GMR
The Original

EVALUATION OF DECISION TREES, NAÏVE BAYES, AND SUPPORT VECTOR MACHINES IN FORECASTING CARDIOVASCULAR DISEASE AND FEATURE SELECTION USING THE CHI-SQUARE (X^2) TEST IN TWO DIFFERENT DATASETS

Banibrata Paul[1]*, Bhaskar Karn[2]

[1]Research Scholar, Department of Computer Science and Engineering, Birla Institute of Technology, Mesra, Ranchi, Jharkhand, India, paul.banibraata1@gmail.com
[2]Associate Professor, Department of Management, Birla Institute of Technology, Mesra, Ranchi, Jharkhand, India, bhaskar@bitmesra.ac.in
*Corresponding Author**:** Banibrata Paul, paul.banibraata1@gmail.com

ABSTRACT
Purpose: Heart disease is a major global concern. In all its manifestations, heart disease is one of the world's leading causes of death. Affordable treatment and early detection are essential for prevention. The correct diagnosis procedure is crucial for every patient. The death ratio is decreased with early detection and prognosis of heart disease. A number of data mining techniques, including Naive Bayes, decision trees, and support vector machines, helps to detect cardiovascular diseases early and accurately.
Methods: Certain data mining approaches, such as Naive Bayes, decision trees, and support vector machines, aid in the accurate prediction of cardiovascular illnesses using two different datasets. Similar to the 'Heart Statlog Cleveland Hungary Final' heart dataset, which has 1190 instances (rows), 11 input characteristics, and 1 output attribute, the 'Heart Failure Clinical Records' cardiac dataset has 299 instances (rows), 12 input attributes, and 1 output attribute. Both datasets are taken from UCI machine learning repository.
Results: Using Decision Tree, Naïve Bayes, and SVM classifiers, respectively, the 'Heart Failure Clinical Records' heart dataset yielded 92%, 81.67%, and 82.67% prediction accuracy. Similarly, 'Heart Statlog Cleveland Hungary Final' heart dataset —achieved 85.23%, 81.88%, and 81.879% prediction accuracy. Finally, the longest paths of all attributes in these two datasets are determined using decision trees.
Conclusion: After determining the effectiveness of three data mining algorithms—Decision Tree, Naïve Bayes, and Support Vector Machine—for two heart datasets, the chi-square test is used to select features in descending order that are strongly associated with the presence of heart disease.

KEYWORDS- Cardio vascular disease, Chi-Square (χ^2) test, data mining, decision tree (C4.5), heart disease prediction system, Naïve Bayes, support vector machine.

1. INTRODUCTION

The human heart is an important and significant organ. It is the key component of the circulatory system.
The fluid aids in the transportation of oxygen, which is required for healthy physiological function. Annually, the heart pumps enough blood to fill a modern tanker.
Heart disease (HFD) is the leading cause of death worldwide, accounting for 17.9 million deaths annually. CVDs include rheumatic heart disease, coronary heart disease, and cerebrovascular illness. Heart attacks and strokes account for more than 40% of CVD deaths, with one-third of these deaths occurring before the age of 70. Tobacco use, dangerous alcohol consumption, a poor diet, and inactivity are the most obvious risk factors for heart disease and stroke. These risk factors can lead to high blood pressure, diabetes, high cholesterol, and being overweight or obese. These 'intermediate risk factors' can be assessed in primary care settings to determine whether there is an increased risk of heart attack, stroke, heart failure, or other consequences. Cardiovascular diseases (CVDs) include the following types:
Coronary heart disease: fat deposits accumulate in the coronary arteries, obstructing or interrupting the patient's heart's blood flow. Cerebrovascular disease is a collection of disorders affecting cerebral blood vessels and blood flow.
Peripheral arterial disease: A disorder known as atherosclerosis that is brought on by an accumulation of lipids, cholesterol, and other materials in and on the arterial walls. Rheumatic heart disease is a condition where rheumatic fever has permanently affected the heart valves. Congenital heart disease refers to structural cardiac abnormalities that are present at birth.
Pulmonary embolism and deep vein thrombosis: A pulmonary embolism (PE) occurs when a blood clot lodges in a lung artery, blocking some of the lung's blood supply. Blood clots often form in the legs, travel to the right side of the heart, and enter the lungs. This condition is known as deep vein thrombosis.

It has been shown that limiting salt intake, eating more fruits and vegetables, engaging in regular physical activity, quitting smoking, and abstaining from alcohol all reduce the risk of cardiovascular disease. To encourage people to adopt and sustain healthy behaviors, health policies must provide conditions that make healthy options inexpensive and accessible. Various data mining approaches, including decision trees, support vector machines, and Naïve Bayes, can predict cardiac disease.

Data Mining

Data mining is an innovative technique for preparing large datasets. It is used for computing and design discovery in large-scale datasets. It is a valuable system for extracting crucial information and unique examples from enormous datasets. The primary purpose of data mining is to extract relevant data from large data sets for subsequent use.

Decision Tree

A decision tree is useful in both regression and classification applications. It is nonparametric and can be used for both regression and classification applications. It provides a solid foundation for more complex algorithms, such as Random Forest (RF), and is easily understandable. It classifies the examples by recursively splitting the training data into subgroups based on attribute values until a stopping condition is met. In addition to managing continuous and categorical data, Decision Tree can do categorization with minimal computing overhead. They can also create explicit rules. This technique consists of two processes: creating a tree and then applying it to the dataset. Some well-known decision tree algorithms are CART, J48, C4.5, CHAID, and ID3. This system employs the C4.5 algorithm derived from these.

Support Vector Machine

Support vector machines (SVMs) were first introduced by V. Vapnik in his work on statistical learning theory. SVM is a supervised machine learning technique that can be used to solve both classification and regression issues. The SVM technique seeks to locate a hyperplane in an N-dimensional space that clearly categorizes data points. Because it can convert low-dimensional input space into higher-dimensional space, the SVM kernel can be used to solve nonlinear separation problems. Support vectors, or extreme points or vectors that help create the hyperplane, are chosen using SVM. SVM builds on the concept of structural risk minimization by combining the ability to tackle the overfitting problem.

It is used to divide data into two binary categories: the existence and absence of cardiac disease, with $y_i = +1$ and -1, respectively. Multiple two-class classifiers can be created to directly extend the technique for multiclass classification. SVM classifiers look for the optimal extraction hyperplane between two classes. This hyperactive, optimum unscrambling plane has various favourable statistical features.

Naïve-Bayes

The Naïve Bayes classifier is based on Bayes' theorem. This classifier technique uses the conditional independence assumption, which stipulates that an attribute value in a particular class is independent of the values of other attributes. The Bayes theorem means the following:

Assume that $X= \{x_1, x_2 \ldots . x_n\}$ is a collection of n qualities. In a Bayesian model, X is regarded as evidence, and H is a hypothesis that indicates X's data falls into a particular class C. The probability that the hypothesis H holds in light of the evidence, or data sample X, is P (H|X).The Bayes theorem states that $P(H|X) = \frac{P(X|H)P(H)}{P(X)}$

Bayesian networks are some of the most effective operative classifiers available. These networks are composed of network-like structures with identical conditional probability.

A Bayesian network is architecturally designed as a directed acyclic graph with nodes representing province variables and edges expressing variable dependencies. Naive Bayes classifiers are now strategies for determining which classification is best for a dataset when certain fundamental rules are met.

2. Background Study

Comparative analysis of several researcher's work through a literature survey are explained in table1

SL.NO	Procedure	Key Findings	Citation
1.	Attention-based hybrid deep learning	Mean improvement of 36.11%, 5.37%, and 1.04% over Random Forest, Unidirectional LSTM, Bi directional LSTM and best attention based unidirectional HDL model	Bhagawati M et al. [2025] [1]
2.	Explainable Boosting Machine (EBM)	AUROC: 0.785; AUPRC: 0.284	Climente-González H et al. [2025] [2]
3.	Biomarkers + traditional features using CatBoost + SHAP	C-index: 0.822 (CVD death)	Zhang X.R et al. [2025] [3]
4.	Framingham Risk Score validation using Regression	C-statistic: 0.765; AUC: 0.82	Amiri M et al. [2025] [4]

5.	XGBoost	AUC: 0.94 internal, 0.78 external	Liu L et al. [2025] [5]
6.	PRS + Composite ML	Continuous risk prediction vs binary	Zhang Y et al. [2025] [6]
7.	ECG + Deep Transfer Learning using DTL + SVM/XG Boost	Accuracy: 96.31%	Panigrahi A et al. [2025] [7]
8.	Wavelet Convolution Transformer	Accuracy: 95%; Precision: 0.93	Wang S et al. [2025] [8]
9.	MR cine series for PH prediction using CNN	Pearson: 0.80; R^2: 0.64	Cheng L.H et al. [2025] [9]
10.	IoT + DL for heart disease using Opt GPDCNN	Accuracy: 99.56%; F1: 99.20%; Sensitivity: 99.17%	Gorapalli Srinivasa Rao et al. [2026] [10]
11	Several machine learning methods exist, including multilayer perceptron, XGBoost, random forest, and decision tree.	87.28%	C. M. Bhatt et al. [2023] [11]
12	Used six algorithms: Ada-Boost classifier, Gradient Boosting, Random Forest, K-Nearest Neighbour, Logistic Regression, and Naïve Bayes.	93.44% (for the Cleveland dataset) and 95% (for the IEEE dataset).	N. Chandrasekhar et al. [2023] [12]
13	The best machine learning algorithms include Decision Tree, KNN, Random Forest, and SVM.	95.4% (maximum accuracy using Random Forest).	M. A. Kadhim et al. [2023] [13]
14	Numerous machine learning techniques exist, such as KNN, Decision Tree, Random Forest, SVM, ANN, and Logistic Regression.	86.89% (best accuracy for Random Forest).	S.S. Dehia et al. [2023] [14]
15	Deep Neural Network and Convolutional Neural Network.	84% to 99% (best accuracy using a deep neural network).	W. A. W. Abu Bakar et al. [2023] [15]
16	Random Forest, Naïve Bayes, Deep Learning Model, and Logistic Regression.	73.78% (maximum accuracy by using the deep learning model).	T.S. Eswar Reddy et al. [2022] [16]
17	Various machine learning classifiers, including SVM, NB, XGBoost, KNN, MLP, and CatBoost.	88.67% to 98.11%, with a mean accuracy of 94.34%.	K. Kanagarathinam et al. [2022] [17]
18	A variety of supervised machine learning techniques, such as MLP, AdaboostM1 (ABM1), RF, DT, KNN, and LR.	Highest accuracy of 100% (using KNN, RF, and DT).	Md. Mamun Ali et al. [2021] [18]
19	Some methods for classifying data from data mining include NB, SVM, KNN, DT, NN, LR, RF, and gradient boosting.	92.85% (highest accuracy using the Random Forest algorithm combined with PCA).	F. Tasnim et al. [2021] [19]
20	Several machine learning methods include KNN, SVM, NB, DT, RF, and ANN.	93.40% (maximum accuracy by using logistic regression).	R. Katarya et al. [2020] [20]
21	A probabilistic clustering technique using the Feature Ranking Voting (FRV) algorithm.	95% to 96%	M.A. Hogo [2020] [21]
22	Supervised machine learning algorithms such as SVM, KNN, and Naïve Bayes using the R programming language.	86.6% (best accuracy by using the Naïve Bayes algorithm).	S. Anitha et al. [2019] [22]
23	Different data mining techniques exist, such as decision trees, random forests, logistic regression, SVM, MLP classification, and Naïve Bayes.	86% (highest accuracy by using logistic regression).	C.S. Wu et al. [2019] [23]
24	A hybridization technique is a combination of Naïve Bayes, SVM, KNN, ANN, Tree (J48), and GA classification algorithms.	89.2%	M. Tarawnch et al. [2019] [24]
25	Decision tree (J48) algorithm.	68%	M. K. Iliyas et al. [2019] [25]
26	Machine learning classification model.	85%	A.K. Dwivedi [2018] [26]
27	Data mining classification methods include NB, SVM, KNN, DT, LR, and Vote (a hybrid method combining Logistic Regression and Naïve Bayes).	87.4% (best accuracy using data mining techniques).	M.S. Amin et al. [2018] [27]

28	A decision tree-based fuzzy medical diagnostic aid system.	63.24 %	O. Terrada et al. [2018] [28]
29	Naïve Bayes and Decision Tree algorithms.	99% (maximum accuracy).	A.S. Karthiga et al. [2017] [29]
30	Using the WEKA program, KStar, J48, SMO (Sequential Minimum Optimization), Bayes-Net, and Multilayer Perceptron are accessible.	89% (maximum accuracy using the SMO algorithm).	M. Sultana et al. [2016] [30]
31	Data mining classification methods include KNN, neural networks, decision tree algorithms, and Naïve Bayes.	80.6% (maximum accuracy).	T. Princy, R. et al. [2016] [31]
32	Random Forest algorithm, J48 method, and Logistic Model Tree algorithm.	83.33% (J48 algorithm), 80% (Logistic Model Tree algorithm), and 86.20% (Random Forest algorithm).	J. Patel et al. [2015] [32]
33	Decision Tree (C 5.0) algorithm.	85.33%	M. Abdar [2015] [33]
34	Particular emphasis is placed on DT, NN, and Naïve Bayes data mining classification modelling techniques.	89% (Decision Tree) 86.53% (Neural Network) 85.53% (Naïve Bayes)	A. Methaila et al. [2014] [34]
35	There are three classification methods: DT, NN, and NB.	94.44% (Naïve Bayes), 96.66% (Decision Tree) and 99.25% (Neural Network) for 13 attributes and obtained 90.74% (Naïve Bayes), 99.62% (Decision Tree), and 100% (Neural Network) for 15 attributes.	K. Thenmozhi et al. [2014] [35]
36	Data mining techniques include Decision Tree, ID3, and CART (Classification and Regression Tree).	83.49% (CART), 72.93% (ID3), and 82.50% (Decision Tree).	V. Chaurasia et al. [2013] [36]
37	Combination of genetic algorithms with KNN.	100%	M.A. Jabbar et al. [2013] [37]
38	Supervised machine learning classification methods, including neural networks, decision trees, and Bayesian classifiers.	95.41% (J48 unpruned with all attributes) 95.56% (J48 unpruned with selected attributes).	A. Taneja [2013] [38]
39	Subtractive clustering methods.	76.67%	L. Muflikhah et al. [2013] [39]
40	Three data mining classification approaches are available: NN, DT, and NB. Precision is used to assess the effectiveness of different methods.	99.62% (Neural Networks), 100% (Decision Trees), and Naïve Bayes (90.74%).	C.S. Dangare et al. [2012] [40]
41	Utilizing data mining methods like Naïve Bayes, Decision Trees, and clustering-based classification.	96.5% (Naïve Bayes), 99.2% (Decision Tree), and 88.3% (classification via clustering).	J. Soni et al. [2011] [41]
42	Hybrid techniques include Nine Voting Equal Frequency, Discretization Gain Ratio, and Decision Tree.	84.1%	M. Shouman et al. [2011] [42]
43	Decision trees, neural networks, and the Naïve Bayes classifier are examples of data mining approaches that yield.	86.12% (Naïve Bayes Classifier), 85.68% (Neural Network), and 80.4% (decision tree).	S. Palaniappan et al. [2008] [43]

Table 1: Comparative analysis of different techniques through a literature survey

3. METHODS

Fig. 1 illustrates the overall flow chart of the proposed hybrid model using decision tree, SVM, and Naïve Bayes classifier

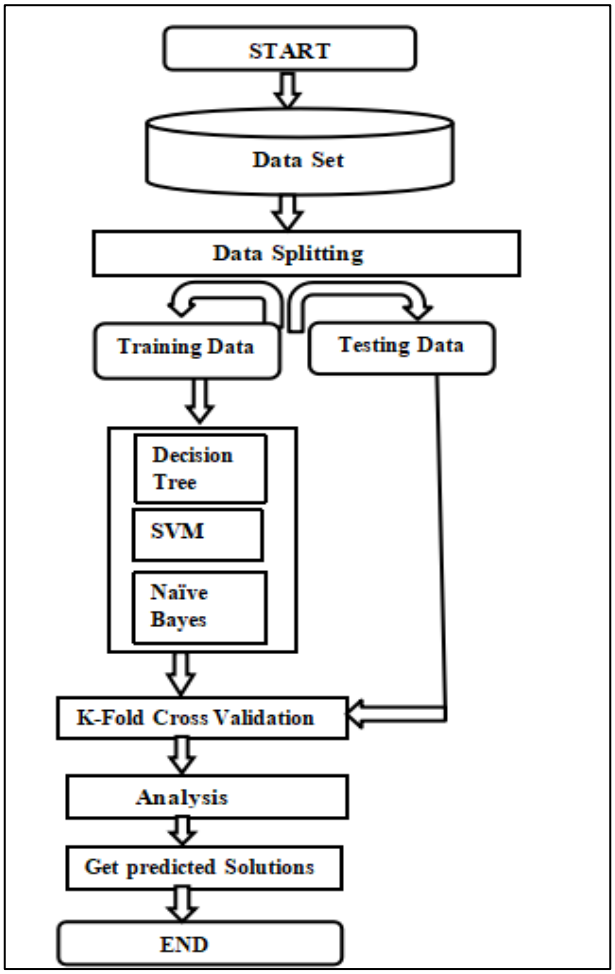

Fig.1: Detailed flow chart of the proposed hybrid model using Decision Tree, SVM, and Naïve Bayes Classifier

3.1 Model Description

3.1.1 DATA MINING

'Data mining' refers to the technique of using data analysis to predict the future and contextualize the past. Data mining combines database technology, machine learning, artificial intelligence, and statistics. Data mining applications are often highly valued. The basic goal of the data mining process is to extract hidden information from a vast dataset and convert it into useful information for later use. Data mining has four basic forms of relationships: classification, grouping, associations, and sequential patterns.

The next sections discuss various layers of data mining algorithms, such as SVM, decision trees, and the Nave Bayes classification approach.

3.1.2 Decision Tree

The decision tree is a common classifier that is straightforward and simple to use. It can handle high-dimensional data without the requirement for parameter configuration or domain knowledge. It generates findings that are easier to read and understand. Decision trees are the only ones that allow you to drill down and view detailed patient profiles. When solving classification issues, the decision tree method outperforms other methods. It is a graph displaying all possible outcomes of a branching decision. This technique consists of two processes: creating a tree and then applying it to the dataset. Some well-known decision tree algorithms are CART, J48, C4.5, CHAID, and ID3.

Decision trees are used in the construction of regression or classification models.

The approach gradually builds a decision tree for each data set by dividing it into smaller and smaller portions. A tree is

the end result of decision nodes and leaf nodes together.

A decision node can contain multiple branches. Each leaf node represents a judgment or classification. The root node is the highest decision node in a tree, and it represents the most powerful predictor. A decision tree may process both numerical and categorical input.

In this study, the C4.5 algorithm is used to create a decision tree.

