***Article Received:** 25 November 2023 **Revised:** 12 December 2023 **Accepted:** 30 January 2024*

A Comparative Study Utilizing Machine Learning Algorithms to Predict Heart Disease in Young and Middle-Aged Adults

Charu Kaushik
Dept . Computer science and Engineering
Manav Rachna International Institute of Research and Studies (MRIIRS)
Faridabad,Haryana,India
charukaushik161263@gmail.com

Kamlesh Sharma
Dept . Computer science and Engineering
Manav Rachna International Institute of Research and Studies (MRIIRS)
Faridabad,Haryana,India
associatedean_ks.academics@mriu.edu.in

Abstract— Early diagnosis is crucial since heart disease is getting more and more common. In the field of medicine, machine learning algorithms are now used to predict cardiac and cardiovascular illness. examining and confirming the functionality of machine learning. Heart disease is becoming more and more commonplace worldwide. A multitude of factors impact the likelihood of a heart attack and other illnesses. In many countries, limited cardiovascular competency makes it difficult to predict complications related to heart disease. One way to predict the possibility of a heart disease-related issue is to use data mining and machine learning techniques to identify which machine learning classifiers are most accurate for various diagnostic applications. Several supervised machine-learning algorithms are evaluated for their effectiveness in predicting cardiac illness. Use the heart disease individual dataset available via Kaggle. This work employs several machine-learning algorithms, including. Using Logistic Regression (LR), Navie Bayes (NB), Extreme Gradient Boost (EGB), K-Nearest Neighbor (K-NN), Support Vector Classifier (SVC), Random Forest (RF), and Decision Tree (DT), a neural network is constructed. Capable of categorizing binary data. For every feature across all deployed, estimated feature significance ratings were supplied. Ways. This helps identify the main risk factors for heart disease in addition to increasing model accuracy and assisting in the best forecast. Lastly, in comparison to all machine learning methods and Neural. The Binary Classification Neural Network, as a network model, produced the highest testing accuracy of more than 90%.

Keywords-Machine Learning , Heart disease , Classification , Neural Network

I. INTRODUCTION

An estimated 17.9 million people worldwide die each year from cardiovascular diseases (CVDs), which account for 31% of all deaths worldwide. Heart attacks and strokes are linked to four out of every five CVD deaths, and under-70s account for one third of these premature deaths. Eleven qualities in this dataset can be used to conjecture the probability of a cardiovascular condition. Heart failure is a common consequence of CVDs. A machine learning model can be very helpful in the early detection and treatment of cardiovascular disease and high-risk patients when one or more risk factors, such as diabetes, hypertension, hyperlipidemia, or an existing condition, are present.

Machine learning (ML)-based disease prediction is now possible thanks to the ever-increasing amount of medical data. In the medical field, ML techniques are frequently utilized. Several of the most well-known machine learning methods, such as the K-Nearest Neighbor, Random Forest, Naive Bayes classifier, Support Vector Machine, and Decision tree, were used in the study to predict heart disease. We also want to conduct a comparative analysis by contrasting the accuracy metrics of the ML algorithms used to predict heart disease. The dataset for the study was downloaded from Kaggle in the csv format and included data mining operations like data collection, data cleaning, data preprocessing, and exploratory data analysis. A comparison of the various machine learning methods used in categorization is provided in the research. With the dataset utilized in the review, the arbitrary timberland yielded the most noteworthy exactness rate, it was closed [1]. Heart disease is one of the world's leading causes of death. 17.9 million people worldwide die each year from heart disease, or 31% of all deaths worldwide. Heart conditions, particularly cardiac arrest, can occur at any time, in any location, and without any symptoms. As a result, determining a patient's likelihood of developing heart disease can help both patients and doctors recognize the possibility of cardiac arrest and take the necessary precautions. The gamble of coronary illness difficulties can be diminished, and effective and preventive patient treatments can be significantly helped by an early finding of coronary illness. A machine learning method is used to train models for the prediction of heart disease in this work using clinical patient data. A correlation analysis of the features in the data was conducted to assist with the study's feature selection. The effectiveness of five machine learning approaches—Naive Bayes, Decision Tree, K-Nearest Neighbor, Support Vector Machine, and Logistic Regression—was then examined. 13 clinical factors were used to train models that would predict cardiac disease, and the results were obtained. Logistic Regression appears to perform well in comparison to the other methods. [2]. One area of flow logical interest is the expectation of cardiovascular illness utilizing AI (ML) calculations. Using the UC Irvine (UCI) Cleveland Heart Disease dataset, this study investigates the efficacy of multiple classifiers, including K-Nearest Neighbors (KNN), AdaBoost, Gaussian Naive Bayes (GNB), support vector machines (SVM), multilayer perceptron (MLP), and random forests. This study aims to compare the speed, precision, and overall efficacy of each classifier using metrics like accuracy and F1 score. This study investigates an additional advantage of fusion approaches for increasing the accuracy of heart disease prediction. The study suggests that the measures could be improved by combining multiple models. The examination concerning coronary illness is as yet in progress, and our review adds to it. Upgrading clinical decision-production for coronary illness avoidance and treatment can be worked with by applying the consequences of our work to make more exact models for coronary illness expectation [3].

Heart disease (HD) is becoming more common by the day, so early detection is essential. Machine learning algorithms (MLA) are currently being used to predict heart or cardiovascular disease in the healthcare industry. Data mining methods like reinforcement, unsupervised, and supervised are useful for examining the vast amount of data in the medical field industry. The Cleveland database of the HD individual dataset in the UCI repository is used to test and validate MLA's performance. An early prediction of HD is made using several machine learning algorithms in this paper, including logistic regression (LR), random forest (RF), and decision tree (DT). The most reliable calculation, with a 94.7 percent precision rate, is the DT, as per our correlation investigation of the three strategies. This value is lower than the previously reported figure of 83.87 percent [4]. Machine learning has found applications in numerous sectors worldwide. It has proven to be extremely useful in healthcare, particularly for diagnosing conditions like heart disease and motor disorders. This study centres around utilizing popular AI calculations to gauge patients' probability of creating heart issues. This study compares decision trees, logistic regression, Naive Bayes, support vector machines (SVM), random forest (RF), and decision trees. The objective is to sort the best classifier so that accurate and reliable predictions can be made. An ensemble classifier that combines weak and strong classifiers is also suggested by the study. This hybrid strategy aims to boost the classifier's performance by employing many training and validation samples. Using the suggested ensemble classifier and the results of the comparison analysis, medical diagnosis and treatment customization could advance. At long last, AI procedures can possibly significantly work on persistent results and in general medical care quality when applied to the medical care area [5]. Coronary heart disease (CHD) is one of the leading causes of death worldwide. Bangladesh and other developing nations face similar difficulties. Until it's too late, many people don't realize that their heart problems are getting worse. As a result, early detection is essential for reducing the number of CHD-related fatalities and serious health effects. Using supervised machine learning (ML) methods, the primary objective of this work is to improve CHD prediction accuracy in the Bangladeshi population. Our research strategy makes use of machine learning methods like KNN, Random Forest, Decision Tree, Naive Bayes, and Binary Logistic Regression Model to predict CHD on two distinct datasets, one from Bangladesh and the other from Canada. ADASYN was used to generate synthetic data for the Bangladeshi dataset with an accuracy of 88.12 percent. However, SMOTE's accuracy was approximately 93.79 percent. By utilizing the Irregular Timberland Calculation, the two exactness's were accomplished. The Canadian dataset yielded the most noteworthy precision (72.33%) for Paired Strategic Relapse [6]. Predicting cardiovascular disease is one of today's most challenging medical tasks. Heart problems have been claiming the lives of approximately one person every minute in recent times. When it comes to analysing the enormous amounts of data in the healthcare sector, machine learning is necessary. Coronary illness expectation is a mind-boggling task, so to forestall related dangers and ready patients, the forecast interaction should be robotized. The proposed study uses a variety of machine learning methods, such as Random Forest, SVM, decision trees, logistic regression, and threat classification, to classify patient risk and predict the likelihood of heart disease. This paper employs comparative analysis to compare the capabilities of various machine learning methods. The SVM algorithm has the highest efficiency, at 94%, when compared to other machine learning techniques. [7]. Due to its significant social impact, improving heart disease detection and treatment has grown in importance. By data mining and the storage of medical records, improved patient management is now possible thanks to the integration of technology and medical diagnosis. Understanding the connection between risk factors in a patient's medical history and how those factors affect their likelihood of developing heart disease is crucial. This study aims to accurately predict heart disease by examining a variety of patient data points. The feature extraction and selection methods are the best choices for a heart disease prediction system. Age, sex, occupation, smoking, obesity, diet, exercise, mental stress levels, type of chest pain, history of chest pain, pressure, ECG, and outcomes have been found to be significant in the diagnosis of heart disease. The heart disease case was analyzed using a variety of machine learning methods, including Multilayer Perceptron (MLP), Naive Bayes (NB), K-nearest Neighbor (K-NN), and Support Vector Machine (SVM). Both datasets contain all features as well as a subset of them. The objective of comparing their respective performances was to identify the methods that produced accurate forecasts. Because of utilizing the chose highlights, irregular woodland had the option to accomplish a greatest exactness pace of 90%, outflanking other computerized reasoning frameworks with the utilization of all info highlights. It appears that the method that is recommended is a promising model for supporting the early detection of heart disease. Our work speeds up treatment, predicts better cardiac illness outcomes, and improves patient outcomes by combining the power of data-driven AI algorithms with careful feature selection [8]. Quite possibly of the main sickness that is presently perceived is coronary illness. With the rising accessibility of data, a plenty of strategies and calculations have been made to precisely conjecture the forecast of coronary illness patients more. Using a Kaggle dataset, this study explains thirteen important processes. The most dependable outcomes, with a 93% exactness rate, were gotten utilizing Backing Vector Machine (SVM), K-Closest Neighbor (KNN), Gullible Bayes, and Irregular Woods. Each person's comparative statement algorithms are also included in the paper's implementation section. In this work, methods for model validation are also used to make the best model for the circumstances. [9]. According to research, cardiovascular diseases (CVD) cardiovascular diseases (CVDs) are prevalent in the population and frequently result in death, according to research. As per late reports, extreme work pressure joined with pressure and hypertension, weight, post-Coronavirus side effects, and way of life changes welcomed on by the pandemic are contributing elements to the sharp expansion in coronary episodes and cardiovascular issues. Decreased developing CVD passing rates must be accomplished by early recognizable proof. In any case, it's trying to watch out for patients and proposition guidance when there aren't an adequate number of prepared doctors or clinical offices. However, the medical industry collects such a large amount of data that it can sometimes be difficult to view and comprehend. An expectation model can be made by utilizing similar information to prepare an AI calculation. This kind of prediction system might be helpful to patients as well as doctors. The various cardiovascular conditions and their risk factors are the subject of this study. This study examines some of the most widely used machine learning algorithms for predicting heart disease using historical data and medical records. We have compared fundamental machine learning algorithms like logistic regression, SVM, and random forest in this article. [10].

