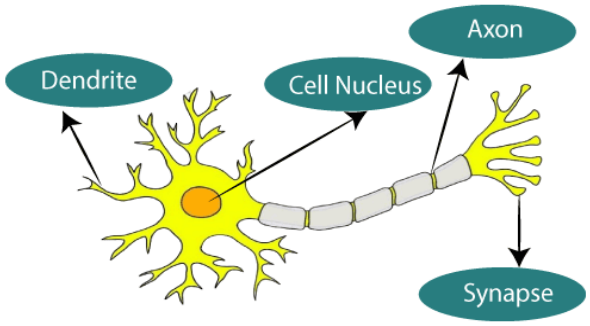


Figure 3.6: Schematic Diagram Of BNN

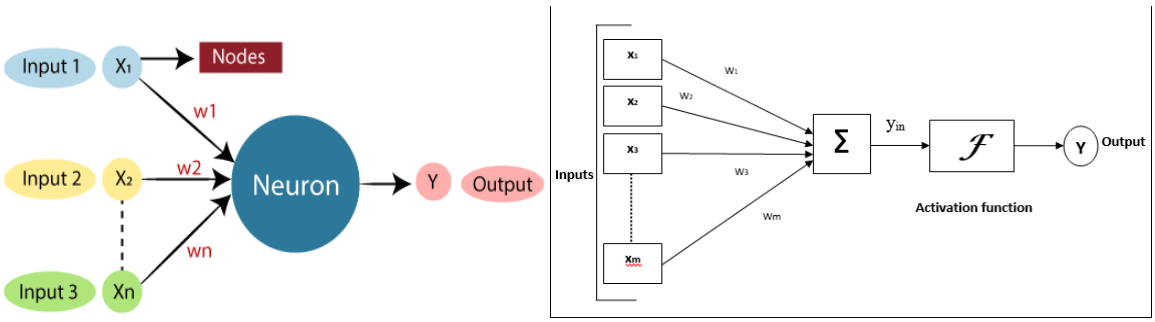


Figure 3.7: The general model of ANN followed by its processing.

Dendrites from Biological Neural Network represent inputs in Artificial Neural Networks, cell nucleus represents Nodes, synapse represents Weights, and Axon represents Output.

3.1.2.2.1 Architecture Of Artificial Neural Network (ANN):

To understand the concept of the architecture of an artificial neural network, we have to understand what a neural network consists of. In order to define a neural network that consists of a large number of artificial neurons, which are termed units arranged in a sequence of layers. Lets us look at various types of layers available in an artificial neural network.

Artificial Neural Network primarily consists of three layers:

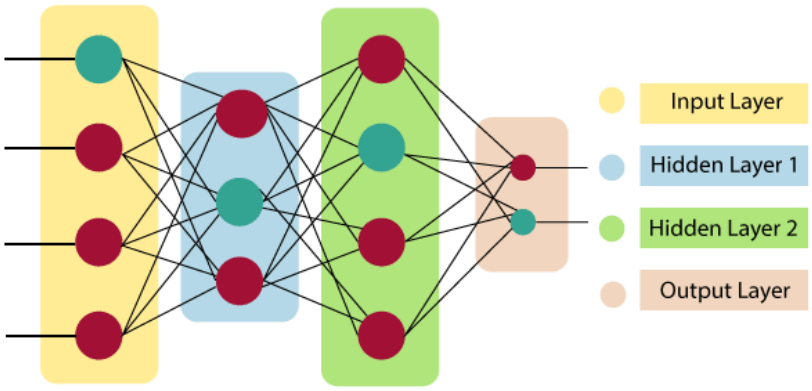


Figure 3.8: Layers of ANN

Input Layer

The input layer is the first layer of an Artificial Neural Network (ANN) and is responsible for receiving raw data. It accepts input in various formats, such as numerical values, text, or images, as provided by the programmer. Each neuron in this layer represents a specific feature of the input data. The primary function of the input layer is to pass the received data to the next layer without any processing. The number of neurons in this layer depends on the number of input features present in the dataset.

For the above general model of artificial neural network, the net input can be calculated as follows –

$$y_{in} = x_1.w_1 + x_2.w_2 + x_3.w_3 \ldots x_m.w_m$$

i.e., Net input $y_{in} = \sum_i^m x_i.w_i$

Hidden Layer

The hidden layer is located between the input and output layers and is responsible for processing data. It performs complex mathematical operations to extract hidden patterns and features from the input. Each neuron in the hidden layer applies a weight, a bias, and an activation function to transform the input data. The number of hidden layers and neurons determines the network's ability to learn and generalize patterns. While increasing hidden layers improves learning, it also increases computational complexity and training time.

Output Layer

The output layer is the final layer of the ANN, responsible for producing the network's results. It takes the processed data from the hidden layers and applies activation functions to generate the final output. The number of neurons in the output layer depends on the type of task being performed, such as classification or regression. For classification tasks, it can have multiple neurons corresponding to different classes, while for regression, it typically has a single neuron. The final output can be a numerical value, a probability, or a class label, depending on the problem being solved.

The output can be calculated by applying the activation function over the net input.

$$Y = F(y_{in})$$

Output = function *netinputcalculated*

3.1.2.2.2 How does an artificial neural network works?

Artificial Neural Network can be best represented as a weighted directed graph, where the artificial neurons form the nodes. The association between the neurons outputs and neuron inputs can be viewed as the directed edges with weights. The Artificial Neural Network receives the input signal from the external source in the form of a pattern and image in the form of a vector. These inputs are then mathematically assigned by the notations x (n) for every n number of inputs.

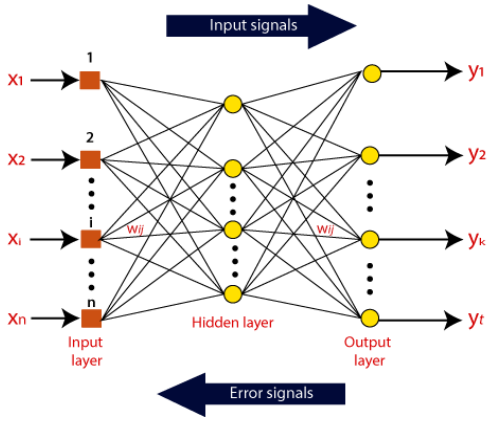


Figure 3.9: Working Of ANN

Afterward, each of the input is multiplied by its corresponding weights (these weights are the details utilized by the artificial neural networks to solve a specific problem). In general terms, these weights normally represent the strength of the interconnection between neurons

inside the artificial neural network. All the weighted inputs are summarized inside the computing unit.

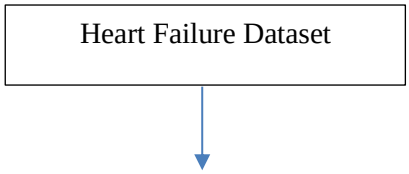
If the weighted sum is equal to zero, then bias is added to make the output non-zero or something else to scale up to the system's response. Bias has the same input, and weight equals to 1. Here the total of weighted inputs can be in the range of 0 to positive infinity. Here, to keep the response in the limits of the desired value, a certain maximum value is benchmarked, and the total of weighted inputs is passed through the activation function.

In an ANN, data flows forward from the input layer to the output layer through a process known as **forward propagation**. Each neuron takes inputs, multiplies them by assigned weights, adds a bias, and then applies an **activation function**. Activation functions, such as **Sigmoid, ReLU (Rectified Linear Unit), or Tanh**, introduce non-linearity, enabling the network to learn complex patterns. The processed information moves layer by layer until the final output is generated.

The performance of an ANN is evaluated using a **loss function**, which measures how far the predicted output is from the actual target. To improve accuracy, the network undergoes **backpropagation**, a method that adjusts the weights by calculating the gradient of the loss function. Using optimization algorithms like **Gradient Descent or Adam Optimizer**, the ANN updates the weights iteratively, minimizing the error over time.

