National College of Ireland

# Scalable and Robust Cloud-Based System for Heart Disease Prediction Using Ensemble Learning

MSc Research Project
Master of Science in Cloud Computing


Manisha Prasad
Student ID: x21231222

School of Computing
National College of Ireland


Supervisor: Vikas Sahni

**National College of Ireland**
**Project Submission Sheet**
**School of Computing**

| | |
|---|---|
| **Student Name:** | Manisha Prasad |
| **Student ID:** | x21231222 |
| **Programme:** | Master of Science in Cloud Computing |
| **Year:** | 2023 |
| **Module:** | MSc Research Project |
| **Supervisor:** | Vikas Sahni |
| **Submission Due Date:** | 14-12-2023 |
| **Project Title:** | Scalable and Robust Cloud-Based System for Heart Disease Prediction Using Ensemble Learning |
| **Word Count:** | 5037 |
| **Page Count:** | 20 |

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

**ALL** internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

| | |
|---|---|
| **Signature:** | |
| **Date:** | 14th December 2023 |

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:**

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies). | ☐ |
| **Attach a Moodle submission receipt of the online project submission**, to each project (including multiple copies). | ☐ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | ☐ |

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# Scalable and Robust Cloud-Based System for Heart Disease Prediction Using Ensemble Learning

Manisha Prasad
x21231222

**Abstract**

The proposed paper presents a comprehensive study outlining the methodology and implementation of a highly scalable and reliable system for predicting heart disease. The research leverages cutting-edge technologies, including cloud computing and machine learning, to develop a robust solution which means that it is designed to be strong and resilient, able to handle various scenarios and provide accurate predictions. The study employs popular ensemble learning techniques, training AdaBoost, Decision Tree, Random Forest, and Stacking Classifier models using Python's scikit-learn module. The system's user interface is developed using Flask for web application development, allowing users to input patient data and obtain disease predictions seamlessly. The deployment is carried out on the AWS Cloud infrastructure, utilizing services such as AWS SageMaker, EC2 instances, and Elastic Beanstalk for rapid and scalable deployment. CodeDeploy integration facilitates smooth deployment pipelines, ensuring easy application changes and maintenance. To ensure the accuracy, precision, and dependability of the generated models, rigorous testing and validation methods are carried out. Overall, this research contributes to the field of illness prediction by combining advanced technology with ensemble learning approaches and cloud capablity, offering a powerful and scalable solution for accurate assessment.

# 1 Introduction

The convergence of cloud computing and machine learning has brought in a new age for healthcare systems, dramatically increasing their efficiency. This paper examine the use of ML and cloud computing to foster a more effective healthcare environment. The combination of cloud computing and ML shows promise in simplifying numerous aspects of the present healthcare scene, such as illness prediction, work offloading, and using AWS Code pipelines with Python (Flask). Cloud computing has also received a lot of interest in healthcare system applications in recent years because of its ability to provide numerous medical services on the Internet. Cloud computing enables the delivery of infrastructure services to a wide number of stakeholders with different and constantly changing needs Abdelaziz et al. (2018) .

## 1.1 Background

The combination of ML and cloud has significantly improved the productivity of disease prediction systems in healthcare. Using the capabilities of these technologies, more effective methods of patient treatment are being created, while the costs associated with

disease prediction are being reduced. This research digs into how ML as well as cloud computing may be used to create an efficient illness prediction system. It specifically investigates how these technologies might improve the precision and speed of illness prediction while also increasing overall efficiency. Cloud computing and machine learning are two formidable and emerging technologies set to change the healthcare sector. Cloud computing makes storage of data and communication more efficient, while ML enables analysis of prediction and automated decision-making. Their confluence has the potential to produce an efficient healthcare ecosystem capable of providing improved illness detection and treatment, improving patient outcomes, and lowering costs. Chronic heart disease is a major global health problem, accounting for a significant number of diseases globally and demonstrating a high death rate, particularly in cardiovascular conditions. The major reason is frequent constriction of the arteries that provide blood to the heart and other essential organs. According to studies, heart disease is one of the most common disorders in the United States, appearing as a variety of symptoms such as shortness of breath, hypertension, obesity, edoema, acid reflux, and strokes.

## 1.2 Aim of the study

The purpose of this research is to analysis into and illustrate the potential impact and efficacy of incorporating cloud computing and machine learning technologies into healthcare systems. This study aims to demonstrate how such integration may significantly improve illness prediction accuracy, optimise healthcare application delivery, and eventually contribute to the establishment of a more efficient and refined healthcare infrastructure. The fundamental goal of this research is to evaluate how the incorporation of cloud computing improves the performance of various machine learning models for sickness prediction in the healthcare industry.

## 1.3 Research Questions

How does the integration of machine learning models with cloud computing enhance the accuracy of disease prediction compared to traditional methods, and what is the comparative performance evaluation of Adaboost, Decision Tree, Random Forest, and Stacking Classifier within a cloud-based healthcare dataset for predicting cardiac diseases through a web application?"

# 2 Related Work

## 2.1 Machine Learning for Disease Prediction

Machine Learning (ML) has transformed disease prediction in healthcare by harnessing massive datasets to improve diagnosis accuracy, prognosis, and therapy options. In this arena, ML algorithms evaluate complicated medical data, finding patterns and relationships that help in the early detection of illnesses, anticipating future hazards, and customising treatment approaches. ML techniques such as supervised learning (such as Support Vector Machines, Random Forests, and Neural Networks), unsupervised learning, and ensemble methods are applied to diverse medical datasets to develop predictive models for diseases such as cancer, cardiovascular issues, diabetes, and infectious diseases. Both Mohan et al. (2019) and Jain and Singh (2018)emphasise the importance of

feature selection in improving the accuracy of illness prediction models, underlining its critical role in healthcare informatics. The relevance of precise classification systems in diagnosing and forecasting illnesses such as breast cancer and other cancer subtypes is echoed by Laghmati et al. (2023). Laghmati et al. (2023) investigate ensemble models and their usefulness in increasing classification accuracy, whereas emphasise the varied uses of ML approaches in predicting cancer development, including ANN, BN, SVM, and DT. Uddin et al. (2019) explore the landscape of supervised ML algorithms, stressing the comparative performance of algorithms such as SVM and Random Forest, highlighting the significance of selecting proper algorithms for illness prediction investigations. As a result, these reviews emphasise the critical role of feature selection, classification accuracy, as well as the diverse application of supervised ML algorithms in improving medical diagnostics and disease prediction, all while aligning with the shared goal of improving clinical decision- making and patient outcomes in healthcare.

| Study | Focus Area | Key Methods/Techniques | Reported Results |
|---|---|---|---|
| Mohan et al., 2019 | Disease prediction and ML algorithms | Feature selection, ML algorithms | Achieved accuracy level of 88.7% for heart disease prediction |
| Jain et al., 2018 | Healthcare informatics, chronic disease prediction | Feature selection, classification | Achieved accuracy level of 53% |
| Laghmati et al., 2023 | Breast cancer classification, ML techniques | Ensemble models, feature selection | XGboost achieved over 96% recall for the Mammographic Mass dataset |
| Kourou et al., 2015 | Cancer progression modelling, ML applications | ANN, BN, SVM, DT | Detailed the application of various ML methods, no specific accuracy reported |
| Uddin et al., 2019 | Supervised ML algorithms, disease prediction | SVM, Random Forest, algorithm comparison | Random Forest outperformed in 53% of studies, achieving the highest accuracy in some cases |
| Ali et al. (2020) | Heart Disease Prediction | Ensemble Deep Learning | Achieved 98.5% accuracy in heart disease prediction, outperforming traditional classifiers. |
| Lalmuanawma et al. (2020) | Covid-19 Management | AI/ML in forecasting, contact tracing, drug dev. | Demonstrated AI/ML's role in Covid-19 management, emphasizing forecasting and drug development. |
| Motwani et al. (2017) | Cardiovascular Imaging (5-year ACM prediction) | Machine Learning (Boosted Ensemble Algorithms) | ML showed higher predictive accuracy for 5-year all-cause mortality compared to traditional metrics. |
| Senders et al. (2018) | Neurosurgical Outcome Prediction | Machine Learning (ML) for outcome prediction | ML's superior performance over conventional methods in predicting neurosurgical outcomes. |
| Alaa et al. (2019) | Cardiovascular Disease Risk Prediction | Automated ML framework (AutoPrognosis) | AutoPrognosis significantly improved CVD risk prediction compared to traditional approaches. |