II. Literature Review

One of the most difficult jobs in medicine is predicting heart illness. Figuring out what's causing this requires a lot of time and work, particularly for medical professionals like doctors. This study predicts cardiac disease using the GridSearchCV in conjunction with a few machines learning methods, including LR, KNN, SVM, and GBC. Five-fold cross-validation is the method used by the system to verify its functionality. For each of these four approaches, a comparison is provided. Performance analysis of the models is done using the datasets for Cleveland, Hungary, Switzerland, Long Beach V, and UCI Kaggle. For both datasets (Hungary, Switzerland &; Long Beach V and UCI Kaggle), the analysis reveals that the Extreme Gradient Boosting Classifier with GridSearchCV yields the highest and almost comparable testing and training accuracies of 100% and 99.03%. Additionally, the study reveals that, for both datasets (Hungary, Switzerland & Long Beach V and UCI Kaggle), the XGBoost Classifier without GridSearchCV yields the highest and almost equivalent testing and training accuracies of 98.05% and 100%. In addition, the suggested technique's analytical outcomes are contrasted with earlier research on heart disease prediction. The Extreme Gradient Boosting Classifier with GridSearchCV appears to be the most accurate hyperparameter among the suggested approaches for measuring accuracy. The main goal of this work is to provide a novel model-creation method for practical issue resolution [11]. One serious ailment that is not fully curable is coronary heart disease (CHD). Early identification of coronary artery disease can help doctors treat patients more effectively. HY_OptGBM, an optimized LightGBM classifier prediction model, was proposed in this work to predict CHD. To optimize the LightGBM classifier, the LightGBM model's hyperparameters were changed. Furthermore, the model was trained with modified hyperparameters, and its loss function was enhanced. Using the most sophisticated hyperparameter optimization framework (OPTUNA), the prediction model's hyperparameters were tuned in this study. The focused loss (FL) is the term used to describe the improved loss function. The Framingham Heart Institute's CHD data was used in this study to assess a prediction model. AUC, sensitivity, specificity, precision, recall, F score, accuracy, MCC, and accuracy were among the metrics used to assess the prediction model's performance. The suggested model outperformed other comparative models with an AUC score of 97.8%. The findings show that applying the suggested strategy can increase the rate of early detection of CHD in the general population. Thus, the expenses related to treating individuals with congestive heart failure may be lessened in turn [12].

Cardiovascular disease (CVD) is one of the major global causes of mortality. A simple, affordable method of diagnosing heart health is electrocardiography (ECG). In this work, a multi-class classifier utilizing a 12-lead electrocardiogram is presented for the prediction of four distinct forms of cardiovascular diseases: myocardial infarction, hypertrophy, conduction disturbances, and ST-T disturbance. Data preprocessing, feature extraction, data preparation and augmentation, and modelling for multi-class CVD classification are the four main processes in the work that is being presented. We train the classifier using the sixteen-time domain enhanced features. Three phases comprise the work: prepping and enhancing the data, training, testing, and verifying the classifier, and extracting features from the raw 12-lead ECG signals. The task is broken down into three stages: prepping and augmenting the data, extracting features from the raw 12-lead ECG signals, and training, validating, and testing the classifier. The effectiveness of five distinct classifiers—Random Forest (RF), K Nearest Neighbors (KNN), Gradient Boost, Adda Boost, and XG Boost—has also been compared and studied. Performance is assessed using F1 scores, accuracy, precision, and recall. In addition, the area under the curve (AUC) is computed and the receiver operating curve (ROC) traced to guarantee the classifier's impartial performance. There has also been talk about using the suggested classifier within the framework of Smart Healthcare [13].

Among the toughest problems facing the medical field is the prediction of cardiac disease. A range of methods, such as LDA, RF, GBC, DT, SVM, and KNN, together with the sequential feature selection feature selection technique, were used by researchers to predict cardiac illness. The K-fold cross-validation method is used by the system for verification. The comparative study was carried out using these six tactics. The models were evaluated using the following datasets: Heart Statlog Cleveland Hungary; Cleveland; Hungray; Switzerland; and Long Beach V. The highest and nearly identical accuracy scores (100%, 99.40% and 100%, 99.76% respectively) were

Article Received: *25 November 2023* ***Revised:*** *12 December 2023* ***Accepted:*** *30 January 2024*

obtained by Random Forest Classifier sfs and Decision Tree Classifier for the Hungary, Switzerland & Long Beach V and Heart Statlog Cleveland Hungary Dataset. The results were contrasted with earlier cardiac prediction-focused studies. Our long-term goal is to expand the model even more to enable its use with different feature selection methods; an additional option would be to employ a random forest classifier. The primary objective of this research is to enhance earlier efforts by devising an innovative and distinct method for constructing the model, and to render the model applicable and user-friendly in practical scenarios [14].

Heart failure and aortic stenosis are two prevalent and devastating cardiovascular disorders that also carry an increased risk of dementia in the elderly. Early detection may help avoid or treat certain conditions, possibly lowering death rates. Using demographic and medical data, machine learning algorithms—especially gradient boosting (GB)—can accurately predict the existence of certain diseases using binary categorization. The goal of the current study is to address the lack of research combining data from all three disorders for multiclass classification. A GB-based model is presented for the multiclass classification of elderly individuals with heart failure, aortic stenosis, and dementia using a dataset gathered from Chiang Rai Prachanukroh Hospital, Chiang Rai, Thailand. Feature engineering approaches are included for optimal accuracy. Additional trees, random forests, k-nearest neighbors, decision trees, support vector machines, and other techniques were used as a comparison. The tree structured Parzen estimator was utilized in conjunction with the Optuna framework to optimize hyperparameters. Preciseness, recall, accuracy, F1 score, area under the receiver operating characteristic curve, area under the precision-recall curve, and Matthews's correlation coefficient were among the performance metrics used to compare the output of each classifier. For comparison, each machine learning algorithm's results are shown separately. Based on these measures, it can be said that, once feature engineering approaches were used, our suggested GB-based model performed better than other comparable models [15].

The classification approach, a machine learning methodology, is used to diagnose heart illness and produces effective results since early detection of heart disease is critical to an individual's survival. In this work, eight distinct machine learning classification techniques were used to examine data from the 2020 survey on the Behavioural Risk Factor Surveillance System (BRFSS) provided by the Centres for Disease Control and Prevention (CDC). These techniques include Adaboost, Multilayer Perceptron (MLP), XGBoost (XGB), k-Nearest Neighbor (k-NN), Logistic Regression (LR), Support Vector Machines (SVM), Naive Bayes (NB), and Decision Tree (DT). The data presented, however, shows an imbalanced distribution of the nominal dependent variable, which is heart disease. To solve this issue, the Synthetic Minority Oversampling Technique Tomek Links (SMOTE -Tomek Link) method was used to stabilize the dependent variable before the classification techniques were applied. The creation of synthetic data has eliminated the imbalance in the data. The data were split into outliers and non-outliers, and an outlier analysis was carried out using traditional statistical techniques to get strong and objective estimations. 10-fold cross-validation was used during the classification procedure to producemore consistent results and improve the comparability of the approaches' performances. As a result, the XGB algorithm was able to identify patterns in the diagnosis of heart illness and diagnose diseases at an accuracy rate of 89% for non-outlier data and 84.6% for outlier data. But in a relatively short amount of time, the k-NN algorithm was able to attain an accuracy rate of 85.6% in the non-outlier data and 81% in the outlier data. To diagnose cardiac disease early and find patterns in the disease, XGB and k-NN algorithms will be used [16].A few conditions, diabetes and heart disease being two of the most common, are the leading causes of death in the modern world. A major source of worry in clinical data analysis is the difficulty of predicting various diseases. The effectiveness of machine learning in supporting decision-making and managing the large amount of healthcare data has been demonstrated. Machine learning is being used to forecast diseases in a few active research. This research presents a novel method to improve universal illness detection by utilizing machine learning techniques to discover important traits. It includes building predictive models for diabetes and heart disease using a variety of characteristics and tried-and-true categorization techniques. The suggested technique uses an ensemble approach and includes a voting classifier that combines AdaBoost, sigmoidal, to attain more accuracy, use SVC and decision tree algorithms. Furthermore, the study applies conventional classifiers and presents a comparative analysis of their individual performances, exposing differences in accuracy [17].

III. Methodology

Python was used in the current study to categorize the dataset on heart disease. It provides an easy-to-use visual representation of the workspace, the dataset, and the process of developing predictive analytics. Pre-processing the data is the first step in the machine learning process, after which feature selection follows. This study's primary contribution was the development of an understandable medical prediction system for heart disease diagnosis using contemporary machine learning techniques. This study trained a neural network that can classify binary data using a variety of machine learning classifier algorithms, such as Logistic Regression (LR), Navie Bayes (NB), Extreme Gradient Boost (EGB), K-Nearest Neighbor (K-NN), Support Vector Classifier (SVC), Random Forest (RF), and Decision Tree (DT).