3.1.3 Methodologies:

The methodology for heart disease prediction involves data collection, preprocessing, model development, and evaluation. The dataset from Kaggle contains 13 columns, with 12 independent features, including patient age (40-95 years) and gender. Preprocessing steps included checking for missing values, removing outliers, and feature selection using a correlation heatmap. Two models were implemented: Support Vector Machine (SVM) and Artificial Neural Network (ANN) algorithms. The steps involved in Heart disease prediction using SVM and ANN is shown in below figure ---



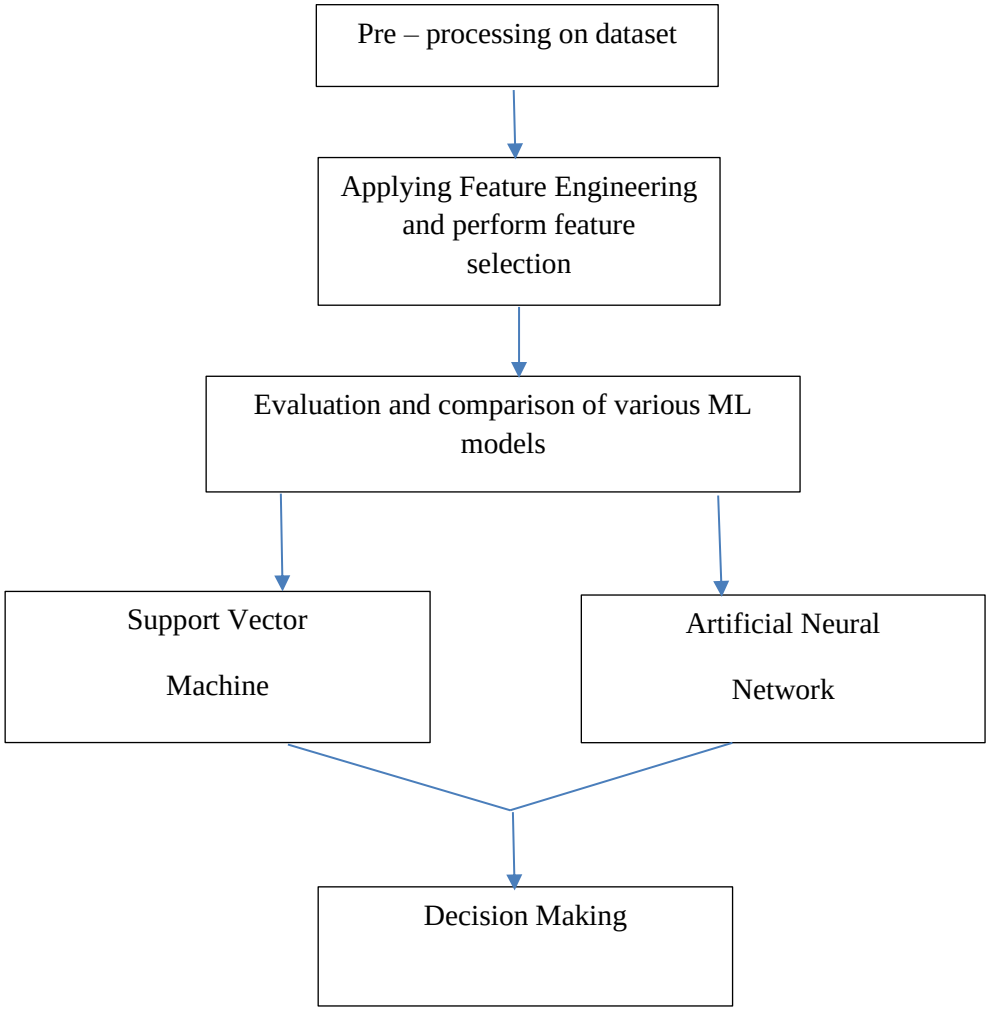


Figure 3.10: Model Flowchart

- **Heart failure dataset**

The work is based on the dataset collected from Kaggle. There are 13 columns in total. Among those, 12 are independent features using which prediction will be done. Here, patients from age 40 to 95 have been selected in this dataset. Male patients are denoted by a gender value 1 and female patients are denoted by a gender value 0.

- **Data pre-processing**

Before training the model, dataset has to be prepressed properly. It has been observed that there is an absence of null values in the dataset. Outliers from the dataset have also been removed in this phase.

- **Applying feature engineering and perform feature selection**

All features in any dataset are not equally important and even leading to negative influence in decision making. So feature selection is an important step in any decision-making system. Heat map has been applied for the selection of relevant features in this suggested model. Correlation heat map:

Heat map represents a two-dimensional information with the help of colors to identify the correlation of different features. Highly correlated features should be removed from the dataset for better performance.

We have checked the correlation using heat map. If two variables are highly correlated, we have dropped one of them. Here, it is checked which features are having correlation >0.1. Rest of the features are dropped.

- **Model development**

In this proposed work, SVM and ANN have been applied.

Support vector machine

One of the most well-known methods for supervised learning is the SVM. This is used in machine learning to solve issues involving both classification and regression

The SVM algorithm's objective is to establish the best line of decision boundary using which n-dimensional space can be divided into categories, so that we may conveniently classify additional data points in the future. This ideal decision boundary is referred to as a hyperplane. The extreme vectors or points that help create the hyperplane are chosen via SVM. These extreme circumstances are described by support vectors. The SVM algorithm is so named for this reason. After applying SVM in our model, the obtained accuracy is 81.6%.

Artificial neural network

A branch of artificial intelligence influenced by biology and modelled after the brain is referred to as ANNs. A computational network called an ANN is typically based on the biological neural network that created the structure of the human brain. The neurons in ANNs are also coupled to one another at different layers of the networks, just as the neurons in the human brain. These neurons are referred to as nodes.

The proposed ANN model has used rectified linear activation (ReLu) and sigmoid activation function for input layer and hidden layer, respectively. The initializer is used in the uniform initializer. Adam and binary_crossentropy have been used as optimizer and loss function, respectively.

After applying ANN in our model, Along with accuracy, other metrics considered are specificity, sensitivity, and precision. The model has achieved 93.54% precision, 87.87% sensitivity, and 83.33% specificity.

- **Result analysis**

The proposed model has been implemented applying SVM and ANN. ANN has achieved better accuracy over SVM. Support vector machine: The SVM classifier is fitted with the training set. Sklearn.svm package has been used to develop the SVM classifier. Since the SVM can be separated linearly Kernel='Linear' has been chosen. The model performance has been evaluated through generated confusion matrix, and accuracy metric has been calculated based on that. To create the confusion matrix, confusion matrix function of sklearn is imported. The calculated accuracy for our project is 81.6%. Achieved precision, sensitivity, and specificity scores are 97.14%, 77.27%, and 93.75%, respectively. The R2 score is 0.25. Artificial neural network: In ANN, sequential model type is used to build a model layer by layer. Each layer consists of weights that correspond to the layer that allows it. Layers are added with activation function ReLu and sigmoid. Units are mentioned as per need. Lastly, the model is compiled with optimizer "adam" and loss function= "binary_crossentropy." The model is trained with train set where epochs=100 and batch_size is 8. The performance of the proposed ANN model has been justified through confusion matrix and then accuracy value has been shown. The calculated accuracy for our project is 86.66%. Achieved precision, sensitivity, and specificity scores are 93.54%, 87.87%, and 83.33%. The R2 score has also been considered which is 0.27.

In this proposed model, both the SVM and ANN have been applied to predict heart disease considering their different advantages. In the result analysis section, the performance of both the classifiers has been analyzed in details and from there it has been observed that ANN has outperformed over the SVM.

- **Conclusion**

In this work, we have trained the machine using SVM and ANN. Applying SVM, accuracy achieved is 81.6%. The ANN model consists of hidden layers, and output of each layer is carried on to the next input which ensures greater accuracy. ANN model is giving an accuracy of 86.66%. This model can be utilized as a classifier for early and accurate prediction of heart disease.

In future, we would utilize different ensemble mechanism and deep learning approach.

CHAPTER 4
PROPOSED METHOD

4.1 Histogram-based Gradient Boosting (HGB):

Histogram-based Gradient Boosting (HGB) is an optimized machine learning algorithm that enhances traditional gradient boosting by implementing **histogram-based feature binning**. This technique significantly improves training speed, memory efficiency, and scalability, making it particularly suitable for large-scale datasets.

