A Project Report

on

"Heart Disease Prediction"

Submitted to
KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER SCIENCE & SYSTEM ENGINEERING
BY

Pravin Kumar Pattnaik	2128081
Sonit Kumar Swain	2128120
Soumyadeep Pal	2128093

UNDER THE GUIDANCE OF
Dr. Amiya Ranjan Panda
Dr. Manoj Kumar Mishra

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESHWAR, ODISHA - 751024
March 2024

KIIT Deemed to be University

School of Computer Engineering
Bhubaneswar, ODISHA 751024

CERTIFICATE

"Heart Disease Prediction"

submitted by

Pravin Kumar Pattnaik	2128081
Sonit Kumar Swain	2128120
Soumyadeep Pal	2128093

This certifies that the project is a record of their legitimate work completed toward partially fulfilling the requirements for the award of a Bachelor of Engineering degree (Computer Science & System Engineering) at KIIT Deemed to be University, Bhubaneswar. Under our direction, this work is completed in 2024.

Date: 10/04/2024

(Dr. Amiya Ranjan Panda)
(Dr. Manoj Kumar Mishra)
Project Guide

Acknowledgements

We are extremely preciously grateful to our tutor Dr. Amiya Ranjan Panda and Dr. Manoj Kumar Mishra for his mentorship and continuous support in ensuring that this project meets its objectives since inception until its completion.

We would also like to acknowledge the contribution of all the participants who partook in this study and gave us data which was useful. The project could not have been accomplished without their cooperation and willingness.

Moreover, We would want to thank my colleagues as well as friends for being there with me right from the start of this work till it is over. Their input has been very vital through constructive criticisms leading to improvement of this report.

Lastly, We especially indebted to our family for their unyielding encouragement, care, and understanding. They have provided us with love and motivation throughout; hence we owe a lot of appreciation to them always being around us.

This project would not have been possible without your guidance and support.

Pravin Kumar Pattnaik
Sonit Kumar Swain
Soumyadeep Pal

ABSTRACT

Cardiovascular diseases are ranked among the top causes of death globally, underlining the need for accurate and timely early identification techniques. Here, we will present a new machine learning based approach to heart disease prediction using a broad dataset that includes clinical, demographic and lifestyle factors. Our approach consists of combining feature selection techniques such as correlation analysis and recursive feature elimination with modern machine learning algorithms like Support Vector Machine, Adaboost, Random Forests, Gradient Boosting etc. to build robust predictive models. These models' accuracy, sensitivity, specificity, and area under the curves were then measured with cross validation to establish their reliability and generalizability given by AUC-ROC curve.

In conclusion, our results are promising because our model could predict with high accuracy and AUC-ROC scores on the validation sets. Our method also assists in detecting heart diseases early by identifying important risk factors and their contribution towards prediction of heart disease. This makes it possible to have personalized risk stratification based on the information gathered from those underlying factors that will reduce the need for invasive tests. When this is combined with actionable insights provided by interpretable models, then decision making at the clinical level becomes easier since these models allow transparency hence health practitioners would be guided accordingly as opposed to guesswork. In summary, using machine learning algorithms may help us achieve a proactive approach towards managing cardiovascular health leading to such positive outcomes as reduced mortality rates due to heart conditions.

Keywords: Cardiovascular disease, Machine learning, Predictive modeling, Risk assessment, Feature selection.

Contents

1. Introduction **1**

2. Basic Concepts/ Literature Review **4**
2.1. Heart Disease 4
2.2. Support Vector Machine 4
2.3. K-Neighbors 5
2.4. Gaussian-NaiveBay 6
2.5. Bernouli-NaiveBay 6
2.6. Decision Tree 7
2.7. Logistic Regression 7
2.8. Random Forest 8
2.9. AdaBoost 9
2.10. ExtraTree 9
2.11. Gradient Boosting 10

3. Problem Statement & Requirement Specifications **11**
3.1. Project Planning 11
3.2. Project Analysis 12
3.3. System Design 13

4. Implementation **14**
4.1. Methodology / Proposal 14
4.2. Result Analysis / Screenshots 15

5. Standar Adopted **21**
5.1. Design Standards 21
5.2. Coding Standards 21
5.3. Testing Standards 22

6. Conclusion and Future Scope **23**
6.1. Conclusion 23
6.2. Future Scope 24

7. References **25**

8. Individual Contributions **26**

9. Plagiarism Report **32**

List of Figures

1. **Analyzing dataset using pandas** **16**
2. **Dataset information** **16**
3. **Plot Relationship of sex having heart disease** **17**
4. **Plot Relationship between chest pain and heart disease** **17**
5. **Plot Relationship of sex having chest pain** **18**
6. **Plot Relationship of sex having BPS** **18**
7. **Histogram of Age having Cholesterol** **19**
8. **Histogram of Age having BPS** **19**
9. **Correlation Matrix** **20**
10. **Comparison table of various models** **20**

Chapter 1

Introduction

Cardiovascular diseases (CVDs) affect a large number of individuals globally, making them a major public health concern. Heart attacks, chest discomfort, and other cardiovascular diseases not only affect people's quality of life but also put a long-term burden on healthcare systems. Enhancing the results of these events and creating efficient intervention strategies are made possible through research. To evaluate patients' cardiovascular health in the past, medical professionals depended on clinical guidelines, routine risk assessments, and monitoring systems. Nevertheless, these techniques frequently lack accuracy and can miss people who require emergency care.

One potential solution to these issues has been the introduction of machine learning algorithms into health care systems in recent years. For the individual to develop and maximum complexity at diagnosis in the 19th century, this model will offer previously unheard-of opportunities for previously unheard-of opportunities for previously unheard-of prospects. The features can be examined, and machine learning algorithms can find complicated connections and patterns in the dataset that conventional statistical techniques might miss. But there are also a lot of obstacles to overcome when implementing machine learning algorithms in healthcare systems. These include concerns about data privacy, the interpretability of model predictions, and the requirement for strong certification and legal clearance. It is crucial that data scientists, regulators, and lawmakers have connections.

Machine learning has the ability to greatly improve patient outcomes and healthcare by predicting cardiovascular risk. Physicians can identify high-risk patients and promptly take preventative measures and therapies to minimize risks and improve individual outcomes. Predictive models based on machine learning also made it possible to allocate resources and take targeted action. The performance of the arrangements can be enhanced by doing this. However, problems with data quality, interpretation, and the model itself need to be resolved for these prediction models to be valid and applicable to clinical settings. All things considered, using machine learning to cardiac care is a ground-breaking tactic that has the potential to transform risk assessment, generalized medicine, and public health comparisons.

This research is an attempt to fully utilize machine learning's amazing potential for cardiovascular health. Our objective is to develop predictive models that can anticipate cardiovascular events and abnormalities by utilizing a variety of techniques, including SDVC, K-Nearest Neighbors (K-Neighbors), Gaussian Naive Bayes (Gaussian-Naïve Bayes), Bernouli-Naïve Bayes, Logistic Regression, Decision Tree, Random Forest, AdaBoost, ExtraTree, and Gradient Boosting. Our primary objective is to create predictive models that are not only accurate but also comprehensible by examining large datasets that comprise a variety of clinical characteristics, demographic information, lifestyle factors, and medical history.

Because cardiovascular disease is a complex condition with many facets, developing a nuanced approach to model development is necessary. As a result, we are focusing our efforts on developing fundamental predictive models, each specifically tailored to identify distinct cardiovascular abnormalities and issues. Our models explore the intricacies of the underlying data in an effort to provide significant new understandings of the mechanisms and risk factors related to cardiovascular disease.

Apart from our predictive abilities, we can offer significant insights into the intricate interactions between many factors that lead to cardiovascular disease. Through astute identification and correlation of the data, these frameworks can unveil novel directions for additional investigation and intervention. Additionally, our models' interpretability guarantees that the forecasts are reasonable and reliable, empowering physicians to make well-informed decisions and giving patients actionable information into their cardiovascular health. By taking a comprehensive approach, we want to increase our understanding of cardiovascular health and open the door to earlier identification, risk assessment, and more successful intervention techniques.

