Research & Reviews: A Journal of
Health Professions
https://journals.stmjournals.com/rrjohp

ISSN: 2277-6192 (Online)
ISSN: 2348-9537 (Print)
Volume 15, Issue 3, 2025
September–December
DOI (Journal): 10.37591/RRJoHP

Review RRJoHP

AI-Powered ECG Prediction System for Detecting Cardiovascular Disease

Mohammad N. Alam[1,*], Vijay Laxmi[2], Baljinder Kaur[2]

Abstract

The proposed AI-powered CardioSmart Analyzer, an electrocardiogram (ECG) prediction system, presents an innovative and scientifically rigorous approach to the real-time automated analysis of ECG signals for diagnosing various heart conditions. This research focused on building a predictive model to identify cardiovascular diseases (CVD) using ECG data. A dataset comprising 2,840 12-lead ECG recordings was gathered from medical facilities in Gazipur, Bangladesh, over the period from June to August 2024. The analysis revealed 68 unique diagnostic categories based on the interpretations of the patients' ECG reports. To carry out the classification and prediction of these cardiovascular conditions, a robust random forest algorithm was implemented. This machine learning model proved to be highly effective, yielding outstanding performance results. In both binary and multi-class classification tasks, the algorithm achieved a remarkable accuracy rate of 100%. The success of this approach highlights its potential application in clinical settings, where automated ECG interpretation could assist healthcare professionals in the early and accurate diagnosis of a wide range of heart-related conditions. Overall, AI-driven ECG-based prediction model exhibited excellent performance in detecting common CVD conditions.

Keywords: ECG prediction, random forest, early detection, CVD condition, interpretation

INTRODUCTION

Healthcare diagnosis of cardiovascular diseases (CVD) relies heavily on electrocardiograms (ECG) because these tests show changes in heart function caused by electrical signals [1, 2]. Standard ECG reading techniques strain healthcare personnel due to their need for extensive amounts of time coupled with specialized medical skills which results in unscheduled test outcome timing. New real-time ECG evaluation technology develops from existing artificial intelligence (AI) and machine learning (ML) technologies in the Automated ECG Prediction System [3–5].

This paper outlines the development, design, and validation of the system, which employs Random Forest (RF) classifiers to analyze ECG signals and classify them into normal or abnormal readings. Additionally, the system's capability to perform multi-class classification allows for the interpretation of specific cardiovascular abnormalities. This innovation holds the potential to improve diagnostic efficiency and accuracy, ultimately enhancing patient outcomes for further treatment [6–8].

***Author for Correspondence**
Mohammad N. Alam
E-mail: mna235@yahoo.com

[1]Research Scholar, Department of CSE, Guru Kashi University, Bathinda, Punjab, India
[2]Assistant Professor, Department of Computer Applications, Guru Kashi University, Bathinda, Punjab, India

Received Date: May 27, 2025
Accepted Date: June 24, 2025
Published Date: November 04, 2025

Citation: Mohammad N. Alam, Vijay Laxmi, Baljinder Kaur. AI-Powered ECG Prediction System for Detecting Cardiovascular Disease. Research and Reviews: A Journal of Health Profession. 2025; 15(3): 51–85p.

Contributions

The aim of this research is outlined as follows:

- To gather and preprocess ECG signals, ensuring the data is prepared for effective analysis and model training.
- To train the predictive model utilizing the processed ECG signal data, leveraging advanced ML techniques.
- To classify CVD into binary and multi-label categories, enabling comprehensive diagnosis.

- To develop and implement a user-friendly graphical user interface (GUI) for the prediction and interpretation of ECG data, facilitating further clinical diagnosis and decision-making.

Organization

The rest of the paper is organized as follows: Section 2 details the materials and methodologies employed in the study. Section 3 discusses the outcomes of the predictive modeling process and includes a dedicated subsection that outlines the GUI developed to support diagnostic interpretation and clinical decision-making. Lastly, Section 4 offers concluding remarks and a future work direction.

MATERIALS AND METHODS

Data Sources

For this study, after obtaining approval from the authority, we collected the ECG reports of Bangladeshi patients who were analyzed to determine the nature of their heart disease at the Khawaja Badrudduja Modern Hospital in Gazipur, Bangladesh. Labels (y) are assigned to instances (x) in the dataset (D) where D = $\{(x_i,y_i)\}_{i=1}^{n}$, where n is the number of samples. After gathering the raw ECG signals, it is processed into a structured format, and we converted them into a CSV file.

ECG Signal

Standard 12-lead ECG traces sampled at 0.67–100 Hz, 25 mm/s, 10 mm/mV, and 2*5.0 s SE-1200 Express machine (Figure 1) were used to capture the ECG signal. The study involved 2840 subjects, 39,760 feature values, and 67 classes, with the data stored in image format. The year of the study was 2024. A sample of the ECG report is shown in Figure 2. The features included in it are gender, age, hr, p, pr, qrs, qt, qtcbz, p_pol, qrs_depol, t_repol, rv5, and sv1 are shown in Figure 3. These features' information is displayed in table I. The dataset D is loaded in tabular form into memory as a DataFrame (D = $x_1, x_2, \ldots, x_n\}$). If the target variable y is missing for any data instance, rows (R_i) are dropped (Tables 1–3):

$$D' = D - \{R_i \mid y_i = \emptyset\} \quad (1)$$

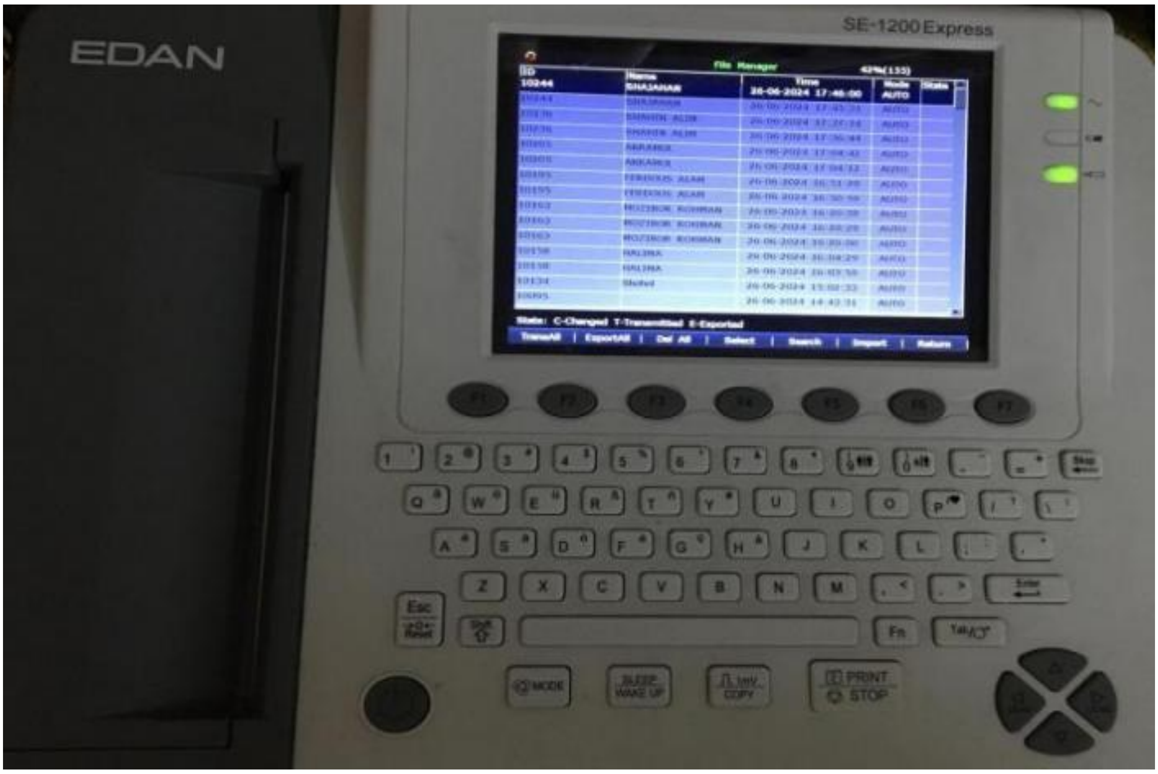


Figure 1. SE-1200 Express ECG machine was used to produce the report.

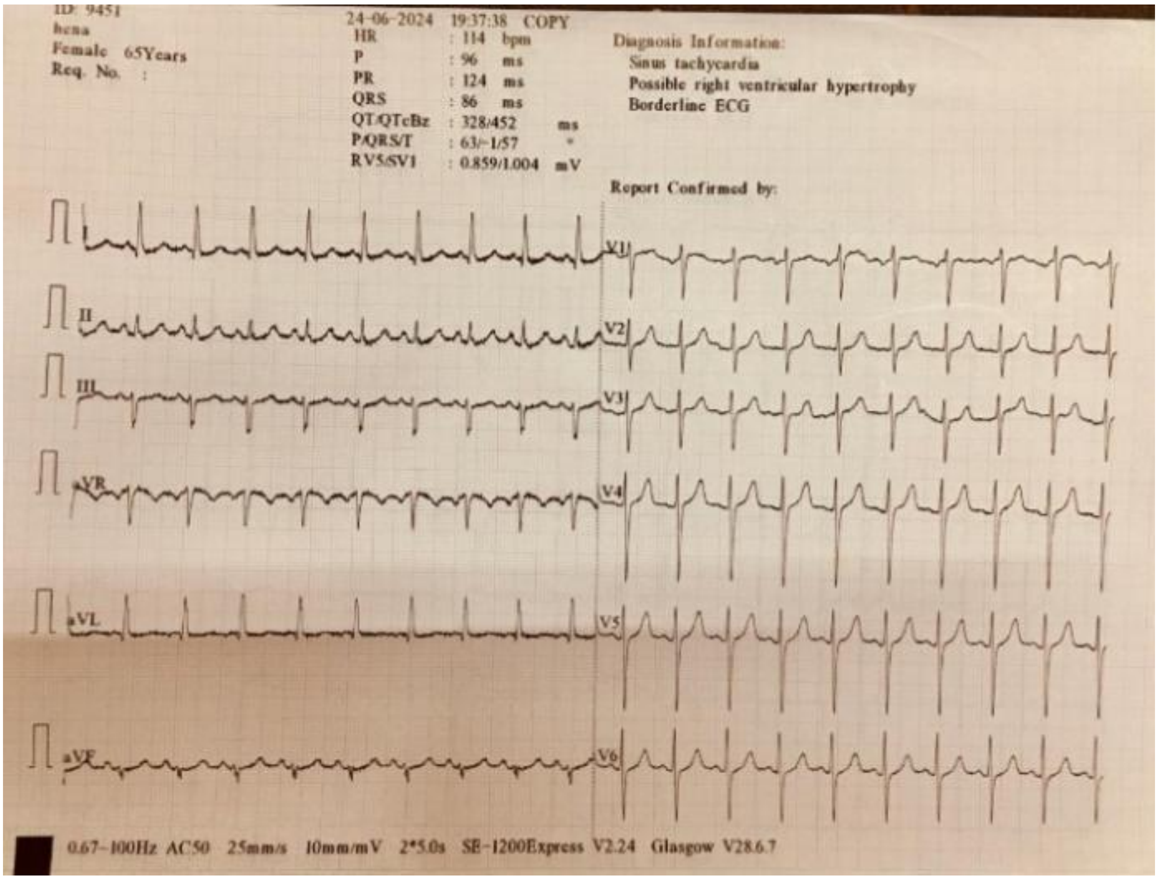


Figure 2. Sample of 12-lead ECG signal collected from hospital.

Classification of ECG Signal

ML techniques were employed to classify ECG signals based on CVD. This involves providing an algorithm with labeled training datasets, where key features are extracted to develop a predictive model. For the classification of ECG signal, we extracted features and corresponding class for each feature to train the model. This method allows the algorithm to accurately determine labels for test data. By generating models from sample data, supervised learning streamlines the decision-making process, making it highly suitable for diagnosing CVD using ECG signals. For training the datasets we conducted feature extraction and classification of CVD in the first stage [9, 10].

Feature Extraction

Clinically significant features, such as the P wave, QRS complex, T wave, heart rate, PR interval, and QT interval, are extracted from ECG report. Figure 3 shows the extracted features and Figure 4 shows the total no. of each feature extracted from ECG datasets. Three types of features are used for this experiment which are described in the following sub sections [11].

Demographic Features

- Gender.
- Age.

Morphological Features

These features describe the shape and amplitude of ECG waveforms, providing insights into structural abnormalities:

- *P (Amplitude/Duration):* Related to atrial depolarization.
- *PR (Interval):* Reflects the conduction time from atria to ventricles.

- *QRS (Duration/Amplitude):* Represents ventricular depolarization.
- *QT (Interval):* Total time of ventricular depolarization and repolarization.
- *QTcbz:* Corrected QT interval for heart rate.
- *P_pol (Polarity):* Characteristics of the P wave.
- *QRS_depol (Polarity/Amplitude):* Features of ventricular depolarization.
- *T_repol (Amplitude):* Represents ventricular repolarization.
- *RV5 (Amplitude):* The R-wave's amplitude in lead V5.
- *SV1 (Amplitude):* The S-wave's amplitude in lead V1.

Temporal Characteristics

These characteristics are crucial for comprehending the rhythm and conduction of the ECG signal since they characterize its temporal aspects.

- *PR (Time Interval):* The interval of time between the beginning of the P wave and the QRS complex.
- *QT (Time Interval):* Represents the total ventricular activity duration.
- *QRS (Duration):* Time taken for ventricular depolarization.

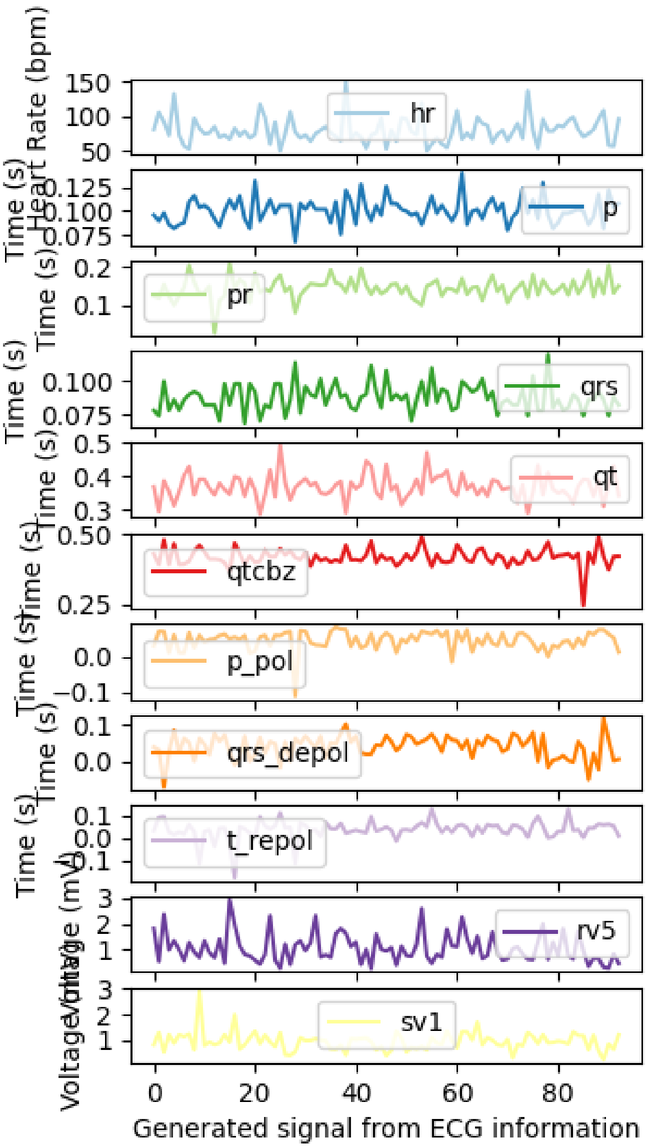


Figure 3. Feature extraction after preprocessing.

Research & Reviews: A Journal of Health Professions
Volume 15, Issue 3
ISSN: 2277-6192 (Online), ISSN: 2348-9537 (Print)

Table 1. ECG feature's description.