3.1.2.1 C4.5 Algorithm:

C4.5 was created by Quinlan Ross. It extends Quinlan's prior ID3 algorithm. C4.5 is known as a statistical classifier since it generates decision trees that can be used for categorization.

3.1.2.2 Entropy:

In a set of examples, entropy is used to measure impurity, disorder, or uncertainty. It determines how decisions set boundaries. Entropy = − Σp(X) log p(X).

3.1.2.3 Information Gain

The decrease in entropy caused by dividing a dataset by an attribute is the source of information gain. The key to creating a decision tree is identifying the property that provides the most information gain.
Step I: Calculate the target's entropy.
Step II: For each branch, an entropy is calculated. It is then proportionally added to obtain the split's overall entropy. In other words, the resulting entropy reduces the pre-split entropy. Either more information is obtained or entropy is reduced. Next, the dataset is separated into groups based on the various attributes.
Step III: Among all the qualities, select the decision node with the greatest information gain.
Step IV(a): A leaf node is a branch with zero entropy.
Step IV (b): A branch with an entropy larger than zero requires additional splitting.
Step V: Iteratively apply the C4.5 algorithm on the non-leaf branches until all data is classified.

3.1.3. Support Vector Machine

It illustrates a supervised learning algorithm. Regression analysis and classification both use SVM.
Near the decision border, data points are referred to as support vectors; all data points are represented as vectors.
The hyperplane in SVM divides the feature space into two groups. It addresses the linear separability issue.
In SVM, feature spaces are classified into two types: convex, closed, and non-overlapping.
Time Complexity of SVMs
O(n) represents the linear support vector.
Non-linear SVM takes $O(n^2)$ to $O(n^3)$, where n is the number of training samples.

3.1.4. Classification using Naïve Bayes Classifier

The Bayes theorem serves as the foundation for the Naive Bayes classifier. The classifier is statistical in nature and assumes no correlation between its attributes. Because this strategy relies on conditional independence, it assumes that the value of an attribute inside a specific class is independent of the values of other attributes. Large datasets benefit immensely from the simplicity of a Naive Bayesian model, which reduces the need for sophisticated iterative parameter estimates. The Naive Bayesian classifier is popular because it can outperform more complex classification approaches and often works well despite its simplicity.

3.1.4.1 Naïve Bayes Classification Algorithm

The posterior probability, P (c|x), can be calculated using the Bayes theorem by combining P (c), P (x), and P (x|c). The naive Bayes classifier operates on the assumption that the values of other predictors have no effect on a predictor's (x) influence on a given class (c). This assumption is known as class conditional independence.
As a result, the target class's conditional probability is

$$P(c|x) = \frac{P(X|C)P(c)}{P(x)} \quad (1)$$

$$= P(x1|c) \times P(x2|c) \times P(x3|c) \ldots\ldots\ldots x\, P(xn|c) \times P(c) \quad (2)$$

P (c|x) is the posterior probability of a class (target) based on a predictor.
The prior probability for the class is P(c).
P (x|c) is a measure of the predictor's likelihood, or probability, given the class.
P(x) denotes the predictor's prior probability.

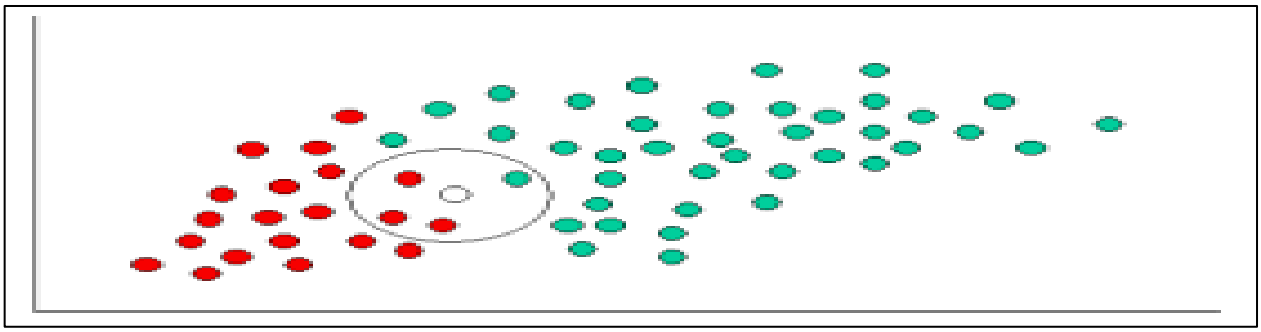

Fig 2: 'RED' and 'GREEN' objects for classification [29]

For each of the 60 items, the prior probability of class membership is as follows: 20 are 'RED' and 40 are 'GREEN'. As a result, we can write:

$$\text{Prior probability for 'GREEN'} = \frac{Number\ of\ GREEN\ objects}{Total\ number\ of\ objects} = \frac{40}{60} \quad (3)$$

$$\text{Prior probability for 'RED'} = \frac{Number\ of\ RED\ objects}{Total\ number\ of\ objects} = \frac{20}{60} \quad (4)$$

We can now classify a new object (the 'WHITE' circle) after determining our prior probability from Fig. 2. It makes reasonable to assume that the more "GREEN" (or 'RED') items there are close to X, the more likely it is that additional cases will belong to a particular color given how tightly the objects are packed. To measure this likelihood, we construct a circle around X that contains a number of predetermined points, independent of their class labels. We then calculate the number of points that each class label has in the circle. Based on this, we calculate the likelihood:

$$\text{Likelihood of X given 'GREEN'} = \frac{Number\ of\ GREEN\ objects\ in\ the\ neighbourhood\ of\ X}{Total\ number\ of\ \ GREEN\ \ objects} \quad (5)$$

$$\text{Likelihood of X given 'RED'} = \frac{Number\ of\ RED\ objects\ in\ the\ neighbourhood\ of\ X}{Total\ number\ of\ \ RED\ \ objects} \quad (6)$$

It is clear that the probability of X given 'GREEN' is lower than the probability of X given 'RED' because the circle in the accompanying picture contains one 'GREEN' object and three 'RED' ones. Accordingly:

$$\text{Likelihood of X given 'GREEN'} = \frac{1}{40} \quad (7)$$

$$\text{Likelihood of X given 'RED'} = \frac{3}{20} \quad (8)$$

The prior probabilities suggest that X might belong to 'GREEN' because there are twice as many 'GREEN' items as 'RED,' but the probability indicates that X belongs to 'RED' because there are more 'RED' things close than 'GREEN.' Named for Rev. Thomas Bayes (1702–1761), the Bayesian analysis uses both the prior and the likelihood to obtain a final classification. The so-called Bayes' rule is then used to convert this final categorization into a posterior probability.

Posterior probability of X being 'GREEN' = Prior probability for 'GREEN' x Likelihood of X given 'GREEN'

$$= \frac{40}{60} \text{ x } \frac{1}{40} = \frac{1}{60} \quad (9)$$

Posterior probability of X being 'RED' = Prior probability for 'RED' x Likelihood of X given 'RED'

$$= \frac{20}{60} \text{ x } \frac{3}{20} = \frac{3}{60} \quad (10)$$

We ultimately classify X as 'RED' since its class membership produces the highest posterior probability. Numerous density functions, including gamma, normal, and Poisson, can be used to characterize Naive Bayes.

3.1.5 K-Fold Cross Validation

The cross-validation procedure is used to evaluate the model after it has been trained on a complementary subset of the data set. In this case, the initial data sets are arbitrarily divided into K folds (D_1, D_2, D_3..D_K), all of which are substantially identical in size and mutually exclusive. There are K iterations of testing and training.

D_2, D_3, D_4, and D_K are used as the training sets for the first iteration, whereas D_1 is designated as the test set.

The training sets are D_1, D_3, D_4, and D_K, whereas the test set is the second repetition, D_2. D_i serves as the test set in this iteration, whereas the remaining D_1, D_2,.$D_{(i+1)}$, $D_{(i-1)}$, and D_K serve as training sets. The number of times each sample is utilized for testing and training is the same. This is how accuracy is calculated for a classification problem: The ratio of successfully classified entries from K iterations to all tuples in the initial data.

4. EXPERIMENTATION (SOLUTION METHODS)

4.1 Data Set

The current study uses the cardiac datasets 'Heart Failure Clinical Records' [46] and 'Heart Statlog Cleveland Hungary Final' [45] to determine whether or not cardiac illness stages exist. There are 299 instances (rows), 12 input attributes, and 1 output attribute in the cardiac dataset for 'Heart Failure Clinical Records'. The input attributes include age, anemia, diabetes, creatinine phosphokinase, ejection fraction, high blood pressure, platelets, serum creatinine, serum sodium, gender, smoking, and bouts of heart failure. Two strengths, 0 and 1, are anticipated in the output. The numbers '1' and '0' stand for 'heart attack' and 'healthy,' respectively. Similar methods are applied to a cardiac dataset system called 'Heart Statlog Cleveland Hungary Final,' which has 1190 instances (rows), 11 input attributes, and 1 output field named 'Goal.' The input features include age, gender, blood pressure, blood sugar, cholesterol, heart rate, ECG, exercise-induced angina, old peak, and ST-slope. The output should have two strengths, 0 and 1, where 0 represents 'normal' and 1 represents 'heart disease.' Two datasets from Tables 2 and 3 are described as follows.

Instances: 299 | Attributes: 12 inputs + 1 output		
Attribute	Range / Categories	Description
Age	[40, 95]	Patient age
Anaemia	[0,1] → 0 = No, 1 = Yes	Anaemia condition
High blood pressure	[0,1]	Hypertension
Creatinine phosphokinase	[23, 7861]	CPK enzyme level
Diabetes	[0,1]	Diabetes presence
Ejection fraction	[14, 80]	Heart pumping efficiency (%)
Platelets	[25,000–850,000]	Platelet count
Serum creatinine	[0.5, 9.4]	Creatinine level
Serum sodium	[113, 148]	Sodium level
Gender	[0,1] → 0 = Female, 1 = Male	Gender

Smoking	[0,1]	0 = absence of habit; 1 = habit of smoking.
Time	[4, 285]	Follow-up period (days)
Predicted Output	[0,1] → 0 = Healthy, 1 = Heart Attack	Target variable

Table 2: 'Heart Failure Clinical Records' cardiac dataset [46]

Instances: 1190 | Attributes: 11 inputs + 1 output		
Attribute	Range / Categories	Description
Age	[28, 77]	Patient age
Gender	[0,1] → 0 = Female, 1 = Male	Gender
Chest Pain Type	[1,2,3,4] → 1 = Typical Angina, 2 = Atypical Angina, 3 = Non-anginal, 4 = Asymptomatic	Chest pain type
Blood Pressure	[85, 200]	Resting systolic BP (mmHg)
Cholesterol	[126, 564] [100, 564]	Serum cholesterol (mg/dl)
Blood Sugar	[0,1] → 0 = ≤120 mg/dl, 1 = >120 mg/dl	FBS
ECG	[0,1,2] → 0 = Normal, 1 = ST-T abnormality, 2 = LVH	Resting ECG results
Max Heart Rate	[71, 202]	Maximum heart rate achieved
Exercise Angina	[0,1] → 0 = No, 1 = Yes	Exercise-induced angina
Old-peak	[0.0, 6.2]	ST depression (exercise vs. rest)
ST-Slope	[1,2,3] → 1 = Upsloping, 2 = Flat, 3 = Downsloping	Slope of ST segment
Predicted Output	[0,1] → 0 = No Heart Disease, 1 = Heart Disease	Target variable

Table 3: 'Heart Statlog Cleveland Hungary Final' cardiac dataset [45]

4.2 Decision Tree

The decision tree method is more suited for solving classification problems. This approach involves two steps: first, a tree is constructed, and then the dataset is subjected to the tree. Among the popular decision tree algorithms are C4.5, CART, CHAID, ID3, and J48. For this system, the C4.5 algorithm is used. C4.5 is the latest version of the ID3 induction algorithm. It is better than the ID3 algorithm. An ID3-like decision tree is produced as a result. Using the idea of information entropy, a decision tree is constructed from the training dataset. Consequently, the term 'statistical classifier' (C4.5) is frequently employed. C4.5 is a popular free data mining program. It looks like a flowchart with a tree structure. In this case, each internal node represents an attribute test, and the topmost node represents the root node. Each branch represents a test outcome, and each leaf node, also known as a terminal node, has a class level. In this instance, 'rectangles' stand in for internal nodes and 'ovals' for leaf nodes.

How Decision Trees are used for Classification

Decision trees can be used to handle multidimensional data. It is possible to construct decision tree classifiers without the need for subject knowledge or parameter sets. Let 'X' stand for the given tuple, whose corresponding class level is unknown. The attribute values of the tuples are then compared using the decision tree. Ultimately, a path that connects the root node to the leaf node containing the class prediction for the tuple is discovered.

A Decision Tree Example Determine whether or not the integer S = 56 in this classifier belongs to Class B. Fig. 3 explains it.

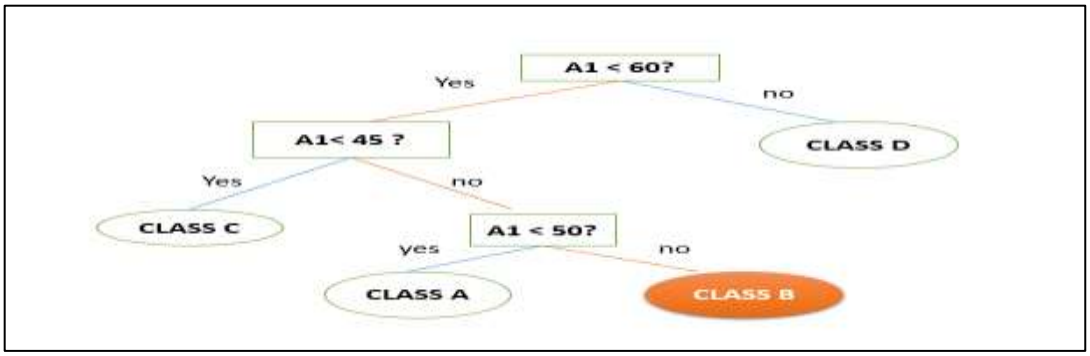

Fig. 3: Clarification of the Decision Tree

C4.5 builds a decision tree using information gain and entropy.

ENTROPY

The information content (I) of a single event or message (e) is defined as $I = -\log_2 p$ where p is the probability of occurrence. $I \geq 0$, as $0 \leq p < 1$.

Simultaneously, it causes or raises doubts about the same quantity of information being transmitted to others.

The term 'entropy' (often represented by the symbols H or E) refers to the degree of doubt, uncertainty, or impurity regarding e. In the real world, gaining information (I) results in the loss of uncertainty (H) by the same amount; therefore, I and H differ only in sign, i.e., I = -H. Thus, the (-) 've' sign denotes merely a loss of doubt in this case.

Starting with a degree of uncertainty (i.e., randomness) in a data set (which represents a message), the other (i.e., the second party; in this case, the learner, C4.5) tries to eliminate the uncertainty from that set by getting data by splitting the existing data set.

The expected or average information of a variable (X) having outcomes {x1, x2,..., xn} and probabilities, say p1, p2,..., pn, is readily described as:

$E(I(X)) = \sum_{i=1}^{n} p(x_i) I(x_i) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i)$

This amount is, indeed, average entropy H(X). Here, I (X) is the probability density function.

The entropy function used by C4.5 is $H(S) = -\sum_{i=1}^{c} p_i \log_2 p_i$

Where pi = portion of S (total number of instances) belonging to state (class) 'i' out of 'c' states (classes).

INFORMATION GAIN

The projected decrease in entropy as a result of separating the samples based on an attribute (A) is defined as information obtained.

Clearly, information (G) acquired by A as a result of data partitioning is equal to the difference between the initial entropy and the entropy after partitioning.

That is, G = H1 - H2.

Thus, information obtained by an attribute A, i.e., gain (S, A), is mathematically stated as: Gain (S, A) = Entropy (S) - $\sum_{v \in Values\ (A)} \frac{|S_v|}{|S|} Entropy\ (S_v)$

Where values (A) = the set of all possible values for the attribute A,

|Sv| = number of examples belonging to value v ∈ Values (A), out of the total number of examples (|S|).

It is, indeed, Gain (S, A) = H1 - H2.

Table 4 provides an explanation of a decision tree using a small dataset with four input qualities and one output attribute. The input attributes include blood pressure, age, gender, and chest discomfort.

Serial	Age	Gender	Chest Pain	Blood Pressure	Existence of Heart Disease
1	Senior	Male	Typical Angina	High	NO
2	Senior	Male	Asymptotic	High	YES
3	Senior	Male	Asymptotic	Normal	YES
4	Youth	Male	Non Anginal Pain	Normal	NO
5	Youth	Female	Atypical Angina	Normal	NO
6	Middle Age	Male	Atypical Angina	Normal	NO
7	Senior	Female	Asymptotic	High	YES
8	Middle Age	Female	Asymptotic	Normal	NO
9	Senior	Male	Asymptotic	Normal	YES
10	Middle Age	Male	Asymptotic	High	YES

Table 4: Dataset containing 4 input attributes and 10 tuples

In the table above, the class attribute 'Existence of Heart Disease' has two values: {No, Yes}. Total tuples = 10, number of 'NO' = 5, and number of 'yes' = {5}. Then s = 10 (5, 5) and Entropy (s) = 1.

Consider the characteristic 'Age': {Youth, Middle-age, Senior}.

Number (youth) = 2. Number (middle age) = 3; Number (senior) = 5.

Under 'Youth,' the number of tuples 'NO'= 2, while the number of tuples 'YES' = 0.

Under 'Middle-age,' the number of tuples 'NO' = 2, while the number of tuples 'yes' = 1.

Under 'Senior,' the number of tuples 'NO' = 1 and the number of tuples 'YES' = 4.

Then, $\sum_{V \in Age} \frac{|S_V|}{|S|}$ Entropy (S_V) = 0.378909.Therefore, Gain (S, Age) = 1- 0.378909 = 0.621090

Now let's look at the attribute 'Gender' = {Male, Female}.

Male number = 7, female number = 3.

There are three 'NO' tuples and four 'yes' tuples under the heading 'Male.'

There are two 'NO' tuples and one 'YES' tuple under the category 'Female.'

Under the heading 'Male,' the number of tuples 'NO' =3 and 'YES' = 4.

Similarly, under the category "Female," the number of tuples 'NO' = 2 and 'yes' = 1.

Then $\sum_{V \in Gender} \frac{|S_v|}{|S|} Entropy\ (S_V)$ = 0.9651477

Gain (S, Gender) = 1- .9651477 = 0.0348522.

Proceeding similarly for the attribute 'CP', then $\sum_{v\in CP}\frac{|S_V|}{|S|}$ Entropy (S_V) =0.3900134

Gain (S, CP) = 1-0.3900134 =0.609986

For the attribute 'BP', $\sum_{v\in BP}\frac{|S_V|}{|S|}$ Entropy (S_V) = 0.875488

Gain (S, BP) = 1- .875488 = 0.124511.