Figure 1: Comparison Table of Machine Learning for Disease Prediction

Ali et al. (2020), Lalmuanawma et al. (2020), and Motwani et al. (2016)underscore ML's potential in enhancing predictive models, showcasing its superiority over conventional methods. Specifically, Ali et al. (2020) emphasize the use of ensemble deep learning for heart disease prediction, akin to Motwani et al. (2016) approach in predicting all-cause mortality using boosted ensemble algorithms. Both studies employ ML to surpass existing clinical risk scores, revealing the efficacy of incorporating ML in cardiovascular risk assessment. Lalmuanawma et al. (2020) similarly advocates for ML and AI technologies in combating the Covid-19 pandemic, emphasizing their role in forecasting, contact tracing, and drug development—a parallel toAlaa et al. (2019) use of ML in predicting outcomes based on extensive parameters from coronary computed tomographic angiography (CCTA). Furthermore, (Senders et al., 2018) highlight ML prowess in neurosurgical outcome prediction, mirroring Motwani et al.'s demonstration of ML's superiority over conventional metrics in predicting all-cause mortality, indicating ML's applicability across medical domains for outcome prediction and prognosis assessment.

## 2.2 Hybrid Approach for Disease Prediction

To improve illness prediction accuracy, Kavitha et al. (2021) and offer hybrid techniques that combine diverse machine learning algorithms such as Random Forest and Decision Trees, and Support Vector Machines with Bootstrap bagging, respectively. For enhanced illness prediction models, advocate for unique hybrid models that integrate diverse methods like as Random Forest with Multivariate Adaptive Regression Splines (MARS) and C4.5 with PRISM learners, respectively. Both Sarkar and Sana (2019) and emphasise the necessity of generic models in illness prediction to overcome the constraints of disease-specific systems. For forecasting skin illnesses and heart diseases, Verma et al. (2020) advocate the use of hybrid methods that include different feature selection techniques and varied machine learning algorithms. Similarly,Abdeldjouad et al. (2020) and Haq et al. (2018) investigate the effectiveness of several machine learning algorithms in predicting cardiovascular disorders, with an emphasis on comparing classifier performance and refining prediction models. Chen et al. (2017)distinguish itself by using a hybrid graph- based recommendation system for miRNA-disease association prediction. The common thread in these studies is the use of various algorithms, feature selection methods, and cross-validation techniques to improve prediction accuracy and efficiency for various diseases, addressing the limitations of traditional diagnostic methods by embracing computational approaches for improved disease prediction and understanding.

## 2.3 Cloud Computing in Heart Disease

Advances in healthcare technology have transformed patient care in recent years, notably in the detection and management of cardiac disorders. The insidious nature of cardiac diseases, which are frequently undiagnosed without advanced technologies, has prompted novel techniques that combine machine learning and cloud computing.Venkatesan et al. (2018) described HealthCloud, a system that integrates machine learning and cloud computing to monitor the health of cardiac patients. Desai et al. (2022) created a system called HealthCloud that focuses on predicting heart disease by assessing several machine learning algorithms using Quality of Service criteria. Gupta et al. (2017) investigated the use of ensemble models to forecast cardiac illnesses using machine learning and cloud computing, reaching impressive accuracy rates.Meanwhile, Muhammad Adnan Khan (2020) suggested a cloud-based approach for heart disease prediction that used Support Vector Machine (SVM) algorithms and achieved a stunning 93.33 percent accuracy.

# 3 Methodology

## 3.1 Overview of Proposed Methodology

The primary goal of this research is to automate heart disease diagnosis by creating a web application interface capable of forecasting the possibility of heart disease based on several contributory variables. After that, the web application is safely deployed to the cloud. The implementation of this report is divided into four key phases: first, a traditional method for data collection and pre-processing is used; second, Amazon SageMaker is used to train the system model; third, the system model is rigorously tested using four distinct machine learning-based algorithms; and finally, the web application is deployed on the cloud using Flask. Notably, Flask acts as the interface, providing safe patient

data storage within the cloud environment. The proposed research's architectural flow is visually depicted in Figure 2, illustrating the sequential progression of these crucial phases.

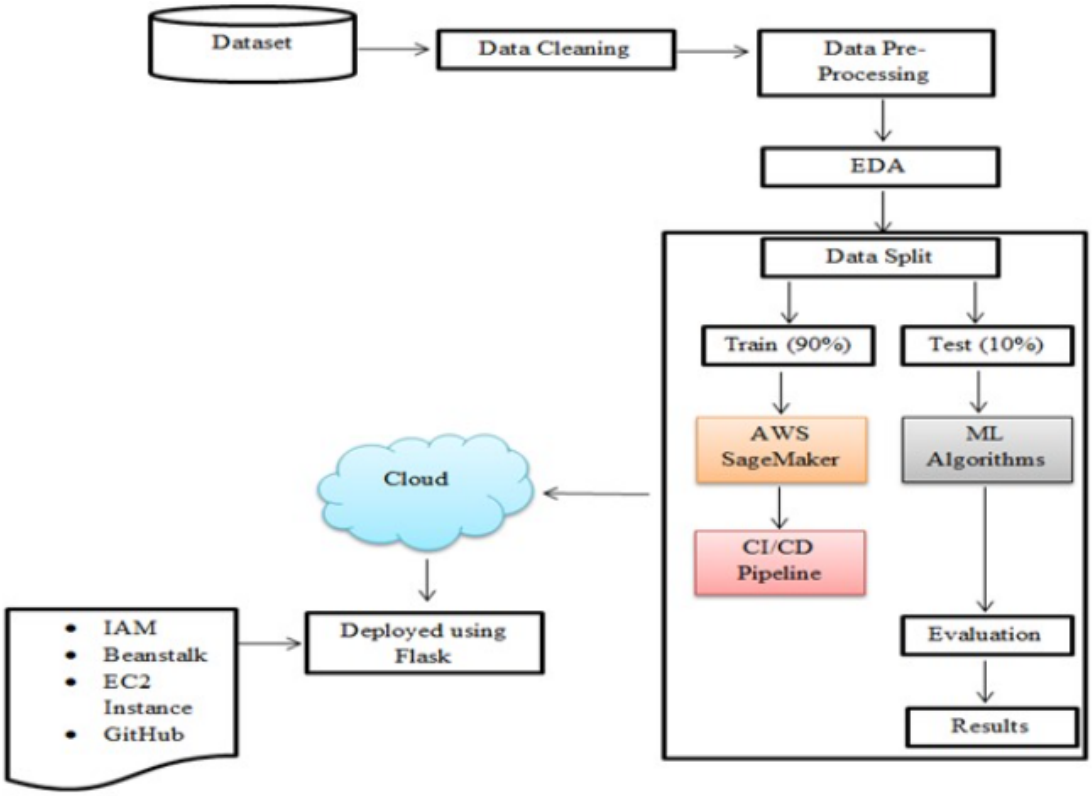


Figure 2: Research Architectural Flow

## 3.2 AWS Sage Maker

Sage Maker on Fig [3] AWS SageMaker is a machine learning service based on Amazon that builds and deploys algorithms based on the basis of a proposed system. It is utilised in the training phase before being moved to a virtual machine where the data is stored and immediately deployed to the cloud. SageMaker integration is largely accomplished through the use of a Jupyter notebook, in which data sources are accessible,

