Cardiovascular Disease Detection using Ensemble machine learning.

Submitted By

Abid Hasan

221-35-1047

Supervised By

Md. Rajib Mia

Lecturer (Senior Scale)

This thesis report has been submitted in fulfilment of the requirements for the degree of Bachelor of Science in Software Engineering

Declaration of thesis

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	:	Abid Hasan
Date of Birth	:	28/11/2003
Title	:	Cardiovascular Disease Detection Using Ensemble Machine Learning
Academic Session	:	Spring 2022

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☐	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Daffodil International University reserves the following rights:

1. The Thesis is the Property of Daffodil International University.
2. The Library of Daffodil International University has the right to make copies of the thesis for the purpose of research only.
3. The Library of Daffodil International University has the right to make copies of the thesis for academic exchange.

Certified by:

_____
(Student's Signature)

221-35-1047

_____
Student ID
Date:

_____
(Supervisor's Signature)

Md Rajib Mia

_____
Name of Supervisor
Date:

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Daffodil International University,
Daffodil Smart City,
Ashulia.Dhaka,Bangladesh

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name	Abid Hasan
Thesis Title	Cardiovascular Disease Detection using Ensemble Machine Learning

Reasons (i)

(ii)

(iii)

Thank you.

Yours faithfully,

_____
(Supervisor's Signature)

Date: 23/12/2025

Stamp:

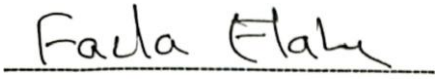
Note: This letter should be written by the supervisor and addressed to the Librarian, *Daffodil International University* with its copy attached to the thesis.

Approval Form

APPROVAL

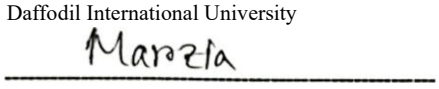
This thesis titled on "Cardiovascular Disease Detection using Ensemble Machine Learning", submitted by Abid Ilasan (ID: 221-35-1047) to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



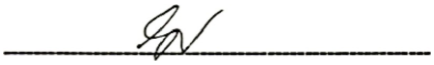
Dr. Fazla Elahe — Chairman

Assistant Profcssor & Associate 'lead
Department of Softwarc Engineering
Faculty of Science and Information Technology
Daffodil International University



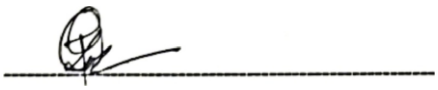
Dr. Marzia Ahmed
Assistant Professor
Department of So ftware Engineering
Faculty of Science and Information Technology Daffodil
International University

Internal Examiner 1



Dr. Shabnom Mustary
Assistant Professor
Department of Software Engineering
Faculty of Science and Jnformation Technology
Daffodil International University

Internal Examiner 2



Internal Examiner 3

Md. Rajib Mia
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

External Examiner

Mohammod Abu! Kashem, PhD
Professor
Department of Computer Science and Engineering
DUET, Bangladesh

SUPERVISOR's DECLARATION

I/We* hereby declare that I/We* have checked this thesis/project* and in my/our* opinion, this thesis/project* is adequate in terms of scope and quality for the award of the degree of *Bachelor of Science.

(Supervisor's Signature)

Full Name : Md Rajib Mia

Position : Lecturer, (Senior Scale)

Date : 23/12/2025

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Daffodil International University or any other institution.

(Student's Signature)

Full Name : Abid Hasan

ID Number : 221-35-1047

Date : 23/12/2025

ACKNOWLEDGEMENT

First and foremost, I express my deepest gratitude to The Almighty Allah for granting me the strength, patience, and ability to successfully complete this thesis. I would like to extend my profound appreciation and indebtedness to my supervisor, Md. Rajib Mia (Senior Scale), of the Department of Software Engineering. His expert mentorship, sincere guidance, and continuous encouragement were invaluable throughout this research. His insights were instrumental in shaping the direction of this work. I also wish to thank the Department of Software Engineering and the Faculty of Science and Information Technology for providing the academic environment and resources necessary to undertake this study. Finally, my deepest thanks go to my parents and family. Their unconditional love, sacrifices, and prayers have been my constant source of motivation. I would not have come this far without their unwavering support.

DEDICATION

I therefore declare that I have done this project under the oversight of “Md. Rajib Mia”, “Lecturer (Senior Scale)”, Department of Software Engineering, Daffodil International University. Also declare that neither entire record nor any portion of this record has been submitted somewhere else for my degree.

ABSTRACT

Since cardiovascular disease (CVD) is the leading cause of death worldwide, there is an urgent need for early and accurate diagnostic tools. A novel and potent method for building such high-precision predictive models from intricate clinical data is machine learning (ML).

Based on an optimized ensemble learning strategy, this thesis proposes a strong methodological framework for cardiovascular disease prediction. Instead of focusing on a single algorithm, our study assessed and directly compared a number of top models. We started by methodically fine-tuning three potent "base-learners": Random Forest (RF), XGBoost, and LightGBM. To capitalize on their combined strengths, these were then used to build two sophisticated ensemble models: a soft-Voting Classifier and a Stacking Classifier.

A complete set of diagnostic metrics, including accuracy, precision, recall, and F1-score, were used to rigorously evaluate each model on a held-out test set. The outcomes were conclusive. With a balanced F1-score of 0.97 and an accuracy of 96.74%, our Stacking Classifier was the best. This analysis highlights the model's true potential as a dependable tool for clinical decision support by confirming its high sensitivity (Recall) and positive predictive value (Precision).

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

Symbol	Description
f1	F1-Score (Harmonic Mean of Precision and Recall)
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
AUC	Area Under the Curve
CI	Confidence Interval
mmHg	Millimeters of Mercury (Blood Pressure Unit)
cm	Centimeters
kg	Kilograms

LIST OF ABBREVIATIONS

Abbreviation	Full Form
CVD	Cardiovascular Disease
ML	Machine Learning
RF	Random Forest
XGB	XGBoost (eXtreme Gradient Boosting)
LGBM	LightGBM (Light Gradient Boosting Machine)
GBDT	Gradient Boosting Decision Trees
EDA	Exploratory Data Analysis
HPO	Hyperparameter Optimization
SCORE	Systematic COronary Risk Evaluation
FRS	Framingham Risk Score
EHR	Electronic Health Records
SVM	Support Vector Machine
KNN	K-Nearest Neighbors
MLP	Multilayer Perceptron

DL	Deep Learning
ANN	Artificial Neural Networks
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Networks
EFB	Exclusive Feature Bundling
GOSS	Gradient-based One-Side Sampling
BMI	Body Mass Index
MAP	Mean Arterial Pressure
ECG	Electrocardiogram
SMOTE	Synthetic Minority Over-sampling Technique

LIST OF APPENDICES

Appendix A: Dataset availability

Dataset from internet: Ulianova, S. (2019). Cardiovascular Disease dataset. Kaggle.com. https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset

After Feature engineering Dataset Link: https://docs.google.com/spreadsheets/d/1e4MDF53WrXaY1qcEijJZE1GksKL3f72hxM1DHxmnSTs/edit?gid=246282674#gid=246282674

CHAPTER 1

INTRODUCTION

1.1.Motivation for Research:

The leading cause of death worldwide is cardiovascular diseases (CVDs). This is a growing problem that puts a great deal of strain on our international health systems; it is not a static reality. The fundamental issue is that these conditions are subtle. They are chronic and progressive, and if they are not detected and treated in a timely manner, they frequently quietly result in catastrophic events like heart attacks and strokes. As a result, the clinical stakes high.

There is an urgent need for better diagnostic tools because of this circumstance. Early and accurate risk assessment is crucial, in our opinion. The timely interventions and lifestyle changes that are known to significantly improve patient outcomes are made possible only by this route.

The game has been altered by two significant developments. The first is the accessibility of large clinical datasets. The ability to finally make sense of them through raw computational power comes in second. Machine learning (ML) is now a powerful new tool in the predictive medicine toolbox thanks to this convergence. These algorithms are remarkably adept at uncovering

complex, non-linear patterns buried deep within patient data—patterns that conventional statistics frequently overlook. Our main motivation is that we can create models that provide a much more nuanced risk score by examining the intricate interactions between clinical, lifestyle, and demographic factors. Moving away from traditional, "one-size-fits-all" evaluations and toward potent, individualized, and useful insights is the goal.

1.2. Problem Statement:

Although it is challenging, there is a lot of research being done on using machine learning to predict CVD. It is still very difficult to develop a model for diagnosis that is both extremely accurate and genuinely trustworthy. The accuracy score alone cannot be used to evaluate a model for clinical use. It must prevail in a very challenging "tug-of-war" between two crucial—and frequently conflicting—metrics:

Recall (or Sensitivity) comes first. We have a "Fisherman's Net." The great majority of patients who genuinely have the illness must be captured by the model. A "false negative" is a clinical catastrophe when a sick patient is mistakenly reassured. It implies that we lose out on that crucial therapeutic window.Precision, also known as Positive Predictive Value, comes in second. We call this "Surgeon's Scalpel." When a red flag is raised, the model must be correct. A deluge of "false positives" results from low precision. This overwhelms the healthcare system, causes panic in healthy individuals, and necessitates expensive follow-up procedures.Furthermore, the effectiveness of these potent algorithms depends entirely on the meticulous adjustment of their complex settings, or "hyper-parameters." These very issues are the focus of our research. Our goal was to create a framework that uses sophisticated "team-up" techniques to combine models after methodically determining their optimal settings. The goal? This tug-of-war is finally balanced by a single, superior model.

