A dissertation submitted to the University of Greenwich
in partial fulfilment of the requirements for the Degree of

Master of Science
in
Management of Business Information Technology

Heart Disease Prediction

Name: Mansi Bipin Pande

Student ID: 001228420

Supervisor: Dr. Ralph Barthel
Submission Date: 22nd December,2023
Word count: 13,326

Contents

Table of figures

Abstract:

The in-depth analysis of heart disease prediction utilizing machine learning in the application tackles the worldwide health concern faced by heart disease. The primary goal is to create a prediction model utilizing a dataset supplemented with medical and demographic information. The procedure starts with thorough data preparation, with an emphasis on data integrity and the management of irrelevant columns and duplicate entries. The section on exploratory data analysis emphasizes the usefulness of visualization tools in understanding data properties and inter-variable interactions. For successful model validation, the model selection process includes selecting appropriate methods and splitting data into training and testing sets. This stage is critical in establishing the generalizability and robustness of the model in real-world circumstances. The importance of hyperparameter tweaking in improving model performance is emphasized, with an emphasis on fine-tuning model parameters to get optimal outcomes.

The model assessment portion is thorough, including measures such as accuracy, precision, and confusion matrices to assess model efficacy. This extensive review is required to ensure the prediction model's reliability and accuracy in detecting possible heart disease patients.

The application concludes by emphasizing the importance of machine learning in enhancing diagnostic capacities in healthcare, notably in the early diagnosis and intervention of heart disease. This method not only helps to reduce the mortality rate linked with heart disease, but it also greatly contributes to the evolution of personalized medicine, in which predictive analytics may be adapted to individual risk factors and health profiles. Overall, the research shows how machine learning may revolutionize healthcare diagnostics, opening the door for more proactive and preventative healthcare methods.

Acknowledgements

I would like to thank Professor Dr. Ralph Barthel for guiding me throughout this project and for being a very supportive supervisor for this MSc project of heart disease prediction case study and I would especially like to thank my professor for providing me with insights in every step of my project and for providing feedback and helping in every step of creating this project.

I'd like to thank Dr. Ralph Barthel and Dr. Simon Scola for accepting the proposal and organizing the project demo. And always encouraged me by providing comments throughout the project cycle, which enabled me to complete the project effectively.

Chapter 1

1.0 Introduction

1.1 Overview:

Machine learning for heart disease prediction entails assessing medical and demographic data to identify persons at risk of heart diseases. This method often involves data preparation to assure accuracy, exploratory data analysis for pattern recognition, and selecting appropriate machine learning models. In addition, hyperparameter adjustment for optimal model performance and extensive model evaluation utilizing metrics such as accuracy and precision are part of the process. This strategy improves early identification and intervention tactics, with the potential to significantly improve healthcare outcomes in heart disease management. This comprehensive application focuses on using machine learning to forecast cardiac disease. It starts with data preparation, emphasizing the need of high-quality data and comprehensive cleansing. The exploratory data analysis segment use visualization tools to decipher data patterns and comprehend variable interactions, which is essential for informed model creation. Model selection is an important procedure that involves the strategic selection of algorithms as well as the partitioning of data into training and testing sets to assess model efficacy. To improve model performance, hyperparameter tweaking is used, and model evaluation is done using measures such as accuracy and precision. The application emphasizes machine learning's transformational potential in improving diagnostic capacities, notably for early identification of cardiac disease, emphasizing its relevance in modern healthcare.

1.2 Aim:

The goal of the "Heart Disease Prediction" is to create a complete machine learning model for predicting heart disease. This entails processing and evaluating a dataset including medical and demographic information. The project covers data pretreatment to assure quality, exploratory data analysis with visualization tools to get deeper insights into the data, and the selection of appropriate machine learning models. The method also includes optimizing hyperparameters for optimal performance and

assessing the model using accuracy and precision measurements. The objective is to improve the diagnostic process, allowing for earlier diagnosis and better healthcare outcomes for heart disease, which is the leading cause of death worldwide. This study exemplifies machine learning's potential to revolutionize healthcare diagnosis.

1.3 Objectives:

Create a Predictive Model: Create a machine learning model that can reliably forecast the likelihood of heart disease in individuals based on a variety of medical and demographic parameters.

Data Analysis and Preprocessing: Analyze and preprocess the heart disease dataset for successful model training, including cleaning, managing missing values, and fixing imbalances.

Understanding and selecting key characteristics that significantly contribute to heart disease risk, hence improving the model's predicted accuracy.

Model Comparison and Selection: Based on performance criteria, analyze multiple machine learning algorithms, and pick the most effective one for heart disease prediction.

Hyperparameter Optimization: The process of fine-tuning the parameters of a given model to improve accuracy and dependability.

Model Evaluation: Using relevant measures like as accuracy, precision, recall, and ROC curves, thoroughly analyze the model to ensure its efficacy in real-world circumstances.

Clinical Relevance and Application: Ensuring that the model's predictions are clinically relevant and that they may be effectively incorporated into healthcare systems for early diagnosis and preventative initiatives.

Advancing Healthcare via Technology: To contribute to the larger objective of better healthcare outcomes and predicting medicine via the use of modern technology, particularly machine learning.

1.4 Motivation behind the project:

Beyond early diagnosis and tailored risk assessment, this machine learning-powered heart disease prediction tool envisions a future in which proactive prevention takes

center stage. Consider individuals who are enabled not just to passively receive risk ratings, but also to actively participate in preventative steps customized to their personal profile. The software might work with fitness trackers and wearables to provide individualized workout suggestions and real-time health monitoring to encourage users to adopt healthy behaviors. Furthermore, the AI capabilities of the software might go beyond individual risk assessment to forecast possible epidemics or identify high-risk communities. The software might identify geographical locations with a greater frequency of heart disease risk factors by evaluating aggregated data, triggering targeted public health measures. This degree of predictive power has the potential to transform community-based healthcare by enabling proactive preventative interventions before symptoms ever appear. Beyond people and communities, the app has the potential to empower healthcare professionals as well. Consider clinicians who have access to AI-powered insights that allow them to personalize treatment regimens and adjust actions based on a patient's individual risk profile. The app might give real-time feedback on medication adherence and alert users to potential drug interactions, allowing doctors to deliver better treatment while avoiding side effects. This seamless collaboration between AI and human expertise could usher in a new era of data-driven, patient-centric healthcare. Finally, this heart disease prediction software is more than just a risk assessment tool. It is a significant step toward a future in which technology enables individuals to actively protect their health, communities to implement focused preventative efforts, and healthcare professionals to provide individualized, data-driven treatment. This forward-thinking strategy ushers in a healthier, more empowered future by opening the way for sophisticated, preventative healthcare solutions that benefit individuals, communities, and the healthcare system.

1.5 Road map to the project report:

Project Planning and Research: Define the project scope and objectives, as well as conduct a literature research to understand existing approaches for predicting heart disease using machine learning.

Data Collection and Analysis:

Collect and analyze relevant datasets, such as medical records, lifestyle statistics, and demographic information. Perform preliminary data analysis to comprehend data properties.

App Design and Development: Outline the user interface and experience of the app. Create the app framework, including features for data entry, analysis, and result presentation.

Machine Learning Model Development: Select appropriate machine learning algorithms. Create and train models with the dataset, concentrating on prediction accuracy and dependability.

Model Testing and Validation: Validate the models' prediction ability using a distinct dataset. Models are adjusted and refined based on test outcomes.

Integration and Deployment: Integrate the machine learning models into the app. Deploy a beta version to get early user input.

User Feedback and App Enhancement: Gather input from users on app usability and model correctness. Make any necessary improvements and changes.

Final Deployment and Marketing: Release the app's final version. Implement marketing techniques to reach out to target users, with an emphasis on healthcare practitioners and those at risk of heart disease.

Ongoing Support and Updates: Continue to support the app. Based on fresh research, data, and user input, update the machine learning models and app features.

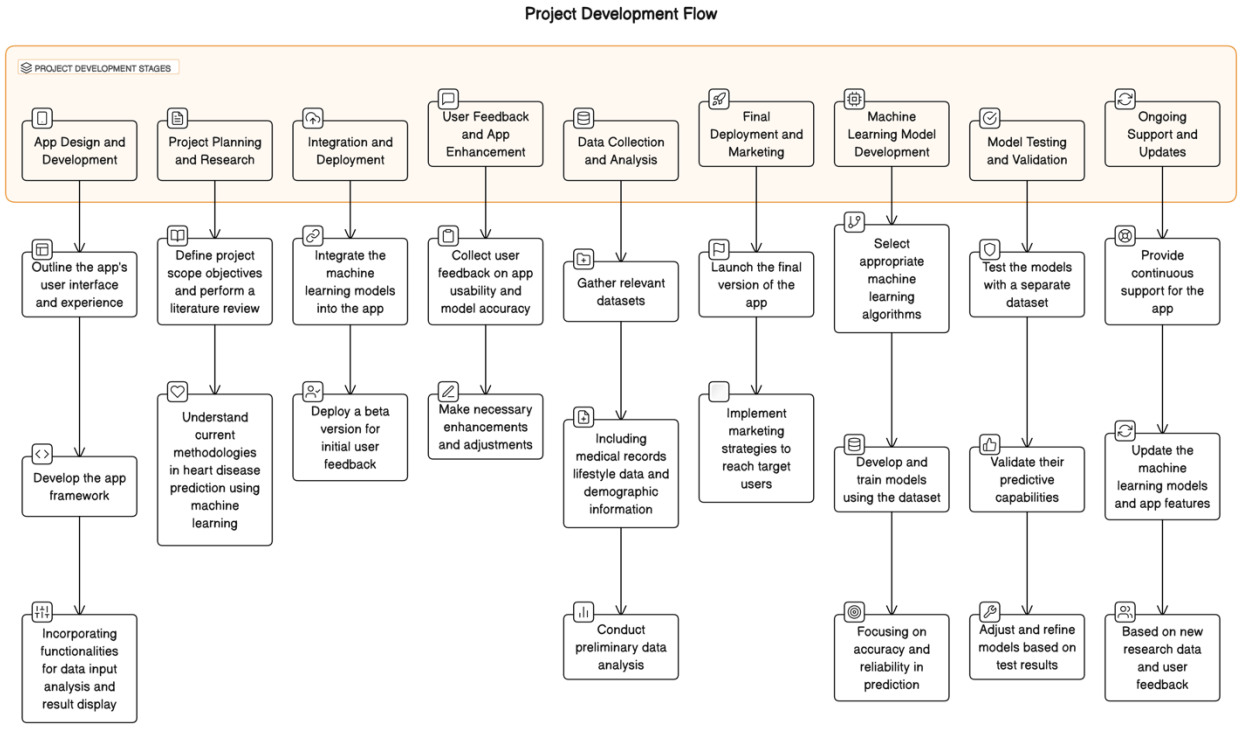


Figure 1

1.6 Project proposed model:

This project's project model is the water flow method, which follows the path illustrated in the flow graphic. In general, when developing any model or prototype, a definite structure is followed to ensure that the product provides the intended results. As a result, we used the water follow approach to develop the project.

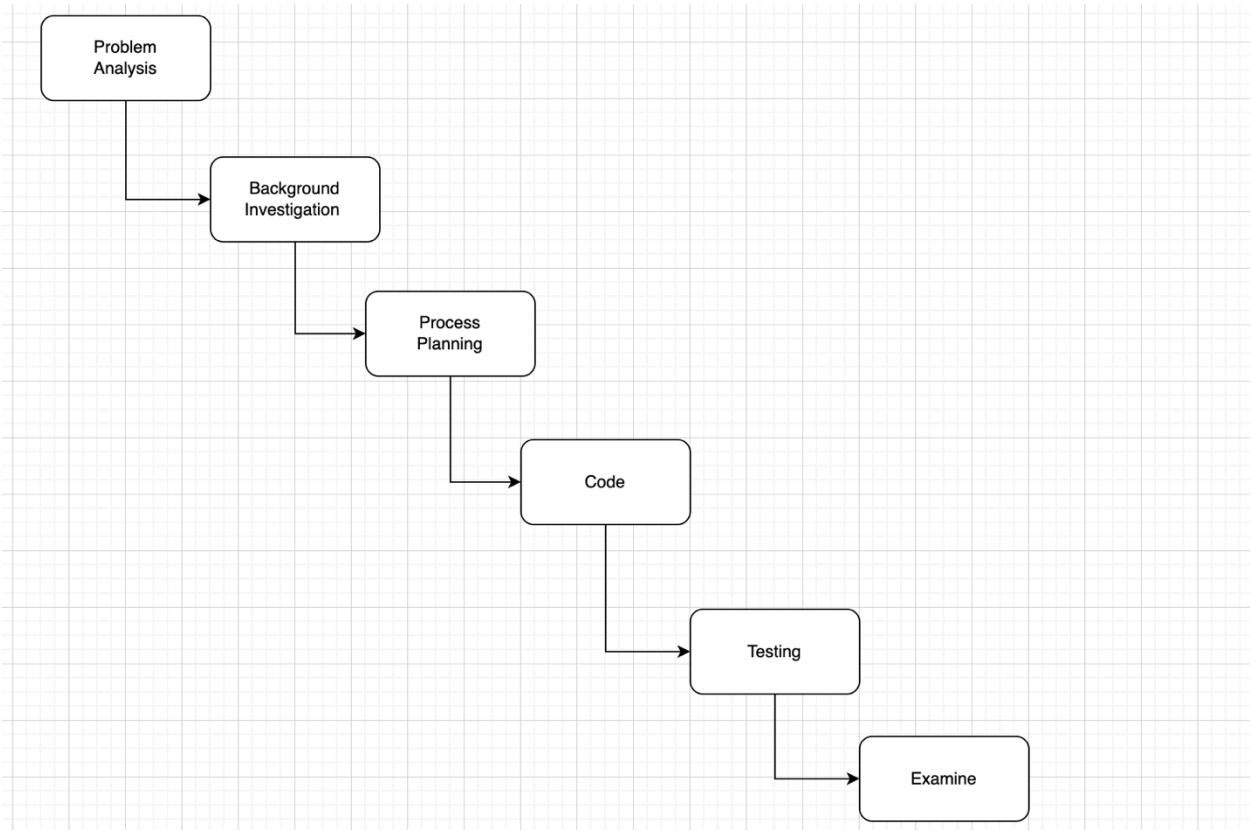


Figure 2 Waterfall Model

Chapter 2

2.0 Literature Review:

2.1 Introduction to Heart Disease and Machine Learning:

Heart disease, the leading cause of death worldwide, is a serious public health concern. Cardiovascular diseases are the most common cause of death worldwide over the last few decades in the developed as well as underdeveloped and developing countries (Dangare C S & Apte S S ,2012). They include a variety of cardiovascular disorders such as coronary artery disease, arrhythmias, and heart failure, all of which contribute significantly to morbidity and healthcare costs. Because of the intricacy and multifaceted nature of cardiac disease, improved diagnostic and prediction methods are required. Accurate detection of heart diseases in all cases and consultation of a patient for 24 hours by a doctor is not available since it requires more sapience, time, and expertise. The diagnosis of heart disease in most cases depends on a complex combination of clinical and pathological data (Chen A H, Huang S Y, Hong P S, Cheng C H & Lin E J ,2 011, September). A tentative design of a cloud-based heart disease prediction system had been proposed to detect impending heart disease using Machine learning techniques (Soni J, Ansari U, Sharma D & Soni S, 2011). Machine learning (ML), a subset of artificial intelligence, has transformed data analysis in healthcare. Because of ML's capacity to analyze large datasets and uncover complicated patterns, it is well suited for heart disease prediction and management. This technology advancement represents a huge shift in healthcare, allowing for earlier diagnosis, individualized treatment strategies, and improved patient outcomes. The incorporation of machine learning (ML) into medical research and practice heralds a new era in illness prediction.

2.2 Historical Development of Heart Disease Prediction Models:

The evolution of cardiac disease prediction models throughout time parallels the progress of medical research and computing technology. Initially, standard statistical approaches were largely used to forecast cardiac disease. The Korean CHD risk model is well-calculated alternations which can be used to predict an individual's risk of CHD and provides a useful guide to identify the groups at high risk for CHD among Koreans(Jee S H, Jang Y, Oh D J, Oh B H, Lee S H, Park S W & Yun Y D ,2014).These early models relied heavily on linear and logistic regression analyses, which, although useful for smaller datasets, frequently fell short of capturing the multidimensional

character of heart disease risk variables. As processing power became more advanced, there was a steady move toward more complicated models. The emergence of decision trees and cluster analysis in the 1980s and 1990s provided more sophisticated ways of assessing data. These strategies enabled a better understanding of how multiple risk factors for heart disease interacted. The millennium's turn signaled the start of the age of powerful machine learning approaches in heart disease prediction. Random Forests, Support Vector Machines (SVM), and Neural Networks were among the first algorithms to be used. These approaches may evaluate bigger datasets more accurately, better managing nonlinear correlations and interactions between various risk variables. An accuracy level of 97.53% accuracy was found from the SVM algorithm along with sensitivity and specificity of 97.50% and 94.94%respectively(Soni J, Ansari U, Sharma D & Soni S, 2011). Deep learning and artificial intelligence (AI) have received a lot of attention in recent years. These technologies, particularly neural networks, have exhibited remarkable pattern recognition and predictive analytics capabilities, making them perfect for difficult jobs such as heart disease prediction. Intelligent Heart Disease Prediction System (IHDPS) using datamining techniques, namely, Decision Trees, Naïve Bayes and Neural Network. is implemented using .NET platform .IHDPS is Web-based, user-friendly, scalable, reliable and expandable system(Soni Jyoti, 2012). Deep learning models can handle massive volumes of data, including unstructured data such as medical imaging and electronic health records, to provide more thorough risk evaluations. The use of big data analytics and the Internet of Medical Things (IoMT) has improved these models even more. Wearable health gadgets and sensors provide real-time health data that allows for dynamic risk assessments and individualized healthcare interventions. This progression from simple statistical tools to powerful AI-driven models demonstrates how far we have come in our understanding and approach to heart disease prediction. It represents a shift from generic statistical models to tailored, data-driven healthcare solutions, opening the door for more effective heart disease prevention and management.

2.3 Key Machine Algorithms in Heart Disease Prediction:

Several major algorithms have emerged as particularly useful in the field of heart disease prediction using machine learning. Logistic regression, a field standard, provides a solid foundation for binary classification issues such as heart disease prediction. It is especially regarded for its interpretability, which allows doctors to comprehend the role of each risk factor.

Another essential tool is decision trees, which provide a simple yet powerful way for categorization. The system uses medical terms such as sex, blood pressure, cholesterol like 13 attributes to predict the likelihood of patient getting a Heart disease(Dangare Chaitrali S and Sulabha S Apte). Because of their tree-like structure, in which each node indicates a choice based on a specific trait, they are simple to read. Decision trees, on the other hand, are prone to overfitting, especially with large datasets. With the emergence of deep learning, neural networks, a more complicated kind of machine learning, have gained significance. These algorithms excel in identifying complex patterns in massive datasets because they are inspired by the structure and function of the human brain. They are very good at dealing with high-dimensional data and have had experience integrating multiple forms of data, such as health records and imaging. Naïve bayes classifier can be trained in supervised learning setting. It uses the method of maximum similarity. It has been worked in complex real-world situation. It requires small amount of training data(Shinde R, Arjun S, Patil P & Waghmare J ,2015). Support Vector Machines (SVM) are also commonly employed in the prediction of cardiac disease. SVMs are good in determining the best border between alternative outputs, which makes them useful for classification applications. SVM classifier creates a hyper plane or set of hyper planes that can be used in high dimensional space for classification and regression analysis (Bashir S, Qamar U & Javed M Y ,2014, November). They are effective with both linear and nonlinear data relationships. Random Forests, a type of ensemble learning, mixes numerous decision trees to increase prediction accuracy while controlling overfitting. This approach is well-known for its great accuracy, resilience, and handling of imbalanced data.

Gradient Boosting Machines (GBM) are another ensemble approach that produces trees in a sequential fashion, with each tree attempting to repair the flaws of the one before it. This method has demonstrated extraordinary performance in a variety of predictive modeling contests and real-world applications. More improved approaches, such as XGBoost and LightGBM, have recently been created, providing efficient, scalable, and quicker versions of gradient boosting. Search constraints and test set validation significantly reduce the number of association rules and produce a set of rules with high predictive accuracy (Ordonez C ,2006).

Each of these techniques has advantages and disadvantages, and the decision is frequently determined by the features of the dataset, the computational resources available, and the level of interpretability necessary. Continuous improvements in machine learning algorithms and

processing capacity are improving their accuracy in cardiac disease prediction, contributing to more accurate and tailored healthcare.

2.4 Comparative Studies and Model Performance:

Several comparative studies in the field of heart disease prediction have analyzed the performance of various machine learning models, offering useful insights into their efficacy and application. These studies often assess models based on parameters such as accuracy, sensitivity, specificity, and receiver operating characteristics (ROC) area under the curve (AUC).