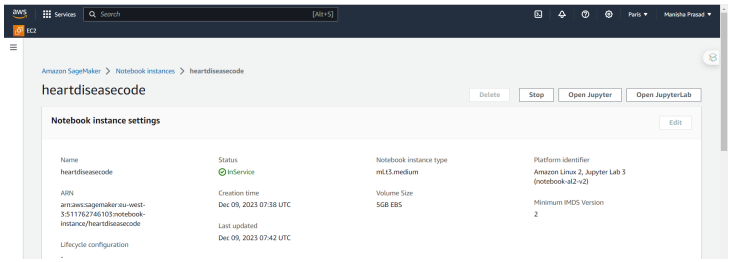


Figure 3: AWS Sage Maker

investigated, and analysed. In addition to this, it includes a number of machine learning-based algorithms that can function efficiently in the cloud's distributed environment. A SageMaker-based framework aids in the customization of algorithms based on the needs of the system model, hence adjusting the workflow.

## 3.3 List of Models

Several List of Models given below in Fig[4]:

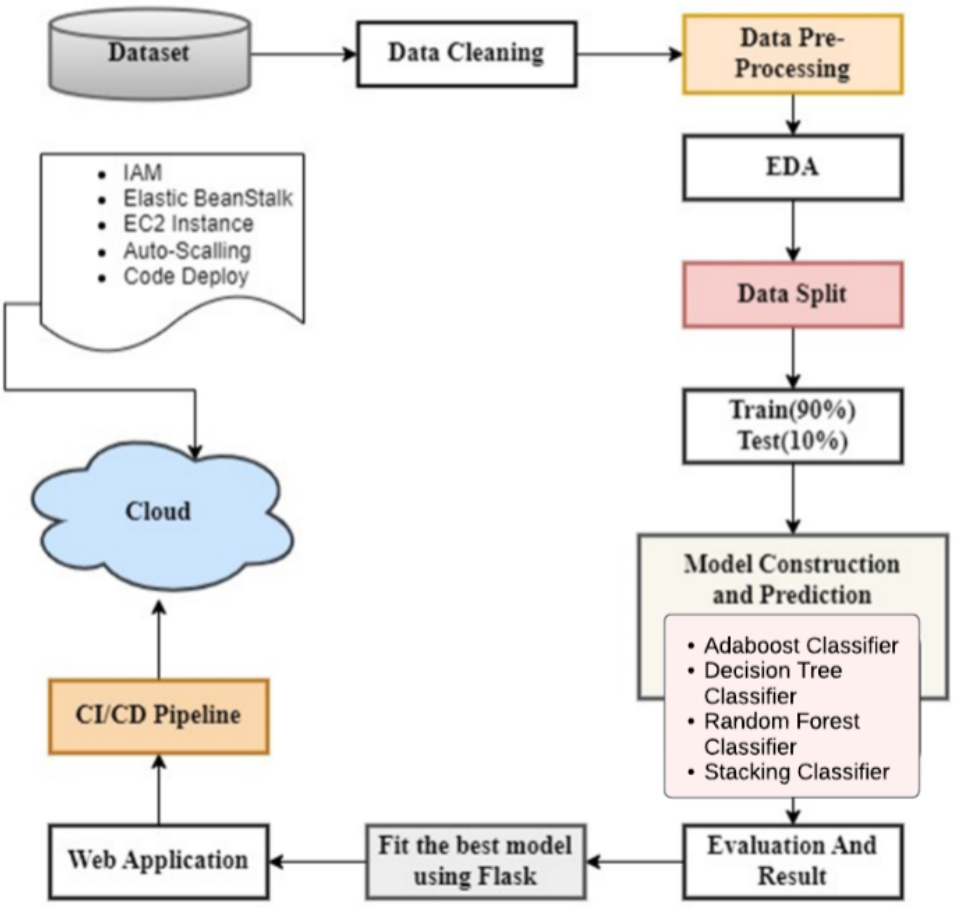


Figure 4: Research Architectural Flow with ML Models

1. AdaBoost Classifier: AdaBoost is a form of ensemble learning that combines numerous weak learners to build a strong classifier. It may efficiently use weak predictive variables in illness prediction to improve overall prediction accuracy. Because of its capacity to adapt and improve prediction based on earlier misclassifications, it is useful in illness prediction models, assisting in the correct detection of patterns associated with various health issues.

2. Decision Tree Classifier: Decision trees divide data into hierarchical structures and make judgments based on the values of characteristics. They are intuitive and convey the significance of features. They are beneficial in healthcare because they give interpretable principles for illness prediction. They provide information on crucial elements that influence illness incidence or development.

3. Random Forest Classifier: A Random Forest is an ensemble of decision trees that leverages multiple trees for predictions, reducing overfitting and enhancing accuracy. Its ability to handle high- dimensional data and feature importance evaluation is crucial in disease prediction, enabling the identification of relevant features for accurate predictions.

4. Stacking Classifier (Best Model): Stacking Classifier combines many classifiers to generate a meta- classifier that improves prediction accuracy. Combining Random Forest and AdaBoost as foundation models, then employing a Decision Tree as a meta-classifier, aids in harnessing the strengths of varied models for more robust illness prediction in healthcare.

# 4 Design Specification

The design specification describes the project's architecture and tools, with an emphasis on Python and AWS services such as Amazon Sagemaker, scikit-learn, Elastic Beanstalk, EC2 instances, Code Pipeline, and Code Deploy. The system design includes Python-based machine learning techniques integrated into the Amazon Sagemaker environment, with model construction and training handled by scikit- learn. AWS technologies such as Elastic Beanstalk make application deployment and scaling easier, while EC2 instances offer the computing capacity required for processing. Continuous integration and deployment are made easier using Code Pipeline and Code Deploy. The design specification specifies how these tools will be integrated to construct a robust and scalable system capable of deploying machine learning models, controlling processes, and assuring optimal performance within the AWS environment.

The fundamental elements and technological foundations of the proposed solution, combining sophisticated methodologies, architectures, and frameworks for robust implementation. It identifies and articulates the requirements that drive the solution's design holistically. For scalable and efficient model training and deployment, an emphasis is made on exploiting cutting-edge technology, such as cloud computing infrastructure, such as AWS SageMaker. The disease prediction system is built on an array of machine learning models, most notably the AdaBoost Classifier, Decision Tree Classifier, Random Forest Classifier, and Stacking Classifier. This research recommends that healthcare systems deploy a cloud and edge computing architecture.