The relevance of our work goes beyond its immediate objectives; by leveraging machine learning systems' identification capabilities to generate new levels of prognostic evaluation and tailored therapy, we hope to spark a paradigm change in cardiovascular health. Though useful, traditional risk assessment techniques frequently rest on shaky presumptions and may miss crucial but subtle cardiovascular risk markers. To get over these constraints, we will integrate machine learning with the abundance of data that is already available to create more nuanced individual nuances.

An important aspect of the transformational potential of our work is the development of predictive models that can be used to identify specific cardiovascular events with greater accuracy Chest pain, myocardial infarction, and heart failure represent manifestations of heart failure severe and potentially life-threatening. By estimating the likelihood of these events occurring, our models enable health care providers to actively intervene to reduce risks and improve patient outcomes. Time is critical in cardiac care, and the ability to anticipate these events can save lives and reduce suffering.

Moreover, our multidisciplinary approach, which combines the domains of cardiology and machine learning, has the potential to promote collaborations between these sectors and stimulate innovation at the nexus of technology and healthcare. We hope to not only push the boundaries of cardiovascular risk prediction but also spark larger initiatives to reduce the prevalence of cardiovascular diseases worldwide by encouraging cooperation and idea sharing. In the end, we

see a world in which personalized medicine and predictive analytics come together to provide more accurate, efficient, and fair cardiovascular treatment, improving population health and well-being worldwide.

Chapter 2

Basic Concepts/Literature Review

1. Heart Disease:-

Heart complications known as cardiovascular disease, is an overall term for disorders that hit the heart or blood arteries. Annually, millions of individuals die from heart disease, making it the top global killer. While heart conditions have a variety of signs and symptoms, from arrhythmias and coronary artery disease to heart attacks, sickness, and birth several including congenital heart problems.

Multiple factors trigge-r heart disease. Modifiable- contributors involve smoking, poor nutrition, lack of exercise-, excessive alcohol intake-, and obesity. Unchangeable aspe-cts are age, gende-r, familial history of cardiac disorders, and pre-existing ailme-nts like diabetes, hype-rtension, or elevate-d cholesterol levels.

For prompt intervention and effective treatment, heart illness must be accurately and early detected. Machine learning techniques offer fascinating prospects for cardiac disease prediction through the building of predictive models employing a variety of data, including personal traits, lifestyle behaviors, medical histories, and clinical records. These models can help doctors give patients personalized treatment strategies and preventive actions by identifying those who are more prone to develop heart-related issues.

Our Python project will use machine learning techniques including SVM, K-Neighbors, Gaussian Naive Bayes, Decision Tree, Random Forest, AdaBoost, Logistic Regression and Gradient Boosting to detect and prevent heart disease early on. By using data-driven strategies, we strive to enhance patient outcomes and advance cardiovascular health in our communities.

2. Support Vector Machine:-

For task classification and prediction, the Support Vector Machine is a machine learning technique that is incredibly versatile and successful. SVM was first created for binary sorting, but it has successfully expanded to cover multiple class sorting and prediction tasks. SVM is capable of forecasting continuous results and differentiating between categories by determining the optimal hyperplane to optimize the separation between data points.

One benefit of Support Vector Machines (SVM) is that it can handle high-dimensional data well, which is perfect for applications with intricate feature spaces. SVM achieves this by employing several kernel function types, such as polynomial, linear, or radial basis function (RBF) kernels, to transform the input data into a higher-dimensional space. Through this modification, SVM can

identify complex relationships and patterns in the data that would not have been apparent in the original feature space.

One useful feature of support vector machines, or SVMs, is their resistance to overfitting, which is especially useful when working with tiny quantities of training data. In order to do this, it maximizes the distance between classes, which aids in the creation of a decision boundary that can function effectively on fresh, untested data and keeps the model from becoming overly accustomed to its training set. Furthermore, SVM gives users more control over the complexity of the model and how well it generalizes to various datasets by allowing them to include class weights and regularization parameters.

By using epsilon-insensitive loss functions, Support Vector Machine (SVM) can be applied to regression analysis tasks in addition to classification tasks. The goal of Support Vector Machines in regression is to find a hyperplane that efficiently represents the training data while minimizing the number of errors that exceed a predetermined threshold. This method enables SVM to successfully combine guaranteeing model simplicity with precise data pattern capture.

SVM offers many benefits, but it also has certain disadvantages, especially when it comes to interpretability and scalability. SVM model training can be computationally taxing, particularly for large datasets, and may need fine-tuning parameters to get the best outcomes. Furthermore, decision boundaries generated by SVM can not always be easy to understand, particularly when working with complex kernel functions or in high-dimensional environments.

Simply expressed, Support Vector Machine is an extremely adaptable and powerful machine learning algorithm that performs well when dealing with complex data and yields dependable outcomes. It is a useful tool in many different domains because to its ability to handle both regression and classification problems and to prevent overfitting.

3. K-Neighbors:-

For problems like as regression and classification, the K-Nearest Neighbors algorithm offers a straightforward yet efficient solution. Predictions are based on the 'k' nearest neighbors of data points in a feature space. The average value or majority class of those neighbors determines the new point's prediction.

KNN makes no assumptions about distributions; it just uses the training data. It is simple to carry out and works well in situations when distributions or data patterns aren't well understood. This simple method compares parametric models that require distribution assumptions. One of KNN's main advantages is its simplicity, which involves few assumptions about the underlying data.

Unlike certain algorithms, KNN does not learn specific model parameters. Rather, it retains all of the training data. KNN builds its predictions on neighboring data points and considers those data points while making predictions. Because of its adaptability, KNN can quickly adjust to new data without having to start over from scratch.
Yet, KNN has shortcomings. Storing everything and going through it all to find every forecast is inefficient and resource-intensive for huge datasets. Another problem is that the performance of KNNs is very dependent on selecting the appropriate number of neighbors (k value) and method

for determining the degree of similarity between data points.

KNN is not without limitations. However, it remains well-liked for machine learning positions. People prefer things that are straightforward and easy to understand. When you don't care about being extremely quick, it functions nicely. KNN is an effective tool for identifying non-linear patterns. It is therefore helpful for taking fresh looks at data. Comparing it to more complicated models is also a good idea. In general, KNN is a useful machine learning tool. Making forecasts and identifying trends requires a simple methodology.

4. Gaussian-Naïve Bayes:-

A probabilistic method for machine learning is the Gaussian Naive Bayes (GNB) algorithm. It focuses on tasks involving classification. A variant of the Naive Bayes classifier is called GNB. The Bayes theorem provides the basis for this classifier. With respect to the class label, it presumes that features are independent. In particular, GNB makes the assumption that feature likelihoods follow a Gaussian distribution, hence the word "Gaussian."

One major benefit of GNB is its ease of use and high computational performance. The amount of training data needed to estimate the Gaussian distribution parameters for every class is quite minimal. It can therefore be used with both small and large datasets. Moreover, the computational cost of GNB increases linearly with the quantity of features. It fits in well with high-dimensional data because of this.

Another benefit of GNB is that it manages noisy information and unnecessary features well. GNB can function rather well even in cases when the premise that features are distinct isn't fully met.

GNB is a wise choice for data sets with a lot of features or when relationships between features are intricate and challenging to moderate effectively due to its robustness.
However, a significant drawback of GNB is that it strongly assumes feature independence. In actuality, features are frequently correlated to a certain extent, which may negatively impact GNB's performance. Additionally, when class distributions are unbalanced or when each class's distributions differ greatly from a conventional bell curve, GNB may have trouble.

GNB (Gaussian Naive Bayes) has a few shortcomings. Nonetheless, it's a well-liked choice for text and opinion classification. It is useful for machine learning because of its simplicity, speed of computation, and aptitude for large data. It serves mostly as a benchmark for comparison with intricate algorithms. Thus, GNB might not succeed on every mission. However, it is able to handle a wide variety of categorization challenges because to its intuitive design and sound reasoning.