Research may, for example, compare classic methods such as logistic regression and decision trees to more modern techniques such as neural networks and support vector machines. Logistic regression, noted for its interpretability, may do well in terms of accuracy, but neural networks may surpass it in dealing with complicated, non-linear interactions in huge datasets.

While decision trees are simple to understand, they may have disadvantages when dealing with high-dimensional data when compared to Random Forests or Gradient Boosting Machines. While decision trees are simple to understand, they may have disadvantages when dealing with high-dimensional data when compared to Random Forests or Gradient Boosting Machines. These ensemble approaches, which combine the predictions of several trees, frequently enhance prediction accuracy and resilience, particularly in the presence of noise and outliers.

Deep learning models have been proven to excel in pattern identification, making them suited for datasets with a large number of features or complicated structures. However, their "black box" character frequently complicates clinical interpretation.

SVMs are renowned for their performance in classification problems, particularly in high-dimensional domains. They have been demonstrated to perform well in the prediction of cardiac disease, albeit careful adjustment of hyperparameters may be required. Recent research has also looked at the usage of ensemble learning approaches such as XGBoost and LightGBM, which combine the capabilities of many models. These approaches have exhibited great predicted accuracy, although they may necessitate substantial computational resources.

In addition, comparative studies frequently emphasize the relevance of feature selection and data preparation in increasing model performance. The characteristics chosen and how they are processed can have a substantial impact on the performance of various machine learning models in predicting heart disease.

Overall, these comparison studies indicate that there is no one-size-fits-all methodology for predicting heart disease. The structure of the dataset, the unique needs of the prediction job, and the trade-off between accuracy and interpretability all influence model selection. Continuous research and comparison studies are critical for directing healthcare practitioners to the most effective machine learning techniques for their unique needs in heart disease prediction.

2.5 Challenges and Limitations in Current Research:

Current guidelines do not support the use of genetic profiles in risk assessment of coronary heart disease (CHD). However, new single nucleotide polymorphisms associated with CHD and intermediate cardiovascular traits have recently been discovered. We aimed to compare several multilocus genetic risk score (MGRS) in terms of association with CHD and to evaluate clinical use(Ganna A, Magnusson P K, Pedersen N L, de Faire U, Reilly M, Ärnlöv J & Ingelsson E ,2013). Aside from data quality, model interpretability, and ethical concerns, current research in machine learning for heart disease prediction has numerous more hurdles and limitations:

2.5.1 Data Privacy and Security:

Medical data mining is used to extract knowledgeable information from a huge amount of medical data. Associative classification is a rule based new approach which integrates association rule mining and classification, if applied on medical data sets, lends them to an easier interpretation (Jabbar M A, Deekshatulu B L & Chandra P 2013, March). With the increased usage of personal health data, it is critical to ensure privacy and security. Compliance with standards like as HIPAA and GDPR is critical, but it may also limit data accessibility and exchange, affecting predictive model development and validation.

2.5.2 Imbalanced Datasets:

Heart illness datasets frequently exhibit imbalance, with much fewer instances of one class (e.g., patients with heart disease) than the other. This mismatch can lead to biased models that forecast the majority class more accurately, limiting their usefulness in real-world circumstances.

2.5.3 Integration into Clinical Workflows:

Integrating these prediction algorithms into established healthcare procedures is difficult. The models must not only be accurate, but they must also integrate easily into the healthcare system, ensuring that they supplement rather than disturb clinical decision-making.

2.5.4 Generalizability and Bias:

Models trained on certain populations' datasets may not generalize well to others. This lack of generalizability might result in skewed predictions, especially for underrepresented populations in training data.

2.5.5 Computational Resources and Accessibility:

Advanced machine learning models, particularly deep learning, need a large amount of computer power. In resource-constrained environments, this might be a hurdle, limiting the accessibility and scalability of these solutions.

2.5.6 Longitudinal Data and Dynamic Prediction:

Many models are trained using static data snapshots, but heart disease risk factors fluctuate over time. Creating models that can dynamically change predictions depending on fresh evidence is a difficult but crucial topic of future study.

2.5.7 Interdisciplinary Collaboration:

Collaboration between doctors, data scientists, and domain specialists is required for effective cardiac disease prediction models. This multidisciplinary approach is required to guarantee that models are medically relevant, technically sound, and practically feasible.

Addressing these difficulties will need a diverse strategy that includes advances in machine learning techniques, stronger data governance, and collaborative initiatives across other disciplines. Overcoming these limitations will be critical in leveraging the full potential of machine learning for cardiac disease prediction as the field advances.

2.6 Recent Advances and Future Directions:

Several unique innovations have distinguished recent improvements in machine learning for cardiac disease prediction, with future approaches promising even greater strides:

2.6.1 Deep Learning and Neural Networks:

Deep Learning and Neural Networks: A developing trend is to use deep learning, specifically convolutional neural networks (CNNs), for more accurate and nuanced processing of medical imaging data such as ECGs and echocardiograms. Recurrent neural networks (RNNs) are also being studied for time-series data such as heart rate variability.

2.6.2 Integration of Genomic Data: Genomic Data Integration: Cutting-edge research is increasingly focusing on merging genomic and proteomic data into prediction models. This

combination might result in more individualized risk assessments and treatment techniques, bringing cardiology closer to the objective of precision medicine.

2.6.3 Wearable Technology and Real-Time Monitoring: Wearable Technology and Real-Time Monitoring: Wearable gadgets that continually monitor vital indicators are becoming more popular. Future models might use this real-time data to forecast dynamic heart illness, perhaps detecting early warning indications of cardiac events.

2.6.4 Explainable AI (XAI): As machine learning models get more complicated, explainable AI is becoming increasingly important. This entails not just constructing models that forecast properly, but also providing insights into how and why specific predictions are formed, which is critical for clinical acceptability.

2.6.5 Federated Learning: Federated learning is developing as a method to solve data privacy problems. This method enables the building of resilient models by training algorithms across several decentralized devices or servers retaining local data samples that are not exchanged.

2.6.6 Combining Traditional and ML Models: Researchers are also investigating hybrid models that combine classic statistical approaches with modern machine learning techniques. These models seek to capitalize on the advantages of both techniques in order to increase forecast accuracy and interpretability.

2.6.7 AI-Driven Risk Factor Analysis: Future models may provide more thorough risk factor analysis, recognizing not just the existence of heart disease but also particular risk factors, allowing for more focused preventative methods.

These developments constitute a paradigm change in cardiac disease prediction, with the emphasis shifting from identifying illness to better understanding and avoiding it. The integration of disparate data sources, the utilization of real-time monitoring, and the creation of more transparent and interpretable models are all poised to change the face of cardiovascular healthcare.

2.7 Learnings from literature survey:

The result of a machine learning experiment on heart disease prediction includes many significant discoveries and their significance for healthcare:

2.7.1 Advancement in Predictive Models: The project has shown that advanced machine learning models, particularly those based on deep learning and neural networks, improve the accuracy and

precision of cardiac disease prediction. These models are capable of processing complicated information such as medical imaging and real-time monitoring data, providing a more nuanced knowledge of cardiovascular risks.

2.7.2 Customized Healthcare: The use of machine learning to forecast cardiac disease fits well with the trend toward customized treatment. These models offer more individualized risk assessments and treatment options by combining varied data sources, including genetic information, possibly transforming patient care.

2.7.3 Enhanced Early Detection: The experiment highlights the potential of machine learning to enhance heart disease detection. Early and correct diagnosis is critical in the management and treatment of cardiovascular disease, decreasing the total load on healthcare systems and improving patient outcomes.

2.7.4 Challenges and Future Directions: While promising, the application of machine learning in healthcare raises issues about data privacy, the need for model interpretability, and guaranteeing fair healthcare access. Future research will focus on overcoming these obstacles, establishing more explainable models, and improving the integration of real-time data from wearable devices.

2.7.5 Efforts in Collaboration: The initiative emphasizes the necessity of multidisciplinary collaboration in improving healthcare technologies. For the effective application and acceptance of machine learning models in clinical settings, collaboration among physicians, data scientists, and patients is essential.

2.7.6 Implications for Healthcare Policy and Education: The findings of this study may be used to inform healthcare policy, with an emphasis on the use of sophisticated technology in illness prediction and management. There is also a need for healthcare personnel to be educated and trained on how to use these new technologies successfully.

Finally, the effort on predicting cardiac disease using machine learning establishes a precedent for the use of modern computational approaches in healthcare. It opens the door to greater research and development, with the goal of achieving a future in which healthcare is more proactive, individualized, and data-driven.

Chapter 3

3.0 Methodology

3.1 Overview:

a) Data Collection and Preprocessing: The first stage is to collect a large dataset related to heart disease, which may include clinical, demographic, and lifestyle information. This data must be preprocessed to guarantee quality and consistency. This comprises missing value management, data normalization, and categorical variable encoding.

b) Feature Selection and Analysis:

Statistical analysis is used to identify key factors that have a substantial influence on heart disease risk. To improve the feature set, techniques like as correlation analysis and feature significance metrics are used, ensuring that the models include the most relevant predictors.

c) Model Development: Model Development: Several machine learning models are created for comparison. These are some examples:

A linear model for binary classification problems is referred to as logistic regression.

Gaussian and Bernoulli distributions Naive Bayes classifiers are probabilistic classifiers based on the Bayes theorem.

Support Vector Classifier (SVC): A sophisticated classification method that works well in high-dimensional domains.

Decision Tree Classifier: An easy-to-interpret model that separates data based on certain criteria.

K Neighbors Classifier: A non-parametric classification approach based on the feature space's closest training samples.

Random Forest Classifier: A decision tree ensemble noted for its excellent accuracy and resilience against overfitting.

Ada Boost Classifier: A strategy that combines weak learners to produce a powerful classifier.

Another ensemble approach that fits base classifiers on random subsets of the original dataset is the bagging classifier.

Gradient Boosting Classifier: A boosting approach in which models are built consecutively, with each new model correcting faults caused by prior ones.

XGBoost Classifier: A distributed gradient boosting library that has been tuned for performance, versatility, and portability.

d) Model Training and Validation:

Each model is trained and validated on the dataset using techniques such as cross-validation. Metrics like as accuracy, precision, recall, and the area under the ROC curve are used to assess model performance.

e) Hyperparameter Tuning: Each model's hyperparameters are fine-tuned to enhance performance. For systematic optimization, techniques such as Grid Search and Randomized Search can be utilized.

f) Model Selection: Each model's hyperparameters are fine-tuned to enhance performance. For systematic optimization, techniques such as Grid Search and Randomized Search can be utilized.

This technique describes a thorough strategy to designing a heart disease prediction application that takes advantage of the characteristics of multiple machine learning algorithms to produce accurate and trustworthy predictions.

3.2 Logistic Regression:

Selecting significant characteristics from the dataset that impact heart disease risk would be the first step in the Logistic Regression process. Patient demographics, blood pressure levels, cholesterol levels, diabetes, smoking status, and other factors may be considered. A subset of the dataset would be used to train the Logistic Regression model. This entails utilizing the chosen characteristics to predict a binary result - whether or not a patient has heart disease. Logistic Regression does this by estimating probabilities with a logistic function, which is especially useful for binary classification tasks. Following training, the model would be validated on a different set of data not utilized in training. This evaluation might make use of metrics such as the confusion matrix, accuracy, precision, recall, and AUC-ROC curves. The interpretability of Logistic Regression is one of its benefits. The model coefficients can provide information on how each trait influences the risk of heart disease, which is useful for clinical decision-making. The Logistic Regression model would be utilized in the application to forecast the risk of heart disease for new patients based on their input data, giving healthcare practitioners with a tool to help in diagnosis and preventative methods.

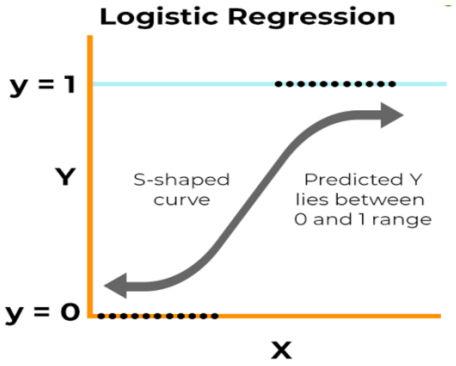


Figure 3 (Source Google)

3.3 Gaussian NB:

Gaussian NB was chosen owing to its effectiveness in handling datasets with continuous characteristics believed to follow a normal (Gaussian) distribution. It's very beneficial when dealing with medical data such as blood pressure measurements, cholesterol levels, and so on. The data is preprocessed before Gaussian NB is applied. Handling missing values, normalizing continuous variables, and possibly converting categorical data to numerical format are all part of this. Gaussian NB operates with the assumption that the characteristics are independent of each other. This involves considering each risk factor (such as age, cholesterol levels, and smoking status) as an independent input to the model in the context of heart disease prediction. The Gaussian NB model is trained using the preprocessed dataset. This entails calculating the mean and variance of the features for each class and then using these parameters to estimate the likelihood of the data. The model estimates the chance of a person having or not having heart disease. It employs Bayes' theorem to determine the probability, taking into consideration the prior probability of each class (heart disease or no heart disease) and the likelihood of the observed data. Following training, the model is evaluated using metrics such as accuracy, precision, recall, and AUC. This aids in understanding how well the Gaussian NB model predicts heart disease. If the model produces promising results, it is integrated into the heart disease prediction application and can be used to assess new patient data. Based on comments and fresh data, the model may be regularly updated to increase its predicted accuracy.

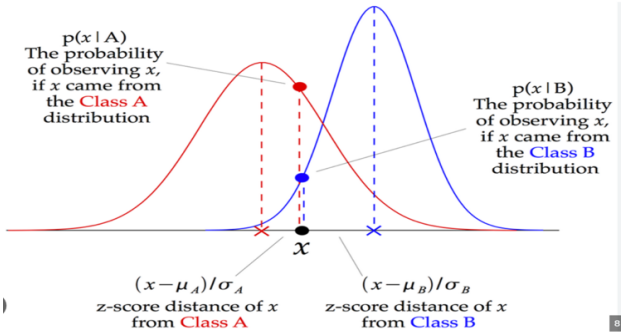


Figure 4 (Source Google)

3.4 Bernoulli NB:

The Bernoulli Naive Bayes classifier is especially well-suited to binary/boolean features. Data preparation in the context of heart disease prediction would entail transforming the dataset to a binary format. This might imply converting continuous variables to binary variables depending on certain thresholds or criteria. This binary dataset would be used to train the Bernoulli Naive Bayes model. Using the Bayes theorem, the model determines the probability of the existence or absence of cardiac disease based on the binary characteristics. The assumption of independence among characteristics is an essential part of Naive Bayes. The algorithm would separately examine each feature's significance to predicting heart disease. Following training, the model would be used to forecast the risk of heart disease in new, previously unreported data. The model's performance is often validated using metrics such as accuracy, precision, recall, and area under the ROC curve.

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)}$$

Likelihood, Prior, Posteriori, Predictor Prior

Figure 5 (Source Google)

3.5 Support Vector Classifier(SVC):

The initial step would be to prepare the data by cleaning, standardizing, and maybe translating it into a format appropriate for machine learning models. Encoding categorical variables and scaling

numerical characteristics are examples of this. The most important characteristics that indicate heart disease would be found. Based on domain knowledge, statistical analysis, or feature significance scores, this might be done. A machine learning library such as scikit-learn would be used to create the Support Vector Classifier. This entails setting up an SVC model with the appropriate hyperparameters. The kernel type (linear, polynomial, radial basis function, etc.), the penalty parameter C, and other SVC hyperparameters would be modified to discover the optimum combination for the model. This might be accomplished through the use of techniques such as Grid Search or Randomized Search. A portion of the data would be used to train the SVC. Fitting the model to the training data and then using it to generate predictions is what this entails. The performance of the model would be assessed using relevant measures, such as accuracy, precision, recall, F1 score, and ROC-AUC score. The model would most likely be evaluated using a separate test dataset that was not observed by the model during training. Finally, the SVC findings would be analyzed in terms of heart disease prediction. If the model is successful, it may be integrated into a larger system or application for practical usage.

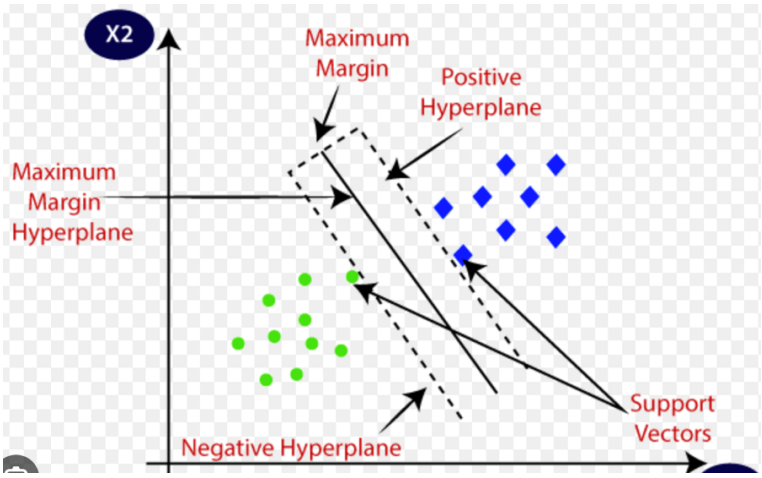


Figure 6 (Source Google)

3.6 Decision tree Classifier:

At first, significant factors strongly connected with heart disease, such as age, cholesterol levels, blood pressure, heart rate, and so on, would be chosen. The Decision Tree Classifier excels at processing both numerical and categorical data. Typically, the dataset would be separated into training and testing sets. The decision tree is built using the training set, and its performance is

evaluated using the testing set. On the training dataset, the Decision Tree Classifier is trained. It generates a decision tree-like model in which each node represents a feature in the dataset and each branch represents a decision rule that leads to an outcome (presence or absence of heart disease). To minimize overfitting and increase model accuracy, parameters such as tree depth, the minimum number of samples necessary to split an internal node, and the minimum number of samples required to be at a leaf node may be set. The model's performance is tested using the test dataset once it has been trained. To test its usefulness in predicting cardiac disease, metrics like as accuracy, precision, recall, and the confusion matrix are commonly utilized. The interpretability of a Decision Tree Classifier is one of its benefits. The decision tree that results may be displayed, making it easier to see how different factors impact the model's predictions.

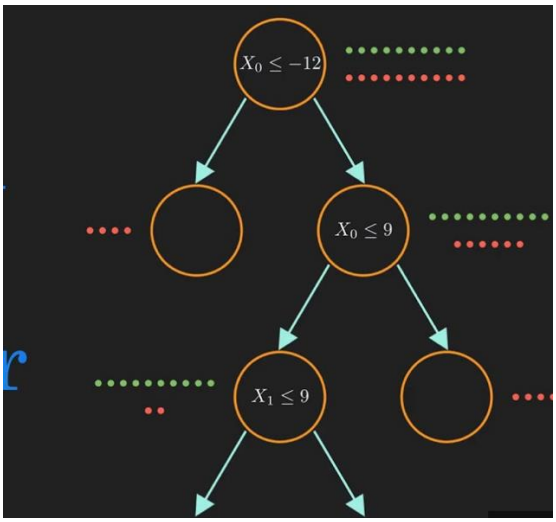


Figure 7 (Source Google)

3.7 K-Neighbors Classifier:

Cholesterol levels, blood pressure, age, and other factors that impact heart disease prediction are chosen from the dataset. To guarantee that distance computations in the KNN method are not biased by the scale of the data, the selected features are preprocessed, which may involve normalization or scaling. A subset of the dataset is used to train the K-Neighbors Classifier. This entails storing the training data's characteristics and accompanying labels. To balance the bias-variance tradeoff, the ideal number of neighbors (K) is found, frequently by cross-validation. The model selects the K nearest neighbors in the training data to a new data point for prediction, and

the most frequent label among these neighbors is assigned to the new data point. Metrics like as accuracy, precision, recall, and the confusion matrix are used to assess the classifier's performance.