AWS and CI/CD is used in the setup using Git, Flask for web application development, and ML techniques for disease prediction. The implementation incorporates Flask for web application development, allowing for an easy-to-use user interface for obtaining illness prediction capabilities. The use of technologies such as AWS services (for example, AWS SageMaker, Elastic Beanstalk, Cloud9 and EC2 instances) and Python-based libraries (for example, scikit-learn) provides smooth development, model training, and deployment procedures. Design specification of the proposed research work also includes terraform that work as Infrastructure-as-Code (IaC) for creating the web application hosting environment for hosting the web application. In this proposed work, Elastic Beanstalk service offered by the Amazon Web Services is provisioned for hosting the web application. Additionally, it includes GitHub repository that act as source control mechanism for flask based web-application. Another AWS service incorporates in the design specification of the proposed work is AWS code pipeline. AWS code pipeline plays a vital role in the process of deployment of the web application. It helps to provide connectivity between source control repository that is GitHub with AWS elastic Beanstalk environ-

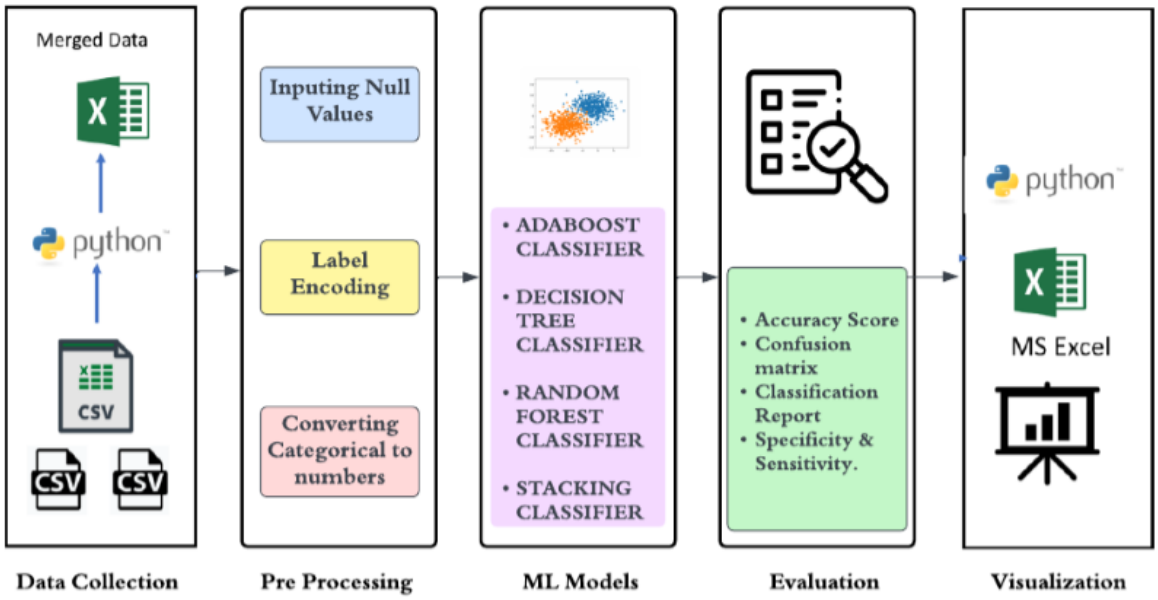


Figure 5: Flow Diagram

ment. Furthermore, a full description of the built solution's final step is provided, which includes the chosen ensemble models, their integration inside the Flask-based web application, and their deployment on the AWS Cloud infrastructure. This complete design specification integrates cutting-edge technology with sophisticated ensemble learning approaches, resulting in a robust and scalable system suited to successfully solve the illness prediction .

# 5 Implementation

It starts with a brief overview of the datasets which have been obtained from Kaggle https://archive.ics.uci.edu/dataset/45/heart+disease Janosi and Detrano (1988). The next part describes the method of data splitting in the ratio of 90 to 10, followed by a complete explanation of cloud-based system deployment and then all four models. The Implementation chapter delves into the practical implementation of the suggested solution, covering the numerous processes from model development to deployment. To guarantee interoperability with machine learning models, the procedure begins with data preparation, which includes resolving missing values, encoding categorical categories, and scaling numerical data. The utilisation of technology such as cloud computing and machine learning may enable the healthcare business to create a highly scalable and dependable solution. The implementation incorporates Flask for web application development, allowing for an easy-to-use user interface for obtaining illness prediction capabilities. The use of technologies such as AWS services (for example, AWS SageMaker and EC2 instances) and Python-based libraries (for example, scikit-learn) provides smooth development, model training, and deployment procedures. Furthermore, a full description of the built solution's final step is provided, which includes the chosen ensemble models, their integration inside the Flask-based web application, and their deployment on the AWS Cloud infrastructure.

This complete design specification integrates cutting-edge technology with sophisticated ensemble learning approaches, resulting in a robust and scalable system suited to successfully solve the illness prediction research topic.

The AdaBoost, Decision Tree, Random Forest, and Stacking Classifier ensemble models are rigorously trained using Python's scikit-learn module. To improve model performance and ensure accurate illness prediction, hyperparameter adjustment is undertaken. Flask integration enables the creation of an intuitive web application by offering an accessible interface for users to input patient data and generate disease predictions based on learned models. Deployment occurs on AWS Cloud infrastructure, taking advantage of services such as AWS SageMaker, EC2 instances, and Elastic Beanstalk for rapid and scalable deployment. CodeDeploy integration simplifies deployment pipelines, allowing smooth application changes and maintenance. Before implementation, rigorous testing and validation methods ensure the model's accuracy, precision, and dependability, assuring its usefulness in real-world healthcare settings.

## 5.1 Dataset Description

The dataset consists of 76 attributes capturing various patient-related factors focused on heart disease assessment, though experiments generally concentrate on a subset of 14 attributes. The dataset primarily leverages the Cleveland database, extensively used in machine learning research. The "goal" attribute signifies the presence of heart disease, ranging from 0 (absence) to 4. ML experiments often focus on distinguishing presence (values 1, 2, 3, 4) as a single value 1 and absence (value 0). The columns include patient ID, age, study origin, gender, chest pain type (with subcategories like typical angina, atypical angina, etc.), resting blood pressure, serum cholesterol levels, fasting blood sugar status, resting electrocardiographic results (with categories like normal, ST-T abnormality, etc.), maximum heart rate achieved, exercise-induced angina, ST depression induced by exercise relative to rest, the slope of peak exercise ST segment, number of major vessels coloured by fluoroscopy, thalassemia indicators (normal, fixed defect, reversible defect), and the predicted attribute (0 or 1 denoting presence or absence of heart disease). This dataset, with a focus on heart disease assessment, provides a rich set of attributes for predictive modelling and analysis within the context of cardiac health assessment.

## 5.2 Data Split

The collected dataset is separated into 90 and 10 for training and testing purposes, respectively, for the sake of implementation. However, it is vital to highlight that the dataset is trained using Amazon SageMaker, and the dataset is tested using the ML algorithms that were chosen. Four machine learning algorithms are employed. In the last stage, after dataset testing, the web app designed to identify disease of heart which is deployed on the cloud with the use of AWS Flask.

## 5.3 Data Visualization

Figure 6 presents an age distribution output screen with ages ranging from 30 to 70, suggesting that there is a specific visual representation, maybe a histogram, bar chart, or any other graphical representation labelled as Figure 6 in the project or analysis

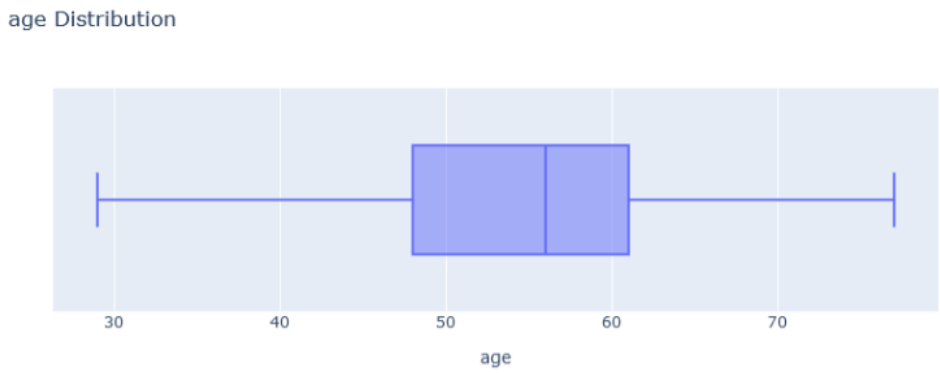


Figure 6: Age Distribution of Individuals (Ages 30-70)

documentation. This graph depicts the distribution of ages within a dataset or setting, with an emphasis on persons aged 30 to 70 years.