Feature	Mean	Std Dev	Distinct	Min	Max
Age	40.608	15.035	61	10	90
hr	81.893	16.334	74	21	150
p	91.887	12.059	36	8	144
pr	142.079	21.161	82	28	214
qrs	87.324	11.891	32	64	172
QT	366.016	45.613	84	144	664
QTCbz	425.552	25.83	102	246	500
p-pol	51.882	20.276	73	–110	117
qrs-depol	40.665	31.782	111	–77	160
t-repol	37.51	34.347	105	–178	154
TV5	1.333	0.583	311	0.089	3.414
SV1	0.919	0.374	277	0.051	2.927

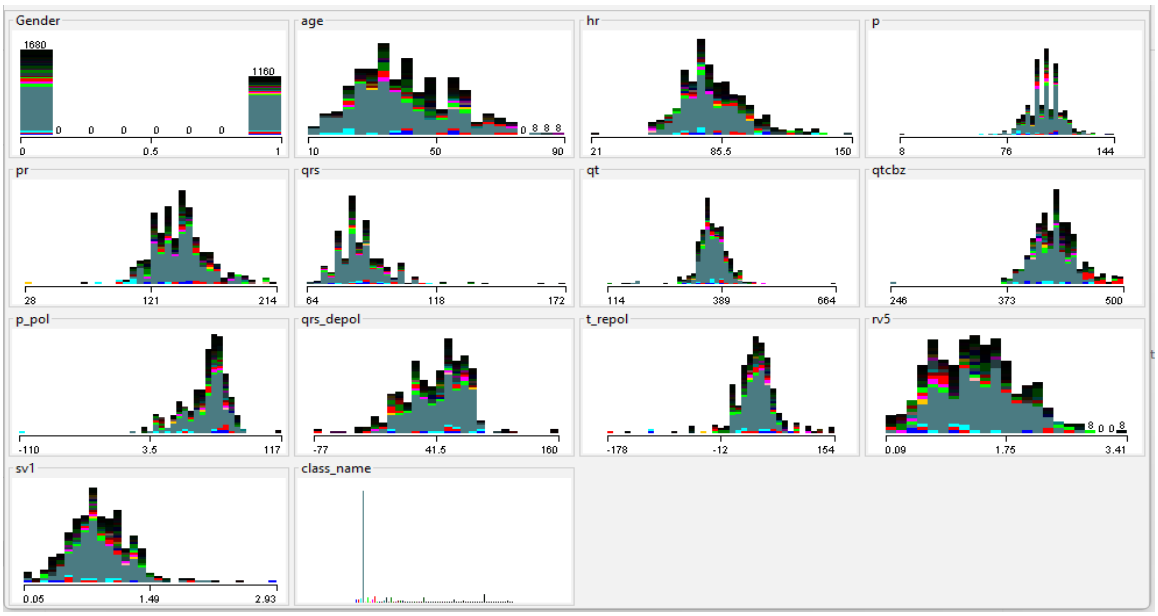


Figure 4. Total features and classes extracted from ECG datasets.

Table 2. Details of used features.

Features	Normal Range	Description
Age	0–120	Years since birth.
Heart Rate (HR)	60–100 BPM	Beats per minute of the heart.
P Wave Duration (P)	80–120 ms	Duration of atrial depolarization.
PR Interval (PR)	120–200 ms	Time for electrical impulse to travel from atria to ventricles.
QRS Duration (QRS)	70–100 ms	Duration of ventricular depolarization.
QT Interval (QT)	Varies, typically <440 ms (men), <460 ms (women)	Total time for ventricular depolarization and repolarization.
Corrected QT Interval (QTcBz)	<450 ms	Corrected QT interval supported on Bazett's formula.
T Wave Repolarization (T_repol)	<440 ms	Duration of ventricular repolarization.
RV5	0.5–1.5 mV	R wave in an ECG's lead V5.
SV1	0.5–1.5 mV	S wave in ECG's lead V1.

ML Model Development

The figure illustrates the step-by-step workflow of developing a ML model, starting from raw data collection to final model evaluation. The process begins with data collection, which involves sourcing the dataset, labeling, formatting, and converting it into a usable structure. The Data Loading step prepares the data in a machine-readable format, typically as a CSV file (Figure 5) [12].

Development of a Predictive and Interpretive Model

The figure shows a step-by-step process for building a predictive and interpretable ML model. Data is loaded, classes are identified, and labels are mapped for binary and multiclass classification. Models are then trained, evaluated, and used to predict conditions. Finally, automated interpretation provides actionable insights for decision-making (Figure 6) [13].

Table 3. List of ECG classifications denoting normal and abnormal conditions identified from the ECG signal.

Class	Condition
0	Normal ECG.
1	Possible left atrial abnormality.
2	Prolonged QT – consider ischemia.
3	Short PR interval.
4	Sinus bradycardia with borderline 1st degree A-V block.
5	Anterior T wave abnormality is nonspecific.
6	Possible right ventricular hypertrophy.
7	Poor R wave propagation.
8	Borderline prolonged QT interval.
9	Inferior T wave abnormality may be age and gender related.
10	Lateral ST-T abnormality due to myocardial ischemia.
11	Sinus tachycardia.
12	Possible anteroseptalinfarct – age undetermined.
13	Inferior T wave abnormality is nonspecific.
14	QRS changes due to LVH but cannot rule out anteroseptal infarct.
15	Inferior infarct – age undetermined.
16	Lateral ST abnormality is nonspecific.
17	Lateral ST elevation – possibly early repolarization.
18	Anteroseptal infarct – possibly acute.
19	Left anterior fascicular block.
20	Right axis deviation.
21	ST junction depression nonspecific.
22	Acute STEMI, Infarct, may be myocardial ischemia.
23	Small inferior Q waves: infarct cannot be excluded.
24	Inferior and anterior T wave abnormality due to myocardial ischemia.
25	Anteroseptal ST-T abnormality may be due to hypertrophy or ischemia.
26	Sinus bradycardia with sinus arrhythmia.
27	Anterior T wave abnormality is borderline due to age and gender.
28	Possible anterior infarct – age undetermined.
29	Septal T wave abnormality is nonspecific.
30	Lateral ST abnormality may be due to myocardial ischemia.
31	Inferior infarct – age undetermined, possible right ventricular hypertrophy.
32	Inferior/lateral ST-T abnormality due to myocardial ischemia.
33	Widespread ST-T abnormality is nonspecific.

34	Inferior ST-T abnormality is borderline for age and gender.
35	Sinus arrhythmia.
36	QRS changes V3/V4 may be due to LVH but cannot rule out inferior infarct.
37	Sinus rhythm with PVC, possible anterior infarct.
38	Leftward axis, left bundle branch block.
39	Inferior infarct, Anterior infarct.
40	Anterior T wave abnormality.
41	Sinus tachycardia, Anterior T wave abnormality is nonspecific.
42	Consider Acute STEMI, Extreme BRADYCARDIA.
43	Right atrial abnormality.
44	Leftward axis.
45	Incomplete RBBB.
46	Inferior ST-T abnormality is nonspecific.
47	Left ventricular hypertrophy.
48	Prolonged QT – consider ischemia, right axis deviation, Biventricular hypertrophy.
49	Sinus bradycardia.
50	Anterior infarct – age undetermined.
51	Inferior/lateral ST-T abnormality due to and/or ischemia.
52	Inferior ST abnormality is nonspecific.
53	Anterolateral ST-T abnormality may be due to hypertrophy and/or ischemia.
54	Normal ECG except for rate.
55	Inferior T wave abnormality is probably due to ventricular hypertrophy.
56	Septal ST abnormality may be due to hypertrophy and/or ischemia.
57	Suggests dextrocardia.
58	Right bundle branch block.
59	Inferior and anterior T wave abnormality is nonspecific.
60	Inferior/lateral ST-T abnormality is nonspecific.
61	Tall waves – consider acute ischemia or hyperischemia.
62	Anterior ST-T wave abnormality due to myocardial ischemia.
63	Possible left anterior fascicular block.
64	Septal and lateral ST-T abnormality may be due to myocardial ischemia.
65	Lateral ST elevation, consider acute infarct.
66	Inferior/lateral ST-T abnormality may be due to hypertrophy and/or ischemia.
67	Demand pacing.

To develop the predictive and interpretive model we use the dataset *D* consisting of *n* samples, where each sample is denoted as xi and belongs to a feature space X. The dataset has corresponding class labels yi from a label space Y, such that:

$$D = \{(xi,yi) \mid xi \in X,\ yi \in Y,\quad i = 1,2,\ldots,n\} \tag{2}$$

Step 1: Load Dataset

The dataset *D* is loaded from CSV. The dataset consists of multiple classes that need to be classified based on certain features.

$$D=\{(x1,y1),(x2,y2),\ldots,(xn,yn)\} \tag{3}$$

Step 2: Identify Classes

The distinct class labels present in D are identified as:

$$C = \{c1, c2, \ldots, ck\}, \text{ where } k = |C| \tag{4}$$

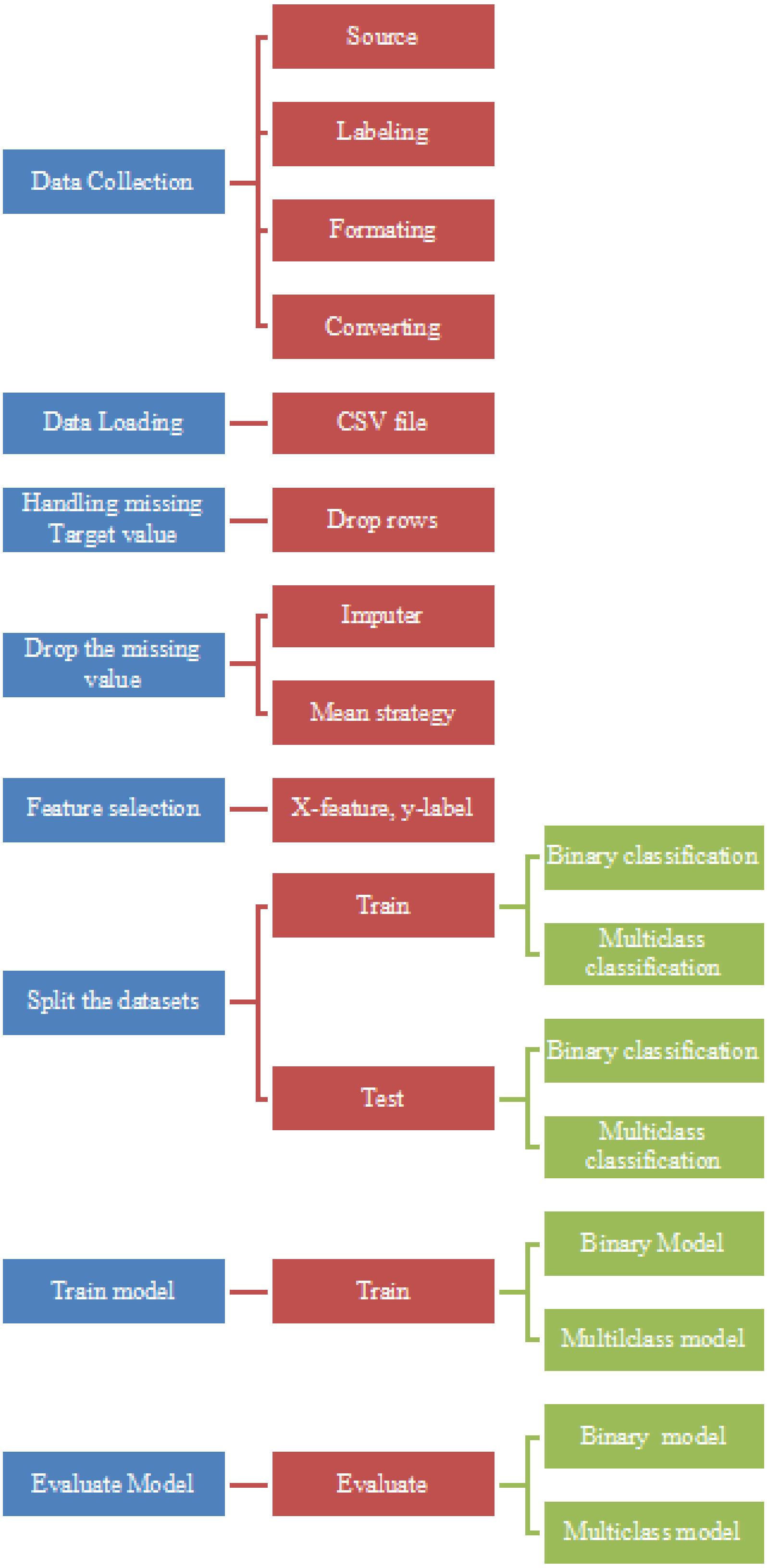


Figure 5. Steps for machine learning model development and evaluation.

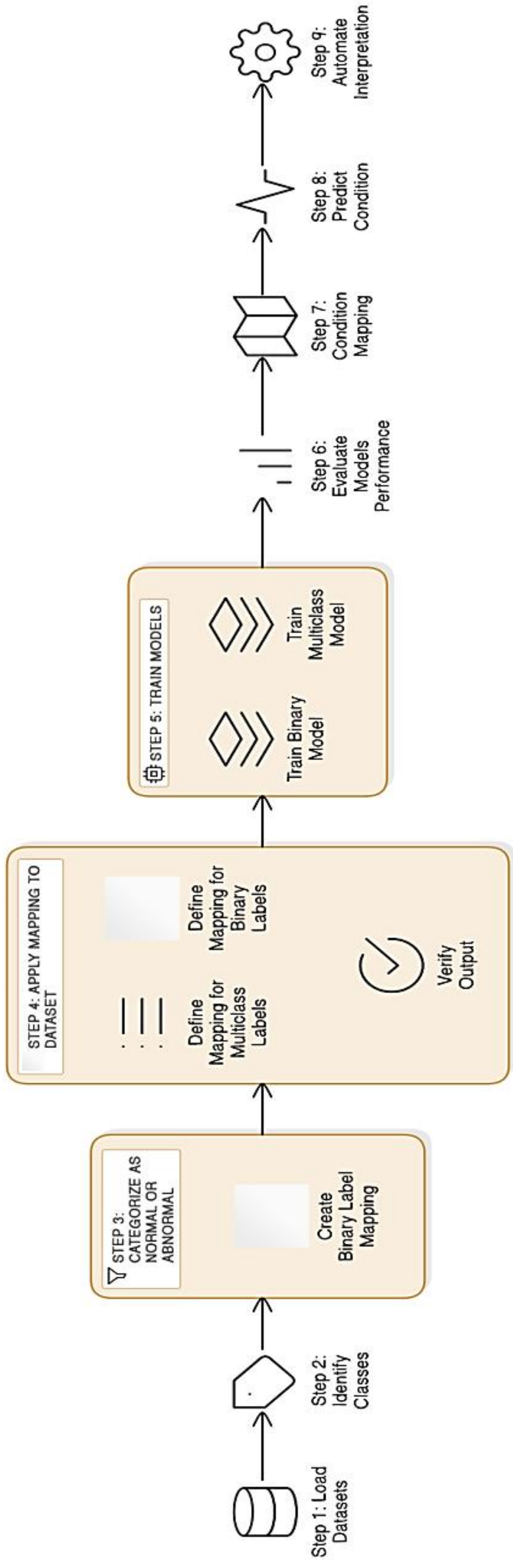


Figure 6. Steps for developing a prediction and interpretation model.

This step determines whether the classification task is binary (k = 2) or multi-class (k > 2).

Step 3: Categorization as Normal or Abnormal
A binary label mapping function *f : C → {0, 1}* is defined, where:

$$f(ci) = \begin{cases} 0, \text{if} c_i \text{iscategorizedas "Normal"} \\ 1, \text{if} c_i \text{iscategorizedas "Abnormal"} \end{cases} \tag{5}$$

This function transforms the original class labels into a binary classification scheme.

Step 4: Apply Mapping to Dataset
Two types of mapping are applied:

- *Binary mapping: yi′ = f(yi)* transforming all labels into the binary class set {0,1}.
- *Multi-class mapping:* If required, a multi-class classification approach is retained by preserving original class distinctions.

$$D' = \{(x1,y1'),(x2,y2'),\ldots,(xn,yn')\} \tag{6}$$

Verification is performed to ensure the correct assignment of transformed labels.

Step 5: Train Models
Two different ML models are trained.

- Binary Classification Model *MB:* A model trained on D′ using a classification function *h : X →* {0,1}.

$$MB = \text{argmax}_{\theta} L(h(X;\theta),Y') \tag{7}$$

where *L* is a loss function (e.g., cross-entropy loss).