HGB follows the principles of ensemble learning and boosting, where multiple weak learners (decision trees) are trained sequentially to minimize errors. Unlike conventional gradient boosting models that evaluate each unique feature value individually, HGB groups feature values into discrete bins (histograms). This process reduces computational complexity, leading to faster model training and improved efficiency while maintaining high predictive accuracy.

Key Features of HGB

- **Fast Training** – Uses histogram binning to optimize computations, reducing processing time.
- **Memory Efficient** – Stores only bin statistics instead of individual feature values, minimizing memory usage.
- **Handles Large Datasets** – Reduces the number of feature comparisons, improving scalability for big data applications.
- **Automatic Handling of Missing Values** – No need for manual imputation; missing values are assigned to a separate bin and processed automatically.
- **Scalable for Classification and Regression** – Supports both types of learning problems efficiently across various industries.

HGB is particularly well-suited for handling massive datasets where traditional models struggle with slow processing speeds and high memory consumption. By using histogram-based binning, HGB significantly reduces the number of feature comparisons needed during training, leading to faster model convergence without compromising accuracy.

One of the most significant advantages of HGB is its ability to handle missing values automatically. Instead of requiring manual imputation methods, such as filling missing values with the mean or median, HGB assigns missing values to a separate bin. This allows the model

to learn patterns even when data points are missing, improving robustness and reducing preprocessing efforts.

Designed for scalability, HGB is highly effective for classification and regression tasks across various industries. It is widely used in applications such as:

- **Medical Diagnosis** – Predicting diseases based on large patient datasets.
- **Fraud Detection** – Identifying fraudulent transactions in financial systems.
- **E-commerce Optimization** – Improving recommendation systems for personalized user experiences.
- **High-dimensional Data Analysis** – Handling structured and unstructured data efficiently.

By leveraging feature binning, decision tree boosting, and automatic handling of missing data, HGB stands out as one of the most efficient gradient boosting techniques available today. It is particularly effective in big data applications, where speed, memory efficiency, and predictive accuracy are critical.

4.2 Working of HGB:

Histogram-based Gradient Boosting (HGB) follows a structured process to optimize model training and improve prediction accuracy. Unlike traditional gradient boosting, which evaluates every unique feature value, HGB groups similar values into bins (histograms), reducing computation time and memory usage. This makes it highly efficient for large datasets.

HGB's workflow consists of four key steps:

- Feature Binning (Histogram-based Splitting)
- Decision Tree Boosting
- Gradient Updates & Learning Rate Adjustment
- Automatic Handling of Missing Values

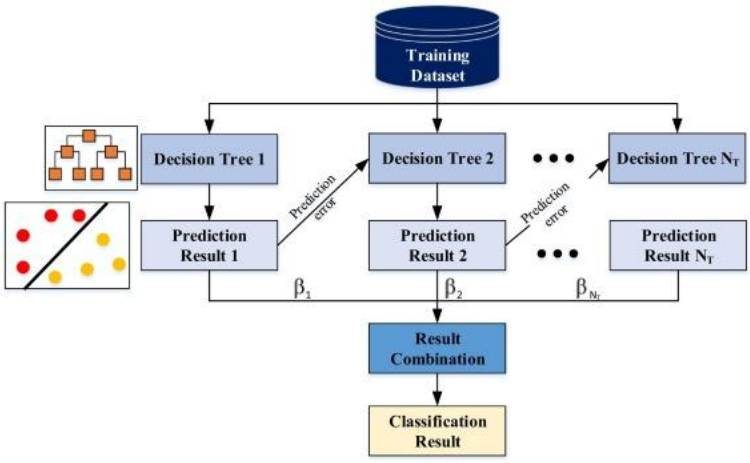


Figure 4.1: Workflow of Histogram-based Gradient Boosting (HGB)

Additionally, several hyper parameters influence the performance of the model, including learning rate, maximum iterations, tree depth, and histogram bin settings.

4.2.1 Feature Binning (Histogram-based Splitting):

What is Feature Binning?

Feature binning is a technique where continuous numerical values are grouped into discrete bins instead of being treated as individual values. This reduces the number of comparisons required when selecting the best split, making training faster and more memory-efficient.

Process of Feature Binning in HGB

- The range of feature values is divided into a fixed number of bins (e.g., 256 bins by default).
- Instead of storing each unique value, HGB stores only the summary statistics (such as mean gradient) for each bin.
- The model makes splitting decisions based on these bin statistics, significantly reducing the number of calculations.

Example

Consider a dataset where the feature "Age" contains values--

[22, 25, 30, 35, 40, 45, 50, 55, 60]

- Traditional gradient boosting evaluates all these values individually when selecting the best split.
- HGB groups these values into bins:
    - Bin 1: 20-30 → {22, 25, 30}
    - Bin 2: 30-40 → {35, 40}
    - Bin 3: 40-50 → {45, 50}
    - Bin 4: 50-60 → {55, 60}

Instead of checking nine values, HGB only checks four bins, reducing computation time.

Mathematical Formula for Feature Binning

Where:

$$Bj = \{Xj^{(1)}, Xj^{(2)}, \dots, Xj^{(k)}\}$$

- Bj represents the set of bins for feature$j$.
- $Xj^{(k)}$represents the range of values in bin k.

The best split is found using information gain:

Where:

$$Gain = \frac{G_L^2}{H_L} + \frac{G_R^2}{H_R} - \frac{(G_L + G_R)^2}{H_L + H_R}$$

- G_L, G_R represent the gradients of left and right child nodes.
- H_L, H_R represent the Hessian (second-order gradient information) for left and right child nodes.

A higher gain means a better split, which is selected during training.

4.2.2 Decision Tree Boosting:

What is Decision Tree Boosting?

Decision tree boosting is a process where multiple weak decision trees are combined sequentially to improve predictive performance. Each tree is trained to correct the errors made by the previous trees.

Process of Decision Tree Boosting

The first decision tree is trained to make an initial prediction.

- The second tree is trained to correct the mistakes of the first tree.
- The process continues, with each new tree improving upon the previous one.
- The final prediction is obtained by combining the outputs of all trees.

Mathematical Formula for Decision Tree Boosting

The model updates iteratively:

$$F_m(x) = F_{m-1}(x) + h_m(x)$$

Where:

- $F_m(x)$ Represents the updated prediction after adding tree m.
- $F_{m-1}(x)$ is the previous prediction.
- $h_m(x)$ is the newly added weak learner (decision tree).

4.2.3 Gradient Updates and Learning Rate Adjustment:

What is Gradient Updating?

Gradient boosting is based on the concept of iteratively minimizing residual errors. Each tree in the sequence is trained to reduce the errors of the previous tree.

Gradient Calculation Formula

$$g_i = \frac{\partial L(Y_i, \widehat{Y}_i)}{\partial \widehat{Y}_i}$$

Where:

- g_i represents the gradient (negative residual error).
- $L(Y_i, \widehat{Y}_i)$ is the loss function (e.g., Mean Squared Error).
- Y_i represents the actual target value.
- $\widehat{Y}_i$ represents the predicted value.

4.2.4 Handling Missing Values Automatically:

How Does HGB Handle Missing Values?

One of the major advantages of Histogram-based Gradient Boosting (HGB) is its automatic handling of missing values without requiring manual imputation techniques, such as replacing missing values with the mean, median, or mode. Unlike traditional gradient boosting models that require preprocessing to handle missing data, HGB treats missing values as a separate category and learns their significance during training.

Instead of discarding or imputing missing values, HGB automatically assigns them to a dedicated bin during the histogram binning process. This approach enables the model to determine whether missing values contain useful information for prediction. If the absence of a value correlates with the target variable, the model will leverage this information to improve predictive accuracy.

Process of Handling Missing Values in HGB

- **Feature Binning for Missing Values:**
    - During histogram binning, missing values are placed in a separate bin rather than being replaced by estimated values.
    - This prevents the introduction of artificial bias into the dataset.

- **Decision Tree Splitting Based on Missing Data:**
    - During training, the model evaluates whether missing values contribute useful predictive information.
    - If missing values help improve classification or regression accuracy, they are used as valid splits in the decision tree.
    - If missing values do not contribute useful information, they are ignored.
- **Gradient Update Consideration:**
    - Missing values are included when computing gradient updates, ensuring the model learns from them.
    - This process helps prevent data loss and improves robustness for real-world applications.