5. Bernouli-Naïve Bayes:-

The Bernoulli Naive Bayes classification algorithm is used. It is one of the algorithms of the Naive Bayes group. It is intended for jobs involving binary categorization. In these challenges, features have values of 0 or 1. Despite its simplicity, Bernoulli Naive Bayes works well. It is helpful for sentiment analysis and text categorization. Every feature in this algorithm is a binary random variable. It shows if a term (such as a word) is present or not in a sample or document. The target class's probability is determined by the model. Each feature's presence or absence is taken into account. Given the class, it is presumed that features are conditionally independent.

Extremely efficient is Bernoulli Naive Bayes. Large datasets with several features can be handled by it. It is therefore excellent for text difficulties. The algorithm requires a small amount of processing power. With sparse binary features, it performs admirably.
However, Bernoulli Naive Bayes is not flawless. Given the class, it is assumed that features are independent. In practice, this presumption might not always hold true. Concerns may also arise from correlated traits. Unbalanced data presents another possible difficulty.

Although simple to use, Bernoulli Naive Bayes is an effective method for binary applications such as text analysis. With a lot of features, it functions nicely. It manages complex data well and is quick and simple to build. But take into account its limitations and presumptions in relation to actual problems.

6. Decision Tree Classifier:-

Decision trees are intelligent instruments that can handle both number guessing and yes/no questions. They divide the information like a pizza. In order to break up the data into smaller portions, each slice verifies a feature. The tree continues to cut until it determines the answer. It resembles a map with a trail of yes/no choices that points you in the right direction.

The ease of comprehension of decision trees is one of their main advantages. Their methodical approach resembles a sketch, with each step verifying a feature before moving on to the next. The explanation for the tree's response can be found in this drawing. That is really useful in situations where you need to understand the how and why, such as at a bank or doctor's office.

Decision trees are robust models that are not impacted by anomalies. They don't require any preprocessing to handle numerical and categorical inputs with ease. Numerous Trees are built by their ensemble forms, such as Random Forests and Gradient Boosting. They become less susceptible to overfitting as a result, increasing predictive power.

Nevertheless, Decision Trees encounter variance problems, particularly on unbalanced, noisy data sets. Their decisions may be unjustly skewed by features having multiple values. Deep, complex trees run the risk of becoming overgrown if regularization or pruning are not used sparingly.

Decision trees are still the go-to solution for many applications despite these drawbacks because of their interpretability, versatility, and ease of use. They are easily adaptable to handle various jobs and data kinds, and they serve as the foundation for more intricate ensemble systems. All things considered, decision trees are an effective machine learning tool that are neither overly complex to comprehend nor overly inadequate in terms of prediction accuracy.

7. Logistic Regression:-

Machine learning relies heavily on logistic regression. This technique predicts if data is part of a certain group. Despite being referred to as "regression," it actually classifications data rather than forecasts it. It uses the sigmoid function, which ranges from 0 to 1, to determine the likelihood that an input is in a class. One important benefit is simplicity; understanding the results is simple. Realistic regret provides probabilities, making it easy to understand the group likelihood of a datum. Additionally, the coefficients show how inputs affect the behavior of the output variable.

Furthermore, the logistic regression technique can be applied to datasets that contain noisy or defective data because it is unaffected by noise and outliers that may exist in the data set. Both numerical and categorical characteristics can be handled by it, however it is insensitive to multicollinearity, which necessitates feature scaling for convergence.

However, there are drawbacks to using logistic regression. Due to the assumption of a linear relationship between the input features and the log-odds of the target variable, it could not always hold true in actual use. When there are nonlinear interactions between the target and the feature, it is occasionally necessary to fit sophisticated models.

Additionally, linear or nearly linear class separation is ideal for the optimal performance of this type of logistic regression. For instance, more sophisticated techniques like neural networks or support vector machines work better when the decision boundary is extremely complex or non-linear.

Notwithstanding these shortcomings, logistic regression is still a crucial part of machine learning, particularly in situations where interpretability and simplicity are crucial. For instance, in jobs involving binary classification, it frequently acts as the default algorithm while more complex algorithms are added using ensemble techniques or larger machine learning pipelines. As a result, logistic regression is a flexible method that balances simplicity and efficacy.

8. Random Forest Classifier:-

Random Forest is a robust and versatile ensemble learning method that is frequently employed for regression and classification tasks. In order for this to function, a large number of decision trees are created during training, and the average prediction (for regression) or mode (for classification) of each tree is then output.

One of Random Forest's main advantages is its capacity to handle high-dimensional data while accounting for a large number of features. Random Forest may capture complex associations between characteristics and the target variable by integrating predictions from numerous decision trees. This prevents overfitting, which makes the model insensitive to both noise and data outliers.

The integrated feature importance measure, another feature provided by Random Forest, enables users to recognize the features that are important for predicting the target variable. This function selection feature in particular enables users to go deeper than what they initially perceive and take action.

Furthermore, Random Forest is suitable for large data sets and computationally demanding tasks since it scales effectively both horizontally and vertically. It may be effectively trained on distributed computing platforms or multicore processors, enabling rapid model building and deployment.

However, because Random Forest is an ensemble model, meaning it is harder to comprehend the reasoning behind its predictions, its interpretability can be limited when compared to individual decision trees. In addition, Random Forest might not work as well as other specialized algorithms in jobs involving data with strong non-linear correlations or problems with severely imbalanced classes.

Nevertheless, because it is adaptable, reliable, and easy to use, Random Forest is still a popular and often used machine learning technique. With excellent results, it is widely utilized for numerous classification and regression tasks in a variety of applications across multiple domains. Generally speaking, Random Forest is one of those potent machine learning methods that can handle challenging problems requiring high-dimensional data while offering scalability and processing efficiency.

9. **Ada Boost:-**

Adaptive Boosting, or AdaBoost, is a well-liked ensemble learning technique that builds a powerful classifier by combining multiple weak learners. Although it may also be used for multiclass classification and regression issues, binary classification is where it excels. In order to improve overall predictive performance, AdaBoost trains a series of weak learners repeatedly on cases from earlier models that were incorrectly classified.

AdaBoost's capacity to handle complicated datasets and capture non-linear correlations between features and the target variable is one of its most important advantages. AdaBoost efficiently focuses on observations that are challenging to classify, producing predictions that are reliable and accurate by iteratively changing the weights of the mistakenly categorized cases. Furthermore, AdaBoost is comparatively resistant to overfitting since it combines several weak learners with low predictive power into a single powerful classifier.

AdaBoost includes integrated feature importance metrics in addition to being able to be paired with multiple weak learner algorithms, including decision trees, linear classifiers, or even neural networks, in order to accommodate diverse kinds of data and modeling scenarios. This is closely related to making it simple to identify the features that are most useful for forecasting the desired variable.

However, AdaBoost's effectiveness could be compromised if base learners are overly complex or prone to overfitting, which results in poor generalization. Furthermore, because it minimizes the possibility of classification errors caused by outliers, it is susceptible to problems involving noisy data or outliers.

AdaBoost is still a highly-liked and often used machine learning algorithm in spite of these drawbacks because of how effectively it handles complex datasets and produces precise predictions. It has proven effective in a number of fields, such as financial forecasting, bioinformatics, and image and speech recognition. All things considered, AdaBoost is a useful addition to the machine learning toolkit that provides a robust and flexible answer for tasks involving regression and classification.

10. **Extra Tree Classifier:-**

Extremely Randomized Trees, or ExtraTrees, are an ensemble learning technique that builds upon Random Forests. During the training phase, it learns several decision trees, but with some modifications.

Decision trees are trained on random subsets of features and training data in ExtraTrees. Even

more randomization is added to the decision tree creation process by generating random thresholds for feature splits at each node rather than searching for the optimal split. This extra unpredictability improves the resilience of the model and lessens overfitting.

ExtraTrees has a number of benefits. It is less prone to overfitting than Random Forests since it often has lower variance, especially in high-dimensional datasets. Additionally, since it doesn't necessitate a thorough search for the optimal split at each node, it may be computationally more efficient.

On the other hand, analyzing individual trees in an ExtraTrees model might be difficult, and adjusting its hyperparameters might take more work. All things considered, ExtraTrees is a strong and effective ensemble learning technique that works well for a variety of regression and classification problems, particularly when working with noisy or high-dimensional data.