The attribute with the biggest information gain, 'Age', will be used to split the training example at the decision tree's root node in fig 4.

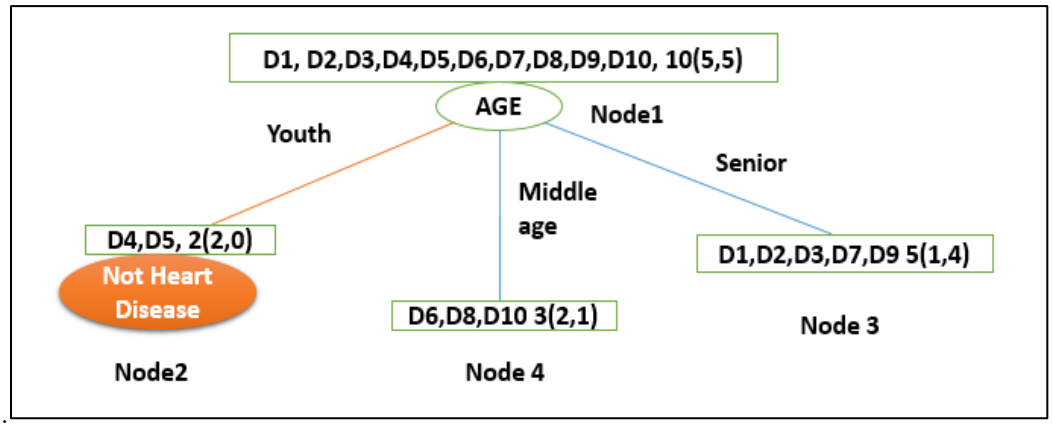

Fig. 4: Formation of a Decision Tree with an Initial State

Now for node 3 [D1, D2, D3, D7, D9, 5(1, 4)], the main table is decomposed into the following table 5:

Serial	Age	Gender	Chest Pain	Blood Pressure	Existence of Heart Disease
1	Senior	Male	Typical Angina	High	NO
2	Senior	Male	Asymptotic	High	YES
3	Senior	Male	Asymptotic	Normal	YES
7	Senior	Female	Asymptotic	High	YES
9	Senior	Male	Asymptotic	Normal	YES

Table 5: Dataset containing 4 input attributes and 5 tuples

Total tuple = 5, tuple 'No' = 1, and tuple 'Yes' = 4. Next, S1 = 5 (1, 4). S1 is equal to 0.721928.

Now, under 'Gender,' we have {Male, Female}, and Number (Male) = 4.

number (female) = 1.

Under 'Male,' the tuple 'No' = 1, and the tuples are labelled 'Yes' =3

Similarly, under 'Female,' the tuple 'No' = 0 and the tuples labelled 'Yes' = 1.

Then $\sum_{v\in Gender}\frac{|S_V|}{|S|}$ Entropy (S_V) =0.6490.

So Gain (S1, Gender) = 0.721928-0.6490 =0.07290

Similarly, for the attribute 'CP', $\sum_{v\in CP}\frac{|S_V|}{|S|}$ Entropy (S_V) = 0, Then Gain (S1, CP) =.721928-0 = 0.721928.

Again, for the attribute 'BP' $\sum_{v\in BP}\frac{|S_V|}{|S|}$ Entropy (S_V) =0.55097 .

Gain (S1, BP) = 0.721928 - 0.55097 = 0.170950. We are currently at Node 3 since Gain (S1, CP) has reached its maximum. Consequently, it gets raised to the root node position.

Now let's look at node 4: [D6, D8, D10, 3(2, 1)]. The following table is separated from the main table: 6

Serial	Age	Gender	Chest Pain	Blood Pressure	Existence of Heart Disease
6	Middle Age	Male	Atypical Angina	Normal	NO
8	Middle Age	Female	Asymptotic	Normal	NO
10	Middle Age	Male	Asymptotic	High	YES

Table 6: Dataset containing 4 input attributes and 3 tuples

In this table for Node 4, the total number of tuples = 3.

Number of tuples 'No' = 2, and 'Yes' = 1. Then S2 = 3(2, 1) and entropy (S2) = 0.918295

Now, number (Male) = 2 and number (Female) = 1 under the characteristic 'Gender' = {Male, Female}.

The number of tuples 'No' = 1 and 'Yes' = 1 under 'Male.'

Similarly, the number of tuples 'No' = 1 and 'Yes' = 0 under 'Female'.

Then, $\sum_{V\in Gender}\frac{|S_V|}{|S_2|}$Entropy (S_V) = 0.667.

Gain (S2, CP) = 0.918295 -0.667 = 0.251295. Again for the attribute 'BP', $\sum_{V\in BP}\frac{|S_V|}{|S_2|}$Entropy (S_V) = 0,

Gain (S2, BP) = 0.918295
Since Gain (S2, BP) = 0.918295 = maximum, it becomes the root node.
The final diagram is described in fig5.

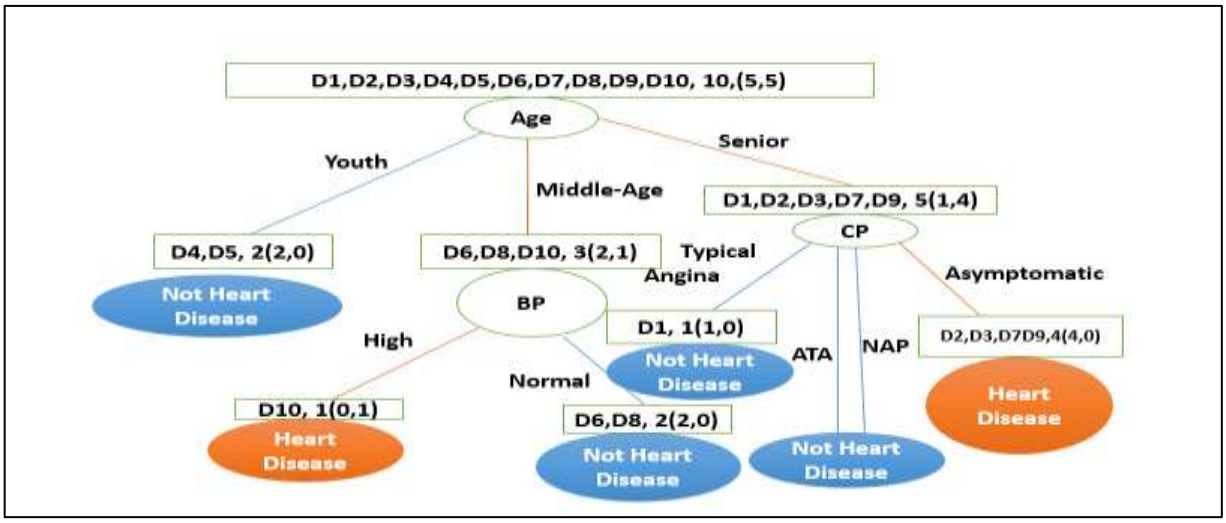

Fig. 5: Formation of a Decision Tree with a Final State

Here ATA = atypical angina and NAP = non-angial pain.
All parameters of the decision tree classifier are described as follows in table 7

Sl No	Sign (character)	Denotation
1	I	Information content or gain (I ≥ 0)
2	H	Loss of uncertainty (degree of doubt regarding e)
3	P	Probability of occurring e ($0 \leq p \leq 1$)
4	E	Expectation
5	X	Variable
6	{x1,x2,x3….xn}	Outcome of the variable X
7	I(X)	Probability Density Function
8	H(X)	Average Entropy
9	S	Total number of instances.
10	pi	Portion of the state
11	A	Attributes
12	G(S,A)	Gained information by A
13	$\lvert S_V \rvert$	Number of examples belonging to the value v ∈ values (A)
14	Entropy (S)	The entropy of the total instance S

Table 7: Parameters in a Decision Tree Classifier and their meaning

4.3 Naïve-Bayes Classifier

Bayes Theorem

Consider the tuple X in class C, where X is observed.
If H is another tuple, the probability of its occurrence in the presence of X is defined as

$$P\left(H/X\right) = \frac{P\,(H \cap X)}{P\,(X)} = \frac{P(H)\,P\left(X/H\right)}{P(X)}$$

Where P (H/X) represents the conditional probability of H in the presence of X.

Naïve-Bayes Classification

Consider D to be the collection of tuples from the training set.
Where C_1, C_2, C_3 ,... C_m are the m numbers of classes, and X = n-dimensional tuples.
Here, n represents the number of attributes.

If the tuple X belongs to the class Ci, then $P\left(Ci/X\right) > P\left(Cj/X\right)$

Where, I $\leq j \leq$ m and i $\neq j$

Then, we maximize $P\left(Ci/X\right)$ such that, $P\left(Ci/X\right) = \frac{P(Ci)\,P\left(X/Ci\right)}{P(X)}$

Since $P\left(Ci/X\right)$ maximize and P(X) is constant, then we maximize P(Ci) $P\left(X/Ci\right)$,
if the class Ci (i= 1,2,3,4,..m) is not equally likely.
Otherwise, if all classes are equally likely, then P (C1) = P (C2) = P (C3) =... = P (Cn).
Then maximize $P\left(X/Ci\right)$ only.

Now to reduce computational time Naïve-Bayes assumes that $P\left(X/_{Ci}\right)$ is conditionally independent.

Therefore, $P\left(X/_{Ci}\right) = \prod_{k=1}^{n} P\left(X_K/_{Ci}\right)$

$= P\left(X_1/_{Ci}\right). P\left(X_2/_{Ci}\right). P\left(X_3/_{Ci}\right)\ldots\ldots. P\left(X_K/_{Ci}\right)$

Time Complexity: O (n*d*c), where 'n' is the number of data points, 'd' is the number of attributes, and 'c' is the number of classes. But for the optimization, it can be reduced to O (n*d).

All parameters of the Naïve-Bayes Classifier are described as follows in table 8:

Sl No	Sign (character)	Denotation
1	X	Tuples
2	C	Class
3	P (H/X)	Probability of the occurrence of H in the presence of X
4	D	Tuples of the training set
5	n	Number of data points
6	C1,C2,C3….Cm	m number of classes.
7	⟨x1, x2, x3 …. xn⟩	n-dimensional tuples
8	D	Number of attributes

Table 8: Parameters in a Naïve-Bayes classifier and their meaning

4.4 SVM Classifier

Decision Boundary

The hyperplane [47] that divides the feature space into two different parts is called the 'Decision Boundary'.

Properties of Hyper-plane

In one dimension, a hyperplane is called a point. In two dimensions, it is called a line. In three dimensions, it is called a plane.

In more dimensions (> 3), it is called a hyperplane.

Hyperplane Properties:

A point is the term used to describe a hyperplane in one dimension. A line is a two-dimensional object. A plane is defined in three dimensions. In higher dimensions (> 3), it is known as a hyperplane.

Some significant factors in the SVM classifier.

Support Vectors: Support vectors are the data points closest to the decision border.

Optimal Hyperplane: The hyperplane that is farthest from the support vectors is referred to as the optimal hyperplane.

Margin: The margin refers to the space between separating hyperplanes and support vectors. Finding out the biggest margin

Choose the dataset first, and then choose two hyperplanes to divide the data so that there are no points in between. Lastly, maximize the margin—their distance.

By far the largest margin will be the area enclosed by the two hyperplanes.

Consider x_i $(i = 1,2 \ldots . n)$ stand for n input vectors and y_i for output vectors with +1 and -1 values. Thus, the original dataset is represented as follows from the set theory:

$D = \{(x_i, y_i) | x_i \in R^P, y_i \in \{-1,1\}\}, \text{for } i = 1 \text{ to } n$ (1)

Now consider w. x + b =1 (2) be the equation of the hyperplane H1

Similarly w. x + b= -1 (3) be the equation of the hyperplane H0

Hyperplanes H0 and H1 are explained in figures 6 and 7.

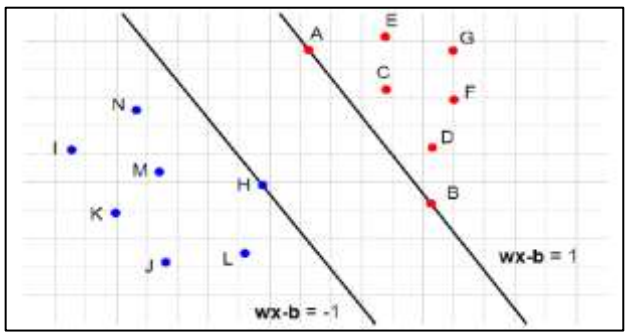

Fig 6: Two hyperplanes satisfying the constraints [47]

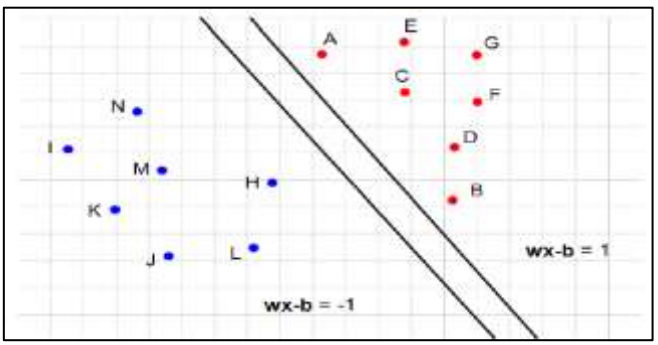

Fig 7: Two hyperplanes also satisfying the constraints [47]

Now for each vector xi, either
w.xi +b $\geq$1 for xi having the class 1 (4)
w.xi +b ≤ -1 for xi having the class -1 (5)
Equations (4) and (5) can be combined into a single constraint.
Now, we start with equation (5) for xi having the class -1, w.xi +b ≤ -1 and multiply both sides by yi (which is always -1 in this equation).
yi (w.xi +b) $\geq$ yi (-1) (6)
This means that equation (6) can be written as
yi (w.xi +b) $\geq$ 1 for xi having the class -1 (7)
Now from (4), as yi = 1, It does not change the sign of the equation.
yi (w.xi +b) $\geq$ 1, for xi having class1 (8)
Now combine equations (7) and (8).
Therefore, yi (w.xi +b) $\geq$ 1, $\forall\ 1 \leq i \leq n$ (9)
We have a single mathematically equivalent constraint (equation 9) in place of equations (4) and (5). They therefore have the same outcome.
Examine the hyperplane H_0's point x_0. Let's define the margin, m. The hyperplane H_1 and x_0 have a perpendicular distance (m). The position of x_0 in H_0 is represented by the distance m between the hyperplanes H_0 and H_1. It is depicted in Figure 8.

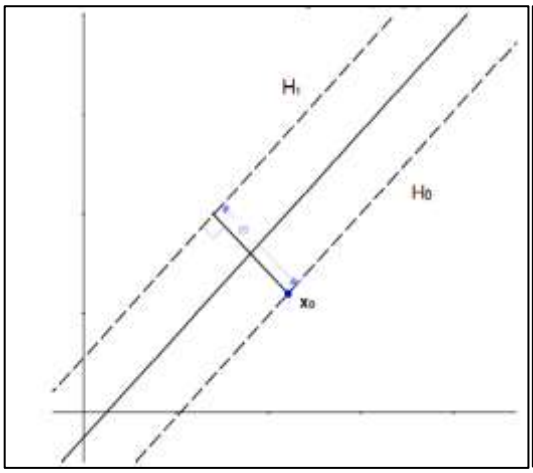

Fig 8: m is the distance between the two hyper planes [47]

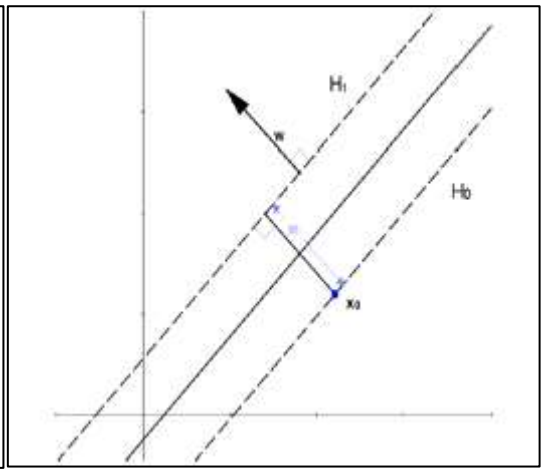

Fig 9: w is perpendicular to H1 [47]

To calculate the margin, or m, we must first establish the perpendicular to the hyperplane H_1.
Since H_1's equation is wx + b = 1, the vector perpendicular to H_1 is w. Figure 9 shows the perpendicular vector w to hyperplane H_1.
Now define the unit vector $\hat{u} = \frac{w}{\|w\|}$ (10) of w
As $\|u\| = 1$ and it has the same direction as w, so the unit vector $\hat{u}$ is also perpendicular to the hyper plane H_1.
Now multiply unit vector $\hat{u}$ by m,
We get K =mu and $\|K\| = m$, (11)
where, K is perpendicular to H1
Therefore, K= mu = m $\frac{w}{\|w\|}$ (12)
Starting at x_0 and adding K, we can find the point $z_0=x_0+K$ in the hyperplane H_1. Figure 10 illustrates this.

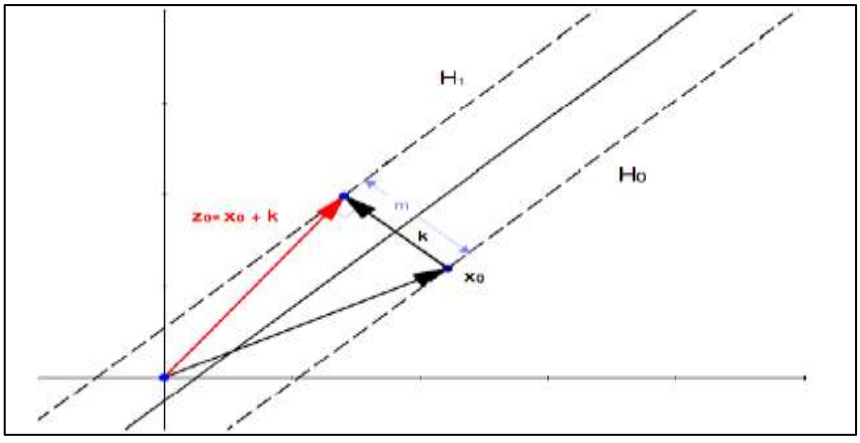

Fig. 10: z0 is a point on H1 [47]

Z_0 lies on the hyperplane H_1. Therefore, $z_0 = x_0 + K$ is in the hyperplane H_1.
Therefore, z_0 lies on the hyperplane H_1. Then $wz_0 + b = 1$. (13)
$w.(x_0 + K) + b = 1$ (14)
From (12) and (14)
$w.(x_0 + m\frac{w}{\|w\|}) + b = 1$ (15)
Now expanding equation (15)
$wx_0 + m\frac{w.w}{\|w\|} + b = 1$ (16)
$wx_0 + m\frac{\|w\|^2}{\|w\|} + b = 1$
$wx_0 + m\|w\| + b = 1$ (17)

$\Rightarrow wx_0 + b = 1 - m\|w\|$ (18)
Now since, x_0 is in H_0 then $wx_0 + b = -1$ (19)
From equations (18) and (19),
$-1 = 1 - m\|w\| \Rightarrow m\|w\| = 2$ (20)
$\Rightarrow m = \frac{2}{\|w\|}$ (21)
We can now only modify the norm of w as a variable in this formula.

$$\begin{cases} \text{When } \|w\| = 1; \text{ then } m = 2 \\ \text{When, } \|w\| = 2; \text{ then } m = 1 \\ \text{When, } \|w\| = 4; \text{ then } m = \frac{1}{2} \end{cases} \quad (22)$$

According to equation (22), the margin decreases as the norm increases. Reducing the norm of w and maximizing the margin are both comparable.
Our goal is to maximize the margins. In other words, pick the hyperplane with the smallest norm $\|w\|$.
This yields the following optimization problem.
Minimize (w,b) $\|w\|$
Subject to $y_i.(wx_0 + b) \geq 1$
(For any i = 1, 2, 3,….n)
Solving this problem is comparable to solving an equation: we identify the pair (w,b) that meets the stated requirements and has the smallest possible norm ‖w‖. This shows that we have the ideal hyperplane equation. Table 9 describes all of the parameters of an SVM classifier.

