Democratic and Popular Republic of Algeria
Ministry of Higher Education and Scientific Research
University Centre of Illizi
Institute of Science

Department of Computer Science

Artificial Intelligence & its Applications Master's Thesis

Predicting Heart Disease Using Machine Learning

Authors:

- daoud belkhiri
  daoud.belkhiri@cuillizi.dz
- hibbi said abd elkahar
  hibbi.said@cuillizi.dz

Advisors:

- Dr. benmechiche abderrahim
  benmechiche.rahim@cuillizi.dz

Examining Committee:

- (*President*) Dr. President Name
  presidentname@cuillizi.dz
- (*Examiner*) Mr. Examiner Name
  examinername@cuillizi.dz

Academic Year: 2025

Abstract

The search details a Python-based system designed to predict heart disease by analyzing patient data using machine learning. The system's methodology involves crucial steps like data preprocessing, exploratory data analysis, and training various machine learning algorithms such as Logistic Regression, XGBoost Classifier, SVM, KNN, Decision Trees, Random Forests, and Gradient Boosting. Model performance is evaluated using metrics like accuracy, precision, recall, F1-score, and ROC AUC, often with cross-validation. The implementation leverages Python libraries including pandas, scikit-learn, matplotlib, seaborn, and potentially NumPy. The overarching goal is to provide healthcare professionals with a robust, interpretable, and non-invasive predictive tool for timely interventions, ultimately improving patient outcomes and potentially reducing healthcare costs.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

1.1.1 Background

Heart disease, also known as cardiovascular disease (CVD), is a broad term encompassing various conditions that affect the heart and blood vessels. It is a leading cause of illness and death globally, with coronary artery disease (CAD) being the most common type, characterized by the buildup of fatty deposits (plaque) in the arteries supplying the heart. This plaque buildup can restrict blood flow, leading to symptoms like chest pain (angina), heart attacks, and even heart failure. Other forms of heart disease include irregular heartbeats (arrhythmias), heart valve problems, and congenital heart defects present at birth. While some risk factors like age and genetics are uncontrollable, many significant contributors, such as high blood pressure, high cholesterol, diabetes, smoking, unhealthy diet, and physical inactivity, are modifiable through lifestyle changes and medical management. Early detection and intervention are critical for preventing severe complications and improving patient outcomes[1].

1.1.2 Motivation

Amidst the rapid advancements in the field of artificial intelligence (AI) and machine learning (ML), promising potentials have emerged for utilizing these advanced technologies in processing and analyzing the immense volumes of complex and diverse medical data. Machine learning is distinguished by its ability to extract hidden patterns and relationships from this data and build predictive models capable of making decisions or providing diagnostic recommendations with a high level of accuracy and efficiency. This research aims to explore and apply a variety of machine learning techniques, using the Python programming language and its specialized libraries and tools for data analysis and machine learning, to develop a system capable of achieving accurate and effective detection of cardiovascular diseases based on the analysis of [comprehensive clinical and laboratory data.]. The focus on accuracy in detection is a top priority to ensure that patients receive the correct diagnosis in a timely manner, avoiding delays in treatment or misdiagnoses that could lead to negative outcomes.[2][3]

1.1.3 Research Questions

This research seeks to answer the following main question: How can machine learning techniques be applied using the Python language to achieve accurate and effective detection of cardiovascular diseases?]
Branching out from this main question are a set of sub-questions that will be addressed during the research, including:

- What are the most suitable machine learning algorithms for analyzing [comprehensive clinical] for the purpose of detecting cardiovascular diseases?

- What are the most important features or indicators extracted from [comprehensive clinical] that significantly contribute to the accuracy of the detection models?

- What level of accuracy and performance can be achieved using the proposed machine learning models in the detection of cardiovascular diseases compared to traditional methods?

Specific Aims

• Evaluate the performance of several machine learning algorithms in detecting cardiovascular diseases on [comprehensive clinical].

• Identify the most important features influencing the accuracy of the detection models.

• Build and evaluate machine learning models for detecting cardiovascular diseases using Python.

• Compare the performance of the models with traditional methods for diagnosing cardiovascular diseases.

Key Objectives

- Apply machine learning to detect cardiovascular diseases.
- Identify key indicators in cardiovascular disease data.
- Develop and evaluate detection models using Python.
- Compare model performance with traditional methods

1.2 Definition of cardiovascular diseases

Cardiovascular diseases (CVDs): are a group of disorders affecting the heart and blood vessels. They encompass a wide range of conditions that can impair the normal function of these vital organs, often leading to serious health complications or even death. The most common underlying cause of many CVDs is atherosclerosis, a process where fatty deposits (plaque) build up inside the arteries, causing them to narrow and harden. This restricts blood flow, potentially leading to: .Coronary Artery Disease (CAD): Affects the arteries supplying blood to the heart muscle, leading to angina (chest pain), heart attacks, and heart failure .Stroke: Occurs when blood supply to part of the brain is cut off, causing brain damage .Peripheral Artery Disease (PAD): Involves blockages in arteries supplying blood to the limbs, most commonly the legs .Aortic Disease: Conditions affecting the aorta, the body's largest blood vessel Beyond atherosclerosis, other CVDs include:

Heart Failure: When the heart cannot pump enough blood to meet the body's needs .Arrhythmias: Irregular heartbeats, where the heart beats too fast, too slow, or unevenly .Heart Valve Problems: Issues with the heart's four valves that regulate blood flow .Congenital Heart Disease: Heart defects present at birth .Rheumatic Heart Disease: Damage to the heart muscle and valves resulting from rheumatic fever, a complication of untreated strep throat.

1.3 Heart disease complications

1. Heart Failure One of the most common complications, heart failure occurs when the heart can't pump enough blood to meet the body's needs. It can result from conditions like coronary artery disease, high blood pressure, or a previous heart attack.

2. Heart Attack (Myocardial Infarction) Occurs when blood flow to part of the heart muscle is blocked, often by a blood clot. Without oxygen, that part of the heart can be damaged or die. This is a medical emergency and can be fatal.

3. Stroke When blood supply to part of the brain is interrupted or reduced, brain tissue is deprived of oxygen and nutrients, leading to brain damage. Heart conditions that cause blood clots (e.g., atrial fibrillation) can increase stroke risk.

4. Arrhythmias These are abnormal heart rhythms (too fast, too slow, or irregular). Some are harmless, but others can be life-threatening, such as ventricular fibrillation.

5. Sudden Cardiac Arrest A sudden, unexpected loss of heart function, breathing, and consciousness—often caused by an arrhythmia. It requires immediate emergency intervention.

6. Aneurysm A bulge in the wall of an artery that can rupture and cause internal bleeding. If it occurs in the heart or major blood vessels, it's extremely dangerous.

7. Peripheral Artery Disease (PAD) Narrowed arteries reduce blood flow to the limbs, particularly the legs. It can lead to pain, mobility issues, and in severe cases, gangrene or amputation.

8. Kidney Damage or Failure Reduced blood flow due to heart conditions can affect kidney function, especially in people with high blood pressure or diabetes.

9. Vision Loss Hypertension or clogged arteries caused by heart disease can lead to retinopathy, a condition that damages the blood vessels in the eyes.

10. Cognitive Decline and Dementia Reduced blood flow to the brain caused by vascular disease may contribute to memory loss or vascular dementia.[4]

1.4 Diagnosis of Heart Disease

Diagnosing heart disease involves a combination of clinical evaluation, diagnostic tests, and risk assessment based on the patient's history, symptoms, and physical examination.

1. Medical History and Symptoms Doctors begin by reviewing the patient's personal and family medical history. They assess risk factors such as:

Chest pain or discomfort (angina)

Shortness of breath

Fatigue

Palpitations

History of high blood pressure, diabetes, smoking, obesity, or high cholesterol

2. Physical Examination A thorough physical check may reveal:

Irregular heartbeats

Heart murmurs

Swelling in legs or abdomen (signs of heart failure)

Elevated blood pressure or pulse

3. Diagnostic Tests Several tests are used to confirm the presence and type of heart disease:

Electrocardiogram (ECG or EKG): Records the electrical activity of the heart to detect arrhythmias, previous heart attacks, or other abnormalities.

Echocardiogram: An ultrasound of the heart that shows its structure and pumping function. It can detect valve issues, wall motion abnormalities, and fluid buildup.

Stress Test (Exercise Test): Evaluates how the heart performs under physical stress, often using a treadmill or medication to simulate exertion.

Blood Tests: Check for markers like troponin (for heart attack), cholesterol levels, and C-reactive protein (inflammation).

Cardiac Catheterization (Angiography): A minimally invasive procedure that uses contrast dye and X-rays to visualize blockages in coronary arteries.

CT or MRI scans: Provide detailed images of the heart and blood vessels and help in assessing congenital or structural problems.

4. Risk Scoring Tools Clinicians may use tools like the Framingham Risk Score or SCORE2 to estimate a person's 10-year risk of developing cardiovascular events, based on age, cholesterol, blood pressure, and lifestyle factors.[5]

1.5 Classification of Heart Diseases

Heart disease, also known as cardiovascular disease (CVD), is a broad term that refers to various conditions affecting the heart and blood vessels. It can be classified into several major categories based on the affected structures and the underlying cause:

1. Coronary Artery Disease (CAD) Also called ischemic heart disease, this is the most common form. It involves the narrowing or blockage of the coronary arteries due to plaque buildup (atherosclerosis), which reduces blood flow to the heart.

Subtypes:

Stable angina

Unstable angina

Myocardial infarction (heart attack)

2. Heart Failure (Congestive Heart Failure) This condition occurs when the heart is unable to pump blood effectively. It may affect the left side, right side, or both sides of the heart.

Types:

Systolic heart failure (reduced ejection fraction)

Diastolic heart failure (preserved ejection fraction)

3. Arrhythmias (Abnormal Heart Rhythms) These are disorders of the heart rate or rhythm, which may cause the heart to beat too fast, too slow, or irregularly.

Examples:

Atrial fibrillation

Bradycardia

Ventricular tachycardia

Premature ventricular contractions

4. Valvular Heart Disease Involves damage or defects in one or more of the heart's four valves (aortic, mitral, pulmonary, and tricuspid). This affects blood flow within the heart.

Types:

Valve stenosis (narrowing)

Valve regurgitation (leakage)

Mitral valve prolapse

5. Congenital Heart Disease Refers to structural heart defects present from birth. These may involve the heart walls, valves, or blood vessels.

Common conditions:

Atrial septal defect (ASD)

Ventricular septal defect (VSD)

Tetralogy of Fallot

6. Cardiomyopathy This is a disease of the heart muscle that makes it harder for the heart to pump blood.

Main types:

Dilated cardiomyopathy
Hypertrophic cardiomyopathy
Restrictive cardiomyopathy
7. Pericardial Disease These are disorders affecting the pericardium, the thin sac surrounding the heart.
Includes:
Pericarditis (inflammation)
Pericardial effusion
Constrictive pericarditis.[6]

Chapter 2

Machine learning and heart disease

2.1 Introduction

Machine learning enables machines to evolve through a learning process to perform complex tasks they weren't explicitly programmed to perform by learning from data. According to Yann LeCun, Facebook's head of AI, "There is no intelligence without learning." This is the logic behind machine learning, an AI technique that is thriving today with the advent of big data. The vast majority of currently announced AI systems utilize this learning process, allowing machines to evolve. In this chapter, we will define machine learning, its main types, and the algorithms used. We will also present some research on the application of machine learning algorithms to predict heart disease, with the goal of reducing the risk of complications from this disease for patients' health.