11. Gradient Boosting:-

Gradient boosting is becoming more and more popular because of its exceptional performance in classification and regression applications. This method uses a series of weak learners, primarily decision trees, combined in a sequential fashion such that each new model learns from the mistakes of its predecessors.

The accuracy of Gradient Boosting depends on the iterative removal of previous models' residuals, which it uses to create superior prediction models. In contrast to standard boosters such as AdaBoost, which modify the weights of incorrectly classified instances, gradient boosting directly optimizes the loss function, resulting in a more robust and efficient convergence.

A benefit of gradient boosting? Its scalability and adaptability are effective. It addresses classification, ranking, and regression issues while taking loss functions and optimization procedures into account. Additionally, it manages missing data, categorical characteristics, and outlier detection well for heterogeneous data sets found in the real world. Gradient Boosting is a proficient method for capturing intricate non-linear correlations between target variables and attributes. ideal for complex patterns in high-dimensional data. It includes feature importance metrics and visualizations as well, providing insights and drawing useful information from the underlying data.

Gradient Boosting is beneficial. However, it is not optimal when used to complex or noisy data. An excessive amount of fitting occurs. Too much is learned by it. That is not good. It need adjustment. Ideal conditions matter. Nonetheless, Gradient Boosting frequently prevails in contests. Better than other approaches. Why? It is strong. It is adaptable. It is easily scaled. It is capable of handling several issues. At the forefront is gradient boosting. It makes quite accurate predictions. Its outcomes can be explained. Gradient Boosting is excellent ensemble learning overall. Nothing compares to its clarity and predicting ability.

Chapter 3

Problem Statement/ Requirement Specifications

This section outlines the goals, data considerations, processing techniques, evaluation methods, and analysis plan for a project that predicts heart disease using machine learning models.

1. Project Planning:-

This part will cover the project's objectives, data collection methods, data preparation procedures, model evaluation techniques, and method of results analysis. A machine learning project must be carefully planned and executed in a methodical manner to be completed successfully. In order to move the project forward successfully, we determined certain actions to take at the planning stage:

- **Define Objectives:**
  - Primary Objective: Develop machine learning models to predict the probability of heart disease in individuals based on various medical features.
  - Secondary Objectives:
    - Compare the performance of different machine learning models (e.g., SVM, Decision Tree, Logistic Regression, Random Forest, etc.) for heart disease prediction.
    - Identify the most significant features influencing heart disease risk.

The initial stage in project planning involved clearly defining the objectives and scope of the study. Our objective was to develop and evaluate machine learning models for predicting cardiovascular diseases using the dataset.

- **Data Acquisition:**

We located and collected the datasets needed for our project. We specifically made advantage of the Kaggle dataset. When deciding which dataset to study, it was crucial to ensure that the data we used was relevant and of the highest quality.

- Data Description: The number of samples 1025 entries with 14 attributes.

- **Data Preprocessing:**

Once the dataset was acquired, we performed extensive data preprocessing to clean, transform, and prepare the data for analysis. This included handling missing values, encoding categorical variables, standardizing numeric features, and addressing data imbalances.

- **Model Selection:**

With the preprocessed dataset and selected features, we proceeded to select and implement multiple machine learning algorithms for heart disease prediction. The chosen algorithms included logistic regression, k-neighbors classifier, decision trees, random forest, AdaBoost, Naive Bayes, support vector machine, gradient boosting, Extra tree classifier.

- **Model Evaluation:**

After training the models, we evaluated their performance using appropriate evaluation metrics such as precision score, accuracy score.

- **Result Analysis:**

The final step involved analyzing the results obtained from model evaluation and drawing conclusions regarding the effectiveness of different algorithms in predicting heart disease. Insights gained from the analysis were used to inform future research directions and potential applications in healthcare.

2. Project Analysis

During the analysis stage of our project, we looked closely at and analyzed the data that came from predicting heart disease using machine learning algorithms. Our objectives were to evaluate the performance of these algorithms, identify the variables influencing their correctness, and make significant findings to inform future decisions.

- **Model Performance Comparison:**

Initially, we examined the performance of ten machine learning algorithms in our study. We assessed each model's precision and accuracy using data from both the training and testing phases. The results demonstrated that certain algorithms performed better in terms of prediction than others.

- **Identification of Top-performing Models:**

According to the evaluation metrics, we identified the top-performing models in terms of their capability to predict accurately disease status. The feed-forward Extra tree classifier following with Random forest emerged as the best-performing algorithm, exhibiting high accuracy and precision on both the training and testing datasets.

- **Limitations and Challenges:**

Despite the promising results obtained, our analysis also revealed certain limitations and challenges associated with the predictive models. These included the need for more extensive feature engineering, addressing class imbalances in the dataset.

3. System Design

● Design Constraints

The software system will have the following constraints:
a. Budget: The project budget is limited and should be kept within the allocated funds.
b. Time: The project timeline is limited, and the software system should be developed and deployed within the given time frame.

● System Architecture

The system architecture for the prediction project encompasses various components and processes designed to effectively predict heart disease status using machine learning models. The architecture involves numerous steps, including data preprocessing, model training, testing, and deployment.

- Data Collection and Preprocessing:
  - Numerous sources, including surveys and clinical databases, are used to collect data. Next, the data is cleaned as part of the preprocessing step to prepare it for analysis. This covers handling missing values, standardizing the data, and encrypting categorical variables.
- Model Training:
  - Multiple machine learning algorithms are trained using the preprocessed data.
  - The models include decision trees, k-neighbors classifier (KNC), random forest, support vector machines (SVM), logistic regression, AdaBoost, naive Bayes like (Gaussian naive bays, Bernoulli naive bays), gradient boosting (GB), Extra tree classifier.

- Model Evaluation and Testing:
  - The trained models are evaluated using performance metrics such as accuracy and precision.

- Model Deployment:
  - Once the best-performing model is identified, it is deployed in a production environment for real-time prediction.

Chapter 4

Implementation

1. Methodology/ Proposal:-

- **Data Exploration and Preprocessing:-**

The initial phase of our project involved data exploration and preprocessing to ensure the dataset's readiness for further analysis. Upon loading the data, we conducted a comprehensive review of its structure, which revealed that it comprises 1025 rows and 14 columns. Each column represents a specific health-related attribute, ranging from indicators of chronic conditions to lifestyle habits. Notably, the target variable, indicates whether an individual has diseases (0 for absence, 1 for presence).

We then read the descriptions of the traits in order to understand their importance. Age, sex, type of chest pain, resting blood pressure, blood sugar, serum cholesterol, and electrocardiogram, as well as the maximum heart rate attained, exercise-induced angina, old peak = ST depression induced by exercise relative to rest, the slope of the peak exercise ST segment, and the number of major vessels (0–3) colored by fluoroscopy are just a few of the many health-related factors covered by these attributes.

Additionally, socio-economic factors such as access to healthcare, income, education level, and demographic details like age and sex were also included. To ensure consistency in data representation, we transformed categorical variables into integers where applicable. Notably, most variables were already in numeric format, facilitating straightforward analysis. We then conducted a thorough examination for missing values, outliers, and duplicated records. Fortunately, our dataset exhibited no missing values or outliers, ensuring its integrity for subsequent analyses.

- **Data Modeling:-**

Ten distinct classification models were employed to train and assess performance on the training and testing sets for our prediction study. These models encompass a range of algorithmic techniques, each with unique benefits and capacities to manage intricate datasets and target variable forecasts. Ten classification models were employed in our study: K-Neighbor Classifier (KNC), Logistic Regression, Support Vector Machine, Decision Tree, Extra Tree Classifier, Random Forest, Gradient Boost, Naive Bayes, and Ada Boost. Thirty percent of the total data set and seventy percent of the training set were used to train each model. Our goal was to apply various classification models to assess performance critically and determine which disease prediction models, based on the properties of the data sets, were the most successful.

.

2. Result Analysis:-

In our result analysis, we assessed the performance of ten distinct classifier models on both the training and testing datasets. The evaluation encompassed various metrics to provide a deep comprehending of each techniques' predictive capabilities. Here, we detail the evaluation outcomes and insights garnered from our analysis.