Sl No	Sign (character)	Denotation
1	xi (i=1,2,3….n)	N input vectors
2	yi	Output vectors having values +1, -1
3	w.x +b =1	Equation of the hyperplane H1
4	w.x+b = -1	Equation of the hyperplane H0
5	M	Margin, which is the distance between hyperplanes H0 and H1
6	w	Perpendicular vector to H1
7	$\|w\|$	Norm of w
8	$\hat{u}$	Unit vector
9	K	Perpendicular vector to H1

Table 9: Parameters in a SVM classifier and their meaning

4.5 Chi-Square (χ^2) Test (Feature Selection)
4.5.1 Role of Chi-Square (χ^2) Test
The chi-square test is useful in detecting categorical clinical factors that are highly associated with the presence of cardiac disease, such as chest pain, gender, family history, smoking, hypertension, and so on. It makes feature selection more efficient, improves model performance by removing unnecessary variables, helps clinicians understand statistical relationships, and supports evidence-based medical decisions.
4.5.2 Step by step analysis in Chi-Square (χ^2) test to find out feature selection

Step 1: Enter a dataset containing the features $X = \{x_1, x_2, \ldots, x_n\}$ and target variable y.
Step 2: If necessary, encode categorical features as numeric.
Step 3: Create a contingency table between feature values and class labels for every feature.
Step 4: Determine the observed frequency O_i (the dataset's actual counts).
Step 5: Calculate the predicted frequency E_i (counts assuming independence).
Step 6: Determine the chi-square value.

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

Step 7: Sort each feature according to its χ^2 scores.

Step 8: Choose which of the top k traits to keep for modeling.

Where:
O_i = observed frequency for category i.
E_i = expected frequency for category i.

4.5.3 Importance of p values in Chi-Square (χ^2) Test

4.5.3.1 Purpose of p-value

The Chi-Square (χ^2) test is a statistical tool for determining a significant connection between two categorical variables. In feature selection for classification models (such as logistic regression), it is typical to examine the association between each independent feature and the target variable.
The p-value indicates whether the observed association between a feature and the target variable occurred by coincidence.

4.5.3.2 How p-value Works

Hypothesis Setup:
The Null Hypothesis (H_0): It states that the feature and target variable are independent and have no relationship.
Alternative Hypothesis (H_1): The characteristic and target variable are dependent, and there is a relationship.

4.5.3.3 Interpretation of p-value

P values range, meaning and decision are explained in table 10.

p-value	Meaning	Decision
$p \leq 0.05$	Smaller p-values indicate **stronger evidence** against H_0, meaning the observed difference is unlikely due to chance. There is a significant relationship.	Reject H_0 → (Feature is important)
$p > 0.05$	larger p-values indicate **weak evidence** against H_0, Weak or no evidence of association	Fail to reject H_0→ (no significant)

Table 10: p values interpretation and corresponding decision

4.5.4 Clarification of observed frequency and expected frequency in the Chi-Square (χ^2) test

The Chi-Square (χ^2) test compares observed and anticipated frequencies in categorical data.

Because the heart disease dataset consists primarily of numerical features, discretization (binning) is a crucial pre-

processing step before computing frequencies.
The following is a clear explanation adapted to the heart disease dataset.

Observed frequency (O)
The observed frequency of a variable is the number of records that fall into each category (or bin) relative to the target class.
Because numerical attributes cannot be used directly, they are first transformed into intervals (bins).
Example: Age vs. Heart Disease.
Suppose:
Target variable: Heart Disease (0 = No, 1 = Yes).
Numerical attribute: Age

Step 1: Convert numerical values into categories

Age Group	Range
Young	≤ 40
Middle	41–60
Old	> 60

Step 2: Create a contingency table (Observed Frequencies)

Age Group	heart disease = 0	Heart disease = 1	Row Total
Young	30	15	45
Middle	50	40	90
	15	50	65
Column Total	95	105	200 (grand total)

Step3. Expected Frequency (E)
Expected frequency represents the frequency that would occur if there were no association between the attribute and the target (null hypothesis).

$$E_{ij} = \frac{(\text{Row Total}_i \times \text{Column Total}_j)}{\text{Grand Total}}$$

Step4. Calculation

For Young and Heart Disease = 0: expected frequency $E_{11} = \frac{45\ x\ 95}{200} = 21.375$; observed frequency $O_{11} = 30$

For Young and Heart Disease = 1: expected frequency $E_{12} = \frac{45\ x\ 105}{200} = 23.625$; observed frequency $O_{12} = 15$

For Middle and Heart Disease = 0: expected frequency $E_{21} = \frac{90x\ 95}{200} = 42.75$; observed frequency $O_{21} = 50$

For Middle and Heart Disease = 1: expected frequency $E_{22} = \frac{90x\ 105}{200} = 47.25$; observed frequency $O_{22} = 40$

For Old and Heart Disease = 0: expected frequency $E_{31} = \frac{65x\ 95}{200} = 30.875$; observed frequency $O_{31} = 15$

For Old and Heart Disease = 1: expected frequency $E_{32} = \frac{65x\ 105}{200} = 34.125$; observed frequency $O_{32} = 50$

Step5. Final Chi-Square statistic result

Once observed and expected frequencies are obtained:

Calculate Chi-Square statistic:

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i} = \frac{(O11-E11)^2}{E11} + \frac{(O12-E12)^2}{E12} + \frac{(O21-E21)^2}{E21} + \frac{(O22-E22)^2}{E22} + \frac{(O31-E31)^2}{E31} + \frac{(O32-E32)^2}{E32}$$

$$= \frac{(30-21.375)^2}{21.375} + \frac{(15-23.625)^2}{23.625} + \frac{(50-42.75)^2}{42.75} + \frac{(40-47.25)^2}{47.25} + \frac{(15-30.875)^2}{30.875} + \frac{(50-34.125)^2}{34.125}$$

$$= 3.480263158 + 3.148809524 + 1.229532164 + 1.112433862 + 8.162449393 + 7.38507326$$

$$= 24.51856136$$

4.6 Performance evaluation

In this study, we employed the confusion matrix [44] tool to analyze the system's effectiveness in precisely diagnosing cardiac disease as well as to quantify accuracy, sensitivity, and specificity. Figure 3 shows how to retrieve the confusion matrix findings.

Confusion Matrix

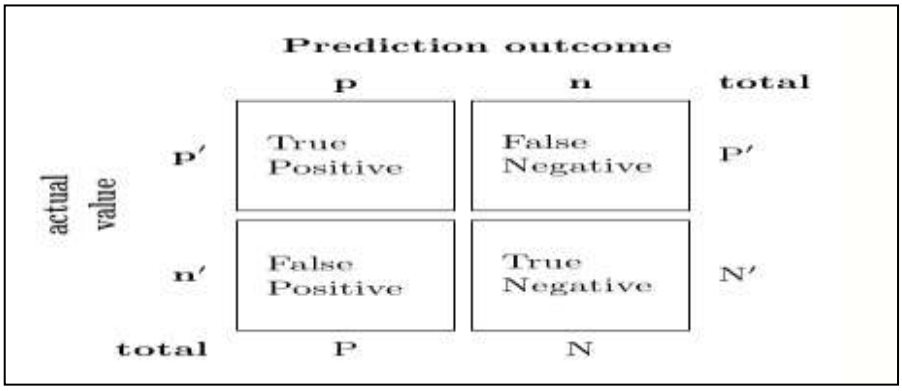

Fig.11: Confusion Matrix [44]

Equations (1), (2), and (3) demonstrate the mathematical calculations' accuracy, sensitivity, and specificity. Examples are provided below.

True Positive (TP): An outcome in which the model correctly predicts the positive class.

True Negative (TN): An outcome in which the model correctly predicts the negative class.

False Positive (FP): When the model mistakenly predicts the positive class.

False Negative (FN): When the model mistakenly predicts the negative class.

Positive (P): The number of true positive cases in the sample.

Negative (N): The number of real negative cases in the specified sample.

$$\text{Accuracy} = \frac{TP + TN}{TP+TN+FP+FN} \quad (1)$$

$$\text{Sensitivity} = \frac{TP}{P} \quad (2)$$

$$\text{Specificity} = \frac{TN}{N} \quad (3)$$

5. Result and Observation

5.1 Experimental Set up

Hardware Setup

Windows x64

RAM: 10GB

Processor: Intel(R) Core (TM) i3 @2.40GHz

Software Setup

Python Language

5.2 Using decision tree the experimental outcome in Heart Failure Clinical Record heart dataset are explained below:

Class	Precision	Recall	F1-Score	Support
0	0.95	0.94	0.95	155
1	0.87	0.90	0.89	69
Accuracy	-	-	0.93	224
Macro Avg	0.91	0.92	0.92	224
Weighted Avg	0.93	0.93	0.93	224

Time elapsed: 18.163013458251953 sec.
Training Accuracy for Decision Tree: 92.85714285714286%
Training Sensitivity for Decision Tree: 95.42483660130719%
Training Specificity for Decision Tree: 87.32394366197182%
Training Precision for Decision Tree: 94.19354838709677%

Confusion matrix for training set using decision tree is explained in fig12.

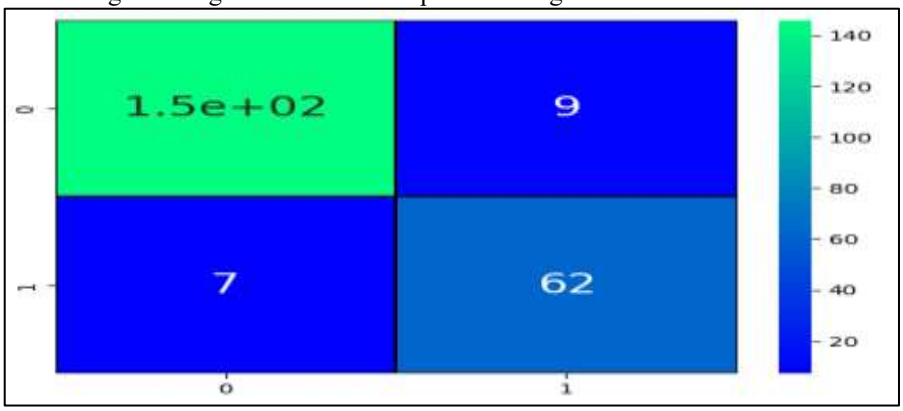

Fig.12: Confusion matrix for training set

Class	Precision	Recall	F1-Score	Support
0	0.96	0.92	0.94	48
1	0.86	0.93	0.89	27
Accuracy	-	-	0.92	75
Macro Avg	0.91	0.92	0.91	75
Weighted Avg	0.92	0.92	0.92	75

Testing Accuracy for Decision Tree: 92.0%
Testing Sensitivity for the Decision Tree: 95.65217391304348%
Testing Specificity for Decision Tree: 86.20689655172413%
Testing Precision for Decision Tree: 91.66666666666666%

Confusion Matrix for Testing set using decision tree is demonstrated in fig13.

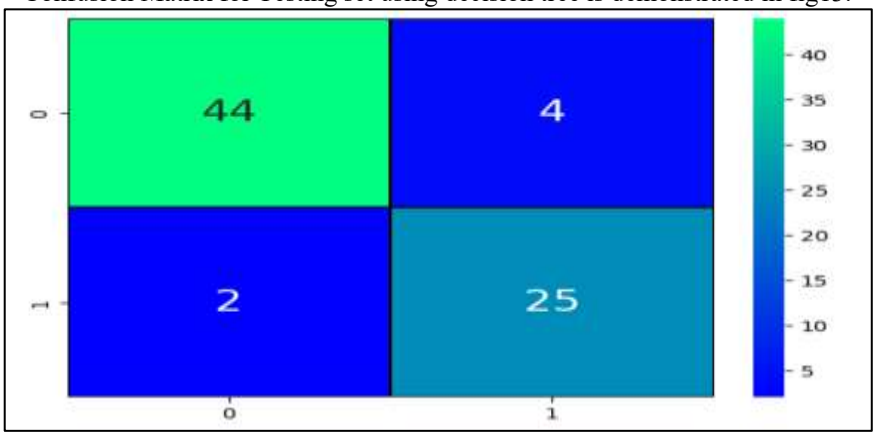

Fig.13: Confusion matrix for testing set

Entropy of Different Nodes:

Node 0: entropy=0.6188335975448243
Node 1: entropy=0.5771777113019608
Node 2: entropy=0.1549190764093953
Node 3: entropy=0.0
Node 4: entropy=0.0
Node 5: entropy=0.0
Node 6: entropy=0.0
Node 7: entropy=0.7342987750760639
Node 8: entropy=0.7552774044535872
Node 9: entropy=0.0
Node 10: entropy=0.6482088240384631
Node 11: entropy=0.5032583347756457
Node 12: entropy=0.8112781244591328
Node 13: entropy=0.0
Node 14: entropy=0.0
Node 15: entropy=0.0
Node 16: entropy=0.501031801427556
Node 17: entropy=0.3313066073264381
Node 18: entropy=0.7282129458410014
Node 19: entropy=0.0
Node 20: entropy=0.7717594215235591
Node 21: entropy=0.8904916402194913
Node 22: entropy=0.0
Node 23: entropy=0.2123030246796273
Node 24: entropy=0.14354361236260166
Node 25: entropy=0.10618783546209642
Node 26: entropy=0.8112781244591328
Node 27: entropy=0.46358749969093305
Node 28: entropy=0.8112781244591328
Node 29: entropy=0.0
Node 30: entropy=0.8483857803777466
Node 31: entropy=0.0
Node 32: entropy=0.7744181671966338
Node 33: entropy=0.7830246729473134
Node 34: entropy=0.0
Node 35: entropy=0.9709505944546686
Node 36: entropy=0.0
Final entropy of the decision tree: 0.6188335975448243
A decision tree of different nodes are explained in fig14.

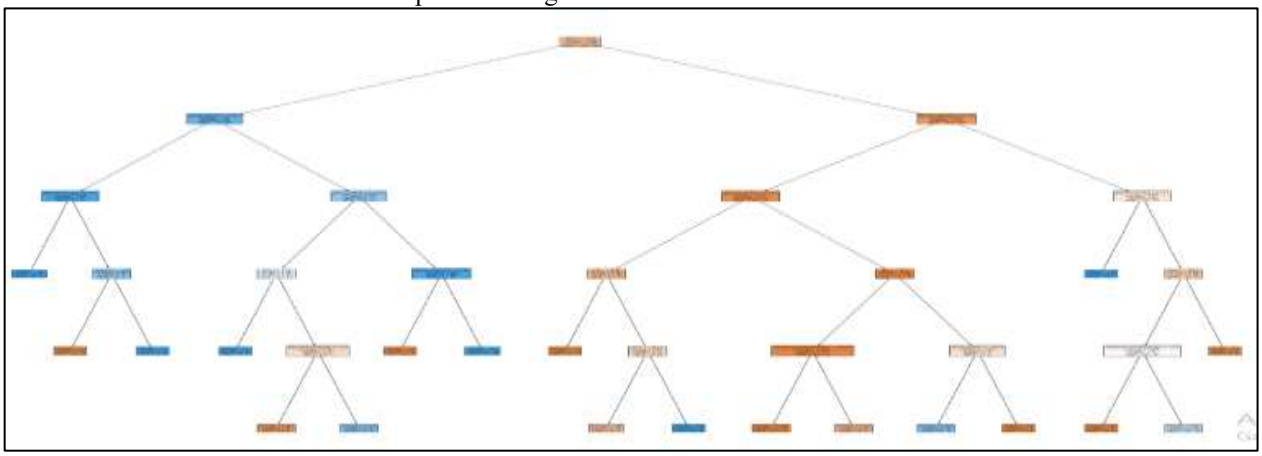

Fig. 14: Decision Tree for Heart Failure Clinical Record Heart Dataset

5.3 Experimental outcome in Heart Failure Clinical Record heart dataset from root attribute to leaf attribute and finding out the longest paths of attributes using decision tree are explained below

5.3.1. Branch [time = yes]
Leaf Node Reached. Final Path: serum creatinine -> ejection Fraction -> serum sodium -> anaemia -> diabetes -> age -> creatinine phosphokinase -> high blood pressure -> platelets -> gender -> smoking -> time In this path if time = yes, then predicted output: high chance of death

5.3.2 Branch [smoking = no]
Leaf Node Reached. Final Path: serum creatinine -> ejection Fraction -> time -> diabetes -> platelets -> high blood pressure -> creatinine phosphokinase -> age -> anaemia -> serum sodium -> gender -> smoking
In this path if smoking = no, then predicted output: high chance of death

5.3.3 Branch [smoking = no]
Leaf Node Reached. Final Path: serum creatinine -> ejection Fraction -> time -> diabetes -> platelets -> high blood pressure -> creatinine phosphokinase -> age -> gender -> serum sodium -> anaemia -> smoking
In this path if smoking = no, then predicted output: high chance of death

5.3.4. Branch [smoking = yes]
Leaf Node Reached. Final Path: serum creatinine -> ejection Fraction -> time -> age -> creatinine phosphokinase -> high blood pressure -> diabetes -> anaemia -> platelets -> serum sodium -> gender -> smoking
In this path if smoking = yes, then predicted output: high chance of death

5.3.5. Branch [smoking = no]
Leaf Node Reached. Final Path: serum creatinine -> ejection Fraction -> time -> platelets -> creatinine phosphokinase -> gender -> anaemia -> age -> diabetes -> high blood pressure -> serum sodium -> smoking

In this path if smoking = no, then predicted output: high chance of death

5.3.6. Branch [serum sodium = low]
Leaf Node Reached. Final Path: serum creatinine -> ejection Fraction -> time -> platelets -> creatinine phosphokinase -> gender -> diabetes -> high blood pressure -> age -> smoking -> anaemia -> serum sodium
In this path if serum sodium = low, then predicted output: low chance of death

5.3.7. Branch [serum sodium = normal]
Leaf Node Reached. Final Path: serum creatinine -> ejection Fraction -> time -> platelets -> creatinine phosphokinase -> gender -> diabetes -> high blood pressure -> age -> smoking -> anaemia -> serum sodium
In this path if serum sodium = normal, then predicted output: high chance of death

5.3.8. Branch [smoking = no]
Leaf Node Reached. Final Path: serum creatinine -> ejection Fraction -> time -> platelets -> creatinine phosphokinase -> gender -> diabetes -> age -> high blood pressure -> anaemia -> serum sodium -> smoking
In this path if smoking = no, then predicted output: high chance of death

5.3.9. Branch [smoking = yes]
Leaf Node Reached. Final Path: serum creatinine -> ejection Fraction -> time -> platelets -> creatinine phosphokinase -> gender -> diabetes -> age -> high blood pressure -> anaemia -> serum sodium -> smoking
In this path if smoking = yes, then predicted output: high chance of death

Additional details are given in Appendix 1.