- Multi-class Model *MM*: a multi-class classification model is trained to differentiate all original classes:

$$MM = \text{argmax}_{\theta} L(h(X;\theta),Y) \tag{8}$$

Step 6: Evaluate Model Performance
The trained models are evaluated using standard performance metrics such as Accuracy, precision, recall, F1-score.

Step 7: Condition Mapping
A set of rules is applied to map predicted outputs to meaningful conditions. This can be expressed as:

$$\widehat{y_i} = \begin{cases} Normal, if M_B(x_i)=0 \\ Abnormal, if M_B(x_i)=1 \end{cases} \tag{9}$$

For a multi-class approach:

$$\widehat{y_i} = \text{argmax}(MM(xi)) \tag{10}$$

Step 8: Automatic Interpretation
The final predictions are mapped to interpretable medical or domain-specific conclusions. This step ensures that model outputs are human-readable and provide meaningful insights for decision-making. However, the interpretation is for abnormal case where the conditions are available.

ECG Prediction and Interpretation

Figure 7 illustrates a decision pathway for predicting and interpreting ECG results. The process begins with an overall prediction to determine if the ECG is normal or abnormal. If the result is normal, a binary prediction model confirms the ECG as normal. If abnormal, a multiclass prediction model identifies the specific abnormality type, followed by interpretation of the condition.

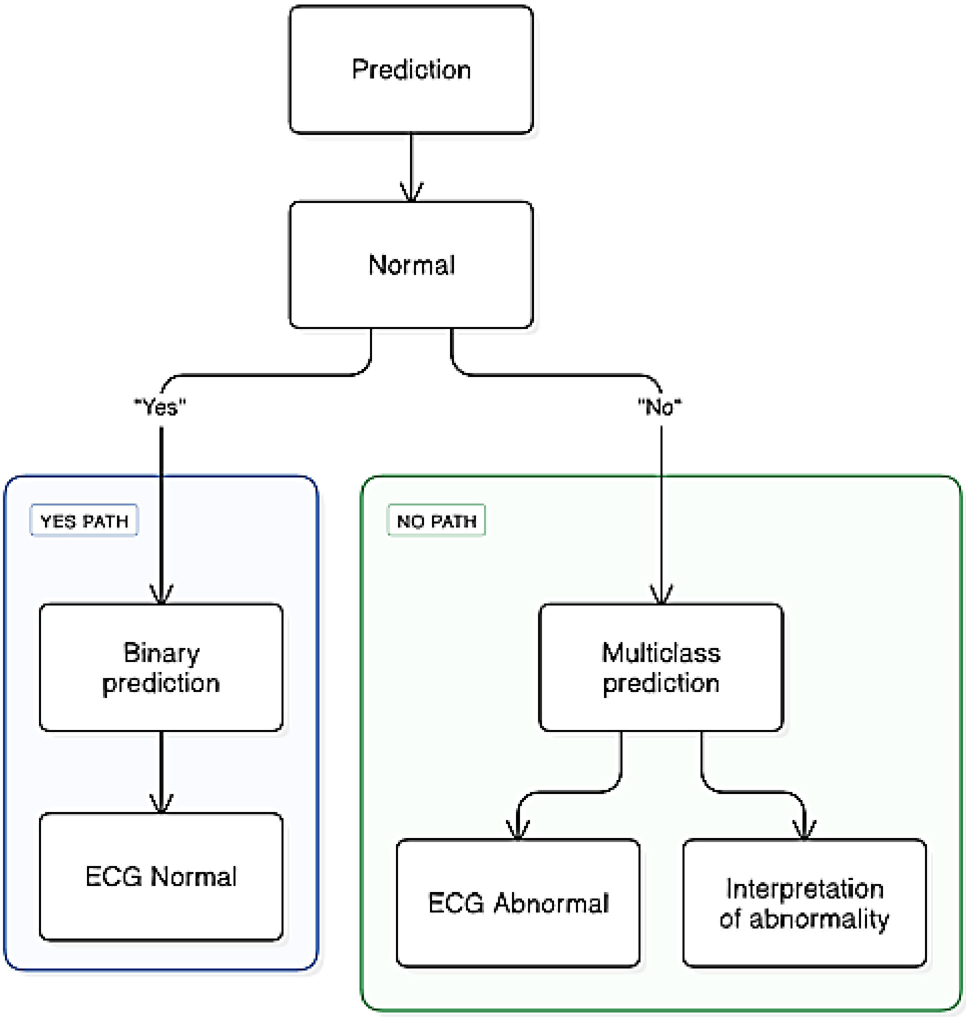


Figure 7. Logic of ECG prediction.

This approach ensures accurate classification and provides detailed insights into ECG abnormalities for better diagnosis and decision-making [14].

Proposed System Architecture and GUI

The Proposed CardioSmart Analyzer integrates a user-friendly interface, a robust backend, a ML model, efficient storage solutions, and a visualization component to classify ECG conditions accurately (Figures 8–12). The frontend provides fields for patient information (patient ID, name, and date) and ECG measurements [15], *X= {Gender, Age, HR, P, PR, QRS, QT, QT_CBZ, P_POL, QRS_DEPOL, T_REPOL, RV5, SV1}* with functionalities for prediction $f\theta(X) \rightarrow \{Normal, Abnormal\}$ saving results, visualizing data as fviz(X)→Graph, and clearing inputs. The backend ensures input validation, processes predictions using a base method Random Forest model fRF(X), and stores results in CSV retrieval. The RF classifier aggregates K decision trees (DTs) (multiple DTs) *(fk)* with:$fRF(X)$ = mode$\{f1(X), f2(X), \ldots, fK(X)\}$, trained to minimize impurity (L. Goldberger et al., 2000; L. Breiman et al., 2001; U. R. Acharya et al., 2017) [3, 4].

$$H = -\sum_{c=1}^{C} p_c \log(p_c) \tag{11}$$

where $C = 67$ represents the number of classes and pc is the probability of class c. For test sample X', the final class prediction is given by majority voting:

$$\hat{y} = \arg\max_{c} \sum_{t=1}^{T} I\big(h_t(X') = c\big) \tag{12}$$

where

- $\hat{y}$→ The predicted class label for X',
- $argmax_c$ → Returns the class that receives the highest number of votes.
- $ht(X')$→ The prediction made by the t-th classifier for input X'.
- $I(ht(X') = c)$ →An indicator function that returns:
- 1 if the classifier $ht(X')$ predicts class c
- 0 otherwise.

RF minimizes the classification error using:

$$L = \frac{1}{N} \sum^{N} I(\hat{y}_i \neq y_i) \tag{13}$$

where

- L → The total loss or classification error (objective function).
- N → The total number of samples in the dataset (number of ECG reports in our case).
- $\hat{y}_i$→ The predicted class label for the ith sample (output of the RF model).
- yi → The actual (true) class label for the ith sample.
- $I(\hat{y}_i \neq y_i)$ → Indicator function:
- If the predicted label $\hat{y}_i$does not match the actual label y_i, then it outputs 1 (misclassification). It means prediction label is incorrect.
- If $\hat{y}_i = y_i$, it outputs 0 (correct classification). It means prediction label is correct.

$\frac{1}{N}$ → Averages the total number of misclassified samples over all N samples, giving the classification error rate. In our case classification outputs include $y \in \{Normal, Abnormal\}$ with detailed interpretations for abnormal cases. Data visualization employs libraries, like Matplotlib, to generate ECG waveform plots, enabling intuitive data interpretation. This cohesive design combines robust ML, efficient data management, and an interactive user interface to deliver accurate predictions and seamless usability. We developed a desktop-based GUI using Tkinter library in python then created an exe file through pyinstaller. Therefore, we installed the pyinstaller first and then we followed the steps below to create the GUI exe file.

- Convert python script into executable file.
- Navigate the directory where python script is saved.
- Run the command in command prompt:
  - script>pyinstaller --onefile --windowed script_name.py
- Once the process is completed, the executable file will be saved in the "dist" folder created by pyinstaller. For example, C:\Users\MNA\AppData\Roaming\Python\Python39\Scripts\dist.

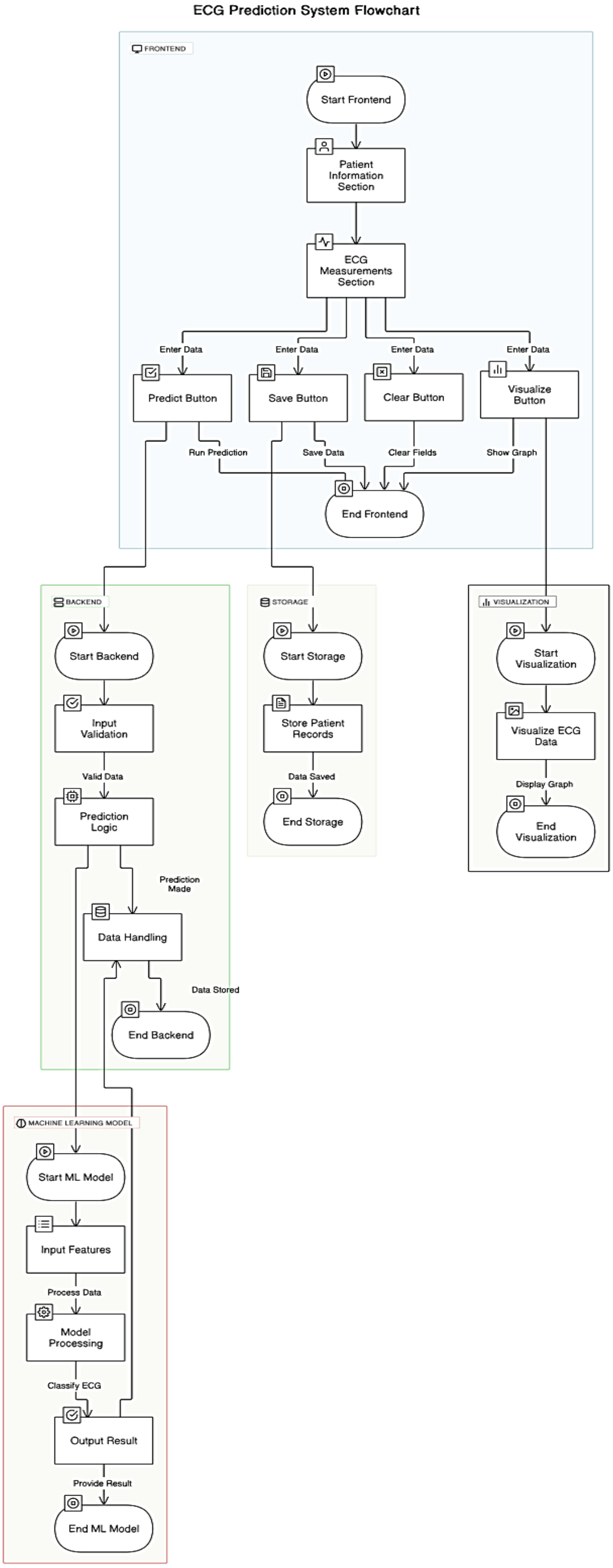


Figure 8. System architecture of ECG condition prediction and interpretation.

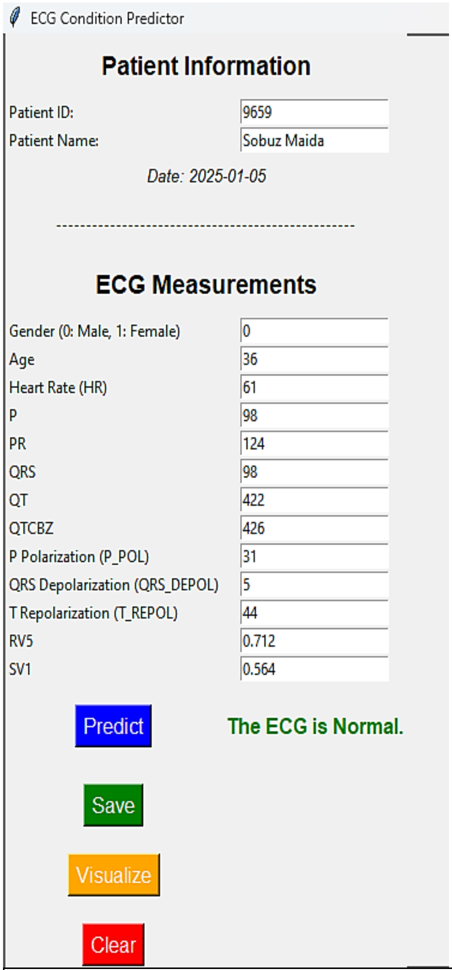


Figure 9. ECG predicted as normal based on patient's data interpretation.

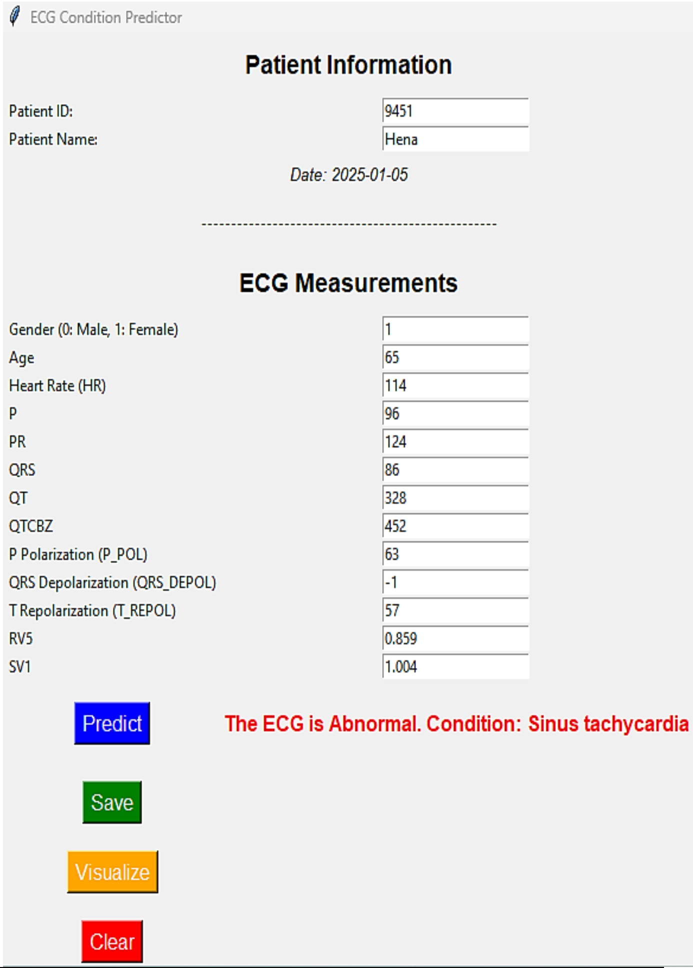


Figure 10. ECG predicted as abnormal based on patient's data interpretation.

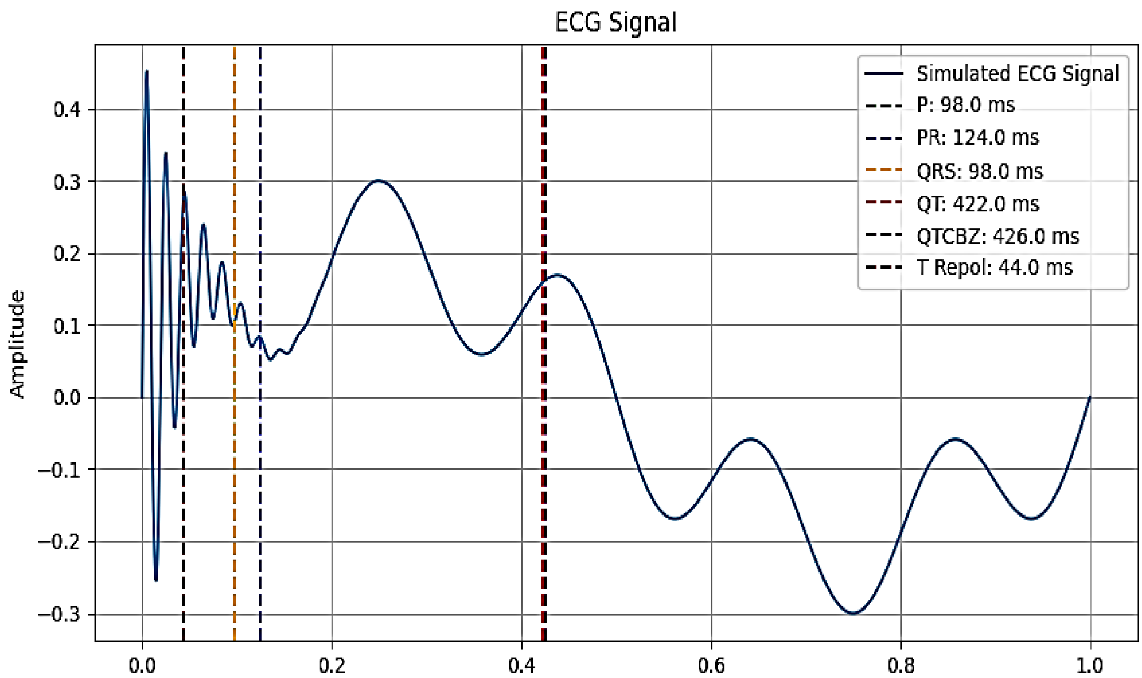


Figure 11. Visualization of the normal ECG signal.