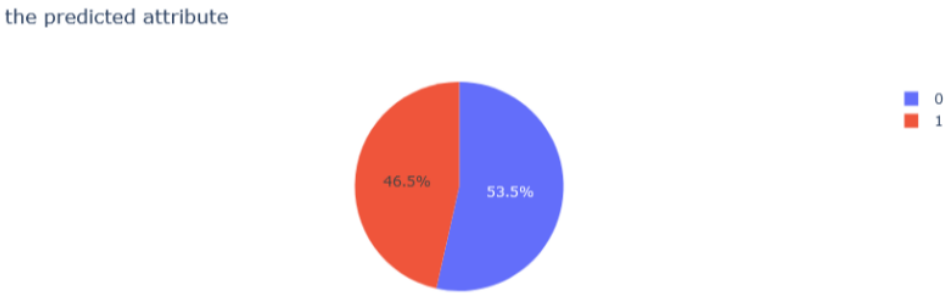


Figure 7: Pie chart- Distribution of Predicted Attributes Denoting presence or absence of heart disease

Figure 7 shows a pie chart displaying the distribution of expected features or classes (in terms of index 0 and 1) within a dataset. The pie chart is broken into pieces that indicate two groups or classes: "red (index=1)" and "blue (index=0)" The "red" category comprises roughly 46.5 per cent i.e absence of heart disease of the pie chart in this representation, while the "blue"; category occupies the remaining 53.5 per cent i.e patient suffering form the heart disease.

Figure 8 depicts a graphical depiction of the association between the several forms of chest pain(asymptomatic, atypical angina, non-anginal, and typical angina) shown on the x-axis and the maximal heart rate reached (thalch) displayed on the y-axis. This depiction most likely uses a scatter plot or a similar graphical method to show how the various forms of chest discomfort connect to the highest heart rate obtained by individuals within the defined range.

Figure 9 depicts a pie chart that depicts the average resting blood pressure for various categories of resting electrocardiographic results:"normal," shown in red and accounting for approximately 33 per cent of the chart; "LV hypertrophy"; shown in blue and accounting for approximately 33.3 per cent; and "ST-T abnormality" shown in green and accounting for approximately 34.8 per cent of the chart. A pie chart is used in this visualisation to show the distribution of average resting blood pressure across different electrocardiographic results. The percentage value of each segment represents the relative

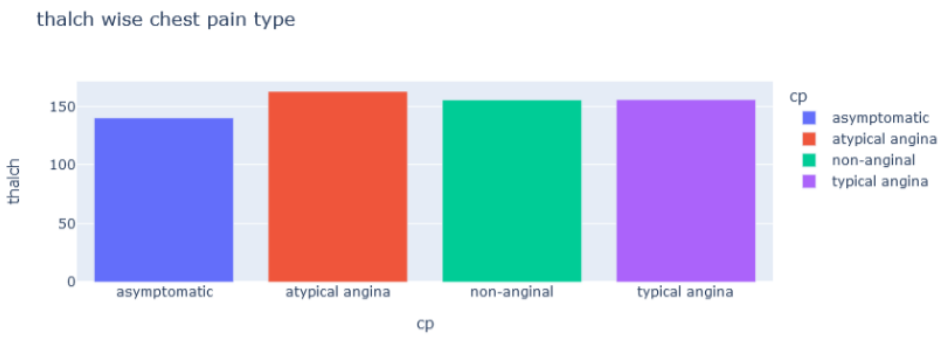


Figure 8: Relationship Between Chest Pain Types and Maximum Heart Rate

contribution of each electrocardiographic category to the total average resting blood pressure. The purpose of this graph is to show the link between resting electrocardiographic findings and their related average resting blood pressure values, as well as to provide an overview of blood pressure averages across different electrocardiographic categories.

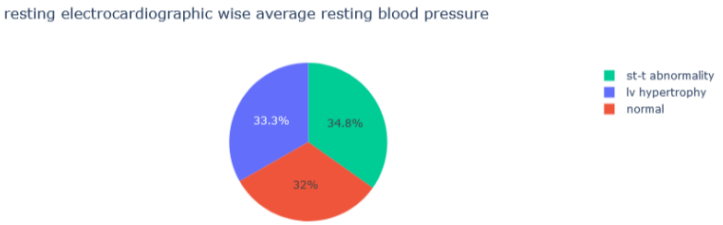


Figure 9: Pie chart- Average Resting Blood Pressure across Resting Electrocardiographic Categories

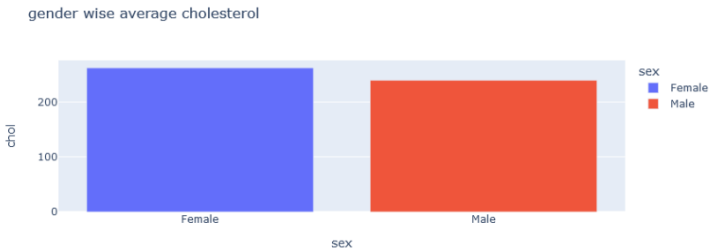


Figure 10: Graphical Representation of the average cholesterol levels by gender.

Figure 10 depicts a graphical representation of the average cholesterol levels by gender. The figure's x-axis displays genders, namely "male" and "female" which are visually separated by colours, most likely "red" for men and "blue" for females. The y-axis, on the other hand, goes from 0 to 200 and indicates the cholesterol levels in this dataset. To compare the average cholesterol levels of boys and females, this visualisation might use a bar chart, a grouped bar chart, or a similar approach.

Figure 11 depicts a histogram of age distributions classified by the presence or absence of heart disease. Ages ranging from 20 to 90 years are indicated along the x-axis. The