Mathematical Representation of Handling Missing Values

For a feature X_j with missing values:

$$B_{NaN=\{X_j|X_j=NaN\}}$$

Where:

- B_{NaN} represents the bin assigned for missing values in feature X_j.
- If missing values contribute to higher accuracy, they are used as part of the decision tree splits.
- If missing values are uninformative, they are ignored by the model.

4.3 Key Hyper parameters in Histogram-based Gradient Boosting (HGB):

Histogram-based Gradient Boosting (HGB) relies on several hyper parameters that control how the model learns and generalizes to new data. Proper tuning of these hyper parameters can significantly impact model performance, training time, and the risk of overfitting. Below is a detailed explanation of the key hyper parameters in HGB, along with relevant mathematical representations.

We use few Hyperparameters to tune:

- Learning Rate (η)
- Maximum Number of Iterations (max_iter)
- Maximum Depth of Trees (max_depth)
- Minimum Samples per Leaf (min_samples_leaf)
- Maximum Number of Leaf Nodes (max_leaf_nodes)
- Maximum Bins (max_bins)

Let us see all hyper parameters:

4.3.1 Learning Rate (η):

Definition:

The learning rate controls the contribution of each new tree to the final prediction. It is a step-size parameter that determines how much the model should adjust in each iteration to minimize the error.

Effect:

- A lower learning rate (e.g., 0.01) ensures gradual updates, reducing the risk of overfitting and improving generalization. However, this requires more boosting iterations to reach optimal performance.
- A higher learning rate (e.g., 0.5) speeds up training but may cause the model to overfit if too aggressive.

  Formula:

  The updated prediction at iteration is given by:

$$F_m(x) = F_{m-1}(x) + \boldsymbol{\eta}.\, h_m(x)$$

Where:

- $F_m(x)$ represents the updated prediction after adding tree m.
- $F_{m-1}(x)$ is the previous prediction.

- η is learning rate.
- $h_m(x)$ is the newly added weak learner (decision tree).

4.3.2 Maximum Number of Iterations (max_iter):

Definition:

This parameter specifies the maximum number of boosting rounds (trees) to be added sequentially to improve the model. It directly influences training time and model complexity.

Formula:

$$F(x) = \sum_{m=1}^{\max\_iter} h_m(x)$$

Where each $h_m(x)$ is a weak learner (decision tree).

Effect:

- A higher number of iterations allow the model to learn more complex relationships, but excessive iterations may lead to overfitting.
- A lower number of iterations may result in underfitting, where the model does not fully capture patterns in the data.
- A common practice is to pair max_iter with early stopping, which halts training once the validation loss stops improving.

4.3.3 Maximum Depth of Trees (max_depth):

Definition:

This parameter sets the maximum depth of each decision tree in the boosting process. It controls how deep the tree can grow before stopping.

Effect:

- **Shallow trees** (low max_depth, e.g., 2-3): Prevent overfitting but may underfit the data.

- **Deep trees** (high max_depth, e.g., 10-15): Capture complex relationships but increase the risk of overfitting.

Mathematical Intuition:

The maximum depth defines the number of consecutive splits a tree can make. Given a maximum depth d, the maximum number of leaves in a binary tree is:

$$L = 2^d - 1$$

Where L is the number of leaves. A large depth leads to more complex models, increasing the risk of capturing noise instead of actual patterns.

4.3.4 Minimum Samples per Leaf (min_samples_leaf):

Definition:

This parameter sets the minimum number of data points required in a leaf node for it to be valid. It ensures that trees do not create overly specific rules based on very few samples.

Effect:

- **Higher min_samples_leaf** (e.g., 10-20): Leads to simpler models, prevents overfitting, and smoothens predictions.
- **Lower min_samples_leaf** (e.g., 1-5): Allows more granular splits, capturing more details in the data but increasing the risk of overfitting.

4.3.5 Maximum Number of Leaf Nodes (max_leaf_nodes):

Definition:

This parameter sets an upper limit on the number of leaf nodes per tree, controlling tree complexity and preventing unnecessary splits.

Formula:

$$\text{Max Leaves} = \text{max\_leaf\_nodes}$$

A decision tree grows until:

- The number of leaves = max_leaf_nodes

- OR **stopping criteria** (such as min_samples_leaf) is met.

Effect:

- A **low max_leaf_nodes** (e.g., 10-20) results in a simpler tree structure, reducing overfitting.
- A **high max_leaf_nodes** allows the tree to grow larger, capturing fine-grained patterns but increasing computation time.

4.3.6 Maximum Bins (max_bins):

Definition:

Determines the number of discrete bins used for feature binning in histogram-based splitting.

Effect:

- A **higher max_bins** allows more precision.
- A **lower max_bins** reduces memory usage and improves speed.

Formula:

Instead of storing raw feature values, HGB discretizes them into bins:

Where:

- is the feature set.
- is the number of bins (default: 256).

This allows the algorithm to efficiently split features without storing large amounts of numerical data.

4.4 Comparison with Other Boosting Algorithms:

Boosting is a powerful ensemble learning technique that combines multiple weak learners (typically decision trees) in a sequential manner, where each new model corrects the errors of the previous ones. Different boosting algorithms have been developed to improve computational efficiency, predictive performance, and scalability.

The figure below compares **Histogram-based Gradient Boosting (HGB)** with **Gradient Boosting (GB), XGBoost, LightGBM, and AdaBoost** based on key characteristics.

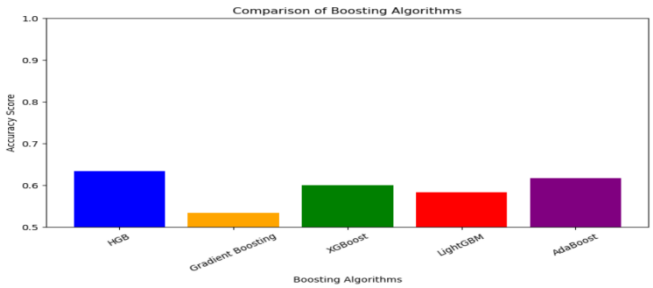


Fig 4.2: Comparison of Boosting algorithms

Feature	HistGradient Boosting (HGB)	Gradient Boosting (GB)	XGBoost (Extreme GB)	LightGBM (LGBM)	AdaBoost
Speed	Faster (Histogram binning)	Slower	Optimized Parallelization	Faster (Histogram learning)	Slower
Memory Efficiency	High (Binned values)	Moderate	Moderate	High (Large datasets)	Low (Full trees)
Large Datasets	Efficient	Struggles	Good	Best suited	Less efficient
Splitting Strategy	Histogram binning	Exact split search	Approximate & exact	Histogram binning	Exact split search
Overfitting Control	Built-in regularization	Requires careful tuning	Strong regularization (L1, L2)	Strong regularization (L1, L2)	Sensitive to noise
Parallelization	Limited	No native parallelism	Highly optimized	Highly optimized	Limited
Tree Growth	Leaf-wise (Optimized)	Level-wise	Depth-wise	Leaf-wise (Faster, deeper)	Depth-wise
Hyperparameter Tuning	Moderate	High	High	High	Low
Best Use Cases	Large datasets, structured data	General purpose	ML competitions	Large-scale apps	Simple models, noisy data

Table 4.1: Boosting Algorithm Comparison: HGB vs Others

4.5 Why Use HGB?

- **Faster Training:** Uses histogram-based binning for feature splitting, reducing computational complexity.
- **Memory Efficient:** Only stores binned values, making it ideal for large datasets.
- **Strong Generalization:** Includes built-in regularization to prevent overfitting.
- **Simple to Tune:** Requires fewer hyperparameter adjustments compared to XGBoost and LightGBM.
- **Great for Large Datasets:** Scales well and efficiently handles millions of data points.
- **High Accuracy:** Competes with XGBoost and LightGBM in predictive performance while being computationally efficient.