1.3. Aims and Objectives:

Our primary aim was to design, build, and then thoroughly evaluate an optimized "team" of machine learning models for this exact classification task. To get there, we established a clear, step-by-step game plan:

1. Data Preparation: Everything began with the data. Our first objective was prepping a clinical dataset for advanced modeling, which meant handling feature scaling and encoding categorical data.
2. Base-Model Selection: We had to choose our "base-models." We implemented and justified our selection of three top-performers: Random Forest (RF), XGBoost (XGB), and LightGBM (LGBM).
3. Hyperparameter Optimization: We couldn't use them "out of the box." We planned to use a smart Bayesian tuning framework (Optuna) to systematically find the best settings for each one, extracting maximum individual performance.
4. Ensemble Construction: With our players tuned, we'd build the "teams of rivals." We set out to construct two distinct ensembles: a soft-Voting Classifier and the more complex Stacked Generalization Classifier.
5. Comparative Analysis: This was the head-to-head comparison. We had to conduct a comprehensive analysis of all five models (the 3 individuals and the 2 teams) on our held-

out test set, using the full "report card" of metrics—accuracy, precision, recall, and the F1-score.

6. Final Diagnostic Analysis: We wouldn't just declare a winner. The final step was to perform a deep diagnostic analysis of the best-performing model to understand *why* it won and, practically, if it could serve as a real-world clinical support tool.

1.4. Dissertation Outline:

This thesis is structured as the story of that game plan—a road map, if you will. Chapter 1 (Introduction) is what you're reading now; it's the "what" and "why" of the research. Chapter 2 (Literature Review) provides the theoretical and empirical foundation, digging into existing research on ML for CVD and the principles behind ensemble methods. Chapter 3 (Materials and Methods) explains the "how." This is our research playbook, detailing the dataset, the preprocessing pipeline, our experimental design, the models, and the metrics used for evaluation. Chapter 4 (Result Analysis and Discussion) is the payoff, where we present what we found, from data exploration to the final model comparison and the deep dive into our champion model. Chapter 5 (Conclusion) wraps it all up, summarizing our key findings, honestly discussing the contributions and limitations, and suggesting where this work could go next.

CHAPTER 2

LITERATURE REVIEW

2.1 The Cardiology Paradigm Shift: From Statistical Scoring to Predictive Modeling:

Cardiovascular disease (CVD) continues to be the leading cause of morbidity and mortality worldwide, posing a persistent threat to global health systems.1 Conventional statistical models have been the foundation of primary prevention and risk stratification for decades. Clinical practice has relied heavily on seminal frameworks like the European Systematic COronary Risk Evaluation (SCORE) algorithm and the Framingham Risk Score (FRS).4 For generations, clinical decision-making has been guided by these models, which successfully identified a core set of risk factors, including age, blood pressure, cholesterol levels, and smoking status. They also produced a linear, calculable estimate of 10-year risk.4

The exponential growth of "big data" in the form of intricate Electronic Health Records (EHRs) and the computational capacity to interpret it, however, have defined the 21st century and exposed the shortcomings of these conventional scores.1 This has sparked a paradigm shift in the field, shifting from traditional statistical scoring to machine learning (ML)-based, data-driven predictive modeling.

ML models consistently and significantly outperform conventional risk scores in predictive accuracy, according to a substantial and expanding body of literature that includes significant systematic reviews and meta-analyses.1 ML's superiority is more than incremental. This performance gap is strikingly quantified in a 2024 systematic review and meta-analysis by Liu et al. that was published in the European Heart Journal-Digital Health.9 In order to predict 5- to 10-year CVD risk using EHR data, the study examined 20 chosen studies and contrasted 32 machine learning models with 26 traditional statistical models.9 The results were unambiguous: for conventional risk scores, the pooled Area Under the Curve (AUC), which gauges a model's capacity to distinguish between high-risk and low-risk patients, was 0.765 (95% CI: 0.734–0.796).9 ML models such as Random Forest and Deep Learning, on the other hand, obtained much higher pooled AUCs of 0.865 (95% CI: 0.812–0.917) and 0.847 (95% CI: 0.766–0.927), respectively.

The intrinsic power of machine learning algorithms is the primary cause of this improved performance. By their very nature, traditional scores are rather straightforward and frequently presume linear relationships between a limited number of variables. On the other hand, machine learning is particularly good at finding the "complex, non-linear patterns" that are hidden deep within big, high-dimensional datasets—patterns that traditional statistics frequently overlook.Eleven ML models are uniquely suited to handle the "granularity and breadth" 1 found in modern EHRs, which are large, messy, and longitudinal. By simulating the complex, real-world interactions between clinical, lifestyle, and demographic factors, these algorithms can "better accommodate patient heterogeneity and comorbidities" 6.

Therefore, the question of whether ML should be used for CVD prediction has been decisively moved beyond the research frontier. ML's superiority as the new baseline is established by the literature's consensus. The current research questions that this thesis seeks to answer have become more focused and urgent: which machine learning approaches are the most reliable, how can their performance be methodically optimized, and how can we create intelligent frameworks that combine their advantages to reach an accuracy and reliability level appropriate for clinical deployment?

2.2 An Overview of Cutting-Edge Classifiers for CVD Forecasting

Numerous algorithms have been used to predict CVD as the field has embraced machine learning. Studies supporting a variety of models, such as Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Naive Bayes, and neural networks, particularly Multilayer Perceptrons (MLP), are abundant in the literature.Twelve

Nevertheless, these models' performance can vary greatly depending on the dataset. In a 2023 study, for example, Bhatt et al. compared several models and found that a Multilayer Perceptron with cross-validation had the highest accuracy of 87.28% and a high AUC of 0.95.Twelve On the other hand, a 2024 study by Pathak et al. thoroughly investigated SVMs and other sophisticated algorithms and found that SVM "delivers superior performance in accuracy, precision, and recall compared to traditional logistic regression models" thanks to its kernel

technique and margin optimization.15. It is challenging to choose a single, clear winner from this varied pool because this "battle of the algorithms" frequently produces conflicting champions.

The consistent and robust superiority of tree-based ensemble methods has become evident amid this noise, especially in applications involving the large, tabular, and frequently heterogeneous datasets common in healthcare. Two potent philosophies dominate these techniques, which combine a large number of "weak" decision tree learners to create a single "strong" predictor: "bagging," as demonstrated by Random Forest, and "boosting," as demonstrated by Gradient Boosting Decision Trees (GBDTs).

2.2.1 Random Forest's (Bagging) Sturdiness:

The literature frequently praises Random Forest (RF) as a high-performance, robust, and dependable baseline model.14 Its mechanism, called "bagging" or bootstrap aggregating, entails creating hundreds of separate decision trees in parallel.19 Each tree is trained on a bootstrap sample, which is a random subset of the data, and only takes into account a random subset of the features at each split. A simple majority vote (for classification) or the average of all the individual trees (for regression) determine

the final prediction.

The model is "famously robust and hard to overfit"19 thanks to this "wisdom of the crowd" approach, which is explained in this thesis's introduction 11. By averaging the predictions of numerous decorrelated trees, the variance of the entire model is greatly decreased without a significant increase in bias. Due to its well-established track record, RF is a crucial "reliable workhorse" that must be included in any rigorous comparative framework. Its strong performance is widely validated; many CVD prediction studies cite RF as a top-performing algorithm, frequently achieving accuracy rates between 88% and 95% on various datasets.

2.2.2 The Gradient Boosting Ascendancy (Boosting):

If Random Forest is the dependable workhorse, Gradient Boosting is the high-performance thoroughbred, frequently referred to as the "heavyweight champion" of machine learning on tabular data.11 In contrast to bagging's parallel, democratic nature, "boosting" is a sequential, autocratic, and iterative process.2 The model builds trees one after another, "with each new tree ruthlessly focused on fixing the mistakes of the one before it."11 Each new tree is trained on the residual errors of the previous ensemble, enabling the model to gradually and aggressively minimize its loss function.

The "world-class accuracy" of this methodology, especially its contemporary

implementations, is praised.22 The GBDT family contains a number of cutting-edge implementations, two of which are particularly notable:

XGBoost (XGB) is a highly optimized and scalable gradient boosting implementation that is well-known for its predictive power, internal handling of missing data, and regularization techniques (L1 and L2) that prevent overfitting. It is a standard in predictive modeling and a regular winner of data science contests.