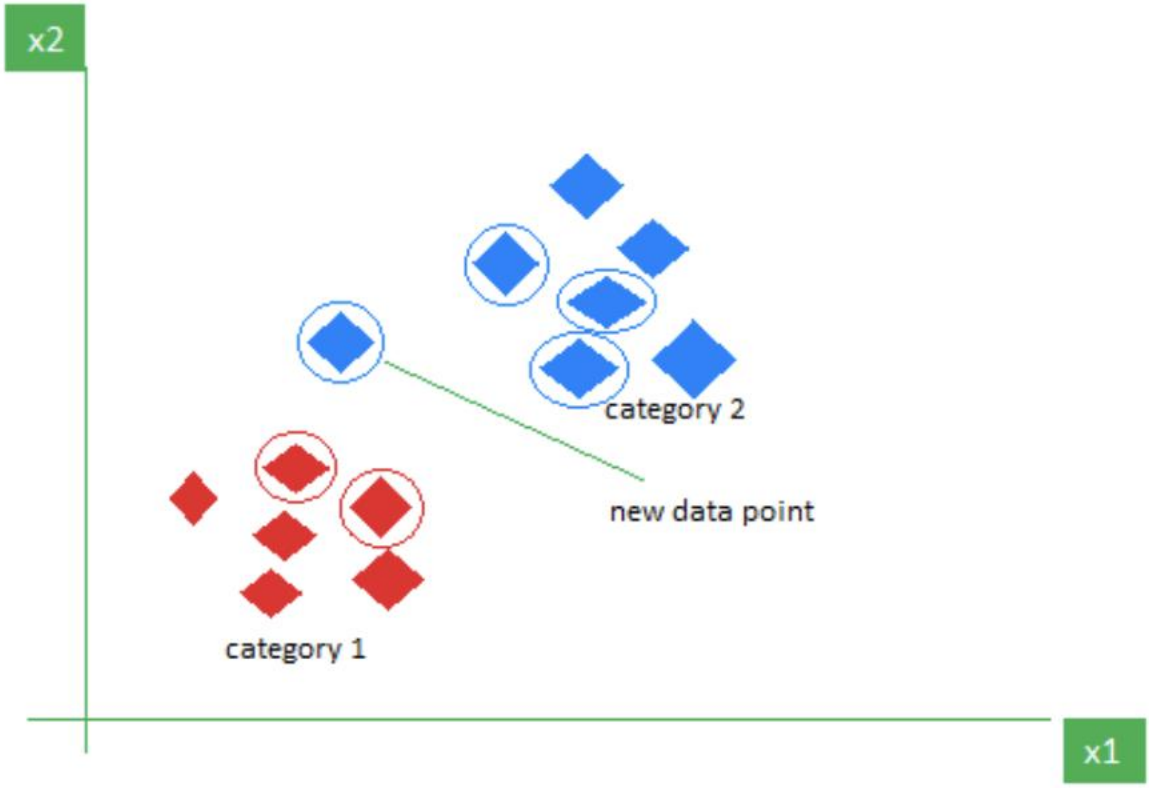


Figure 8 (Source google)

3.8 Random Forest Classifier:

The first step is to get the data ready for the Random Forest Classifier. Cleaning the data, dealing with missing values, normalizing numerical characteristics, and encoding categorical variables are all part of this process. Because of its capacity to rate the value of numerous features for the prediction job, Random Forest is useful in feature selection. It aids in determining which risk factors for heart disease are more prevalent. On the preprocessed dataset, the Random Forest Classifier is trained. At training time, this ensemble learning approach constructs a large number of decision trees and outputs the class that is the mode of the classes (classification) of the individual trees. To maximize the Random Forest model's performance, key parameters such as the number of trees in the forest (n_estimators), the maximum depth of trees, and the minimum amount of samples necessary to split an internal node are set.The Random Forest Classifier's performance is assessed using measures such as accuracy, precision, recall, F1-score, and area

under the ROC curve. The model's efficacy and generalizability may be evaluated via cross-validation.

Finally, the trained Random Forest model is used to generate predictions on new data in order to identify people at risk of heart disease.

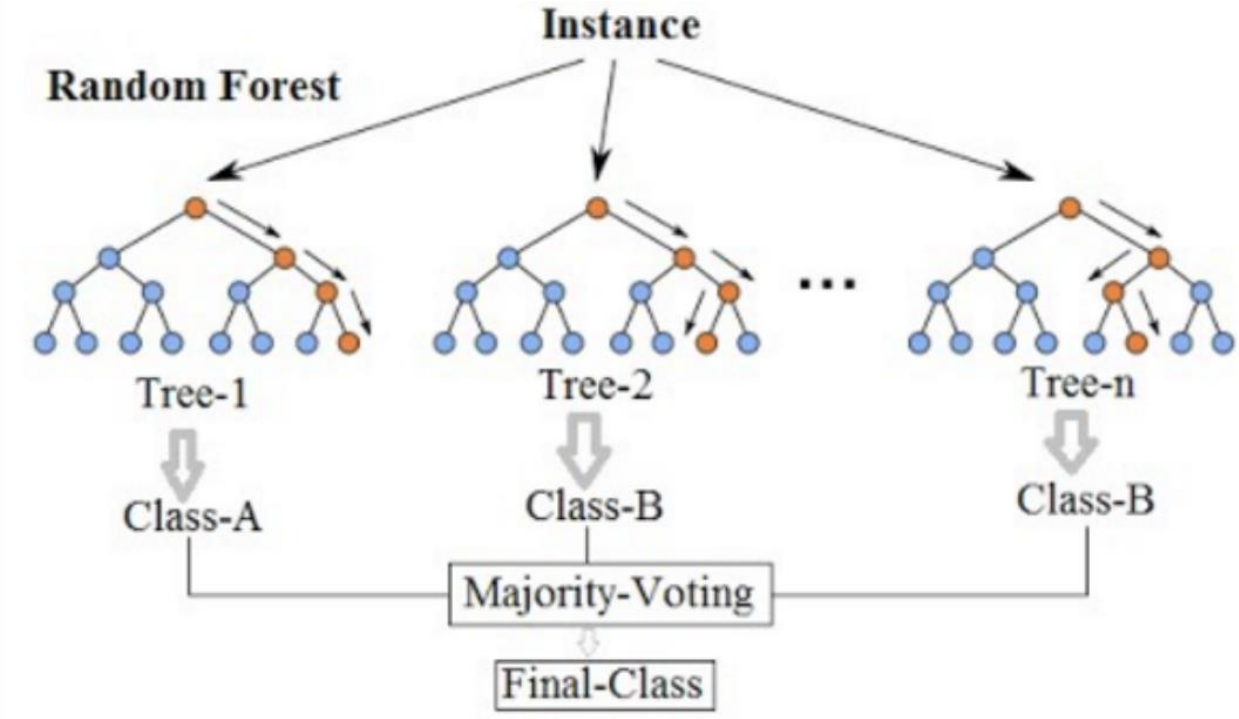


Figure 9 (Source google)

3.9 Ada Boost Classifier:

The data must be preprocessed before utilizing the AdaBoost Classifier. This comprises partitioning the dataset into features and labels, dealing with missing values, normalizing or standardizing the data, and partitioning the dataset into training and testing sets. The AdaBoost Classifier is initialized, often using default or provided hyperparameters. AdaBoost, which stands for Adaptive Boosting, creates a powerful classifier by merging many weak classifiers. The dataset is used to train the AdaBoost model. AdaBoost adds weights to each training instance during training. It gives extra weight to examples that were misclassified in prior rounds, causing the model to focus on tough situations in later iterations. Important hyperparameters for AdaBoost parameters, such as the number of weak learners (trees) and the learning rate, can be tweaked to improve the model's performance. Cross-validation and grid search are frequently employed for

this purpose. The model is tested on the test set after training using measures such as accuracy, precision, recall, F1 score, and area under the ROC curve. This assessment aids in comprehending the model's efficacy in forecasting cardiac disease. AdaBoost can also reveal which characteristics are most relevant in predicting heart disease, which is useful for clinical knowledge and further study. If the AdaBoost model produces positive findings, it may be used in a clinical context or incorporated into a healthcare application to forecast the risk of heart disease.

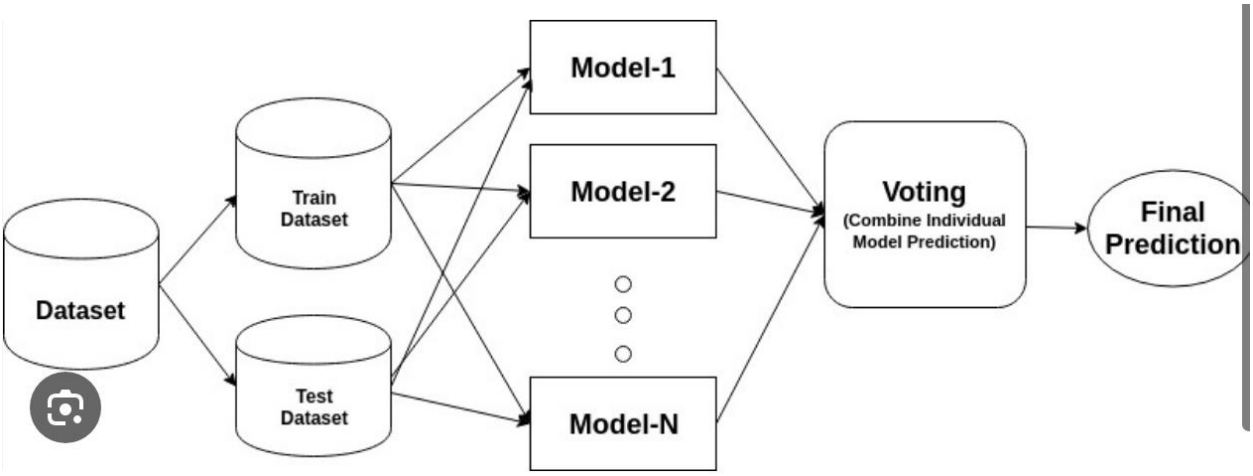


Figure 10 (Source google)

3.10 Gradient Boosting Classifier:

The Gradient Boosting Classifier would be trained on a dataset including numerous risk factors for heart disease. This dataset might comprise, among other things, patient demographics, blood test results, lifestyle factors, and medical history. Gradient Boosting is well-known for its capacity to deal with a huge number of variables and automatically identify the most relevant predictors of heart disease. These characteristics are used by the model to learn and forecast the chance of a patient acquiring heart disease. It accomplishes this by gradually assembling an ensemble of weak prediction models, often decision trees. Hyperparameters such as the number of trees, learning rate, and depth of the trees would be fine-tuned to improve the model's performance. The Gradient Boosting Classifier's ability to predict heart disease would be measured using measures such as accuracy, precision, recall, and the area under the ROC curve. Cross-validation methods such as k-fold cross-validation may be used to validate the model's efficacy.

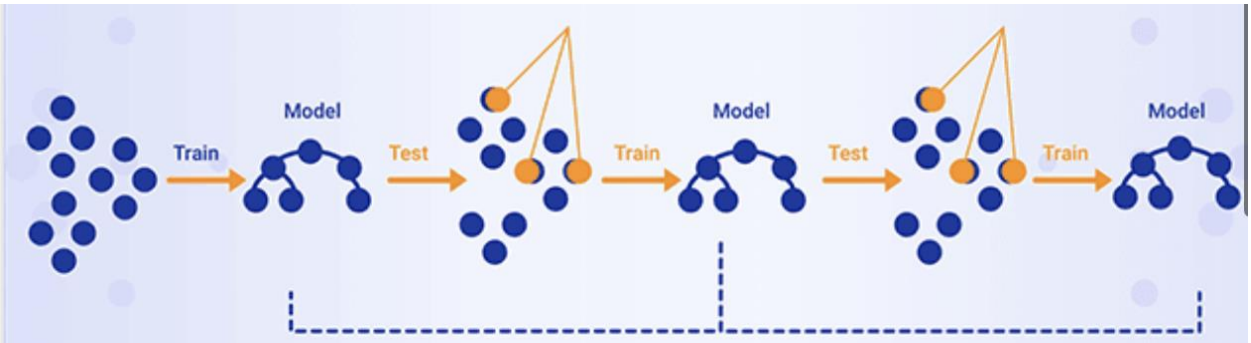


Figure 11 (Source google)

3.11 XGBoost Classifier:

Typically, the dataset would be preprocessed before being applied to the XGBoost Classifier. This comprises missing value management, categorical variable encoding, and numerical feature normalization.The next stage might be to identify the most important characteristics for predicting heart disease. This decision can be made based on domain knowledge or on feature significance ratings supplied by preliminary XGBoost model runs. The transformed dataset would then be used to train the XGBoost Classifier. XGBoost is well-known for its efficiency and performance, and it excels at handling huge and complicated datasets. It can also handle unbalanced datasets, which are typical in medical data. Hyperparameters such as learning rate, maximum depth of trees, and number of trees would be changed to improve the XGBoost model. Grid Search or Randomized Search techniques might be used for this purpose. The XGBoost model's performance would be assessed using relevant measures such as accuracy, precision, recall, F1-score, and the AUC-ROC curve. This is an important stage in determining how effectively the model predicts cardiac disease. Given the significance of interpretability in healthcare applications, the project may additionally include understanding the predictions of the XGBoost model. This might be accomplished with the use of techniques such as SHAP (SHapley Additive exPlanations), which aid in understanding the contribution of each feature to the model's predictions. Finally, if the XGBoost model produces positive results, it will be included into the heart disease prediction program, making it available for usage by healthcare professionals or patients.

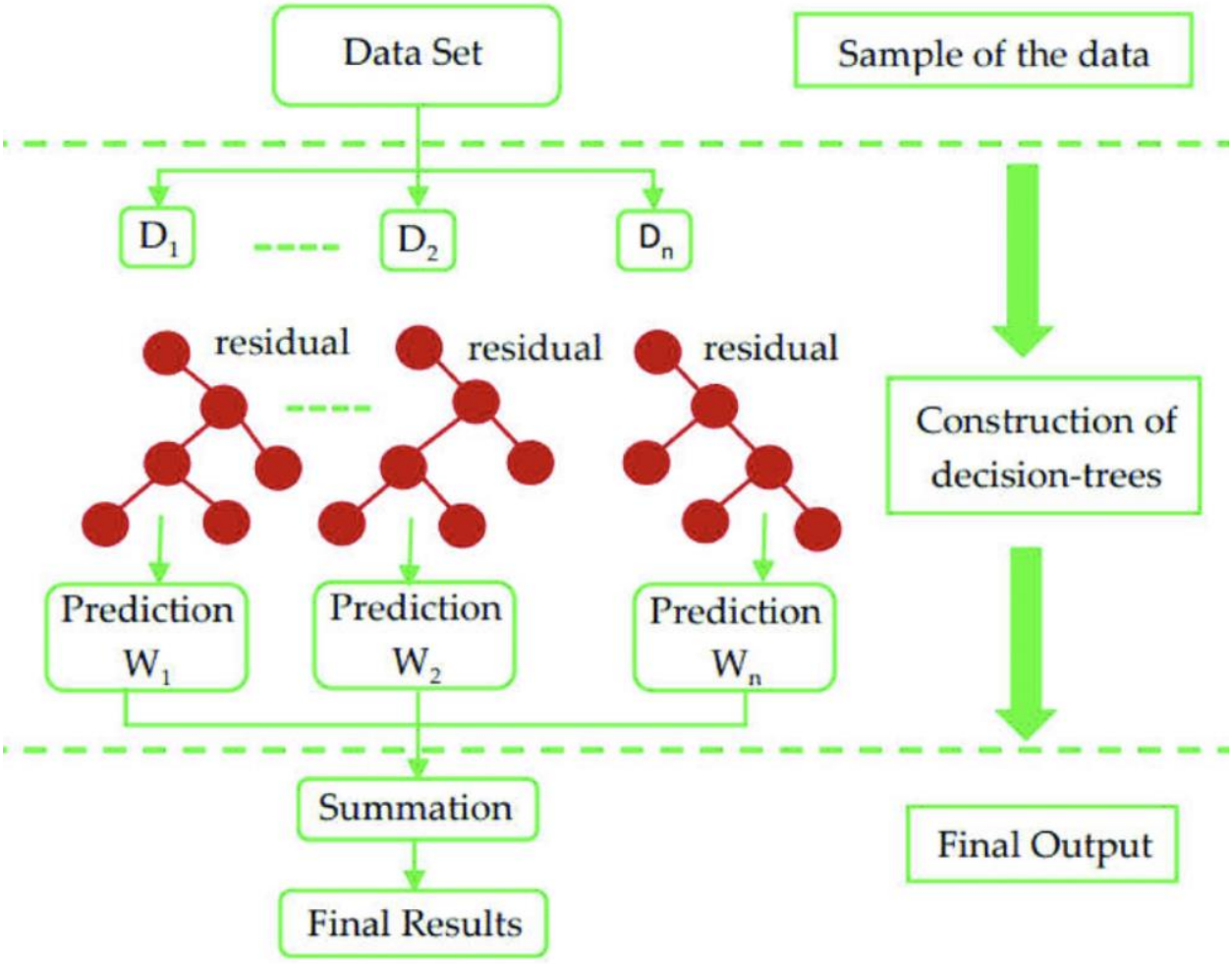


Figure 12 (Source google)

Chapter 4

4.0 Working on Data Sets

4.1 Data collection:

4.1.1 Electronic medical records (EMRs):

These are a great source of data for predicting cardiac disease. They offer specific patient information that is critical for studying the numerous variables that influence cardiac disease. EMRs' comprehensive nature enables a comprehensive approach to illness prediction and management. Demographics of the Patients: Basic demographic information such as age, gender, ethnicity, and socioeconomic status are frequently included in EMRs. These variables are important because they can impact the chance of acquiring heart disease. Age and gender, for example, are established risk factors, with older people and specific genders being predisposed to heart disease. The medical history of a patient, as recorded in EMRs, comprises past diagnoses, treatments, and hospitalizations. This data gives insight into comorbidities that may raise the risk of heart disease, such as diabetes or hypertension.

EMRs capture laboratory tests such as cholesterol levels, blood pressure measurements, blood sugar levels, and other indicators related to heart health. Certain biomarkers with elevated levels can suggest an increased risk of heart disease. EMRs also include information on a patient's treatments, such as drugs, surgical procedures, and lifestyle advice. This data can be used to assess the efficacy of various heart disease treatment options.

Some EMRs incorporate information on lifestyle characteristics such as smoking status, alcohol usage, food, and levels of physical activity. These characteristics contribute significantly to heart health and can be utilized to develop individualized preventative measures. Because cardiac disease frequently has a hereditary component, information regarding family medical history, if available, might give further insights. Results from diagnostic tests such as ECGs, echocardiograms, and angiograms may be included in EMRs, providing immediate insight into a patient's cardiovascular health. Integration with wearable health equipment enables EMRs to include real-time monitoring data, such as continual heart rate or blood pressure readings, resulting in a more dynamic and up-to-date picture of a patient's heart health. Using EMR data for heart disease prediction requires overcoming data privacy issues, uniformity across various healthcare providers, and guaranteeing the quality and completeness of the records.

4.1.2 Public Datasets:

Publicly available datasets are essential in heart disease prediction studies, acting as significant tools for researchers all around the world. These databases, which are frequently offered by healthcare facilities, research groups, or government bodies, provide a wide range of information that might be useful in understanding and forecasting cardiac disease.

Public datasets are widely available, giving them an excellent starting place for researchers, particularly those with low resources or in the early phases of their research. These datasets often include a wide range of data, such as demographic information, medical histories, lifestyle variables, and clinical assessments. This variety allows researchers to investigate different elements of cardiac disease and its predictions. Public datasets frequently comply to particular data collecting and processing standards, assuring a level of quality and uniformity. This uniformity is essential for comparative research and meta-analyses.

They serve as industry standards. These datasets may be used by researchers to test and evaluate the efficiency of various machine learning models and methods. The availability of shared datasets enables researcher collaboration and aids in the validation and replication of scientific discoveries, which is a cornerstone of scientific research. Typically, these datasets are anonymized and curated with ethical issues in mind, allowing researchers to use them without jeopardizing patient privacy. The analysis of these datasets allows academics to discover insights applicable to real-world circumstances, increasing the practical worth of their study. However, researchers must be mindful of the limits of public databases, such as inherent biases, missing data, and sample representativeness. As technology advances, more data from wearable devices and digital health records are being included in public databases, enabling more complete insights into heart health. Public datasets in cardiac disease prediction are crucial for stimulating innovation, advancing disease understanding, and driving advances in predictive modeling. Their importance in developing effective and reliable prediction models cannot be emphasized, and they have become a cornerstone of cardiovascular research.

4.1.3 Clinical Trials:

Data from clinical trials or epidemiological studies focusing on cardiovascular health are critical sources of information for heart disease prediction research. These sources give abundant,

scientifically rigorous data that can considerably improve knowledge of the dynamics of cardiac disease. Clinical trials collect thorough data on patient reactions, side effects, and results. They are frequently used to examine the efficacy of novel therapies or interventions. These studies are often well-controlled and provide high-quality data. This data can give insights into how different variables affect heart disease outcomes under certain situations for machine learning. Epidemiological studies provide a larger perspective, frequently exploring the trends, causes, and impacts of health and illness situations in specified groups. They can give complete data on heart disease risk factors and incidence rates, which is critical for predictive modeling.

Longitudinal data gathering is used in both clinical trials and epidemiological studies, which follow patients across time. This is especially useful in understanding the course of cardiac disease and the long-term effects of numerous risk factors. Epidemiological studies frequently contain a varied population, collecting data from various demographic and geographic groupings. This variety contributes to the development of more generalizable and inclusive models. Data from these sources is considered ‘real-world evidence' since it reflects genuine patient experiences and results. For its practical application, RWE is increasingly appreciated in healthcare studies, particularly heart disease prediction. They can give complete data on heart disease risk factors and incidence rates, which is critical for predictive modeling.