4.5.1 Advantages of HGB:

- **Faster Training:** HGB uses histogram-based binning, which significantly reduces the number of feature comparisons, making training much faster than traditional gradient boosting.
- **Memory Efficient:** Instead of storing all unique feature values, HGB stores only bin statistics, reducing memory usage and making it ideal for large datasets.
- **Handles Missing Values Automatically:** Unlike traditional models that require manual imputation, HGB assigns missing values to a special bin, allowing the model to learn whether missing data is informative.
- **Reduces Overfitting:** HGB supports regularization techniques like early stopping, min_samples_leaf, and max_leaf_nodes, which prevent overfitting and enhance generalization.
- **Better Handling of High-Cardinality Features:** HGB efficiently processes categorical features with many unique values by grouping them into bins instead of treating each value separately.
- **Scalability for Large Datasets:** Due to its computational efficiency, HGB is well-suited for big data applications, real-time predictions, and high-dimensional feature spaces.
- **Robust to Noisy Data:** By aggregating feature values into bins, the impact of outliers and noisy data is reduced compared to traditional gradient boosting models.

CHAPTER 5
RESULT & ANALYSIS

5. RESULT ANALYSIS AND DISCUSSION

5.1 PERFORMANCE EVALUATION METRICS OF HEART DISEASE PREDICTION

This section presents the performance evaluation of the HistGradient Boosting (HGB) algorithm compared to Support Vector Machine (SVM) and Artificial Neural Network (ANN) for heart disease prediction. The primary objective of this study is to determine which model provides the most accurate and reliable predictions based on key performance metrics. To assess the effectiveness of each model, we consider the following evaluation metrics – accuracy, precision, sensitivity, specificity. By comparing these metrics across the three models, we aim to identify the most efficient, accurate, and reliable approach for predicting heart disease. The results provide insights into the strengths and weaknesses of each model, helping to determine the suitability of HGB over traditional SVM and ANN approaches in medical prediction tasks.

To evaluate the performance of a classification model, various metrics derived from the **confusion matrix** are used. These metrics help analyze different aspects of a model's effectiveness, particularly in scenarios where accuracy alone is not sufficient.

5.1.1 Confusion Matrix

A **confusion matrix** is a tabular representation that compares actual and predicted values in a classification model. It helps in calculating various performance metrics.

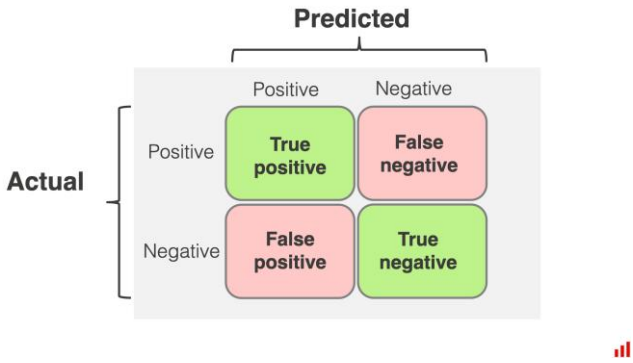


Figure 5.1: shows the confusion matrix

- **True Positive (TP):** The model correctly identifies a patient who actually has heart disease.
- **True Negative (TN):** The model correctly identifies a patient who does not have heart disease.
- **False Positive (FP):** The model incorrectly identifies a patient who actually has heart disease as healthy.
- **False Negative (FN):** The model incorrectly identifies a patient who actually is healthy as having heart disease.

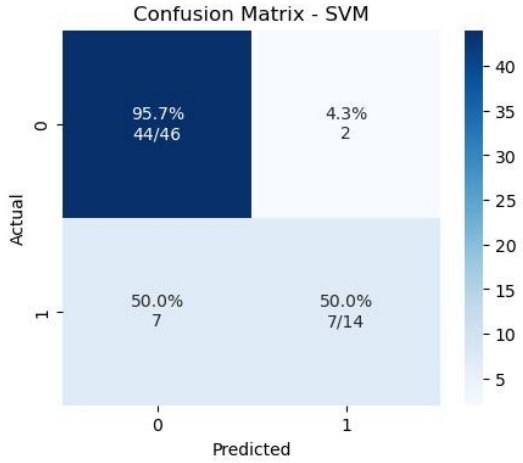


Figure 5.2: Confusion Matrix of SVM

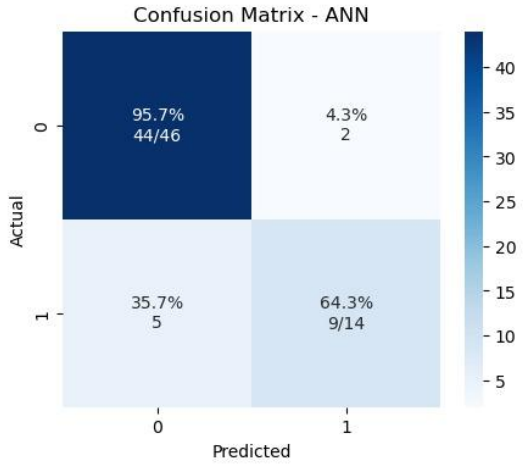


Figure 5.2: Confusion Matrix of ANN

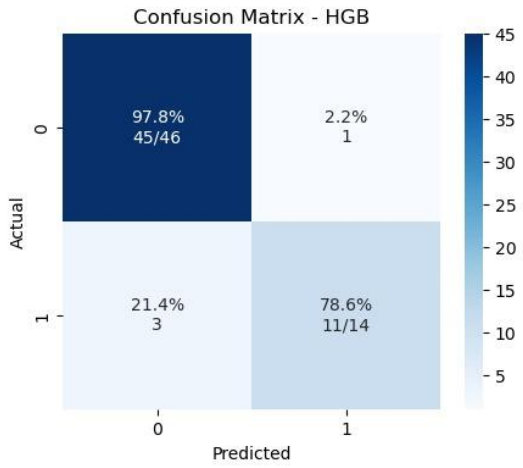


Figure 5.4: Confusion Matrix of HGB

The performance of the three models—SVM, ANN, and HistGradient Boosting (HGB)—was evaluated using confusion matrices to determine their effectiveness in heart disease prediction. All three models performed well in identifying non-heart disease cases (Class 0), with SVM and ANN correctly classifying 95.7% and HGB achieving a slightly higher rate of 97.8%. Overall, the HGB model demonstrated superior predictive ability, making it the most reliable among the three for heart disease detection.

5.2 Key Performance Metrics

5.2.1 Accuracy: - Overall Correctness of the Model

Definition:

Accuracy measures the proportion of correctly classified instances (both positive and negative) among the total number of predictions. It gives a general idea of how well the model performs but can be misleading if the dataset is imbalanced.

Formula:

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

5.2.2 Precision: - How Many Predicted Positives Are Actually Positive

Definition:

Precision (also called **Positive Predictive Value**) measures how many of the positive predictions made by the model are actually correct. It is useful in scenarios where false positives are costly, such as in fraud detection or medical testing.

Formula:

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

5.2.3 Recall (Sensitivity): - How Well the Model Detects Actual Positives

Definition:

Recall (also known as **Sensitivity or True Positive Rate**) measures how well the model identifies actual positive cases. It is crucial in applications like **cancer detection**, where missing a positive case (false negative) is dangerous.

Formula:

$$Sensitivity\ (Recall) = \frac{TP}{TP + FN}$$

5.2.4 Specificity: - How Well the Model Detects Actual Negatives

Definition:

Specificity (also known as **True Negative Rate**) measures how well the model identifies actual negative cases. It is important in applications where false positives are costly, such as **spam detection** or **criminal investigations**.

Formula:

$$Specificity = \frac{TN}{TN + FP}$$

5.2.5 F1-Score:-

The **F1 score** is a metric used to evaluate the performance of a classification model. It is the **harmonic mean of precision and recall**, which makes it a good measure when you want to balance both false positives and false negatives.

Formula:

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

SI.NO	Algorithms	Accuracy	Precision	Recall	F1 Score	Specificity
1.	SVM	85.00%	77.78%	50.00%	60.87%	95.65%
2.	ANN	88.33%	81.82%	64.29%	72.00%	95.65%
3.	HGB	93.33%	91.67%	78.57%	84.62%	97.83%

Table 5.1: Performance Metrics for heart disease prediction.