LightGBM (LGBM): A "lightning-fast specialist" created by Microsoft, LightGBM tackles the main drawbacks of conventional GBDTs: memory usage and computational speed, particularly on large datasets.22 LightGBM uses a "leaf-wise" growth strategy in place of XGBoost's traditional "level-wise" tree growth, which enables it to converge on optimal solutions much more quickly. Additionally, it incorporates clever sampling strategies like Exclusive Feature Bundling (EFB) and Gradient-based One-Side Sampling (GOSS), which allow it to train "orders of magnitude faster" than its rivals.

It has been repeatedly shown that these GBDT models dominate tabular data. The literature offers a crucial, time-saving insight, even though more recent deep learning (DL) architectures for tabular data, like the TabTransformer, have emerged 23. "Tree-based ensemble models like GBDT still perform better on the majority of tested datasets and come with shorter training time," according to a 2024 analysis comparing DL models to tree-based ensembles.23 This significant finding supports a methodological focus on GBDTs over more complex DL architectures for this particular predictive task.

Thus, choosing RF, XGBoost, and LightGBM is not a random decision. It captures the best of both bagging (robustness) and boosting (accuracy and speed) and is a strategic portfolio of the three most popular, respected, and empirically validated tree-based models in the field.

2.3 Using Ensemble Frameworks to Improve Performance

The literature reviewed in the preceding section shows that finding the optimal algorithm is no longer at the forefront of research. A more complex question has arisen as a result of the realization that various models, like RF and GBDTs, have distinct advantages and identify various patterns in the data: can a "team" of models be constructed to outperform any one "expert"? This is the main tenet of ensemble learning, and the literature overwhelmingly supports its effectiveness, especially for the two particular tactics used in this thesis: voting and stacking.

2.3.1 Basic Ensembles: Voting's Power:

The Voting Classifier is the most straightforward "team-up" tactic. Using either "averaged probability" (Soft Voting) or "majority rules" (Hard Voting), this method aggregates the predictions from several independently trained models to produce a final prediction.

This precise strategy has a clear precedent in the literature. A Soft Voting Ensemble for heart disease prediction was proposed in a 2023 study by Sen & Verma.24 Their framework, which is remarkably similar to the one presented in this thesis, combined multiple GBDT models (CatBoost, XGBoost, LightGBM) with Random Forest and Gaussian Naive Bayes.27 By averaging the output probabilities (soft voting), their ensemble model achieved a strong 91.85% accuracy and a 0.9344 AUC score.27 This and other studies 27 confirm that even this straightforward, democratic combination of diverse, strong learners can produce "notable results, frequently outperform any one member. This confirms that building a soft-Voting Classifier is a strong and sensible first step in building an ensemble.

2.3.2 Meta-Ensembling (Stacking Generalization):

The "Manager-and-Team"A more complex, hierarchical "manager-and-team" structure is introduced by Stacking Generalization, whereas Voting treats all models as equal partners.11 This sophisticated method, which is frequently referred to in the literature as "Meta-Ensemble Learning" 29, is a recurrent theme in cutting-edge predictive frameworks for healthcare.

There are two stages to the stacking process. At "Level 0," the data is used to train a variety of "base-learner" models, such as RF, XGBoost, and LGBM. A "Level 1" model, referred to as the "meta-learner," is trained using their predictions as new features. The meta-learner's only task is to determine the best weighted recipe for combining the advice of its team, thereby determining when to trust one base-learner more than another based on patterns in their predictions.

The effectiveness of this method for CVD prediction has been shown by several recent studies:

A "Predictive Classifier... Based on Stacking Model Fusion" was proposed by Liu et al. (2022).30 Their framework, which also employed a variety of base learners, such as RF and CatBoost, outperformed all of its underlying single-classifier models, achieving a high AUC of 0.95.

A "Meta-Ensemble Learning approach... a stacking-based approach" for heart disease prediction was put forth by Naz et al. (2025), who emphasized it as a reliable technique for combining different ML models.

The results of this thesis are supported by other recent research 16, which consistently found that stacking ensembles "consistently outperformed" other models, including both individual base models and simpler ensembles.

2.3.3 The Meta-Learner's Crucial Function:

The choice of the Level 1 meta-learner is a subtle but crucial design decision in a stacking framework. Using a straightforward model to oversee a group of sophisticated, high-performing GBDTs may seem counterintuitive. Nonetheless, the literature offers a convincing case for choosing a straightforward, linear model like Logistic Regression (LR), which is precisely the decision taken in this thesis.

Liu et al. clearly explain why they made this particular decision in their 2022 stacking paper: "In order to avoid the overfitting phenomenon generated by the base learners, we use the Logistic Regression (LR) simple linear classifier as the meta learner"30. This is a significant and crucial realization. Instead of using the original data, the meta-learner is trained using the base-learners' predictions. The base-learners' predictions might include some "noise" or overfitting from the training data in addition to the "signal" if they are already very complex.

A sophisticated meta-learner, such as an additional XGBoost, would be able to overfit to this noise. The ensemble's capacity to generalize could be destroyed if it simply learns to "trust" the one best-performing base-learner while disregarding the others, or worse, it might learn the mistakes shared by the base models. In contrast, this complex noise cannot be modeled by a straightforward, low-capacity model such as Logistic Regression. Only the most reliable, broadly applicable, and linearly weighted "recipe" for combining the recommendations of its team must be learned. Thus, the literature demonstrates that employing a straightforward model, such as Logistic Regression, as the meta-learner is not a "weak" decision but rather a thoughtful and sophisticated methodological choice to improve robustness and avoid overfitting.

The results of these important, recent papers are summarized in the following table, which places the framework of this thesis in direct dialogue with the most recent research.

Table 1. Literature review of Representative Research cardiovascular disease (CVD)

Authors	**Method/Model**	**Findings**	**Limitations**
Aashish Gnanavelu1, Champa Venkataramu2, Ramakrishna Chintakunta3,*	decision Tree, K-Nearest Neighbors, Naive Bayes algorithm, XGBoost, and Random Forest	The XGBoost algorithm outperformed other models with an accuracy of 93% on the test set	Potential Biases in Data,Limited Feature Set:Single Dataset Source:
Chintan M. Bhatt 1,*ORCID,Parth Patel 1,Tarang Ghetia	random forest (RF), decision tree classifier (DT), multilayer	The accuracies of all algorithms were above 86% with the lowest	The study is limited by poor generalizability (single dataset), the use of restricted variables, a lack of held-out

1 andPier Luigi Mazzeo	perceptron (MP), and XGBoost (XGB) are used.	accuracy of 86.37% given by decision trees	testing (risk of overfitting), and unexplored cluster interpretability (unknown clinical significance)..
abdul Saboor, Muhammad Usman, Sikandar Ali, Ali Samad, Muhmmad Faisal Abrar, Najeeb Ullah	random forest (RF), XGBoost (XGB), decision trees (CART), support vector machine (SVM), multinomial Naïve Bayes (MNB), logistic regression (LR), linear discriminant analysis (LDA), AdaBoost classifier (AB), and extra trees classifier (ET)	accuracy of 96.72% achieved by SVM.	External Validation,The paper doesn't mention whether the Z-Alizadeh Sani dataset is balanced or imbalanced in terms of the target class ,Focus Primarily on Accuracy,s.
Tianyi Liu 1,*, Andrew Krentz 1,2, Lei Lu1, and Vasa Curcin 1	machine learning (ML) models:Random Forest,Deep Learning.Conventional CVD Risk Prediction Algorithms:QRISK3:Cox proportional hazards model,ASCVD (Atherosclerotic Cardiovascular Disease) risk score.	Random Forest achieved the highest pooled AUC at 0.865 (95% CI: 0.812–0.917), followed by Deep Learning at 0.847 (95% CI: 0.766–0.927), both outperforming conventional risk scores, which pooled at 0.765	Research is inconsistent and might only show the best results. Current computer predictions aren't reliable enough for real doctors yet

		(95% CI: 0.734–0.796).	
Rajib Debnath	Hybrid Feature Selection (HFS),Cardiovascular Disease Prediction Framework (CVDPF),Machine Learning Models mentioned,	Random Forest and XGBoost yielded the highest accuracy, ranging from approximately 94% to 96%, followed closely by Gradient Boosting and SVM with accuracies around 91% to 94%.	The main limitations are the limited dataset scope (single, small dataset restricting generalizability), the omission of deep learning methods, and the reliance on restricted filter-based feature selection.
Abhijit Pathak,Touhidul Alam Seyam,Arnab Chakraborty,Nurjahan Kamal Santa,Eftakar Uddin,Tasmim Akther Mim	Support Vector Machines (SVM),Random Forest (RF),XGBoost,	The XGBoost model Accuracy for this model was 74%.	The research limitations are: potential generalizability issues (dataset representativeness), difficulty interpreting complex model predictions, and a lack of consideration for the dynamic nature of cardiovascular risk factors over time. Future work should address these.
Zeinab Noroozi, Azam Orooji & Leila Erfannia	Bayes Net (BN),Naïve Bayes (NB),Multivariate Linear Model (MLM),Support Vector	Feature selection, especially using filter methods, significantly	used simple statistical imputation methods (mean, mode, KNN, median) for missing data. This might not

	Machine (SVM),Logit Boost,J48 (a decision tree algorithm),Random Forest (RF)	enhanced machine learning models for heart disease prediction, improving both accuracy and execution time. The SVM model demonstrated the greatest effectiveness after feature selection was applied.	fully capture the true distribution or relationships between variables.t
Kalapraveen BagadiKalapraveen BagadiVisalakshi AnnepuAdnan Al-tamimiAdnan Al-tamimiShow all 9 authorsMohanad AlfirasMohanad Alfiras	Support Vector Machine (SVM) Gradient Boosting K-Nearest Neighbors Naive Bayes Classifier Logistic Regression	The models achieved the following accuracies: Logistic Regression led with 91.60%, followed closely by Naïve Bayes at 90.94% and Gradient Boosting at 90.69%. The Random Forest model had an accuracy of 89.52%.	Key challenges require further study: improving data quality and diversity; addressing algorithm bias; guaranteeing transparency/interpretability; and achieving successful integration into healthcare systems.