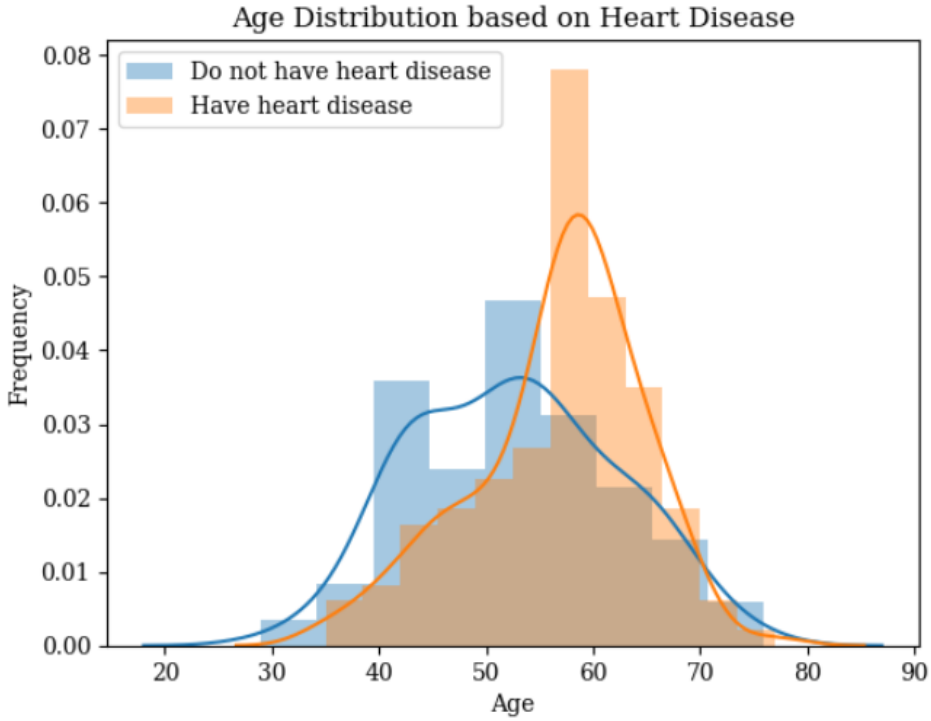


Figure 11: Histogram- Age Distribution by Heart Disease Status

colours used in the image distinguish the groups: "blue" symbolises people who do not have heart disease, while "sand colour" depicts those who do, while "skin colour" also depict the person who have a heart disease.

## 5.4 Adaboost Classifier Model

Due of the different sizes and distributions across the characteristics in this project, the dataset underwent a critical preprocessing phase prior to applying the machine learning models. To remedy this, the 'StandardScaler' function from the Python package 'scikit-learn' was used. This function standardises the data by removing the mean and scaling to unit variance, resulting in a mean of 0 and a standard deviation of 1. This normalisation technique was critical to bring all features to a comparable size and avoid characteristics with greater ranges from dominating model training. Following data normalisation, the AdaBoost Classifier model was used to predict illness. AdaBoost is an ensemble learning strategy that creates a strong learner from a group of weak learners. It works iteratively, concentrating on examples that were incorrectly identified in prior rounds to increase overall accuracy. The 'AdaBoostClassifier' class from 'sklearn.ensemble' module in Python's scikit-learn library aided in the use of AdaBoost in this implementation. By default, the 'AdaBoostClassifier' incorporates decision trees as weak learners, but it may also support alternative base estimators. AdaBoost adds greater weights to misclassified cases over time, allowing succeeding models to focus on successfully predicting such occurrences. This iterative procedure is repeated until the stated number of boosting iterations is attained or there is no further increase in prediction accuracy.

## 5.5 Decision Tree Classifier Model

The Decision Tree Classifier, a fundamental machine learning algorithm, is employed in this project for disease prediction owing to its interpretability and effectiveness in handling non-linear relationships within data. This model functions by recursively splitting the dataset based on attribute values to produce a tree-like structure, as implemented by the 'DecisionTreeClassifier' class from the 'sklearn.tree' module in Python's scikit-learn library. At each node, the algorithm chooses the most discriminative characteristic to split the data, utilising metrics such as Gini impurity or information gain to optimise homogeneity among the resultant subsets. This sequential splitting process is repeated, resulting in a tree with leaf nodes representing the final predictions. Because of the model's simplicity, decision rules may be easily interpreted, offering insights into feature significance and linkages, which is critical in healthcare contexts for identifying variables impacting illness incidence.

## 5.6 Random Forest Classifier Model

This model functions by recursively splitting the dataset based on attribute values to produce a tree-like structure, as implemented by the 'DecisionTreeClassifier' class from the 'sklearn.tree' module in Python scikit-learn library. At each node, the algorithm chooses the most discriminative characteristic to split the data, utilising metrics such as Gini impurity or information gain to optimise homogeneity among the resultant subsets. This sequential splitting process is repeated, resulting in a tree with leaf nodes representing the final predictions. Because of the model's simplicity, decision rules may be easily interpreted, offering insights into feature significance and linkages, which is critical in healthcare contexts for identifying variables impacting illness incidence.

## 5.7 Stacking Classifier Model

The Stacking Classifier was chosen as the major model in this project's illness prediction framework and the best model because to its combination of varied models for improved predictive accuracy. Using a Decision Tree as the meta-classifier, this ensemble model integrates the predictions of many base classifiers, especially Random Forest and AdaBoost in this implementation. The individual predictions of the base classifiers are contributed, and the meta-classifier combines these outputs to construct a final prediction, using the collective expertise of multiple models for more robust and accurate illness prediction. The Stacking Classifier uses the benefits of each base model inside its framework, compensating for their unique limitations and developing a more complete and accurate prediction model.

## 5.8 Deployment on Cloud

Deployment occurs on AWS Cloud infrastructure Fig [12], taking advantage of services such as AWS SageMaker, EC2 instances, and Elastic Beanstalk for rapid and scalable deployment. CodeDeploy integration simplifies deployment pipelines, allowing smooth application changes and maintenance. The provision of AWS infrastructure such as Elastic Beanstalk Fig[13] is achieved with the help of terraform and AWS cloud9. Cloud9 is a cloud based Integrated Development Environment (IDE) offered by the AWS. Whereas,

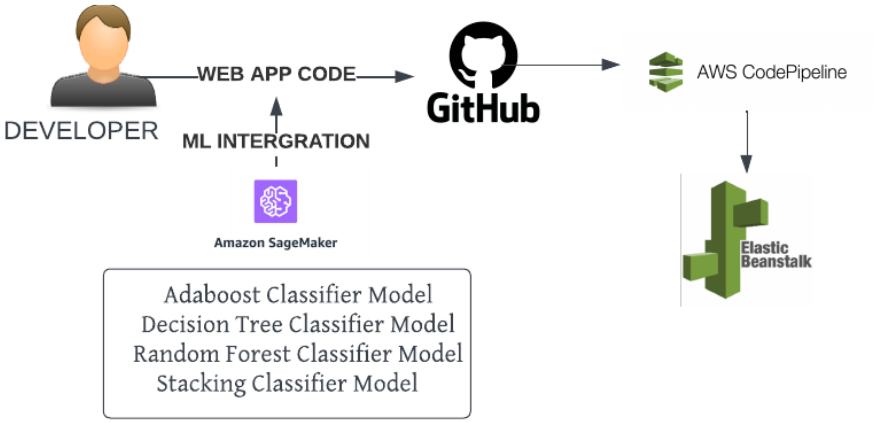


Figure 12: CLOUD ARCHITECTURE

terraform also known as Infrastructure-as-Code (IaC) is a scripting tool used to automate the cloud-based resource provisioning process. AWS Elastic Beanstalk environment provisioned with the collaboration of AWS cloud9 and terraform is used as a hosting environment for flask-based web application. Furthermore,the GitHub repository included in the proposed work implementation act as a source control management. This repository contains source code of the flask-based web application.Implementation also includes AWS code pipeline that works with GitHub in order to deploy the web application in the AWS Elastic Beanstalk environment. AWS code pipeline is a deployment service offered by the AWS to automate the deployment process. GitHub repository used in the implementation is configured with a webhook which automatically trigger the CI/CD pipeline created in the AWS code pipeline whenever new commit is occurred in the repository.