5.4 Using a decision tree, the experimental outcome in the Heart Statlog Cleveland Hungary final heart dataset is explained below:
Training Set:

Class	Precision	Recall	F1-Score	Support
0	0.92	0.81	0.86	422
1	0.84	0.94	0.89	470
Accuracy	-	-	0.87	892
Macro Avg	0.88	0.87	0.87	892
Weighted Avg	0.88	0.87	0.87	892

Time elapsed: 29.09041428565979 Sec
Training Accuracy for Decision Tree: 87.4439461883408%
Training Sensitivity for Decision Tree: 91.8918918918919%
Training Specificity for Decision Tree: 84.2911877394636%
Training Precision for Decision Tree: 80.56872037914692%

Confusion matrix for training set using decision tree is explained in fig 15.

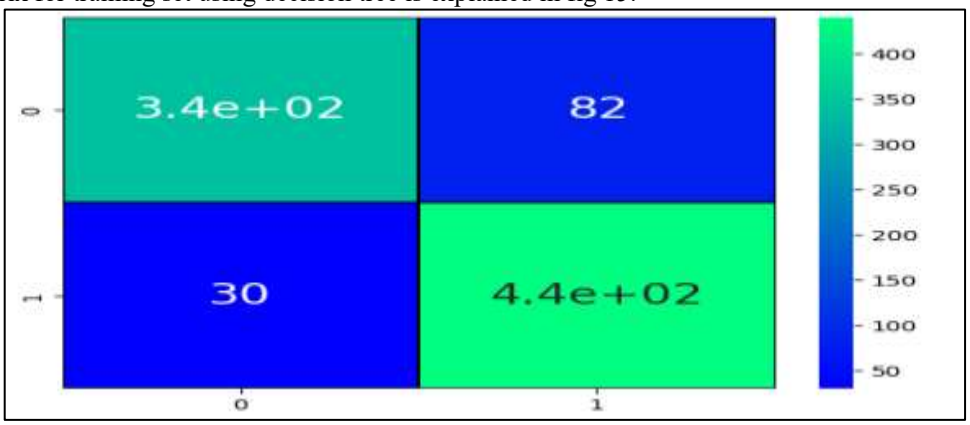

Fig. 15: Confusion matrix for the training set

Testing Set:

Class	Precision	Recall	F1-Score	Support
0	0.88	0.79	0.83	139
1	0.83	0.91	0.87	159

Class	Precision	Recall	F1-Score	Support
Accuracy	-	-	0.85	298
Macro Avg	0.86	0.85	0.85	298
Weighted Avg	0.85	0.85	0.85	298

Testing Accuracy for Decision Tree: 85.23489932885906%
Testing Sensitivity for Decision Tree: 88.0%
Testing Specificity for Decision Tree: 83.23699421965318%
Testing Precision for Decision Tree: 79.13669064748201%

Fig 16 explains the confusion matrix for testing set using decision trees.

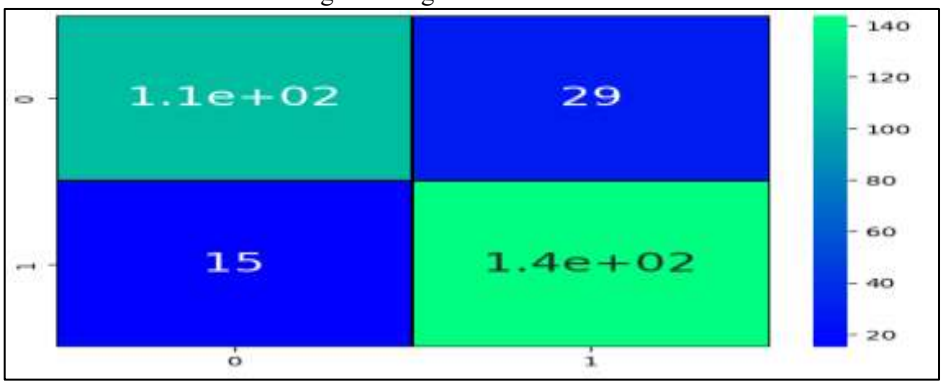

Fig. 16: Confusion matrix for the testing set

Entropy of Different Nodes:
Node 0: entropy=0.751098582169833
Node 1: entropy=0.5921273465496808
Node 2: entropy=0.32751239702877577
Node 3: entropy=0.2945457864515948
Node 4: entropy=0.13730242450351712
Node 5: entropy=0.13837092586152114
Node 6: entropy=0.0
Node 7: entropy=0.5938164188022539
Node 8: entropy=0.7335379291086666
Node 9: entropy=0.0
Node 10: entropy=0.0
Node 11: entropy=0.8919272134885369
Node 12: entropy=0.8318995711732381
Node 13: entropy=0.38335413017959974
Node 14: entropy=0.9182958340544896
Node 15: entropy=0.27138958750811115
Node 16: entropy=0.8542170615609418
Node 17: entropy=0.9890934397021431
Node 18: entropy=0.0
Node 19: entropy=0.7134646505976849
Node 20: entropy=0.0
Node 21: entropy=0.0
Node 22: entropy=0.0
Node 23: entropy=0.5342814838144135
Node 24: entropy=0.38431154412649704
Node 25: entropy=0.9957274520849256
Node 26: entropy=0.6560856057748817
Node 27: entropy=0.8733174229630523
Node 28: entropy=0.5622200674726635
Node 29: entropy=0.5925630247991779
Node 30: entropy=0.0
Node 31: entropy=0.9980008838722996
Node 32: entropy=0.0
Node 33: entropy=0.8065571015365084
Node 34: entropy=0.3804101172796514
Node 35: entropy=0.7024665512903903
Node 36: entropy=0.18717625687320816
Node 37: entropy=0.915799537013519
Node 38: entropy=0.9709505944546686
Node 39: entropy=0.5665095065529053
Node 40: entropy=0.44962930086671127
Node 41: entropy=0.7429700767845483
Node 42: entropy=0.8108804125949092
Node 43: entropy=0.9910760598382222
Node 44: entropy=0.0
Node 45: entropy=0.2371669158130201
Node 46: entropy=0.7642045065086203
Node 47: entropy=0.0
Node 48: entropy=0.33663765749618935
Node 49: entropy=0.2517612506409174
Node 50: entropy=0.33327411457793576
Node 51: entropy=0.0
Node 52: entropy=0.555381020554554
Node 53: entropy=0.9575534837147482
Node 54: entropy=0.0
Final entropy of the decision tree: 0.751098582169833

Fig. 17 explain decision tree of different nodes

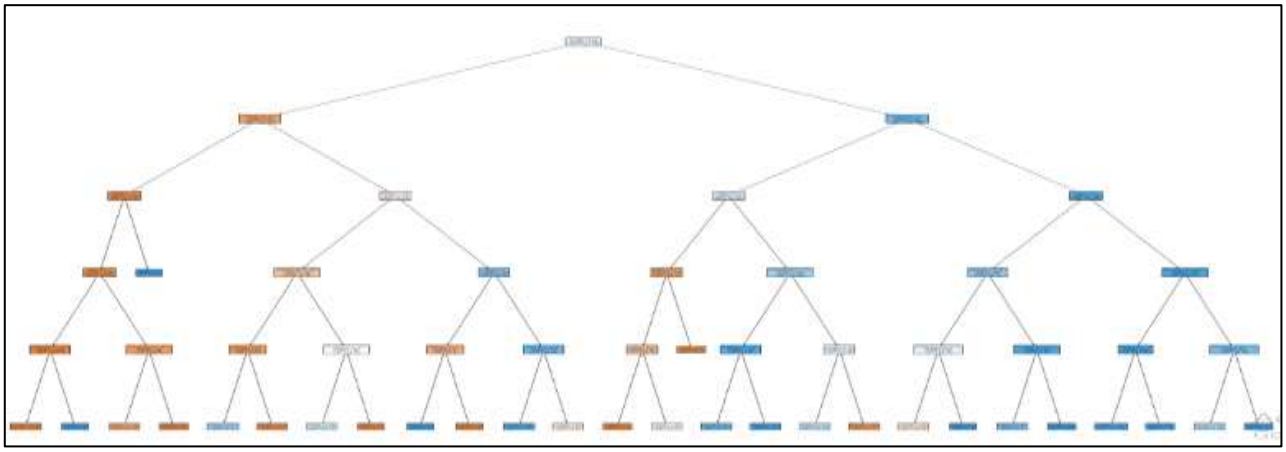

Fig. 17. Decision tree for the Heart Statlog Cleveland Hungary final heart dataset

5.5 Experimental outcome in 'Heart Statlog Cleveland Hungary' final heart dataset from root attribute to leaf attribute and finding out the longest paths of attributes using decision tree are explained below

5.5.1. Branch [exercise angina = yes]
Leaf Node Reached. Final Path: ST slope -> max heart rate -> chest pain type -> cholesterol -> oldpeak -> resting ecg -> resting bps -> age -> gender -> fasting blood sugar -> exercise angina
In this path if exercise angina = yes, then predicted output: low chance of death

5.5.2. Branch [fasting blood sugar = normal]
Leaf Node Reached. Final Path: ST slope -> max heart rate -> chest pain type -> cholesterol -> exercise angina -> resting ecg -> oldpeak -> age -> gender -> resting bps -> fasting blood sugar
In this path if fasting blood sugar = normal, then predicted output: low chance of death

5.5.3. Branch [oldpeak = risk]
Leaf Node Reached. Final Path: ST slope -> max heart rate -> gender-> exercise angina -> chest pain type -> cholesterol -> age -> resting bps -> fasting blood sugar -> resting ecg -> oldpeak
In this path if oldpeak = risk, then predicted output: low chance of death

5.5.4. Branch [exercise angina = yes]
Leaf Node Reached. Final Path: ST slope -> max heart rate -> gender -> chest pain type -> cholesterol -> age -> oldpeak -> resting bps -> fasting blood sugar -> resting ecg -> exercise angina
In this path if exercise angina = yes, then predicted output: low chance of death

5.5.5. Branch [fasting blood sugar = normal]
Leaf Node Reached. Final Path: ST slope -> max heart rate -> gender -> chest pain type -> cholesterol -> resting bps -> oldpeak -> exercise angina -> resting ecg -> age -> fasting blood sugar
In this path if fasting blood sugar = normal, then predicted output: Low chance of death

5.5.6. Branch [oldpeak = risk]
Leaf Node Reached. Final Path: ST slope -> max heart rate -> gender -> chest pain type -> age -> resting ecg -> cholesterol -> exercise angina -> resting bps -> fasting blood sugar -> oldpeak
In this path if oldpeak = risk, then predicted output: low chance of death

5.5.7. Branch [exercise angina = no]
Leaf Node Reached. Final Path: ST slope -> max heart rate -> gender -> chest pain type -> age -> resting ecg -> cholesterol -> oldpeak -> resting bps -> fasting blood sugar -> exercise angina
In this path if exercise angina = no, then predicted output: low chance of death

5.5.8. Branch [exercise angina = yes]
Leaf Node Reached. Final Path: ST slope -> max heart rate -> gender-> chest pain type -> age -> resting ecg -> cholesterol -> oldpeak -> resting bps -> fasting blood sugar -> exercise angina
In this path if exercise angina = yes, then predicted output: low chance of death

5.5.9. Branch [age = middle age]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> exercise angina -> fasting blood sugar -> resting ecg -> oldpeak -> max heart rate -> resting bps -> cholesterol -> gender -> age
In this path if age = middle age, then predicted output: low chance of death

5.5.10. Branch [gender = female]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> exercise angina -> fasting blood sugar -> resting ecg -> oldpeak -> max heart rate -> resting bps -> cholesterol -> age -> gender
In this path if gender = female, then predicted output: low chance of death

5.5.11. Branch [gender = male]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> exercise angina -> fasting blood sugar -> resting ecg -> oldpeak -> max heart rate -> resting bps -> cholesterol -> age -> gender
In this path if gender = male, then predicted output: low chance of death

5.5.12. Branch [max heart rate = medium]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> exercise angina -> fasting blood sugar -> resting ecg -> resting bps -> gender -> oldpeak -> cholesterol -> age -> max heart rate
In this path if max heart rate = medium, then predicted output: low chance of death

5.5.13. Branch [max heart rate = high]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> exercise angina -> fasting blood sugar -> resting ecg -> resting bps -> gender -> oldpeak -> cholesterol -> age -> max heart rate
In this path if max heart rate = high, then predicted output: low chance of death

5.5.14. Branch [max heart rate = low]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> exercise angina -> fasting blood sugar -> resting ecg -> resting bps -> gender -> oldpeak -> cholesterol -> age -> max heart rate
In this path if max heart rate = low, then predicted output: low chance of death

5.5.15. Branch [max heart rate = medium]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> exercise angina -> fasting blood sugar -> resting ecg -> resting bps -> gender -> oldpeak -> cholesterol -> age -> max heart rate
In this path if max heart rate = medium, then predicted output: low chance of death

5.5.16. Branch [max heart rate = high]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> exercise angina -> fasting blood sugar -> resting ecg -> resting bps -> gender -> oldpeak -> age -> cholesterol -> max heart rate
In this path if max heart rate = high, then predicted output: low chance of death

5.5.17. Branch [fasting blood sugar = normal]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> exercise angina -> oldpeak -> gender -> cholesterol -> resting ecg -> age -> max heart rate -> resting bps -> fasting blood sugar
In this path if fasting blood sugar = normal, then predicted output: low chance of death

5.5.18. 5.5.19. Branch [oldpeak = low]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> age -> resting ecg -> max heart rate -> cholesterol -> gender -> fasting blood sugar -> resting bps -> exercise angina -> oldpeak
In this path if oldpeak = low, then predicted output: low chance of death

5.5.19. Branch [exercise angina = no]
Leaf Node Reached. Final Path: ST slope -> chest pain type -> oldpeak -> resting bps -> gender -> resting ecg -> age -> max heart rate -> cholesterol -> fasting blood sugar -> exercise angina
In this path if exercise angina = no, then predicted output: low chance of death

Appendix 2 provides detailed information.

5.6 Using Naïve-Bayes Classifier experimental outcome in 'Heart Failure Clinical Record' heart dataset is explained below:
Training data = 75%, Testing data = 25%; K=10 Fold Cross Validation
Training Set

Class	Precision	Recall	F1-Score	Support
0	0.79	0.94	0.86	163
1	0.80	0.47	0.60	76
Accuracy	-	-	0.79	239
Macro Avg	0.80	0.71	0.73	239
Weighted Avg	0.80	0.79	0.78	239

Time Elapsed: 14.005379676818848 Sec
Training Accuracy for Naive Bayes: 79.49790794979079%
Training Sensitivity for Naive Bayes: 79.38144329896907%
Training Specificity for Naive Bayes: 80.0%
Training Precision for Naive Bayes: 94.47852760736197%

Figure 18 explains the confusion matrix for training set using Naïve-Bayes classifier

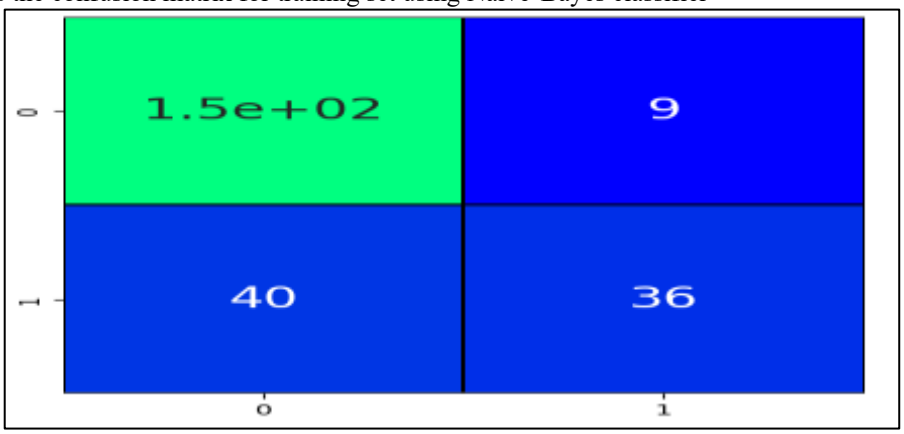

Fig. 18: Confusion matrix for the training set

For testing data

Class	*Precision*	*Recall*	*F1-Score*	*Support*
0	*0.81*	*0.95*	*0.87*	*40*
1	*0.85*	*0.55*	*0.67*	*20*
Accuracy	-	-	*0.82*	*60*
Macro Avg	*0.83*	*0.75*	*0.77*	*60*
Weighted Avg	*0.82*	*0.82*	*0.80*	*60*

Time Elapsed: 4.831974744796753 Sec
Testing Accuracy for Naive Bayes: 81.66666666666667
Testing Sensitivity for Naive Bayes: 80.85106382978722
Testing Specificity for Naive Bayes: 84.61538461538461
Testing Precision for Naive Bayes: 95.0

Fig.19 demonstrate confusion matrix for testing set using Naïve-Bayes Classifier

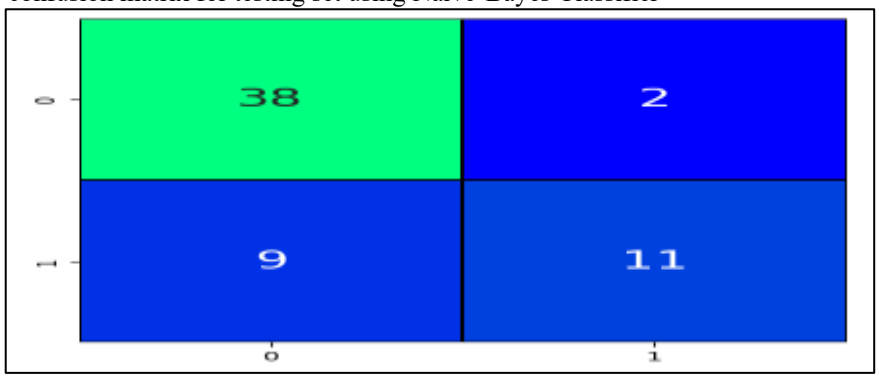

Fig. 19: Confusion matrix for the testing set

5.7 Using the Naïve-Bayes Classifier, experimental outcomes in the Heart Statlog Cleveland Hungary final heart dataset are explained below:
Training data = 75%; testing data = 25%; K = 10 Fold Cross Validation
Training Set