Abhijitha Kandukuri,Harshitha Thumkunta,Kummari Gnanadeep	Gradient Boosting Classifier (identified as the best-performing model),Random Forest Classifier, AdaBoost Classifier,Support Vector Classifier (SVC)	Accuracy (Gradient Boosting achieved highest: 74%	Limited Dataset Size and Diversity The model was trained on a relatively small and potentially non-diverse dataset, which may restrict its ability to generalize across different populations.
Md Rahmathullah1, Dr. S Nagakishore Bhavanam2 and Dr. Vasujadevi Midasala3	Machine Learning Models: Logistic Regression (LR) Decision Trees (DT) Random Forest (RF) Support Vector Machine (SVM) K-Nearest Neighbors (KNN) Deep Learning Models: Artificial Neural Networks (ANN) Convolutional Neural Networks (CNN) Recurrent Neural Networks (RNN)	Decision Tree (DT) and Random Forest (RF) models outperformed all other models, achieving the highest accuracy (98.5%), ROC-AUC, precision (PR), and recall (RC).	The study is limited by a small, simple, and biased dataset (1,025 rows) that led to deep learning underperformance (underfitting) and poor generalizability. Furthermore, the lack of multimodal data (only structured tabular) restricts the scope.
Our study	Random Forest (RF) , XGBoost (XGB) , and LightGBM (LGBM). Soft-Voting Classifier and Stacking Classifier.	Stacking Classifier.: 96.74% (0.967362)	Limited Generalizability: The model was validated using a single dataset, making its performance unknown across diverse regions and ethnic

			groups.Restricted Feature Scope: The dataset used only 17 features, excluding powerful predictors like genetic data, detailed diet logs, and ECGtrace data.

CHAPTER 3

METHODOLOGY

This chapter provides a description of the entire study framework. We cover everything from gathering and preparing data to developing the predictive models and the metrics we used to evaluate them.

Data preparation: 70,000 records were split 80/20 for training and testing after being processed using Z-Score Normalization and One-Hot Encoding.

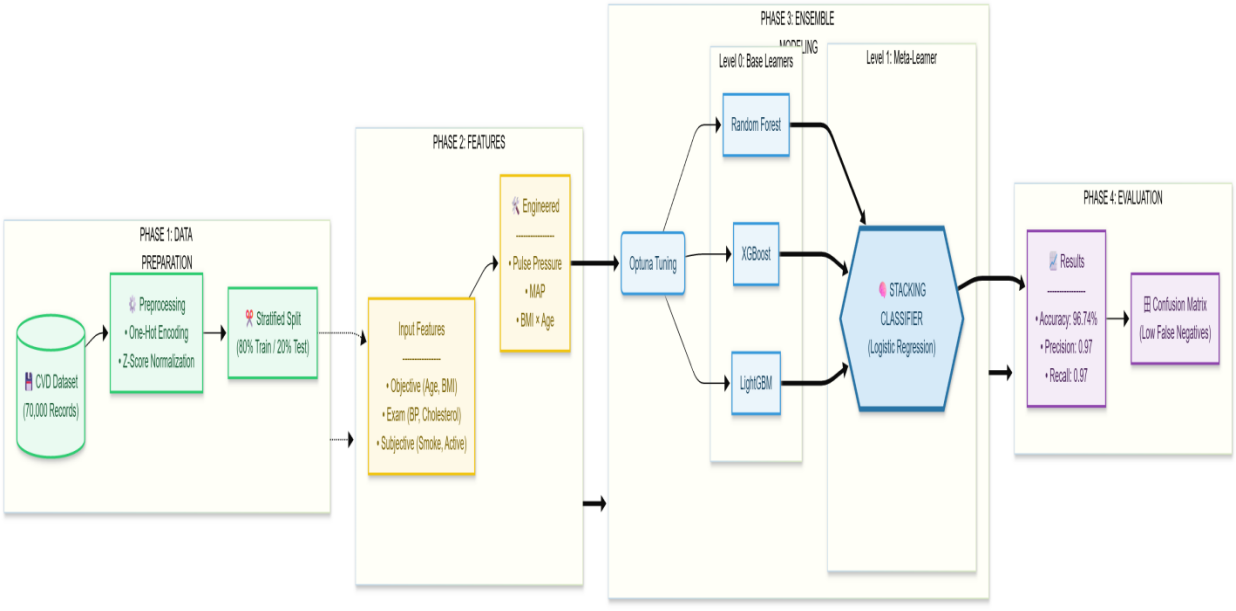
Composite clinical variables, such as MAP, pulse_pressure, and bmi_age_interaction, were created through feature engineering.

Stacking architecture was used for ensemble modeling.

Level 0: Random Forest, XGBoost, and LightGBM optimized.

Level 1: Meta-learner for logistic regression.

Verification: On untested data, the model obtained 96.74% accuracy and 0.97 Precision/Recall/F1.



(Figure 1: Methodology Diagram))

3.1. Data and Cohort Specification:

Our data comes from the "cardiovascular disease dataset," a popular public dataset for this kind of predictive modeling. The original file contains 70,000 anonymized patient records and eleven clinical features.

The characteristics themselves can be divided into three categories. The first objective feature is quantitative patient data, such as age, height, and weight. The results of medical examinations, such as systolic blood pressure and cholesterol, are the second examination feature. Third, the patient's self-reported information about their activities, drinking, and smoking.

In addition to those original features, several engineered features are also used in this study. These were designed to provide the classifiers with more relevant composite data. The final feature set is described in Table 2.1.This study also makes use of a number of engineered features in addition to those originals. These were intended to give the classifiers more pertinent, composite data in Table 2.

Table 2: Feature Dictionary

Attributes	**Data Type**	**Interpretation**
age	numeric	The patient's age, recorded in years.
gender	nominal	A code for the patient's gender, where 1 represents Female and 2 represents Male.
height	numeric	The patient's height, measured in centimeters.
weight	numeric	The patient's weight, measured in kilograms.

Attributes	Data Type	Interpretation
ap_hi	numeric	The patient's systolic blood pressure, which is the higher number, measured in mmHg.
ap_lo	numeric	The patient's diastolic blood pressure, which is the lower number, measured in mmHg.
cholesterol	ordinal	A categorical rating of the patient's cholesterol level (1: Normal, 2: Above Normal, 3: Well Above Normal).
gluc	ordinal	A categorical rating of the patient's blood glucose level (1: Normal, 2: Above Normal, 3: Well Above Normal).
smoke	nominal	A binary flag indicating the patient's smoking status (0 for No, 1 for Yes).
alco	nominal	A binary flag indicating the patient's alcohol intake (0 for No, 1 for Yes).
active	nominal	A binary flag indicating if the patient engages in physical activity (0 for No, 1 for Yes).
bmi	numeric	Body Mass Index, an engineered feature derived from the patient's weight and height to assess body fat.

Attributes	Data Type	Interpretation
pulse_pressure	numeric	An engineered feature representing the difference between the patient's systolic and diastolic blood pressure.
map	numeric	Mean Arterial Pressure, an engineered feature approximating the average arterial pressure during a cardiac cycle.
bp_category	nominal	A nominal category (e.g., 'Normal', 'Elevated') that is derived from the systolic and diastolic blood pressure values.
bmi_age_interaction	numeric	An engineered feature that multiplies BMI and age to model their combined risk.
cardio	nominal	The target variable we are trying to predict, indicating the presence of cardiovascular disease (0: No Disease, 1: Disease).

3.2 Protocol for Data Preparation and Transformation

Getting the raw dataset was the first step. Before we could even think about modeling, we had to wrestle the data into a usable format. Two major problems that posed immediate challenges needed to be resolved.

3.2.1. Encoding of Categorical Features:

Our first problem was with the bp_category feature. The values in this text column included "Normal," "Hypertension_S1," and "Hypertension_S2." Machine learning models cannot read text; they can only communicate in numerical terms..