Evaluation Metrics:

Confusion Matrices: By categorizing the predictions into four outcomes in a binary classification task, the confusion matrix gives a basic idea of the model's performance:

- True Positive (TP): Algorithm correctly predicts the positive class.
- False Positive (FP): Algorithm wrongly identifies the positive class for a negative instance.
- True Negative (TN): Algorithm correctly predicts the negative class.
- False Negative (FN): Algorithm wrongly identifies the negative class for a positive instance.

These outcomes are used to calculate different performance metrics, which includes accuracy and precision, which evaluates how effectively the model categorizes inputs into their suitable categories. Let's explore the equations for these performance metrics.

Specificity: Out of all the positive predictions the model makes, specificity is the number of correct forecasts. The ratio of the total number of true positives to false positives is used to compute true positives.

Precision = TP/(TP+FP)

Confusion Matrices: By grouping predictions into four possible outcomes, the confusion matrix in a binary classification task gives a general overview of a model's performance:

- True Positive (TP): The best class is accurately predicted by the algorithm.
- False Positive (FP): For negative cases, the algorithm determines the right object class wrongly.
- True Negative (TN): The negative class is accurately predicted by the algorithm.
- False Negative (FN): When positive examples are identified by the algorithm, the negative class is wrongly identified.

To determine how successfully a model can classify inputs into the correct classes, data is used to calculate precision and accuracy performance metrics. In this instance, let's explore the variability of these indicators.

Specificity: Out of all the positive predictions the model makes, specificity is the number of correct forecasts. The ratio of the total number of true positives to false positives is used to compute true positives.

Precision = TP/(TP+FP)

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	0
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1020	59	1	1	140	221	0	1	164	1	0.0	2	0	2	1
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3	0
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2	0
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2	1
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3	0

Fig.1- Analyzing dataset using pandas

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1025 entries, 0 to 1024
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1025 non-null   int64
 1   sex       1025 non-null   int64
 2   cp        1025 non-null   int64
 3   trestbps  1025 non-null   int64
 4   chol      1025 non-null   int64
 5   fbs       1025 non-null   int64
 6   restecg   1025 non-null   int64
 7   thalach   1025 non-null   int64
 8   exang     1025 non-null   int64
 9   oldpeak   1025 non-null   float64
 10  slope     1025 non-null   int64
 11  ca        1025 non-null   int64
 12  thal      1025 non-null   int64
 13  target    1025 non-null   int64
dtypes: float64(1), int64(13)
memory usage: 112.2 KB
```

Fig.2- Dataset information

```
<Axes: xlabel='target', ylabel='count'>
```

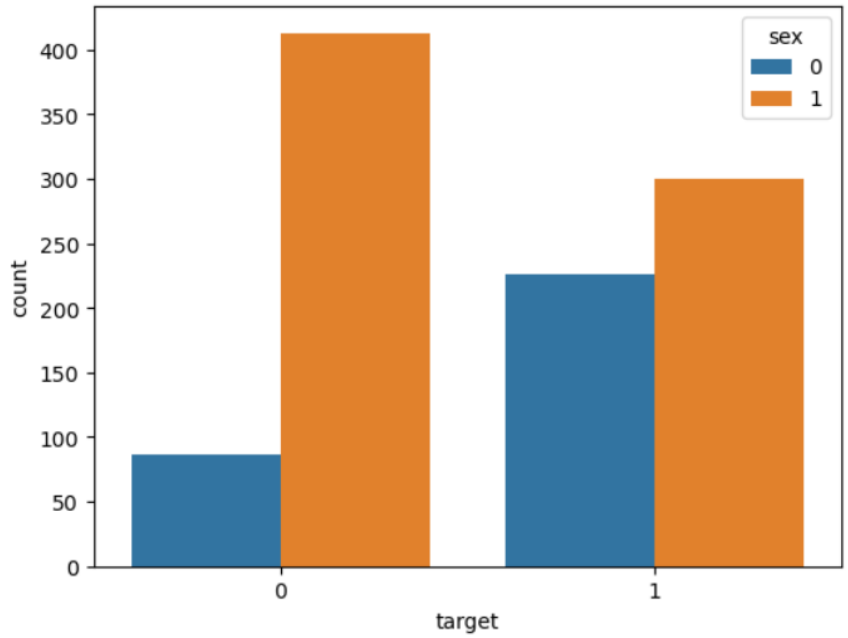

Fig.3- Plot Relationship of sex having Heart Disease

```
<Axes: xlabel='target', ylabel='count'>
```

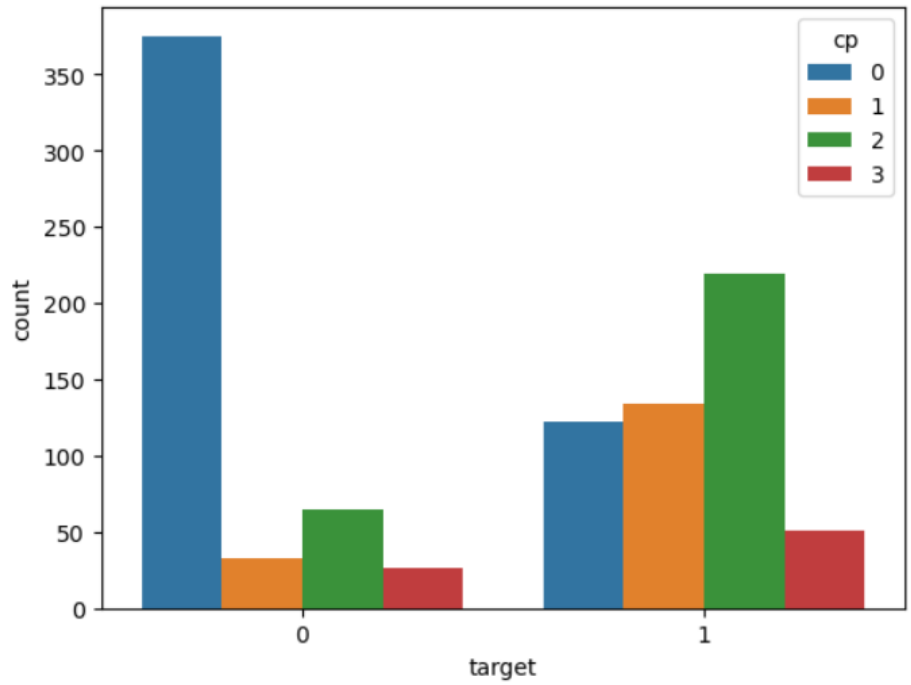

Fig.4- Plot Relationship between chest pain and Heart disease

```
<Axes: xlabel='sex', ylabel='count'>
```

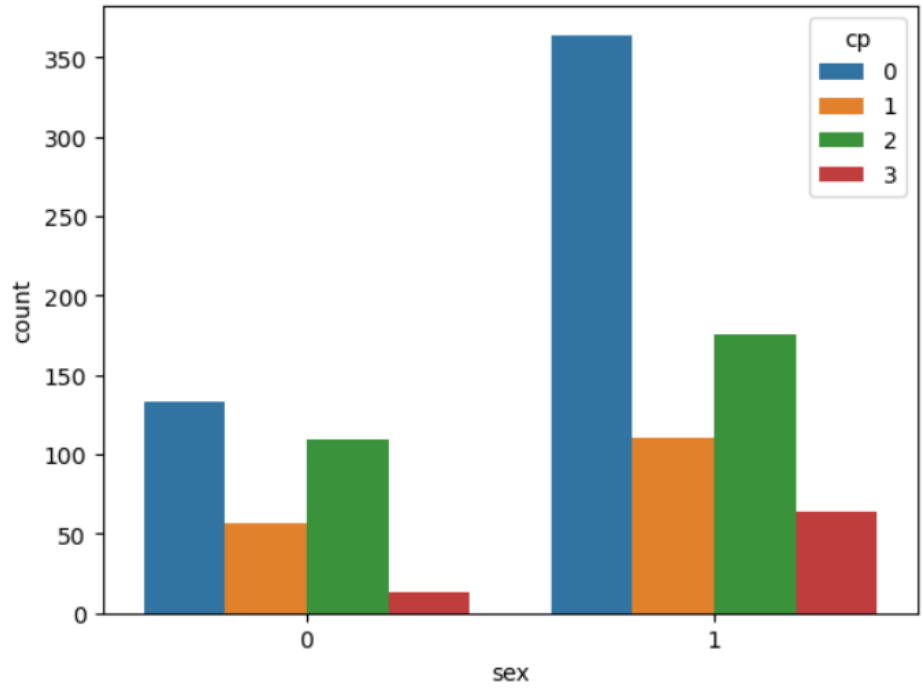

Fig.5- Plot Relationship of sex having chest pain