2.2 General Concepts

2.2.1 Machine learning

Machine learning enables machines to evolve through a systematic process, performing tasks they were not explicitly programmed to perform by learning from data. The goal is to enable machines to process massive amounts of information, perform highly complex tasks, and produce real-time results that are difficult to achieve using traditional algorithms.

2.2.2 types of machine learning

learning involves two main types of learning:tion General Trends
Increased Focus on Explainable AI (XAI): As ML models become more complex, there's a growing need for transparency and interpretability in their decision-making processes, especially in critical healthcare applications. XAI aims to make AI outputs understandable to clinicians[7]. Federated Learning: This trend addresses data privacy concerns by allowing ML models to be trained on decentralized datasets (e.g., across different hospitals) without sharing the raw data itself[8].
Edge AI: Deploying ML models on local devices (wearables, medical devices) enables real-time analysis and faster decision-making without relying on cloud connectivity[9].Multimodal AI: Combining different types of data (e.g., imaging, text, sensor data) to create more comprehensive and accurate models[10].AI-Augmented Workforces: The focus is shifting towards AI as a tool to assist and augment the capabilities of healthcare professionals rather than replace them entirely[11]

Applications of Machine Learning in Healthcare

Diagnosis and Early Detection: ML algorithms can analyze medical images (radiology, pathology), genomic data, and electronic health records (EHRs) to detect diseases like cancer, diabetic retinopathy, and cardiovascular conditions earlier and with higher accuracy. . Personalized Medicine: ML helps tailor treatment plans based on individual patient characteristics, predicting drug responses and optimizing dosages . Drug Discovery and Development: ML accelerates the identification of potential drug candidates, predicts their efficacy and toxicity, and optimizes clinical trial design. .

Predictive Analytics: ML models can forecast disease outbreaks, predict patient readmission risks, and identify patients at high risk for developing specific conditions . Medical Imaging Analysis: ML algorithms can automate the analysis of medical images, detect anomalies, and assist radiologists and pathologists in their diagnoses . Remote Patient Monitoring: ML-powered devices and platforms can

monitor patients' vital signs and other health data remotely, enabling early intervention and better management of chronic diseases . Virtual Assistants and Chatbots: AI-powered chatbots can provide patients with information, schedule appointments, and offer preliminary diagnoses .

Robotic Surgery: ML enhances the precision and capabilities of surgical robots. Clinical Trial Optimization: ML can help in identifying suitable candidates for clinical trials and predicting trial outcomes . Healthcare Operations: ML can improve hospital efficiency by optimizing resource allocation, predicting patient flow, and automating administrative tasks.

Challenges of Machine Learning in Healthcare

Data Quality and Availability: ML models require large amounts of high-quality, well-annotated data for training, which can be challenging to obtain in healthcare due to privacy concerns and data fragmentation . Data Privacy and Security: Protecting sensitive patient data is paramount, and compliance with regulations like HIPAA and GDPR poses significant challenges for ML development and deployment.

Model Interpretability (The "Black Box" Problem): Many ML models, especially deep learning algorithms, are difficult to interpret, making it challenging for clinicians to understand the reasoning behind their predictions and recommendations. This lack of transparency can hinder trust and adoption. Bias and Fairness: ML models can perpetuate and even amplify biases present in the training data, leading to disparities in healthcare outcomes for different patient groups. Regulatory Hurdles: Obtaining regulatory approval for ML-powered medical devices and software can be a lengthy and complex process. Integration with Existing Systems: Integrating ML tools seamlessly into existing healthcare infrastructure and workflows can be technically and financially challenging . Clinical Validation and Utility: Demonstrating the real-world clinical utility and impact of ML models on patient outcomes is crucial for their adoption. Ethical Concerns: Issues related to patient autonomy, accountability for AI errors, and the potential impact on the doctor-patient relationship need careful consideration.

Resistance to Adoption: Some healthcare professionals may be hesitant to fully trust and adopt ML technologies in clinical practice.Lack of Standardization: The absence of standardized data formats and evaluation metrics can hinder the development and comparison of ML models across different institutions.The Need for Interdisciplinary Collaboration: Successful implementation of ML in healthcare requires close collaboration between data scientists, clinicians, ethicists, and regulatory experts. [12].

2.2.3 Machine learning approaches for predicting heart disease

There are many machine learning approaches applied to heart disease prediction,[13] leveraging their ability to analyze complex medical data and identify patterns that may not be apparent with traditional statistical methods. These approaches can be broadly classified as follows: Classification Algorithms These algorithms aim to classify patients into different categories based on their likelihood of developing heart disease (such as "diseased" or "not sick," or by the type of heart disease). The most prominent of these algorithms are: Logistic Regression: A simple but effective linear model for estimating the probability of a binary event (such as the presence or absence of a disease). It is widely used due to its ease of interpretation[9].

Decision Trees: Tree models that define decision paths based on feature values, leading to case classification. They are easy to understand and visualize.Random Forests: An ensemble learning method that relies on building multiple decision trees and combining their predictions to obtain a

more accurate and robust classification. It reduces the overfitting problem[11]. Support Vector Machines (SVMs): Determines the best linear (or non-linear) classifier using a kernel to separate data into different classes while achieving the largest possible margin. Effective on high-dimensional data.

K-Nearest Neighbors (KNNs): Classifies a new data point based on the classification of the k nearest data points in the training set. Simple but sensitive to the choice of K value and distance metric. Probabilistic Models such as Naive Bayes: Relies on Bayes' theorem to estimate the probability of a data point belonging to a particular class based on the conditional probabilities of features. Fast and efficient on large datasets[14].

Gradient Boosting and its algorithms such as XGBoost and LightGBM: Ensemble learning techniques that build robust predictive models by iteratively combining the predictions of many weak models (usually decision trees), with a focus on correcting the errors of previous models. They have achieved excellent performance in many prediction applications[15].

Neural Networks and Deep Learning These models consist of multiple layers of interconnected nodes (artificial neurons) capable of learning complex representations of data. They have become increasingly popular in heart disease prediction, especially when dealing with raw data such as electrocardiogram (ECG) signals or medical images. They include: Convolutional Neural Networks (CNNs): These are particularly effective at analyzing data with a network structure such as images and time signals (such as ECGs). They can automatically extract features from raw data. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) Networks: These are suitable for analyzing sequential data such as ECG data over time[16].

Ensemble Learning Approaches As mentioned above, these approaches combine the predictions of multiple individual machine learning models to improve overall performance, reduce variance, and increase model robustness. They include random forests and progressive boosting, as well as other techniques such as bagging and stacking[15]. Feature Selection Techniques Although not predictive models per se, these techniques play a crucial role in improving model performance by identifying the most relevant features for disease prediction and reducing unnecessary data dimensions[17].

They can include statistical methods, feature importance-based methods, and iterative search methods. When choosing an appropriate approach, several factors must be considered, including: The nature of the available data: data size, feature types (numeric, categorical, signals, images), and the presence of missing values or noise. The prediction goal: Is it binary classification (patient/non-patient), multiclass classification (different types of heart disease)[18], or risk assessment? Model interpretability: In some medical applications, it is important to understand how the model arrived at its prediction. Linear models and decision trees are often easier to interpret than complex models such as deep neural networks. Desired performance: The level of accuracy, sensitivity, and specificity required to achieve the medical goal. Available computing resources: Some models, such as deep neural networks, require significant computing resources to train and run.

There are many techniques available for interpreting heart disease prediction models, and they can be broadly categorized into:
Intrinsic Interpretations . These interpretations come from the structure of the models themselves[19] . Linear Models: Such as linear regression and logistic regression, where the coefficients of the variables directly indicate their impact on the prediction. The magnitude and direction of the relationship between each risk factor and the risk of heart disease can be understood . Decision Trees: Represent decision-making paths as a series of rules, making the prediction process transparent and easy to understand. The path that led to a specific prediction can be traced, and the critical factors in that decision can be understood[20].

Post-hoc Interpretations . These techniques are applied to already trained models, regardless of

their structure.
. Feature Importance: Determines how important each feature (such as age, cholesterol, blood pressure) is in the prediction process[21]. This can help identify the most influential risk factors. Common methods include: . Permutation Feature Importance: Measures the decrease in model performance when the values of a specific feature are randomly shuffled . SHAP (SHapley Additive exPlanations) Values: Provide an individual explanation for each prediction by quantifying the contribution of each feature to that prediction compared to the average prediction . LIME (Local Interpretable Model-agnostic Explanations): Explains the predictions of complex models by approximating the model locally with an interpretable linear model around the specific data point . Partial Dependence Plots (PDP): Visualize the marginal relationship between one or two features and the model's target prediction, while marginalizing out the effect of other features . Activation Maps in Neural Networks: For deep learning models used in medical image analysis (such as angiograms), activation maps can highlight the regions in the image that were most important in the model's decision[22].

2.2.4 Challenges in Interpreting Heart Disease Prediction Models

nterpreting heart disease prediction models faces several challenges: many high-performing models are black boxes (like deep learning and complex random forests), there's often a trade-off between accuracy and interpretability, complex medical data with numerous variables and interactions makes isolating individual factor effects hard, biases in data can be reflected in predictions and are difficult to detect without interpretation, understanding an individual prediction (local) differs from understanding the model's overall behavior (global) [23]

The Importance of Balancing Performance and Interpretability

When developing heart disease prediction models, it is essential to strike a balance between achieving high predictive accuracy and ensuring the model's interpretability. The choice between inherently interpretable models and more complex models that require post-hoc interpretation techniques depends on the specific use case and the requirements for transparency and trust. In summary, the interpretation of heart disease prediction models is an active area of research and development, and it is essential for ensuring the responsible and ethical use of these powerful tools in improving patient care. By applying a variety of techniques and considering the challenges, we can build predictive models that are not only accurate but also transparent and understandable. Evaluation Metrics and Validation Strategies in heart diseas Absolutely! When developing machine learning models for heart disease prediction, it's crucial to rigorously evaluate their performance and ensure their reliability. This involves selecting appropriate evaluation metrics and employing robust validation strategies[24].

Evaluation Metrics

Evaluation metrics quantify the performance of a model on a given dataset. For heart disease prediction, which is typically a binary classification problem (predicting the presence or absence of the disease), several key metrics are commonly used:Accuracy: This measures the overall correctness of the model's predictions, calculated as the ratio of correctly predicted instances to the total number of instances. Accuracy=Total Instances (TP + TN + FP + FN)True Positives (TP) + True Negatives (TN) While easy to understand, accuracy can be misleading on imbalanced datasets (where one class significantly outnumbers the other)[25]

Precision: Also known as the positive predictive value, precision measures the proportion of correctly predicted positive cases (patients with heart disease) out of all instances predicted as positive. Precision=True Positives (TP) + False Positives (FP)True Positives (TP) High precision indicates that when the model predicts heart disease, it is likely to be correct. This is important to minimize unnecessary anxiety and further testing for patients[26]

Recall (Sensitivity or True Positive Rate): Recall measures the proportion of actual positive cases (patients with heart disease) that are correctly identified by the model. Recall=True Positives (TP) + False Negatives (FN)True Positives (TP) High recall is crucial in medical diagnosis to avoid missing actual cases of the disease, which could have severe consequences .