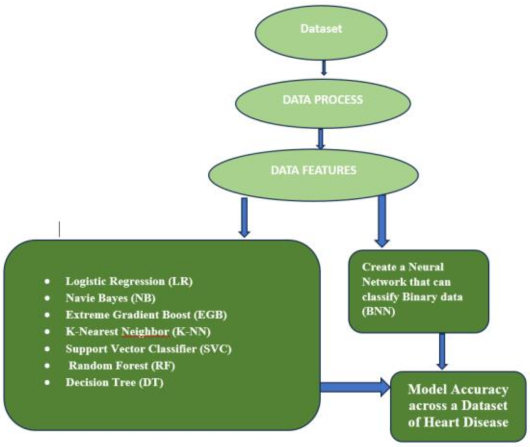

Dataset
DATA PROCESS
DATA FEATURES
Logistic Regression (LR)
Navie Bayes (NB)
Extreme Gradient Boost (EGB)
K-Nearest Neighbor (K-NN)
Support Vector Classifier (SVC)
Random Forest (RF)
Decision Tree (DT)
Create a Neural Network that can classify Binary data (BNN)
Model Accuracy across a Dataset of Heart Disease

Fig 1: Methodology Follow

A. Data Preprocessing

Following the acquisition of creative datasets, pre-processing was done on the gathered data. To choose pertinent traits, the sequential forward selection method was applied. To optimize the parameters of the hyper parameter optimization techniques search was employed. Lastly, to see whether the models could forecast cardiac illness, they were examined and tested. Using 5-fold cross-validation, the data in our suggested model are verified. The model that is suggested Logistic Regression (LR), Navie Bayes (NB), Extreme Gradient Boost (EGB), K-Nearest Neighbor (K-NN), Support Vector Classifier (SVC), Random Forest (RF), and Decision Tree (DT) and create a neural network that can classify Binary data.

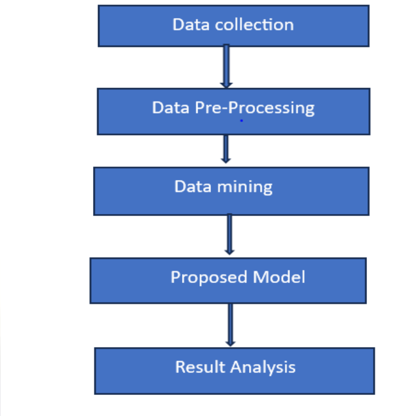
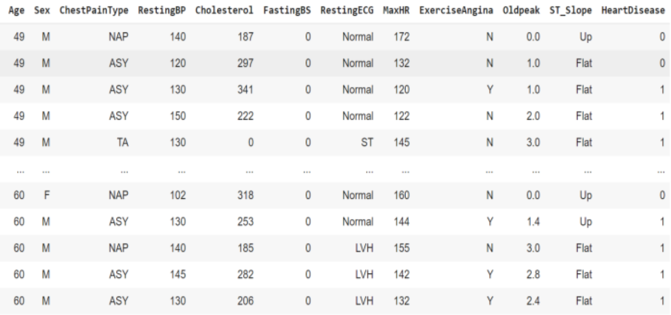


Fig 2: Data Procedure

B. Dataset

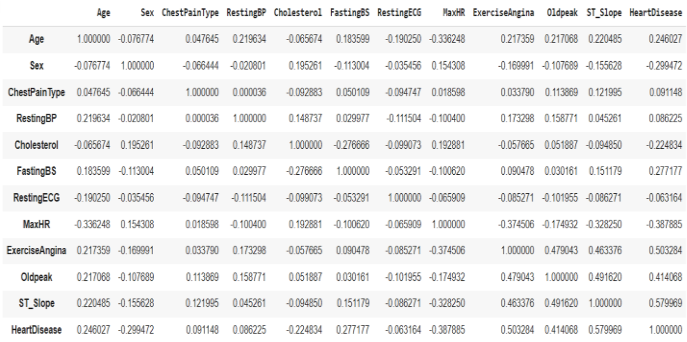
Table 1: Qualitative Data Table

Attributes	Sort
Patient's Age	In Years
Patient's Sex	[M: Male, F: Female]
Type of Chest Pain (ChestPainType)	[TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic]
Blood Pressure (RestingBP)	Getting some rest Blood pressure at rest (in millimeters mercury
Cholesterol	Serum cholesterol is measured in milligrams per deciliter.
Blood sugar (Fasting BS)	Fasting blood sugar, or FastingBS: [1: if FastingBS > 120 mg/dl, 0: otherwise]
Electrocardiogram (RestingECG)	Results of a resting electrocardiogram [Normal: Typical— According to Estes' criteria, LVH indicates probable or definitive left ventricular hypertrophy. ST is defined as having ST-T wave abnormalities (T wave inversions and/or ST elevation or depression of > 0.05 mV).
Maximum heart rate (MaxHR)	A numeric value between 60 and 202
ExerciseAngina (	Exercise-induced angina [Y: Yes, N: No] is also known as Exercise Angina.
Oldpeak	ST (a numerical value expressed in terms of depression)
ST_Slope	The peak exercise's slope ST section [Down: down sloping, Up: upsloping, Flat: flat]
HeartDisease	output class [0: Normal, 1: heart disease]



Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina	Oldpeak	ST_Slope	HeartDisease
49	M	NAP	140	187	0	Normal	172	N	0.0	Up	0
49	M	ASY	120	297	0	Normal	132	N	1.0	Flat	0
49	M	ASY	130	341	0	Normal	120	Y	1.0	Flat	1
49	M	ASY	150	222	0	Normal	122	N	2.0	Flat	1
49	M	TA	130	0	0	ST	145	N	3.0	Flat	1
...	...	...	...	...	...	...	...	...	...	...	...
60	F	NAP	102	318	0	Normal	160	N	0.0	Up	0
60	M	ASY	130	253	0	Normal	144	Y	1.4	Up	1
60	M	NAP	140	185	0	LVH	155	N	3.0	Flat	1
60	M	ASY	145	282	0	LVH	142	Y	2.8	Flat	1
60	M	ASY	130	206	0	LVH	132	Y	2.4	Flat	1

Fig 3: Heart Disease Dataset



	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina	Oldpeak	ST_Slope	HeartDisease
Age	1.000000	-0.076774	0.047645	0.219634	-0.065674	0.183599	-0.190250	-0.336248	0.217359	0.217068	0.220485	0.246027
Sex	-0.076774	1.000000	-0.066444	-0.020801	0.195261	-0.113004	-0.035456	0.154308	-0.169991	-0.107689	-0.155628	-0.299472
ChestPainType	0.047645	-0.066444	1.000000	0.000036	-0.092883	0.050109	-0.094747	0.018598	0.033790	0.113869	0.121995	0.091148
RestingBP	0.219634	-0.020801	0.000036	1.000000	0.148737	0.029977	-0.111504	-0.100400	0.173298	0.158771	0.045261	0.086225
Cholesterol	-0.065674	0.195261	-0.092883	0.148737	1.000000	-0.276666	-0.099073	0.192881	-0.057665	0.051887	-0.094850	-0.224834
FastingBS	0.183599	-0.113004	0.050109	0.029977	-0.276666	1.000000	-0.053291	-0.100620	0.090478	0.030161	0.151179	0.277177
RestingECG	-0.190250	-0.035456	-0.094747	-0.111504	-0.099073	-0.053291	1.000000	-0.065909	-0.085271	-0.101955	-0.086271	-0.063164
MaxHR	-0.336248	0.154308	0.018598	-0.100400	0.192881	-0.100620	-0.065909	1.000000	-0.374506	-0.174932	-0.328250	-0.387885
ExerciseAngina	0.217359	-0.169991	0.033790	0.173298	-0.057665	0.090478	-0.085271	-0.374506	1.000000	0.479043	0.463376	0.503284
Oldpeak	0.217068	-0.107689	0.113869	0.158771	0.051887	0.030161	-0.101955	-0.174932	0.479043	1.000000	0.491620	0.414068
ST_Slope	0.220485	-0.155628	0.121995	0.045261	-0.094850	0.151179	-0.086271	-0.328250	0.463376	0.491620	1.000000	0.579969
HeartDisease	0.246027	-0.299472	0.091148	0.086225	-0.224834	0.277177	-0.063164	-0.387885	0.503284	0.414068	0.579969	1.000000

Fig 4: Features of the data, using the built-in pandas describe.