Longitudinal data gathering is used in both clinical trials and epidemiological studies, which follow patients across time. This is especially useful in understanding the course of cardiac disease and the long-term effects of numerous risk factors. Epidemiological studies frequently contain a varied population, collecting data from various demographic and geographic groupings. This variety contributes to the development of more generalizable and inclusive models. Data from these sources is considered ‘real-world evidence' since it reflects genuine patient experiences and results. For its practical application, RWE is increasingly appreciated in healthcare studies, particularly heart disease prediction. Clinical trial and epidemiological data are subject to strong ethical rules and permission requirements, ensuring patient rights and privacy are protected. Researchers must deal with issues such data heterogeneity, unpredictability in data quality, and potential biases in research designs.

These data sources are frequently the product of joint work across research institutes, hospitals, and universities, which fosters a collaborative research environment. Initiatives such as data sharing platforms and research consortia are making clinical and epidemiological data more

available to researchers all around the world. To build more complete datasets for predictive modeling, there is a rising trend of merging data from traditional clinical trials with real-world data sources.

4.1.4 Primary and Secondary Data:

If we want to use primary data, we must approach a specific institution and ask for ethical approval to use the data for official objectives such as research, project goals, and so on. As part of a contractual deal. In these circumstances, data may be collected by numerous methods such as interviews, surveys, advertising, questionnaires, and so on. If the data we are planning to utilize is secondary, we do not need to obtain any permissions because this type of data is already provided by the source or other data-providing sources under legal process. Data may be acquired via government websites, third-party sources like GitHub, Kaggle, tech data, and so on, as well as data offered by commercial suppliers like APIs.

4.1.5 Data Specifications of data used for heart disease prediction:

The data set used in the prediction for heart disease is basically collected from Kaggle data Source.

4.2 Data preprocessing process:

4.2.1 Handling Missing Values:

Handling missing values is a crucial feature of data preparation in machine learning, and it is especially important in healthcare datasets such as those used to forecast heart disease. The method used to deal with missing data can have a substantial influence on the performance and reliability of predictive models. Here's a more in-depth look at dealing with missing values.

Before employing any approach, it is critical to comprehend why data is absent. Missing data is classified as either missing completely at random (MCAR), missing at random (MAR), or missing not at random (MNAR). Depending on these categories, the technique for dealing with missing data may change. While basic, it is useful for datasets with randomly distributed missing values and does not dramatically distort the data distribution. Based on comparable data points, KNN imputation restores missing values. It uses distance metrics to determine the 'K' nearest data points and imputes missing values based on the mean or median of these neighbors. This approach is more advanced and can yield more accurate imputations than mean/median/mode imputation,

especially in datasets with visible clusters of data points. Missing values are anticipated using this approach based on observed values. To forecast missing values, a regression model is developed utilizing characteristics and full data. While this strategy is useful, it might add bias if the model overfits the observed data. Multiple imputation, as opposed to single imputation, produces several imputed datasets. The findings of these datasets are then combined for analysis. This approach is advantageous because it accommodates for ambiguity in imputations and produces more robust findings. Certain algorithms, such as Random Forests, can manage missing values on their own. They can divide nodes based on the available data or impute values based on the tree structure. Delete may be a possibility in some circumstances, particularly when the fraction of missing data is tiny. This can be done either listwise (removing whole records if any single value is absent) or pairwise (using all available data for each computation). This strategy, however, runs the danger of erasing important data. Deep learning techniques such as autoencoders are used in advanced ways to imput missing values. These algorithms may detect complicated patterns and correlations in data, perhaps enabling more accurate imputations. It is critical to examine the influence of missing data handling on the model after imputation. This entails evaluating model performance on datasets with and without imputation, as well as determining if the imputation procedure produced any bias. In conclusion, dealing with missing values is a vital step that needs careful analysis and a grasp of the dataset's properties as well as the underlying causes of missingness. The approach used should be compatible with the nature of the data as well as the unique needs of the heart disease prediction task at hand.

4.2.3 Data Cleaning:

Data cleaning is an important step in the data pretreatment phase, particularly in healthcare-related initiatives such as heart disease prediction. It assures the data's integrity and quality, which has a direct impact on predictive model performance. Several critical activities are usually included in the process: It is critical to identify and remove duplicate records. Duplicates can occur for a variety of reasons, including data input mistakes or the merging of entries from numerous sources. They can provide biased findings and reduce the model's accuracy. This entails detecting and rectifying data problems. Typographical, labeling, or data entry errors are all possibilities. Impossible values, such as a negative age or an unreasonable blood pressure result, must be rectified or eliminated, for example. Outliers are data points that deviate dramatically from the rest

of the observations. They might be the result of measuring mistakes or truly exceptional instances. Outliers may be identified using statistical approaches such as Z-scores and IQR (Interquartile Range), as well as visual methods such as box plots. Outliers must be handled carefully; they can be deleted, converted, or suitably addressed to avoid skewing the results. Data validation rules can be used to guarantee that data fulfills particular quality criteria. Checks for valid ranges (e.g., age between 0 and 120), obligatory fields (ensuring critical data such as patient ID is included), and format checks (e.g., ensuring dates are in the right format) are examples of these rules. It is critical to ensure consistency across the dataset. Inconsistencies in data recording may arise, especially if data is acquired from numerous sources. This procedure includes the standardization of words and units (for example, translating all measurements to a single unit). While technically part of data cleansing, dealing with missing data is an important issue that demands special attention. It entails tactics like as imputation, deletion, or the use of algorithms that can deal with missing information. This may entail converting data into an analysis-ready format. Categorical data, for example, may need to be encoded, while continuous variables may need to be binned. Integration is critical for combining data from many sources to create a single dataset. This entails matching data from several sources and addressing inconsistencies. Each stage of the data cleansing process must be documented for openness and repeatability. This documentation should include the methods employed as well as the reasoning behind each decision. The data must be checked for quality assurance once it has been cleaned. This may entail statistical analysis to guarantee that the data is clean and suitable for usage. The necessity of proper data cleansing cannot be emphasized in heart disease prediction, where data-driven decisions might have serious health consequences. This technique not only improves the model's accuracy but also increases the dependability and trustworthiness of the model's forecast insights.

4.2.4 Data Transformation:

Data transformation is an important preprocessing step in the context of heart disease prediction models. It entails transforming the raw data into a more acceptable format or structure for analysis, particularly using machine learning models. In addition to the fundamental logarithmic or square root transformations, the following strategies are used to increase the model's performance: This procedure adjusts numerical values in the dataset to a common scale while preserving variances in value ranges. Min-max normalization, for example, rescales data to a 0-1 range, which is very

beneficial for algorithms that are sensitive to data scale, such as neural networks. Standardization (normalization of Z-scores): This method produces data with a mean of zero and a standard deviation of one. It is critical for models. Support Vector Machines and k-nearest neighbors, for example, assume that all features are centered around zero and have variance in the same order. Box-Cox and Yeo-Johnson transformations are examples of power transformations. They are more adaptable than standard logarithmic or square root transformations and may be used with data that contains negative values (Yeo-Johnson) or requires a more dynamic transformation method (Box-Cox). Discretization (Binning): This process includes converting continuous variables to discrete categories, which can help to simplify the model and comprehend the impacts of different variable ranges. category Encoding: Techniques such as one-shot encoding and label encoding convert category data to numerical representation. Because most machine learning models cannot handle categorical data in their raw form, this is critical. Feature Engineering is a process that entails Support Vector Machines and k-nearest neighbors, for example, assume that all features are centered around zero and have variance in the same order. Box-Cox and Yeo-Johnson transformations are examples of power transformations. They are more adaptable than standard logarithmic or square root transformations and may be used with data that contains negative values (Yeo-Johnson) or requires a more dynamic transformation method (Box-Cox). Discretization (Binning): This process includes converting continuous variables to discrete categories, which can help to simplify the model and comprehend the impacts of different variable ranges. category Encoding: Techniques such as one-shot encoding and label encoding convert category data to numerical representation. Because most machine learning models cannot handle categorical data in their raw form, this is critical. Feature Engineering is a process that entails Each of these data transformation approaches has a place and value based on the qualities of the data at hand and the needs of the predictive model being utilized. The modification used can have a considerable impact on the model's performance, hence a mix of these strategies is frequently used to properly prepare the dataset for the heart disease prediction job. Data that has been properly converted may lead to more accurate, efficient, and interpretable models, which is critical in healthcare applications because prediction accuracy can have a direct impact on patient outcomes.

4.2.5 Data Splitting:

Data splitting is an important step in preparing a dataset for use in machine learning models, particularly in applications such as heart disease prediction. It entails separating the dataset into different subsets, which are commonly referred to as the training set and the testing (or validation) set. This section is critical to the creation of an accurate and dependable prediction model. Here's a more detailed explanation: The fundamental purpose of data splitting is to test the model's performance on unknown data, hence measuring its generalizability. The training set is used to train the model, while the testing set is used to assess its prediction power.

The training set is used to train the machine learning model. It contains both the input features and the target features. The size of the training set has a substantial influence on the model's learning; a bigger training set often gives more information, resulting in a better-trained model.

The testing set is used to assess the performance of the model. This data must not be used during the training phase. The performance of the model on the testing set predicts how it will perform in real-world circumstances. Validation Set: A third subset called the validation set is sometimes utilized, especially in complicated models like deep learning. This collection aids in the fine-tuning of model parameters and the prevention of overfitting. It functions as a stopgap between training and testing, ensuring that the model is not underfit or overfit. Split Ratios: The ratio at which the dataset is divided into training and testing sets can be adjusted. 70:30, 80:20, and 90:10 (training: testing) are common splits. The ratio used is determined on the quantity of the dataset and the nature of the challenge. Cross-Validation: For a more robust assessment, cross-validation might be employed instead of a basic train-test split. The data is divided into 'k' subsets in k-fold cross-validation, and the model is trained and assessed 'k' times, each time using a different subset as the test set and the rest as the training set. Stratified Splitting: For unbalanced datasets in which one class is greatly underrepresented, stratified splitting assures that the distribution of classes in both training and testing sets is identical to that of the original dataset. Splitting data chronologically rather than randomly might be more efficient in datasets where temporal variables are relevant (such as time-series data). influence on Model Performance: The way data is partitioned can have a major influence on model performance. Incorrect splitting might result in biased or erroneous models.

Data splitting may be done effectively using machine learning libraries such as Scikit-learn, which provide methods for separating datasets into training and testing sets.

To summarize, data splitting is an important step in model development since it ensures that the heart disease prediction model is evaluated under real-world situations. It is a preventative measure against overfitting and a necessary step in validating the model's prediction power.

Chapter 5

5.0 Model Designing and structure:

5.1 Major Steps involved in designing the model:

The Heart Disease Prediction project takes a comprehensive approach to model creation and structure, relying on a variety of machine learning models. Preprocessing the dataset is the first step, followed using several classifiers such as Logistic Regression, Gaussian NB, Bernoulli NB, Support Vector Classifier, and others. To establish efficacy, each model is trained and evaluated on the dataset, with an emphasis on metrics such as accuracy and precision. The most appropriate model is chosen based on its performance, ensuring that the option matches with the objective of properly forecasting cardiac disease. This multi-model method enables a thorough examination of several predictive techniques, improving the resilience and reliability of the final prediction model.

Step 1: Uploading the data:

A large dataset helps train the heart disease prediction algorithm to discover patterns and connections between characteristics and the existence of heart disease. The dataset is essential for assessing the model's performance on fresh, previously unknown data in real-world circumstances. A varied dataset allows the model to generalize effectively while avoiding overfitting to training data.

Step 2: Checking for Nulls:

In heart disease prediction, checking for nulls maintains data integrity and eliminates biased predictions. Filling in the blanks enhances model performance and feature choices. Understanding null patterns improves dataset quality and reduces bias. The proper treatment of nulls is critical for making accurate and fair forecasts about heart disease.

Step 3: Data Preparation:

In addition to verifying for null values and translating age from days to years, the Heart Disease Prediction concentrates on preparing and normalizing critical physiological indicators such as

systolic blood pressure. This entails verifying that the dataset's units and scales are consistent. Furthermore, the team may examine the distribution of these parameters to discover any potential abnormalities or outliers that might impair model accuracy. Addressing these issues is critical for preserving data quality and ensuring that machine learning model inputs are trustworthy and indicative of real-world settings.

Step 4: Data Analysis and Visualization:

The Heart Disease Prediction project's data processing and visualization phase goes beyond simple patterns. I investigated more intricate linkages within the data using Matplotlib. This involves examining the relationships between numerous risk variables, such as cholesterol levels, smoking behaviors, and exercise frequency, and the occurrence of heart disease. These visualizations help in the discovery of hidden patterns and insights, which are required for the development of an accurate prediction model. They also give a clear visual view of the data, allowing for simpler communication of results and insights to stakeholders with less technical knowledge.

Step 5: Feature Selection and Engineering:

The Heart Disease Prediction project relies heavily on feature selection and engineering to improve model accuracy. To identify relevant aspects, I used approaches such as correlation analysis and principal component analysis. This step entails assessing clinical markers such as heart rate, BMI, and genetic factors, as well as filtering the dataset to focus on characteristics that are most predictive of heart disease. This focused strategy results in a more streamlined and effective model capable of detecting minor patterns linked with cardiovascular risk.

Step 6: Data Splitting:

The critical procedure of data splitting is painstakingly managed in the Heart Disease Prediction project to balance training and testing datasets. This strategic divide is more than simply data distribution; it is a calibrated method to ensuring the model is efficiently trained and extensively evaluated. This enhances the model's capacity to learn from a wide range of data while also carefully evaluating its prediction accuracy and generalizability. This stage is critical in developing a strong prediction model that can predict heart disease risks in a variety of circumstances.

Step 7: Standardization:

The standardization of the dataset is critical in the Heart Disease Prediction project. Scaling features to a standard range or distribution is required for models that are sensitive to the size of input data, such as logistic regression and support vector machines. The team guarantees that each characteristic contributes equally to the model's predictions by standardizing, preventing bias towards variables with bigger magnitudes. This step not only improves model accuracy but also speeds up algorithm convergence during training, resulting in more efficient and effective model performance.

Step 8: Model Implementation and Evaluation:

The implementation and assessment of models in the Heart Disease Prediction project go beyond logistic regression and support vector classifiers. The project includes a number of machine learning algorithms, each of which has been carefully validated for prediction capacity. Models are evaluated not just on precision and accuracy, but also on recall, F1-score, and AUC-ROC. This complete review approach assures that the selected model not only properly predicts but also balances the trade-offs between sensitivity and specificity, which is critical in medical diagnosis scenarios. This broad approach to evaluation is critical for establishing an accurate and effective cardiac disease prediction tool.

Step 9: Optimization and Configuration:

Optimization and configuration are critical phases in developing the models in the Heart Disease Prediction project. The procedure entails precise tweaking of parameters in ensemble models like as learning rates, regularization approaches, and tree-specific factors. The goal of this fine-tuning is to improve the models' learning efficiency and forecast accuracy while avoiding overfitting. The research finds a balance between model complexity and performance by carefully modifying these parameters, ensuring that the final models are not only accurate but also resilient and generalizable to new, unknown data. This tuning step is crucial for guaranteeing the accuracy and reliability of the heart disease prediction models.

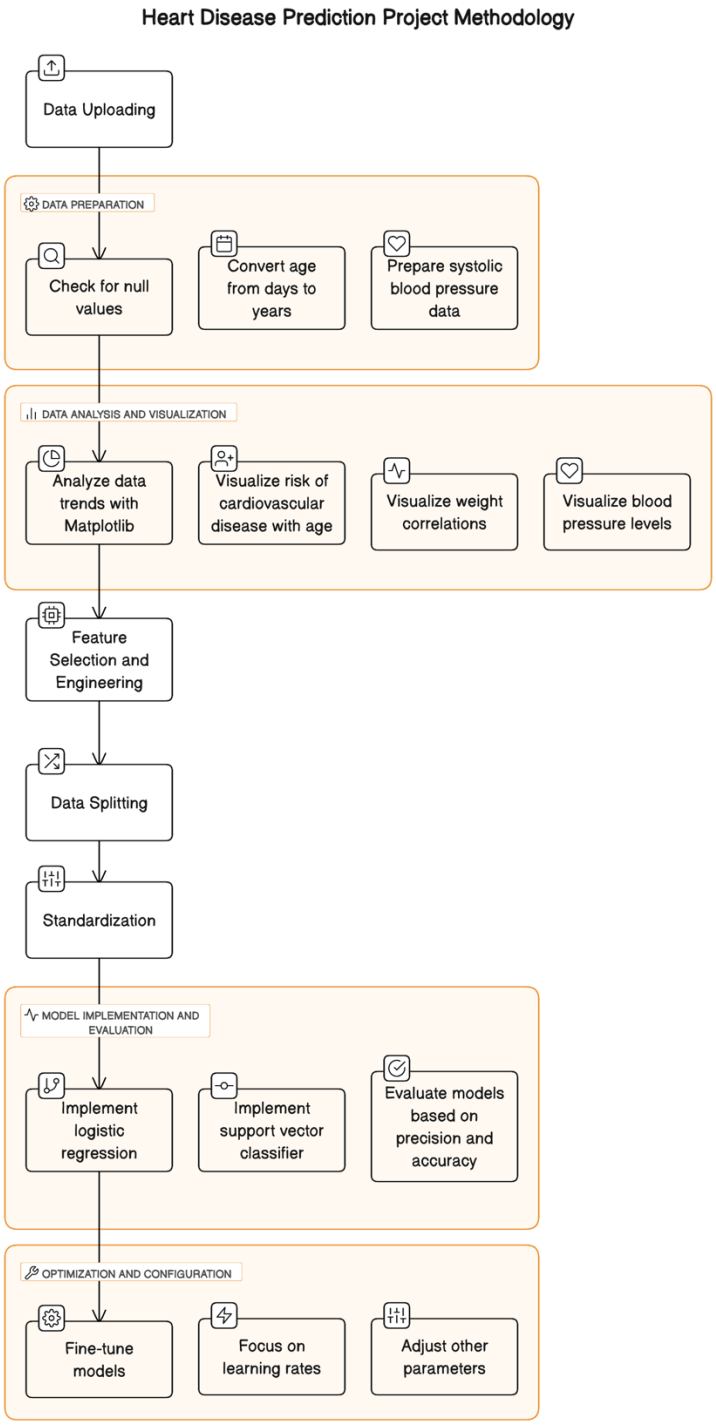

Heart Disease Prediction Project Methodology
Data Uploading
DATA PREPARATION
Check for null values
Convert age from days to years
Prepare systolic blood pressure data
DATA ANALYSIS AND VISUALIZATION
Analyze data trends with Matplotlib
Visualize risk of cardiovascular disease with age
Visualize weight correlations
Visualize blood pressure levels
Feature Selection and Engineering
Data Splitting
Standardization
MODEL IMPLEMENTATION AND EVALUATION
Implement logistic regression
Implement support vector classifier
Evaluate models based on precision and accuracy
OPTIMIZATION AND CONFIGURATION
Fine-tune models
Focus on learning rates
Adjust other parameters

Figure 13

5.2.1 Software’s used in the prediction of heart disease:

The main software’s which are used during designing this application are

1.Jupyter Notebook:

Jupyter Notebook is a free and open-source online software that lets you create and share documents with live code, equations, visualizations, and narrative prose. It is frequently used in data cleansing and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and many other applications. Because of its interactive, exploratory computing environment, Jupyter Notebooks are popular in data science, scientific computing, and academic research. They support a variety of computer languages, including Python, R, and Julia, and make it simple to share results with others.

2.Kaggle:

Kaggle is a popular online platform for data science and machine learning. It gives users access to a range of datasets, sponsors competitions in which participants use data analysis to solve real-world issues, and provides a collaborative platform for sharing and exploring data. Kaggle's community is made up of data scientists, statisticians, and machine learning enthusiasts from all walks of life. Users may search for datasets, submit datasets, investigate kernels (code scripts), and compete to solve data science tasks. Kaggle also provides a data science workbench in the cloud, making it an excellent resource for learning, testing, and applying data science and machine learning abilities.

5.2.2 Programming language used:

Python is an interpreted high-level programming language noted for its readability and adaptability. It's extensively utilized in a variety of industries, including web development, data research, AI, scientific computing, and automation. Python is compatible with a variety of programming paradigms, including procedural, object-oriented, and functional programming. Its wide ecosystem of third-party packages and extensive standard library make it a popular choice for developers and researchers. Python's syntax is intended to be plain and accessible, making it a good language for novices while meeting the demands of expert users.

5.2.3 Libraries Used:

1.Pandas:

Pandas is a popular open-source Python toolkit for data analysis and manipulation. It offers quick, versatile, and expressive data structures that enable dealing with structured (tabular, multidimensional, possibly heterogeneous) and time series data simple and intuitive. It's especially well-suited for working with and analyzing data in DataFrame formats, which are analogous to SQL tables or Excel spreadsheets. Pandas is a core Python data science tool that provides data cleaning, transformation, and aggregation functionality, making it a go-to tool for exploratory data analysis and data pretreatment.