```
<Axes: xlabel='sex', ylabel='trestbps'>
```

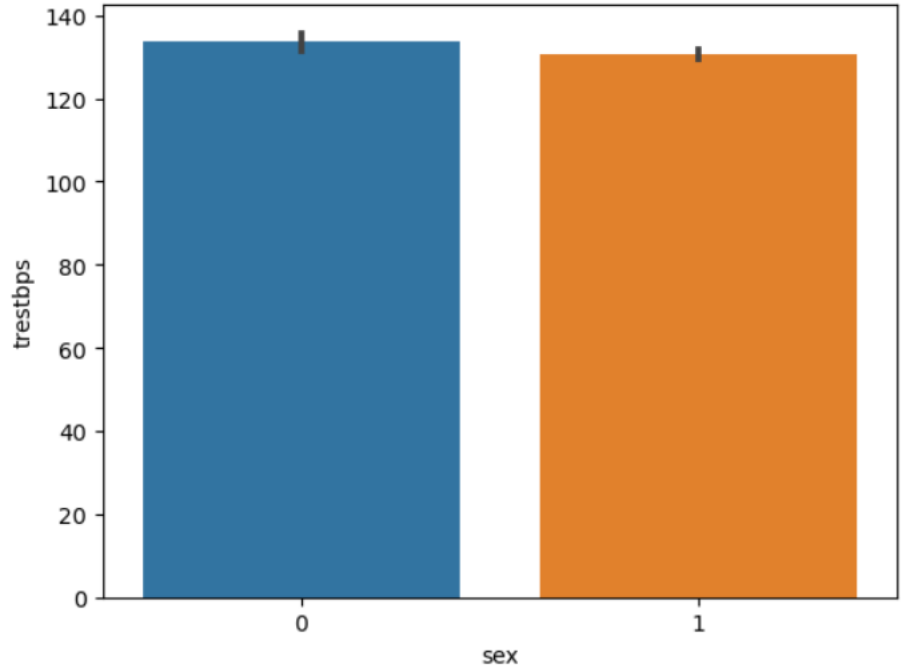

Fig.6- Plot Relationship of sex having BPS

```
<Axes: xlabel='age', ylabel='chol'>
```

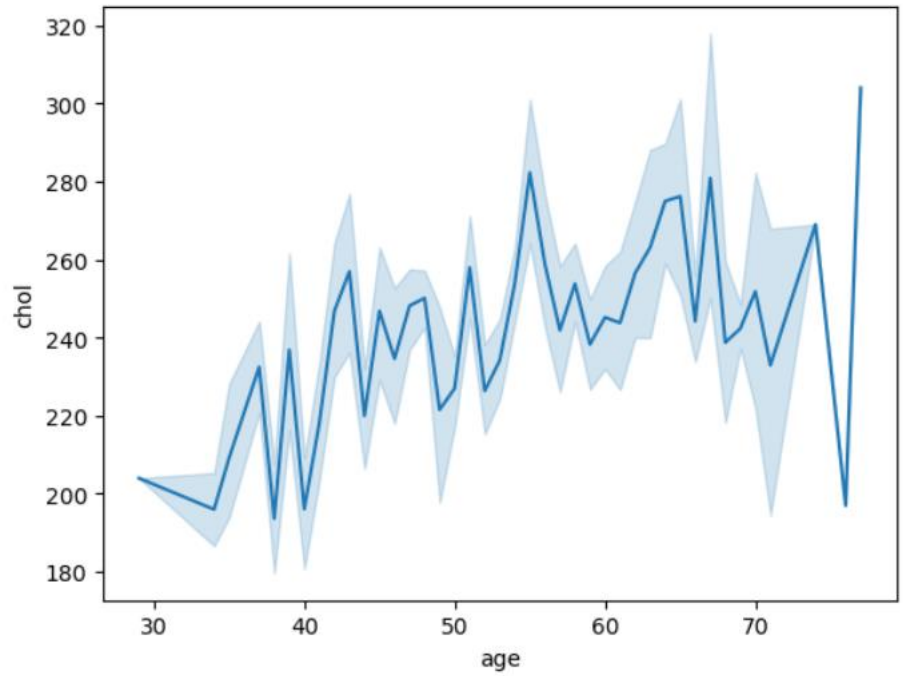

Fig.7- Histogram of Age having Cholesterol