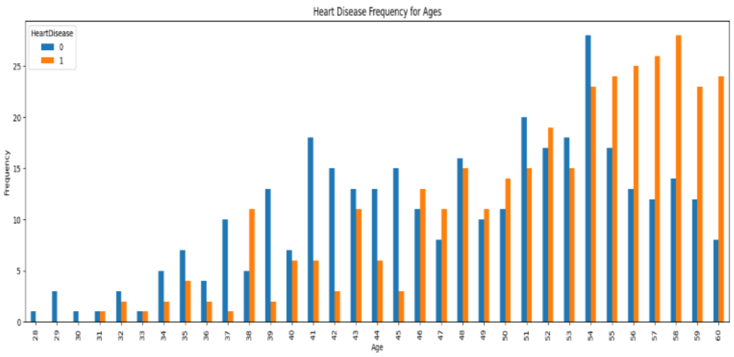


Fig 5: Represents the Heart Disease Frequency Ages

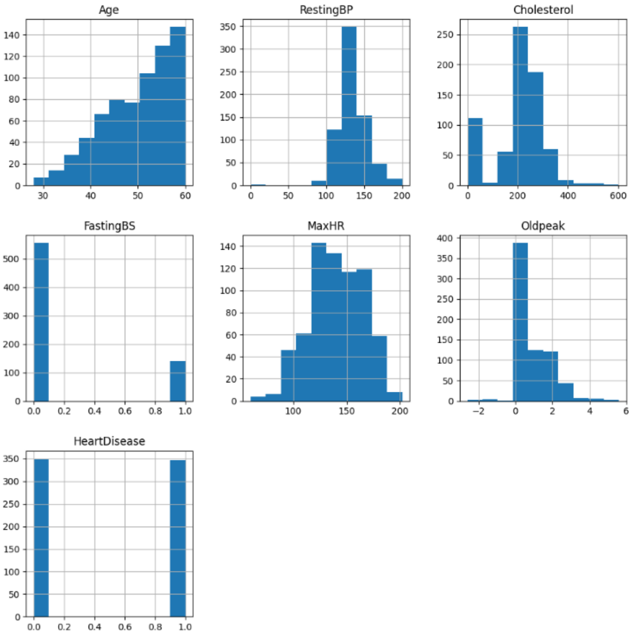


Fig 6: Each variable's histogram

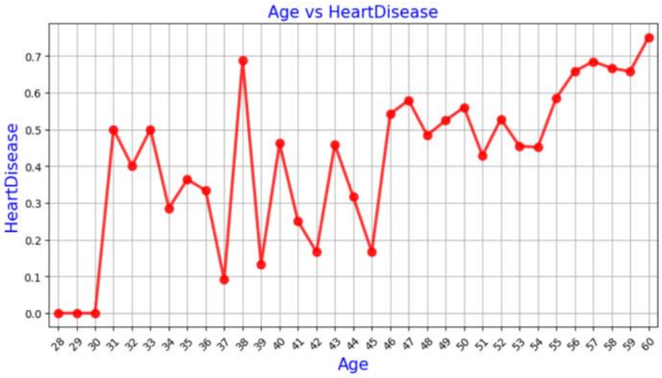


Fig 7: Plot Age Vs Heart Disease

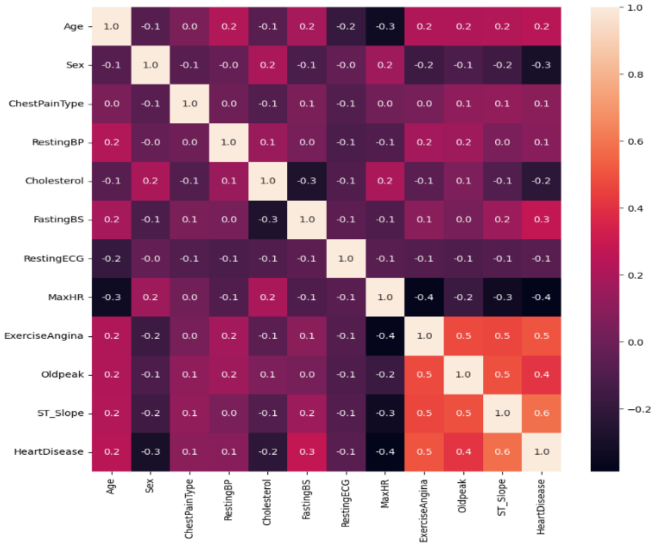


Fig 8: Heatmap

C. *Models used*

a) Creating Binary Classification Neural Network Model

Here, we'll go into a little more detail on binary classification. Binary classification is the term used to describe classification issues with two class labels. The normal situation is represented by one class in most binary classification problems, and the aberrant condition is represented by the other. Neural network architecture is inspired by the structure of the human brain. Neural cells in the human brain communicate electrically with one another to form a sophisticated, highly interconnected network that aids in information processing. Analogously, artificial neurons comprise an artificial neural network, which collaborates to resolve an issue. Artificial neural networks are software programs or algorithms that, at their foundation, use computing systems to complete mathematical computations. Artificial neurons are software modules, also known as nodes.
Uncomplicated architecture for neural networks Three layers comprises an artificial neural network with interconnected neurons: The Input Layer: The artificial neural network's input layer is where external data comes in. Before moving on to the next layer, input nodes process, examine, or classify the data. Layer Hidden: Additional hidden layers or the input layer are the sources of input for hidden layers. The number of hidden layers in artificial neural networks can be very high. Before moving on to the next layer, each hidden layer examines and refines the output from the one before it. layer of output: The output layer displays the outcome of all the artificial neural network's data processing operations. Both one and more nodes are possible. An example of this would be an output layer with one output node that would return a result of 1 or 0 for a binary (yes/no) classification problem. But the output layer may contain multiple output nodes if we are dealing with a multi-class classification problem.

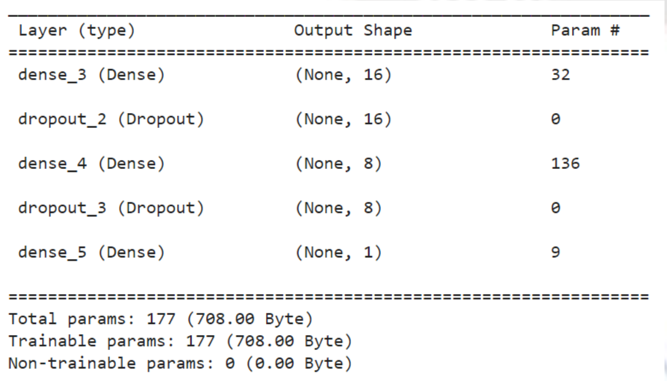
Steps involve in Neural Network:

Step 1: Establish a paradigm for approximation.
Step 2: Establish the data set.
Step 3: Specify the network architecture.
Step 4: Improve your neural network.
Step 5: Boost performance in generalization.
Step 6: Test outcomes.
Step 7: Use the model.

Algorithm Used:

```
# Modify the input_dim argument in the first Dense layer to match the shape of X_train
def create_binary_model():
 # Create model
 model = Sequential ()
model.add(Dense(16, input_dim=X_train.shape[1], kernel_initializer='normal',  kernel_regularizer=regularizers.l2(0.001),activation='relu'))
 model.add(Dropout(0.25))
```

```
    model.add(Dense(8,kernel_initializer='normal',  kernel_re
    gularizer=regularizers.l2(0.001),activation='relu'))
    model.add(Dropout(0.25))
     model.add(Dense(1, activation='sigmoid'))
     # Compile model
     adam = Adam(lr=0.001)
     model.compile(loss='binary_crossentropy',
optimizer='rmsprop', metrics=['accuracy'])
       return model.
    binary_model = create_binary_model()
```



```
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 dense_3 (Dense)              (None, 16)                32

 dropout_2 (Dropout)          (None, 16)                0

 dense_4 (Dense)              (None, 8)                 136

 dropout_3 (Dropout)          (None, 8)                 0

 dense_5 (Dense)              (None, 1)                 9

=================================================================
Total params: 177 (708.00 Byte)
Trainable params: 177 (708.00 Byte)
Non-trainable params: 0 (0.00 Byte)
```

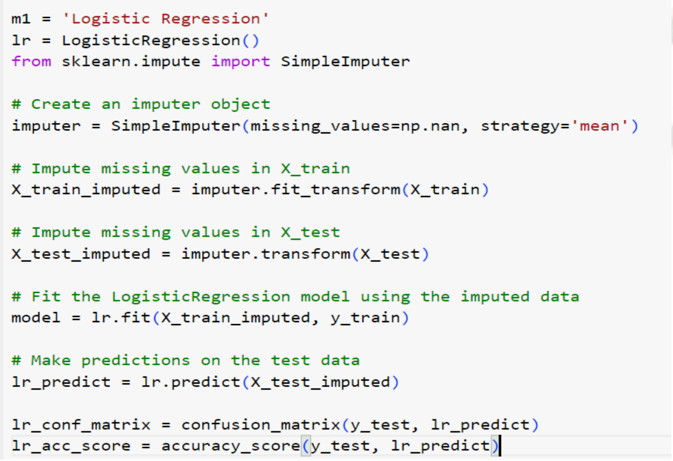
Fig 9: Neural Network Model Constructed

b) Logistic Regression

For categorization and predictive analytics, this kind of statistical model—also referred to as the logit model—is frequently employed. Because the result is a probability, the dependent variable has a range of 0 to 1. A logit transformation is performed to the odds in logistic regression, which are the probability of success divided by the probability of failure. This logistic function is also referred to as the log odds or the natural logarithm of odds.

1+exp(-pi) = 1/(logit(pi))

Beta_0 + Beta_1*X_1 +... + B_k*K_k = ln(pi/(1-pi))



```
m1 = 'Logistic Regression'
lr = LogisticRegression()
from sklearn.impute import SimpleImputer

# Create an imputer object
imputer = SimpleImputer(missing_values=np.nan, strategy='mean')

# Impute missing values in X_train
X_train_imputed = imputer.fit_transform(X_train)

# Impute missing values in X_test
X_test_imputed = imputer.transform(X_test)

# Fit the LogisticRegression model using the imputed data
model = lr.fit(X_train_imputed, y_train)

# Make predictions on the test data
lr_predict = lr.predict(X_test_imputed)

lr_conf_matrix = confusion_matrix(y_test, lr_predict)
lr_acc_score = accuracy_score(y_test, lr_predict)
```

Fig 10: Logistic Regression Algorithm

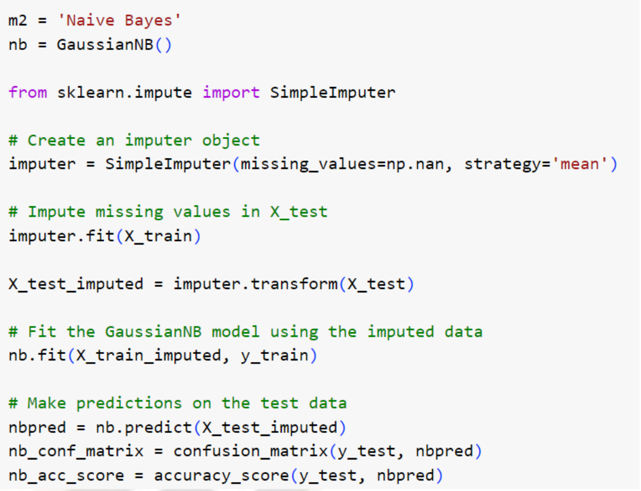
c) Navie Bayes

Using Bayes' Theorem as their foundation, naive Bayes classifiers are a group of classification algorithms. Instead of being a single algorithm, it is a family of algorithms that are united by the idea that each pair of features being classified stands alone. Let us first have a look at a dataset. Rapid machine learning model creation with fast prediction capabilities is facilitated by the Naïve Bayes classifier, one of the most straightforward and efficient classification methods. For classification issues, the Naïve Bayes method is employed. In text classification, it is heavily utilized. Since each word in the data represents a feature, text classification problems include high-dimensional data. It is applied in sentiment analysis, rating classification, spam filtering, and other areas.

Steps Involve in Navie Bayes:

Step 1: Divide into Classes.
Step2: Compile the dataset.
Step 3: Compile Information by Class.