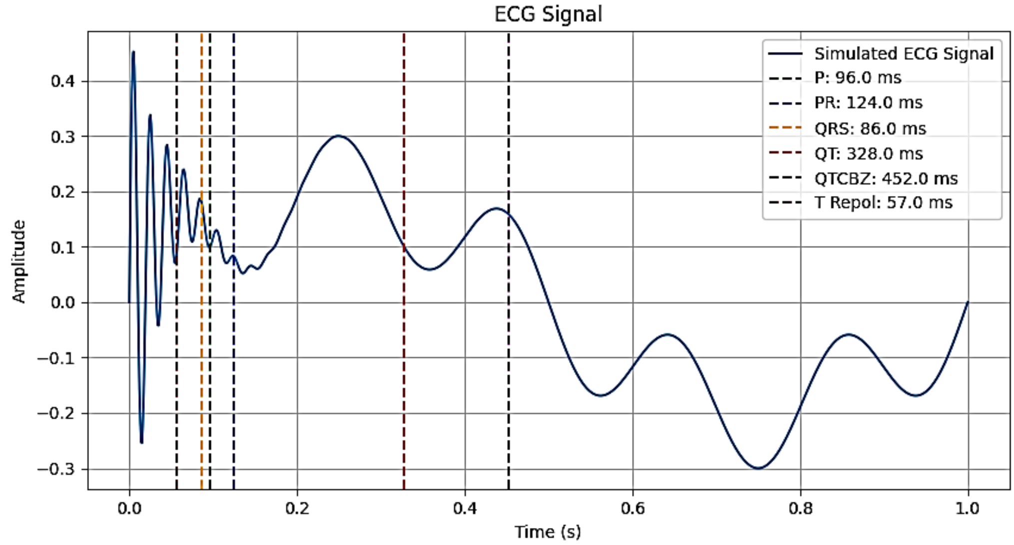


Figure 12. Visualization of the predicted abnormal condition: sinus tachycardia.

RESULTS AND DISCUSSION

Model Evaluation

The performance of each trained model is evaluated using appropriate metrics with datasets splitting criteria (80-20) and 70-30). Nevertheless, the comparative analysis of DT, Support Vector Machine (SVM), Logistic Regression, k-Nearest Neighbors (k-NN), and Gradient Boosting (GB) have been shown with RF. The evaluation method is given below:

- *Evaluation 1:* For binary classification (e.g., accuracy, precision, recall, F1-Score).
- *Evaluation 2:* For multiclass classification (e.g., accuracy, precision, recall, F1-Score).

Performance Evaluation

The system's performance was evaluated using several key metrics, including accuracy, precision, recall, F1-score, and AUC-ROC. Details are given in (Figures 13 and 14, Table 5).

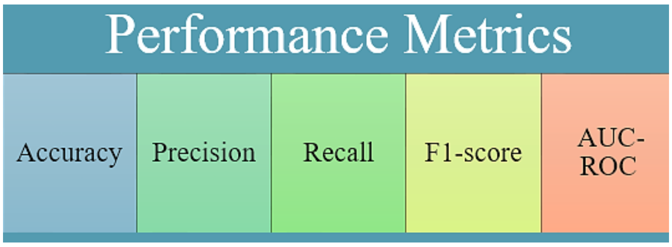


Figure 13. Performance evaluation metrics.

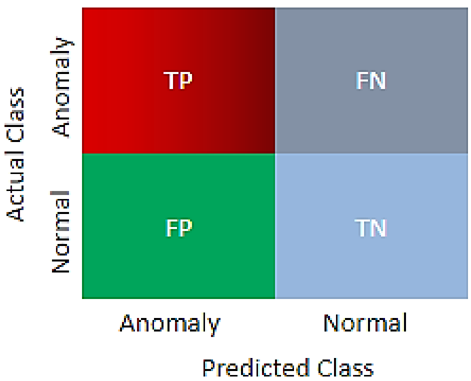


Figure 14. Confusion matrix.

Table 5. Performance metrics and formula.

Metrics	Definition	Formula
Accuracy	The proportion of cases that were accurately predicted to all instances. It measures general accuracy.	$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
Precision	The proportion of accurately forecasted positive observations to all of the predictions. It measures exactness.	$Precision = \frac{TP}{TP + FP}$
Recall (Sensitivity)	The proportion of all actual positive observations to accurately project positive observations. It measures completeness.	$Recall = \frac{TP}{TP+FN}$
F1-score	The harmonic mean of precision and recall is used to compute the F1 score, a classification performance statistic. It balances precision and recall.	$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$
AUC-ROC	AUC-ROC is a performance metric used to evaluate the classification ability of a model, especially for binary and multiclass classification problems.	Corrected AUC (Area Under ROC Curve) Formula $\mathrm{AUC} = \int_0^1 \mathrm{TPR(FPR)}\, d(\mathrm{FPR})$

where

- *True Positives (TPs):* are accurately anticipated positive cases.
- *True Negative (TN):* are negative situations that were accurately predicted
- False positives, or FPs, are Type I errors that are incorrectly expected to be positive.
- *False Negative (FN):* A Type II error in which the prediction is incorrectly negative

The representation of the binary and multi-class classification performance of CardioSmart Analyzer: The table summarizes the model's performance, presenting key evaluation metrics for both binary and multi-class classification tasks, demonstrating the accuracy and reliability of the CardioSmart Analyzer (Tables 6–9) [16].

Table 6. Details performance metrics of multiclass classification (80–20 split).

Model	Accuracy	Macro Avg (P, R, F1)	Weighted Avg (P, R, F1)
RF	0.99	0.98, 0.98, 0.98	0.99, 0.99, 0.99
DT	0.99	0.98, 0.98, 0.98	0.99, 0.99, 0.99
SVM	0.92	0.97, 0.91, 0.91	0.93, 0.92, 0.90
LR	0.71	0.67, 0.64, 0.64	0.64, 0.71, 0.64
KNN	1.00	0.98, 0.98, 0.98	0.99, 1.00, 0.99
GB	0.99	0.98, 0.98, 0.98	0.99, 0.99, 0.99

Table 7. Performance metrics of binary class (80–20 split).

Model	Accuracy	Precision (0,1)	Recall (0,1)	F1-Score (0,1)
RF	1.00	1.00, 1.00	1.00, 1.00	1.00, 1.00
DT	1.00	1.00, 1.00	1.00, 1.00	1.00, 1.00
SVM	0.68	0.66, 0.70	0.75, 0.60	0.70, 0.65
LR	0.67	0.66, 0.69	0.73, 0.61	0.69, 0.64
KNN	1.00	1.00, 1.00	1.00, 1.00	1.00, 1.00
GB	0.95	0.91, 1.00	1.00, 0.90	0.95, 0.95

Table 8. Details performance metrics of multiclass classification (70–30 split).

Model	Accuracy	Macro Avg (P, R, F1)	Weighted Avg (P, R, F1)
RF	1.00	0.98, 0.98, 0.98	0.99, 1.00, 0.99
DT	1.00	0.98, 0.98, 0.98	0.99, 1.00, 0.99
SVM	0.89	0.95, 0.89, 0.90	0.90, 0.89, 0.87
LR	0.70	0.69, 0.65, 0.65	0.60, 0.70, 0.63
KNN	1.00	0.98, 0.98, 0.98	0.99, 1.00, 0.99
GB	1.00	0.98, 0.98, 0.98	0.99, 1.00, 0.99

Table 9. Performance metrics of binary class (70–30 split).

Model	Accuracy	Precision (0,1)	Recall (0,1)	F1-Score (0,1)
RF	1.00	1.00, 1.00	1.00, 1.00	1.00, 1.00
DT	1.00	1.00, 1.00	1.00, 1.00	1.00, 1.00
SVM	0.89	0.95, 0.89	0.90, 0.87	0.90, 0.85
LR	0.67	0.66, 0.69	0.73, 0.61	0.69, 0.64
KNN	1.00	1.00, 1.00	1.00, 1.00	1.00, 1.00
GB	0.96	0.92, 1.00	1.00, 0.92	0.96, 0.96

- *Results after datasets splitting on 80% training set and 20% testing set:* The figure presents the outcomes of model development after splitting the dataset into 80% for training and 20% for testing. This division ensures that the model is trained on a substantial portion of the data while reserving a separate set for unbiased performance evaluation. The results highlight the model's ability to generalize effectively and provide reliable predictions (Figures 15 and 16) [17].
- *Results after datasets splitting on 70% training set and 30% testing set:* The figure illustrates model performance after dividing the dataset into 70% for training and 30% for testing, ensuring balanced learning and reliable evaluation of predictive accuracy (Figures 17 and 18) [18].
- *Confusion metrics after datasets splitting on 80% training set and 20% testing set:* The figure shows the confusion matrix summarizing the model's classification performance, highlighting correct and incorrect predictions on the testing set (Figure 19(a–l)) (Tables 10–12) [19].

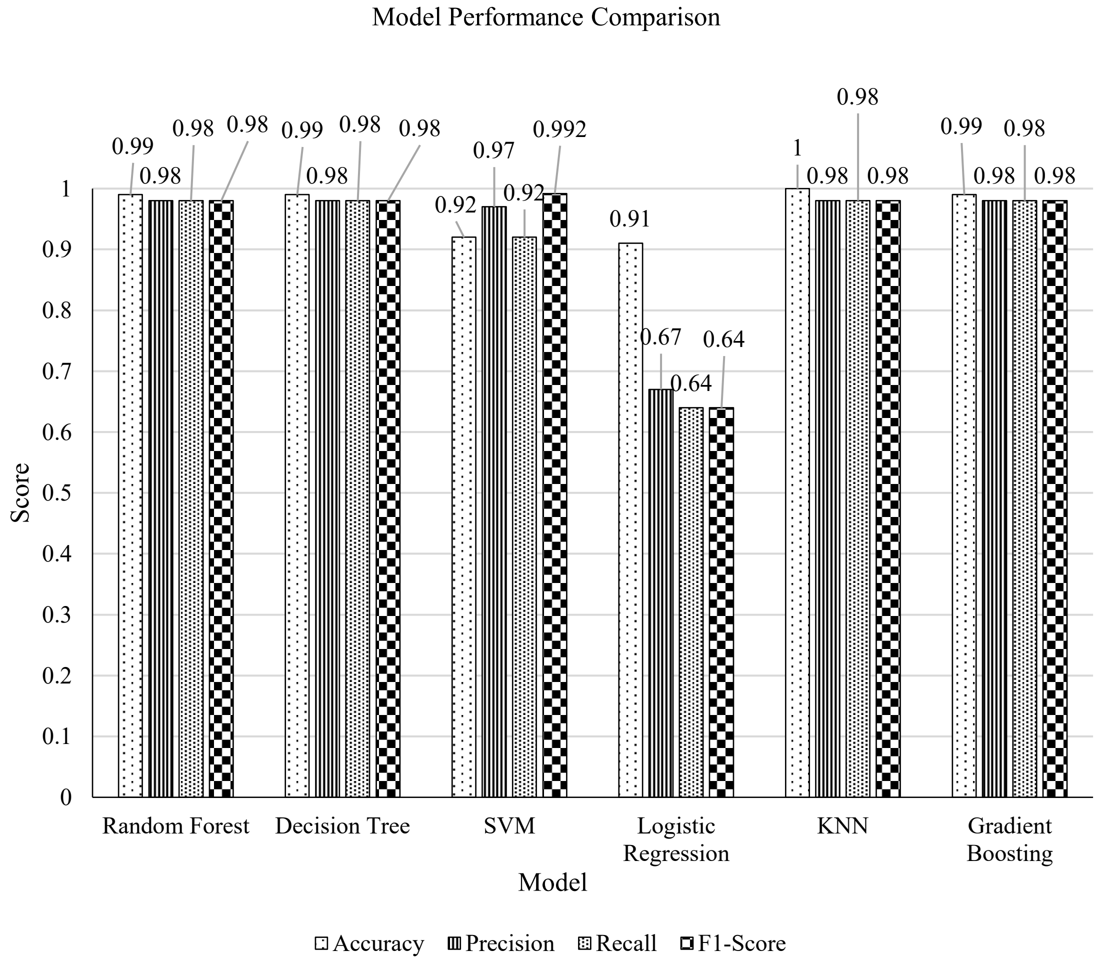


Figure 15. Comparison of multiclass classification model (split 80-20).

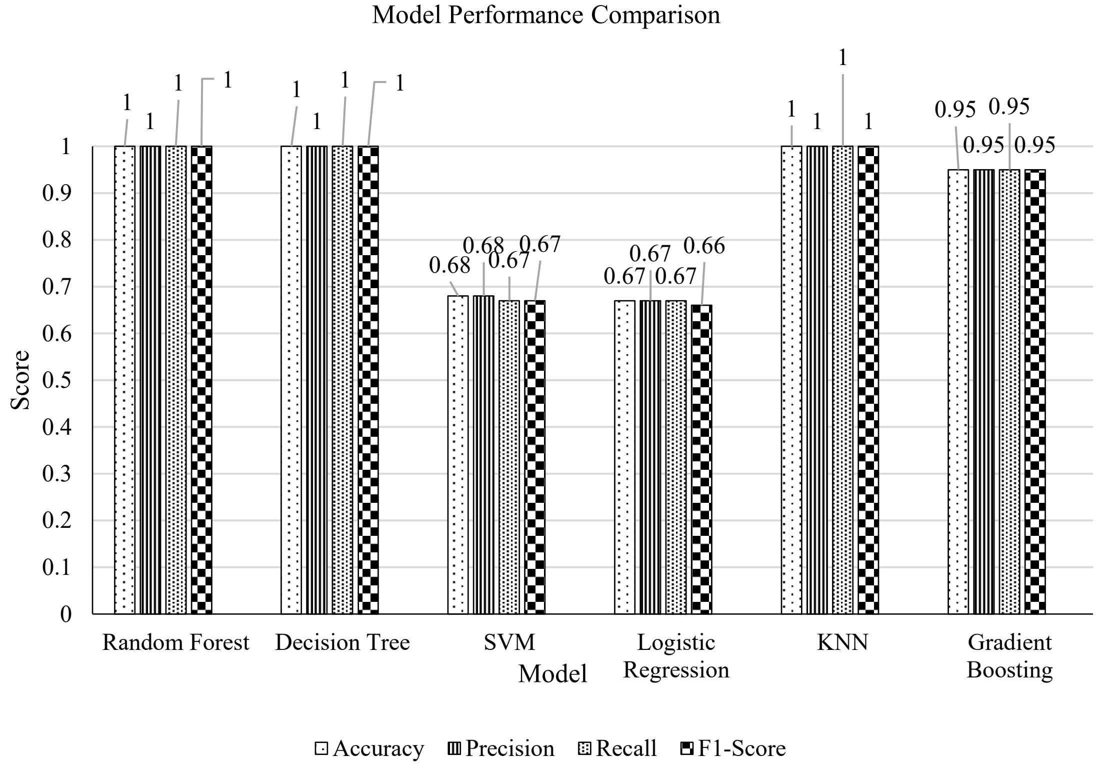


Figure 16. Comparison of binary classification model (split 80-20).