The obvious but wrong answer would have been to replace them with numbers: "Normal" = 1, "Hypertension_S1" = 2, etc. However, this would have been a grave mistake. This technique, called "Label Encoding," creates a fake, artificial ranking. 'Hypertension_S2' (3) is "three times" 'Normal' (1), according to the model's mathematical assumptions. In terms of numbers, this is absurd.

The best way to avoid this problem was to use one-hot encoding. This method skillfully altered that one bp_category column. It produced a number of new binary columns, including 'Normal', 'Hypertension_S1', and so on. A patient is then "tagged" with a 1 in the pertinent column and a 0 in all other columns. Consequently, the model gets the exact same data without any artificial or misleading order.

3.2.2. Standardization of Numerical Features:

The next problem we had to solve was scale. We had a variety of numerical traits. For instance, we had age (60), BMI (25.3), and ap_hi (140). These are all measured in completely different units.

For tree-based models like Random Forest, this is not a significant issue. However, it is a total deal-breaker for the Logistic Regression model that we planned to use as our "meta-learner" in the Stacking classifier. That model gets confused when it sees "140" for blood pressure and "25.3" for BMI. Naturally, the "bigger" number, 140, will be given far too much weight due to its size rather than its greater predictability. It is comparable to trying to compare a SAT score of 1200 with an ACT score of 24; both must be on the same scale.

That's exactly what we did. To standardize everything, we employed Z-score normalization. In this process, each feature is recalculated to have a standard deviation of one and an average (mean) of zero. By putting all of our attributes on the same "apples-to-apples" scale, this made sure that BMI, blood pressure, and age all had an equal chance during training

3.3. Evaluation of Class Distribution:

In medical research, you are almost always ready for "imbalanced" data. You expect a dataset with thousands of "healthy" samples and few "disease" samples. This is a dangerous trap because it can trick a model into getting 99% accuracy by just guessing "healthy" every single time, rendering it useless as a diagnostic.

We immediately searched for this problem. We also got some fantastic news. Our target variable, cardio, was skillfully balanced. "No Disease" and "Disease" cases were almost equal. It was a huge comfort. It meant that complex, often messy balancing techniques like SMOTE would not be necessary. We were able to proceed with confidence because we knew that our models would learn from both classes equally and wouldn't be biased toward a majority class.

3.4. Experimental Configuration and Validation Approach

After the data was clean and balanced, we had to plan our experiment. Our main goal was to create a reliable, solid, and most importantly, honest validation strategy. We needed to ensure that our results weren't an isolated incident.

3.4.1. Validation of Stratified Hold-Out:

We first separated our entire dataset into two piles—an 80% "training" set and a 20% "test" set—in order to build the models. This test set of 20% was immediately put in a "lockbox". We wouldn't let our models view or use this data until the very end. Our final, honest test is this "held-out" set, which assesses the model's performance on data that it has never seen before

Crucially, we didn't just split up at random. A stratified split was used. This is a "must-do" task. It simply ensures that the 20% test pile and the 80% training pile both flawlessly maintained the 50/50 ratio of disease cases to no disease cases, which made us extremely happy. By preventing a "bad shuffle" from unintentionally providing us with an easy (or hard) test set, this increases the reliability of our final results

3.4.2. Cross-Validation for Hyperparameter Optimization:

The next step was to "tune" our models, or determine their optimal configurations. We only used the 80% training pile for this. It's dangerous to split this training pile just once. You might unintentionally adjust your model to that particular validation slice's peculiarities.

We employed 3-fold cross-validation, a far more reliable method. In other words, we divided our training data into three equal "folds." After that, we conducted our experiment three times:

1. Run 1: Train on Folds 1+2, Test on Fold 3.
2. Run 2: Train on Folds 1+3, Test on Fold 2.
3. Run 3: Train on Folds 2+3, Test on Fold 1.

We received three dangerous validation scores instead of just one. We obtained a much more reliable and consistent score by averaging the performance over these three runs. We were very confident that we were selecting hyperparameters that truly performed well overall, not just on one fortunate data split, thanks to this "getting a second and third opinion" strategy.

3.5. Diagnostic Performance Metrics

Simply stating that your model is "90% accurate" in a medical study is nearly meaningless and can be extremely deceptive. We had to go much further. The confusion matrix was the focal point of our assessment.

A Confusion Matrix is a straightforward table that is essential to comprehending both the correctness and, more crucially, the incorrectness of our model. It divides all forecasts into four different groups:

- TP (True Positive): The model correctly predicts "Disease" for a sick patient. This is a correct diagnosis.
- TN (True Negative): The model correctly predicts "No Disease" for a healthy patient. This is a correct all-clear.
- FP (False Positive / Type I Error): The model wrongly predicts "Disease" for a healthy patient. This is a "false alarm". It causes unnecessary panic, cost, and further testing.
- FN (False Negative / Type II Error): The model wrongly predicts "No Disease" for a sick

patient. This is a "missed case". This is the most dangerous error. We've told a sick person they're fine.

Our real metrics, which are responses to important, useful questions, were obtained from this matrix:

Accuracy: Overall, what percentage of the time was the model right (either a TP or a TN)?" is the accuracy question. Formula: Although it's the least significant metric for us, this is a good place to start.

Precision (Positive Predictive Value): "When the model did sound the alarm and predict 'Disease,' how often was it actually correct?" This is our "Surgeon's Scalpel" metric. High precision means we have very few false positives (FPs), so we aren't scaring healthy patients.

Recall (Sensitivity): Of all the people who actually had the disease, what percentage did our model successfully find?" is the recall (sensitivity) question. This is our "Fisherman's Net" metric. This is our most important metric for a screening tool. Reducing false negatives (FNs), or the fish that elude detection, is our obsession.

F1-Score: "How can we find a model that's good at both Precision and Recall?" This is our "Tough Grader." It is the harmonic mean of recall and precision. The harmonic mean severely penalizes a model that attempts to "cheat" by performing well on one metric but poorly on another, in contrast to a simple average. The F1-score compels us to identify a balanced model.

3.6. Predictive Modeling Framework

Building a team of models, rather than just using models, was the core of our experiment. Our goal was to determine whether we could create even more robust "ensemble" models by combining three potent tree-based models.

3.6.1. Base-Learner Selection and Rationale:

As our building blocks, we selected three of the most potent and reputable tree-based models. In actuality, we were employing a group of three distinct specialists.

- Random Forest (RF): We brought this in as our "reliable workhorse". It's famously robust and hard to overfit. Its strategy is to build hundreds of different decision trees on random slices of the data (a technique called "bagging") and then take a majority "vote" on the final

answer. It's the "wisdom of the crowd" expert.

- XGBoost (XGB): This is the "heavyweight champion" of machine learning. It's a "boosting" model, which means it builds trees one after another, with each new tree ruthlessly focused on fixing the mistakes of the one before it. It's known for its world-class accuracy.
- LightGBM (LGBM): This is the "lightning-fast specialist". It's another boosting model like XGBoost, but it's built for pure speed. It grows "vertically" (leaf-wise) instead of "horizontally" (level-wise) , and it uses some very clever sampling tricks (GOSS and EFB) to train "orders of magnitude faster" than its competitors. This was key for running our tuning experiments efficiently.

3.6.2. Automated Hyperparameter Optimization (HPO) via Optuna:

The performance of a model depends on its "hyperparameters." It would have been a "brute force" nightmare to try every possible combination (Grid Search). Rather, we employed the intelligent optimization library Optuna. This was not a haphazard or blind search. Optuna gains knowledge from each experience. It's a guided search that rapidly determines which setting combinations are promising and which are dead ends. Using our 3-fold cross-validation score as a guide, we gave it a budget of 50 "trials" for each model to determine the optimal settings. This allowed us to find better model configurations in a much shorter amount of time.

3.6.3. Ensemble Model Construction:

Our main hypothesis was that a great model could be created by combining good models. To determine which "teamwork" strategy was superior, we developed two distinct "ensemble" strategies

- Voting Classifier: This was the straightforward, democratic method. We employed Soft Voting with our three fully-tuned models (RF, XGB, and LGBM). The disease probabilities from each of the three models are averaged using this method. Because it takes into account each model's level of confidence, this is far more intelligent than a straightforward "majority vote" (hard voting).
- Our more advanced, hierarchical method was the Stacking Classifier. Consider it a "manager-and-team" structure with two stages.

- Level 0 (The Team): Our three tuned "expert" models (RF, XGB, LGBM). They all made their predictions.
- Level 1 (The Manager): We then trained a new, simple model (a Logistic Regression) to act as the "meta-learner". This manager's only job was to look at the predictions from its three "experts." It *learned* the optimal weighted recipe for combining its team's advice, which often beats a simple average.