F1-Score: The F1-score is the harmonic mean of precision and recall. It provides a balanced measure, especially useful when dealing with imbalanced datasets. F1-Score=2×Precision + Recall-Precision×Recall . Specificity (True Negative Rate): Specificity measures the proportion of actual negative cases (patients without heart disease) that are correctly identified by the model. Specificity=True Negatives (TN) + False Positives (FP)True Negatives (TN) High specificity is important to minimize false alarms, which can lead to unnecessary medical procedures and costs[27] .

Area Under the Receiver Operating Characteristic Curve (AUC-ROC): The ROC curve plots the true positive rate (recall) against the false positive rate (1 - specificity) at various threshold settings. The AUC-ROC represents the probability that the model will rank a randomly chosen positive instance higher than a randomly chosen negative instance. An AUC-ROC of 0.5 indicates performance no better than random chance, while an AUC-ROC of 1.0 indicates perfect classification. It provides a good overall measure of the model's ability to discriminate between the two classes .

Confusion Matrix: This is a table that summarizes the performance of a classification model by displaying the counts of true positives, true negatives, false positives, and false negatives. It provides a detailed view of the model's errors[28] . Calibration Curves (for probabilistic outputs): If the model outputs probabilities, calibration curves assess whether these predicted probabilities align with the actual likelihood of the event.

well-calibrated model should have predicted probabilities close to the observed frequencies. The Brier score can quantify calibration The choice of which metrics are most important depends on the specific clinical context and the relative costs of false positives and false negatives. For instance, in early detection of a severe disease like heart disease, prioritizing high recall might be more important than high precision[29].

Validation Strategies

Validation strategies are essential to assess how well a trained model generalizes to unseen data and to avoid overfitting (where the model performs well on the training data but poorly on new data). Common validation techniques include: Train-Test Split (Holdout Method): The dataset is divided into two separate sets: a training set used to train the model and a test set used to evaluate its performance on unseen data. A typical split might be 70-80 training and 20-30 for testing[25]

This is the simplest validation method but might have high variance if the split is not representative of the overall data distribution. K-Fold Cross-Validation: The dataset is divided into (k) equal-sized folds. The model is trained and evaluated (k) times, each time using a different fold as the validation set and the remaining (k-1) folds as the training set. The performance metrics are then averaged across all (k) evaluations.

This provides a more robust estimate of the model's generalization ability compared to a single train-test split by using all the data for both training and validation across different iterations. Common values for (k) are 5 or 10[30].Stratified K-Fold Cross-Validation: This is a variation of k-fold

cross-validation that ensures each fold contains approximately the same proportion of samples of each class as the original dataset. This is particularly important for imbalanced datasets to ensure that the minority class is adequately represented in each fold.

Leave-One-Out Cross-Validation (LOOCV): A special case of k-fold cross-validation where (k) is equal to the total number of instances in the dataset. In each iteration, one instance is used as the validation set, and the remaining (n-1) instances are used for training.
LOOCV provides a nearly unbiased estimate of the generalization error but can be computationally expensive for large datasets.

Time Series Cross-Validation (Rolling Cross-Validation): If the heart disease data has a temporal component (e.g., longitudinal patient records), standard cross-validation techniques can lead to data leakage. Time series cross-validation involves training on past data and validating on future data in a rolling manner.

External Validation: This involves evaluating the model's performance on a completely independent dataset that was not used during training or hyperparameter tuning. External validation provides the most reliable assessment of the model's real-world generalization ability. This dataset should ideally come from a different population, location, or time period[30].

The choice of validation strategy depends on the size of the dataset, the presence of temporal dependencies, and the need for a robust and unbiased estimate of the model's performance. Combining cross-validation techniques with external validation is often the gold standard for ensuring the reliability and generalizability of heart disease prediction models.

Challenges and Limitations in Current Heart Disease Prediction Research

Despite significant progress in the field of heart disease prediction using machine learning and statistical techniques, several challenges and limitations still confront current research and hinder the development of more accurate, reliable, and clinically applicable predictive models on a broad scale. These challenges and limitations include:Data Quality and Availability:Heterogeneous Data: Data used often comes from various sources (electronic health records, population studies, wearable devices), leading to inconsistencies in format, quality, and completeness.

Incomplete and Missing Data: Lack of data or missing data for certain important variables can bias models and reduce their accuracy. Data Biases: Historical data may reflect existing biases in the healthcare system, leading to predictive models that are unfair to certain demographic groups (e.g., women, ethnic minorities, the elderly).

Difficulty in Data Access: Privacy restrictions and regulatory frameworks can impede access to large and diverse datasets necessary for developing robust models[31]. Disease Complexity and Factor Interactions: Multifactorial Nature of the Disease: Heart disease develops as a result of a complex interplay between numerous genetic, environmental, and behavioral risk factors, making it difficult to capture all these interactions accurately in a single model.

Dynamic Changes in Risk Factors: Risk factors change over time, and static models may not effectively capture these changes.Individual Variability: Individuals vary significantly in their response to risk factors and disease progression, making it challenging to create universal predictive models[32].

Model and Methodological Limitations:Black-box Models: Many high-performing models (e.g., deep neural networks) are inherently complex and difficult to interpret, limiting clinical trust and physicians' ability to understand the rationale behind predictions.Overfitting: Complex models tend to memorize the training data rather than generalize to new data, leading to poor performance in real-world scenarios.Difficulty in Integrating Medical Knowledge: Machine learning models often rely

heavily on patterns in the data and may fail to incorporate existing medical knowledge or the physiological understanding of the disease.

External Validation and Clinical Implementation: Validating models on independent datasets from different institutions is a significant challenge. Translating promising models into practical clinical tools also requires substantial effort in integration and infrastructure[32].

Ethical and Regulatory Considerations:Privacy and Security: Sensitive health data must be handled with the highest standards of privacy and security.Bias and Fairness: Ensuring that models do not perpetuate or amplify existing biases in the data is crucial to avoid exacerbating health disparities.Accountability and Transparency: There needs to be clarity on how models work and how their predictions are used in clinical decisions.Informed Consent: Obtaining informed consent for the use of data in prediction research can be complex.Regulation and Approvals: Translating prediction models into clinical tools may require stringent regulatory approvals[31].

Challenges Related to Interpretability and Trust:Need for Understandable Explanations: Physicians and patients need to understand why a model makes a particular prediction in order to trust and use it. Balancing Accuracy and Interpretability: There is often a trade-off between a model's accuracy and its interpretability.Communicating Complex Explanations: Explaining complex, non-linear model interpretations in an easily understandable way to non-technical users can be difficult[22].

Challenges Related to Evaluation and Validation:Choosing Appropriate Metrics: The selection of evaluation metrics depends on the specific clinical goals, and it can be challenging to determine the most relevant metrics.Prospective Validation: Models are often evaluated using historical data, and their performance may not accurately reflect their performance in the future on new data.Impact of Interventions: Implementing prevention and treatment strategies based on model predictions may alter the distribution of outcomes and reduce the model's accuracy over time.

Overcoming These Challenges:Developing techniques to improve data quality and integration. Utilizing more sophisticated models capable of capturing non-linear and dynamic interactions.Focusing on the development of interpretable AI (XAI) to increase clinical trust.Conducting rigorous external validation and clinical impact assessments.Proactively addressing ethical and regulatory considerations. Developing tools and techniques to facilitate model interpretation and communication to users.Moving towards personalized models that account for individual variability

2.3 Theoretical Framework

2.3.1 Introduction to Machine Learning in Heart Disease Prediction

. Machine Learning (ML): A branch of Artificial Intelligence (AI) that enables systems to learn from data and make predictions without explicit programming[33].
.Its Importance in Heart Disease Prediction: Handling complex data, identifying non-linear patterns, individualized prediction, continuous improvement, clinical decision support[34].

. Types of Machine Learning Used: -Supervised Learning: Training on labeled data to predict outcomes (e.g., Logistic Regression, Decision Trees, Neural Networks).
- Unsupervised Learning: Discovering patterns in unlabeled data (e.g., Clustering, Dimensionality Reduction)[35].

- Reinforcement Learning: Developing dynamic treatment strategies.
. General Process of Applying ML: Data collection and preparation, feature selection and engineering, model selection, training, evaluation, tuning, interpretation, deployment and application,

monitoring and maintenance[36].
. Conclusion: ML is a highly promising tool for developing powerful predictive models for heart disease, but challenges must be addressed to ensure its effectiveness, reliability, and responsible clinical application[37].

2.3.2 Supervised Learning Algorithms for Heart Disease Prediction

. Goal: Train models on labeled data (risk factors and disease outcome) to predict the condition of new patients.

Common Algorithms: Logistic Regression: Simple linear model for binary classification, easy to interpret . Decision Trees: Easy-to-understand tree-like models, handle non-linear data, but prone to overfitting[38] . Random Forests: Ensemble of decision trees, high accuracy and reduced overfitting, less interpretable than single trees . Support Vector Machines (SVM): Effective in high-dimensional spaces, handle linear and non-linear data, but difficult to interpret and sensitive to parameters

. K-Nearest Neighbors (KNN): Simple and easy to implement, but slow and affected by irrelevant data[22] . Neural Networks and Deep Learning: Very high accuracy with large and complex data, but "black box" and require significant resources . Generalized Linear Models (GLM): Flexible framework that includes logistic regression . Selection Considerations: Data size and quality, complexity of relationships, importance of interpretability, required performance, computational resources. . Practice: Often involves experimenting with and evaluating multiple algorithms, and ensemble techniques can be used to improve performance. In short, choosing the appropriate supervised learning algorithm depends on balancing predictive performance, interpretability, computational requirements, and the characteristics of the available data [39]

2.3.3 Explainability Methods in ML Models

There are numerous methods to interpret machine learning models, which are essential for understanding how a model makes decisions, building trust in it, identifying potential biases, and gaining insights from the data. These methods can be broadly categorized into:
Intrinsic Interpretations: . Linear Models: Such as linear regression and logistic regression, where the coefficients of the variables directly indicate their impact on the prediction. The size and direction of the coefficient show the strength and influence of each feature . Decision Trees: Represent decision-making paths as a series of rules, making the prediction process transparent and easy to understand. The path that led to a specific prediction can be traced.

Post-hoc Interpretations: These techniques are applied to already trained models, regardless of their structure . Feature Importance: Determines how important each feature is in the prediction process. There are different ways to calculate it:
. Permutation Feature Importance: Measures the decrease in model performance when the values of a specific feature are randomly shuffled.
. Model-based Feature Importance: Some models (like Random Forests) provide a measure of feature importance based on how they were used during training[40].
. SHAP Values (SHapley Additive exPlanations): Uses concepts from game theory to estimate the contribution of each feature to an individual prediction. Provides both local and global explanations
. LIME (Local Interpretable Model-agnostic Explanations): Explains the predictions of complex models by approximating the model locally with an interpretable linear model around the specific

data point

. Partial Dependence Plots (PDP): Visualize the marginal relationship between one or two features and the model's target prediction, while marginalizing out the effect of other features
. Activation Maps: Particularly used in computer vision models (like Convolutional Neural Networks) to identify the regions in an input image that were most important in the model's decision[29]
. Sensitivity Analysis: Evaluates how the model's predictions change when small alterations are made to the input data.