2.Numpy:

NumPy is a key Python library for scientific computing. It has sophisticated array operations, mathematical functions, random number generation, linear algebra, and Fourier transform capabilities. Its fundamental data structure, NumPy arrays, is more efficient and powerful than ordinary Python lists, especially for big arrays. NumPy's performance makes it a must-have library for work in data science, statistics, engineering, and other fields. For a wide range of data analysis and visualization applications, it is frequently used in concert with other libraries like as Pandas and Matplotlib.

3. Matplotlib:

Matplotlib is a Python package that allows you to create static, animated, and interactive visualizations. It has a wide range of tools for creating various graphs and charts, such as line plots, scatter plots, bar charts, histograms, and more. Matplotlib's ability to generate publication-quality figures makes it extremely adaptable and extensively utilized in the data science field. It is particularly useful for visualizing data during exploratory data analysis, and its interoperability with other data manipulation tools such as NumPy and Pandas makes it an essential component of the Python data science toolkit.

4. Sklearn:

Scikit-learn, sometimes known as sklearn, is a popular open-source Python machine learning package. It is based on NumPy, SciPy, and Matplotlib and includes a variety of tools for data

mining and data analysis. Sklearn provides a wide range of methods for classification, regression, clustering, dimensionality reduction, model selection, and preprocessing. It is well-known for its simplicity and ease of use, making it suitable for both novice and expert practitioners of machine learning and data science.

5.Seaborn:

Seaborn is a Matplotlib-based Python visualization package that provides a high-level interface for creating visually appealing and useful statistical visuals. It's ideal for creating complicated charts from data in Pandas data structures. Seaborn makes it easier to create data visualizations by providing a range of plot options and appealing default styles. Its capacity to provide data in a visually appealing and interpretable style makes it popular in data science procedures.

Chapter 6

6.0 Results and Analysis

6.1 Results:

The Heart Disease Prediction project included the development and testing of multiple machine learning models. Logistic Regression, Gaussian NB, Bernoulli NB, Support Vector Classifier, Decision Tree Classifier, K Neighbors Classifier, Random Forest Classifier, Ada Boost Classifier, Bagging Classifier, Gradient Boosting Classifier, and XGBoost Classifier were among the models used. Each model was evaluated using criteria such as accuracy, precision, and confusion matrices. The results show that different models performed differently, with some getting greater accuracy and precision ratings than others. This thorough methodology allows for a detailed evaluation of the models' ability to predict cardiac disease.

6.2 Collected Data set of heart disease prediction:

The dataset is drawn from four datasets dating back to 1988: Cleveland, Hungary, Switzerland, and Long Beach V. It consists of 76 qualities, one of which is the target attribute, which signals the existence of cardiac disease in a patient. Integer values reflect the target property, with 0 indicating no disease and 1 indicating the existence of disease. Age, gender, type of chest pain, blood pressure, cholesterol levels, resting electrocardiographic results, maximum heart rate achieved, exercise-induced angina, ST depression induced by exercise, slope of the peak exercise ST segment, number of major vessels colored by fluoroscopy, and thalassemia status are all included in the dataset. To protect patients' privacy, the names and social security numbers in the dataset have been substituted with dummy values.

index	id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
0	0	18393	2	168	62	110	80	1	1	0	0	1	0
1	1	20228	1	156	85	140	90	3	1	0	0	1	1
2	2	18857	1	165	64	130	70	3	1	0	0	0	1
3	3	17623	2	169	82	150	100	1	1	0	0	1	1
4	4	17474	1	156	56	100	60	1	1	0	0	0	0
5	8	21914	1	151	67	120	80	2	2	0	0	0	0
6	9	22113	1	157	93	130	80	3	1	0	0	1	0
7	12	22584	2	178	95	130	90	3	3	0	0	1	1
8	13	17668	1	158	71	110	70	1	1	0	0	1	0

Figure 14 Dataset

6.3 Data Preprocessing:

Data Upload: Importing the comprehensive heart disease dataset.

Index and ID Removal: Dropping 'index' and 'id' columns as they are not relevant for analysis.

Null Value Check: Ensuring data integrity by checking for and addressing null values.

Age Conversion: Transforming the age from days to years.

Data Description: Analyzing data frame specifics like mean, standard deviation, and value ranges.

Systolic Blood Pressure Adjustment: Correcting implausible 'ap_hi' values, as values over 200 or under 60 could indicate medical anomalies.

Gender Determination: Assuming '1' represents male and '2' female based on average weight differences.

Visual Analysis with Matplotlib: Using visual tools to observe correlations between age, weight, blood pressure, and cardiovascular disease presence.

Feature Correlation Heatmap: Creating a heatmap to understand feature correlations.

Dropping Highly Correlated Features: Removing features with high correlation to avoid redundancy.

Data Skewness Analysis: Checking data distribution and skewness.

Dataset Splitting: Dividing data into training (70%) and testing (30%) sets, with a constant random state for consistent splitting.

Standardization: Standardizing the dataset to ensure uniformity across features.