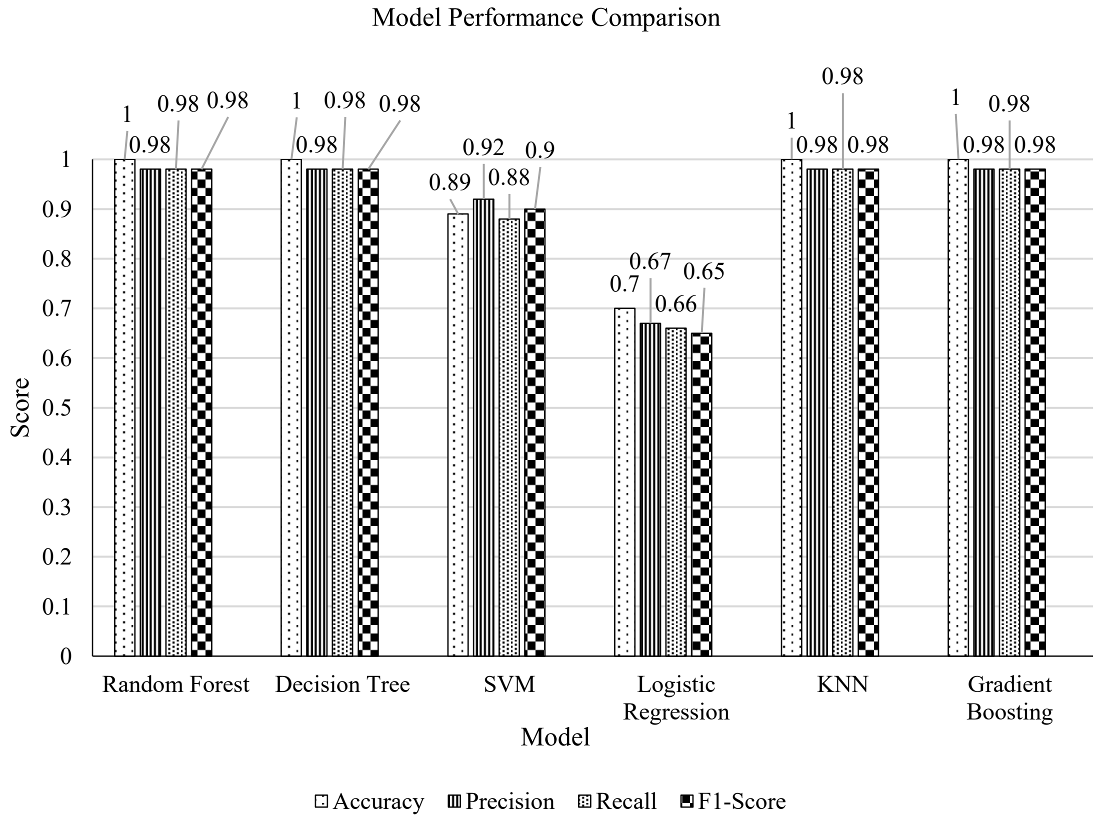


Figure17. Comparison of multiclass classification model (split 70-30).

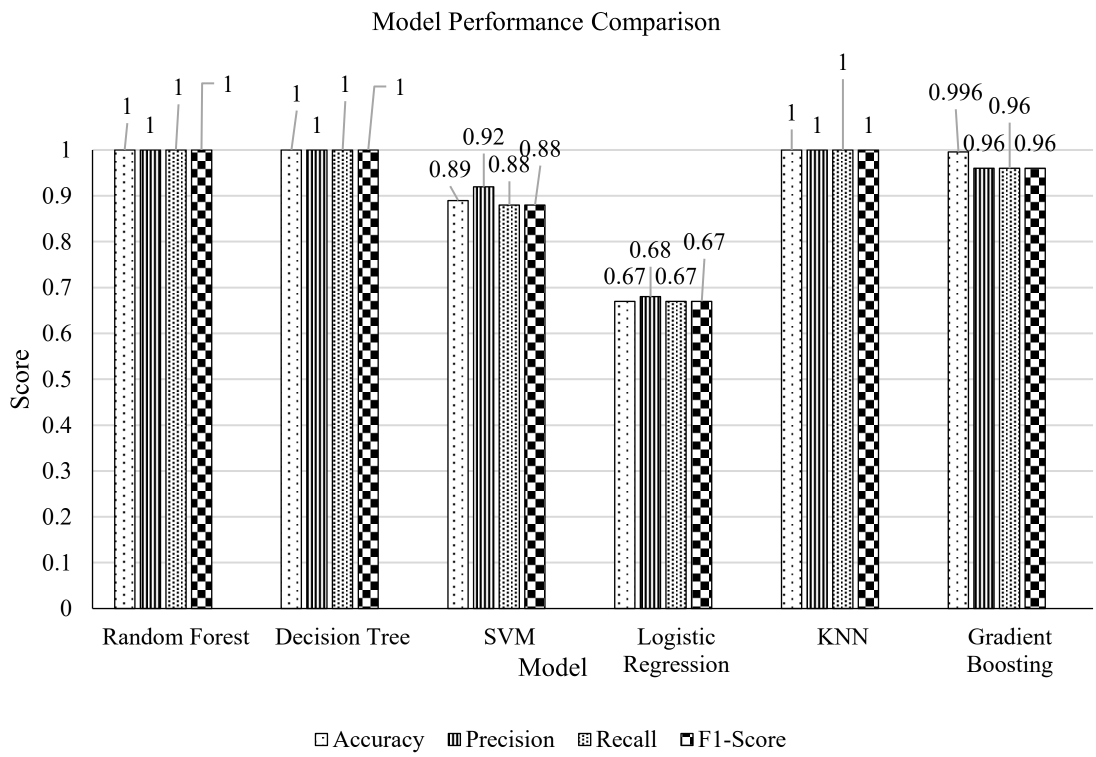
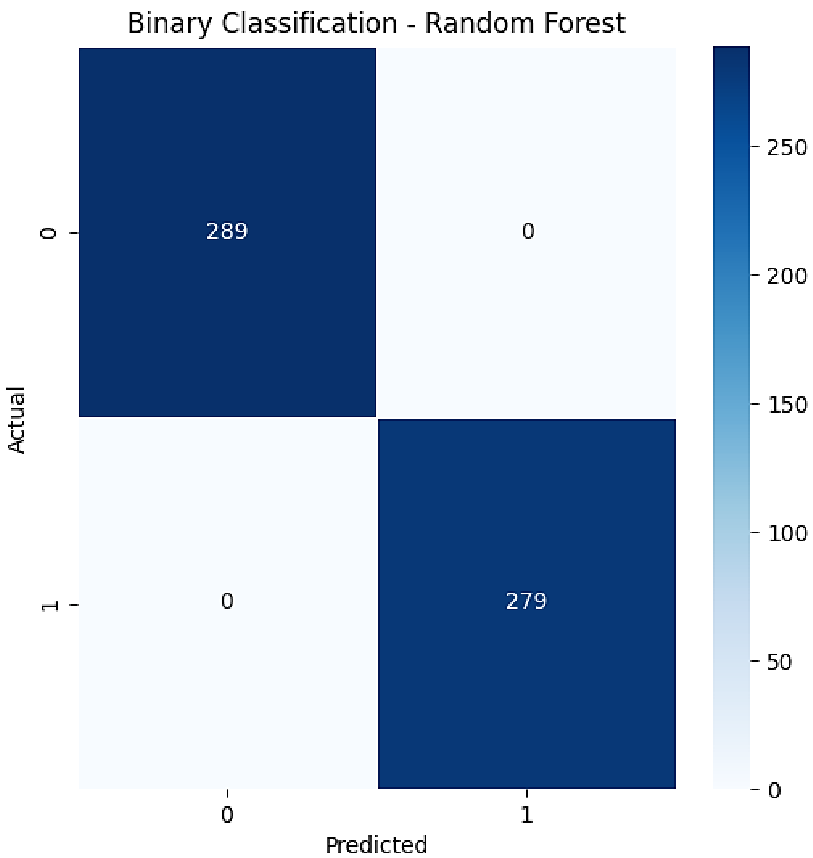
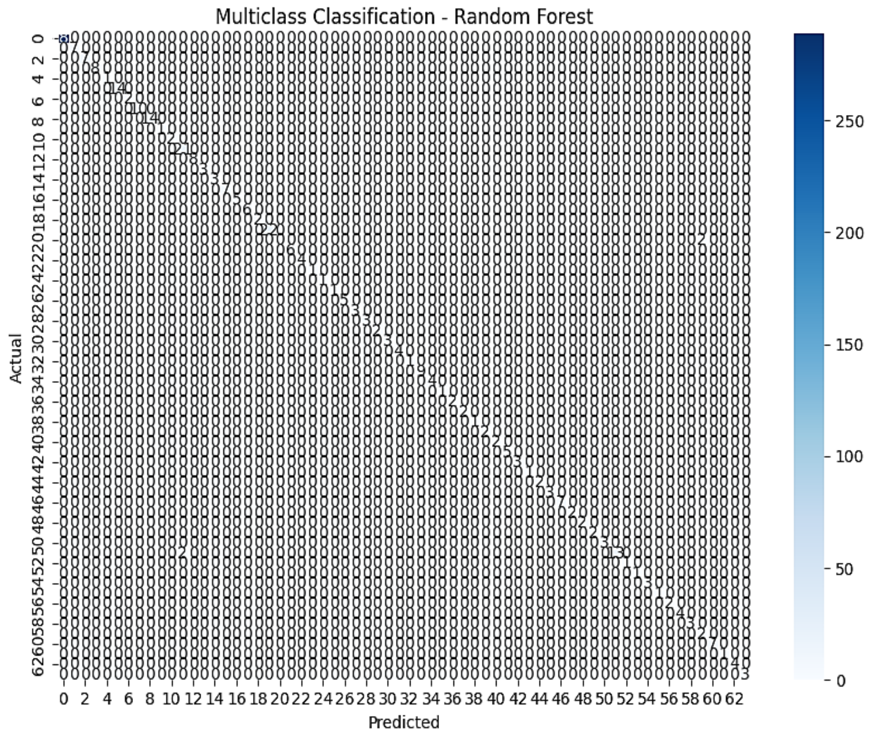


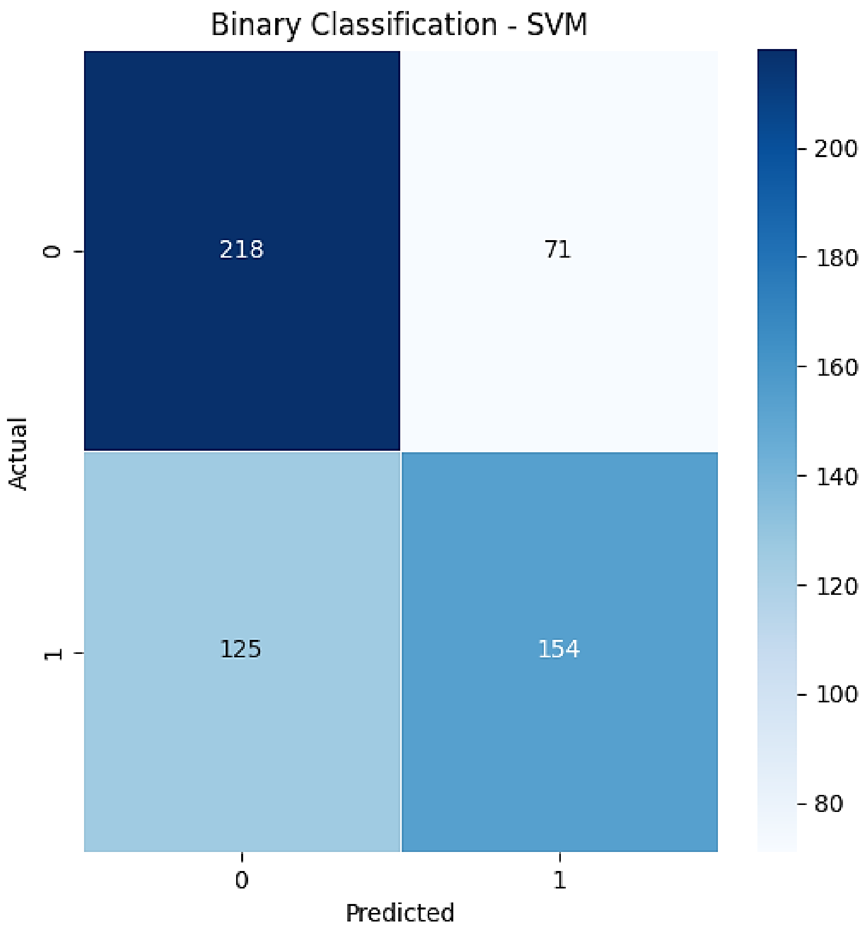
Figure 18. Comparison of binary classification model (split 70-30).



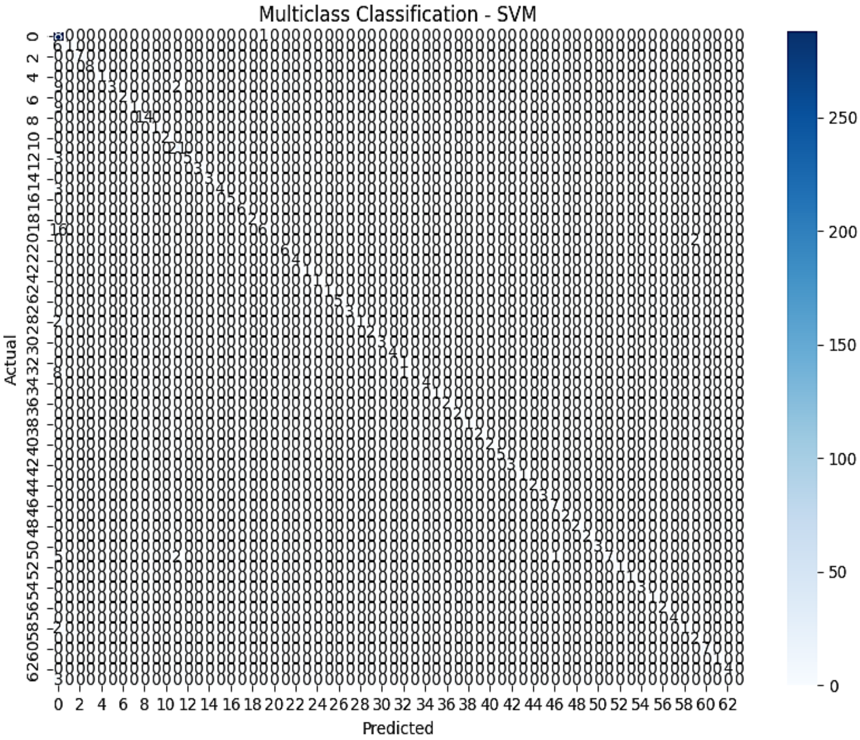
(a)


Multiclass Classification - Random Forest
Actual
Predicted
0
50
100
150
200
250

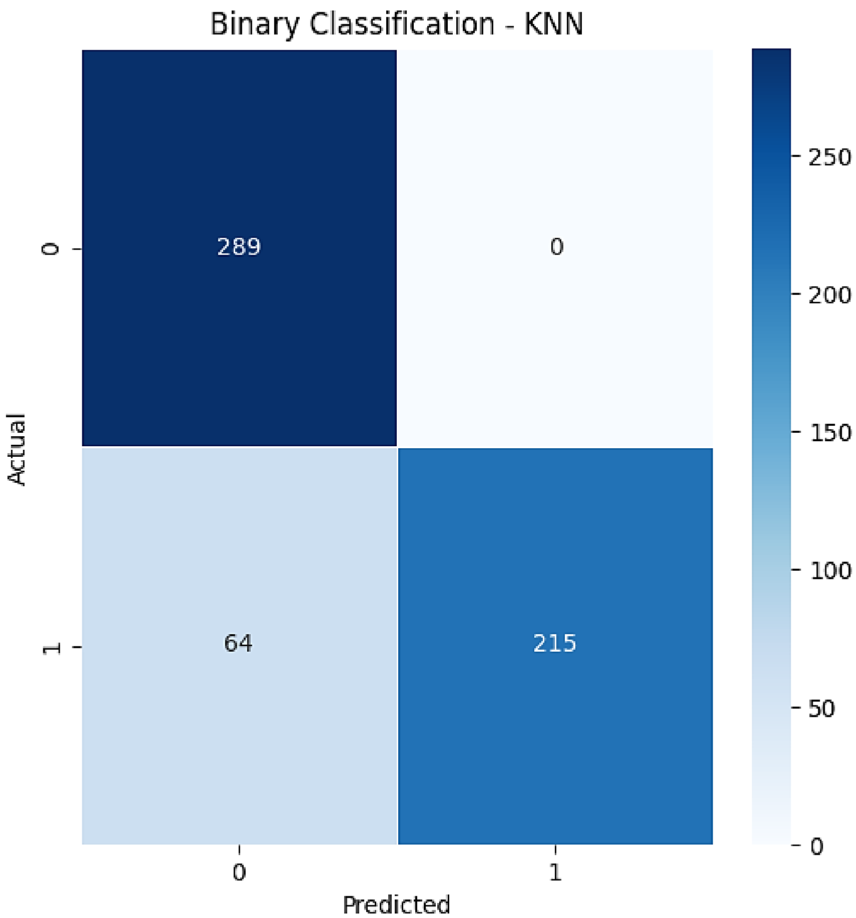
(b)