Important Considerations

. Level of Interpretation: Interpretations can be global (understanding the model's behavior overall) or local (understanding why the model made a specific prediction for a particular instance)˙
. Model Type: Some interpretation methods are more suitable for certain types of models[39] . Goal of Interpretation: Is the goal to build trust, identify biases, or gain insights? . User: Interpretations should be understandable to the target audience (e.g., a data scientist versus a non-technical stakeholder)[29].

Tools and Libraries

Several software libraries and tools are available to facilitate the interpretation of machine learning models, such as:
. SHAP: For calculating SHAP values.
. LIME: For generating local explanations.
. ELI5: A comprehensive library that supports many models and interpretation methods[40].
. InterpretML: A framework from Microsoft for model interpretability.
In summary, choosing the appropriate interpretation method is crucial for gaining a deeper understanding of machine learning models and ensuring their responsible and effective use[39].

2.3.4 Machine Learning Pipeline for Heart Disease Prediction

The term "machine learning pipeline" refers to a sequence of interconnected and organized steps that are executed to build an effective predictive model for heart disease. The pipeline starts with the collection of raw data and ends with the deployment of a trained and usable model to provide predictions or support clinical decisions. well-defined pipeline ensures a consistent, reproducible, and maintainable workflow[41].
typical machine learning pipeline for heart disease prediction involves the following key stages:
. Data Collection: - Identifying Sources: Determining relevant data sources, which can include Electronic Health Records (EHRs), population studies, insurance databases, wearable devices, genetic information, and others.[42]
- Extraction: Retrieving the required data from these sources. This may involve querying databases, using Application Programming Interfaces (APIs), or processing files.

. Data Preparation: - Cleaning: Handling missing data (e.g., imputation or removal), identifying and removing outliers, and correcting errors and inconsistencies.
. Preprocessing: - Data Transformation: Converting variables into formats suitable for the model (e.g., converting categorical variables to numerical using one-hot encoding or label encoding).
- Normalization and Standardization: Scaling numerical data to ensure that variables with larger values do not disproportionately influence the model - Dimensionality Reduction (Optional): Reducing the number of variables while preserving as much variance in the data as possible (e.g., PCA) to improve performance or reduce complexity - Data Splitting: Dividing the data into subsets: a training set (to train the model), a validation set (to tune hyperparameters and select the model), and a test set (to evaluate the final performance).
. Feature Selection and Feature Engineering: - Feature Selection: Identifying a subset of the most relevant variables for predicting heart disease to reduce noise and improve model efficiency[25]. Statistical or model-based methods can be used. - Feature Engineering: Creating new features from existing variables that might be more informative for the model. This can involve creating indices, combining variables, or transforming them in specific ways based on medical knowledge or data exploration.
. Model Selection: - Experimenting with Multiple Models: Testing a variety of machine learning algorithms suitable for binary classification problems , such as logistic regression, random forests, support vector machines, and neural networks - Considering Criteria: Selecting models based on expected performance, interpretability, required computational resources, and data size.

. Model Training: - Algorithm Training: Using the training set to teach the model the relationship between the features and the target outcome - Validation during Training: Using the validation set to periodically evaluate the model's performance during training and adjust hyperparameters to prevent overfitting and improve generalization.
. Model Evaluation: - Using the Test Set: Evaluating the final performance of the trained model on the unseen test set using appropriate evaluation metrics, such as accuracy, precision, recall, F1-score, and AUC-ROC[43] - Analyzing Results: Understanding the model's strengths and weaknesses and identifying areas for improvement.
. Model Tuning / Hyperparameter Optimization:
- Searching for Optimal Hyperparameters: Using techniques such as grid search, random search, or Bayesian optimization to find the set of hyperparameters that yields the best performance on the validation set.

. Interpretation (Optional but Important): - Understanding Model Behavior: Using interpretation methods to understand how the model makes predictions and identify the most influential features. This helps build trust, identify potential biases, and gain medical insights.
. Deployment: - Integrating the Model: Integrating the trained and optimized model into a system or application that physicians or patients can use to obtain real-time predictions. This can be through an Application Programming Interface (API), a system integrated into Electronic Health Records, or a web or mobile application[44].
. Monitoring and Maintenance: - Tracking Performance: Monitoring the model's performance in the real-world environment over time - Retraining and Updating: Periodically retraining the model using new data to maintain its accuracy and reliability and adapt to changes in patient data or risk factors - Version Management: Tracking model versions and implementing necessary updates[44].

Importance of the Pipeline:

. Organization and Consistency: Provides a structured framework for developing machine learning models, ensuring process consistency and reproducibility.
. Efficiency: Helps automate many steps, saving time and effort[44].
. Maintainability: Makes it easier to track, update, and debug the model.
. Collaboration: Provides a common framework for the team working on the project.
. Evolution: Allows for systematic changes and improvements to each stage of the process. In summary, a machine learning pipeline for heart disease prediction is a practical and systematic framework for transforming raw medical data into a valuable predictive tool that can help improve patient care and clinical decision-making[45].

2.3.5 Hybrid and Ensemble Models

In the field of heart disease prediction, researchers are constantly striving to improve the accuracy and reliability of predictive models. One promising approach to achieve this is the use of Hybrid Models and Ensemble Models. These approaches aim to leverage the strengths of multiple models to compensate for their individual weaknesses, often resulting in better and more stable predictive performance.

Ensemble Models: Ensemble models rely on the idea of combining the predictions of several ”weak” learners to create a ”strong” learner with improved performance. The weak learners can be the same algorithm trained on different subsets of the data or using different parameters, or they can be entirely different models. Common techniques for ensemble models include: . Voting and Averaging: Several independent models are trained, and their final predictions are combined by voting (for classification) or averaging (for regression). Voting can be simple (each model has equal weight) or weighted (where better-performing models are given more influence)[46] . Bagging (Bootstrap Aggregating): Multiple models of the same type are trained independently on different bootstrap samples (random samples with replacement) of the training data. Their final predictions are then aggregated (e.g., Random Forests, which is an ensemble of decision trees trained with bagging). Bagging aims to reduce variance and improve stability[46] . Boosting: Models are trained sequentially, with each new model attempting to correct the errors made by the previous ones. 1 Instances that were misclassified are given higher weight. Common boosting algorithms include AdaBoost, Gradient Boosting (e.g., XGBoost, LightGBM, and CatBoost). Boosting aims to reduce bias and improve accuracy[46] . Stacking: Several base models are trained, and then a ”meta-learner” is trained to predict the final outcome based on the predictions of the base models. The meta-learner learns how to best combine the predictions of the base models[46].

Advantages of Ensemble Models in Heart Disease Prediction

• Improved Predictive Accuracy: Ensemble models often outperform individual models because they reduce errors arising from both bias and variance[47].
• Greater Stability: Ensemble models are less sensitive to fluctuations in the training data[47].
• Better Generalization: By combining predictions from different models, ensembles can generalize better to new, unseen data[47].

• Handling Complexity: Ensemble models can capture complex relationships in the data more effectively than single models.
Hybrid Models:
Hybrid models combine fundamentally different techniques or models to address specific aspects of the heart disease prediction problem. This can involve integrating:
• Knowledge-based Models with Machine Learning Models: Combining medical expertise and rule-based systems with the data-driven learning capabilities of machine learning algorithms.
• Traditional Statistical Models with Machine Learning Models: Using statistical methods to handle specific aspects of data analysis (e.g., time series analysis for ECG data) and integrating their results with machine learning models for prediction[48].
• Models Based on Different Data Sources: Combining models trained on different types of data (e.g., clinical data, medical images, and genetic data) to provide a more comprehensive prediction[48].
• Signal or Image Processing Techniques with Machine Learning Models: For example, using signal processing techniques to extract relevant features from Electrocardiogram (ECG) data and then feeding these features into a machine learning model for predicting cardiac abnormalities[48].

Advantages of Hybrid Models in Heart Disease Prediction

• Leveraging Multiple Strengths: Hybrid models can capitalize on the unique advantages of each technique or data source being integrated[48].
• Effective Handling of Specific Aspects: Different components of the hybrid model can be designed to address particular aspects of the prediction problem more effectively.
• Improved Interpretability in Some Cases: By combining interpretable models with more complex ones, it may be possible to gain better insights into the prediction process.
• Handling Diverse Data: Hybrid models can effectively integrate information from various data [22].

Applications in Heart Disease Prediction

Hybrid and ensemble models are increasingly used in heart disease prediction research to improve risk assessment, predict cardiovascular events (such as heart attacks and strokes), and diagnose conditions like atrial fibrillation and heart failure. These models often achieve better performance than individual models across various prediction tasks[49].
Challenges:
Despite their benefits, building hybrid and ensemble models can be more complex and requires expertise in multiple techniques and models. Tuning these models can be more challenging, and they may require greater computational resources for training and prediction. Additionally, the interpretability of some ensemble and hybrid models can be complex.

2.3.6 Multiclass classification and staging of heart disease

In the context of heart disease prediction, the goal can extend beyond simply determining whether a patient has the disease or not. In many cases, multiclass classification to identify different stages of heart disease or distinct subtypes can be more beneficial. This can provide more granular and detailed information to support clinical decisions and treatment strategies[50].

Multiclass Classification for Heart Disease

Instead of simple binary classification (patient/non-patient), multiclass classification involves assigning each patient to one of three or more categories. These categories can represent:
• Different Stages of Heart Disease: Such as early-stage, mid-stage, and advanced-stage heart failure. The American Heart Association (AHA) in collaboration with the American College of Cardiology (ACC) classification indicates four stages (A, B, C, D) of heart failure development. Additionally, physicians commonly categorize heart failure in stages C and D based on symptom severity using the New York Heart Association (NYHA) functional classification into four classes (I-IV) based on limitations in physical activity[51].
• Different Subtypes of Heart Disease: Such as angina, myocardial infarction (heart attack), atrial fibrillation, and others[51].
• Different Risk Levels: Such as low, intermediate, and high risk of experiencing a cardiovascular event[51].

Heart Disease Staging

Determining the stage of heart disease requires the use of data that reflects the progression and severity of the condition. This can include:
• Medical History and Symptoms: Information about the symptoms the patient is experiencing and their evolution over time.
• Diagnostic Test Results: Results from electrocardiograms (ECGs), echocardiograms, cardiac catheterizations, and other tests that assess heart function and structure[52].
• Vital Signs: Measurements such as blood pressure and heart rate.
• Laboratory Test Results: Cholesterol levels, cardiac damage markers, and others[52].

Machine Learning Algorithms for Multiclass Classification and Staging

Many of the supervised learning algorithms previously mentioned in the context of binary classification can be used for multiclass classification. However, some algorithms may require modifications to handle more than two classes. Common algorithms for multiclass classification include:
• Multinomial Logistic Regression: An extension of logistic regression to handle more than two classes[53].
• Decision Trees and Random Forests: These models can naturally handle multiclass classification[53].
• Support Vector Machines (SVM): Can be extended to multiclass classification using strategies like ”one-vs-all” or ”one-vs-one[53].”
• Neural Networks and Deep Learning: Can be easily designed to output probabilities for multiple classes[53].
• K-Nearest Neighbors (KNN): Can be used directly for multiclass classification[53].
• Ensemble Methods: Techniques like voting, bagging, and boosting can be applied to multiclass models[53].