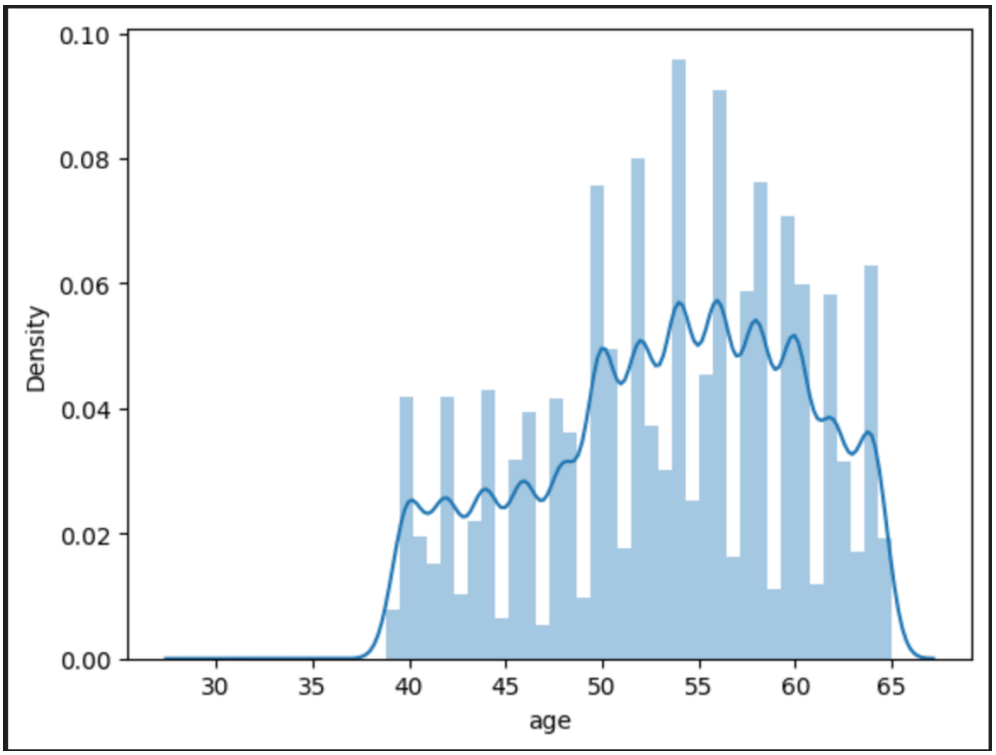


Figure 15 Distplot

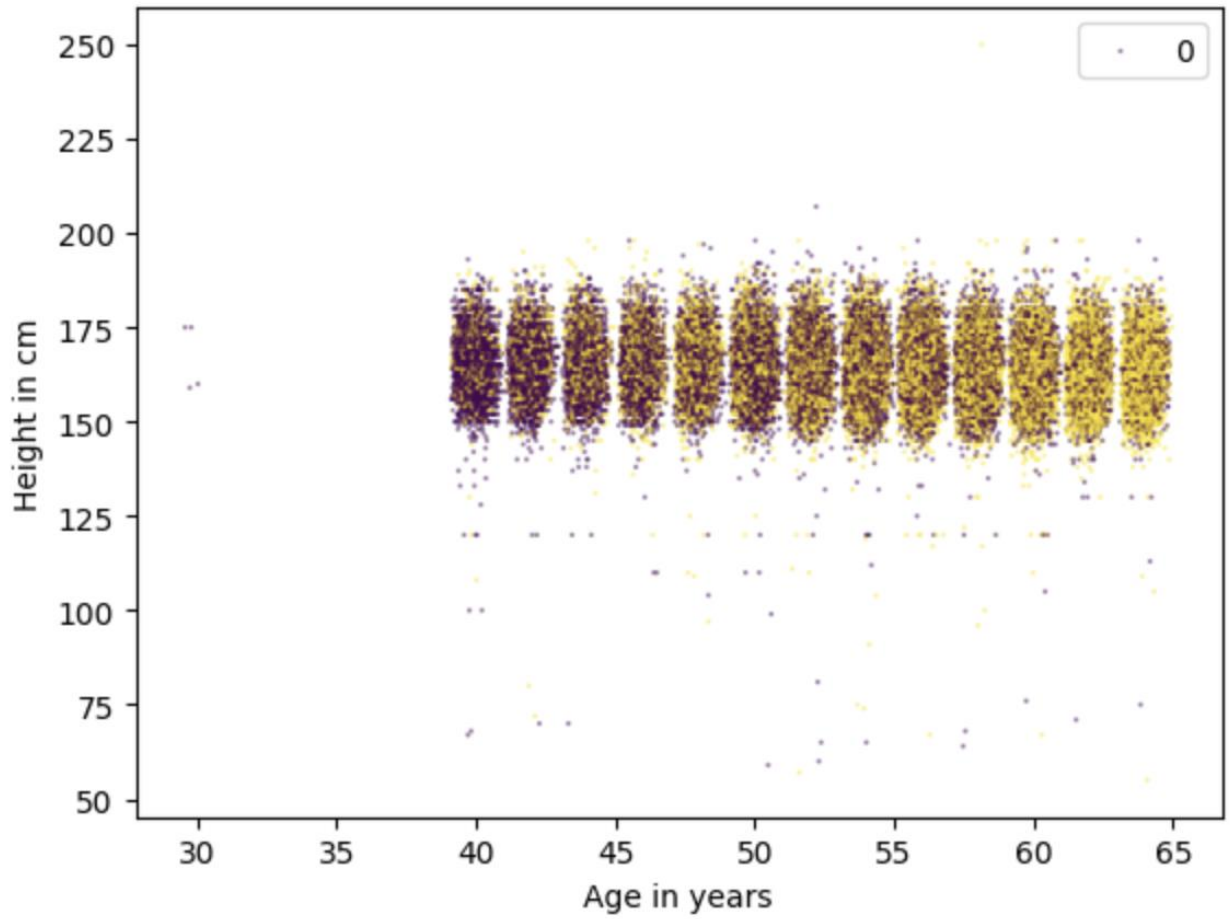


Figure 16 Scatterplot Heigh/Age

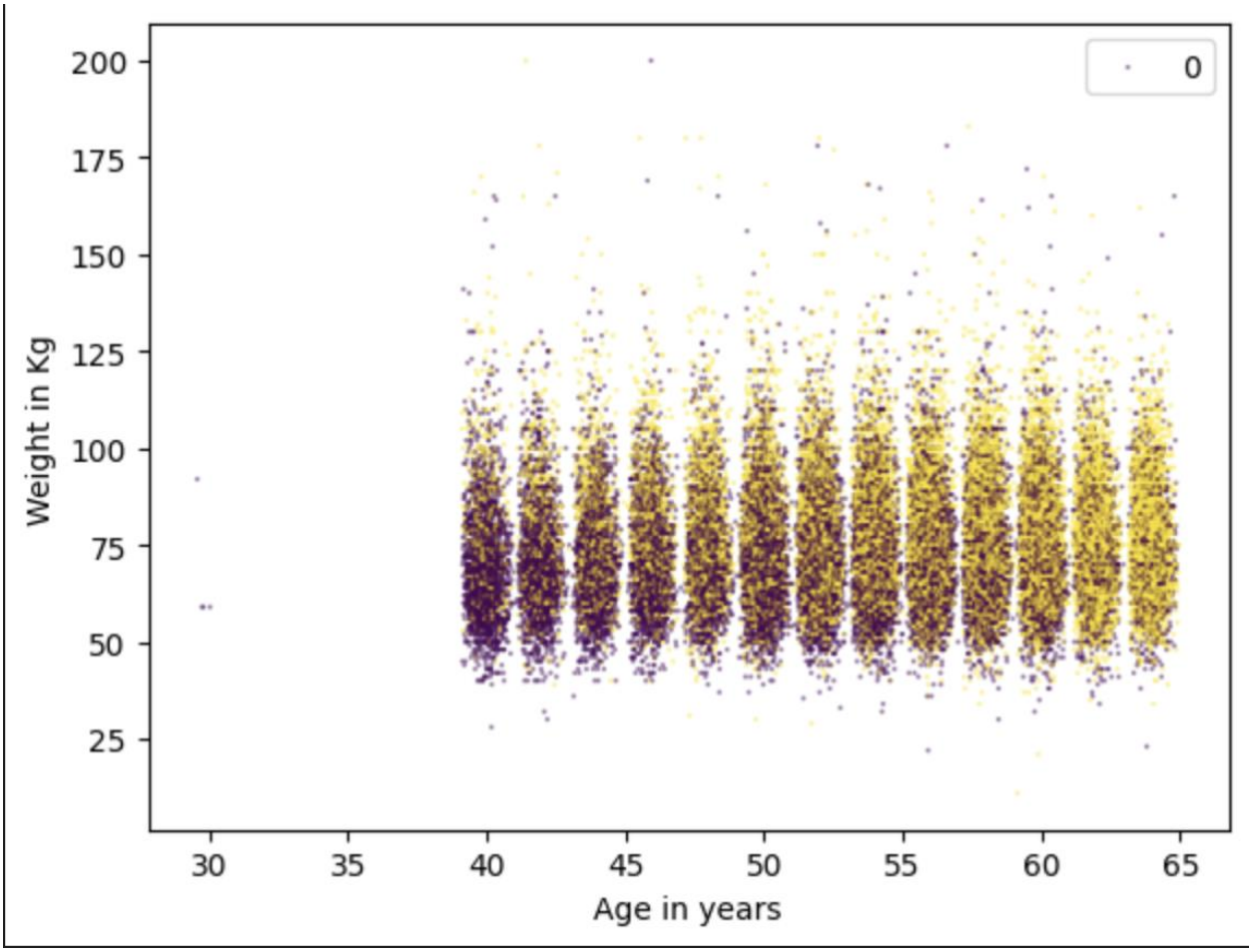


Figure 17 Scatterplot Weight/Age

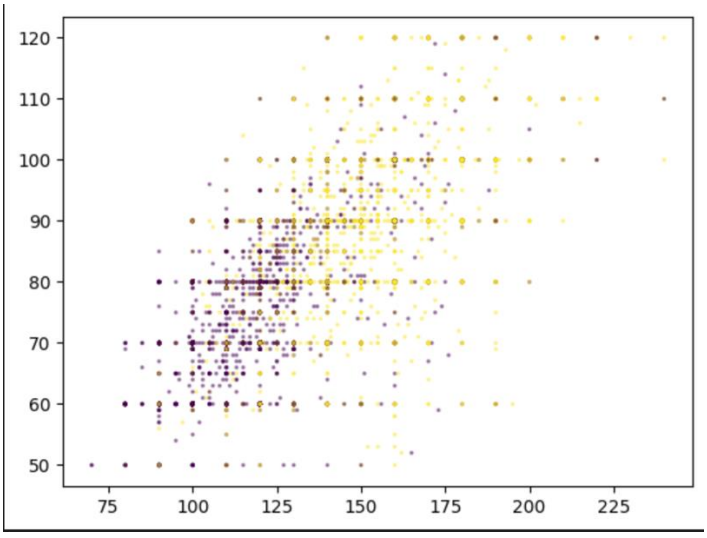


Figure 18 Correlation ap_hi/ap_low

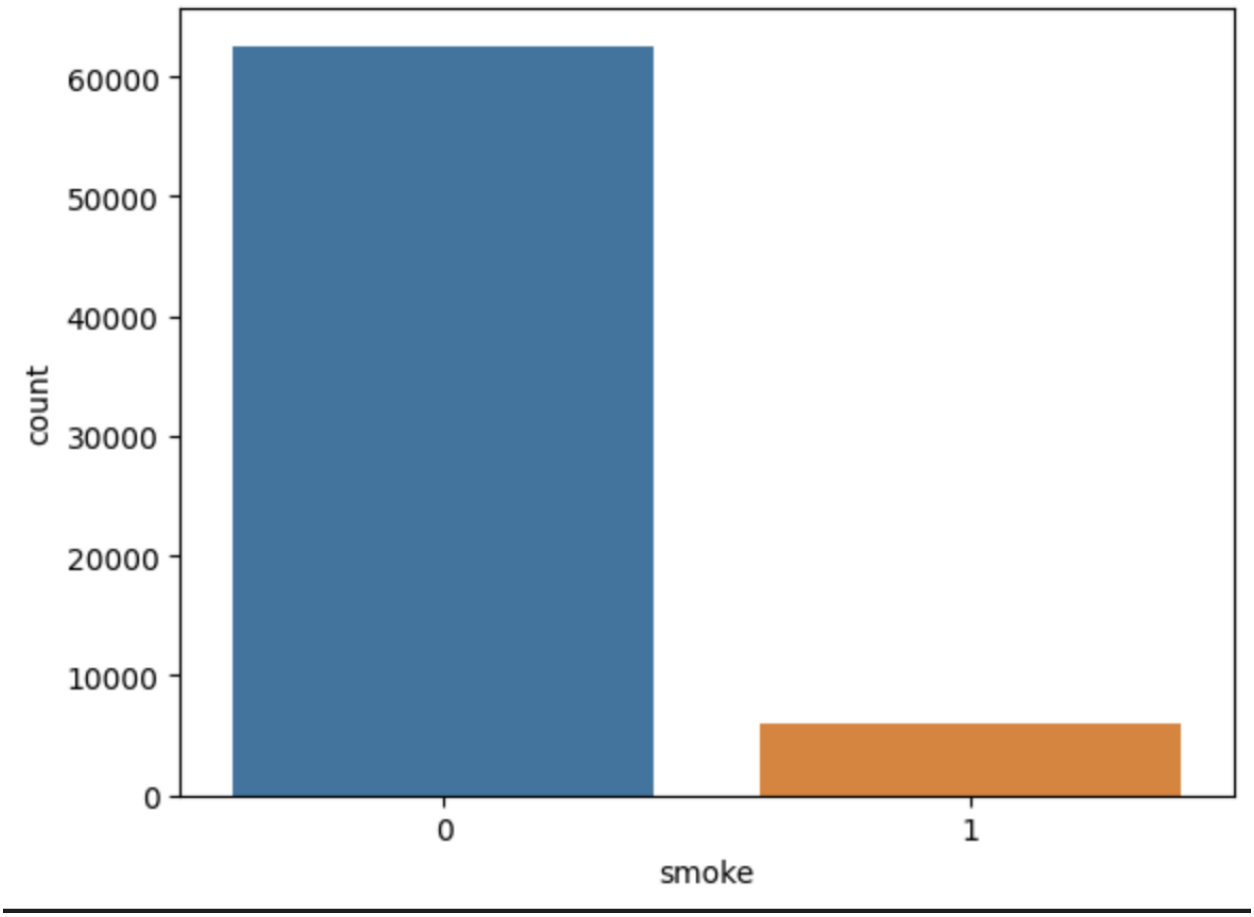


Figure 19 Graph count/smoke

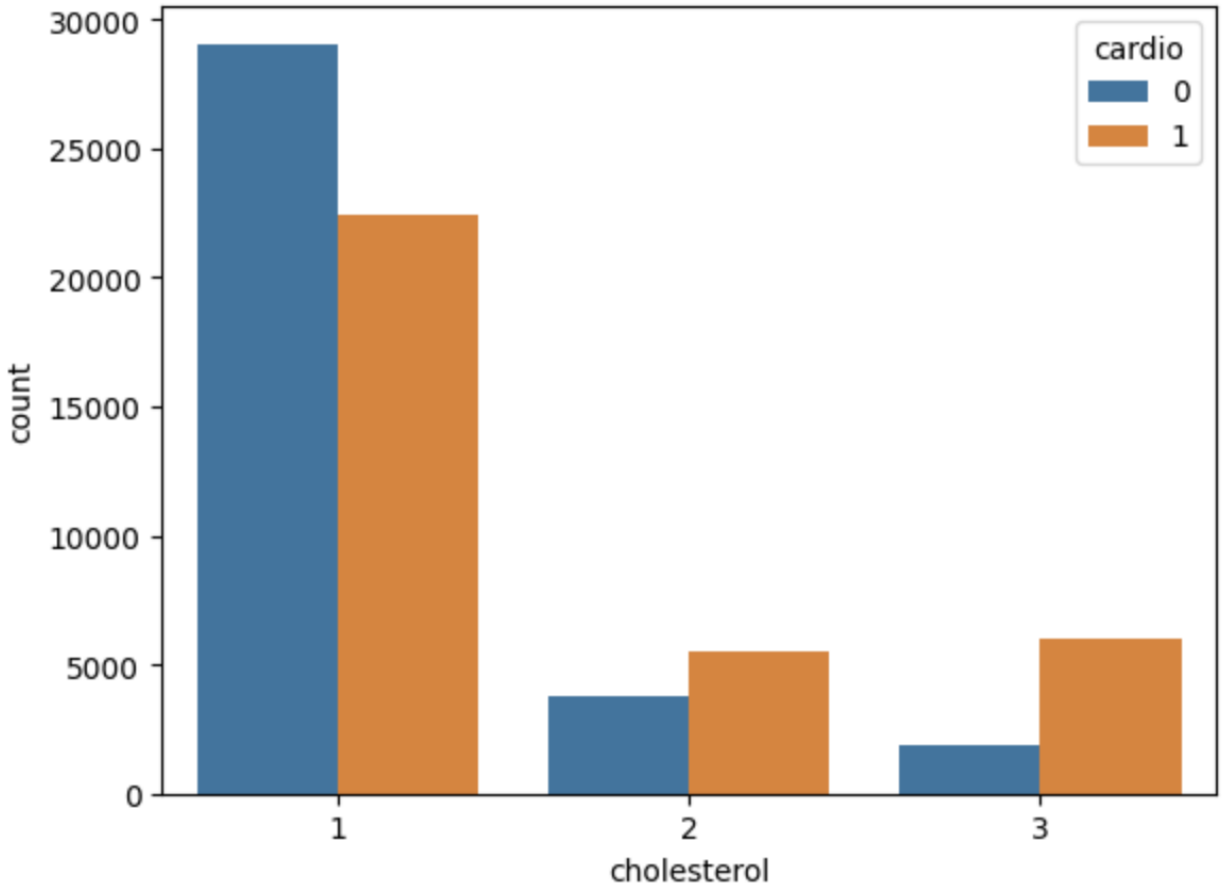


Figure 20 Graph count/cholestrol

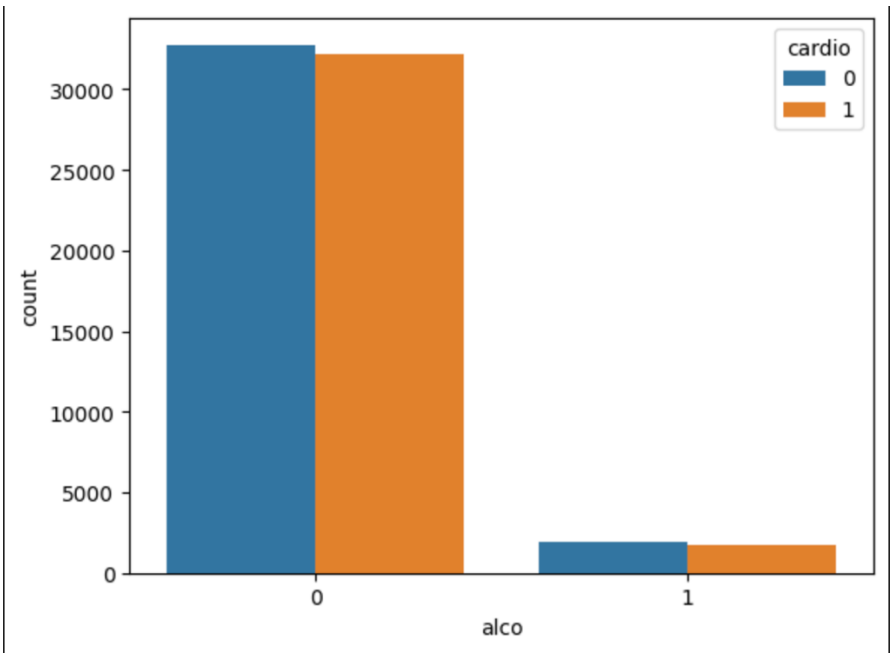


Figure 21 Graph Count/alcohol

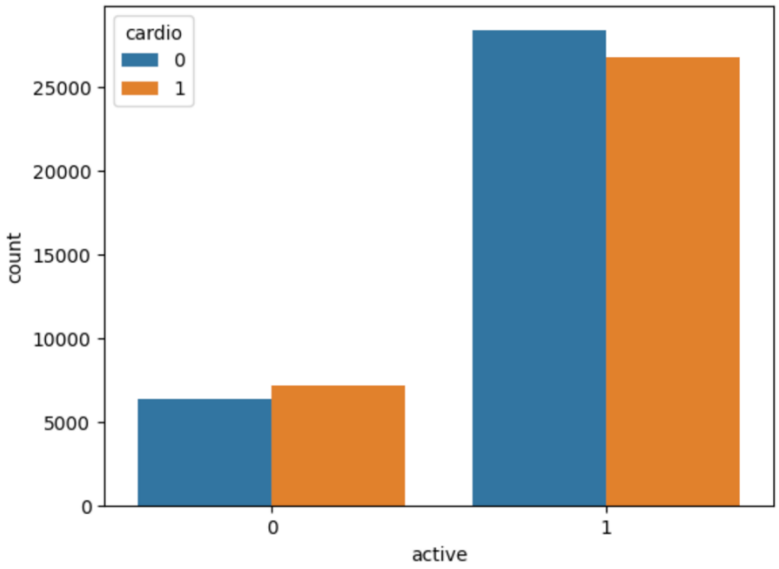


Figure 22 Graph count/active

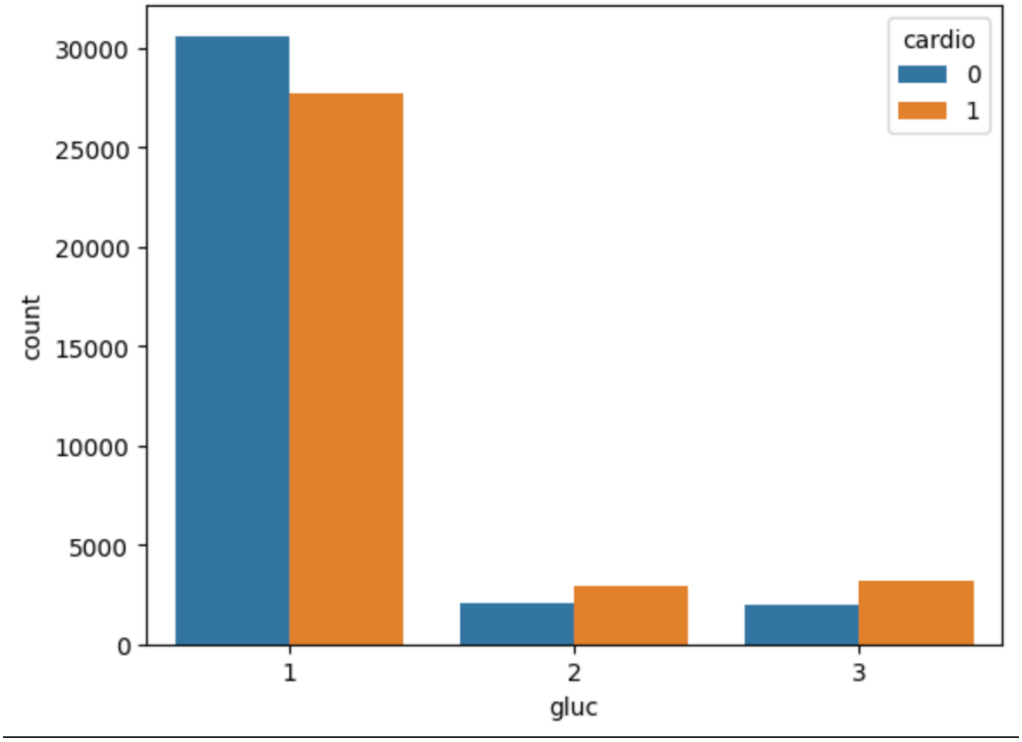
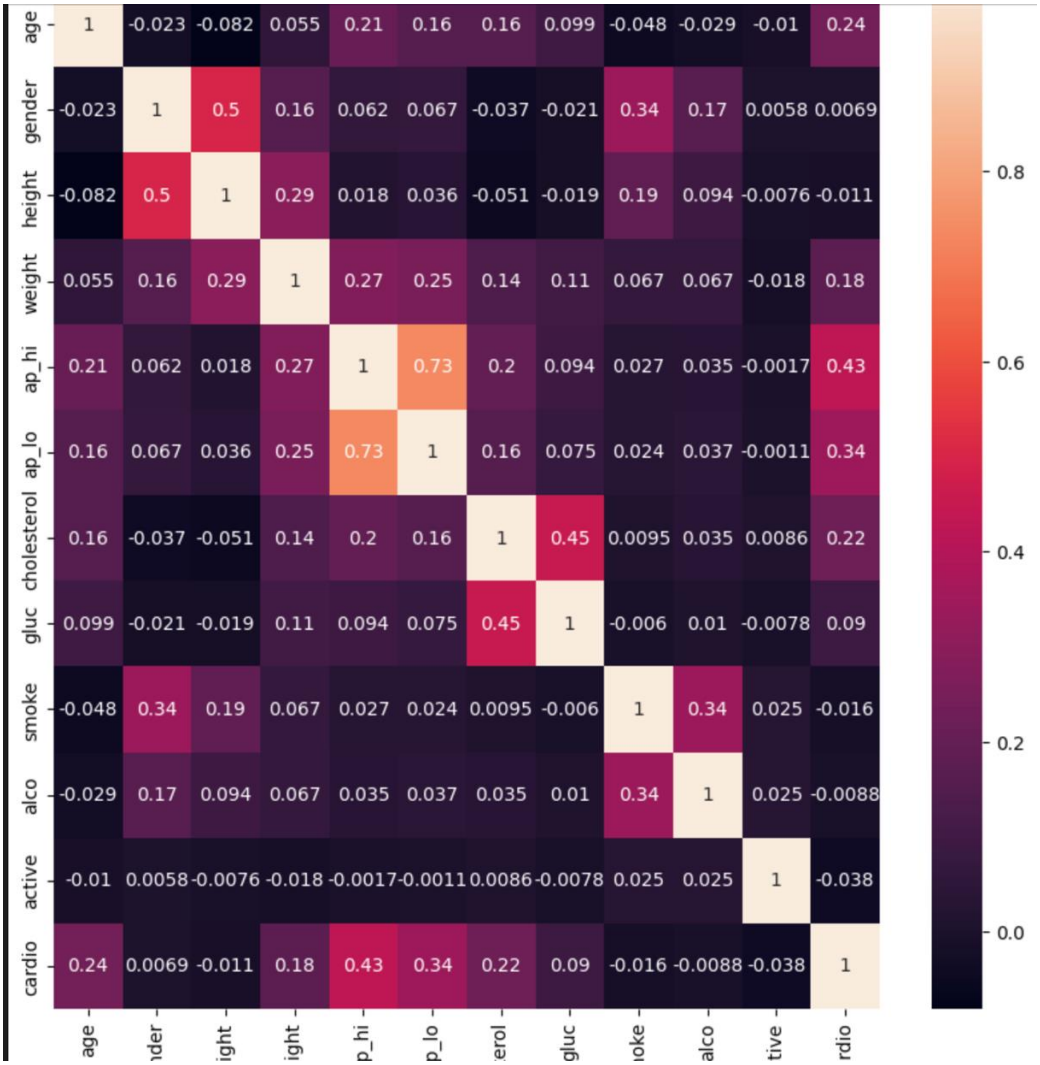


Figure 23 Graph count/glucose

Figure 24 Heatmap


age 1 -0.023 -0.082 0.055 0.21 0.16 0.16 0.099 -0.048 -0.029 -0.01 0.24
gender -0.023 1 0.5 0.16 0.062 0.067 -0.037 -0.021 0.34 0.17 0.0058 0.0069
height -0.082 0.5 1 0.29 0.018 0.036 -0.051 -0.019 0.19 0.094 -0.0076 -0.011
weight 0.055 0.16 0.29 1 0.27 0.25 0.14 0.11 0.067 0.067 -0.018 0.18
ap_hi 0.21 0.062 0.018 0.27 1 0.73 0.2 0.094 0.027 0.035 -0.0017 0.43
ap_lo 0.16 0.067 0.036 0.25 0.73 1 0.16 0.075 0.024 0.037 -0.0011 0.34
cholesterol 0.16 -0.037 -0.051 0.14 0.2 0.16 1 0.45 0.0095 0.035 0.0086 0.22
gluc 0.099 -0.021 -0.019 0.11 0.094 0.075 0.45 1 -0.006 0.01 -0.0078 0.09
smoke -0.048 0.34 0.19 0.067 0.027 0.024 0.0095 -0.006 1 0.34 0.025 -0.016
alco -0.029 0.17 0.094 0.067 0.035 0.037 0.035 0.01 0.34 1 0.025 -0.0088
active -0.01 0.0058 -0.0076 -0.018 -0.0017 -0.0011 0.0086 -0.0078 0.025 0.025 1 -0.038
cardio 0.24 0.0069 -0.011 0.18 0.43 0.34 0.22 0.09 -0.016 -0.0088 -0.038 1
age
ight
ight
p_hi
p_lo
erol
gluc
oke
alco
tive
rdio
0.8
0.6
0.4
0.2
0.0

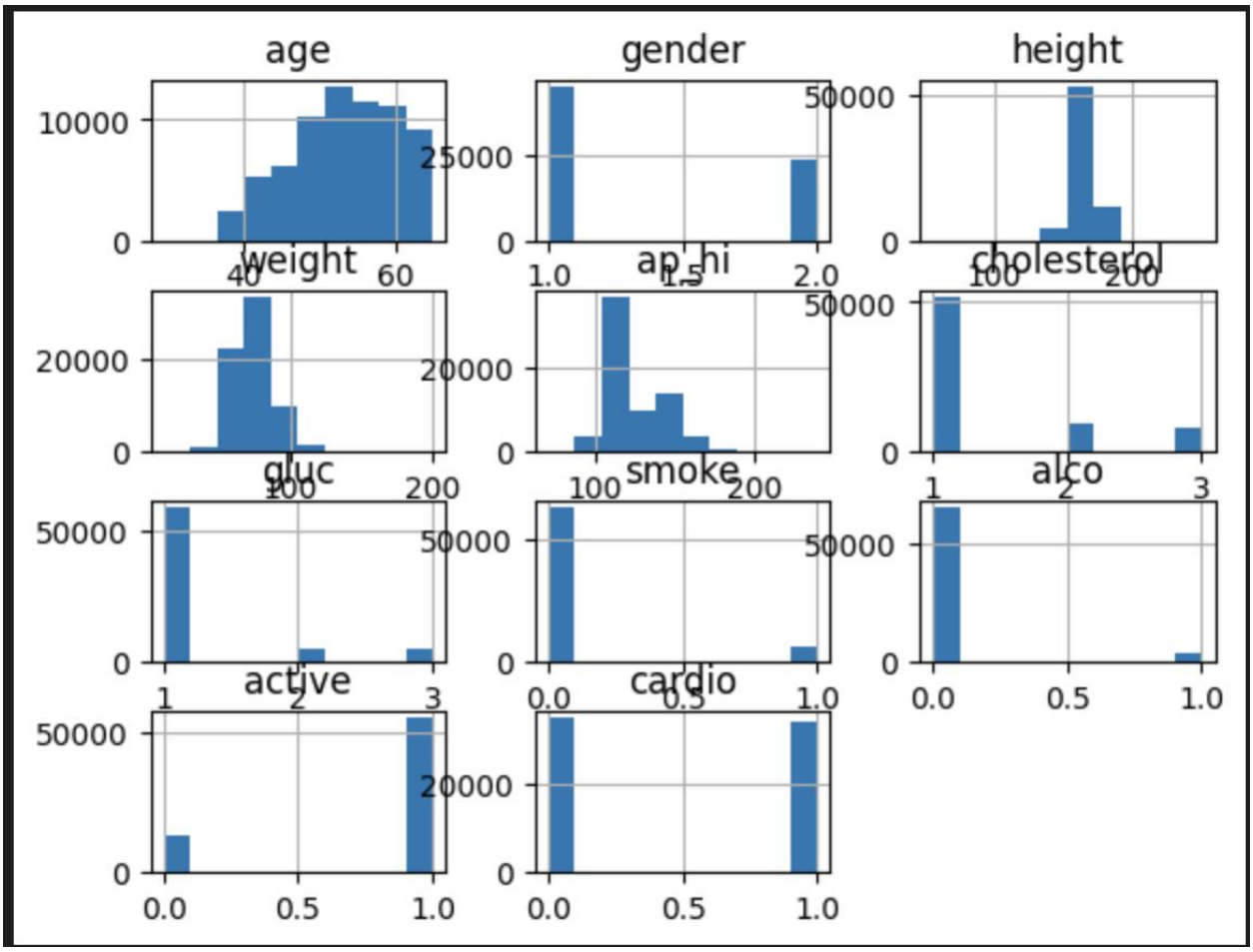


Figure 25 Histograph

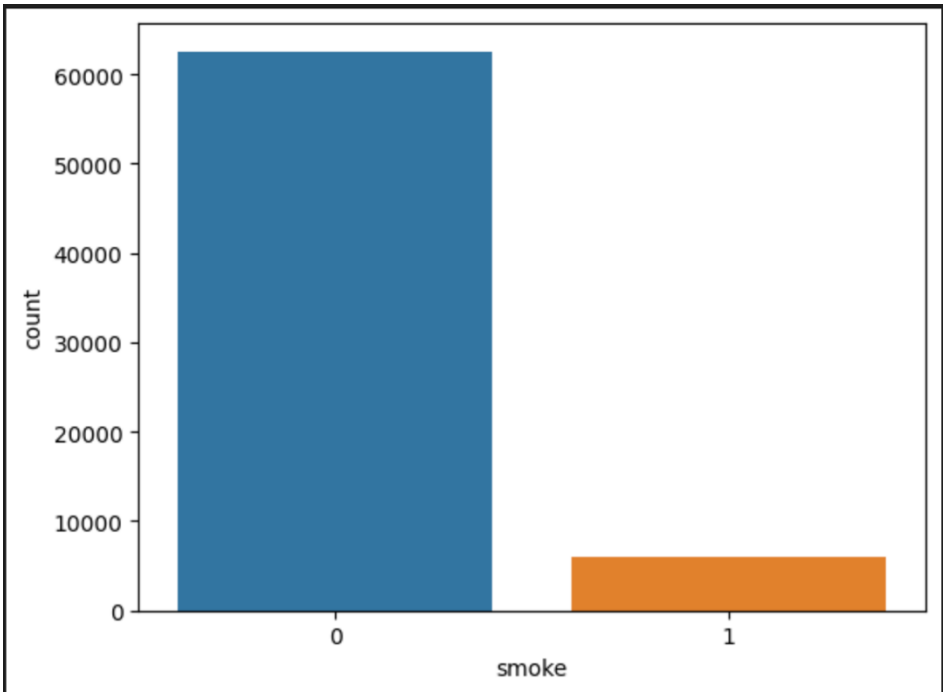


Figure 26 Graph count/smoke

6.4 Model Building and Evaluation:

Model Selection: Choosing appropriate machine learning models for heart disease prediction, such as logistic regression, decision trees, random forests, or neural networks.

Model Training: Using the training dataset to train these models. This involves feeding the data into the models and allowing them to learn the patterns associated with heart disease.

Hyperparameter Tuning: Adjusting the models' hyperparameters to optimize their performance.

Model Evaluation: Assessing the performance of the models using the testing dataset. This typically involves calculating metrics like accuracy, precision, recall, and F1 score.

Model Comparison: Comparing the performance of different models to select the best one for heart disease prediction.

Final Model Selection: Based on the evaluation, Gradient Boosting Classifier was the most effective model.

Hyperparameter Testing:The detailed analysis of the hyperparameter tuning on the Gradient Boosting Classifier in Heart Disease Prediction project includes evaluating various parameters like learning rate, number of estimators (N_estimators), maximum depth of trees (Max_Depth),

minimum samples for splitting a node (Min_samples), and maximum features (Max_features). The analysis revealed that:

A learning rate of 0.1 is optimal before the model begins to overfit.