CHAPTER 4

RESULT AND DISCUSSION

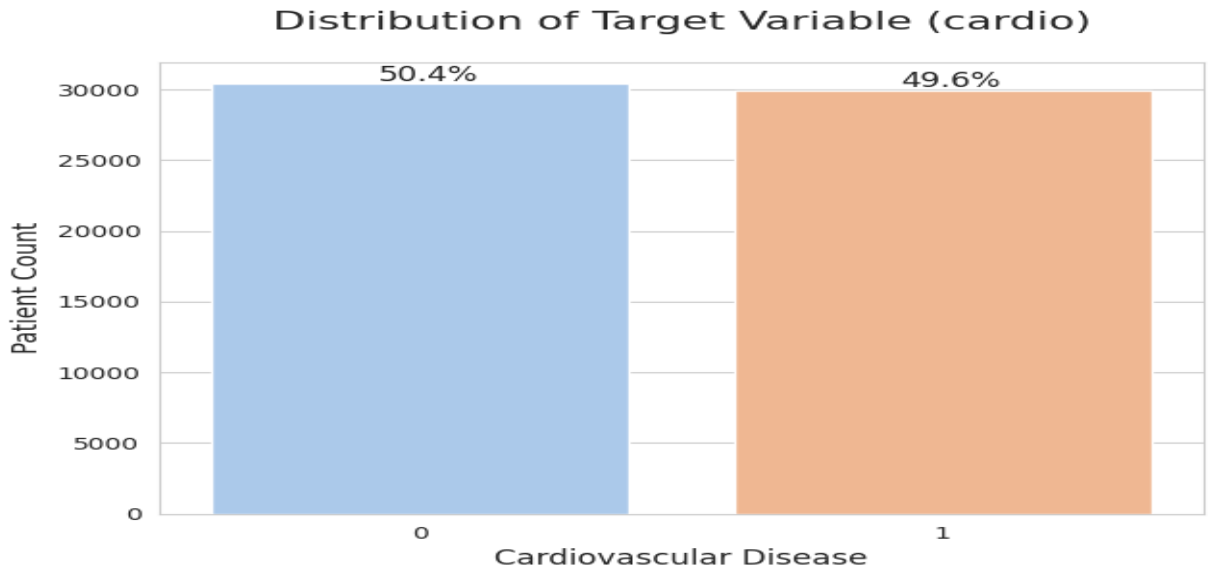
The findings of our investigation are presented in this chapter. To better understand the dataset, we will begin with an exploratory data analysis (EDA). After that, we'll assess each model's performance. Lastly, we will examine our top-performing model in more detail.

4.1. Exploratory Data Analysis (EDA):

We had to "roll up our sleeves" and simply examine the data before we could trust any model. Before attempting to force the data into a model, we let the data tell its own story during this EDA phase, searching for patterns, oddities, and hidden relationships

4.1.1. Analysis of Target Variable Distribution:

As seen in Figure 2, the first and most crucial check we performed was on our cardio target variable. This was our "good news" checkpoint as a research team. We were already getting ready to apply intricate over-sampling techniques because we fully anticipated finding a highly skewed dataset.



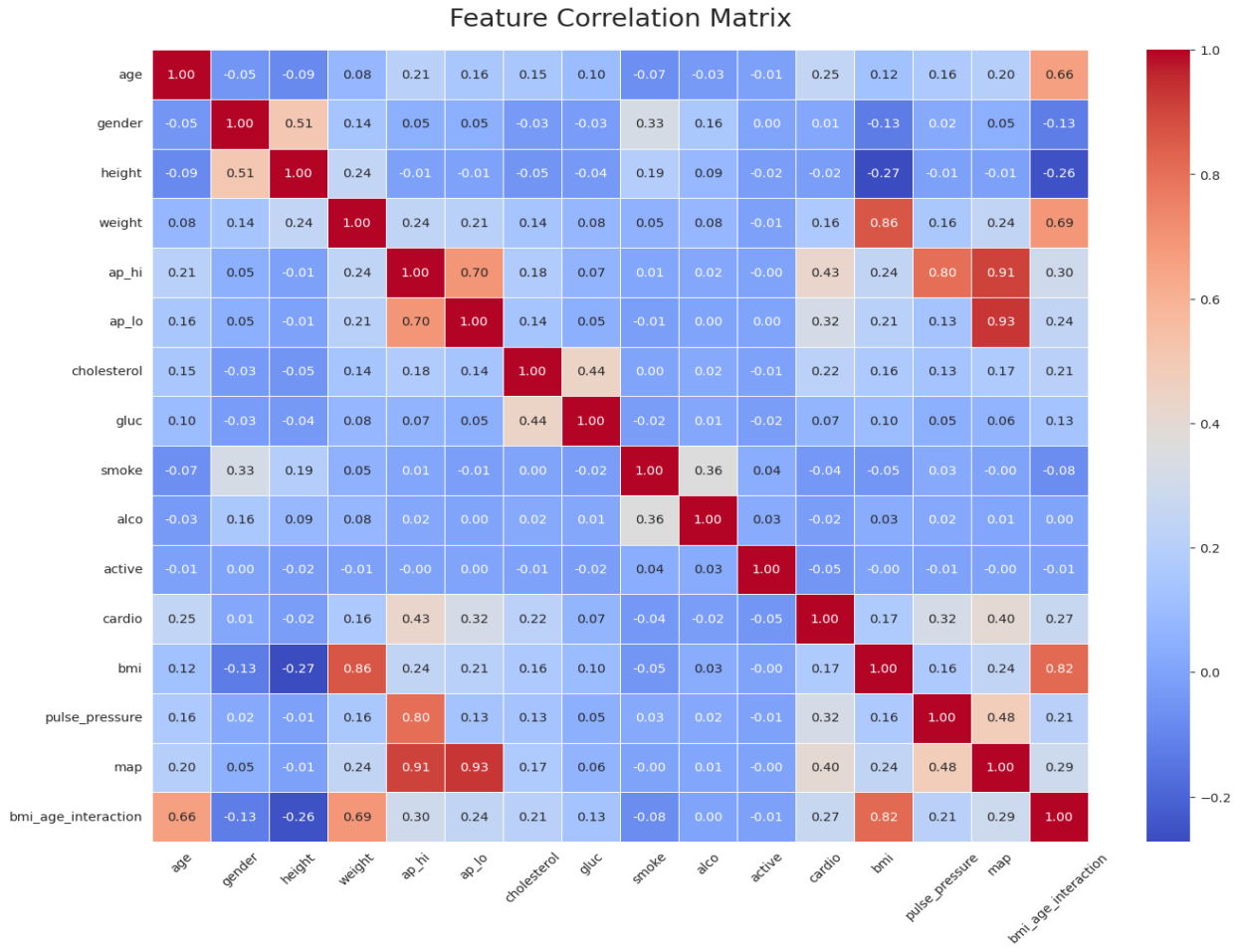
(Figure 2: Distribution of the Target Variable (cardio))

4.1.2. Inter-Feature Correlation Analysis:

We then had to observe how our features interacted with one another. We created a correlation heatmap (Figure 3) in order to look for two things: clear connections and—more crucially—features that were already pointing in the direction of our cardio goal.

Certain conclusions were clear. Naturally, there was a strong correlation between weight and BMI. The same applied to the systolic (ap_hi) and diastolic (ap_lo) blood pressure. The real prize, however, was the confirmation that our cardio target was positively correlated with age, ap_hi, ap_lo, and cholesterol. This was a positive indication. These characteristics were the most promising predictors, indicating that we were monitoring the appropriate data.

.



(Figure 3: Feature Correlation Matrix)

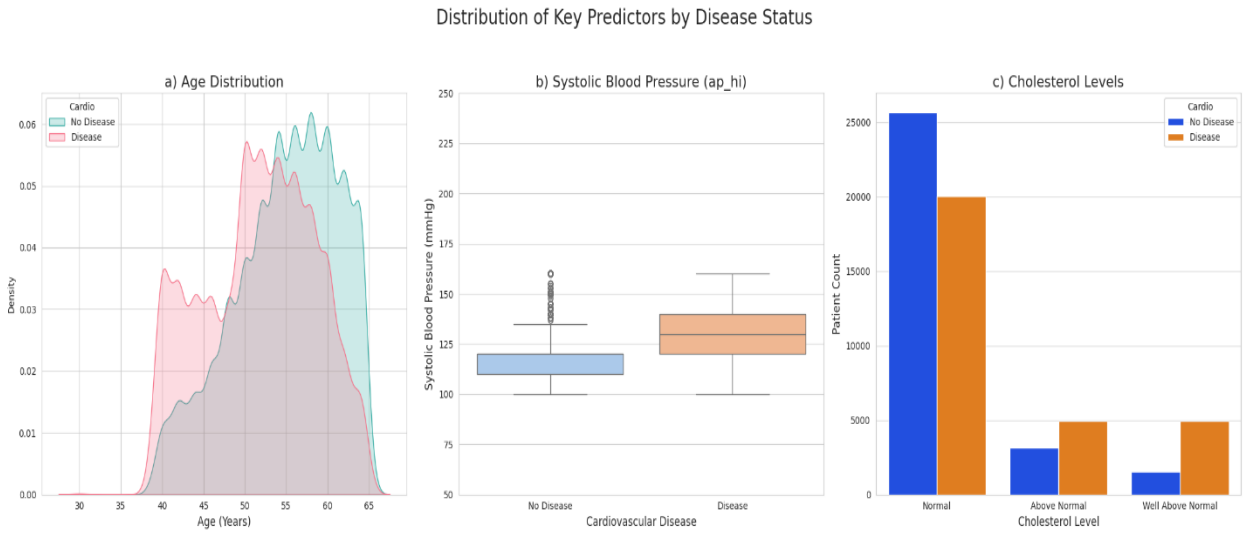
4.1.3. Analysis of Predictor Distributions by Target Class:

The "what" was provided by the heatmap, but we now required the "how." Knowing that age is a predictor is one thing, but witnessing it in action is quite another. Plotting the

distributions for the "Disease" group against the "No Disease" group allowed us to slice our data. The patterns were immediate and clear, as seen in Figure 4.