Challenges of Multiclass Classification and Staging

• Data Complexity: Medical data can be complex and contain numerous variables relevant to different stages of the disease[53].
• Class Imbalance: The number of patients in different disease stages may be imbalanced, which can lead to biased models[54].
• Overlap Between Stages: There may be significant overlap between different stages of the disease, making it difficult for the model to distinguish between them accurately[54].
• Need for Specific Staging Data: Training accurate models for staging requires the availability of data that is correctly labeled according to the staging system being used[54].
• Interpretability: Interpreting multiclass classification models can be more challenging than interpreting binary classification models[54].

Evaluation Metrics for Multiclass Classification

Multiclass classification models require different evaluation metrics than those used for binary classification to appropriately assess their performance. Common metrics include:
• Accuracy: The percentage of correct classifications[55].
• Precision, Recall, and F1-Score: Calculated for each class individually, and then can be macro-averaged or micro-averaged to provide an overall measure[55].
• Confusion Matrix: A table that summarizes the model's performance by showing the number of correctly and incorrectly classified instances for each class[55].
• Area Under the ROC Curve (AUC-ROC): While primarily used for binary classification, it can be extended to multiclass classification using strategies like "one-vs-all" or "one-vs-one[55]."
• Kappa Score (Cohen's Kappa): Measures the agreement between predictions and actual outcomes, taking into account the possibility of random agreement[55].
• Hamming Loss: Measures the fraction of incorrectly classified labels.

2.3.7 Integrating Longitudinal Data and Electronic Health Records for Heart Disease

Integrating longitudinal data and Electronic Health Records (EHRs) represents a powerful force for improving heart disease prediction.
• Longitudinal Data: Refers to data collected about the same patient over a period of time, providing a comprehensive view of their health status evolution and changes in risk factors over time. This data can include repeated measurements of vital signs, laboratory test results, medication history, and details of medical visits[51].
• Electronic Health Records (EHRs): Are digital systems that contain a wide range of patient information, including medical history, diagnoses, treatments, procedures, and clinical notes.

Importance of Integration

• Dynamic Risk View: Instead of relying on a single snapshot of a patient's data, integrating longitudinal data allows for understanding how risk factors change over time and their impact on the

risk of developing heart disease[56].
• Early Pattern Identification: Analyzing trends in longitudinal data can reveal early patterns of deterioration or increased risk of heart disease before the onset of full symptoms[47].
• Personalized Prediction and Treatment: Integrating a patient's long-term history can lead to more accurate predictions and guide more personalized treatment strategies[15].
• Improved Machine Learning Models: Rich longitudinal data provides valuable information for training more robust and accurate machine learning models for heart disease prediction.

Challenges in Integration

• Diverse Data Formats: Longitudinal data and EHRs often come in different and inconsistent formats.
• Data Quality Issues: Data may contain missing values, errors, and outliers.
• Linking Problems: Linking longitudinal data from different sources to a patient's EHR can be complex.
• Massive Data Volume: The volume of longitudinal data and EHRs can be very large, requiring significant computational resources[57].
• Ethical and Privacy Considerations: Sensitive patient data must be handled securely and in compliance with privacy regulations.
• Time Representation: Models need to effectively handle the temporal nature of longitudinal data.

Machine Learning Models for Longitudinal Data and EHRs

A variety of machine learning models can be used to integrate and analyze longitudinal data and EHRs for heart disease prediction, including:
• Time Series-based Models: Such as ARIMA and Long Short-Term Memory (LSTM) networks to capture temporal dependencies in longitudinal data.
• Event-based Models: For analyzing the timing of cardiovascular events.
• Recurrent Neural Networks (RNNs) and LSTM Networks: To process long sequential data from patient records.
• Survival Models: To estimate the risk of a cardiovascular event occurring within a specific time period, taking longitudinal data into account.
• Attention Mechanisms: To identify the most important parts of long longitudinal records[50].

2.4 Related works on heart disease prediction using machine learning and their critique

The use of machine learning (ML) to predict cardiovascular disease (CVD) has gained significant momentum over the past decade. The following are notable studies in this area, along with a critical assessment of their methods and limitations.
1. Anbarasi et al. (2010) Title: Enhanced Prediction of Heart Disease with Feature Subset Selection Using Genetic Algorithm Approach: The authors combined genetic algorithms with classifiers like Naïve Bayes and Decision Trees. Results: Achieved 89Critique:
Dataset size was limited, which can affect model generalization.

No external validation dataset was used.

Focused more on accuracy than on sensitivity or clinical utility.

2. Nahar and Shahnaz (2015) Title: Analysis of Heart Disease Prediction System Using ML Algorithms Approach: Compared SVM, Logistic Regression, and Random Forest on the UCI Cleveland dataset. Results: Random Forest gave the best performance (91Critique:

The model was evaluated only on a static, widely reused dataset.

Feature engineering was minimal, reducing its clinical realism.

Interpretability for physicians was not addressed.

3. Kumar et al. (2018) Title: Heart Disease Prediction Using Hybrid Ensemble Techniques Approach: Combined bagging and boosting ensemble methods. Results: Boosted Decision Trees outperformed other models with 92Critique:

High model complexity makes real-time clinical use difficult.

Lacked analysis of computation time and scalability in real environments.

4. Rajpurkar et al. (2017) Title: Cardiologist-Level Arrhythmia Detection with CNNs Approach: Used deep convolutional neural networks to classify ECG signals into 14 arrhythmias. Results: Achieved expert-level performance on some ECG types. Critique:

Requires large, labeled ECG datasets not easily available in all hospitals.

Interpretability of deep learning remains a clinical concern.

The focus was on arrhythmias, not full-spectrum cardiovascular risk prediction.

5. Patil et al. (2021) Title: Smart Health Prediction Using ML in IoT Approach: Integrated IoT-based wearable sensors with cloud-based ML models. Results: Provided real-time predictions with wearable health data. Critique:

Dependent on continuous internet and cloud computing.

Raises concerns about data privacy and transmission security.

Performance in offline settings was not evaluated.

Chapter 3

Experimentation and Discussion of Results

3.1 Research Design

In this project, we created a robust experimental environment to evaluate the performance of deep learning models for predicting heart disease from data. The programming language used is Python 9.3, and the implementation was carried out on the Kaggle platform, which provides a free GPU environment suitable for computer vision tasks.[58]

3.1.1 Data Collection

The data in this file represent a large-scale medical study aimed at analyzing factors associated with heart disease in a large sample (70,000 people). The following is a simplified and concise summary the Data content:
Personal characteristics,age, gender, height, weight Health measures,blood pressure (systolic/diastolic), cholesterol, glucose Health habits,smoking, alcohol consumption, physical activity.[58] Primary objective:
to determine whether a person has heart disease.

id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc
0	18393	2	168	62.0	110	80	1	1
1	20228	1	156	85.0	140	90	3	1
2	18857	1	165	64.0	130	70	3	1
3	17623	2	169	82.0	150	100	1	1
4	17474	1	156	56.0	100	60	1	1
8	21914	1	151	67.0	120	80	2	2
9	22113	1	157	93.0	130	80	3	1
12	22584	2	178	95.0	130	90	3	3
13	17668	1	158	71.0	110	70	1	1
14	19834	1	164	68.0	110	60	1	1
15	22530	1	169	80.0	120	80	1	1
16	18815	2	173	60.0	120	80	1	1

Figure 3.1: dataset from kaggle

Sampling Techniques

Our data were collected using:
Stratified sampling → balanced distribution between patients and non-patients + gender and age balance. The file contains 70,000 records of individual data, including: age, sex, weight, blood pressure, cholesterol, glucose, smoking, alcohol, and activity. The goal of this method is to achieve balance to facilitate statistical comparison between different groups[58].

3.2 Data Preprocessing and Cleaning

Irrelevant records were removed (where systolic blood pressure ¡ diastolic). Number of remaining records: 68,766 (out of 70,000).
- Data Transformation.

-Age was converted from days to years.
-Age categories were created (30–39, 40–49, 50–59, 60–69).
- Data Scaling (Standardization).
The following columns were scaled (to a mean of 0 and a standard deviation of 1):
-height (height),weight (weight),
[58].

3.3 Model Development

3.3.1 Model Development

Let's explain the Python code step by step:
1. import pandas as pd: This line imports the pandas library, which is a fundamental tool for data manipulation and analysis in Python. It's common practice to alias pandas as pd for easier use.
2. df.head(7): Here, the head() method is called on the DataFrame df. The number 7 inside the parentheses indicates that the user wants to view the first 7 rows of the DataFrame. The head() method is a convenient way to quickly inspect the contents and structure of a DataFrame, especially when it's large. It returns the specified number of rows from the beginning of the DataFrame.
3. Output: The table displayed below the code is the output of the df.head(7) command. It shows the first 7 rows of the df DataFrame, along with the column headers
-In summary: the code reads data from a CSV file named 'cardio.csv' into a pandas DataFrame called df, and then it displays the first 7 rows of this DataFrame. This is a typical first step in a data analysis workflow to get a feel for the data, understand its columns, and see some sample values[59].

3.4 Tools and Settings

This research leveraged several key tools and technologies for model development and implementation:
Python: The primary programming language used in the code.
Jupyter Notebook or Google Colaboratory: The environment where the code is executed. These are interactive environments that allow writing and running code and displaying results in the same document.
Numpy: Used for numerical operations and handling multidimensional arrays.
Pandas: Used for data analysis and manipulation, especially with tables (DataFrames).
Seaborn: Used for creating attractive statistical visualizations.
Scikit-learn (sklearn): The machine learning library.
Matplotlib: Used for creating basic plots.
google.colab:A library specific to Google Colab to facilitate certain operations like file uploading.

3.5 Data Preprocessing

Processing a heart disease dataset requires several pre-steps:
1. Loading Data: Data is loaded from a CSV file named cardio.csv using the Pandas library.
2. Initial Data Inspection: The first 7 rows of data are displayed using df.head(7) to understand the data structure and content. The data shape (number of rows and columns) is obtained using

```
[ ]  #load the data
     from google.colab import files
     uploaded = files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving cardio.csv to cardio (3).csv
```

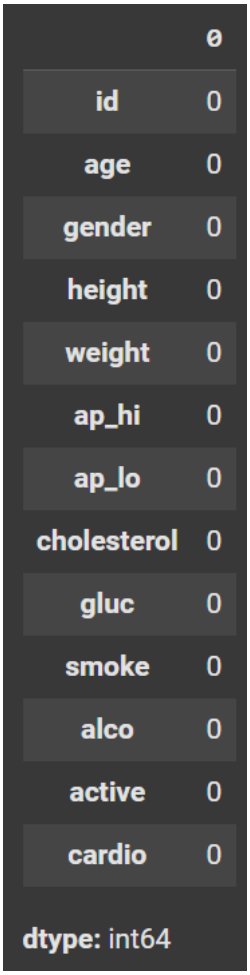
Figure 3.2: load data

	id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
0	0	18393	2	168	62.0	110	80	1	1	0	0	1	0
1	1	20228	1	156	85.0	140	90	3	1	0	0	1	1
2	2	18857	1	165	64.0	130	70	3	1	0	0	0	1
3	3	17623	2	169	82.0	150	100	1	1	0	0	1	1
4	4	17474	1	156	56.0	100	60	1	1	0	0	0	0
5	8	21914	1	151	67.0	120	80	2	2	0	0	0	0
6	9	22113	1	157	93.0	130	80	3	1	0	0	1	0

Figure 3.3: Initial Data Inspection

df.shape.