5.3 Model Performance Analysis:

The model performance of SVM, ANN and HGB is shown below figure 5.5, 5.6, 5.7, 5.8 as follows---

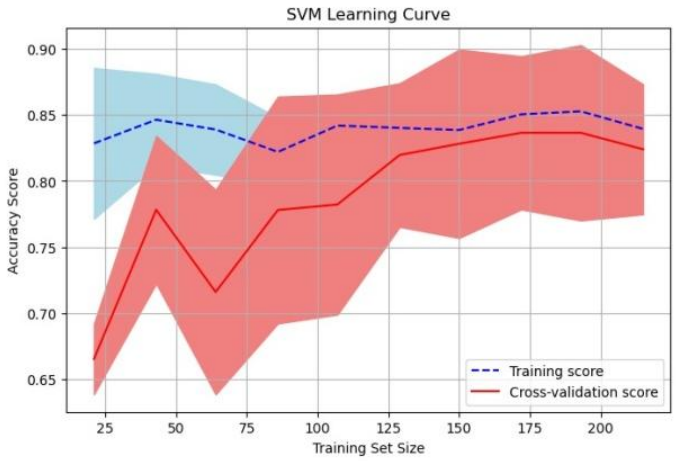


Figure 5.5: SVM Learning Curve

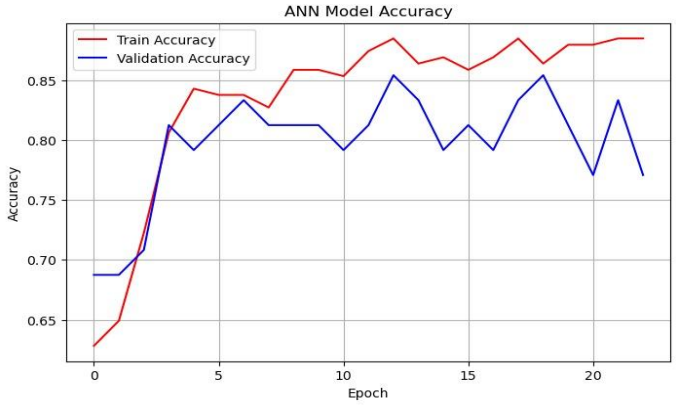


Figure 5.6: ANN Model Accuracy

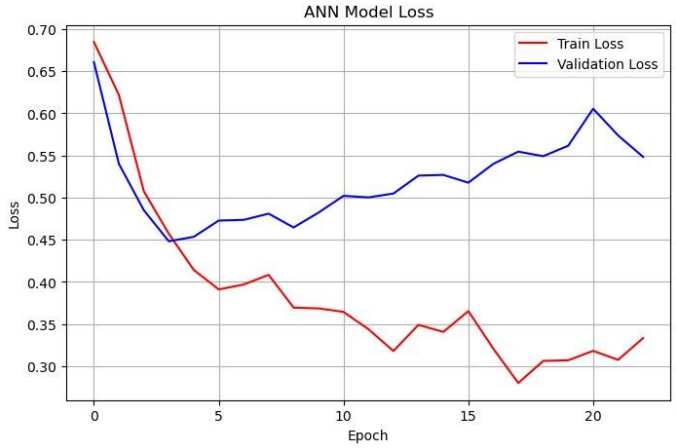


Figure 5.7: ANN Model Loss

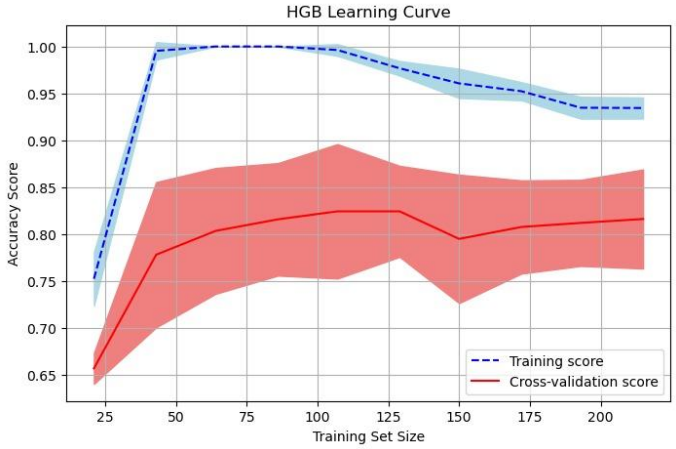


Figure 5.8: HGB Learning Curve

The above graphs show how well the SVM, ANN, and HGB models perform. The SVM model has stable training accuracy but fluctuating validation results. The ANN model improves quickly but starts overfitting after a few epochs, as seen in the loss and accuracy curves. The HGB model reaches very high training accuracy early but shows signs of overfitting due to the gap with validation scores. Overall, all models perform well, but tuning is needed for better generalization.

5.4 ROC-AUC & Model Performance Comparison:

The **ROC-AUC** (Receiver Operating Characteristic – Area Under the Curve) is a widely used evaluation metric for binary classification models. The **ROC curve** is a graphical representation that plots the **True Positive Rate (TPR)** against the **False Positive Rate (FPR)** at various threshold levels. The **TPR**, also known as sensitivity or recall, measures how well the model correctly identifies positive instances, while the **FPR** measures how often the model incorrectly labels negative instances as positive.

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

The **AUC** (Area Under the Curve) quantifies the overall ability of the model to discriminate between the positive and negative classes.

$$AUC = \sum_{i=1}^{n-1} (x_{i+1} - x_i).(\frac{y_{i+1} + y_i}{2})$$

Its value ranges from 0 to 1, where a value of **1.0** represents a perfect classifier, **0.5** indicates no discrimination (equivalent to random guessing), and a value less than 0.5 suggests the model is performing worse than random. A higher AUC value indicates better model performance, as it shows a greater capability to distinguish between the two classes across all possible classification thresholds.

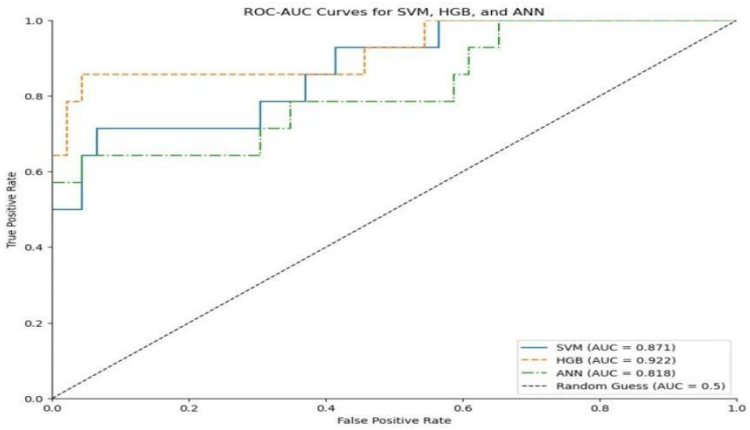


Figure 5.9: ROC-AUC Curves for SVM, ANN, and HGB

The figure presents ROC curves for three machine learning models—Support Vector Machine (SVM), Histogram-based Gradient Boosting (HGB), and Artificial Neural Network (ANN)—evaluating their performance in a binary classification task. The x-axis denotes the False Positive Rate (FPR), and the y-axis represents the True Positive Rate (TPR), with the Area Under the Curve (AUC) serving as a performance metric. SVM achieves an AUC of 0.871 (blue curve), HGB leads with the highest AUC of 0.922 (orange dashed curve), and ANN records an AUC of 0.818 (green dashed curve), indicating varying levels of effectiveness. The black diagonal line, representing a random guess with an AUC of 0.5, acts as a baseline, highlighting that all three models significantly outperform random classification, with HGB demonstrating the best ability to distinguish between positive and negative classes.