Binary Classification - SVM
218
71
125
154
Actual
Predicted
0
1
80
100
120
140
160
180
200

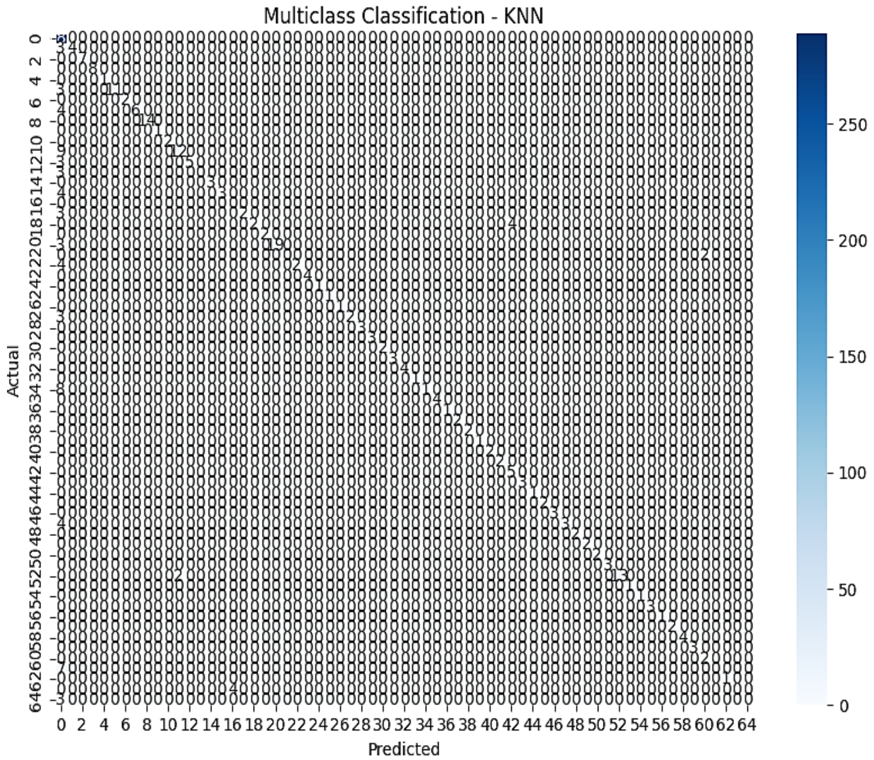
(c)


Multiclass Classification - SVM
Actual
Predicted
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62
250
200
150
100
50
0

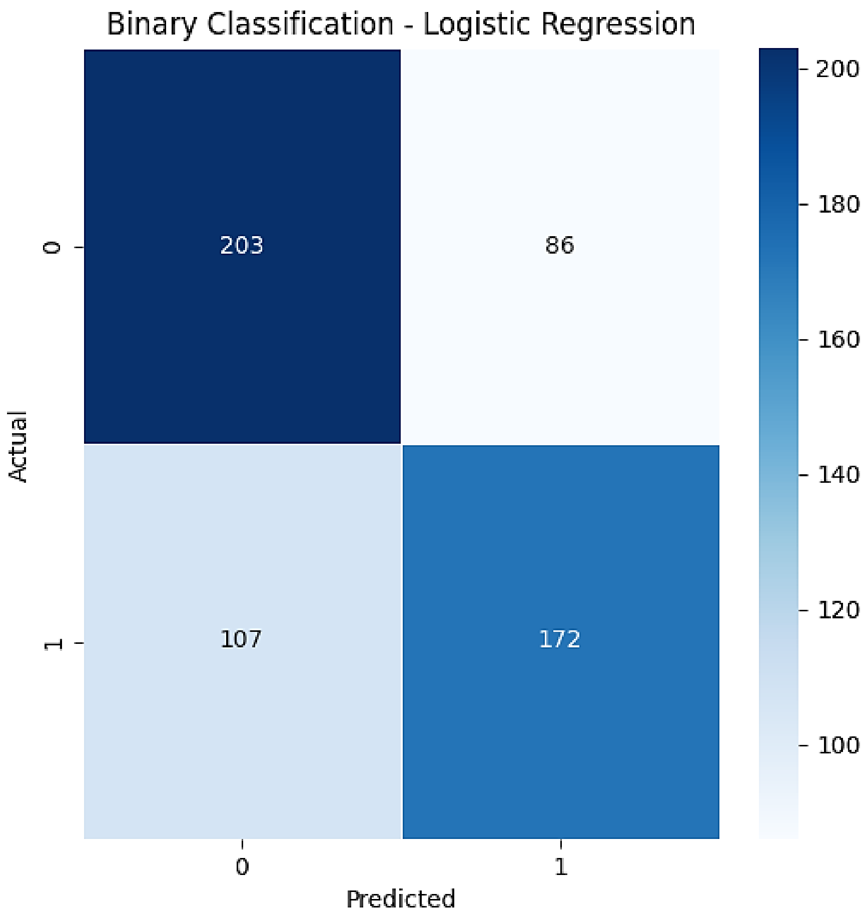
(d)


Binary Classification - KNN
289
0
64
215
Actual
Predicted
0
1
250
200
150
100
50
0

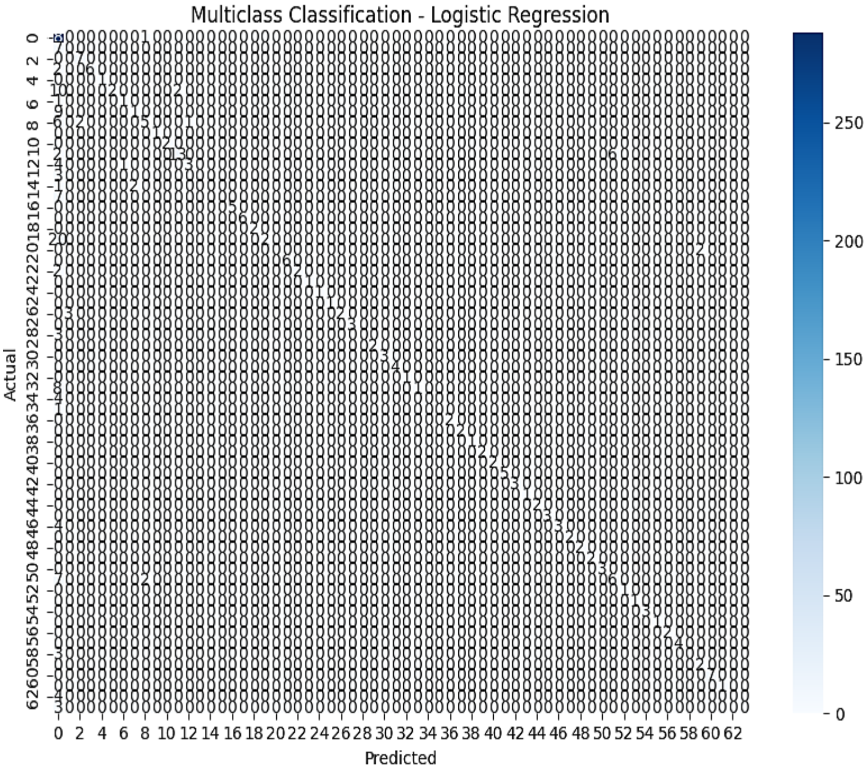
(e)


Multiclass Classification - KNN
Actual
Predicted
250
200
150
100
50
0

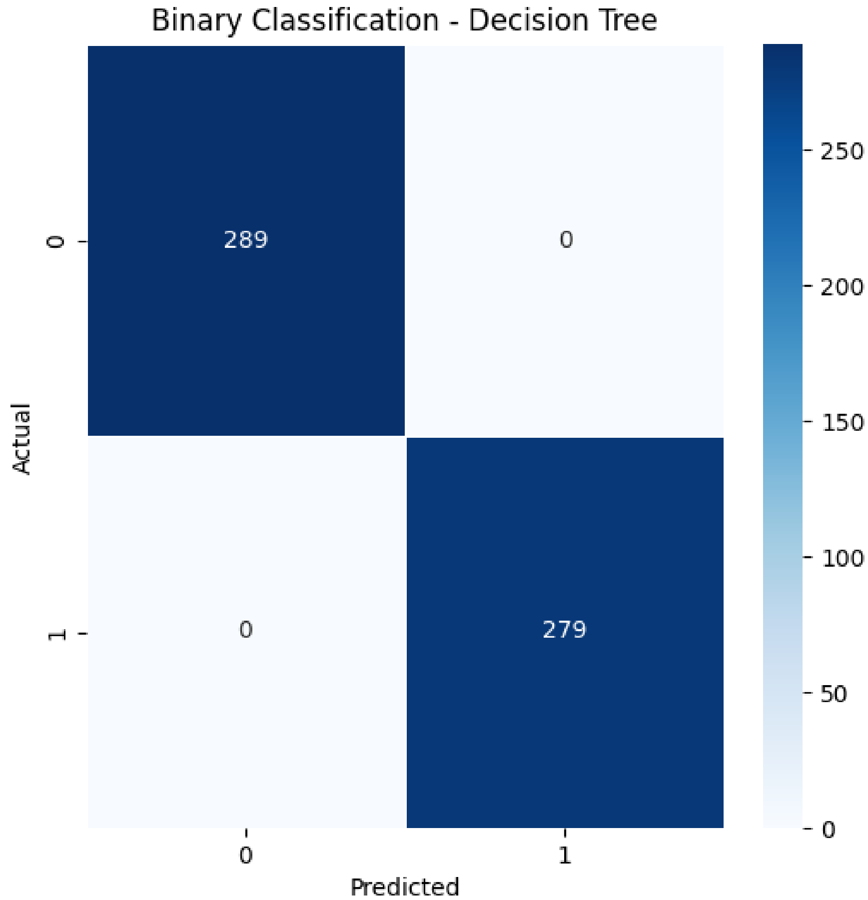
(f)


Binary Classification - Logistic Regression
203
86
107
172
Actual
Predicted
0
1
200
180
160
140
120
100

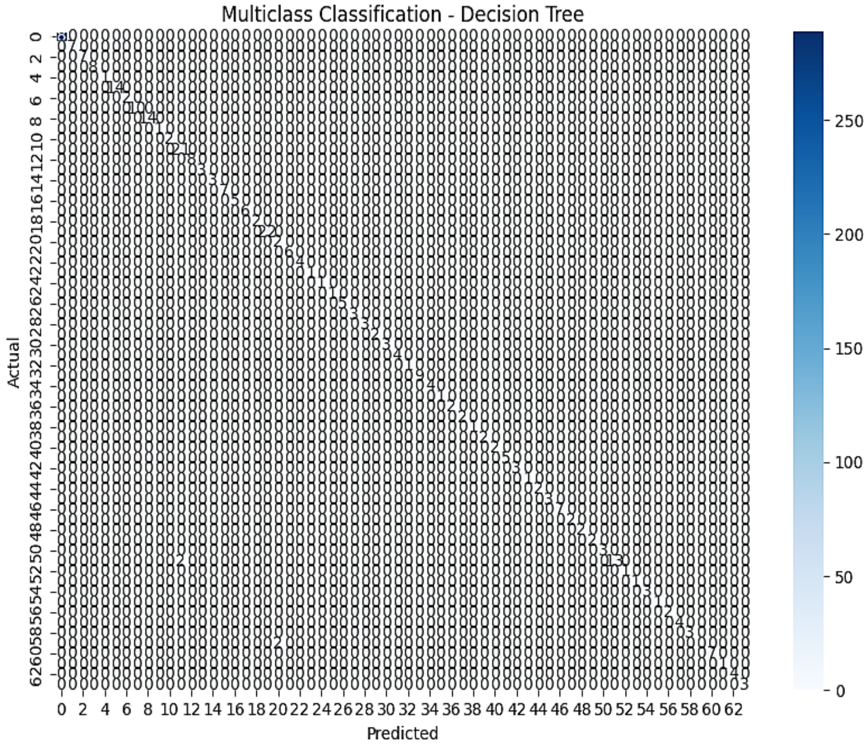
(g)


Multiclass Classification - Logistic Regression
Actual
Predicted
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62
0
50
100
150
200
250

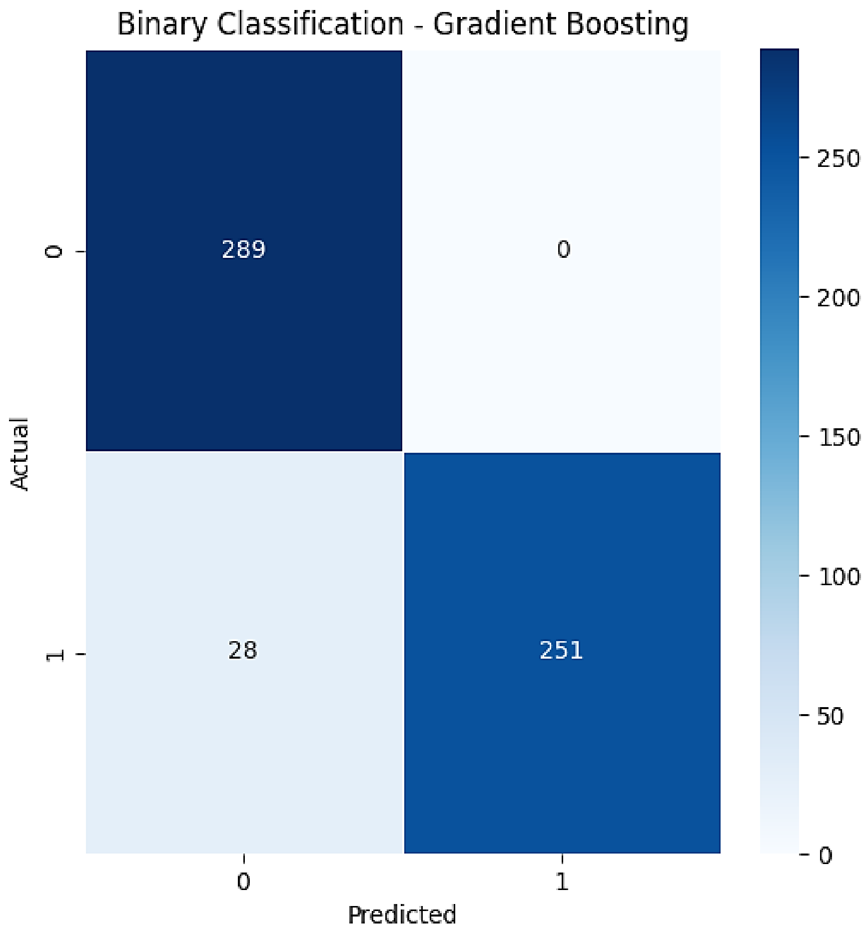
(h)


Binary Classification - Decision Tree
Actual
Predicted
289
0
0
279
0
1
0
50
100
150
200
250

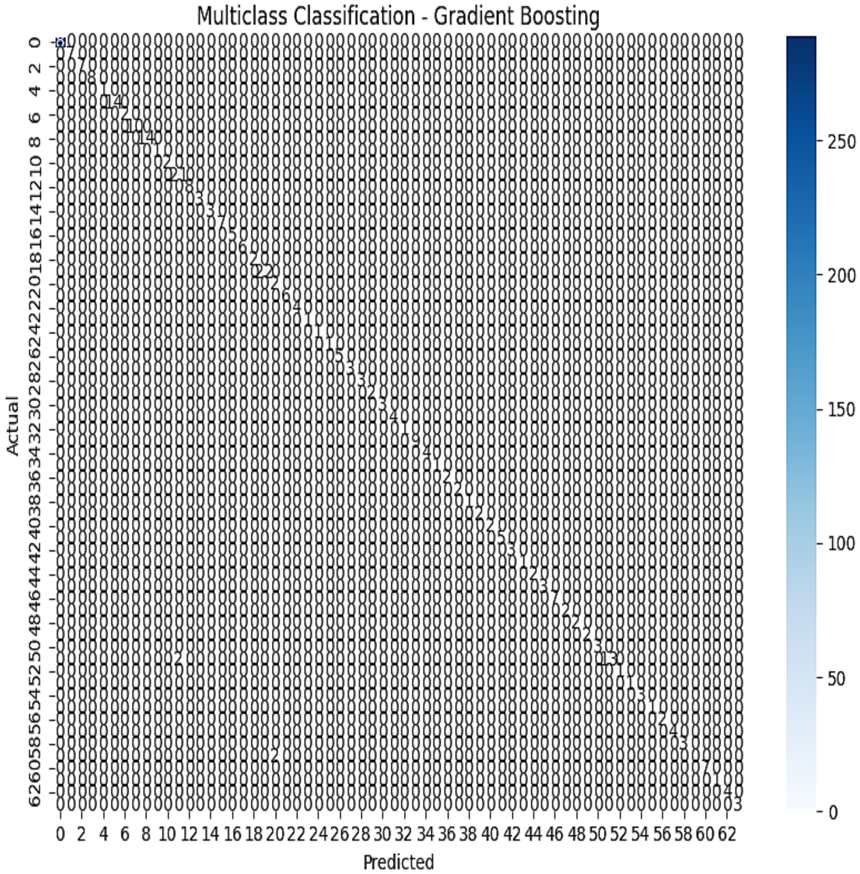
(i)


Multiclass Classification - Decision Tree
Actual
Predicted
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62
250
200
150
100
50
0

(j)


Binary Classification - Gradient Boosting
289
0
28
251
Actual
Predicted
0
1
250
200
150
100
50
0

(k)



(l)

Figure 19. (a–l) Binary and multiclass confusion metrics (split 80-20).