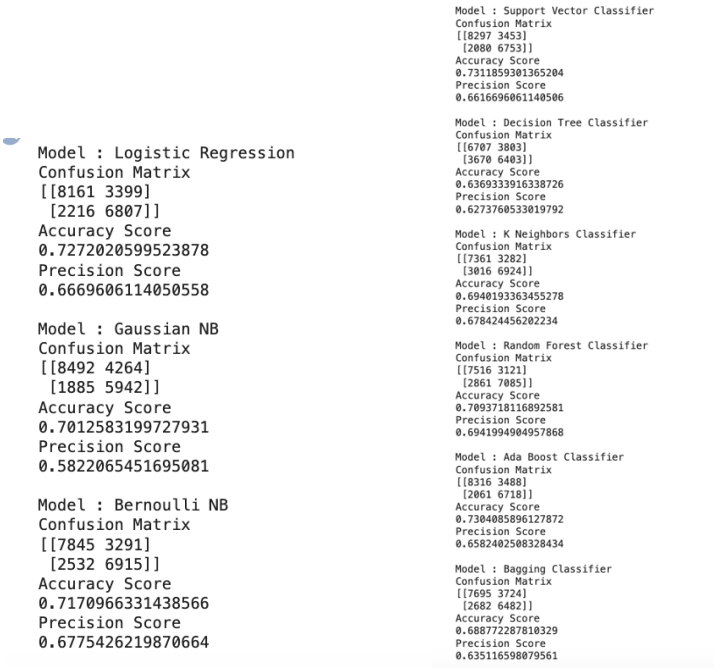
The model tends to overfit with N_estimators greater than 100.

A Max_Depth of 3 is ideal before overfitting occurs.

Min_samples at 100 yields optimal results.

Max_features set at 4 offers the best performance.

These insights guided the configuration of the Gradient Boosting Classifier to achieve the best results on unseen data. The project focuses on precision as a key metric due to the importance of correctly classifying the presence or absence of heart disease. The conclusion draws on these findings, suggesting specific settings for the Gradient Boosting Classifier to optimize performance for new, unseen data.



```
Model : Gradient Boosting Classifier
Confusion Matrix
[[7993 3090]
 [2384 7116]]
Accuracy Score
0.7340523733177865
Precision Score
0.6972369194591417

Model : XGBoost Classifier
Confusion Matrix
[[7958 3163]
 [2419 7043]]
Accuracy Score
0.7288053247825875
Precision Score
0.6900842641583382
```