Gaussian Probability Density Function is the fourth step.
Class Probabilities is Step 5.



```
m2 = 'Naive Bayes'
nb = GaussianNB()

from sklearn.impute import SimpleImputer

# Create an imputer object
imputer = SimpleImputer(missing_values=np.nan, strategy='mean')

# Impute missing values in X_test
imputer.fit(X_train)

X_test_imputed = imputer.transform(X_test)

# Fit the GaussianNB model using the imputed data
nb.fit(X_train_imputed, y_train)

# Make predictions on the test data
nbpred = nb.predict(X_test_imputed)
nb_conf_matrix = confusion_matrix(y_test, nbpred)
nb_acc_score = accuracy_score(y_test, nbpred)
```

Fig 11: Navie Bayes Algorithm

d) Random Forest Classifier

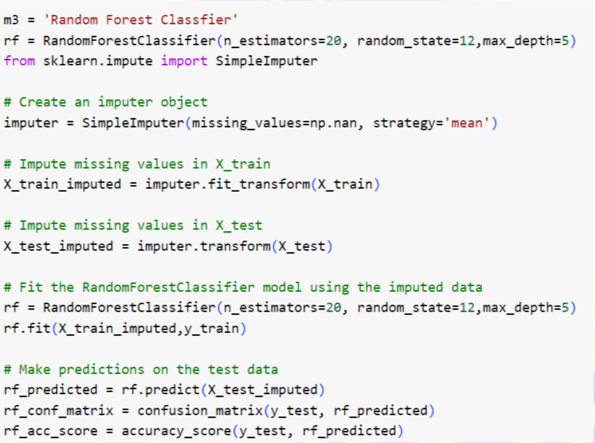
Within the category of supervised learning techniques comes the well-known machine learning algorithm Random Forest. It is applicable to machine learning issues involving both classification and regression. The foundation of this approach lies in the notion of ensemble learning, which involves merging several classifiers to address intricate issues and enhance the model's functionality. According to the description, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." The random forest predicts the outcome based on the majority votes of predictions made by each decision tree, as opposed to

Article Received: *25 November 2023* ***Revised:*** *12 December 2023* ***Accepted:*** *30 January 2024*

depending only on one decision tree. Because there are more trees in the forest, accuracy is higher, and overfitting is avoided.

Steps Involve in Random Forest algorithm:

Step 1: Choose K data points at random from the training set.
Step 2: Construct the decision trees linked to the chosen data points (subsets).
Step 3: Decide how many decision trees you want to construct by setting the number N.
Step 4: Go back to Steps 1 and 2.
Step 5: For each decision tree's predictions pertaining to new data points, identify which category the new data points belong to base on the majority vote.



```
m3 = 'Random Forest Classfier'
rf = RandomForestClassifier(n_estimators=20, random_state=12,max_depth=5)
from sklearn.impute import SimpleImputer

# Create an imputer object
imputer = SimpleImputer(missing_values=np.nan, strategy='mean')

# Impute missing values in X_train
X_train_imputed = imputer.fit_transform(X_train)

# Impute missing values in X_test
X_test_imputed = imputer.transform(X_test)

# Fit the RandomForestClassifier model using the imputed data
rf = RandomForestClassifier(n_estimators=20, random_state=12,max_depth=5)
rf.fit(X_train_imputed,y_train)

# Make predictions on the test data
rf_predicted = rf.predict(X_test_imputed)
rf_conf_matrix = confusion_matrix(y_test, rf_predicted)
rf_acc_score = accuracy_score(y_test, rf_predicted)
```

Fig 12: Random Forest Classifier Algorithm

e) Extreme Gradient Boost:

A machine learning algorithm under ensemble learning is called eXtreme Gradient Boosting, or XGBoost. This approach is used for supervised learning applications like classification and regression. With repeated iterations, XGBoost constructs a predictive model by aggregating the predictions of several distinct models—typically decision trees. The method involves gradually bringing in weaker members of the ensemble, each of them concentrating on fixing the mistakes committed by the previous members. During training, a preset loss function is minimized using a gradient descent optimization method.In order to prevent overfitting, regularization techniques are incorporated into the XGBoost Algorithm, which also incorporates parallel processing for efficient calculation and the capacity to handle complicated relationships in data.

There are three basic steps in the gradient boosting ensemble technique:

Step 1: Target variable y is predicted by an initial model, denoted as F0. A related residual (y – F0) will be used with this model.

Step 2: The residuals from the preceding phase are used to fit a new model, h1.

Step 3: This now results in F1, the boosted form of F0, from the combination of F0 and h1. Between F0 and F1, the mean squared error will be less.

$$F_1(x) <- F_0(x) + h_1(x)$$

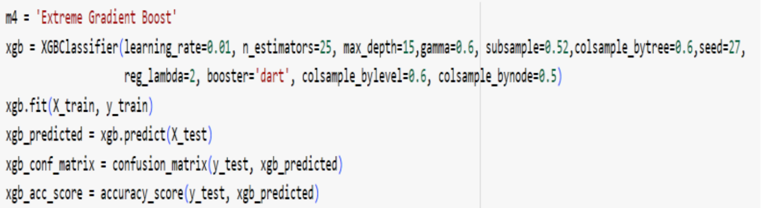
We might model after the F1 residuals and produce a new model, F2, to enhance the performance of F1:

$$F_2(x) <- F_1(x) + h_2(x)$$

Until residuals are as low as feasible, this can be carried out for 'm' iterations:

$$F_m(x) <- F_{m-1}(x) + h_m(x)$$

The functions established in the earlier steps are not disturbed in this case by the additive learners. Rather, they provide their own knowledge to correct the mistakes.



```
m4 = 'Extreme Gradient Boost'
xgb = XGBClassifier(learning_rate=0.01, n_estimators=25, max_depth=15,gamma=0.6, subsample=0.52,colsample_bytree=0.6,seed=27,
                    reg_lambda=2, booster='dart', colsample_bylevel=0.6, colsample_bynode=0.5)
xgb.fit(X_train, y_train)
xgb_predicted = xgb.predict(X_test)
xgb_conf_matrix = confusion_matrix(y_test, xgb_predicted)
xgb_acc_score = accuracy_score(y_test, xgb_predicted)
```

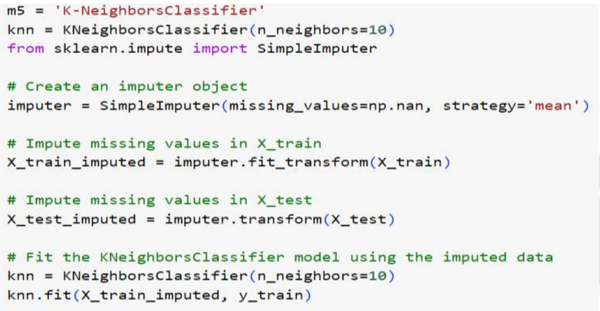
Fig 13: Extreme Gradient Boost Algorithm

f) K-Neighbors Classifier

One of the most basic machine learning algorithms, based on the supervised learning approach, is K-Nearest Neighbor. The new case is placed in the category most comparable to the existing categories by the K-NN method, which assumes that the new instance and its data are like the cases that are already accessible. A new data point is classified using the K-NN algorithm using similarity, which stores all the available data. This indicates that new data can be quickly and simply categorized using the K-NN algorithm into a well-suited category. The K-NN algorithm is mostly used to solve classification problems, while it can also be used to solve regression problems. The K-NN algorithm is mostly used to solve classification problems, while it can also be used to solve regression problems. Because K-NN is a non-parametric approach, it doesn't make any assumptions about the underlying data. It is also known as a lazy learner algorithm because, rather than learning straight away from the training set, it saves the dataset and acts on it when it comes time to classify. The KNN algorithm simply stores the dataset during the training phase and classifies fresh data into a category that closely resembles the stored data.
The following procedure provides an explanation of how K-NN functions:

Step 1: Decide which neighbor's K number to choose
Step 2: Determine the K number of neighbors' Euclidean distance.
Step 3: Using the computed Euclidean distance, choose the K closest neighbors.

Step 4: Determine how many data points there are in each category among these k neighbors.
Step 5: Put the additional data points in the category where the neighbor count is at its highest.
Step 6: The model is prepared.

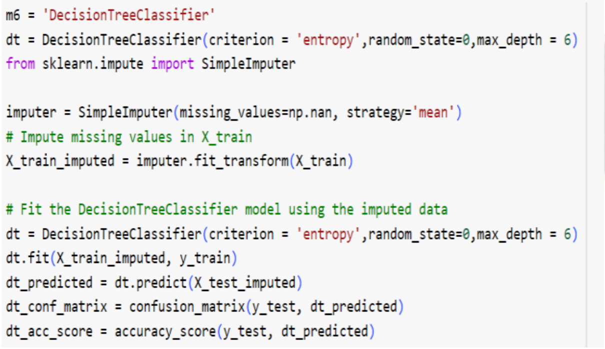


```
m5 = 'K-NeighborsClassifier'
knn = KNeighborsClassifier(n_neighbors=10)
from sklearn.impute import SimpleImputer

# Create an imputer object
imputer = SimpleImputer(missing_values=np.nan, strategy='mean')

# Impute missing values in X_train
X_train_imputed = imputer.fit_transform(X_train)

# Impute missing values in X_test
X_test_imputed = imputer.transform(X_test)

# Fit the KNeighborsClassifier model using the imputed data
knn = KNeighborsClassifier(n_neighbors=10)
knn.fit(X_train_imputed, y_train)
```

Fig 14: K-Neighbors Classifier Algorithm

g) Decision Tree Classifer :

In a Decision tree, the calculation starts at the root hub to gauge the class of the provided dataset. The strategy continues to the following hub by following the branch and looking at the upsides of the root quality with those of the record (genuine dataset) property. The strategy continues by looking at the quality worth of the resulting hub with that of the past sub-hubs by and by. The cycle is done until it arrives at the tree's leaf hub. The technique beneath can assist you with understanding the whole cycle better:
Stage 1: Root hub (containing whole dataset) ought to be the beginning stage of the tree, as indicated by S.
Stage 2: Utilize the Quality Choice Measure (ASM) to figure out which property in the dataset is the best.
Stage 3: Gap the S into subsets that remember likely qualities for the most desirable characteristics for stage three.
Stage 4: Make the hub in the choice tree that has the best property.
Stage 5: Using the subsets of the dataset produced in Sync 3, recursively plan new choice trees. The last hub in this strategy is alluded to as a leaf hub when it arrives where it can't classify the hubs a lot further.