```
<Axes: xlabel='age', ylabel='trestbps'>
```

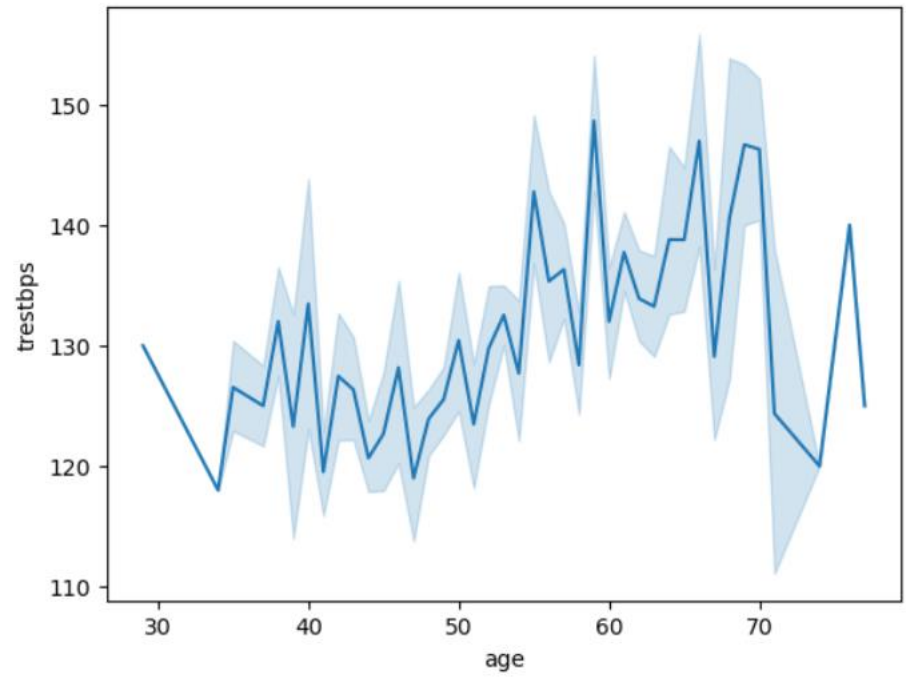

Fig.8- Histogram of Age having BPS

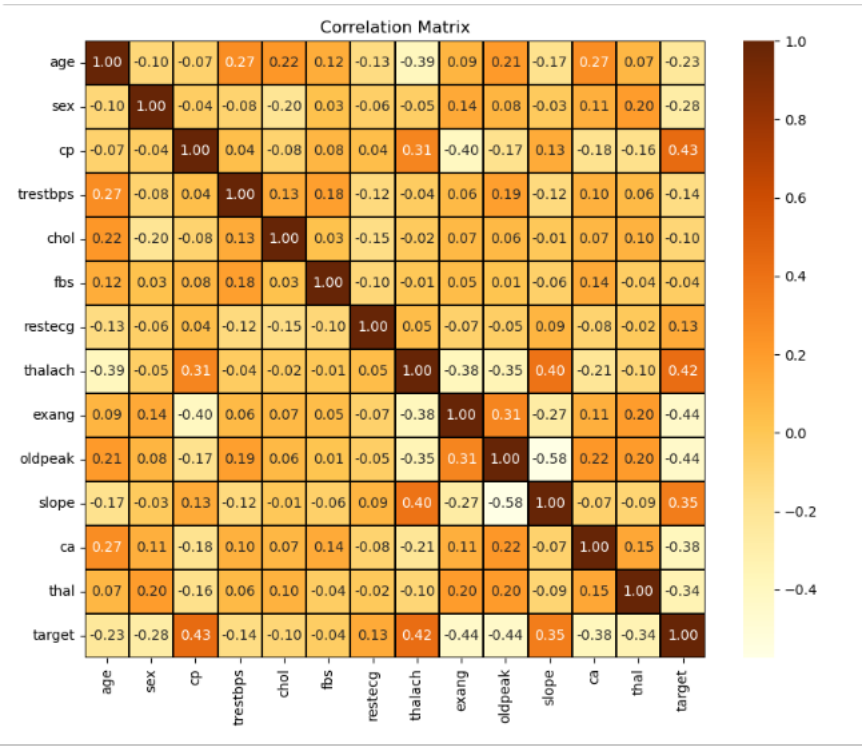

Fig.9- Correlation Matrix

	Algorithm	Accuracy	Precision
8	ExtraTree	1.000000	1.000000
6	Random Forest	0.985366	1.000000
9	Gradient Boosting	0.897561	0.872727
7	AdaBoost	0.878049	0.890000
4	Decision Tree	0.843902	0.784000
1	K-Neighbors	0.834146	0.800000
0	SVC	0.814634	0.760000
2	Gaussian-NaiveBay	0.800000	0.754098
3	Bernoulli-NaiveBay	0.795122	0.765217
5	Logistic Regression	0.795122	0.756303

Fig.10- Comparison table of various models

Chapter 5

Standard Adopted

1. **Design Standards**
   We complied with accepted preparation guidelines to ensure the accuracy and quality of the data. This includes filling in the gaps in the data, locating and eliminating anomalies, and standardizing the data to guarantee correctness and cross-specialty comparability.

   Our heart disease prediction model strategy made use of tried-and-true data science techniques. Support Vector Machine (SVM), K-Neighbors, Hilbert-Bernoulli and Gaussian Naive Bayes, Decision Trees, Logistic Regression, AdaBoost, ExtraTree, Random Forest and Gradient Boosting are some of these methods. By employing these methods, we make sure that our predictions are solid and certain. Our goal was to make the models more predictive.

2. **Coding Standards**
   We adhered to the PEP 8 style guide's rules to guarantee readability and consistency in our Python code. This made it easier for us to keep the project's formal legal framework and procedure in place.

   In order to improve reusability and maintainability, we employed a modular code design strategy. This required us to separate our code into several courses and activities, each with a distinct function. This method of code classification made maintenance easier and promoted code reuse across project components.

   We also give rule lifecycle and maintenance top priority by covering the entire software with comments and documentation. This information made it easier for developers to comprehend and make necessary adjustments by providing us with insight into how particular rules operated.

3. Testing Standards

Following standard procedure, we split our dataset into distinct training and testing sessions in order to more accurately assess the performance of our model.

We examined the outcomes of two distinct clustering methods to determine which model was best for our data set.

In order to validate our model's performance and self-check for overfitting, we employed cross-validation techniques. This made it possible for us to assess our model's effectiveness and generalizability on unseen data.

Additionally, we measured the efficacy of our algorithm using a variety of evaluation measures, including the confusion matrix. These metrics offered insightful information about the model's performance and accuracy across classes or classes.

Throughout the process, we were committed to upholding strict guidelines to guarantee the durability, dependability, and maintainability of our heart disease prediction system.

Chapter 6

Conclusion & Future Scope

Conclusion

Finally, our heart disease prediction Python project has shown the adaptability and power of machine learning models. We obtained important insights into the performance of eight distinct algorithms and whether they are suitable for heart disease prediction by putting them into practice and assessing them: Logistic Regression, Support Vector Machine, K-Neighbors, Gaussian Naive Bayes, Decision Tree, Random Forest, AdaBoost, and Gradient Boosting ,we have gained valuable insights into their performance and suitability for predicting heart disease

Our comprehensive analysis revealed that each model offered unique strengths and capabilities in tackling the complexities of heart disease prediction. From the flexibility of SVM to the simplicity of Decision Trees, and the ensemble learning prowess of Random Forest and Gradient Boosting, each algorithm contributed to the overall predictive accuracy and robustness of our system.

Furthermore, it was shown how crucial model selection, hyperparameter tuning, and assessment metrics are to raising our project's prediction performance. We tested for use with unseen data and made sure our models were dependable and generalizable by employing strategies like confusion matrices and cross-validation to prevent overfitting.

All things considered, our work emphasizes how critical it is to use a variety of machine learning methods in healthcare applications like cardiovascular prediction. Future studies and adjustments to these models show promise for boosting screening precision, facilitating early intervention, and eventually improving patient outcomes in terms of cardiovascular health.

Future Scope

The future scope of our Python project on heart disease prediction using eight machine learning models presents several promising avenues for further exploration and development:

Model Ensemble Techniques Examine state-of-the-art ensemble methods for integrating various models' predictions. It is possible to investigate methods like weighted averaging, stacking, and blending to improve the precision and resilience of forecasts.

Feature Engineering: Find creative ways to improve characteristics so you can comprehend the data better and derive useful insights from it. You can enhance the model's overall performance by putting strategies into practice like picking key characteristics, simplifying the data, and tailoring features to particular domains.

- **Hyperparameter Tuning:** Conduct comprehensive hyperparameter tuning for each model to optimize their performance further you can adjust model parameters and boost prediction accuracy by using techniques like grid search, random search, and Bayesian optimization..

- **Integration of Deep Learning:** Examine the ways in which deep learning techniques, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), can be applied to comprehend complex relationships and patterns in data. Predictive accuracy can be increased by combining deep learning models with traditional machine learning models.

- **Real-time Prediction:** The project can be expanded to provide a real-time cardiac disease prediction system. We can give patients ongoing health status monitoring and early identification of potential heart issues by fusing wearable technology with healthcare.

- **Interpretability and Explainability:** To facilitate their use in medical settings, make the models more transparent and clearer. To comprehend how the model generates decisions, techniques like as evaluating the significance of features, visualizing the model, and employing surrogate models can be applied.

- **Validation on Diverse Datasets:** Determine how successfully the models can be applied to diverse groups by testing their accuracy on a variety of datasets that represent different populations and demographics. This will attest to the models' dependability and effectiveness in a range of real-world scenarios.

- **Clinical Integration and Validation:** Work together with specialists and medical professionals to confirm the model's efficacy and dependability in actual clinical situations. Conduct research investigations and clinical trials to assess how the models affect patient outcomes and clinical decision support.

Our research can enhance cardiac health monitoring and diagnostics by examining these pathways, which will ultimately improve patient outcomes.

References

1. *Panda, Amiya Ranjan, et al. "System for Detecting Drowsiness in Drivers." 2023 OITS International Conference on Information Technology (OCIT), 2023, pp. 738–342. IEEE Xplore, https://doi.org/10.1109/OCIT59427.2023.10431259*
2. *Banerjee, Amitayas, et al. "Comparative Analysis of Machine Learning and ANN Models for Mortality Prediction in RTAs." 2023 OITS International Conference on Information Technology (OCIT), 2023, pp. 698–702. IEEE Xplore, https://doi.org/10.1109/OCIT59427.2023.10431379.*
3. *Nayak, Amlan, et al. "Language Detection Using Machine Learning." 2023 OITS International Conference on Information Technology (OCIT), 2023, pp. 732–37. IEEE Xplore, https://doi.org/10.1109/OCIT59427.2023.10430539.*
4. *Chakraborty, Srijita, et al. "Predicting Diabetes: A Comparative Study of Machine Learning Models." 2023 OITS International Conference on Information Technology (OCIT), 2023, pp. 743–48. IEEE Xplore, https://doi.org/10.1109/OCIT59427.2023.10431372.*
5. *Tiwari, Abhinandan Kumar, et al. "Parametric Examination on Optimized Deep Learning Based Melanoma Detection." 2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2021, pp. 1–8. IEEE Xplore, https://doi.org/10.1109/ICECCT52121.2021.9616885.*
6. *Swetanisha, Subhra, et al. "Change Detection Using Machine Learning Models: A Case Study on the Puri District of Odisha, India." 2021 19th OITS International Conference on Information Technology (OCIT), 2021, pp. 100–04. IEEE Xplore, https://doi.org/10.1109/OCIT53463.2021.00030.*