Figure 27 Algorithm Scores

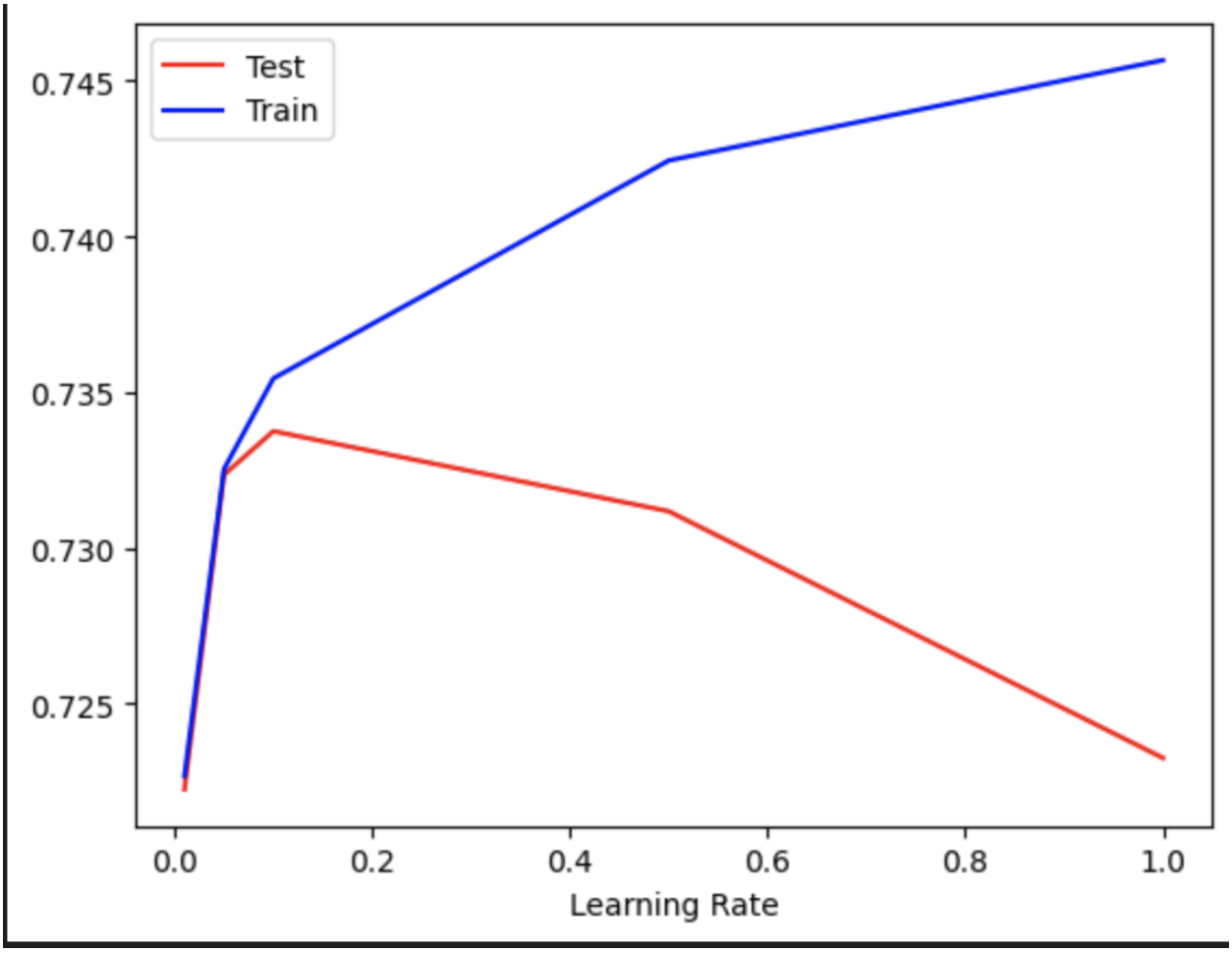


Figure 28 Hyperparameter tuning on Gaussian

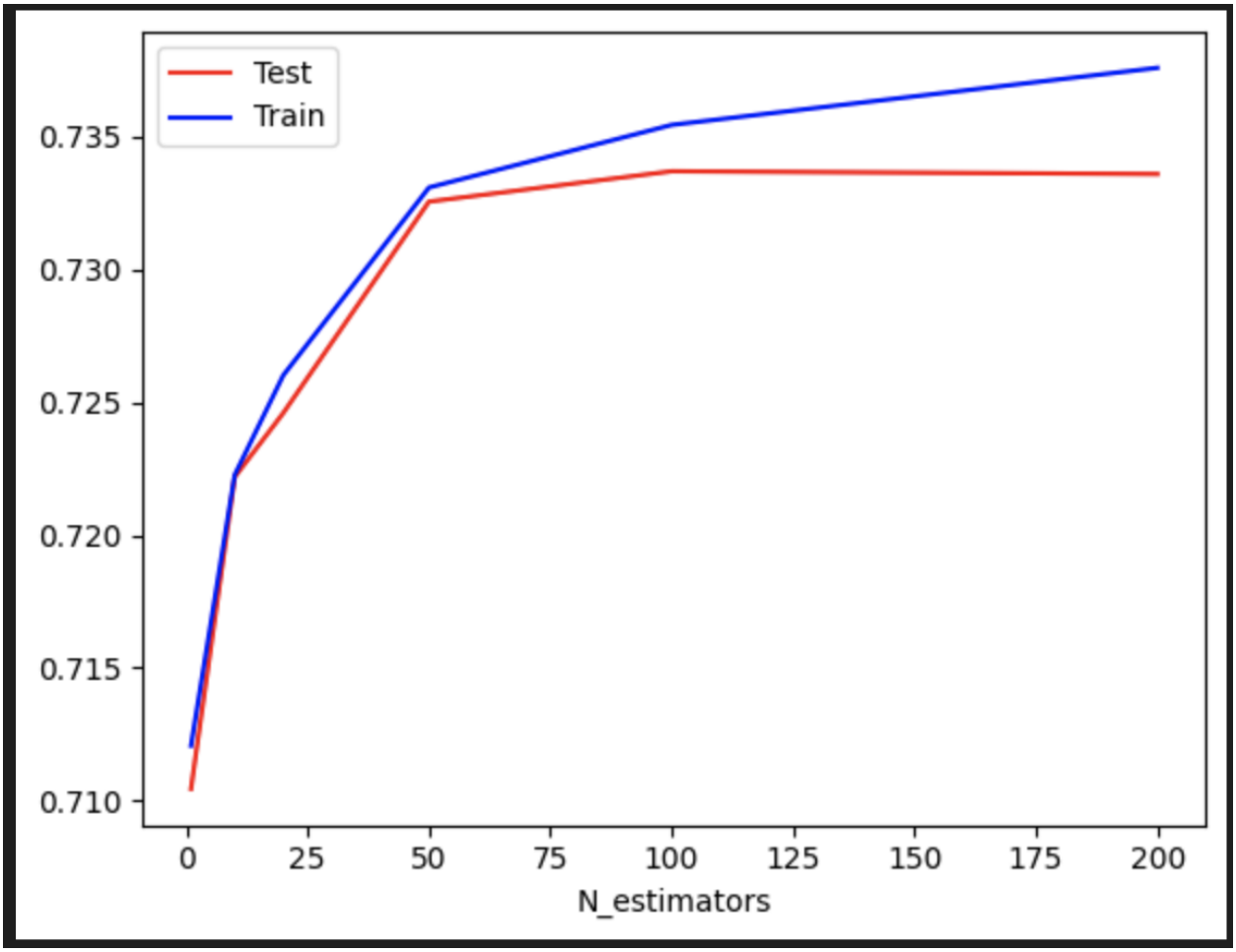


Figure 29 Hyperparameter tuning N_estimators

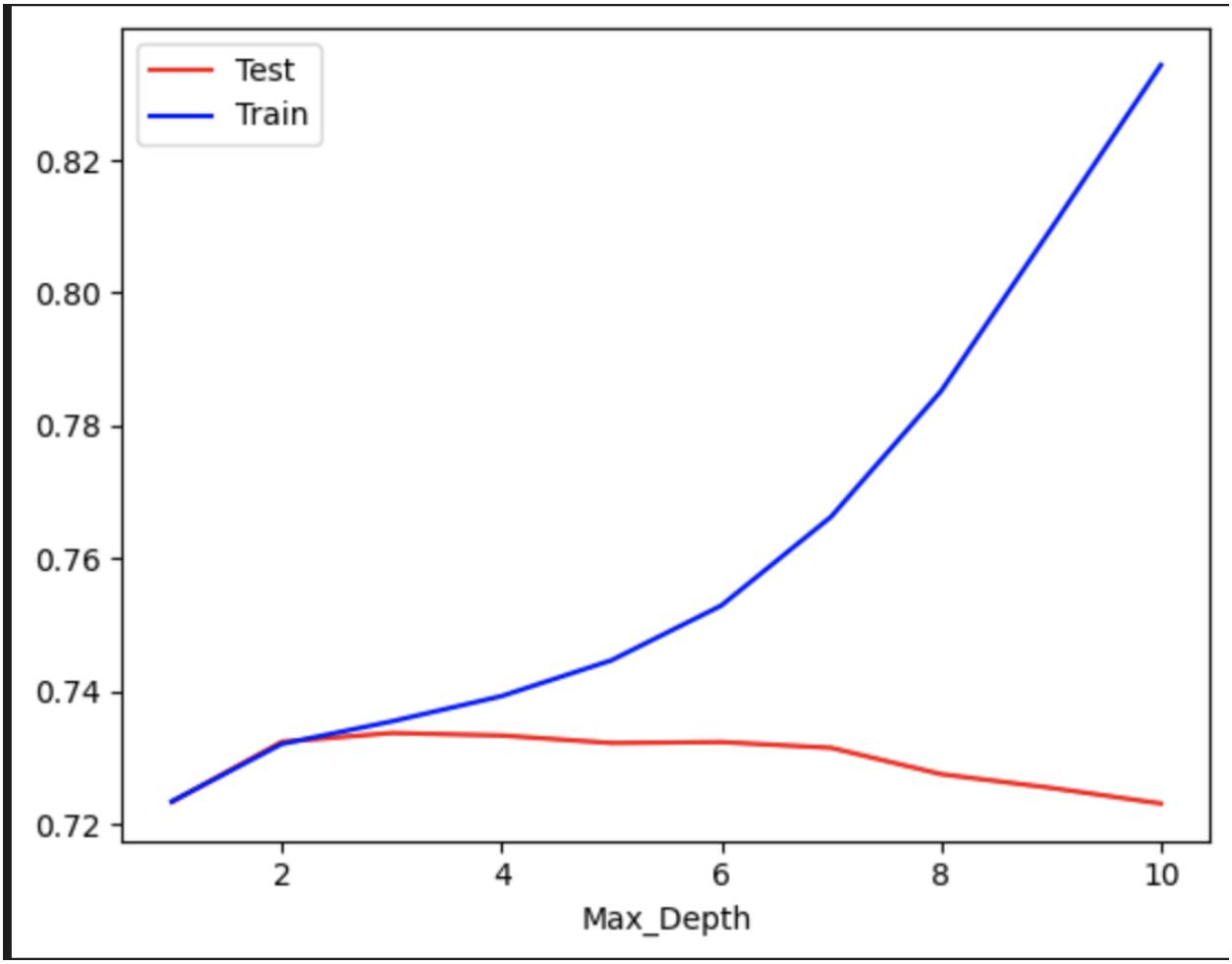


Figure 30 Hyperparameter tuning Max_Depth

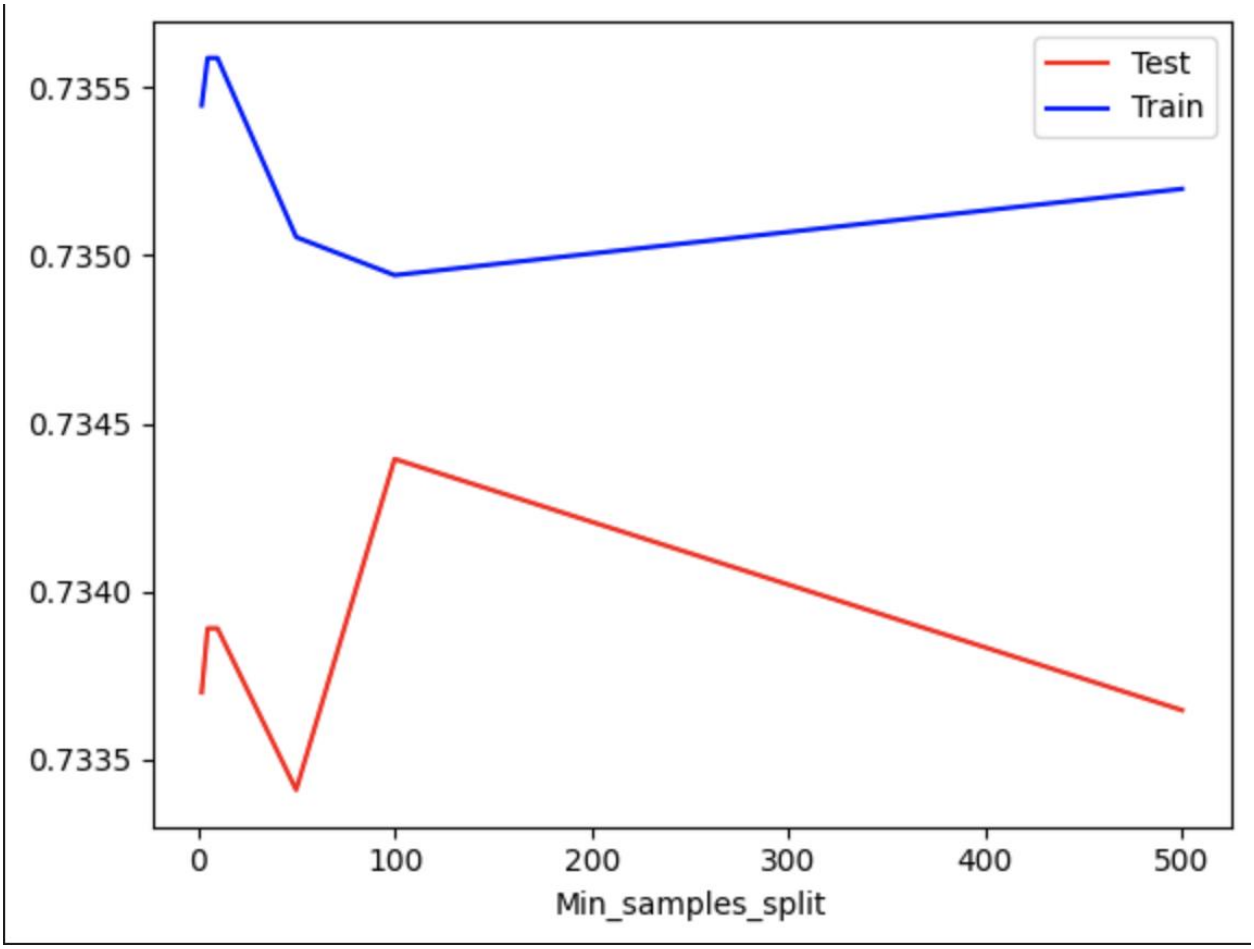


Figure 31 Hyperparameter tuning Min_samples_split

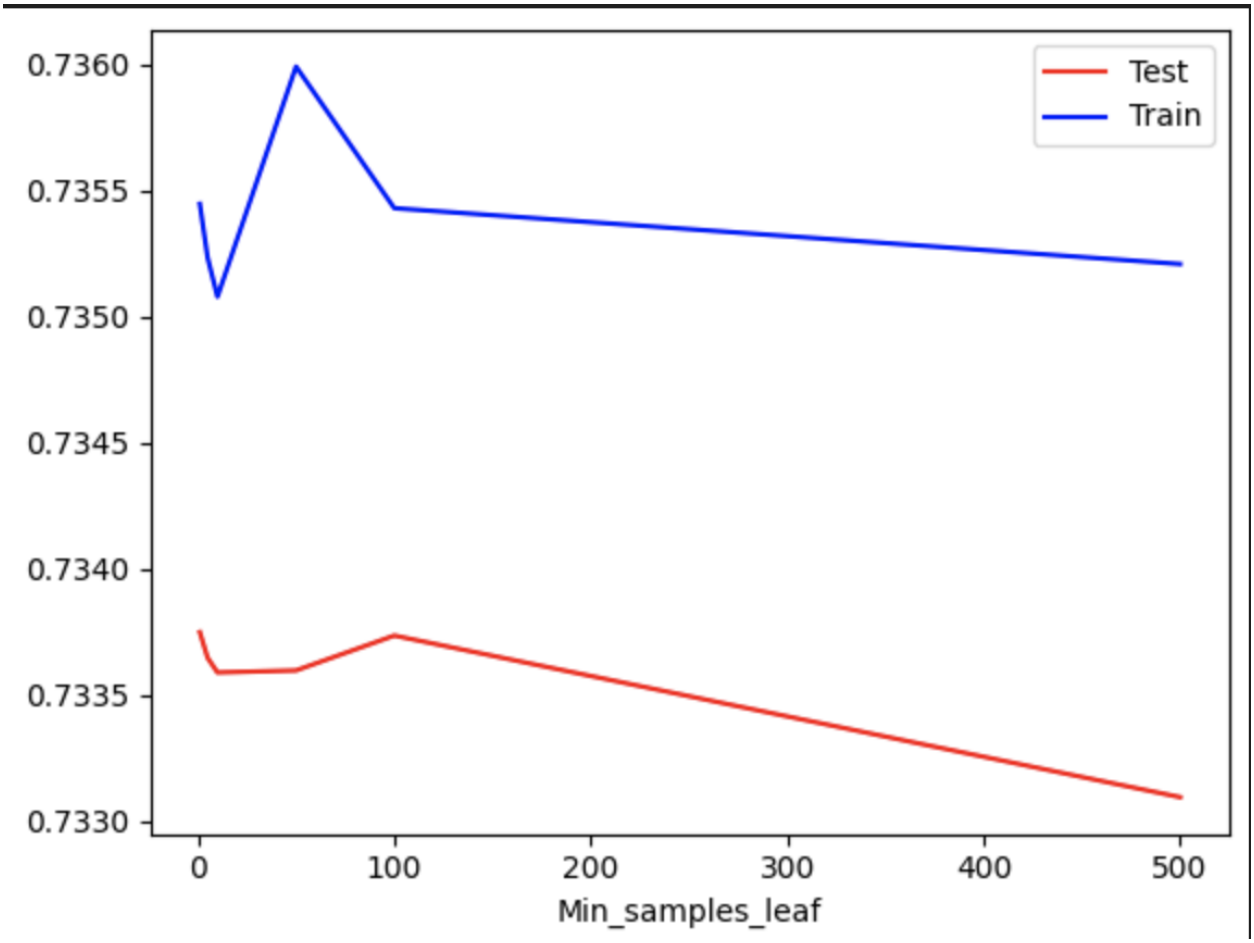


Figure 32 Hyperparameter tuning Min_samples_leaf

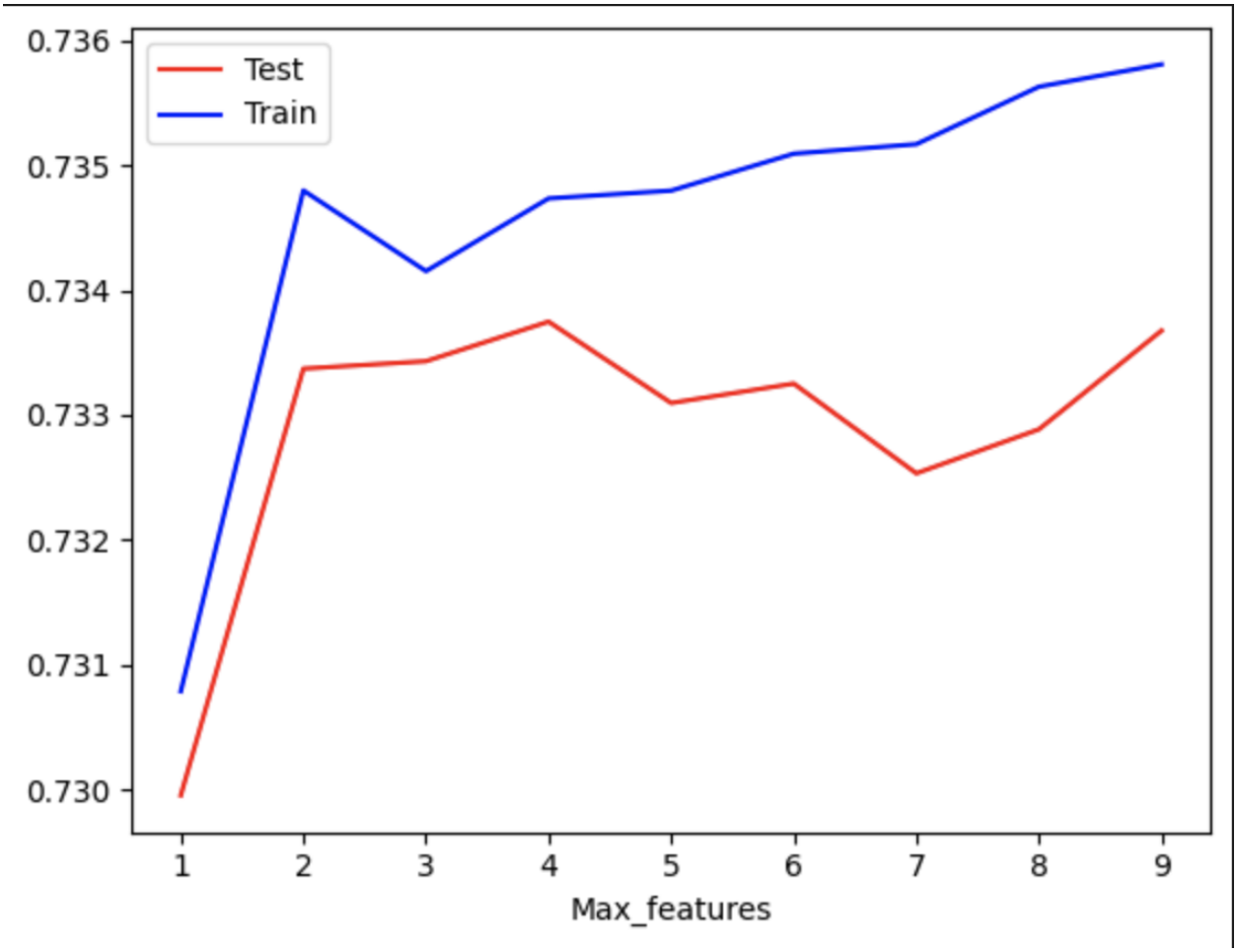


Figure 33 Hyperparameter tuning max_features

Chapter 7

7.0 Conclusion:

We emphasize on the machine learning's disruptive significance in the field of healthcare diagnostics. It focuses effectiveness in utilizing advanced computational approaches for the early identification and control of heart disease, a crucial step toward customized and predictive healthcare. The findings show that such technologies have the potential to alter the approach to cardiac disease by supporting more proactive, data-driven, and patient-centric healthcare solutions. This research demonstrates the expanding significance of machine learning in improving diagnostic accuracy and patient outcomes in the healthcare business. The conclusion of the Heart Disease Prediction project encapsulates the effectiveness of various machine learning models in predicting heart disease. The project successfully integrated a comprehensive dataset from multiple sources, ensuring a rich and diverse data pool for analysis. In the data preprocessing phase, rigorous steps were taken to ensure data quality, including null value checks, age adjustments, and standardization. The project's core involved the meticulous evaluation of different models, with Gradient Boosting Classifier emerging as the most effective. The detailed hyperparameter tuning of this model underscored the importance of fine-tuning for optimal model performance. This project demonstrates the critical role of machine learning in predicting heart disease, emphasizing the potential of these technologies in advancing healthcare diagnostics and personalized medicine.

7.1 Issues related to social, ethical, legal factors:

Addressing ethical considerations and data privacy is paramount in the Heart Disease Prediction app, extending beyond initial compliance. Continuous monitoring and updating of privacy practices are essential to align with evolving data protection regulations. Transparency with users regarding data usage and safeguarding mechanisms builds trust. Additionally, incorporating ethical AI principles, like fairness and non-discrimination, ensures that the app's predictive models do not perpetuate biases. It's also crucial to involve diverse stakeholders, including legal experts, ethicists, and patient advocacy groups, in discussions about data usage and privacy policies. This approach ensures that the app not only meets legal requirements but also aligns with broader societal values and ethical standards. By prioritizing user privacy and ethical considerations, the project can foster

greater user acceptance and confidence, which is key to its success and impact in the healthcare sector.

Chapter 8

8.0 Suggestions for future works:

Suggestions for future work in the Heart Disease Prediction project can include:

1.Expanding Dataset Sources: Incorporating more diverse datasets from various geographical locations to enhance the model's generalizability.

2.Advanced Feature Engineering: Exploring more sophisticated feature engineering techniques to uncover deeper insights from the data.

3.Incorporating Real-Time Data: Utilizing real-time health data from wearable devices to improve predictive accuracy.

4.Deep Learning Approaches: Experimenting with deep learning models, which might be more effective in capturing complex patterns in data.

5.Interdisciplinary Collaboration: Working closely with healthcare professionals to gain more insights into clinical aspects of heart disease.

6.User Interface Design for the App: Developing a more intuitive and user-friendly interface for the application to increase its usability.

7.Longitudinal Studies: Conducting longitudinal studies to understand the impact of various factors over time.

8. Ethical and Privacy Aspects: Continuing to focus on the ethical implications and privacy concerns related to the use of personal health data.

9.Clinical Trials: Validating the model's effectiveness and safety through clinical trials.

10.Customization for Specific Populations**: Tailoring the model to address specific needs of different population groups, considering factors like age, gender, and genetic predispositions.

References:

[1] Soni J, Ansari U, Sharma D & Soni S (2011). Predictive data mining for medical diagnosis: an overview of heart disease prediction. International Journal of Computer Applications, 17(8), 43-8

[2] Dangare C S & Apte S S (2012). Improved study of heart disease prediction system using data mining classification techniques. International Journal of Computer Applications, 47(10), 44-8.

[3] Ordonez C (2006). Association rule discovery with the train and test approach for heart disease prediction. IEEE Transactions on Information Technology in Biomedicine, 10(2), 334-43.

[4] Shinde R, Arjun S, Patil P & Waghmare J (2015). An intelligent heart disease prediction system using k-means clustering and Naïve Bayes algorithm. International Journal of Computer Science and Information Technologies, 6(1), 637-9.

[5] Bashir S, Qamar U & Javed M Y (2014, November). An ensemble-based decision support framework for intelligent heart disease diagnosis. In International Conference on Information Society (i-Society 2014) (pp. 259-64). IEEE.
ICCRDA 2020
IOP Conf. Series: Materials Science and Engineering 1022 (2021) 012072
IOP Publishing
doi:10.1088/1757-899X/1022/1/012072
9

[6] Jee S H, Jang Y, Oh D J, Oh B H, Lee S H, Park S W & Yun Y D (2014). A coronary heart disease prediction model: the Korean Heart Study. BMJ open, 4(5), e005025.

[7] Ganna A, Magnusson P K, Pedersen N L, de Faire U, Reilly M, Ärnlöv J & Ingelsson E (2013). Multilocus genetic risk scores for coronary heart disease prediction.

Arteriosclerosis, thrombosis, and vascular biology, 33(9), 2267-72.

[8] Jabbar M A, Deekshatulu B L & Chandra P (2013, March). Heart disease prediction using lazy associative classification. In 2013 International Mutli-Conference on Automation, Computing,Communication, Control and Compressed Sensing (iMac4s) (pp. 40- 6). IEEE.

[9] Dangare Chaitrali S and Sulabha S Apte. "Improved study of heart disease prediction system using data mining classification techniques." International Journal of Computer Applications 47.10 (2012): 44-8.

[10] Soni Jyoti. "Predictive data mining for medical diagnosis: An overview of heart disease prediction." International Journal of Computer Applications 17.8 (2011): 43-8.

[11] Chen A H, Huang S Y, Hong P S, Cheng C H & Lin E J (2011, September). HDPS: Heart disease prediction system. In 2011 Computing in Cardiology (pp. 557-60). IEEE

Appendix:

In [ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [ ]:
```
# Importing Dataset

df = pd.read_csv('heart_data 2.csv')
df.head()
```

In [ ]:
```
# No need to keep index and id as they would not affect our analysis

df.drop(['index','id'],axis=1,inplace=True)
df.head()
```

In [ ]:
```
# Checking for nulls

df.isna().sum()
```

In [ ]:
```
# Varying age leads us to consider these duplicated values

df[df.duplicated()]
```

In [ ]:
```
# Age is in days
np.round(df['age']/365,2)
```

In [ ]:
```
# Converting to years
df['age']=np.round(df['age']/365,2)
df['age']
```

In [ ]:
```
df.describe()
```

In [ ]:
```
sns.distplot(df['ap_hi'])
```

In [ ]:
```
# We can see that ap_hi which symbolises systolic blood pressure (Blood Pressure when your heart is beating) has
# maximum of 16020, it is medically impossible for it to go over 200 or be under 60 as patient will start having sym
# failure
df[df['ap_hi']>250].head()
```

In [ ]:
```
df[df['ap_hi']<60]
```

In [ ]:
```
# A safer cutoff for ap_hi is the range 60-250
df = df[(df['ap_hi']<=250) & (df['ap_hi']>=60)]
df.reset_index()
df.head()
```

In [ ]:
```
df[df['ap_lo']>120]
```

In [ ]:
```
df[df['ap_lo']<50]
```

```
In [ ]: df = df[(df['ap_lo']<=120) & (df['ap_lo']>=50)]
        df.reset_index()
        df.head()

In [ ]: df[df['ap_hi']<df['ap_lo']]

In [ ]: df=df[df['ap_hi']>df['ap_lo']]
        df[df['ap_hi']<df['ap_lo']]

In [ ]: df[df['gender']==2]['weight'].mean()

In [ ]: df[df['gender']==1]['weight'].mean()

In [ ]: # Safe to assume 1 represents Male and 2 represents Female as a female would weigh less than a man on an average

In [ ]: sns.distplot(df['age'])

In [ ]: # Gluc Smoke and alco are categorical variables so their skewness doesnt count
        df.skew()

In [ ]: # Dataset looks balanced from target perspective
        df['cardio'].value_counts()

In [ ]: plt.scatter(df['age'],df['height'],c=df['cardio'],alpha=0.4,s=0.5)
        plt.xlabel('Age in years')
        plt.ylabel('Height in cm')
        plt.legend(df['cardio'])

In [ ]: plt.scatter(df['age'],df['weight'],c=df['cardio'],alpha=0.4,s=0.5)
        plt.xlabel('Age in years')
        plt.ylabel('Weight in Kg')
        plt.legend(df['cardio'])

In [ ]: # We can see a slight correlation of increasing weight with presence of cardio vascular diseases

In [ ]: plt.scatter(df['ap_hi'],df['ap_lo'],c=df['cardio'],alpha=0.4,s=2)

In [ ]: # Similar result is observed in people having high systolic and diastolic blood pressure

In [ ]: sns.countplot(data=df,x='smoke')

In [ ]: sns.countplot(data=df,x='cholesterol',hue='cardio')

In [ ]: sns.countplot(data=df,x='alco',hue='cardio')

In [ ]: sns.countplot(data=df,x='active',hue='cardio')

In [ ]: sns.countplot(data=df,x='gluc',hue='cardio')
```

In [ ]:
```
plt.figure(figsize=(10,10))
sns.heatmap(df.corr(),annot=True)
```

In [ ]:
```
# ap_lo has high correlation with ap_hi and a lower correlation than ap_hi with cardio, so we are dropping it

df.drop(['ap_lo'],axis=1,inplace=True)
df.head()
```

In [ ]:
```
df.dtypes
```

In [ ]:
```
# Classification problem
```

In [ ]:
```
df.skew()
```

In [ ]:
```
df.hist()
```

In [ ]:
```
df.columns
```

In [ ]:
```
sns.countplot(data=df,x='smoke')
```

In [ ]:
```
# Values for machine learning model
X = df.iloc[:,:-1].values
y = df['cardio'].values
```

In [ ]:
```
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score,precision_score, confusion_matrix
```

In [ ]:
```
# Splitting the dataset into training and testing sets. Test size is 30% while training size is 70%. Random state se
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3,random_state=101)
```

In [ ]:
```
from sklearn.preprocessing import StandardScaler
```

In [ ]:
```
sc=StandardScaler()
```

In [ ]:
```
# Standardizing the dataset. Training dataset is standardized with training mean while testing data is also standard
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

In [ ]:
```
X_train.shape
```

In [ ]:
```
# Logistic regression model on training data.
from sklearn.linear_model import LogisticRegression
lr=LogisticRegression()
lr.fit(X_train,y_train)
```

In [ ]:
```
# We care about precision as we are interested in correct classification of 1s and 0s hence precision as a metric.
y_pred=lr.predict(X_test)
print(confusion_matrix(y_pred,y_test))
print(precision_score(y_pred,y_test))
```

In [ ]:
```
# Support vector classifier as 2nd model with precision as a metric.
from sklearn import svm
svc=svm.SVC()
svc.fit(X_train,y_train)
y_pred=svc.predict(X_test)
print(confusion_matrix(y_pred,y_test))
print(precision_score(y_pred,y_test))
```

In [ ]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import GaussianNB,BernoulliNB,MultinomialNB
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier,AdaBoostClassifier,BaggingClassifier,GradientBoostingClassifier
from xgboost import XGBClassifier
```

In [ ]:
```
lr = LogisticRegression()
gnb = GaussianNB()
bnb = BernoulliNB()
bgx = XGBClassifier()
cbg = GradientBoostingClassifier()
cb = BaggingClassifier()
cba = AdaBoostClassifier()
cfr = RandomForestClassifier()
cnk = KNeighborsClassifier()
ctd = DecisionTreeClassifier()
cvs = SVC()
```

In [ ]:
```
models = {'Logistic Regression':lr,
          'Gaussian NB' : gnb,
          'Bernoulli NB' : bnb,
          'Support Vector Classifier' : svc,
          'Decision Tree Classifier' : ctd,
          'K Neighbors Classifier' : cnk,
          'Random Forest Classifier' : cfr,
          'Ada Boost Classifier' : cba,
          'Bagging Classifier' : cb,
          'Gradient Boosting Classifier' : cbg,
          'XGBoost Classifier' : bgx}
```

In [ ]:
```
for name,algo in models.items():
  algo.fit(X_train,y_train)
  y_pred = algo.predict(X_test)
  confusion = confusion_matrix(y_pred,y_test)
  accuracy = accuracy_score(y_pred,y_test)
  precision = precision_score(y_pred,y_test)
  print(f'\nModel : {name}')
  print(f'Confusion Matrix ')
  print(confusion)
  print(f'Accuracy Score ')
  print(accuracy)
  print('Precision Score ')
  print(precision)
```

In [ ]:
```
from keras import Sequential
from keras.layers import Dense
```

In [ ]:
```
classifier = Sequential()
```

In [ ]:
```
classifier.add(Dense(units = 20, activation = 'relu',input_shape=(10,)))
classifier.add(Dense(units = 20, activation = 'relu'))
classifier.add(Dense(units = 1, activation = 'sigmoid'))
```

In [ ]:
```
classifier.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
```

In [ ]:
```
ann=classifier.fit(X_train,y_train,verbose=1)
```

In [ ]:
```
# Since gradient boosting classifier had highest metrics we perform hyperparameter tuning on it
```

Hyper parameter Tuning

In [ ]:
```
from sklearn.metrics import roc_curve, auc
```

In [ ]:
```
learning_rates=[1,0.5,0.1,0.05,0.01]
n_estimators=[1,10,20,50,100,200]
max_depth=[1,2,3,4,5,6,7,8,9,10]
minsamplessplit=[2,5,10,50,100,500]
minsamplesleaf=[1,5,10,50,100,500]
maxfeatures=list(range(1,X_train.shape[1]))
```

In [ ]:
```
train=[]
test=[]
for x in learning_rates:
  gbc=GradientBoostingClassifier(learning_rate=x)
  gbc.fit(X_train,y_train)

  y_train_pred=gbc.predict(X_train)
  fpr,tpr,t=roc_curve(y_train,y_train_pred)
  final = auc(fpr,tpr)
  train.append(final)

  y_test_pred=gbc.predict(X_test)
  fpr,tpr,t=roc_curve(y_test,y_test_pred)
  final = auc(fpr,tpr)
  test.append(final)

plt.plot(learning_rates,test,c='r',label='Test')
plt.plot(learning_rates,train,c='b',label='Train')
plt.legend()
plt.xlabel('Learning Rate')
```

In [ ]:
```
train=[]
test=[]
for x in n_estimators:
  gbc=GradientBoostingClassifier(n_estimators=x)
  gbc.fit(X_train,y_train)

  y_train_pred=gbc.predict(X_train)
  fpr,tpr,t=roc_curve(y_train,y_train_pred)
  final = auc(fpr,tpr)
  train.append(final)

  y_test_pred=gbc.predict(X_test)
  fpr,tpr,t=roc_curve(y_test,y_test_pred)
  final = auc(fpr,tpr)
  test.append(final)

plt.plot(n_estimators,test,c='r',label='Test')
plt.plot(n_estimators,train,c='b',label='Train')
plt.legend()
plt.xlabel('N_estimators')
```

In [ ]:
```
train=[]
test=[]
for x in max_depth:
  gbc=GradientBoostingClassifier(max_depth=x)
  gbc.fit(X_train,y_train)

  y_train_pred=gbc.predict(X_train)
  fpr,tpr,t=roc_curve(y_train,y_train_pred)
  final = auc(fpr,tpr)
  train.append(final)

  y_test_pred=gbc.predict(X_test)
  fpr,tpr,t=roc_curve(y_test,y_test_pred)
  final = auc(fpr,tpr)
  test.append(final)

plt.plot(max_depth,test,c='r',label='Test')
plt.plot(max_depth,train,c='b',label='Train')
plt.legend()
plt.xlabel('Max_Depth')
```

In [ ]:
```
train=[]
test=[]
for x in minsamplessplit:
  gbc=GradientBoostingClassifier(min_samples_split=x)
  gbc.fit(X_train,y_train)

  y_train_pred=gbc.predict(X_train)
  fpr,tpr,t=roc_curve(y_train,y_train_pred)
  final = auc(fpr,tpr)
  train.append(final)

  y_test_pred=gbc.predict(X_test)
  fpr,tpr,t=roc_curve(y_test,y_test_pred)
  final = auc(fpr,tpr)
  test.append(final)

plt.plot(minsamplessplit,test,c='r',label='Test')
plt.plot(minsamplessplit,train,c='b',label='Train')
plt.legend()
plt.xlabel('Min_samples_split')
```

In [ ]:
```
train=[]
test=[]
for x in minsamplesleaf:
  gbc=GradientBoostingClassifier(min_samples_leaf=x)
  gbc.fit(X_train,y_train)

  y_train_pred=gbc.predict(X_train)
  fpr,tpr,t=roc_curve(y_train,y_train_pred)
  final = auc(fpr,tpr)
  train.append(final)

  y_test_pred=gbc.predict(X_test)
  fpr,tpr,t=roc_curve(y_test,y_test_pred)
  final = auc(fpr,tpr)
  test.append(final)

plt.plot(minsamplesleaf,test,c='r',label='Test')
plt.plot(minsamplesleaf,train,c='b',label='Train')
plt.legend()
plt.xlabel('Min_samples_leaf')
```

In [ ]:
```
train=[]
test=[]
for x in maxfeatures:
  gbc=GradientBoostingClassifier(max_features=x)
  gbc.fit(X_train,y_train)

  y_train_pred=gbc.predict(X_train)
  fpr,tpr,t=roc_curve(y_train,y_train_pred)
  final = auc(fpr,tpr)
  train.append(final)

  y_test_pred=gbc.predict(X_test)
  fpr,tpr,t=roc_curve(y_test,y_test_pred)
  final = auc(fpr,tpr)
  test.append(final)
```