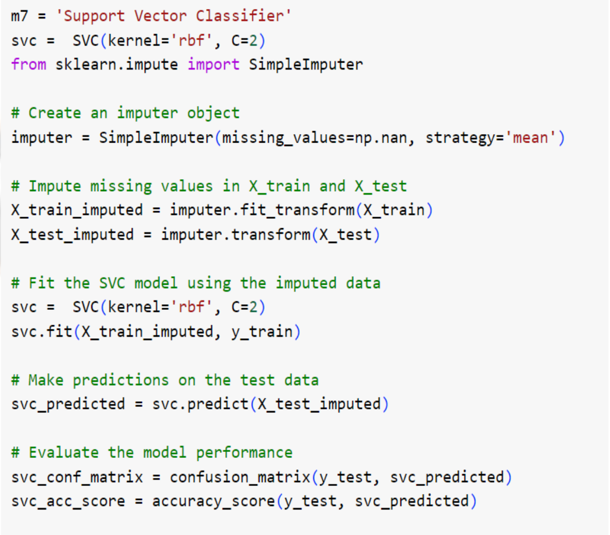
```
m6 = 'DecisionTreeClassifier'
dt = DecisionTreeClassifier(criterion = 'entropy',random_state=0,max_depth = 6)
from sklearn.impute import SimpleImputer

imputer = SimpleImputer(missing_values=np.nan, strategy='mean')
# Impute missing values in X_train
X_train_imputed = imputer.fit_transform(X_train)

# Fit the DecisionTreeClassifier model using the imputed data
dt = DecisionTreeClassifier(criterion = 'entropy',random_state=0,max_depth = 6)
dt.fit(X_train_imputed, y_train)
dt_predicted = dt.predict(X_test_imputed)
dt_conf_matrix = confusion_matrix(y_test, dt_predicted)
dt_acc_score = accuracy_score(y_test, dt_predicted)
```

Fig 15 : Decision Tree Classifier

h) Support Vector Classifier:

By defining a straight boundary between two classes, an essential direct SVM classifier works. At the end of the day, every data of interest on one side of the line will address a class, and every data of interest on the opposite side will be doled out to an unmistakable class. This infers that the quantity of lines to browse might be limitless. Since it chooses the ideal line for characterizing your data of interest, the direct SVM strategy performs better compared to a portion of different calculations, for example, k-closest neighbors. The line that partitions the information and is as distant from the nearest data of interest as plausible is the one it chooses. Utilize support vector machines (SVMs) is to distinguish complex relationship between your information without expecting you to play out a ton of manual changes. It is a magnificent decision for minuscule datasets with tens to countless qualities. Since they can deal with perplexing, small datasets, they as a rule track down additional exact discoveries than different calculations.
Steps Involve in SVC:
Step1: Get the dataset imported.
Step2: Investigate the information to determine their appearance.
Step3: Before processing the data
Step4: Divide the data into labels and characteristics.
Step5: Sort the data into sets for testing and training.
Step6: Put the SVM method to the test
Step7: Make a few educated guesses
Step8: Analyses the algorithm's outcomes.

It is important to avoid over-fitting when selecting the kernel functions and regularization term if there are many more features than data points. Estimates of probability are not directly provided by SVMs. An expensive five-fold cross-validation method is used to calculate those. its lengthy training time makes it work best with tiny sample sets.



```
m7 = 'Support Vector Classifier'
svc =  SVC(kernel='rbf', C=2)
from sklearn.impute import SimpleImputer

# Create an imputer object
imputer = SimpleImputer(missing_values=np.nan, strategy='mean')

# Impute missing values in X_train and X_test
X_train_imputed = imputer.fit_transform(X_train)
X_test_imputed = imputer.transform(X_test)

# Fit the SVC model using the imputed data
svc =  SVC(kernel='rbf', C=2)
svc.fit(X_train_imputed, y_train)

# Make predictions on the test data
svc_predicted = svc.predict(X_test_imputed)

# Evaluate the model performance
svc_conf_matrix = confusion_matrix(y_test, svc_predicted)
svc_acc_score = accuracy_score(y_test, svc_predicted)
```

Fig 16: Support Vector Classifier

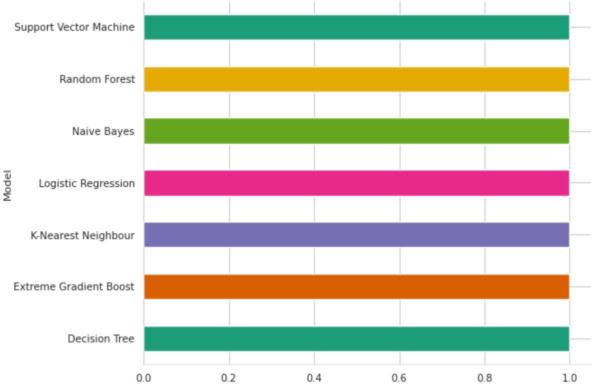
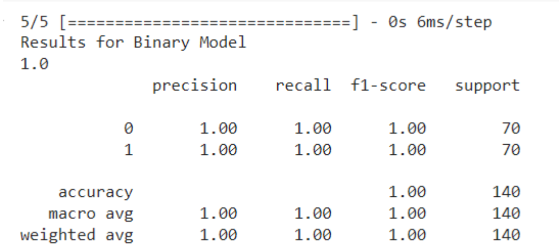


Fig 17: Categorical Distribution of Machine Learning Algorithm

IV. Results

Binary Classification Neural Network Prediction on Heart disease dataset.



```
5/5 [==============================] - 0s 6ms/step
Results for Binary Model
1.0
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        70
           1       1.00      1.00      1.00        70

    accuracy                           1.00       140
   macro avg       1.00      1.00      1.00       140
weighted avg       1.00      1.00      1.00       140
```

Fig 18: Result of Binary Classification using Neural Network

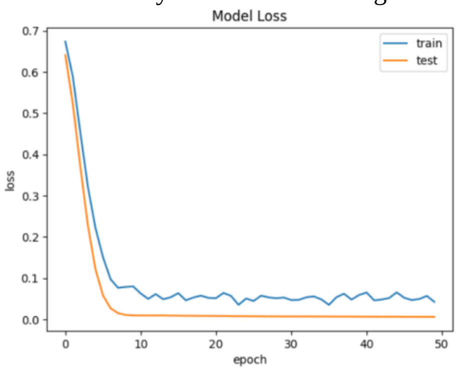


Fig 19: Neural Network Model Loss

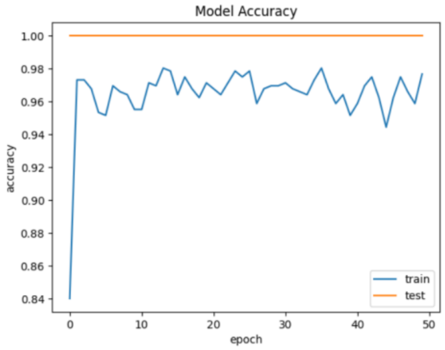
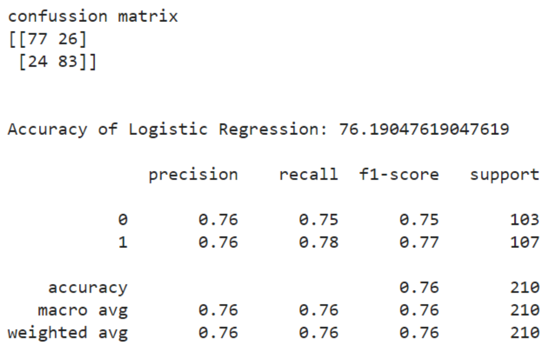
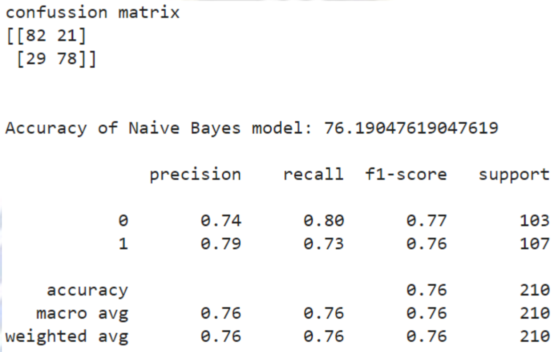


Fig 20: Accuracy of Binary Classification Model



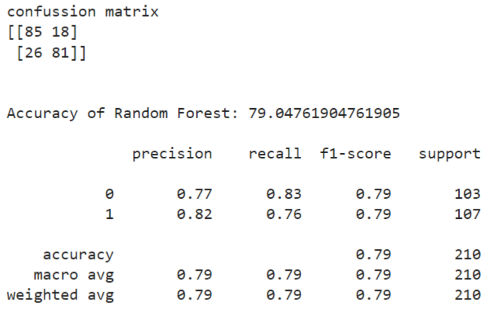
```
confussion matrix
[[77 26]
 [24 83]]


Accuracy of Logistic Regression: 76.19047619047619

              precision    recall  f1-score   support

           0       0.76      0.75      0.75       103
           1       0.76      0.78      0.77       107

    accuracy                           0.76       210
   macro avg       0.76      0.76      0.76       210
weighted avg       0.76      0.76      0.76       210
```

Fig 21: Accuracy of Logistic Regression Model on Heart disease dataset



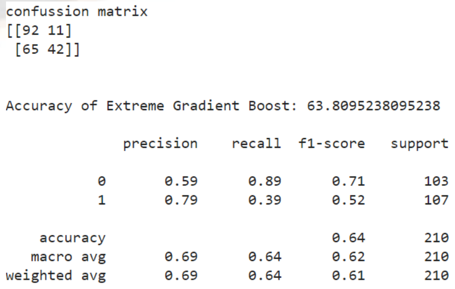
```
confussion matrix
[[82 21]
 [29 78]]


Accuracy of Naive Bayes model: 76.19047619047619

              precision    recall  f1-score   support

           0       0.74      0.80      0.77       103
           1       0.79      0.73      0.76       107

    accuracy                           0.76       210
   macro avg       0.76      0.76      0.76       210
weighted avg       0.76      0.76      0.76       210
```