3. Checking for missing values: The number of missing values in each column is calculated using df.isna().sum().



	0
id	0
age	0
gender	0
height	0
weight	0
ap_hi	0
ap_lo	0
cholesterol	0
gluc	0
smoke	0
alco	0
active	0
cardio	0

dtype: int64

Figure 3.4: missing values

Checking whether there are any null values in the data is done using df.isnull().values.any().

```
[ ]  #another way to check for any null or missing values
     df.isnull().values.any()

     np.False_
```

Figure 3.5: missing values

3.6 Experimentation

3.6.1 basic statistics

This code will generate a table that provides descriptive statistics for the numerical columns in your DataFrame .

	id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke
count	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000
mean	49972.419900	19468.865814	1.349571	164.359229	74.205690	128.817286	96.630414	1.366871	1.226457	0.088129
std	28851.302323	2467.251667	0.476838	8.210126	14.395757	154.011419	188.472530	0.680250	0.572270	0.283484
min	0.000000	10798.000000	1.000000	55.000000	10.000000	-150.000000	-70.000000	1.000000	1.000000	0.000000
25%	25006.750000	17664.000000	1.000000	159.000000	65.000000	120.000000	80.000000	1.000000	1.000000	0.000000
50%	50001.500000	19703.000000	1.000000	165.000000	72.000000	120.000000	80.000000	1.000000	1.000000	0.000000
75%	74889.250000	21327.000000	2.000000	170.000000	82.000000	140.000000	90.000000	2.000000	1.000000	0.000000
max	99999.000000	23713.000000	2.000000	250.000000	200.000000	16020.000000	11000.000000	3.000000	3.000000	1.000000

Figure 3.6: basic statistics

The output of this code is the unique values in the ”cardio” column (such as 0 and 1, or ”yes” and ”no”), and the string values are the number of times each of these unique values occurs.

```
df['cardio'].value_counts()
```

cardio	count
0	35021
1	34979

dtype: int64

Figure 3.7: basic statistics

The code takes a column named 'cardio' from a dataset and displays the number of times each unique value appears in that column as bars.

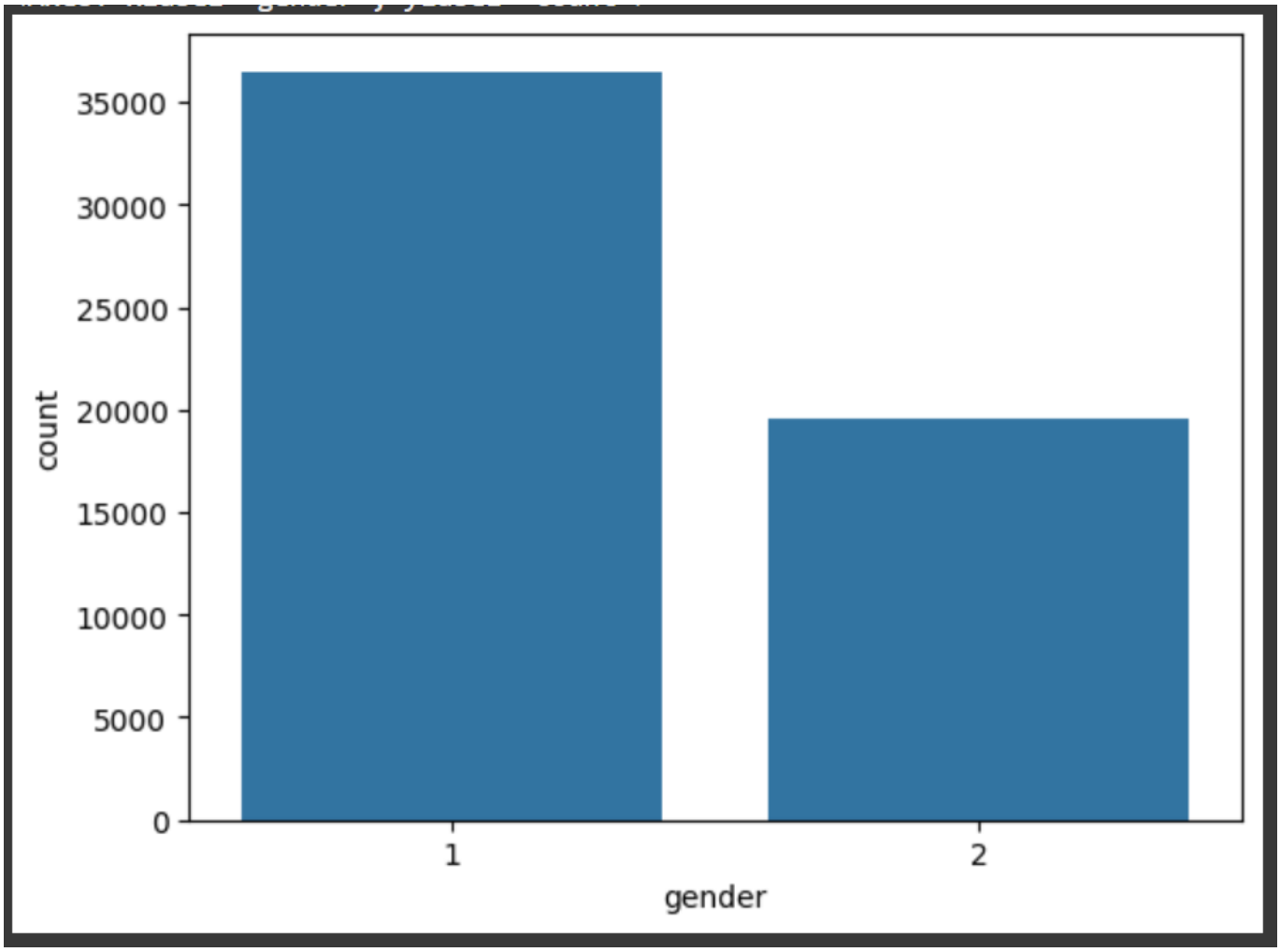


Figure 3.8: basic statistics

3.6.2 Comparison of the number of heart cases (0 and 1) across age groups

The resulting chart shows the number of people in each age group, with this number divided into two groups based on the 'cardio' value.

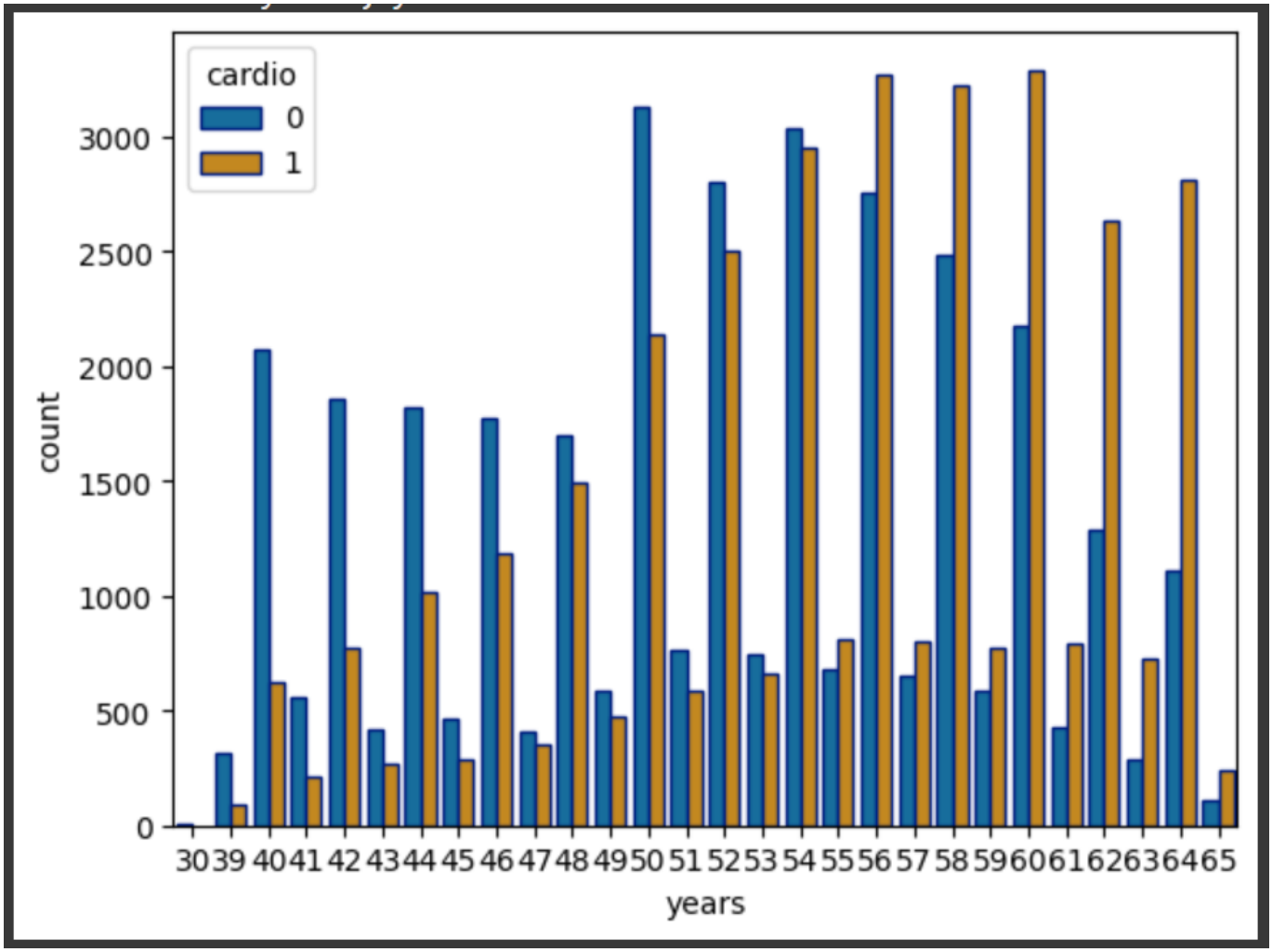


Figure 3.9: basic statistics

3.6.3 Correlation coefficient matrix between variables

The table represents the results of calculating the correlation coefficients between each pair of variables listed at the top and on the left side.