7. *Drożdż, K.; Nabrdalik, K.; Kwiendacz, H.; Hendel, M.; Olejarz, A.; Tomasik, A.; Bartman, W.; Nalepa, J.; Gumprecht, J.; Lip, G.Y.H. Risk factors for cardiovascular disease in patients with metabolic-associated fatty liver disease: A machine learning approach. Cardiovasc. Diabetol.* ***2022****, 21, 240.*
8. *Murthy, H.S.N.; Meenakshi, M. Dimensionality reduction using neuro-genetic approach for early prediction of coronary heart disease. In Proceedings of the International Conference on Circuits, Communication, Control and Computing, Bangalore, India, 21–22 November 2014; pp. 329–332.*
9. *Benjamin, E.J.; Muntner, P.; Alonso, A.; Bittencourt, M.S.; Callaway, C.W.; Carson, A.P.; Chamberlain, A.M.; Chang, A.R.; Cheng, S.; Das, S.R.; et al. Heart disease and stroke statistics—2019 update: A report from the American heart association. Circulation* ***2019****, 139, e56–e528.*
10. *Shorewala, V. Early detection of coronary heart disease using ensemble techniques. Inform. Med. Unlocked* ***2021****, 26, 100655.*
11. *Mozaffarian, D.; Benjamin, E.J.; Go, A.S.; Arnett, D.K.; Blaha, M.J.; Cushman, M.; de Ferranti, S.; Després, J.-P.; Fullerton, H.J.; Howard, V.J.; et al. Heart disease and stroke statistics—2015 update: A report from the American Heart Association. Circulation* ***2015****, 131, e29–e322.*
12. *Maiga, J.; Hungilo, G.G.; Pranowo. Comparison of Machine Learning Models in Prediction of Cardiovascular Disease Using Health Record Data. In Proceedings of the 2019 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS), Jakarta, Indonesia, 24–25 October 2019; pp. 45–48.*
13. *Li, J.; Loerbroks, A.; Bosma, H.; Angerer, P. Work stress and cardiovascular disease: A life course perspective. J. Occup. Health* ***2016****, 58, 216–219.*
14. *Purushottam; Saxena, K.; Sharma, R. Efficient Heart Disease Prediction System. Procedia Comput. Sci.* ***2016****, 85, 962–969.*
15. *Soni, J.; Ansari, U.; Sharma, D.; Soni, S. Predictive Data Mining for Medical Diagnosis: An Overview of Heart Disease Prediction. Int. J. Comput. Appl.* ***2011****, 17, 43–48.*
16. *Mohan, S.; Thirumalai, C.; Srivastava, G. Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques. IEEE Access* ***2019****, 7, 81542–81554.*

Heart Disease Prediction

Pravin Kumar Pattnaik

2128081

Abstract-

In this study, we describe how one individual helped a group Python project that used many machine learning models to predict cardiac disease. Ten various algorithms, including logistic regression, decision trees, and random forests, were meticulously implemented and enhanced by our team, with an emphasis on refining the models via data preparation, parameter adjustments, and cross-validation. In order to increase the precision and intelligibility of our forecasts, we also looked into techniques like merging models and evaluating the significance of individual features. This study emphasizes the importance of developing trustworthy tools for early detection and intervention of cardiac problems using Python's tools and approaches.

Individual contribution and findings:

He took the lead in data collection, analysis, and preparation for the heart disease prediction Python project, making sure the dataset was ready for usage. He carefully reviewed the data, prepared it using a number of methods, and employed a range of models, such as random forests, logistic regression, SVM, decision trees, and Naive Bayes. His collaboration made sure the project continued to move toward its goals. His assessments of the models yielded insightful criticism for improving diagnostic precision.

Individual contribution to project report preparation:

He contributed for the following portions:
5. Standards Adopted
6. Conclusion and Future Scope
References

Individual contribution for project presentation and demonstration:

He concentrated on graphs, applications, outcomes, and conclusions. He found useful applications, examined data to provide reliable findings, came to wise judgments, and made understandable, eye-catching visualizations. His work significantly improved the way we presented and demonstrated our project..

Full Signature of Supervisor:

Full signature of the student:

Heart Disease Prediction

Sonit Kumar Swain

2128120

Abstract-

In this study, we describe how one individual helped a group Python project that used many machine learning models to predict cardiac disease. Ten various algorithms, including logistic regression, decision trees, and random forests, were meticulously implemented and enhanced by our team, with an emphasis on refining the models via data preparation, parameter adjustments, and cross-validation. In order to increase the precision and intelligibility of our forecasts, we also looked into techniques like merging models and evaluating the significance of individual features. This study emphasizes the importance of developing trustworthy tools for early detection and intervention of cardiac problems using Python's tools and approaches.

Individual contribution and findings:

He oversaw the creation of graphs and the analysis of findings for a Python project that predicted heart disease. He made sure the model's performance was thoroughly evaluated and that the graphics were simple to comprehend. He meticulously created a range of visual aids that provided insightful information about the dataset and the efficacy of various models. His methodical methodology made it possible to compare algorithms like logistic regression, random forests, SVM, decision trees, and Naive Bayes in a meaningful way, which aided in decision-making throughout the project. His in-depth examination of the graphs and findings provided helpful criticism to raise the models' accuracy, which in turn raised diagnostic precision.

Individual contribution to project report preparation:

He contributed for the following portions:

1. Introduction
2. Basic Concepts/Literature Review

Individual contribution for project presentation and demonstration:

He wrote the introduction, used a variety of machine learning models, provided creative solutions that improved accuracy and clarity, and provided an assessment of the project's progress.

Full Signature of Supervisor:

Full signature of the student:

Heart Disease Prediction

Soumyadeep Pal

2128093

Abstract-

In this study, we describe how one individual helped a group Python project that used many machine learning models to predict cardiac disease. Ten various algorithms, including logistic regression, decision trees, and random forests, were meticulously implemented and enhanced by our team, with an emphasis on refining the models via data preparation, parameter adjustments, and cross-validation. In order to increase the precision and intelligibility of our forecasts, we also looked into techniques like merging models and evaluating the significance of individual features. This study emphasizes the importance of developing trustworthy tools for early detection and intervention of cardiac problems using Python's tools and approaches.

Individual contribution and findings:

In order to ensure that the dataset was ready for analysis, he took on the task of designing and implementing models for the Python project that predicted heart disease. He thoroughly examined the data and applied several methods to effectively clean and process it. Using his expertise, he expanded the analysis capabilities by applying a variety of models, including logistic regression, random forests, SVM, decision trees, and Naive Bayes. His strong teamwork made the project's progress toward its objectives easier to manage. Furthermore, his analyses of the models provided insightful information for raising diagnostic precision. He made a significant contribution by closely evaluating each model's performance.

Individual contribution to project report preparation:

He contributed for the following portions:
3. Problem Statement / Requirement Specifications
4. Implementation

Individual contribution for project presentation and demonstration:

He oversaw the model architecture for the heart disease prediction project, creating and putting into practice a variety of machine learning models. He also gave a thorough project summary, and precisely explained the process and findings.

Full Signature of Supervisor:

Full signature of the student:

Heart Disease Prediction

ORIGINALITY REPORT

6%	4%	3%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to University of Westminster Student Paper	1%
2	academic-accelerator.com Internet Source	1%
3	content.yudu.com Internet Source	1%
4	link.springer.com Internet Source	<1%
5	assets.researchsquare.com Internet Source	<1%
6	Submitted to National School of Business Management NSBM, Sri Lanka Student Paper	<1%
7	Submitted to Colorado Technical University Student Paper	<1%
8	Md Sakir Ahmed, Abhijit Bora. "chapter 1 A Comprehensive Approach for Using Hybrid Ensemble Methods for Diabetes Detection", IGI Global, 2024 Publication	<1%