Class	Precision	Recall	F1-Score	Support
0	0.83	0.86	0.84	422
1	0.87	0.84	0.85	470
Accuracy	-	-	0.85	892

Class	Precision	Recall	F1-Score	Support
Macro Avg	0.85	0.85	0.85	892
Weighted Avg	0.85	0.85	0.85	892

Time elapsed: 19.40032434463501 sec.
Training Accuracy for Naive Bayes: 84.64125560538116%
Training Sensitivity for Naive Bayes: 82.6086956521739%
Training Specificity for Naive Bayes: 86.5934065934066%
Training Precision for Naive Bayes: 85.54502369668246%
Confusion matrix for training set using Naïve-Bayes classifier is explained in fig.20

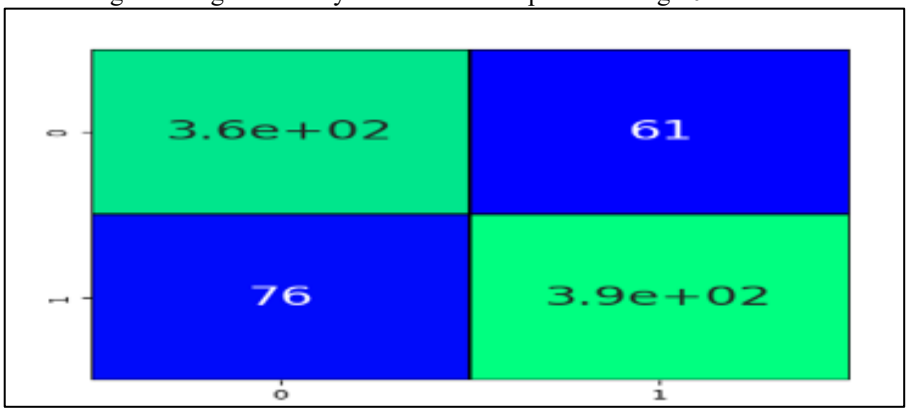

Fig. 20: Confusion matrix for the training set

Testing Set

Class	Precision	Recall	F1-Score	Support
0	0.81	0.80	0.80	139
1	0.83	0.84	0.83	159
Accuracy	-	-	0.82	298
Macro Avg	0.82	0.82	0.82	298
Weighted Avg	0.82	0.82	0.82	298

Time Elapsed: 4.768684387207031 Sec
Testing Accuracy for Naive Bayes: 81.87919463087249%
Testing Sensitivity for Naive Bayes: 81.02189781021897%
Testing Specificity for Naive Bayes: 82.6086956521739%
Testing Precision for Naive Bayes: 79.85611510791367%

Fig.21 explain confusion matrix for testing set using Naïve-Bayes Classifier

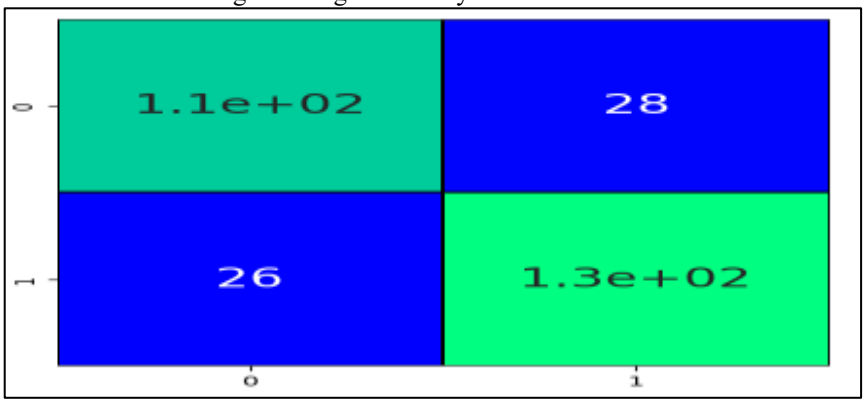

Fig. 21: Confusion matrix for the testing set

5.8 Using the SVM classifier, experimental outcomes in the Heart Failure Clinical Record Heart dataset are explained below:
Training data = 75%; testing data = 25%; K = 10 Fold Cross Validation
Training Set

Class	Precision	Recall	F1-Score	Support
0	0.84	0.92	0.88	155
1	0.78	0.61	0.68	69
Accuracy	-	-	0.83	224
Macro Avg	0.81	0.77	0.78	224
Weighted Avg	0.82	0.83	0.82	224

Time Elapsed: 536.5351357460022 Sec
Training Accuracy for SVM: 82.58928571428571%
Training Sensitivity for SVM: 84.11764705882354%
Training Specificity for SVM: 77.77777777777779%
Training Precision for SVM: 92.25806451612904%

Fig.22 demonstrate confusion matrix for training sets using SVM classifiers

Fig. 22: Confusion matrix for the training set

Testing Set

Class	Precision	Recall	F1-Score	Support
0	0.81	0.96	0.88	48
1	0.89	0.59	0.71	27
Accuracy	-	-	0.83	75
Macro Avg	0.85	0.78	0.79	75
Weighted Avg	0.84	0.83	0.82	75

Time Elapsed: 536.2378947734833 Sec
Testing Accuracy for SVM: 82.66666666666667%
Testing Sensitivity for SVM: 80.7017543859649%
Testing Specificity for SVM: 88.88888888888889%
Testing Precision for SVM: 95.83333333333334%

Confusion matrix for testing sets using SVM classifiers is explained in fig 23.

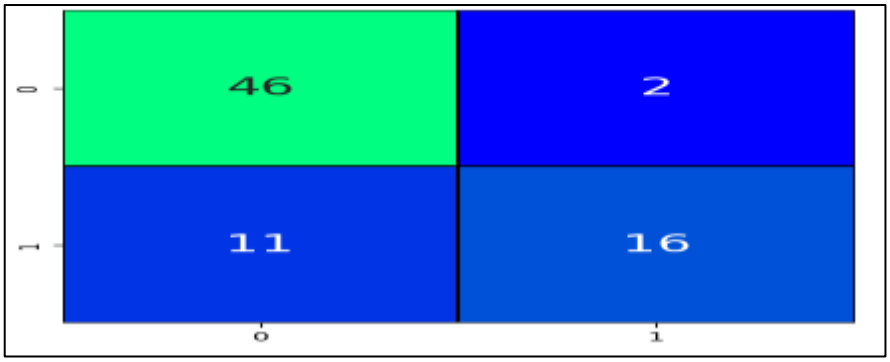

Fig. 23: Confusion matrix for the testing set

5.9 Using SVM Classifier, experimental outcomes in the Heart Statlog Cleveland Hungary final heart dataset are explained below:

Training data = 75%; testing data = 25%; K = 10 Fold Cross Validation
Training Set

Class	Precision	Recall	F1-Score	Support
0	0.83	0.84	0.84	422
1	0.85	0.85	0.85	470
Accuracy	-	-	0.85	892
Macro Avg	0.84	0.84	0.84	892
Weighted Avg	0.85	0.85	0.85	892

Time Elapsed: 110.58180022239685 Sec
Training Accuracy for SVM: 84.52914798206278%
Training Sensitivity for SVM: 83.49056603773585%
Training Specificity for SVM: 85.47008547008546%
Training Precision for SVM: 83.88625592417061%

Confusion matrix for training sets using SVM classifier is explained in fig. 24

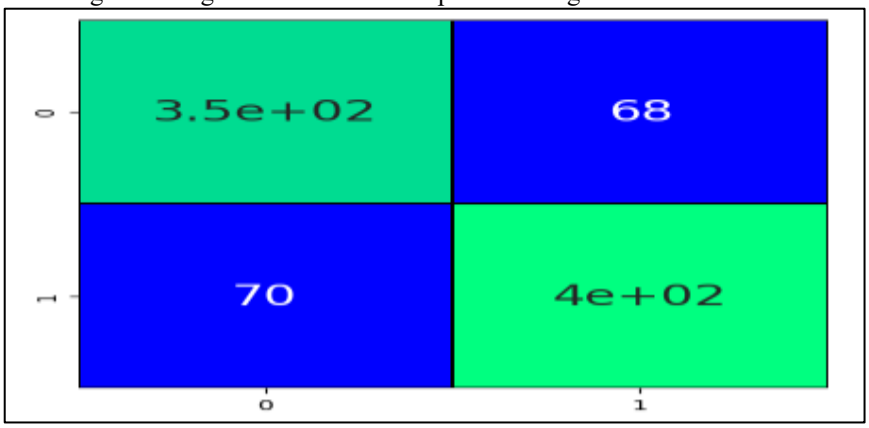

Fig. 24: Confusion matrix for training set

For testing data

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.81	0.79	0.80	139
1	0.82	0.84	0.83	159
Accuracy	-	-	0.82	298
Macro Avg	0.82	0.82	0.82	298
Weighted Avg	0.82	0.82	0.82	298

Time Elapsed: 77.00212049484253 Sec.
Testing Accuracy for SVM: 81.87919463087249%
Testing Sensitivity for SVM: 81.48148148148148%
Testing Specificity for SVM: 82.20858895705521%
Testing Precision for SVM: 79.13669064748201%

Fig 25 explain confusion matrix for testing sets using SVM classifiers

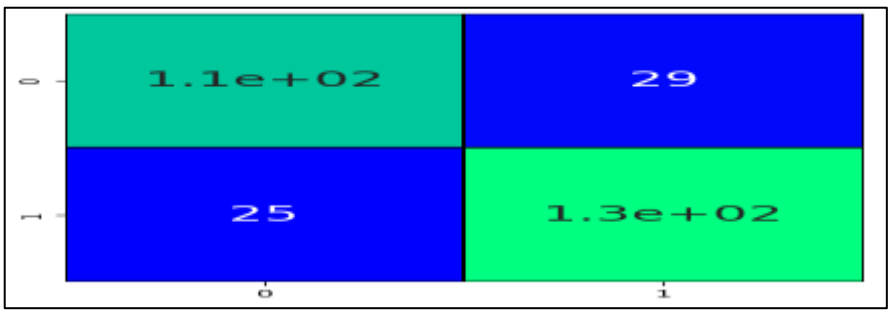

Fig. 25: Confusion matrix for the testing set

5.10 Different performance measurements using distinct classifiers such as Decision Tree, Naïve Bayes, and Support Vector Machine of the 'Heart Failure Clinical Records' heart dataset

All performance measurements of the 'Heart Failure Clinical Records' heart dataset are explained in table11.

Testing Accuracy%			**Testing Sensitivity%**			**Testing Specificity%**			**Testing Precision%**			**Execution Time (Sec)**		
Decision Tree	Naïve Bayes	SVM	Decision Tree	Naïve Bayes	SVM	Decision Tree	Naïve Bayes	SVM	Decision Tree	Naïve Bayes	SVM	Decision Tree	Naïve Bayes	SVM
92%	81.67 %	82.67 %	95.65 %	80.85 %	80.70%	86.26 %	84.625%	88.89%	91.67 %	95%	95.83%	18.163	14.005	536.54

Table 11: All performance measurements of the 'Heart Failure Clinical Records' heart dataset

5.11 Different performance of the 'Heart Failure Clinical Records' heart dataset

The following graphical illustrations are introduced in fig.26.

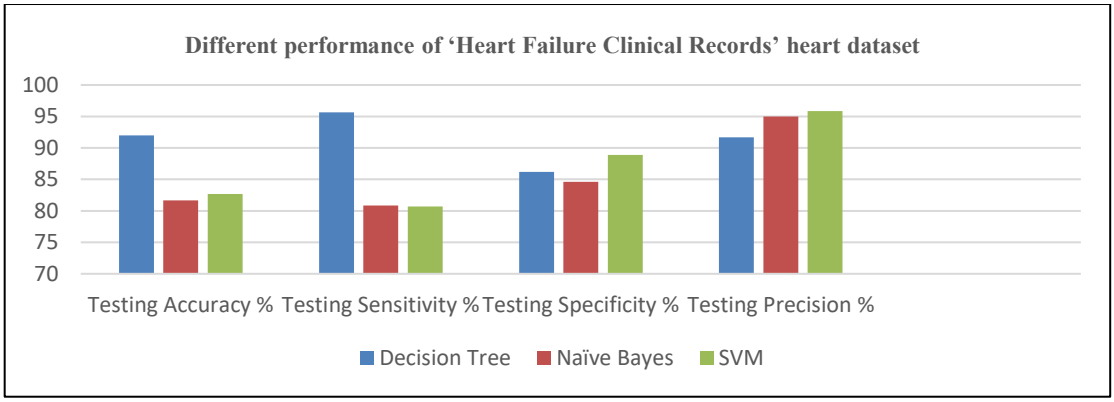

Fig. 26: Accuracy%, Sensitivity%, Specificity%, and Precision% in Heart Failure Clinical Record Heart Dataset

Observation from table11 and fig26.

Table 11 and Fig. 26 show that the 'Heart Failure Clinical Record' dataset achieves the best testing accuracy (92%) in the decision tree and the lowest (81.67%) in the Naïve Bayes classifier. Testing sensitivity is highest (95.65%) in decision trees and lowest (80.70%) in SVMs. Testing specificity is highest (88.89%) in SVM and lowest (84.615%) in the Naïve Bayes classifier. Similarly, the testing precision varies from max: 95.83% in SVM to min: 91.67% in decision trees. The Naïve Bayes classifier runs in 14.005 seconds, whereas the SVM classifier takes 536.535 seconds.

5.12 Different performance measurements using distinct classifiers such as Decision Tree, Naïve Bayes, and Support Vector Machine of the 'Heart Statlog Cleveland Hungary final' heart dataset

All performance measurements of the 'Heart Statlog Cleveland Hungary Final' heart dataset are explained in table12.

Testing Accuracy%			**Testing Sensitivity%**			**Testing Specificity%**			**Testing Precision%**			**Execution Time (Sec)**		
Decision Tree	Naïve Bayes	SVM	Decision Tree	Naïve Bayes	SVM	Decision Tree	Naïve Bayes	SVM	Decision Tree	Naïve Bayes	SVM	Decision Tree	Naïve Bayes	SVM
85.23 %	81.88 %	81.88 %	88%	81.02 %	81.48%	83.24 %	82.61%	82.21%	79.14 %	79.86 %	79.14%	29.09	19.40	110.58

Table 12: All performance measurements of the 'Heart Statlog Cleveland Hungary Final' heart dataset

5.13 Different performance of the 'Heart Statlog Cleveland Hungary final' heart dataset

Fig27 explain Accuracy%, Sensitivity%, Specificity%, and Precision% in the 'Heart Statlog Cleveland Hungary final' heart dataset

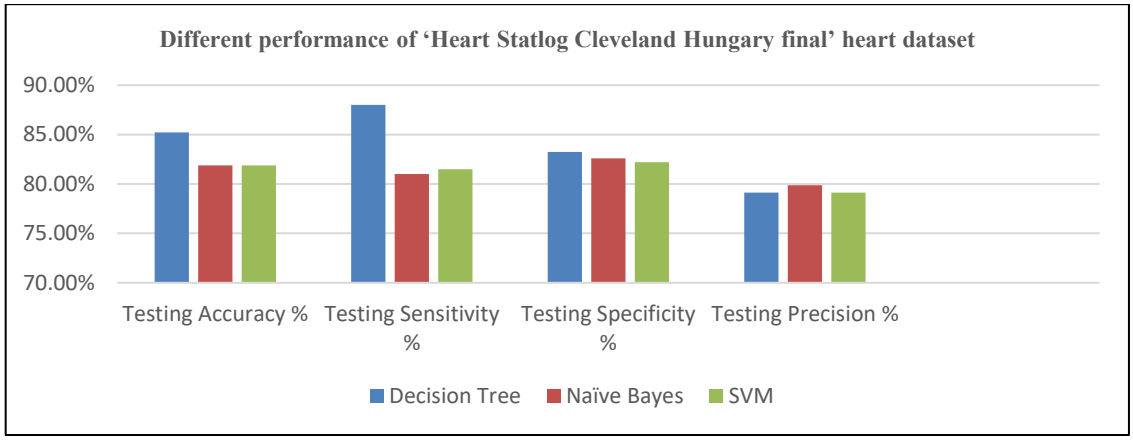

Fig. 27: Accuracy%, Sensitivity%, Specificity%, and Precision% in the 'Heart Statlog Cleveland Hungary final' heart dataset

Observation from table12 and fig27.

According to the experimental results, which are displayed in Table 12 and Fig. 27, the testing accuracy for the 'Heart Statlog Cleveland Hungary final' heart dataset is lowest (81.879%) in the SVM and Naïve Bayes classifiers and highest (85.23%) in the decision tree. The decision tree has the highest testing sensitivity (88%), while the Naïve Bayes classifier has the lowest (81.021%).

Decision trees have the highest testing specificity (83.236%), whereas SVMs have the lowest (82.280%). In a similar vein, the Naïve Bayes classifier has the highest testing precision (79.856%), while SVM and Decision Tree have the lowest (79.136%). Finally, the SVM had the longest execution time (110.5818 seconds), whereas the Naïve Bayes classifier had the shortest (19.4003 seconds).

5.14 Chi-Square (χ^2) test of Heart Failure Clinical Record heart dataset

Chi-square (χ^2) test results and corresponding p-values of the 'Heart Failure Clinical Record heart' dataset is explained in table 13.

Attribute Sl No	Feature	Chi2score	p-value
11	time	14.0001	0.0002
7	serum creatinine	3.4464	0.0634
0	age	2.2828	0.1308
4	diabetes	1.7989	0.1798
5	ejection fraction	1.0396	0.3079
1	anaemia	0.8486	0.3569
2	high blood pressure	0.2538	0.6144
8	serum sodium	0.2492	0.6177
6	platelets	0.0354	0.8508
10	smoking	0.0175	0.8948
9	gender	0.0066	0.9351
3	creatinine phosphokinase	0.0008	0.9769

Table13: Chi-Square (χ^2) test results and corresponding p-values of Heart Failure Clinical Record heart dataset

5.15. Chi-Square (χ^2) test of 'Heart Statlog Cleveland Hungary final' heart dataset

Chi-square (χ^2) test results and corresponding p-values of 'Heart Statlog Cleveland Hungary final' are explained in table 14.