- *Confusion metrics after datasets splitting on 70% training set and 30% testing set:* The figure displays the confusion matrix, illustrating the model's classification performance and prediction accuracy on the 30% testing set (Figure 20(a–l)) (Tables 13–15).

Table 10. Binary classification performance.

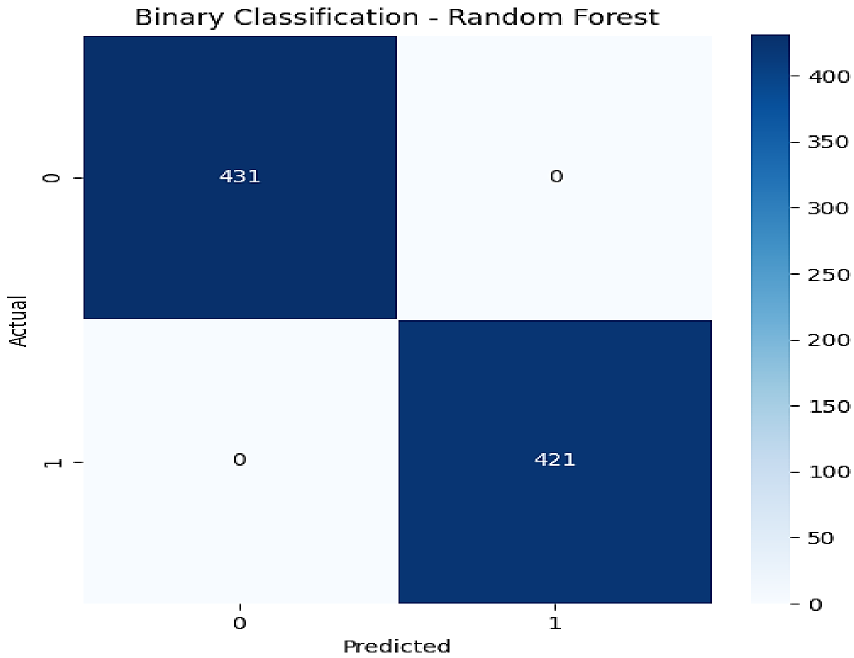
Model	TP	TN	FP	FN	Observations
RF	279	289	0	0	Perfect classification, no errors
DT	279	289	0	0	Flawless classification but risk of overfitting
SVM	251	289	0	28	Improved generalization over Decision Tree
LR	215	289	0	64	Better than SVM, worse than RF & Boosting
KNN	154	218	71	125	Struggles with misclassifications
GB	172	203	86	107	High false positives and negatives, poor performance

Table 11. Multiclass classification performance.

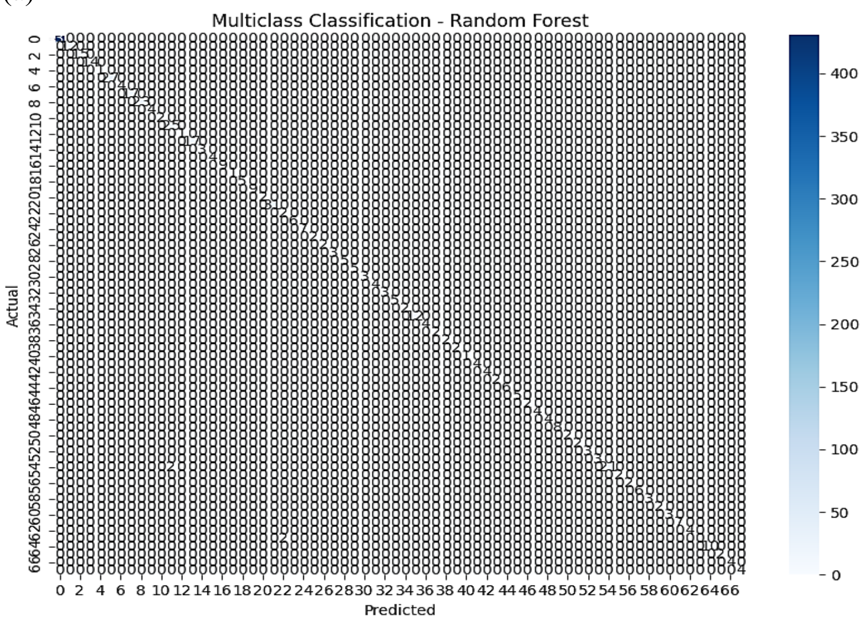
Model	Confusion Matrix Pattern	Key Observations
RF	Strong diagonal dominance	Highly accurate predictions, best overall.
DT	Strong diagonal, sharp separation	Risk of overfitting but excellent classification.
SVM	Slightly higher misclassifications	Good generalization, better than Decision Tree.
LR	Good diagonal but more noise	Moderate accuracy, slightly weaker than Random Forest.
KNN	More noise than Random Forest	Struggles compared to ensemble models.
GB	Most misclassifications	Weakest performer, poor boundary distinction.

Table 12. Performance analysis of models.

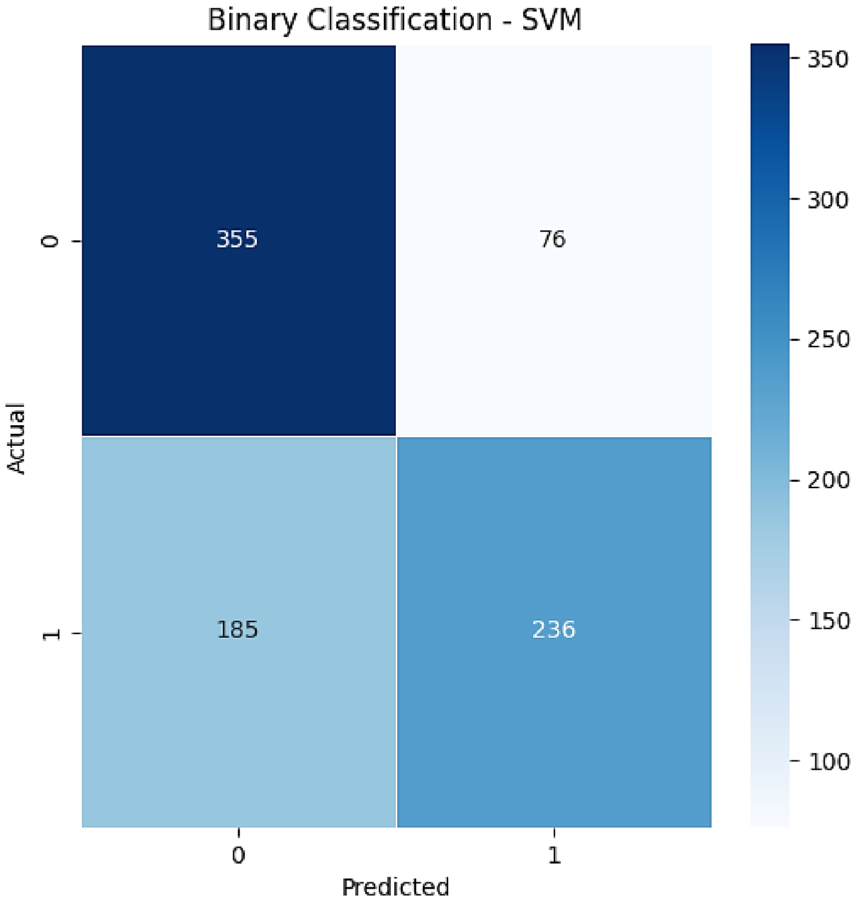
Model	Binary Classification	Multiclass Classification	Key Observations
RF	Perfect accuracy (TP = 279, TN = 289, FP = 0, FN = 0)	Strong diagonal, minimal misclassifications.	Best overall performer.
DT	Perfect accuracy	One of the best performers.	Excellent choice but may overfit.
SVM	28 false negatives (weaker recall)	Slightly more misclassifications than others.	Good, but not as strong as RF, DT.
LR	Moderate false negatives (FN = 64)	More errors than RF, DT.	Moderate performer.
KNN	High false negatives (FN = 125) and false positives (FP = 71)	More misclassifications than RF, DT.	Weak performance.
GB	High false positives (FP = 86) and false negatives (FN = 107)	Most misclassifications.	Least recommended model.



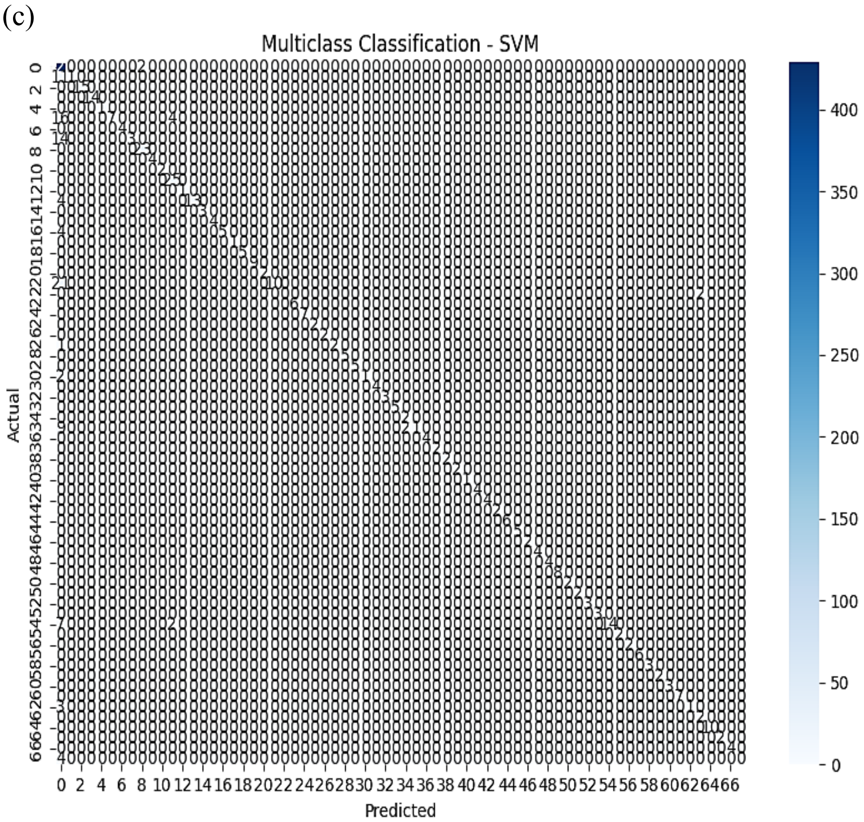
(a)



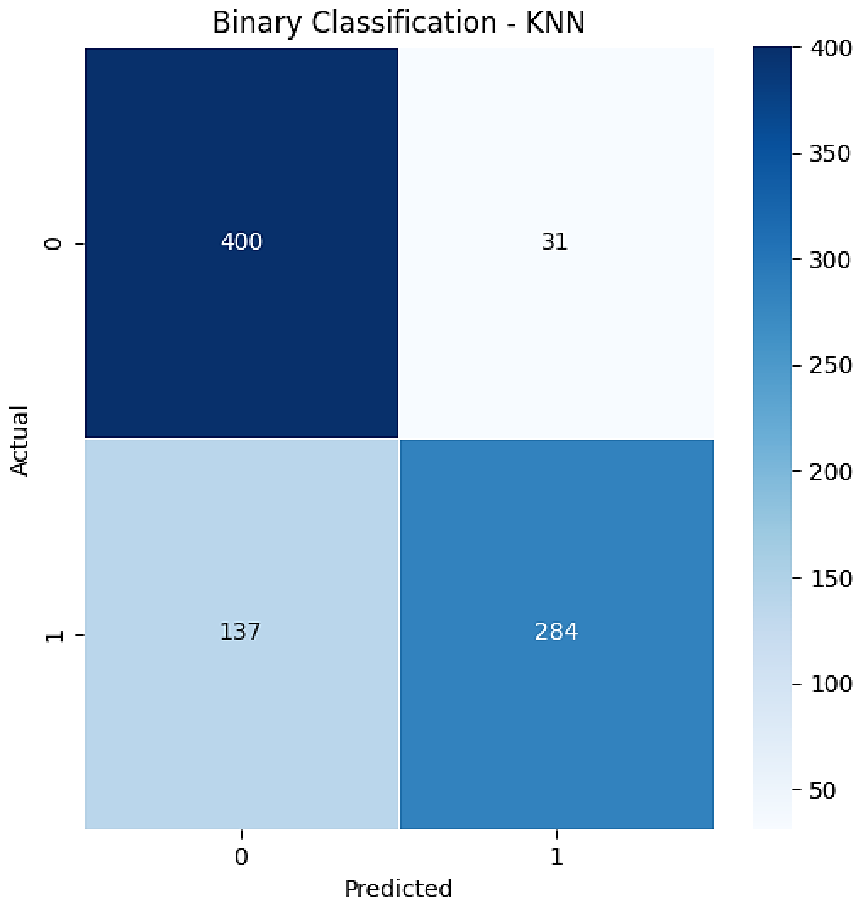
(b)


Binary Classification - SVM
355
76
185
236
Actual
Predicted
0
1
350
300
250
200
150
100

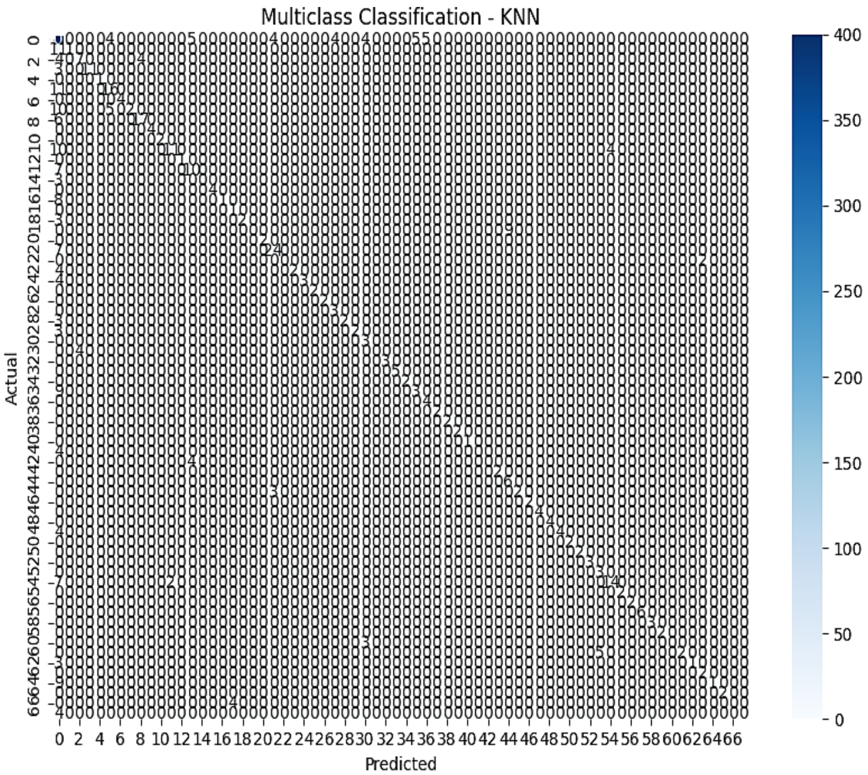
(c)