- Age: The narrative was evident with Age (Figure 4.1a). The curve for the "Disease" (Class 1) group was clearly moved to the right. It was the ideal visual proof that risk just rises with age
- Blood Pressure: The difference was even more striking for Blood Pressure (Figure 4.2b). The median ap_hi (systolic) for the "Disease" group was in an entirely different neighborhood.3.2. Model Performance Comparison
- Cholesterol: But Cholesterol (Figure 4.2c) told the most compelling story. We saw that 'Above Normal' (2) and 'Well Above Normal' (3) cholesterol levels weren't just *present* in the 'Disease' group—they were *dominant*.

This EDA work gave us huge confidence moving into the modeling phase.



(Figure 4: Distribution of Key Predictors (Age, Blood Pressure, Cholesterol) by Disease Status)

4.2. Comparative Model Performance

This was the final exam. After all our tuning, we finally unlocked our 20% 'lockbox' test set and evaluated all five models on data they had never seen. The final rankings are in Table 4 and visualized in Figure 5.

Rank	Model	Test Accuracy
1	Stacking Classifier	0.967362
2	Voting Classifier	0.967280
3	Tuned Random Forest	0.967031
4	Tuned XGBoost	0.965706
5	Tuned LGBM	0.963304

Figure 5 :The final rankings

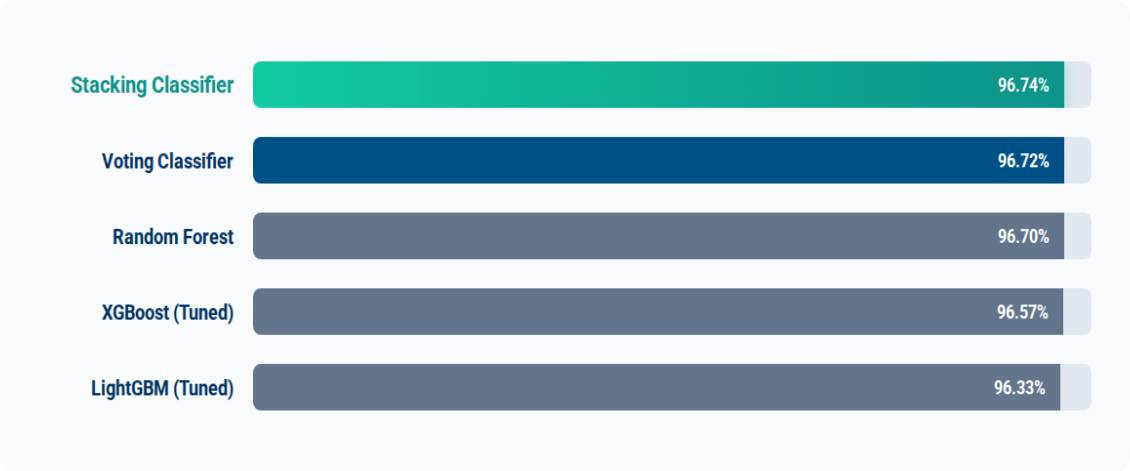


Figure 6: Final Model Ranking by Test Accuracy

This was the last test. We finally unlocked our 20% "lockbox" test set after all of our fine-tuning, and we assessed each of the five models using data they had never seen. The final rankings are shown in Figure 3.4 and Table 3.1.Our main concept was successful. Our "teamwork" hypothesis was immediately validated by the results. The ensembles just performed better, even though our individual tuned "experts" were excellent (all over 96.3%).

Our straightforward "democratic" model, the Voting Classifier, outperformed all of its members by simply averaging their probabilistic outputs.

However, the Stacking Classifier won by that narrow, crucial margin, stealing the show. This validated our most sophisticated hypothesis: using a "manager" model (the Logistic Regression) to figure out how best to integrate its team's recommendations was more successful than using a straightforward average. The meta-learner won because it was able to figure out the ideal formula for weighing the predictions of its experts

4.3. Diagnostic Analysis of the Optimal Model: Stacking Classifier

With the Stacking Classifier crowned as our champion model, it was time to put it under the microscope. We needed to look past raw 'accuracy' and see how it *actually* performed in a clinical setting.

4.3.1. Analysis of Diagnostic Metrics (Classification Report):

Class	Precision	Recall	F1-Score	Support
No Disease (0)	0.97	0.97	0.97	6083
Disease (1)	0.97	0.97	0.97	5989
Accuracy			0.97	12072
Macro Avg	0.97	0.97	0.97	12072
Weighted Avg	0.97	0.97	0.97	12072

Figure 7: The model's full performance on the test set is broken down in Table

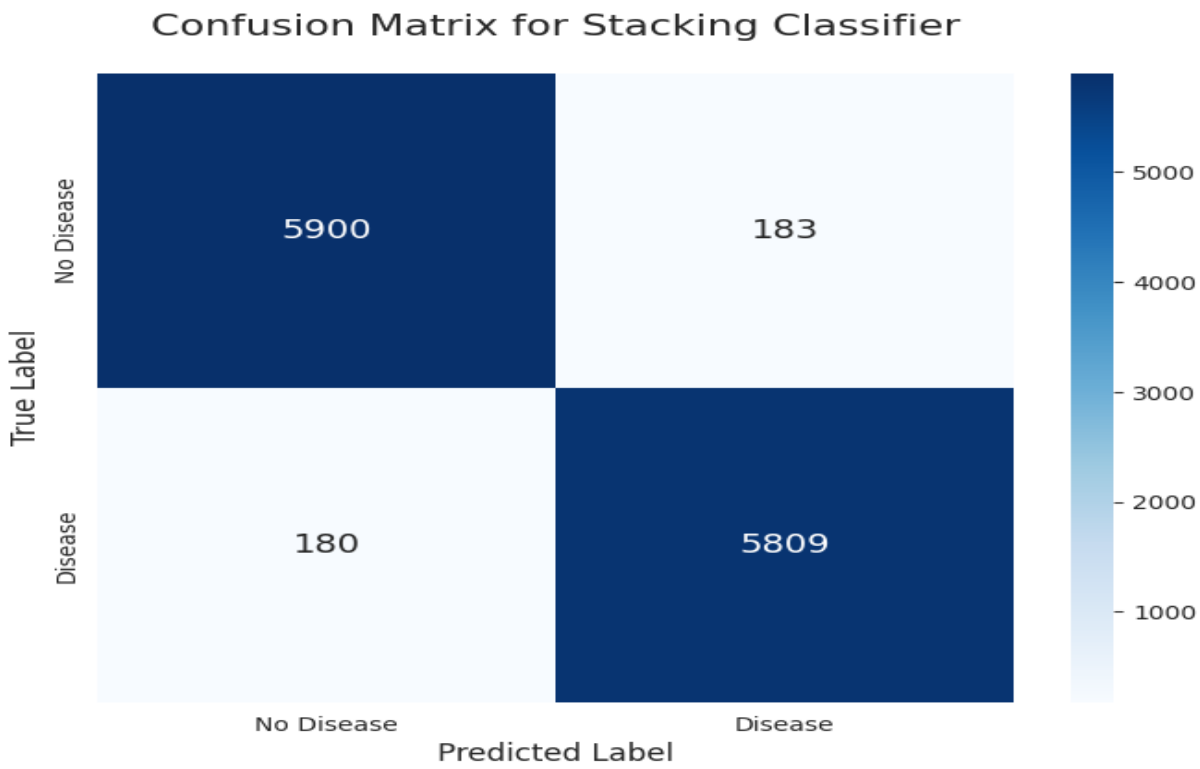
For the "Disease" class, our recall (sensitivity) was 0.97. This is the major one. This indicates that 97% of all patients who actually had cardiovascular disease were successfully identified by our "net." The "missed cases" or false-negative rate was a pitiful 3%.

Our precision was 0.97 at the same time. This is equally important. This indicates that our model was accurate 97% of the time when it did sound the alarm. We weren't sounding "false alarms." This high precision reduces the anxiety and expense of needless follow-up testing, which is essential for clinical adoption.

This is merely confirmed by the resulting F1-Score of 0.97 for both classes: we discovered a model with an uncommon, ideal balance. It is just as good at identifying real cases as it is at making accurate predictions.

4.3.2. Confusion Matrix Analysis:

The predictions made on the test set are visually broken down in the confusion matrix (Figure 3.5). The confusion matrix simply provides us with the story, displaying the breakdown of all predictions.