Attribute Sl No	Feature	Chi2score	p-value
8	exercise angina	168.510099	1.565251e-38
5	blood sugar	43.768039	3.696933e-11
2	chest pain type	32.736310	1.055470e-08
1	gender	27.344827	1.702192e-07
10	ST-slope	23.174980	1.479093e-06

7	max heart rate	11.630786	6.486915e-04
9	old-peak	7.161644	7.447888e-03
0	age	5.619130	1.776547e-02
4	cholesterol	3.772219	5.211045e-02
6	ecg	3.375608	6.616816e-02
3	blood pressure	0.225386	6.349661e-01

Table14: Chi-Square (χ^2) test results and corresponding p-values of 'Heart Statlog Cleveland Hungary final' heart dataset

Observation from table13 and table 14

Features that are highly correlated with the existence of heart disease are listed below in descending order using the chi-square test.

In descending order (highest precedence to lowest precedence), the features of the 'Heart Failure Clinical Record' cardiac dataset are:

time-> serum creatinine -> age -> diabetes -> ejection fraction -> anaemia -> high blood pressure -> serum sodium -> platelets-> smoking -> gender -> creatinine phosphokinase.

In descending order (highest precedence to lowest precedence), the features of the 'Heart Statlog Cleveland Hungary final' heart dataset are

exercise angina -> blood sugar -> chest pain type -> gender -> ST-Slope -> max heart rate -> old-peak -> age -> cholesterol -> ecg -> blood pressure.

6. Comparison Table

Performance Analysis	**Heart Failure Clinical Records Heart Dataset**			**Heart Statlog Cleveland Hungary final heart dataset**		
	Decision Tree	**Naïve Bayes Classification**	**Support Vector Machine**	**Decision Tree**	**Naïve Bayes Classification**	**Support Vector Machine**
Testing Accuracy%	92 %	81.67 %	82.67 %	85.23 %	81.879 %	81.879 %
Testing Sensitivity%	95.65 %	80.85 %	80.70 %	88 %	81.021 %	81.481 %
Testing Specificity%	86.20 %	84.615 %	88.89 %	83.236 %	82.608 %	82.208 %
Testing Precision%	91.67%	95 %	95.83 %	79.136 %	79.856 %	79.1367 %
Execution Time (Sec)	18.163 sec	14.005 sec	536.535 sec	29.09 sec	19.4003 sec	110.5818 sec

Table 15: Output Results Comparison

Figures (28, 29, 30, 31, 32, 33) and Table 15, respectively, compare a number of performance metrics between two heart datasets. These comparisons highlight significant differences in accuracy, sensitivity, and specificity, shedding light on the strengths and weaknesses of each dataset. Such insights are crucial for guiding future research and improving predictive models in cardiology. Accuracy percentage, sensitivity, specificity, precision percentage, and execution time in seconds are metrics for decision tree, Naïve Bayes, and SVM methods.

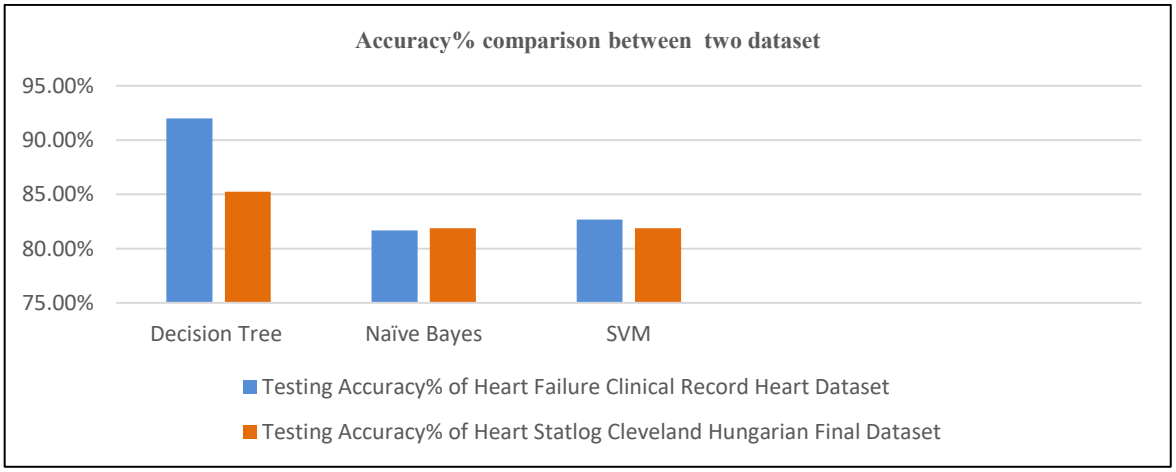

Fig. 28: Accuracy% comparison in Decision Tree, Naïve Bayes, and SVM between Two Datasets

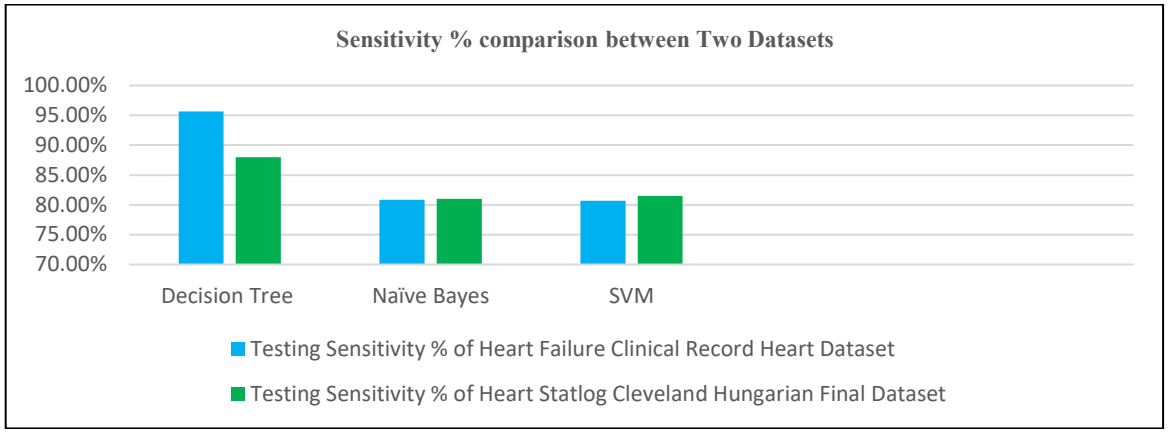

Fig. 29: Sensitivity % Comparison in Decision Tree, Naïve Bayes, and SVM between Two Datasets

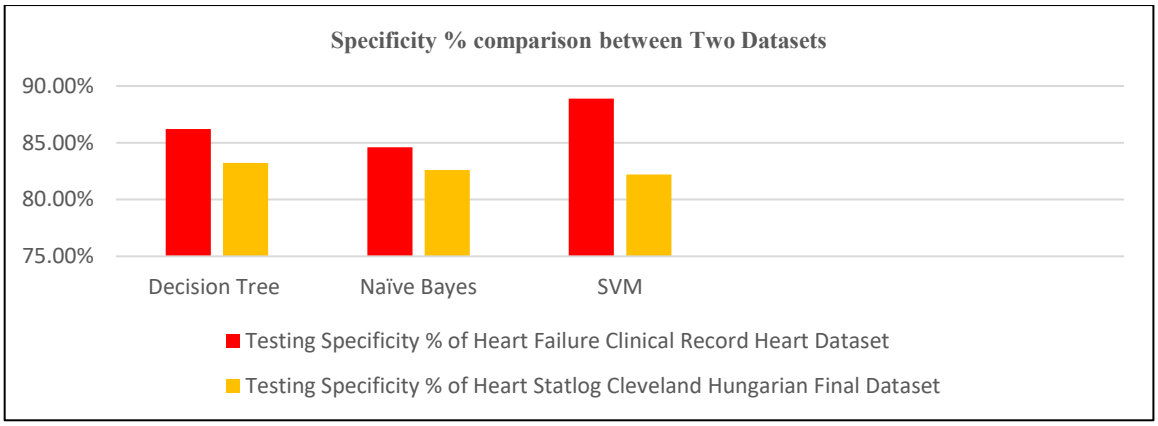

Fig. 30: Specificity% comparison in Decision Tree, Naïve Bayes, and SVM between Two Datasets

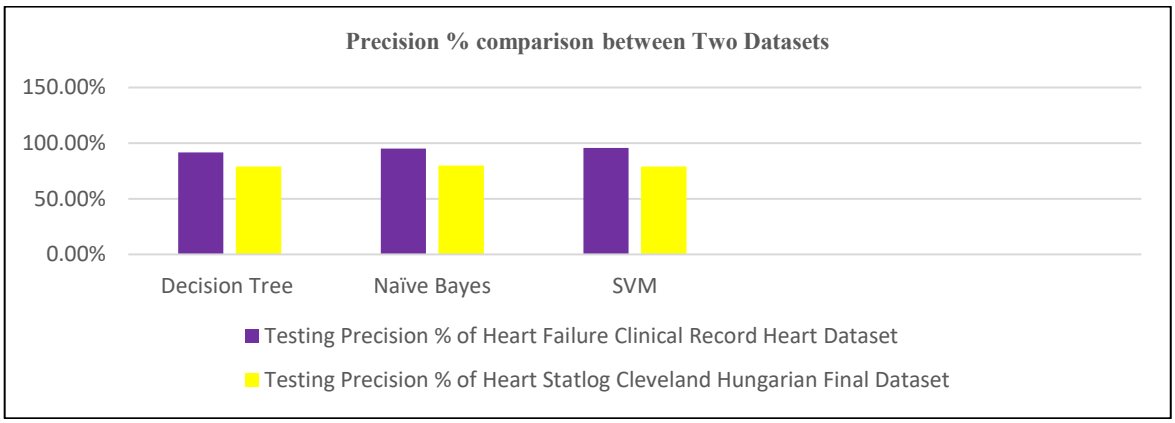

Fig. 31: Precision% comparison in Decision Tree, Naïve Bayes, and SVM between Two Datasets

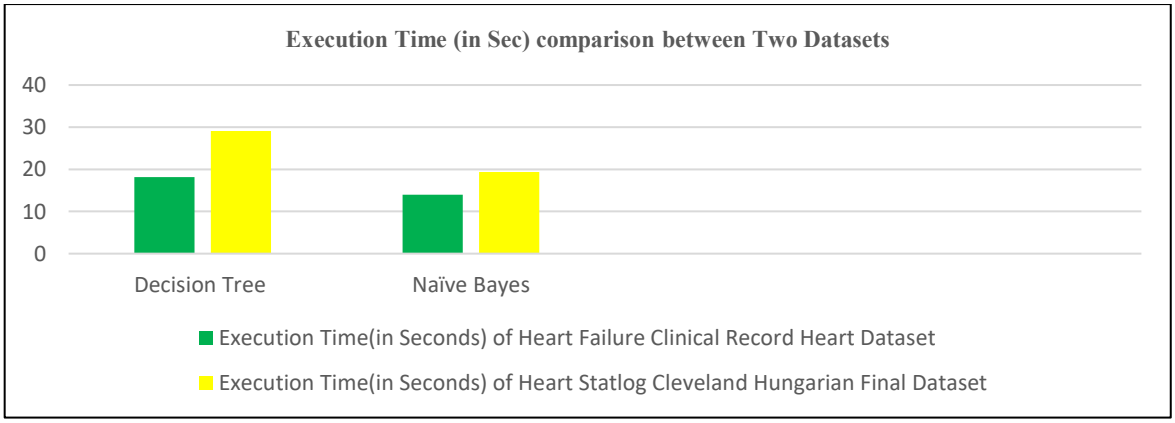

Fig. 32: Execution Time (in Sec) Comparison in Decision Tree, Naïve Bayes, and Two Datasets

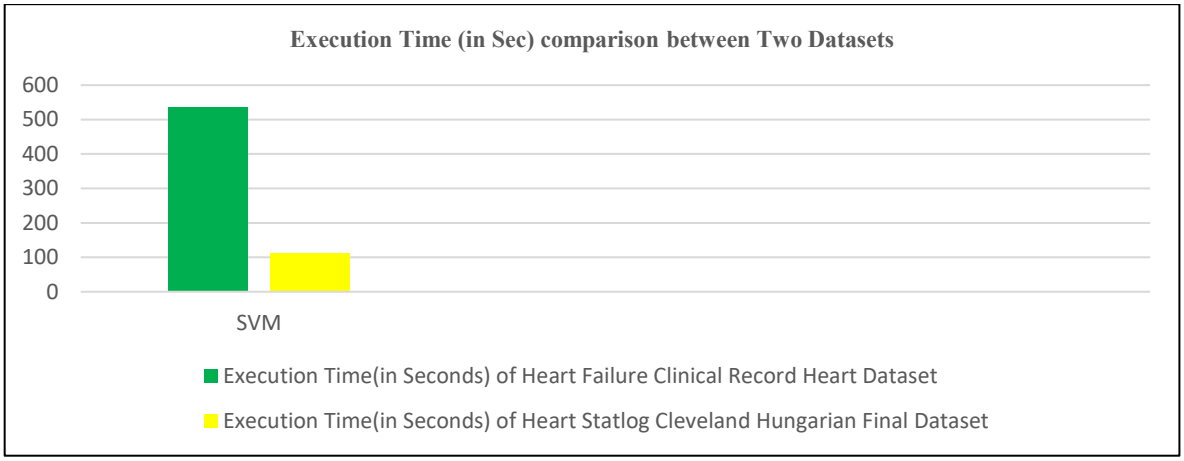

Fig. 33: Execution Time (in Sec) Comparison in SVM between Two Datasets

7. DISCUSSION

According to this study, utilizing two distinct datasets, three data mining algorithms—Decision Tree, Naïve Bayes, and Support Vector Machine—could reliably predict the prevalence of heart disease. A decision tree is a type of tree structure that looks like a flowchart. Each internal node in this instance represents an attribute test, while the topmost node represents the root node. Every branch stands for a test result. Every terminal node, sometimes referred to as a leaf node, has a class level. Leaf nodes are represented by 'OVALs,' whereas interior nodes are represented by 'RECTANGLES.' Let X be the given tuple in the decision tree that has an unknown class level. The attribute values of the tuples are compared with the decision tree. The class prediction for that specific pair of tuples is found at the leaf node, which is reached by tracing a path from the root node. Multidimensional data can be handled via decision trees.

The Naïve Bayes classifier is based on the Bayes theorem. This classifier method is predicated on conditional independence, which is the idea that an attribute value in a given class is independent of the values of other attributes. Bayesian networks are now one of the best classifiers available. These networks are composed of network-like structures with matching conditional probabilities. A Naive Bayesian model's simplicity substantially benefits large datasets because it does away with the necessity for intricate iterative parameter estimates. The Naive Bayesian classifier is widely used because it often performs surprisingly well for its simplicity and can outperform more sophisticated classification methods. The classifier is statistical in nature and does not assume that its attributes are dependent on one another. Since this approach uses conditional independence, it is predicated on the idea that an attribute's value inside a certain class is independent of the values of other attributes.

Support vector machines (SVM) are a supervised machine learning technique for both classification and regression problems. In an N-dimensional space, the SVM approach searches for a hyperplane that clearly classifies the data points. The SVM kernel's ability to transform low-dimensional input space into higher-dimensional space makes it useful for solving non-linear separable issues. SVM is used to choose extreme points or vectors that help create the hyperplane, also known as support vectors. The concept of structural risk minimization, which includes the capacity to manage the overfitting problem, is extended by SVM. This method not only enhances the model's generalization capabilities but also ensures that it remains robust against variations in the data. Consequently, SVM has become a popular choice in various applications, including image recognition, text classification, and bioinformatics, where high-dimensional data is prevalent.

According to the comparison table (Table 15) above, the 'Heart Failure Clinical Record' heart dataset had the highest testing accuracy (92%), testing sensitivity (95.65%), testing specificity (86.20%), and testing precision (91.67%) for the Decision Tree classifier with an execution time of 18.163 sec. Similarly, the 'Heart Statlog Cleveland Hungarian Final' heart dataset had the lowest testing accuracy (85.23%), testing sensitivity (88%), testing specificity (83.236%), and testing precision (79.136%) with an execution time of 29.09 sec.

The 'Heart Failure Clinical Record' heart dataset with an execution time of 14.005 seconds yielded minimum testing accuracy (81.879%), testing sensitivity (80.85%), maximum testing specificity (84.615%), and maximum testing precision (95%). Similarly, the 'Heart Statlog Cleveland Hungarian Final' heart dataset with an execution time of 19.4003 seconds yielded maximum testing accuracy (81.879%), testing sensitivity (81.021%), minimum testing specificity (82.608%), and minimum testing precision (79.856%).

Lastly, the 'Heart Failure Clinical Record' heart dataset with an execution time of 536.535 seconds yielded maximum testing accuracy (82.67%), testing specificity (88.89%), testing precision (95.83%), and minimum testing sensitivity (80.70%) for the SVM classifier, while the 'Heart Statlog Cleveland Hungarian Final' heart dataset with an execution time of 110.5818 seconds yielded minimum testing accuracy (81.879%), testing specificity (82.280%), and maximum testing sensitivity (81.481%).

7.1 Significance of Chi-Square (χ^2) test of Heart disease prediction

Particularly in feature selection and exploratory data analysis, the Chi-Square (χ^2) test is critical for forecasting heart disease. Here is a concise, exam-friendly explanation of its importance.

Identifies Important Risk Factors
If the χ^2 value is high and the **p-value is low (< 0.05)**, it indicates that the feature is strongly related to heart disease.
The χ^2 test determines whether there is a noteworthy correlation between:
Both the goal variable (the presence or absence of cardiac disease) and categorical input variables (such as gender, smoking status, and type of chest discomfort)
When the p-value is low (< 0.05) and the χ^2 value is high, it suggests that the trait has a substantial correlation with heart disease.
Helps in Feature Selection
Features with high χ^2 values are deemed more significant in heart disease prediction models, whereas features that are irrelevant or weakly connected can be eliminated. This results in faster computation and less overfitting.
Validates Medical Assumptions
The following well-established medical theories are supported by the χ^2 test: the type of chest pain is linked to heart disease, exercise-induced angina is linked to cardiac risk, and the incidence of heart disease varies by gender. Statistical validation strengthens clinical trust in the model.
Improves Model Interpretability
By using the χ^2 test, the model becomes more visible, allowing researchers and doctors to comprehend the significance of particular features for medical decision-support systems.
Supports Classification Models
χ^2 is particularly helpful prior to using models such as decision trees, logistic regression, and Naïve Bayes. Statistically significant features are advantageous for these models.

8. Limitations and Challenges

Medical diagnosis is viewed as an important but complex task that calls for speed and precision. It would be very beneficial if the diagnosis could be automated. Clinical decisions are often based less on the abundance of information hidden in the database and more on the knowledge and experience of the doctor. This strategy leads to inadvertent biases, mistakes, and excessive medical expenses, all of which lower the standard of treatment given to patients. A knowledge-rich environment provided by data mining has the potential to greatly enhance clinical decision-making.