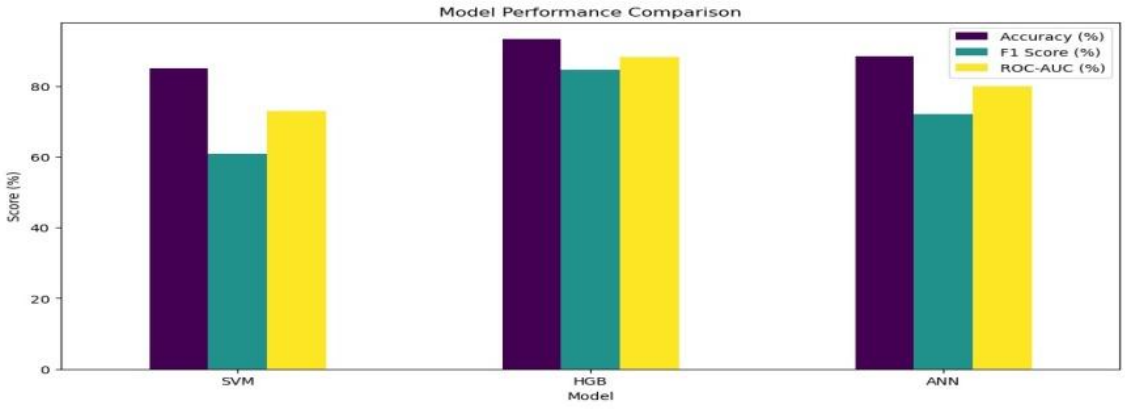


Figure 5.10: Model Performance Comparison

The bar chart compares the performance of SVM, HGB, and ANN models based on Accuracy (purple), F1 Score (teal), and ROC-AUC (yellow). HGB shows the highest scores

across all metrics, followed by ANN and SVM, with ROC-AUC consistently above 70% for all models.

5.5 Comparison of Train vs Test Accuracy for SVM, ANN, and HGB Models:

The comparison of train vs test accuracy for SVM, ANN, and HGB are shown in below as follows----

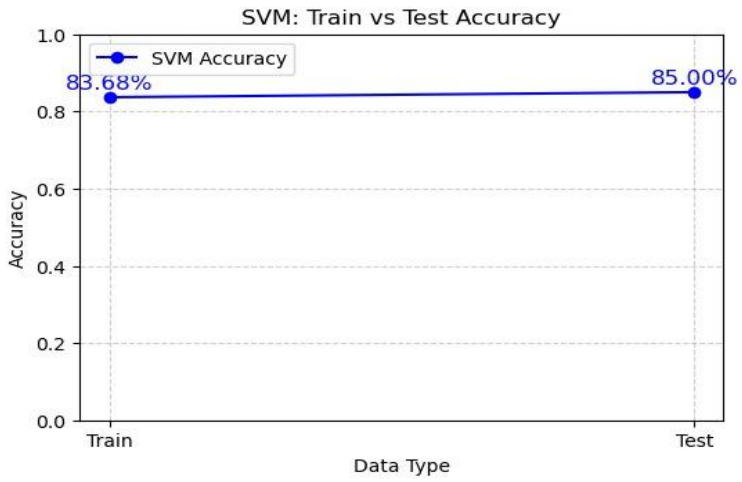


Figure 5.11: SVM - Train vs Test Accuracy

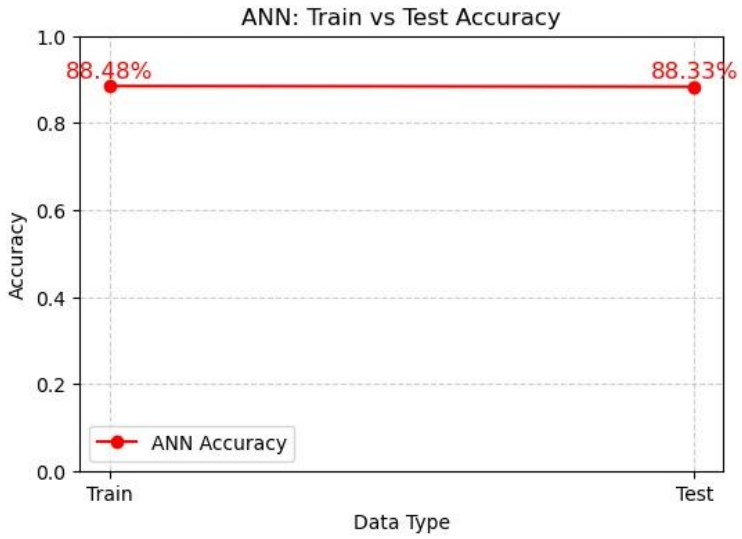


Figure 5.12: ANN – Train vs Test Accuracy

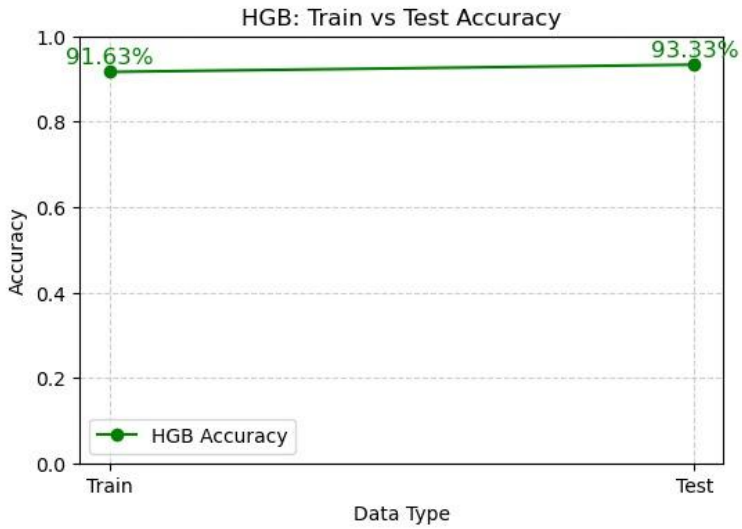


Figure 5.13: HGB – Train vs Test Accuracy

The figures illustrate the train and test accuracy of SVM, ANN, and HGB models, with SVM achieving 83.68% (train) and 85.00% (test), ANN at 88.48% (train) and 88.33% (test), and HGB at 91.63% (train) and 93.33% (test). Each model shows a slight increase or stability in accuracy from training to test data, indicating good generalization. HGB demonstrates the highest accuracy in both phases, followed by ANN and SVM. The close alignment between train and test accuracies suggests minimal overfitting across all models.

CHAPTER 6

CONCLUSION & FUTURE WORK

6. CONCLUSION & FUTURE WORK

6.1 Conclusion

This project presents a comparative study between Support Vector Machine (SVM), Artificial Neural Network (ANN), and the proposed HistGradient Boosting (HGB) algorithm for heart disease prediction. The evaluation of all three models was carried out using performance metrics such as accuracy, precision, recall, F1-score, and AUC-ROC. The goal was to determine the most efficient model for predicting heart disease based on medical data. Among the models tested, the HGB algorithm consistently outperformed SVM and ANN across all evaluation metrics. It achieved higher accuracy.

The HGB model was efficient and fast. It can handle challenges like missing data, irrelevant information, and uneven distribution of cases (such as more healthy patients than sick ones) better than other models. Moreover, it required less time to train compared to Artificial Neural Networks, which can be slow and need a lot of resources. This makes HGB a great choice for systems that need quick and accurate predictions.

6.2 Future Work

In the future, there are several ways to enhance the performance and reliability of the heart disease prediction model. One approach is to improve the model by carefully tuning its parameters using advanced methods, which can lead to more accurate results. Additionally, incorporating more diverse patient information—such as family medical history, lifestyle habits, or real-time clinical data—can help the model better understand health risks and make more precise predictions. As the data becomes more complex or includes time-based inputs like ECG signals, advanced deep learning algorithms such as Convolutional Neural Networks (CNNs) or Long Short-Term Memory networks (LSTMs) can be explored for better pattern recognition. Furthermore, other efficient machine learning models like XGBoost, CatBoost, and LightGBM can be considered, as they work well with large and complex data, offer high accuracy and fast performance on large datasets. By exploring these improvements, the system can become more accurate, faster, and more helpful in supporting early detection of heart disease.

CHAPTER 7

BIBILIOGRAPHY

7. BIBILIOGRAPHY

[1] El-Sofany H, Bouallegue B, El-Latif YMA. A proposed technique for predicting heart disease using machine learning algorithms and an explainable AI method. Sci Rep. 2024 Oct 7;14(1):23277. doi: 10.1038/s41598-024-74656-2. PMID: 39375427; PMCID: PMC11458608.

[2] Nashif, Shadman & Raihan, Rakib & Islam, Md Rasedul & Imam, Mohammad. (2018). Heart Disease Detection by Using Machine Learning Algorithms and a Real-Time Cardiovascular Health Monitoring System. World Journal of Engineering and Technology. 06. 854-873. 10.4236/wjet.2018.64057.