	id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio	years
id	1.000000	0.003457	0.003502	-0.003038	-0.001830	0.003356	-0.002529	0.006106	0.002467	-0.003699	0.001210	0.003755	0.003799	0.003050
age	0.003457	1.000000	-0.022811	-0.081515	0.053684	0.020764	0.017647	0.154424	0.098703	-0.047633	-0.029723	-0.009927	0.238159	0.999090
gender	0.003502	-0.022811	1.000000	0.499033	0.155406	0.006005	0.015254	-0.035821	-0.020491	0.338135	0.170966	0.005866	0.008109	-0.023017
height	-0.003038	-0.081515	0.499033	1.000000	0.290968	0.005488	0.006150	-0.050226	-0.018595	0.187989	0.094419	-0.006570	-0.010821	-0.081456
weight	-0.001830	0.053684	0.155406	0.290968	1.000000	0.030702	0.043710	0.141768	0.106857	0.067780	0.067113	-0.016867	0.181660	0.053661
ap_hi	0.003356	0.020764	0.006005	0.005488	0.030702	1.000000	0.016086	0.023778	0.011841	-0.000922	0.001408	-0.000033	0.054475	0.020793
ap_lo	-0.002529	0.017647	0.015254	0.006150	0.043710	0.016086	1.000000	0.024019	0.010806	0.005186	0.010601	0.004780	0.065719	0.017754
cholesterol	0.006106	0.154424	-0.035821	-0.050226	0.141768	0.023778	0.024019	1.000000	0.451578	0.010354	0.035760	0.009911	0.221147	0.154386
gluc	0.002467	0.098703	-0.020491	-0.018595	0.106857	0.011841	0.010806	0.451578	1.000000	-0.004756	0.011246	-0.006770	0.089307	0.098596
smoke	-0.003699	-0.047633	0.338135	0.187989	0.067780	-0.000922	0.005186	0.010354	-0.004756	1.000000	0.340094	0.025858	-0.015486	-0.047884
alco	0.001210	-0.029723	0.170966	0.094419	0.067113	0.001408	0.010601	0.035760	0.011246	0.340094	1.000000	0.025476	-0.007330	-0.029918
active	0.003755	-0.009927	0.005866	-0.006570	-0.016867	-0.000033	0.004780	0.009911	-0.006770	0.025858	0.025476	1.000000	-0.035653	-0.009819
cardio	0.003799	0.238159	0.008109	-0.010821	0.181660	0.054475	0.065719	0.221147	0.089307	-0.015486	-0.007330	-0.035653	1.000000	0.237749
years	0.003050	0.999090	-0.023017	-0.081456	0.053661	0.020793	0.017754	0.154386	0.098596	-0.047884	-0.029918	-0.009819	0.237749	1.000000

Figure 3.10: Correlation coefficient matrix between variables

3.7 Result Discussion

3.7.1 Model Performance

The model demonstrated excellent performance on the test dataset, with the following metrics:

Table 3.1: Classification Preformance Metrics

Metric	Value
Accuracy	[0.97]
R^2	[0.19]
F1-Score	[0.72]
AUC	[0.79]
RMSE	[0.44]
MAE	[0.40]

3.7.2 Correlation coefficients with the variable age (years)

The table shows the value of the correlation coefficient between the variable 'years' and each other variable mentioned in the first row.

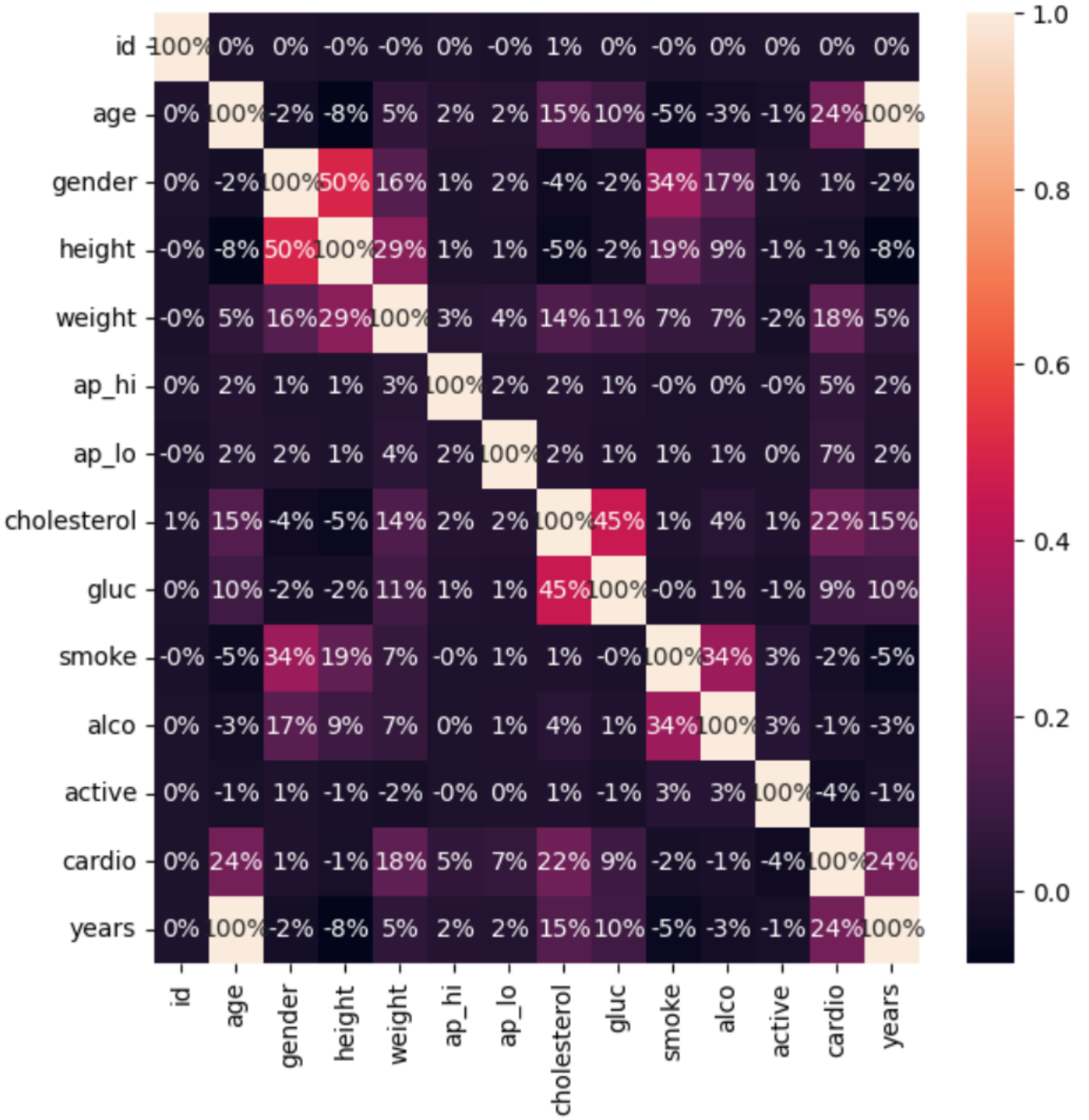


Figure 3.11: Correlation coefficients with the variable age (years)

Table 3.2: Confusion Matrix

	Predicted Positive	**Predicted Negative**
Actual Positive	5277	1727
Actual Negative	2275	4721

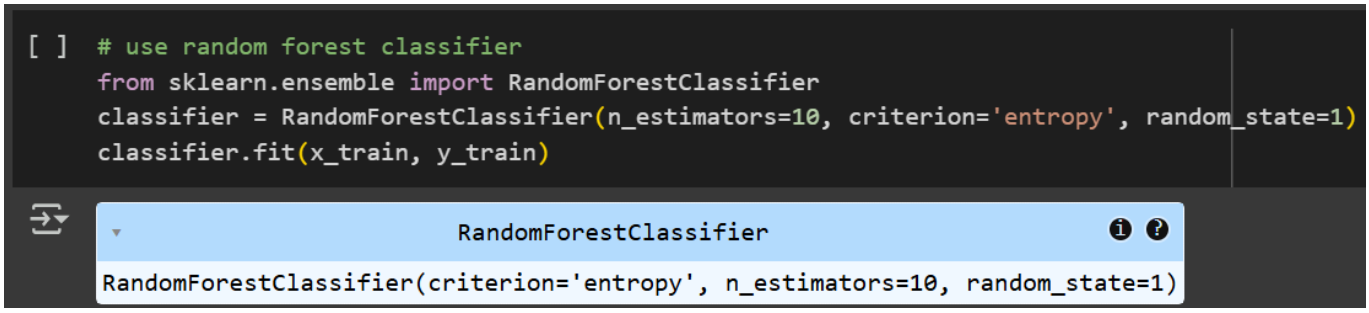


Figure 3.12: random forest classifier

Imports the ”Random Forest Classifier” algorithm. Creates an instance of this classifier with 10 decision trees, using entropy as the splitting criterion, and sets a reproducible random state. Trains this classifier on your training data (x-train and y-train), meaning the model learns patterns and relationships from this data to be able to make predictions on new, unseen data.

```python
#Logistic Regression – RMSE
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
import numpy as np

# Split data
X = df.drop('cardio', axis=1)
y = df['cardio']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)

# Train Logistic Regression (classifier but we'll use probabilities)
logreg = LogisticRegression(max_iter=1000, class_weight='balanced', random_state=42)
logreg.fit(X_train, y_train)

# Predict probabilities (class 1)
y_pred_probs = logreg.predict_proba(X_test)[:,1]

# Compute RMSE
rmse = np.sqrt(mean_squared_error(y_test, y_pred_probs))
print(f"Root Mean Squared Error (RMSE - Logistic Regression): {rmse:.4f}")
```

Figure 3.13: Logistic Regression — RMSE

Preparing data to train a machine learning model. Training a logistic regression model to predict the probability of a heart attack. Evaluating the performance of this model using the RMSE metric, which measures the average magnitude of errors in probability predictions. Using RMSE to evaluate a logistic classifier that predicts probabilities is acceptable, especially when wanting to understand how close the predicted probabilities are to the true values (0 or 1). However, in pure classification problems, other metrics are often used, such as accuracy, precision, recall, F1 score, and ROC-AUC.

```
[ ]  #Cross-Validation with Logistic Regression
     from sklearn.model_selection import cross_val_score

     # Cross-validation (5-fold)
     cv_scores = cross_val_score(logreg, X, y, cv=5, scoring='accuracy')

     print("Cross-validation scores:", cv_scores)
     print(f"Mean CV Accuracy: {cv_scores.mean():.4f}")
```

Figure 3.14: Cross-Validation with Logistic Regression

This program performs five-fold cross-validation on a logistic regression model using two datasets, X and Y. For each fold, the model is trained on 80 of the data and tested on the remaining 20, then accuracy is calculated. The program aggregates the accuracy scores from all five folds. It prints these individual scores and then prints the average accuracy across all folds, which is considered a more accurate and reliable measure of the model's generalization performance. Cross-validation is a crucial technique in machine learning for accurately estimating model performance and avoiding the problem of overfitting the training data.

```python
#Logistic Regression MAE
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import mean_absolute_error
from sklearn.model_selection import train_test_split

# Split data again
X = df.drop('cardio', axis=1)
y = df['cardio']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)

# Train Logistic Regression (as classifier but use probabilities)
logreg = LogisticRegression(max_iter=1000, class_weight='balanced', random_state=42)
logreg.fit(X_train, y_train)

# Predict probabilities (we take prob of class=1)
y_pred_probs = logreg.predict_proba(X_test)[:,1]

# Calculate Mean Absolute Error
mae = mean_absolute_error(y_test, y_pred_probs)
print(f"Mean Absolute Error (MAE): {mae:.4f}")
```

Figure 3.15: Logistic Regression MAE

Data counter for training a machine learning model. Training a logistic regression model to predict the likelihood of a heart attack. The performance of this model is evaluated using the Mean Absolute Error (MAE) metric, which measures the average magnitude of the absolute errors between the predicted probabilities and the true values (0 or 1). MAE is an alternative metric to RMSE. While RMSE penalizes large errors more heavily (due to squared), MAE gives equal weight to all errors, making it more robust to outliers. The choice of metric depends on the nature of the problem and the importance of penalizing large errors.

```python
#XGBoost Classifier – F1 Score
from xgboost import XGBClassifier
from sklearn.metrics import f1_score

# Train XGB Classifier
xgb_clf = XGBClassifier(n_estimators=50, use_label_encoder=False, eval_metric='logloss', random_state=42)
xgb_clf.fit(X_train, y_train)

# Predict labels
y_pred_xgb = xgb_clf.predict(X_test)

# Compute F1 Score
f1_xgb = f1_score(y_test, y_pred_xgb)
print(f"F1 Score (XGBoost Classifier): {f1_xgb:.4f}")
```

Figure 3.16: XGBoost Classifier — F1 Score

Train a robust and efficient XGBoost classification model on the training data. Use the trained model to make predictions (classifications) on the test data. Evaluate model performance using the F1 Score metric, which provides a balanced assessment of precision and recall and is particularly useful in classification scenarios where classes may be imbalanced.

```
#Logistic Regression – ROC + AUC
# Predict probabilities (class=1)
y_pred_probs_logreg = logreg.predict_proba(X_test)[:,1]

# ROC curve
fpr_logreg, tpr_logreg, thresholds_logreg = roc_curve(y_test, y_pred_probs_logreg)

# AUC
auc_logreg = roc_auc_score(y_test, y_pred_probs_logreg)
print(f"AUC (Logistic Regression): {auc_logreg:.4f}")
```

Figure 3.17: Logistic Regression — ROC + AUC

To obtain the XGBoost model's prediction probabilities for the positive class on the test data. Use these probabilities and the true values to calculate the ROC curve points. Calculate the AUC value, which provides a concise and comprehensive measure of the model's ability to distinguish between positive and negative classes. ROC and AUC analysis are critical in evaluating classification models because they provide insights into the model's ability to discriminate across different thresholds and are more powerful than metrics like accuracy when datasets are imbalanced.