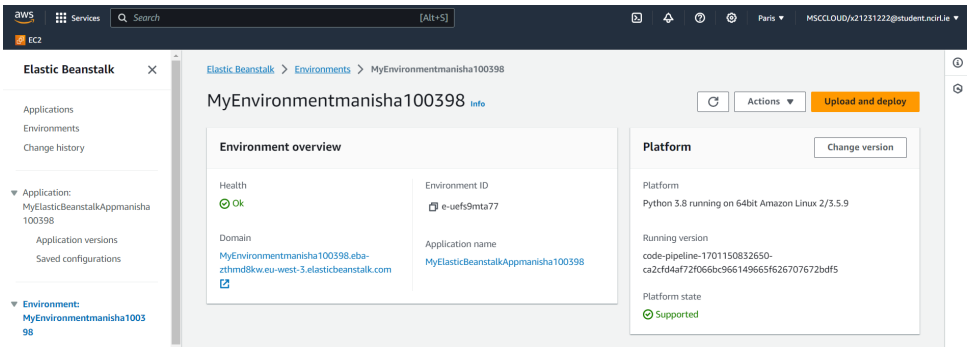


Figure 13: AWS ElastickBeanstalk

# 6 Evaluation

Upon successful deployment and implementation, the evaluation of the Adaboost Classifier, Decision Tree Classifier, Random Forest Classifier, and Stacking Classifier models encompasses a comprehensive suite of evaluation metrics to ensure robust assessment. The

evaluation matrices applied include Accuracy Score, offering an overall measure of correct predictions across all classes; Classification Report, providing insights into precision, recall, and F1-score for individual classes, enabling a detailed understanding of model performance per class; Specificity, representing the models ability to correctly identify true negatives among all negatives; and Sensitivity, depicting the models proficiency in correctly identifying true positives among all actual positives. These matrices offer a multifaceted view of each models predictive performance, addressing different aspects of accuracy, precision, and sensitivity across diverse classes within the disease prediction context.

## 6.1 Evaluation Matrix for Adaboost Classifier Model

The Accuracy Score for the Adaboost Classifier Model, quantified at approximately 0.8333 or 83.33% , represents Fig [14] the proportion of correctly predicted instances among the total instances evaluated. This accuracy score indicates that the model accurately predicted the presence or absence of the disease in roughly 83.33% of the cases within the dataset.

Table 1: Class-wise Sensitivity and Specificity Evaluation

| Class | Sensitivity | Specificity |
|---|---|---|
| 0 | 0.857143 | 0.812500 |
| 1 | 0.812500 | 0.857143 |

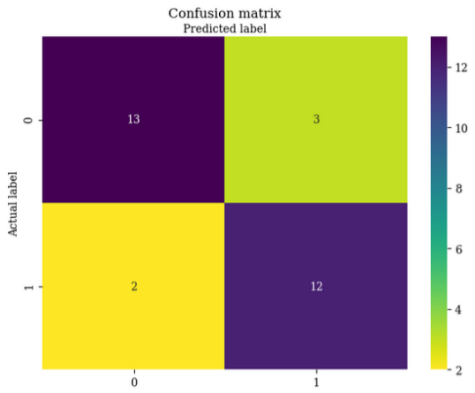


Figure 14: Confusion Matrix for Adaboost Model

## 6.2 Evaluation Matrix for Decision Tree Classifier Model

An Accuracy Score of 0.8333 (or 83.33%) for the Decision Tree Classifier Model indicates that roughly 83.33% of the instances were correctly predicted by the model. This score reflects the proportion of accurately predicted instances, showcasing Fig [15]the model's overall performance in classifying both positive and negative cases within the dataset.

Table 2: Class-wise Sensitivity and Specificity Evaluation

| Class | Sensitivity | Specificity |
|---|---|---|
| 0 | 0.642857 | 1.000000 |
| 1 | 1.000000 | 0.642857 |

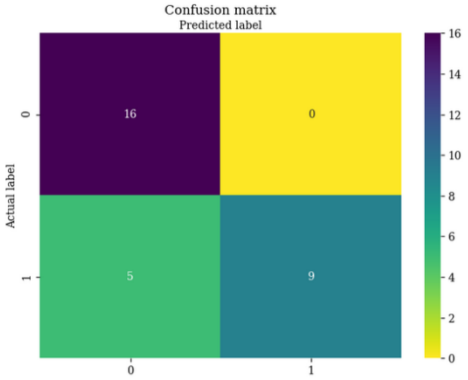


Figure 15: Confusion Matrix for Decision Tree Model

## 6.3 Evaluation Matrix for Random Forest Classifier Model

An Accuracy Score of 0.8333 (or 83.33%) for the Random Forest Classifier Model denotes that approximately 83.33% of the instances were correctly classified by the model. This metric illustrates fig [16] the overall correctness of the model's predictions across the dataset. However, while accuracy provides an essential measure of overall performance, it might not encapsulate the model's behaviour comprehensively, especially in scenarios with imbalanced classes or nuanced classification tasks.

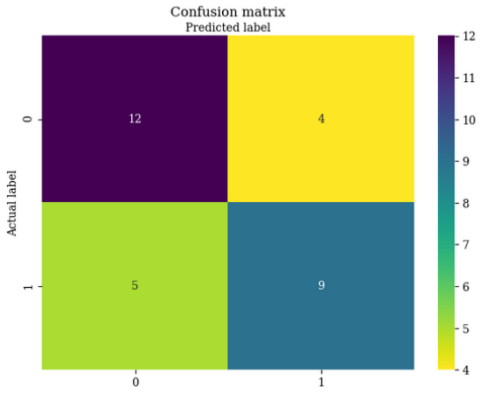


Figure 16: Confusion Matrix for Random Forest Model

## 6.4 Evaluation Matrix for Stacking Classifier Model

An Accuracy Score of 0.9 (or 90%) for the Stacking Classifier Model signifies that approximately 90% of the instances were correctly predicted by the model. This Fig [17] metric reflects the model's overall correctness in classifying instances across the dataset.

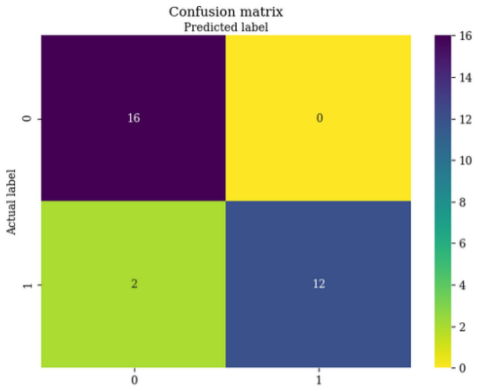


Figure 17: Confusion Matrix for Stacking Model

A higher accuracy score suggests a stronger performance in predicting both positive and negative cases within the dataset.

Table 3: Class-wise Sensitivity and Specificity Evaluation

| **Class** | **Sensitivity** | **Specificity** |
|---|---|---|
| 0 | 0.785714 | 1.000000 |
| 1 | 1.000000 | 0.785714 |

Table 4: Comparison of Machine Learning Model Accuracy Scores

| **Model** | **Accuracy Score** |
|---|---|
| Adaboost Classifier | 0.8333 |
| Decision Tree Classifier | 0.8333 |
| Random Forest Classifier | 0.8333 |
| Stacking Classifier | 0.9 |

# 7 Conclusion and Future Work

The study's conclusion emphasises the critical importance of machine learning models in disease prediction in healthcare settings. The Stacking Classifier as shown in Table [4] performed the best, with better accuracy in anticipating illness presence or absence. The importance of accurate predictions in healthcare decision-making is highlighted in this paper, with an emphasis on the possible integration of these models into clinical procedures to improve diagnostic and treatment planning. The findings support the effectiveness of ensemble learning approaches, especially in dealing with the intricacies of healthcare information. This research investigates the use of cloud computing in healthcare systems, especially the use of AWS and ML algorithms for disease prediction. The best-performing model in this study was the Stacking Classifier, which combined Random Forest and AdaBoost with a Decision Tree as a meta-classifier. Its ensemble technique displays resilience and efficacy in illness prediction, with a much better accuracy score when compared to other models, indicating intriguing prospects for real-world healthcare

applications. Despite the optimistic results, the study has inherent limitations. The size, breadth, and possible biases of the dataset may have an influence on model applicability to varied patient groups. Furthermore, the uneven distribution of classes within the dataset may have an impact on model performance, prompting additional investigation and possibly augmentation via improved sampling techniques or more extended data-gathering activities. Furthermore, the model's; practical deployment in dynamic clinical situations is limited by their dependence on retrospective data and the lack of real-time validation.