[3] Srinivasan S, Gunasekaran S, Mathivanan SK, M B BAM, Jayagopal P, Dalu GT. An active learning machine technique based prediction of cardiovascular heart disease from UCI-repository database. Sci Rep. 2023 Aug 21;13(1):13588. doi: 10.1038/s41598-023-40717-1. Erratum in: Sci Rep. 2024 Jul 23;14(1):16905. doi: 10.1038/s41598-024-66981-3. PMID: 37604952; PMCID: PMC10442398.

[4] Ahmad, A.A.; Polat, H. Prediction of Heart Disease Based on Machine Learning Using Jellyfish Optimization Algorithm. *Diagnostics* 2023, *13*, 2392. https://doi.org/10.3390/diagnostics13142392.

[5] C. Boukhatem, H. Y. Youssef and A. B. Nassif, "Heart Disease Prediction Using Machine Learning," 2022 Advances in Science and Engineering Technology International Conferences (ASET), Dubai, United Arab Emirates, 2022, pp. 1-6, doi: 10.1109/ASET53988.2022.9734880. keywords: {Heart;Support vector machines;Radio frequency;Machine learning algorithms;Focusing;Forestry;Predictive models; heart disease prediction; machine learning; support vector machine; multilayer perceptron;naïve Bayes; random forest},

[6] Bhatt, C.M.; Patel, P.; Ghetia, T.; Mazzeo, P.L. Effective Heart Disease Prediction Using Machine Learning Techniques. Algorithms 2023, *16*, 88. https://doi.org/10.3390/a16020088.

[7] Rindhe, Baban & Ahire, Nikita & Patil, Rupali & Gagare, Shweta & Darade, Manisha. (2021). Heart Disease Prediction Using Machine Learning. International Journal of Advanced Research in Science, Communication and Technology. 267-276. 10.48175/IJARSCT-1131.

[8] S. Mohan, C. Thirumalai and G. Srivastava, "Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques," in IEEE Access, vol. 7, pp. 81542-81554, 2019, doi: 10.1109/ACCESS.2019.2923707. keywords: {Diseases;Heart;Data mining;Support vector machines; Feature extraction; Machine learning; Predictive models; Machine learning; heart disease prediction; feature selection; prediction model; classification algorithms; cardiovascular disease (CVD)},

[9] L. Ali et al., "An Optimized Stacked Support Vector Machines Based Expert System for the Effective Prediction of Heart Failure," in IEEE Access, vol. 7, pp. 54007-54014, 2019, doi: 10.1109/ACCESS.2019.2909969. keywords: {Support vector machines; Predictive models;Heart;Expert systems;Kernel;Optimization;Diseases;Clinical expert system; feature selection; heart failure prediction; hybrid grid search algorithm; support vector machine},

[10] M. Akhil jabbar, B.L. Deekshatulu, Priti Chandra, Classification of Heart Disease Using K-Nearest Neighbor and Genetic Algorithm, Procedia Technology, Volume 10, 2013, Pages 85-94, ISSN 2212-0173, https://doi.org/10.1016/j.protcy.2013.12.340.

[11] Singh, Y.K., Sinha, N., Singh, S.K. (2017). Heart Disease Prediction System Using Random Forest. In: Singh, M., Gupta, P., Tyagi, V., Sharma, A., Ören, T., Grosky, W. (eds) Advances in Computing and Data Sciences. ICACDS 2016. Communications in Computer and Information Science, vol 721. Springer, Singapore. https://doi.org/10.1007/978-981-10-5427-3_63.

[12] Hajiarbabi M. Heart disease detection using machine learning methods: a comprehensive narrative review. J Med Artif Intell 2024;7:21.

[13] Bani Hani SH, Ahmad MM. Machine-learning Algorithms for Ischemic Heart Disease Prediction: A Systematic Review. Curr Cardiol Rev. 2023;19(1):e090622205797. doi: 10.2174/1573403X18666220609123053. PMID: 35692135; PMCID: PMC10201879.

[14] Das, Ranjit Chandra & Das, Madhab Chandra & Hossain, Md & Rahman, Md & Hossen, Helal & Hasan, Rakibul. (2023). Heart Disease Detection Using ML. 0983-0987. 10.1109/CCWC57344.2023.10099294.

[15] Gangadhar, Mandadi & Sai, Kalyanam & Kumar, Salem & Kumar, Kanaparti & Kavitha, Modepalli & Aravinth, S.S.. (2023). Machine Learning and Deep Learning Techniques on

Accurate Risk Prediction of Coronary Heart Disease. 227-232. 10.1109/ICCMC56507.2023.10083756.

[16] D., Roja & Vellela, Sai & Sk, Khader Basha & B., Venkateswara Reddy. (2023). Coronary Heart Disease Prediction and Classification using Hybrid Machine Learning Algorithms. 10.1109/ICIDCA56705.2023.10099579.

[17] Jahed R, Asser O, Al-Mousa A. Using Personal Key Indicators and Machine Learning-based Classifiers for the Prediction of Heart Disease. 2023 International Conference on Smart Computing and Application (ICSCA). Hail: IEEE; 2023.

[18] Chopra, Shreya & Kalra, Nidhi & Rani, Rinkle. (2023). Identification of Cardiovascular Disease using Machine Learning and Ensemble Learning. 186-192. 10.1109/ICIDCA56705.2023.10099508.

[19] Gola, Kamal & Arya, Shikha. (2023). Satin Bowerbird Optimization-Based Classification Model for Heart Disease Prediction Using Deep Learning in E-Healthcare. 296-298. 10.1109/CCGridW59191.2023.00063.

[20] Shaik, Mohammed & Sreeja, Radhandi & Zainab, Safa & Sowmya, Panthangi & Akshay, Thipparthy & Sindhu, Sudireddy. (2023). Improving Accuracy of Heart Disease Prediction through Machine Learning Algorithms. 41-46. 10.1109/ICIDCA56705.2023.10100244.

[21] Sharma, Vibhor. (2023). A Novel Prediction System to Diagnose Heart Disease. 10.1109/ICICT57646.2023.10133988.

[22] Sen, Kaustav & Verma, Bindu. (2023). Heart Disease Prediction Using a Soft Voting Ensemble of Gradient Boosting Models, RandomForest, and Gaussian Naive Bayes. 1-7. 10.1109/INCET57972.2023.10170399.

[23] Varshini, Guggulla & Ramya, Ananthaneni & Sravya, Chitrakavi & Kumar, Vinod & Shukla, Brajesh. (2023). Improving Heart Disease Prediction of Classifiers with Data Transformation using PCA and Relief Feature Selection. 1644-1649. 10.1109/ICEARS56392.2023.10085401.

[24] Mahmud, Tanjim & Barua, Anik & Begum, Manoara & Chakma, Eipshita & Das, Sudhakar & Sharmen, Nahed. (2023). An Improved Framework for Reliable Cardiovascular

Disease Prediction Using Hybrid Ensemble Learning. 1-6. 10.1109/ECCE57851.2023.10101564.

[25] Ramesh, Hridya & Pathinarupothi, Rahul. (2023). Performance Analysis of Machine Learning Algorithms to Predict Cardiovascular Disease. 1-8. 10.1109/I2CT57861.2023.10126428.

[26] Abdellatif, Abdallah & Abdellatef, Hamdan & Kanesan, Jeevan & Chow, Chee Onn & Chuah, Joon Huang & Gheni, Hassan. (2022). An Effective Heart Disease Detection and Severity Level Classification Model Using Machine Learning and Hyper parameter Optimization Methods. IEEE Access. 10. 1-1. 10.1109/ACCESS.2022.3191669.

[27] Siamak, Aram & Sadeghian, Roozbeh & Abdellatif, Iheb & Nwoji, Stanley. (2019). Diagnosing Heart Disease Types from Chest X-Rays Using a Deep Learning Approach. 910-913. 10.1109/CSCI49370.2019.00173.

[28] Patro, Sibo & Padhy, Dr. Neelamadhab & Sah, Rahul. (2022). An Ensemble Approach for Prediction of Cardiovascular Disease Using Meta Classifier Boosting Algorithms. International Journal of Data Warehousing and Mining. 18. 1-29. 10.4018/IJDWM.316145.