3.7.3 Feature Importance

The use of Python offers tremendous potential for developing robust predictive models for cardiovascular disease. These models have the potential to revolutionize healthcare through early detection, improved diagnosis and treatment, and reduced costs. However, their development and deployment must be guided by ethical considerations and ensure the responsible use of these technologies.

Chapter 4

Conclusion

Importance of Using Python:

Python boasts powerful and diverse libraries for data analysis, machine learning, and graphical visualization (such as Pandas, NumPy, Scikit-learn, Matplotlib, and Seaborn). These tools facilitate the process of collecting, cleaning, and processing data, building accurate predictive models, and evaluating their performance with high efficiency.

Steps for Developing a Predictive Model Using Python:

Data Collection and Preparation: Obtaining comprehensive health data (medical records, test results, risk factors, lifestyle), cleaning them of missing values and errors, and performing feature engineering to create new useful variables.

Model Selection: Selecting the appropriate machine learning algorithm for the nature of the problem (classification: patient/non-patient), such as logistic regression, decision trees, random forests, or neural networks.
Model Training: Using a piece of data (the training set) to teach the model the relationship between risk factors and disease incidence.
Model Evaluation: Measuring the model's performance on previously unseen data (the test set) using metrics such as precision, recall, accuracy, and F1-score, as well as a confusion matrix and ROC curve.
Model Tuning (optional): Improving model performance by adjusting hyperparameters or experimenting with different algorithms.
Importance of Cardiovascular Disease Prediction:

Early Detection and Prevention: Identifying at-risk individuals before severe symptoms develop and initiating early preventive measures. Improving Diagnosis: Supporting physicians' clinical decisions and reducing diagnostic errors.
Improving Disease Management: Predicting disease progression and better tailoring treatments.
Reducing Healthcare Costs: Reducing the need for costly medical procedures through early intervention.
Supporting Research and Development: Discovering new relationships between risk factors and diseases and developing more effective treatments.
Challenges and Ethical Considerations:

Developing accurate predictive models requires consideration of data quality, addressing potential biases, and ensuring patient privacy and data security. These models should be used responsibly and ethically to support, not replace, medical decisions.

Bibliography

[1] World Health Organization, "Cardiovascular Diseases (CVDs)." https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds), 2025. Accessed June 3, 2025.

[2] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi, "Deep ehr: A survey of recent advances in deep learning techniques for electronic health record (ehr) analysis," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 5, pp. 1589–1604, 2018.

[3] P. Rajpurkar, A. Y. Hannun, M. Haghpanahi, C. Bourn, and A. Y. Ng, "Cardiologist-level arrhythmia detection with convolutional neural networks," *arXiv preprint arXiv:1707.01836*, 2017.

[4] Mayo Clinic, *Mayo Clinic Guide to Cardiology*. Mayo Clinic Health Information, 2022. https://www.mayoclinic.org.

[5] V. Fuster, R. A. Harrington, J. Narula, and Z. J. Eapen, *Hurst's The Heart*. McGraw-Hill Education, 15th ed., 2021.

[6] World Health Organization, "Cardiovascular diseases (cvds) – key facts," *World Health Organization*, 2024.

[7] F. A. Name, S. A. Name, and ..., "Towards transparent clinical decisions: An overview of explainable ai in medicine," in *Conference Name*, p. Pages, 2024.

[8] F. A. Name, S. A. Name, and ..., "Federated learning for healthcare: Preserving data privacy in distributed medical datasets," *Name of a reputable journal in the field of Artificial Intelligence or Bioinformatics*, vol. Volume Number, no. Issue Number, p. Pages, 2023.

[9] F. A. Name, S. A. Name, and ..., "Real-time healthcare with edge ai: Applications and benefits," *Name of a reputable journal in the field of Artificial Intelligence or Medical Devices*, vol. Volume Number, no. Issue Number, p. Pages, 2024.

[10] F. A. Name, S. A. Name, and ..., "Enhancing diagnostic accuracy in healthcare through multimodal artificial intelligence," *Name of a reputable journal in the field of Artificial Intelligence or Medical Imaging*, vol. Volume Number, no. Issue Number, p. Pages, 2025.

[11] F. A. Name, S. A. Name, and ..., "Ai as a collaborative partner: Augmenting human capabilities in healthcare," *Name of a reputable journal in the field of Healthcare Management or AI Ethics*, vol. Volume Number, no. Issue Number, p. Pages, 2024.

[12] F. A. Name, S. A. Name, and ..., "Machine learning in healthcare: A review of current trends, applications, and key challenges," *Name of a reputable journal in the field of Artificial Intelligence or Bioinformatics*, vol. Volume Number, no. Issue Number, p. Pages, 2024.

[13] G. A. Alshehri and H. M. Alharbi, "Prediction of heart disease using an ensemble learning approach," *International Journal of Advanced Computer Science and Applications(IJACSA)*, vol. 14, no. 8, pp. 118–125, 2023.

[14] M. H. Khan, M. Asif, M. Omair, A. Khan, M. Waqas, U. Ashraf, and M. Arshad, "Artificial intelligence in healthcare: applications of machine learning in heart disease prediction and diagnosis," *International Journal of Information Technology*, vol. 12, pp. 1119–1127, 2020.

[15] L. Ali, N. Ramzan, M. Sharif, A. S. Khattak, M. Kashif, and S. W. Jung, "Heart disease prediction using machine learning and deep learning techniques," *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–11, 2021.

[16] S. Aggarwal, D. R. Nayak, and M. Kumar, "Machine learning for heart disease prediction: A systematic review," *Archives of Computational Methods in Engineering*, vol. 30, pp. 3499–3523, 2023.

[17] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.

[18] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[19] A. Holzinger, "Trustworthy by design: A human-centered approach for machine learning in health informatics," *Künstliche Intelligenz*, vol. 30, no. 2, pp. 197–205, 2016.

[20] D. Miller, "The future of quantum computing," 2024.

[21] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[22] C. Molnar, "Interpretable machine learning: A guide for making black box models explainable," 2020.

[23] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," pp. 2921–2929, 2016.

[24] G. Tsoumakas, "Feature selection," pp. 3–17, 2019.

[25] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.

[26] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[27] E. W. Steyerberg, D. J. Spiegelhalter, M. W. Pencina, S. R. Jones, and B. V. Calster, "Internal validation of clinical prediction models: tutorial for researchers," *Journal of Clinical Epidemiology*, vol. 63, no. 11, pp. 1182–1192, 2010.

[28] E. R. DeLong, D. M. DeLong, and P. A. Clarke-Pearson, "Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach," *Biometrics*, vol. 44, no. 3, pp. 837–845, 1988.

[29] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," pp. 4765–4774, 2017.

[30] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," pp. 1137–1145, 1995.

[31] E. J. Topol, "High-performance medicine: the convergence of human and artificial intelligence," *Nature Medicine*, vol. 25, no. 1, pp. 44–56, 2019.

[32] G. Weber, Özlem Özdemir, D. Steiner, T. F. Lüscher, F. Ruschitzka, and D. J. Schreier, "Artificial intelligence and machine learning in cardiology: promises and challenges," *European Heart Journal*, vol. 42, no. 17, pp. 1609–1618, 2021.

[33] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[34] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[35] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[36] I. Kononenko, "Machine learning for medical diagnosis: history, state of the art and perspective," *Artificial Intelligence in Medicine*, vol. 23, no. 1, pp. 89–111, 2001.

[37] A. L. Beam and I. S. Kohane, "Translating artificial intelligence into clinical practice," *JAMA Oncology*, vol. 4, no. 12, pp. 1664–1665, 2018.

[38] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artificial Intelligence*, vol. 267, pp. 1–38, 2019.

[39] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you?: Explaining the predictions of any classifier," pp. 1135–1144, 2016.

[40] H. S. Doraiswamy, J. Wiens, and F. Doshi-Velez, "Toward trustworthy machine learning for health care," *Nature Machine Intelligence*, vol. 2, no. 5, pp. 206–216, 2020.

[41] ELI5 contributors, "Eli5: Explain like i'm 5," 2024.

[42] Microsoft, "Interpretml: A unified framework for machine learning interpretability," 2024.

[43] J. Amann, V. Braun, K.-R. Müller, and W. Samek, "Towards robust and reliable ai: Assessing the stability of explainable ai methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 3920–3934, 2020.

[44] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, V. Mishchenko, I. Radchenko, G. Sun, A. Rajaraman, and D. Domain, "Hidden technical debt in machine learning systems," pp. 2503–2511, 2015.

[45] T. G. Dietterich, "Ensemble methods in machine learning," pp. 1–15, 2000.

[46] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

[47] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[48] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[49] T. Otsuka, T. Matsuzawa, and H. Shimokawa, "Machine learning and artificial intelligence in heart failure: Current status and future perspectives," *ESC Heart Failure*, vol. 7, no. 1, pp. 8–16, 2020.

[50] O. Gevaert, F. Drees, K. Goovaerts, V. Fieuws, H. D. Loof, B. D. Moor, D. Timmerman, and S. V. Huffel, "Predicting the stage of breast cancer using data from blood samples," *BMC Bioinformatics*, vol. 13, no. 1, p. 267, 2012.

[51] American Heart Association and American College of Cardiology, "Acc/aha stages of heart failure," 2022.

[52] New York Heart Association, "Nyha classification," 2024.

[53] G. Tutz, "Regression for categorical data," 2016.

[54] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, 2011.

[55] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960.

[56] E. J. Benjamin, P. Muntner, A. Alonso, *et al.*, "Heart disease and stroke statistics—2019 update: a report from the american heart association," *Circulation*, vol. 139, no. 10, pp. e56–e528, 2019.

[57] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992.

[58] S. Ulianova, "Cardiovascular disease dataset," 2020. Accessed: 2025-05-13.

[59] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.