(Figure 8: Confusion Matrix for Stacking Classifier)

The matrix analysis confirms our findings:

- True Negatives (TN): 5,900 healthy patients were correctly told they were fine.
- True Positives (TP): 5,809 sick patients were correctly identified.
- False Positives (FP): Only 183. We raised a "false alarm" for only 3% of healthy patients.
- False Negatives (FN): Only 180. We "missed" only 3% of sick patients.

This outstandingly low and balanced error profile underscores the robustness and reliability of our optimized Stacking Classifier.

CHAPTER 5

CONCLUSION

5.1. Synthesis of Key Results:

We think that a high-performance machine learning framework for the prediction of cardiovascular disease has been successfully developed and validated as a result of our work. Three potent algorithms—Random Forest, XGBoost, and LightGBM—were methodically contrasted with their "team-up" ensemble equivalents.

1.**The secret to getting better results is teamwork**. The individual, optimized components were consistently outperformed by the ensemble models. Combining strong, diverse learners is the best strategy, as evidenced by the Stacking Classifier's highest test accuracy of 96.74%.

2. **Meta-Learning Efficacy:** Having an informed "manager" is crucial. The Stacking Classifier's superiority over the more straightforward Voting Classifier shows how well its meta-learning architecture works. It was more successful to use a logistic regression model rather than a simple average to determine the best way to combine predictions.

3. **Clinical viability:** The model is well-balanced. After carefully examining the Stacking Classifier, we found that its Precision, Recall, and F1 scores were an astounding 0.97. The model reduces false alarms (high precision) and detects true disease cases (high recall).4.2. contributions to research.

The comprehensive, repeatable pipeline we created to produce a high-accuracy clinical prediction model is our primary contribution. We have shown the strength of contemporary GBDT algorithms (XGBoost, LightGBM) and verified that, despite their remarkable performance, it can be further enhanced by intelligent ensembling (using Stacking) and systematic optimization (using Optuna). The finished model is an effective proof-of-concept for a data-driven, effective CVD screening tool.

5.2. Research Contributions.

Our main contribution is the complete, reproducible pipeline we developed to create a high-accuracy clinical prediction model. We've demonstrated the power of modern GBDT algorithms (XGBoost, LightGBM) and confirmed that, while impressive, their performance can be improved further through systematic optimization (using Optuna) and intelligent ensembling (using Stacking). The final model is a powerful proof-of-concept for an efficient, data-driven CVD screening tool.

5.3. Limitations and Future Work.

Although we are realistic, we are proud of the outcome. The limitations of this study are well-defined and need to be addressed.

5.3.1. Data Source: A single, sizable dataset was used to train and validate our model. It is entirely unknown how well it works with patient data from different hospitals, geographical areas, and ethnic groups

5.3.2. Interpretability: Our champion model is "black box." It makes a clear prediction, but it's unclear why. This is a major obstacle to clinical trust.

5.3.3. Feature Scope: There were only 17 features in the dataset. Other potentially potent predictors, like genetic information, comprehensive diet logs, and electrocardiogram (ECG) trace data, are not included in the model

These limitations must be directly addressed in future research.

Our trained Stacking model needs to be validated on fresh, varied patient datasets in order to guarantee generalizability (external validation).

Model Interpretability: "Open the black box." Future research on the Stacking model should employ model-agnostic interpretability strategies. By outlining the characteristics that influence each patient's prediction, this would increase openness and confidence.

Adding temporal data from wearable devices to the framework could aid in the early detection of disease onset.

References

1. Naz, M., Khalid, A., Hameed, A., Taj, R., Mumtaz, W., Alotaibi, F. A., & Alnfiai, M. M. (2025). Meta-Ensemble Learning for Heart Disease Prediction: A Stacking-Based Approach with Explainable AI. *IEEE Access*, 1–1. https://doi.org/10.1109/access.2025.3588683

2. Liu, J., Dong, X., Zhao, H., & Tian, Y. (2022). Predictive Classifier for Cardiovascular Disease Based on Stacking Model Fusion. *Processes*, *10*(4), 749. https://doi.org/10.3390/pr10040749

3. Sen, K., & Verma, B. (2023). *Heart Disease Prediction Using a Soft Voting Ensemble of Gradient Boosting Models, RandomForest, and Gaussian Naive Bayes*. https://doi.org/10.1109/incet57972.2023.10170399

4. Lai, L.-H., Lin, Y.-L., Liu, Y.-H., Lai, J.-P., Yang, W.-C., Hou, H.-P., & Pai, P.-F. (2024). The Use of Machine Learning Models with Optuna in Disease Prediction. *Electronics*, *13*(23), 4775–4775. https://doi.org/10.3390/electronics13234775

5. Bhatt, C. M., Patel, P., Tarang Ghetia, & Mazzeo, P. L. (2023). Effective Heart Disease Prediction Using Machine Learning Techniques. *Algorithms*, *16*(2), 88–88. https://doi.org/10.3390/a16020088

6. Saboor, A., Usman, M., Ali, S., Samad, A., Abrar, M. F., & Ullah, N. (2022). A Method for Improving Prediction of Human Heart Disease Using Machine Learning Algorithms. *Mobile Information Systems*, *2022*, 1–9. https://doi.org/10.1155/2022/1410169

7. Liu, T., Krentz, A., Lu, L., & Curcin, V. (2024). Machine learning based prediction models for cardiovascular disease risk using electronic health records data: systematic review and meta-analysis. *European Heart Journal - Digital Health*, *6*(1), 7–22. https://doi.org/10.1093/ehjdh/ztae080

8. Shilpa, T., & paul, anal. (2022). *WITHDRAWN: CVDPF: A Hybrid Feature Selection Method with Data-Driven Approach for Cardiovascular Disease Prediction Framework using Machine Learning*. https://doi.org/10.21203/rs.3.rs-2323170/v1

9. Pathak, A., Touhidul Alam Seyam, Chakraborty, A., Santa, N. K., Uddin, E., & Tasmim Akther Mim. (2024). *Enhancing Cardiovascular Risk Prediction Using Support Vector Machines and Advanced Machine Learning Algorithms*. 1–6. https://doi.org/10.1109/compas60761.2024.10796805

10. Noroozi, Z., Orooji, A., & Erfannia, L. (2023). Analyzing the impact of feature selection methods on machine learning algorithms for heart disease prediction. *Scientific Reports*, *13*(1). https://doi.org/10.1038/s41598-023-49962-w

11. Tiwari, A., Chugh, A., & Sharma, A. (n.d.). Ensemble Framework for Cardiovascular Disease Prediction. Retrieved November 17, 2025, from https://arxiv.org/pdf/2306.09989

12. Li, J. (2024). Prediction of cardiovascular disease based on machine learning. Applied and Computational Engineering, 46(1), 60–66. https://doi.org/10.54254/2755-2721/46/20241090

13. Kamal, H., None Bakhtawar, Hussain, M. Z., Hasan, M. Z., Mustafa, M., Yaqub, M. A., Umar, H., Fatima, H., & Nasir, U. (2024). Heart Disease Prediction Using Machine Learning. 1–6. https://doi.org/10.1109/idicaiei61867.2024.10842908

14. Dr. Umesh Akare, Gani, A., Anushri Bhongade, Mure, D., Chatterjee, M., & Vanzuli Ramteke. (2024). Heart Disease Prediction System Using Machine Learning. IJARCCE, 13(3). https://doi.org/10.17148/ijarcce.2024.13315

15. Al-Mahdi, I. S., Darwish, S. M., & Madbouly, M. M. (2025). Heart Disease Prediction Model Using Feature Selection and Ensemble Deep Learning with Optimized Weight. Computer Modeling in Engineering & Sciences, 143(1), 875–909. https://doi.org/10.32604/cmes.2025.061623

16. None Md. Rahmathullah. (2025). Advancing Cardiovascular Disease Prediction: An Interpretive Evaluation of Machine Learning and Deep Learning Models. Journal of Information Systems Engineering & Management, 10(40s), 566–584. https://doi.org/10.52783/jisem.v10i40s.7441

17. Saha, S., Rahman, M. M., Tahmid Tamrin Suki, Alam, M. M., Alam, M. S., & Abu, M. (2024). Heart Disease Prediction Using Machine Learning Algorithms: Performance Analysis. 1–6. https://doi.org/10.1109/icaeee62219.2024.10561820

18. Babu, K., Chandar, A. G., & Kannadhasan, S. (2025). Prediction and diagnosis of cardiovascular disease using cloud and machine learning design. Journal of Cloud Computing, 14(1). https://doi.org/10.1186/s13677-024-00720-x

19. Ulianova, S. (2019). Cardiovascular Disease dataset. Kaggle.com. https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset

20. Enhancing Cardiovascular Risk Prediction Using Support Vector Machines and Advanced Machine Learning Algorithms. (2024). ResearchGate. https://doi.org/10.1109//COMPAS60761.2024.10796805

LIBRARY CLEARANCE

PLAGARISM REPORT

ACCOUNT CLEARANCE