Fig 22: Accuracy of Navie Bayes Model on Heart disease dataset



```
confussion matrix
[[85 18]
 [26 81]]


Accuracy of Random Forest: 79.04761904761905

              precision    recall  f1-score   support

           0       0.77      0.83      0.79       103
           1       0.82      0.76      0.79       107

    accuracy                           0.79       210
   macro avg       0.79      0.79      0.79       210
weighted avg       0.79      0.79      0.79       210
```

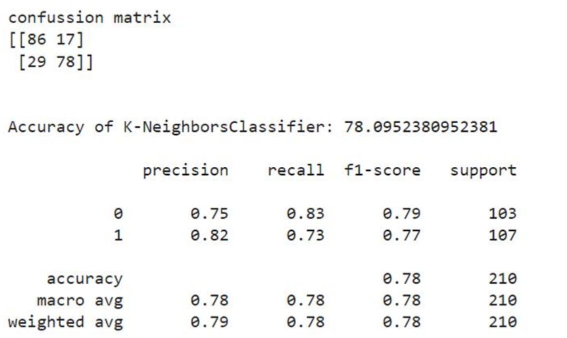
Fig 23: Accuracy of Random Forest Classifier Model on Heart disease dataset



```
confussion matrix
[[92 11]
 [65 42]]


Accuracy of Extreme Gradient Boost: 63.8095238095238

              precision    recall  f1-score   support

           0       0.59      0.89      0.71       103
           1       0.79      0.39      0.52       107

    accuracy                           0.64       210
   macro avg       0.69      0.64      0.62       210
weighted avg       0.69      0.64      0.61       210
```

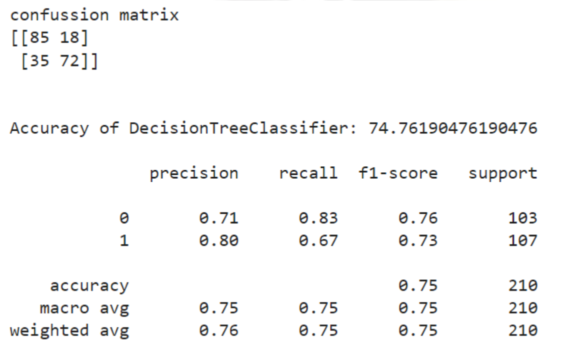
Fig 24: Accuracy of Extreme Gradient Boost on Heart disease dataset

***Article Received:** 25 November 2023 **Revised:** 12 December 2023 **Accepted:** 30 January 2024*



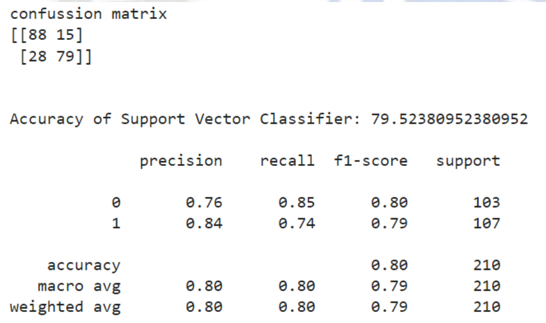
```
confussion matrix
[[86 17]
 [29 78]]


Accuracy of K-NeighborsClassifier: 78.0952380952381

              precision    recall  f1-score   support

           0       0.75      0.83      0.79       103
           1       0.82      0.73      0.77       107

    accuracy                           0.78       210
   macro avg       0.78      0.78      0.78       210
weighted avg       0.79      0.78      0.78       210
```

Fig 25: Accuracy of K-Neighbors Classifier on Heart disease dataset



```
confussion matrix
[[85 18]
 [35 72]]


Accuracy of DecisionTreeClassifier: 74.76190476190476

              precision    recall  f1-score   support

           0       0.71      0.83      0.76       103
           1       0.80      0.67      0.73       107

    accuracy                           0.75       210
   macro avg       0.75      0.75      0.75       210
weighted avg       0.76      0.75      0.75       210
```

Fig 26: Accuracy of Decision Tree Classifier on Heart disease dataset



```
confussion matrix
[[88 15]
 [28 79]]


Accuracy of Support Vector Classifier: 79.52380952380952

              precision    recall  f1-score   support

           0       0.76      0.85      0.80       103
           1       0.84      0.74      0.79       107

    accuracy                           0.80       210
   macro avg       0.80      0.80      0.79       210
weighted avg       0.80      0.80      0.79       210
```

Fig 27: Accuracy of Support Vector Classifier Heart disease dataset

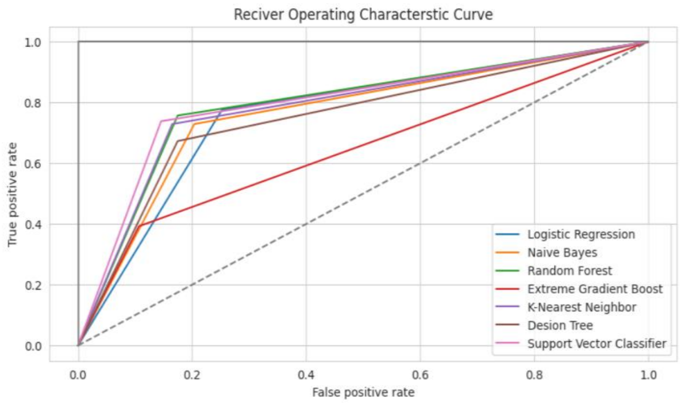
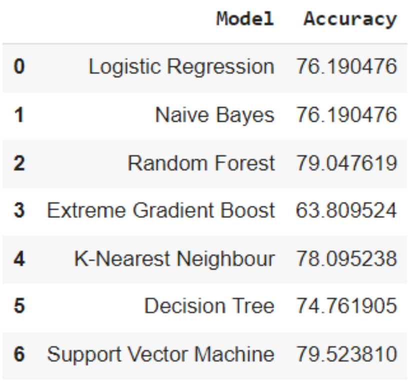


Fig 28: Combine Curve of Algorithms results.



	Model	Accuracy
0	Logistic Regression	76.190476
1	Naive Bayes	76.190476
2	Random Forest	79.047619
3	Extreme Gradient Boost	63.809524
4	K-Nearest Neighbour	78.095238
5	Decision Tree	74.761905
6	Support Vector Machine	79.523810

Fig 29: Combine table form results of Machine Learning algorithms on Heart Disease Dataset

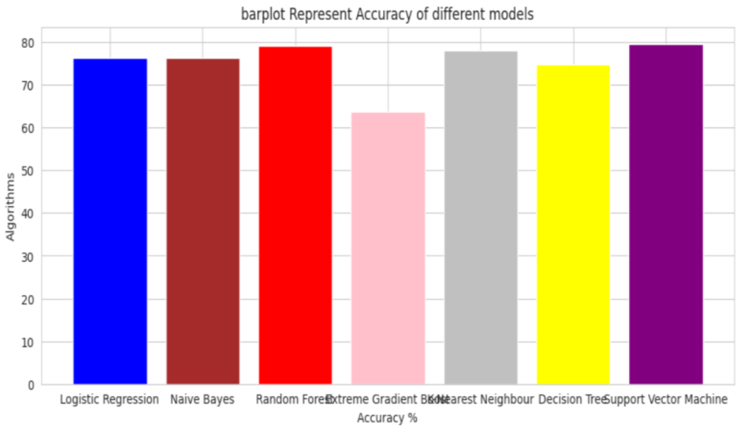


Fig 30: Accuracy of different models

V. Conclusion

Early identification of abnormalities in heart illnesses and long-haul life protection will be worked with by understanding how crude medical care information connected with heart data is handled. To decipher crude information and produce a new and imaginative finding of coronary illness, AI methods were applied in this review. In the clinical field, coronary illness forecast is critical and troublesome. If deterrent measures are carried out quickly and the illness is distinguished in its beginning phases, the demise rate can be essentially diminished. It would be ideal to extend this exploration significantly further, so genuine world datasets are utilized for the examinations as opposed to simply hypothetical models and reenactments. Various mixes of AI approaches can be utilized in this exploration's future headings to further develop forecast techniques. Furthermore, novel component choice methods can be made to get a more exhaustive comprehension of the significant perspectives and work on the precision of coronary illness forecast. A strategy for productively and precisely diagnosing and foreseeing coronary illness is profound learning. When contrasted with different techniques, the proposed model performed discernibly better regarding exactness, awareness, and explicitness. Later, desire to work on this technique by using patient picture information connected with coronary illness.

References

[1] M. Rana, M. Z. Ur Rehman and S. Jain, "Comparative Study of Supervised Machine Learning Methods for Prediction of Heart Disease," 2022 IEEE VLSI Device Circuit and System (VLSI DCS), Kolkata, India, 2022, pp. 295-299, doi: 10.1109/VLSIDCS53788.2022.9811495. keywords: {Heart;Support vector machines;Measurement;Machine learning algorithms;Medical services;Very large scale integration;Prediction algorithms;Machine Learning;Heart Disease;Exploratory Data Analysis;Machine Learning Algorithms},

[2] S. Chua, V. Sia and P. N. E. Nohuddin, "Comparing Machine Learning Models for Heart Disease Prediction," 2022 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), Kota Kinabalu, Malaysia, 2022, pp. 1-5, doi: 10.1109/IICAIET55139.2022.9936861. keywords: {Heart;Support vectormachines;Machine learning;Cardiac arrest;Medical services;Predictive models;Prognostics and health management;Clinical parameters;heart disease prediction;data mining;machine learning models},

[3] N. Goel, N. P. Yadav, P. Prakarti and A. Pandey, "Comparative Analysis of Single Classifier Models against Aggregated Fusion Models for Heart Disease Prediction," 2023 International Conference on Disruptive Technologies (ICDT), Greater Noida, India, 2023, pp. 576-580, doi: 10.1109/ICDT57929.2023.10150611. keywords: {Heart;Analytical models;Machine learning algorithms;Gaussian processes;Predictive models;Prediction algorithms;Partitioning algorithms;classifiers;machine learning;heart disease;prediction;accuracy;random forests;ensemble models;adaboost;neural networks support vector machines;decision trees;logistic regression;k-nearest neighbors},