8.1 Decision Tree (DT): Limitations & Challenges in CVD Prediction

Limitations
Overfitting
Datasets with heart disease are easily over-fitted by decision trees, particularly when the dataset is limited or noisy (as is often the case with clinical data).
Instability
A completely different tree may result from minor adjustments to patient data, such as blood pressure or cholesterol levels.
Bias toward dominant features
Even if a feature is not clinically the most essential, it tends to dominate if it has more split points, such as continuous attributes like age.
Limited generalization
A tree may have trained on data from one hospital won't generalize well to other groups.
Challenges in CVD Prediction
Predicting CVD presents a number of challenges, such as managing missing clinical values (such as incomplete lab tests), capturing the intricate relationships between risk factors including age, diabetes, smoking, and blood pressure, and performance declines when there is a class imbalance (more non-disease instances than illness cases).

8.2 Naïve Bayes (NB): Limitations & Challenges in CVD Prediction

Limitations
Strong independence assumption
Naïve Bayes may overestimate the independence of features in CVD data, such as the relationship between obesity, blood pressure, and cholesterol.
Lower accuracy for complex relationships
When interactions between risk variables have a substantial impact on the outcome of a disease, it suffers.
Challenges in CVD Prediction
Correlated clinical aspects challenge the main premise, while continuous medical traits necessitate discretization or distribution modelling, potentially impacting outcomes. Finally, it performs less well when disease patterns are non-linear.

8.3 Support Vector Machine (SVM): Limitations & Challenges in CVD Prediction

Limitations
High computational cost
When working with large cardiovascular datasets, training gets slow.
Kernel selection complexity
Selecting the appropriate kernel (linear, RBF, or polynomial) has a significant impact on prediction performance.

Low interpretability
When making clinical decisions, SVM acts like a black box, which is problematic.
Sensitive to noise and outliers
Measurements in medical data are frequently noisy, which impairs SVM performance.
Challenges in CVD Prediction
It requires careful adjustment of the parameters (C, γ), and it is challenging to explain the forecasts to physicians. The effect is that when data is extremely unbalanced, performance may suffer.

9. CONCLUSION AND FUTURE WORK

SL.NO	Methods	Key Findings	References
1	Cardiovascular Disease Prediction using Decision Trees, Naïve Bayes, and SVM for the 'Heart Failure Clinical Record' heart dataset	92% (Decision Tree) 81.67% (Naïve Bayes) 82.67% (SVM)	In this study
2	Cardiovascular Disease Prediction using Decision Trees, Naïve Bayes, and SVM for the 'Heart Statlog Cleveland Hungarian' Final Heart Dataset	85.23% (Decision Tree) 81.879% (Naïve Bayes) 81.879% (SVM)	In this study
3	Different machine learning techniques exist, such as decision trees, XGBoost, random forests, and multilayer perceptron.	87.28%	C. M. Bhatt et al. [2023] [1]
4	Numerous machine learning techniques exist, such as KNN, DT, Random Forest, SVM, ANN, and logistic regression.	86.89% (best accuracy for Random Forest).	S.S. Dehia et al. [2023][4]
5	Random Forest, Naïve Bayes, Deep Learning Model, and Logistic Regression.	73.78% (maximum accuracy by using the deep learning model).	T.S. Eswar Reddy et al. [2022] [6]
6	The R programming language was used to supervise machine learning algorithms including SVM, KNN, and Naïve Bayes.	86.6% (best accuracy by using the Naïve Bayes algorithm).	S. Anitha et al. [2019] [9]
7	Several data mining methods, including Naïve Bayes, Random Forest, Decision Trees, SVM, Logistic Regression, and MLP classification.	86% (highest accuracy by using logistic regression)	C.S. Wu et al. [2019] [10]
8	Decision tree (J48) algorithm	68%	M. K. Iliyas et al. [2019] [12]
9	Machine learning classification model.	85%	A.K. Dwivedi [2018] [13]
10	A decision tree-based fuzzy medical diagnostic aid system.	63.24 %	O. Terrada et al. [2018] [15]
11	Techniques used in data mining classification include neural networks, KNNs, decision trees, and Naïve Bayes.	80.6% (maximum accuracy)	T. Princy, R. et al. [2016] [18]
12	Decision Tree (C 5.0) algorithm.	85.33%	M. Abdar [2015] [20]
13	Subtractive clustering methods	76.67%	L. Muflikhah et al. [2013] [26]
14	Hybrid techniques include decision trees, gain ratios, and the discretization of nine voting frequencies.	84.1%	M. Shouman et al. [2011] [29]

Table 16: Accuracy Comparison and Evaluation

9.1 The contribution of this work and the contrast of the suggested techniques with other techniques

This paper discusses the challenge of distilling and restricting the various algorithms utilized in medical prediction. Decision trees, Naive Bayes, and support vector machines are data mining techniques utilized to investigate the factors impacting heart disease.
We use the UCI machine learning repository's 'Heart Failure Clinical Records' and 'Heart Statlog Cleveland Hungary Final' heart datasets for cardiac data. The 'Heart Failure Clinical Records' cardiac dataset contains 299 instances (rows), 12 input attributes (such as age, anemia, creatinine phosphokinase, diabetes, ejection fraction, high blood pressure, platelets, serum creatinine, serum sodium, gender, smoking, and heart failure episodes), and 1 output attribute. Decision Tree, Naïve Bayes, and SVM classifiers yielded 92%, 81.67%, and 82.67% prediction accuracy, respectively, on the dataset.
With 1190 instances (rows), the Decision Tree, Naïve Bayes, and SVM classifiers produced 85.23%, 81.88%, and 81.879% prediction accuracy, respectively, for the 'Heart Statlog Cleveland Hungary Final' heart dataset. There were 11 input characteristics, including age, gender, blood pressure, cholesterol, blood sugar, heart rate, ECG, exercise-induced angina, old peak, and ST-slope, and one output attribute.
The aforementioned comparison evaluation table (Table 16) demonstrates that the suggested technique yields the best results for accurately predicting levels of cardiac disorder with the shortest processing time (18.163 sec) and the highest prediction accuracy (max 92%) when compared to other methods (hybrid or simple).
In decision trees, with specific attribute values are utilized to find the longest paths of all features in these two datasets and predict whether or not the cardiovascular disease exists.
Finally, using the chi-square test, traits that are strongly linked with the presence of heart disease are listed in descending order.
The features of the 'Heart Failure Clinical Record' cardiac dataset are listed below in descending order (highest precedence to lowest precedence):
Time -> serum creatinine -> age -> diabetes -> ejection fraction -> anemia -> high blood pressure -> sodium -> platelets -> smoking -> gender -> creatinine phosphokinase.

The features of the 'Heart Statlog Cleveland Hungary final' heart dataset are listed in descending order (highest precedence to lowest precedence):
Exercise angina -> blood sugar -> chest pain type -> gender -> ST-slope -> maximum heart rate -> old-peak -> age -> cholesterol -> ECG -> blood pressure.

9.2 Advantages of Decision Tree, Naïve Bayes and support vector machines in heart disease prediction

Advantages of Decision Tree in Heart Disease Prediction
Medical decision-support systems frequently use decision trees due to their interpretability.
Advantages:
Easy to understand and interpret
Physicians can observe the decision-making process in action (e.g., if cholesterol > threshold → high risk).
Handles both numerical and categorical data
Heart datasets, including information on age, blood pressure, cholesterol, and the type of chest discomfort, can benefit from this.

No need for data normalization
works effectively without scaling, which lowers the complexity of prepossessing.

Feature importance identification
It assists in identifying important risk factors, such as blood pressure, age, or ECG findings.
Fast training and prediction
It is appropriate for clinical decision support in real time.

Supports rule-based clinical reasoning
Medical guidelines and output rules are in good alignment. When model transparency is crucial in the healthcare industry, it is extremely beneficial.

Advantages of Naïve Bayes in Heart Disease Prediction
The Bayes theorem-based probabilistic classifier Naïve Bayes is useful for estimating medical risk.

Advantages:
Computationally efficient
Extremely quick training and prediction, even with big datasets on heart disease.

Works well with small datasets
Beneficial when patient data is scarce.

Handles missing values effectively

Frequently found in actual clinical data.

Provides probabilistic outputs
It generates the likelihood of heart disease (85% risk, for example), which helps with clinical judgment.

Robust to irrelevant features
Less impacted by redundant or noisy medical features.

Simple implementation
Integrating into healthcare systems is simple. Ideal for risk probability assessment and early screening.

Advantages of Support Vector Machine (SVM) in Heart Disease Prediction
A strong machine learning algorithm with a reputation for making accurate predictions is SVM.

Advantages:
High classification accuracy
Efficacious in differentiating between patients with and without diseases.

Handles high-dimensional data
It possesses a wide range of clinical features.

Effective for non-linear relationships
Complex patterns of cardiac illness are captured by kernel functions (polynomial, RBF).

Strong generalization ability
Particularly in small-to-medium datasets, it reduces overfitting.

Robust to outliers
Reliability is increased by margin maximization.

Works well with imbalanced data (with tuning)
Crucial in medical datasets because the majority of instances are healthy. Perfect for predicting performance and precise diagnosis.

9.3 Future Work
To automate the prediction of cardiac disease, the suggested work must be improved and expanded. All methods must be compared, and clinical data from healthcare institutions and agencies must be acquired for optimal accuracy.

Diagrams and Figures Citation:
Reference [29] is given in Figure 2.
Citing the source [47] and providing the URL https://www.svm-tutorial.com/2015/06/svm-understanding-math-part-3/Figures 6, 7, 8, 9, and 10 are provided.
The citation for Fig. 11 comes from reference [44].

Both writers used the Python environment to create the remaining diagrams and figures based on the experimental data.

Table Citation:
For experimental purposes during their PhD studies, both authors created all of the tables.

REFERENCES
[1] Bhagawati M, Gupta S, Paul S, Mantella L, Johri AM, Laird JR, Tiwari E, Khanna NN, Nicolaides A, Singh R, Al-Maini M (2025) Attention-based hybrid deep learning models and its scientific validation for cardiovascular disease risk stratification. Biomedical Signal Processing and Control 108: 107824.
[2] Climente-González H, Oh M, Chajewska U, Hosseini R, Mukherjee S, Gan W, Traylor M, Hu S, Fatemifar G, Ghouse J, Del Villar PP (2025) Interpretable machine learning leverages proteomics to improve cardiovascular disease risk prediction and biomarker identification. Communications Medicine 5(1): 170.
[3] Zhang XR, Zhong WF, Liu RY, Huang JL, Fu JX, Gao J, Zhang PD, Liu D, Li ZH, He Y, Zhou H (2025) Improved prediction and risk stratification of major adverse cardiovascular events using an explainable machine learning approach combining plasma biomarkers and traditional risk factors. Cardiovascular Diabetology 24(1): 153.
[4] Amiri M, Mousavi M, Noroozzadeh M, Azizi F, Ramezani TF (2025) Cardiovascular disease risk prediction by Framingham risk score in women with polycystic ovary syndrome. Reproductive Biology and Endocrinology 23(1):19.

[5] Liu L, Zhang L, Zhang D, Guan T, He T, Liang B, Zhao J (2025) Risk prediction of cardiovascular events in peritoneal dialysis patients. BMC nephrology 26(1) :177.
[6] Zhang Y, Fahed AC (2025) Breaking binary in cardiovascular disease risk prediction. npj Cardiovascular Health 2(1):2.
[7] Panigrahi A, Pati A, Sahu B, Pati AK, Chowdhury S, Aurangzeb K, Javaid N, Aslam S (2025) Advanced ECG signal analysis for cardiovascular disease diagnosis using AVOA optimized ensembled deep transfer learning approaches. Computers, Materials & Continua 84(1).
[8] Wang S, Hu J, Du Y, Yuan X, Xie Z, Liang P (2025) WCFormer: An interpretable deep learning framework for heart sound signal analysis and automated diagnosis of cardiovascular diseases. Expert Systems with Applications 276: 127238.
[9] Cheng LH, Sun X, Elliot C, Condliffe R, Kiely DG, Alabed S, Swift AJ, van der Geest RJ, Kiely DG, Watson L, Armstrong I (2025) Mean pulmonary artery pressure prediction with explainable multi-view cardiovascular magnetic resonance cine series deep learning model. Journal of Cardiovascular Magnetic Resonance 27(1): 101133.
[10] GorapalliSrinivasa R, Muneeswari G (2026) Smart healthcare: A novel deep learning based Opt GPDCNN framework for heart disease prediction on the IoT platform. Biomedical Signal Processing and Control 112: 108594.
[11] Bhatt CM, Patel P, Ghetia T, Mazzeo PL (2023) Effective heart disease prediction using machine learning techniques, Algorithms 16 (2): 88–101. https://doi.org/10.3390/a16020088.
[12] Chandrasekhar N, Peddakrishna S (2023) Enhancing heart disease prediction accuracy through machine learning techniques and optimization, Processes 11(4): 1210–1240. https://doi.org/10.3390/pr11041210.
[13] Kadhim MA, Radhi AM (2023) Heart disease classification using optimized Machine learning algorithms, Iraqi Journal For Computer Science and Mathematics 4(2): 31-42. https://doi.org/10.52866/ijcsm.2023.02.02.004.
[14] Dahia SS, Szabo C (2023) Implementing Machine Learning to predict the 10-year risk of Cardiovascular Disease. Qeios. https://doi.org/10.32388/1SVUCI.
[15] Bakar WAWA, Josdi NLNB, Man MB, Zuhairi MAB (2023) A Review: Heart Disease Prediction in Machine Learning & Deep Learning. In 2023 19th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), IEEE 150-155.
[16] Reddy TSE, Sripathi SR, Akula D, Palaniswamy S, Subramani R (2022) Cardiovascular Disease Prediction using Machine Learning and Deep Learning. In 2022 6th International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), IEEE 1-5.
[17] Kanagarathinam K, Sankaran D, Manikandan R (2022) Machine learning-based risk prediction model for cardiovascular disease using a hybrid dataset, Data & Knowledge Engineering 140: 102042.
[18] Ali MM, Paul BK, Ahmed K, Bui FM, Quinn JMW, Moni MA (2021) Heart disease prediction using supervised machine learning algorithms: Performance analysis and comparison. Computers in Biology and Medicine 136:104672.
[19] Tasnim F, Habiba SU (2021) A comparative study on heart disease prediction using data mining techniques and feature selection. In 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), IEEE 338-341.
[20] Katarya R, Meena SK (2021) Machine learning techniques for heart disease prediction: a comparative study and analysis Health and Technology 11(1): 87-97.
[21] Hogo MA (2020) A proposed gender-based approach for diagnosis of the coronary artery disease. SN Applied Sciences. 2 :1-17.
[22] Anitha S, Sridevi N (2019) Heart disease prediction using data mining techniques. Journal of analysis and Computation 48-55.
[23] Wu CH, Badshah M, Bhagwat V (2019) Heart disease prediction using data mining techniques. In Proceedings of the 2019 2nd international conference on data science and information technology 7-11.
[24] Tarawneh M, Embarak O (2019) Hybrid approach for heart disease prediction using data mining techniques. In advances in internet, data and web technologies: the 7th international conference on emerging internet, data and web technologies (EIDWT-2019), Springer International Publishing 447-454.
[25] Iliyas MK, Shaikh IS (2019) Prediction of heart disease using decision tree. Allana Management Journal of Research, Pune 9:1-5.
[26] Dwivedi AK (2018) Performance evaluation of different machine learning techniques for prediction of heart disease. Neural Computing and Applications 29:685-693.
[27] Amin MS, Chiam YK, Varathan KD (2019) Identification of significant features and data mining techniques in predicting heart disease. Telematics and Informatics 36: 82-93.
[28] Terrada O, Cherradi B, Raihani A, Bouattane O (2018) A fuzzy medical diagnostic support system for cardiovascular diseases diagnosis using risk factors. In 2018 International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS), IEEE 1-6.
[29] Karthiga AS, Mary MS, Yogasini M (2017) Early prediction of heart disease using decision tree algorithm. International Journal of Advanced Research in Basic Engineering Sciences and Technology (IJARBEST) 3(3) : 1-17.
[30] Sultana M, Haider A, Uddin MS (2016) Analysis of data mining techniques for heart disease prediction. In 2016 3rd international conference on electrical engineering and information communication technology (ICEEICT), IEEE 1-5.
[31] Thomas J, Princy RT (2016) Human heart disease prediction system using data mining techniques. In 2016 international conference on circuit, power and computing technologies (ICCPCT), IEEE 1-5.
[32] Patel J, Upadhyay T, Patel S (2015) Heart disease prediction using machine learning and data mining technique, Heart Disease 7(1) :129-137.

[33] Abdar M (2015) Using decision trees in data mining for predicting factors influencing of heart disease. Carpathian Journal of Electronic and Computer Engineering 8(2): 31–36.
[34] Methaila A, Kansal P, Arya H, Kumar P (2014) Early heart disease prediction using data mining techniques. Computer Science & Information Technology Journal 24: 53-59.
[35] Thenmozhi K, Deepika P (2014) Heart disease prediction using classification with different decision tree techniques. International Journal of Engineering Research and General Science 2(6): 6-11.
[36] Chaurasia V, Pal S (2013) Early prediction of heart diseases using data mining techniques. Caribbean journal of Science and Technology 1: 208-217.
[37] jabbar MA, Deekshatulu BL, Chandra P (2013) Classification of heart disease using k-nearest neighbor and genetic algorithm, Procedia technology. 10: 85-94.
[38] Taneja A (2013) Heart disease prediction system using data mining techniques. Oriental Journal of Computer science and technology 6(4): 457-466.
[39] Muflikhah L, Wahyuningsih Y (2013) Fuzzy rule generation for diagnosis of coronary heart disease risk using substractive clustering method. Journal of Software Engineering and Applications 6: 372-378.
[40] Dangare CS, Apte SS (2012) Improved study of heart disease prediction system using data mining classification techniques. International Journal of Computer Applications 47(10): 44-48.
[41] 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): 43-48.
[42] Shouman M, Turner T, Stocker R (2011) Using Decision Tree for Diagnosing Heart Disease Patients. Proceedings of the 9-th Australasian Data Mining Conference (AusDM11) 23-30.
[43] Palaniappan S, Awang R (2008) Intelligent heart disease prediction system using data mining techniques. IJCSNS International Journal of Computer Science and Network Security. 8(8) :108-115.
[44] Chetwyn RA, Erdődi L (2021) Cheat detection in cyber security capture the flag games-an automated cyber threat hunting Approach. Proceedings of the 28th C&ESAR 3056: 175-190.
[45] Siddhartha M (2020) Cleveland Hungarian Statlog heart dataset. Multivariate data. https://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease.
[46] Chicco D (2020) Heart Failure Clinical Records dataset. BMC medical Informatics and decision making. https://archive.ics.uci.edu/ml/datasets/Heart+failure+clinical+records.
[47] Alexandre K (2015) SVM - Understanding the math. The optimal hyperplane. https://www.svm-tutorial.com/2015/06/svm-understanding-math-part-3