### 7.1 Future Works

Future research efforts will seek to address noted shortcomings and improve model applicability. This involves increasing dataset variety, correcting for class imbalances, and incorporating real-time data inputs for model validation and modification. Exploration of more advanced ensemble approaches, incorporation of domain-specific characteristics, and application of interpretability techniques to explicate model decision- making processes are options for improving prediction accuracy and promoting trust in model outputs in healthcare contexts. Continuous validation and engagement with medical specialists pave the path for machine learning models to be seamlessly integrated into clinical processes, eventually promoting better informed and accurate healthcare choices.

## References


Abdelaziz, A., Elhoseny, M., Salama, A. S. and Riad, A. (2018). A machine learning model for improving healthcare services on cloud computing environment, *Measurement* **119**: 117–128.
**URL:** *https://www.sciencedirect.com/science/article/pii/S0263224118300228*

Abdeldjouad, F. Z., Brahami, M. and Matta, N. (2020). A hybrid approach for heart disease diagnosis and prediction using machine learning techniques, *in* M. Jmaiel, M. Mokhtari, B. Abdulrazak, H. Aloulou and S. Kallel (eds), *The Impact of Digital Technologies on Public Health in Developed and Developing Countries*, Springer International Publishing, Cham, pp. 299–306.

Alaa, A. M., Bolton, T., Di Angelantonio, E., Rudd, J. H. F. and van der Schaar, M. (2019). Cardiovascular disease risk prediction using automated machine learning: A prospective study of 423,604 uk biobank participants, *PLOS ONE* **14**(5): 1–17.
**URL:** *https://doi.org/10.1371/journal.pone.0213653*

Ali, F., El-Sappagh, S., Islam, S. R., Kwak, D., Ali, A., Imran, M. and Kwak, K.-S. (2020). A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion, *Information Fusion* **63**: 208–222.
**URL:** *https://www.sciencedirect.com/science/article/pii/S1566253520303055*

Chen, X., Niu, Y.-W., Wang, G.-H. and Yan, G.-Y. (2017). Hamda: Hybrid approach for mirna-disease association prediction, *Journal of Biomedical Informatics* **76**: 50–58.
**URL:** *https://www.sciencedirect.com/science/article/pii/S1532046417302332*

Desai, F., Chowdhury, D., Kaur, R., Peeters, M., Arya, R. C., Wander, G. S., Gill, S. S. and Buyya, R. (2022). Healthcloud: A system for monitoring health status of heart

patients using machine learning and cloud computing, *Internet of Things* **17**: 100485.
**URL:** *https://www.sciencedirect.com/science/article/pii/S2542660521001244*

Gupta, N., Ahuja, N., Malhotra, S., Bala, A. and Kaur, G. (2017). Intelligent heart disease prediction in cloud environment through ensembling, *Expert Systems* **34**.

Haq, A. U., Li, J. P., Memon, M. H., Nazir, S. and Sun, R. (2018). A hybrid intelligent system framework for the prediction of heart disease using machine learning algorithms, *Mobile Information Systems* **2018**: 3860146.
**URL:** *https://doi.org/10.1155/2018/3860146*

Jain, D. and Singh, V. (2018). Feature selection and classification systems for chronic disease prediction: A review, *Egyptian Informatics Journal* **19**(3): 179–189.
**URL:** *https://www.sciencedirect.com/science/article/pii/S1110866517300294*

Janosi, Andras, S. W. P. M. and Detrano, R. (1988). Heart Disease, UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C52P4X.

Kavitha, M., Gnaneswar, G., Dinesh, R., Sai, Y. R. and Suraj, R. S. (2021). Heart disease prediction using hybrid machine learning model, *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, 2021 6th International Conference on Inventive Computation Technologies (ICICT), pp. 1329–1333.
**URL:** *https://doi.org/10.1109/ICICT50816.2021.9358597*

Laghmati, S., Hamida, S., Hicham, K., Cherradi, B. and Tmiri, A. (2023). An improved breast cancer disease prediction system using ml and pca, *Multimedia Tools and Applications* .
**URL:** *https://doi.org/10.1007/s11042-023-16874-w*

Lalmuanawma, S., Hussain, J. and Chhakchhuak, L. (2020). Applications of machine learning and artificial intelligence for covid-19 (sars-cov-2) pandemic: A review, *Chaos, Solitons Fractals* **139**: 110059.
**URL:** *https://www.sciencedirect.com/science/article/pii/S0960077920304562*

Mohan, S., Thirumalai, C. and Srivastava, G. (2019). Effective heart disease prediction using hybrid machine learning techniques, *IEEE Access* **7**: 81542–81554.

Motwani, M., Dey, D., Berman, D. S., Germano, G., Achenbach, S., Al-Mallah, M. H., Andreini, D., Budoff, M. J., Cademartiri, F., Callister, T. Q., Chang, H.-J., Chinnaiyan, K., Chow, B. J., Cury, R. C., Delago, A., Gomez, M., Gransar, H., Hadamitzky, M., Hausleiter, J., Hindoyan, N., Feuchtner, G., Kaufmann, P. A., Kim, Y.-J., Leipsic, J., Lin, F. Y., Maffei, E., Marques, H., Pontone, G., Raff, G., Rubinshtein, R., Shaw, L. J., Stehli, J., Villines, T. C., Dunning, A., Min, J. K. and Slomka, P. J. (2016). Machine learning for prediction of all-cause mortality in patients with suspected coronary artery disease: a 5-year multicentre prospective registry analysis, *European Heart Journal* **38**(7): 500–507.
**URL:** *https://doi.org/10.1093/eurheartj/ehw188*

Muhammad Adnan Khan, Sagheer Abbas, A. A. . A. D. H. A. M. F. K. A.-u.-R. R. A. N. (2020). Intelligent cloud based heart disease prediction system empowered with supervised machine learning, *Computers, Materials & Continua* **65**(1): 139–151.
**URL:** *http://www.techscience.com/cmc/v65n1/39558*

Sarkar, B. K. and Sana, S. S. (2019). An e-healthcare system for disease prediction using hybrid data mining technique, *Journal of Modelling in Management* **14**(3): 628–661.
**URL:** *https://doi.org/10.1108/JM2-05-2018-0069*

Uddin, S., Khan, A., Hossain, M. E. and Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction, *BMC Medical Informatics and Decision Making* **19**(1): 281.
**URL:** *https://doi.org/10.1186/s12911-019-1004-8*

Venkatesan, C., Karthigaikumar, P. and Satheeskumaran, S. (2018). Mobile cloud computing for ecg telemonitoring and real-time coronary heart disease risk detection, *Biomedical Signal Processing and Control* **44**: 138–145.
**URL:** *https://www.sciencedirect.com/science/article/pii/S1746809418300983*

Verma, A. K., Pal, S. and Tiwari, B. B. (2020). Skin disease prediction using ensemble methods and a new hybrid feature selection technique, *Iran Journal of Computer Science* **3**(4): 207–216.
**URL:** *https://doi.org/10.1007/s42044-020-00058-y*