[4] A. Bhowmick, K. D. Mahato, C. Azad and U. Kumar, "Heart Disease Prediction Using Different Machine Learning Algorithms," 2022 IEEE World Conference on Applied Intelligence and Computing (AIC), Sonbhadra, India, 2022, pp. 60-65, doi: 10.1109/AIC55036.2022.9848885. keywords: {Heart;Radio frequency;Machine learning algorithms;Medical services;Predictive models;Prediction algorithms;Classification algorithms;DT;RF;LR;Heart Disease prediction;Machine learning algorithms;Supervised learning},

[5] S. E. Freeda, N. S. T and T. C. E. Selvan, "Evaluating the Effective Machine Learning Techniques for Early Prediction of Heart Disease," 2023 7th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2023, pp. 703-708, doi: 10.1109/ICECA58529.2023.10394798. keywords: {Heart;Support vector machines;Machine learning algorithms;Prediction algorithms;Random forests;Medical diagnostic imaging;Diseases;Machine;learning;Heart disease prediction;Comparative analysis;Classifiers},

[6] R. A. Rashid, N. Binte Salam, S. Raisa, A. Noor and N. N. Choudhury, "Coronary Heart Disease Prediction On Small Datasets: A Comparative Analysis," 2023 26th International Conference on Computer and Information Technology (ICCIT), Cox's Bazar, Bangladesh, 2023, pp. 1-5, doi: 10.1109/ICCIT60459.2023.10441358. keywords: {Heart;Machine learning algorithms;Sociology;Prediction algorithms;Statistics;Random forests;Synthetic data;Coronary Heart Disease;Supervised Machine Learning;K-Nearest Neighbor;Decision Tree;Binary logistic Regression;Naive bayes;Random Forest;SMOTE;ADASYN},

[7] B. Anishfathima, R. Vikram, S. R. T, M. Sri Vishnu and C. Venumadhav, "A Comparative Analysis on Classification Models to predict Cardio-vascular disease using Machine Learning Algorithms," 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, 2022, pp. 259-264, doi: 10.1109/ICAIS53314.2022.9741831. keywords: {Support vector machines;Heart;Machine learning algorithms;Cardiac disease;Predictive models;Prediction algorithms;Classification algorithms;Heart disease;SVM;Random Forest;Machine Learning;Prediction},

[8] N. Susitha, H. Selvi, N. B. Mahesh Kumar, V. Nagarajan, S. S. Ali and L. S. Kumar, "Comparative Analysis of Machine Learning Techniques for Heart Disease Prediction," 2023 Third International Conference on Ubiquitous Computing and Intelligent Information Systems (ICUIS), Gobichettipalayam, India, 2023, pp. 496-500, doi: 10.1109/ICUIS60567.2023.00088. keywords: {Heart;Support vector machines;Pain;Cardiac disease;Prediction algorithms;Feature extraction;Bayes methods;Machine Learning;Heart Disease Prediction;AI;Naive Bayes;Support Vector Machine},

[9] A. Basak, M. S. Rahman and M. Rahman, "Prediction of Heart Disease Using an Approach Based on Machine Learning," 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2022, pp. 1-5, doi: 10.1109/ICCCNT54827.2022.9984555. keywords: {Heart;Support vector machines;Machine learning algorithms;Computational modeling;Cardiac disease;Forestry;Prediction algorithms;Heart Disease Prediction;Machine-Learning;SVM;KNN;Naïve Bayes;Random Forest},

[10] M. Das and G. Srivastava, "A Comparative Study of Supervised Learning Algorithms for Heart Disease Prediction," 2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, 2023, pp. 703-710, doi: 10.1109/ICTACS59847.2023.10390482. keywords: {Heart;Industries;Support vector machines;Machine learning algorithms;Supervised learning;Predictive models;Data models;Cardiovascular Diseases;Machine Learning;Supervised Learning;Random Forest;Logistic Regression;SVM},

[11] G. N. Ahmad, H. Fatima, S. Ullah, A. Salah Saidi and Imdadullah, "Efficient Medical Diagnosis of Human Heart Diseases Using Machine Learning Techniques With and Without GridSearchCV," in IEEE Access, vol. 10, pp. 80151-80173, 2022, doi: 10.1109/ACCESS.2022.3165792.keywords: {Machine learning;Heart;Diseases;Predictive models;Medical diagnostic imaging;Cardiac disease;Prediction algorithms;Heart disease;support vector machine (SVM);logistic regression (LR);gradient boosting classifier (GBC);GridSearchCV},

[12] H. Yang, Z. Chen, H. Yang and M. Tian, "Predicting Coronary Heart Disease Using an Improved LightGBM Model: Performance Analysis and Comparison," in IEEE Access, vol. 11, pp. 23366-23380, 2023, doi: 10.1109/ACCESS.2023.3253885.keywords: {Cardiology;Cardiovascular diseases;Machine learning;Predictive models;Performance evaluation;Diseases;Medical treatment;Optimization;Heart;Data models;Coronary heart disease;hyperparameter optimization;LightGBM;loss function;machine learning;OPTUNA},

[13] N. Sinha, M. A. G. Kumar, A. M. Joshi and L. R. Cenkeramaddi, "DASMcC: Data Augmented SMOTE Multi-Class Classifier for Prediction of Cardiovascular Diseases Using Time Series Features," in IEEE Access, vol. 11, pp. 117643-117655, 2023, doi: 10.1109/ACCESS.2023.3325705.keywords: {Electrocardiography;Heart;Feature extraction;Cardiovascular

diseases;Medical services;Solid modeling;Machine learning;Machine learning;Nearest neighbor methods;Cardiovascular disease (CVD);PTB-XL data;machine learning;smart healthcare;ECG;heart failure;XG boost (XGB);random forest (RF);cat boost;K nearest neighbor (KNN);gradient boost (GB)}

[14] G. N. Ahmad, S. Ullah, A. Algethami, H. Fatima and S. M. H. Akhter, "Comparative Study of Optimum Medical Diagnosis of Human Heart Disease Using Machine Learning Technique With and Without Sequential Feature Selection," in IEEE Access, vol. 10, pp. 23808-23828, 2022, doi: 10.1109/ACCESS.2022.3153047.keywords: {Diseases;Heart;Prediction algorithms;Medical diagnostic imaging;Machine learning;Random forests;Feature extraction;Heart disease;sequential feature selection;DT;RF;SVM;GBC;LDA;confusion matrix · ROC curve},

[15] K. Yongcharoenchaiyasit, S. Arwatchananukul, P. Temdee and R. Prasad, "Gradient Boosting Based Model for Elderly Heart Failure, Aortic Stenosis, and Dementia Classification," in IEEE Access, vol. 11, pp. 48677-48696, 2023, doi: 10.1109/ACCESS.2023.3276468. keywords: {Cardiovascular diseases;Dementia;Heart;Older adults;Machine learning;Medical diagnostic imaging;Parameter estimation;Aortic stenosis;cardiovascular disease;dementia;ensemble methods;feature engineering;heart failure;hyperparameter optimization;machine learning},

[16] B. Akkaya, E. Sener and C. Gursu, "A Comparative Study of Heart Disease Prediction Using Machine Learning Techniques," 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, 2022, pp. 1-8, doi: 10.1109/HORA55278.2022.9799978. keywords: {Heart;Support vector machines;Machine learning algorithms;Surveillance;Machine learning;Prediction algorithms;Classification algorithms;heart disease;machine learning;classification;outlier analysis},

[17] S. V, S. S and B. G, "Multi-Disease Prediction Using Machine Learning," 2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS), Chennai, India, 2023, pp. 1-8, doi: 10.1109/ICCEBS58601.2023.10449201. keywords: {Heart; COVID-19;Static VAr compensators;Predictive models;Diabetes;Diseases;Testing;Heart;Machine learning;Decision tree;Sigmoida SVC;AdaBoost;Diabetes},

[18] M. C. Das et al., "A comparative study of machine learning approaches for heart stroke prediction," 2023 International Conference on Smart Applications, Communications and Networking (SmartNets), Istanbul, Turkiye, 2023, pp. 1-6, doi: 10.1109/SmartNets58706.2023.10216049. keywords: {Heart;Support vector machines;Radio frequency;Measurement;Logistic regression;Machine learning algorithms;Stroke (medical condition);Heart Stroke;AdaBoost;Random Forest;Heart Stroke prediction;Machine learning techniques},

[19] S. Jamal, W. A. Elenin and L. Chen, "Developing and Evaluating Data-Driven Heart Disease Prediction Models by Ensemble Methods on Different Data Mining Tools," 2023 IEEE 14th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 2023, pp. 0678-0683, doi: 10.1109/UEMCON59035.2023.10315997. keywords: {Heart;Analytical models;Predictive models;User interfaces;Feature extraction;Boosting;Data models;Heart disease;ensemble technique;machine learning;WEKA;Orange;data mining},

[20] S. Rathi, A. Das, A. Gupta, J. Bagrecha, U. Mahajan and K. Patankar, "Exploring the Effectiveness of Various Machine Learning Models in Predicting Heart Failure: A Comparative Study," 2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 2024, pp. 1241-1249, doi: 10.1109/IDCIoT59759.2024.10467507. keywords: {Heart;Machine learning algorithms;Machine learning;Medical services;Predictive models;Programming;Prediction algorithms;Heart Disease;Machine Learning;Prediction;Unsupervised Machine Learning;Supervised Machine Learning;Random Forest;Decision Tree;Support Vector Machine;K Nearest Neighbors;Python Programming},

[21] J. Raval, J. P. Verma, S. N. M. Islam, R. Jain and N. Thakur, "AI Based Prediction for Heart Disease: A Comparative Analysis and an Improved Machine Learning Approach," 2022 6th Asian Conference on Artificial Intelligence Technology (ACAIT), Changzhou, China, 2022, pp. 1-9, doi: 10.1109/ACAIT56212.2022.10137923. keywords: {Heart;Support vector machines;Machine learning algorithms;Machine learning;Cardiac arrest;Prediction algorithms;Classification algorithms;Heart disease data set;Data mining;Machine learning;Predictive analysis;Multi-Layer Perceptron (MLP)},

[22] Nagavelli U, Samanta D, Chakraborty P. Machine Learning Technology-Based Heart Disease Detection Models. J Healthc Eng. 2022 Feb 27; 2022:7351061. doi: 10.1155/2022/7351061. PMID: 35265303; PMCID: PMC8898839.