Multiclass Classification - SVM
Actual
Predicted
400
350
300
250
200
150
100
50
0

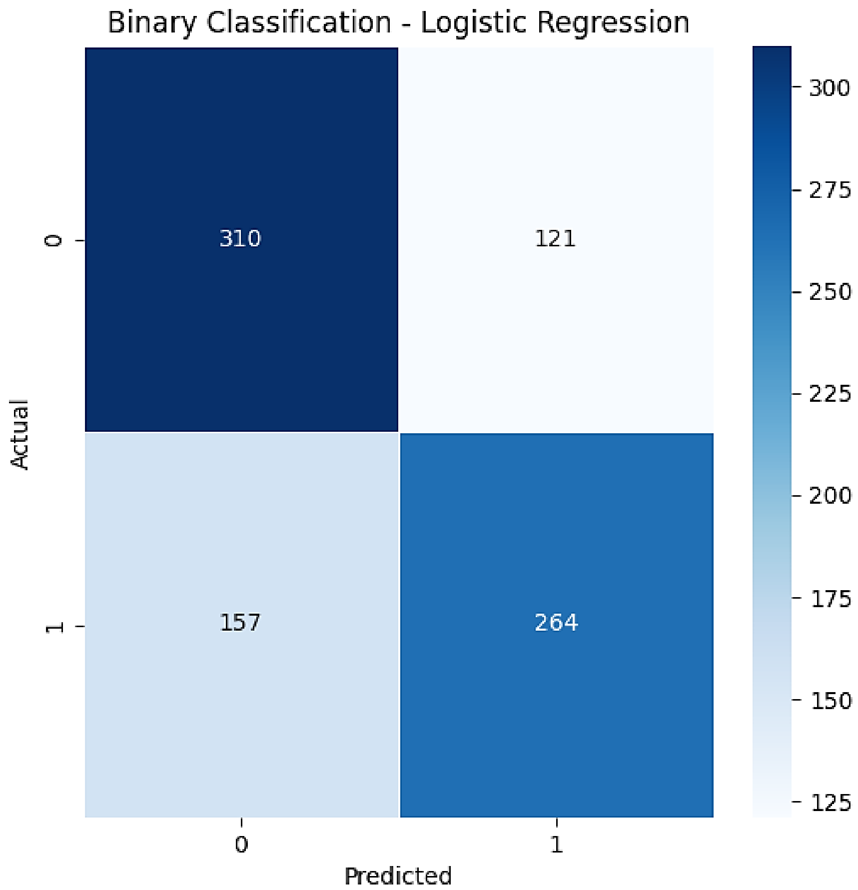
(d)


Binary Classification - KNN
400
31
137
284
Actual
Predicted
0
1
0
1
400
350
300
250
200
150
100
50

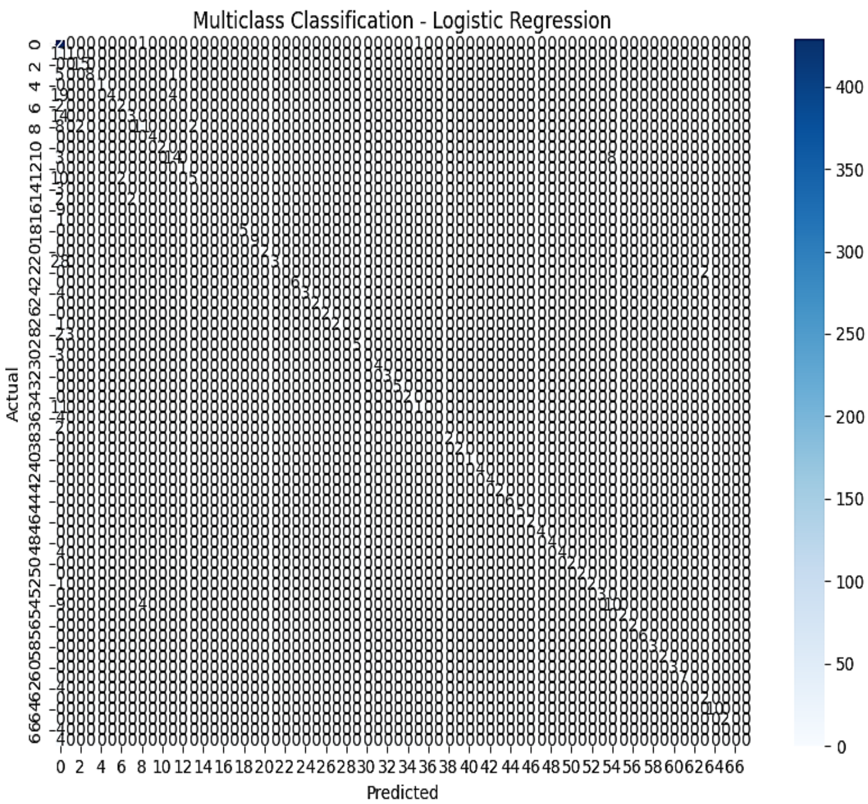
(e)


Multiclass Classification - KNN
Actual
Predicted
400
350
300
250
200
150
100
50
0

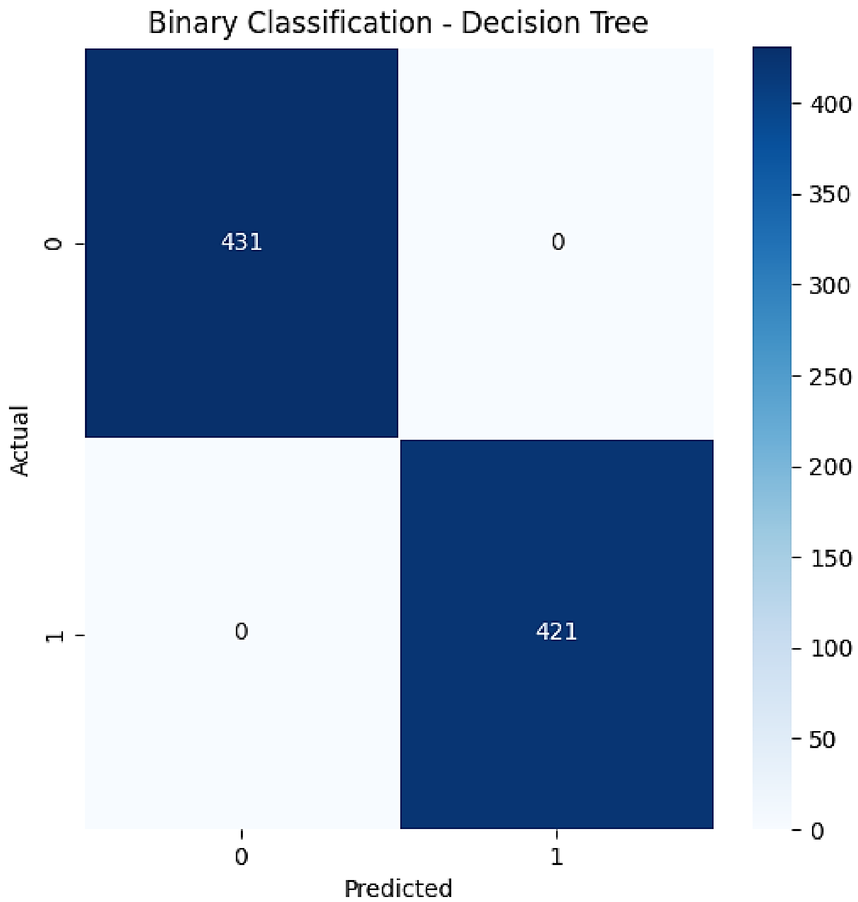
(f)


Binary Classification - Logistic Regression
310
121
157
264
0
1
Actual
0
1
Predicted
300
275
250
225
200
175
150
125

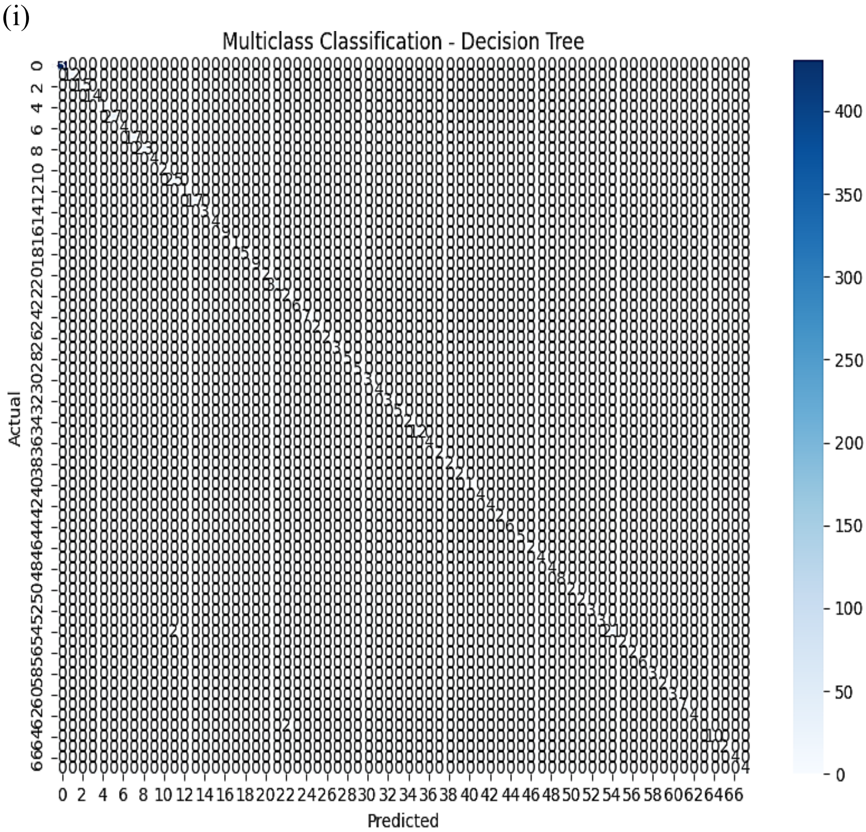
(g)


Multiclass Classification - Logistic Regression
Actual
Predicted
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66
400
350
300
250
200
150
100
50
0

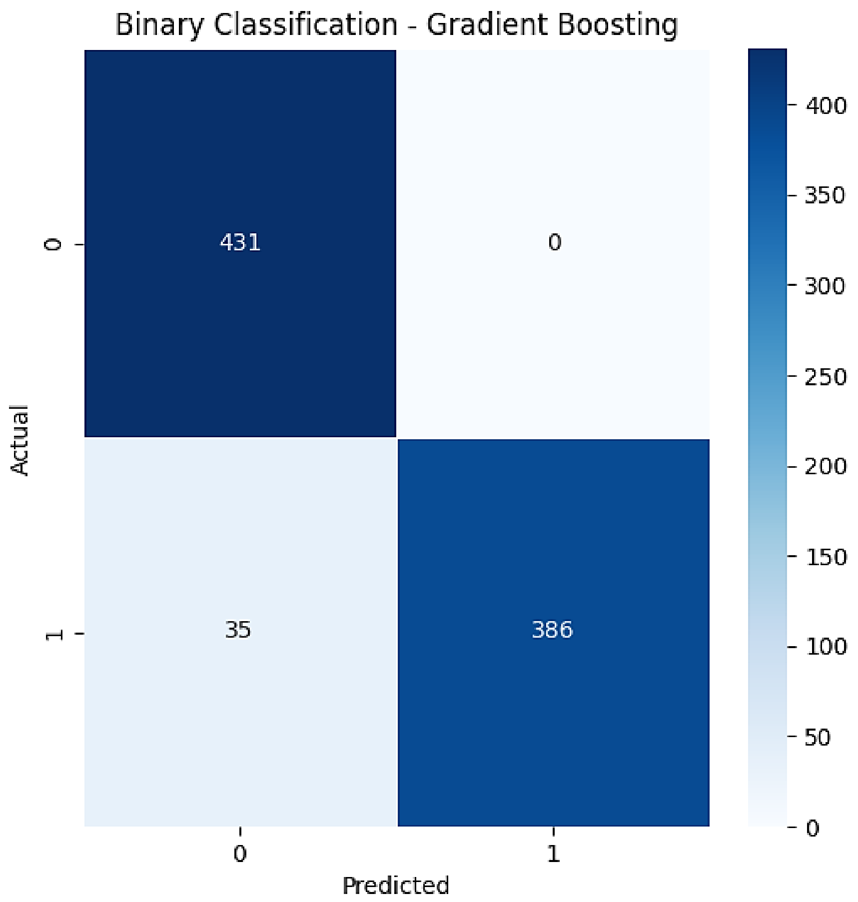
(h)


Binary Classification - Decision Tree
Actual
Predicted
431
0
0
421
0
1
0
1
400
350
300
250
200
150
100
50
0

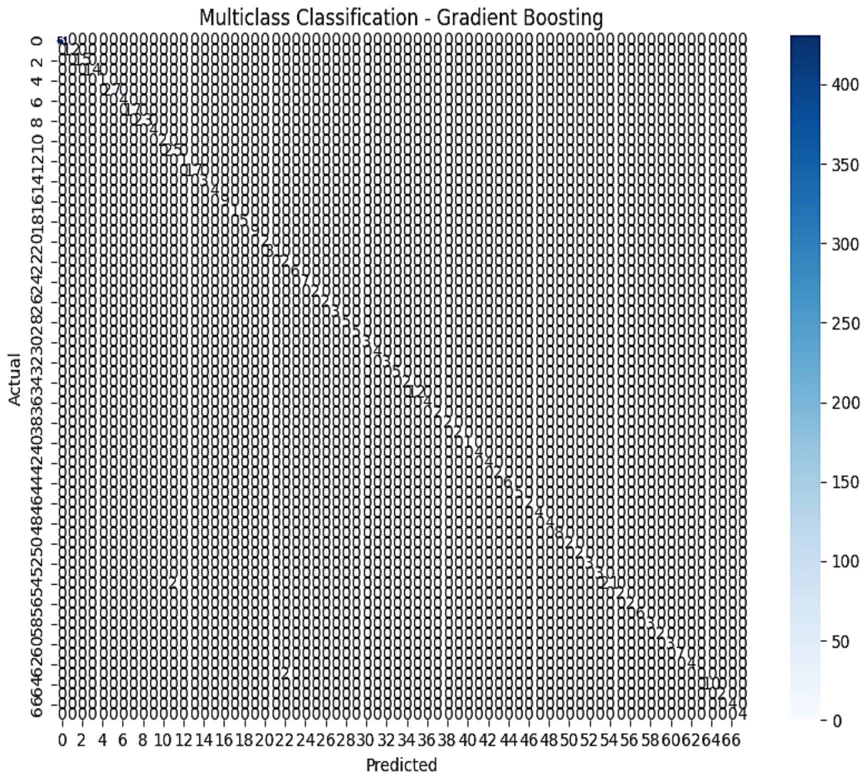
(i)


Multiclass Classification - Decision Tree
Actual
Predicted
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66
400
350
300
250
200
150
100
50
0

(j)



(k)



(l)

Figure 20. (a–l) Binary and multiclass confusion metrics (split 70-30).

Table 13. Binary classification performance.

Model	TN	FP	FN	TP	Observation
RF	431	0	0	421	Perfect classification, 100% accuracy.
DT	355	76	185	236	High false negatives and false positives, misclassification issues.
SVM	400	31	137	284	Moderate performance, better than SVM but worse than Random Forest.
LR	310	121	157	264	Worst performance, high false positives and false negatives.
KNN	431	0	0	421	Perfect classification with 100% accuracy, precision, and recall.
GB	431	0	35	386	35 false negatives indicate the model struggles with recall for class 1 but maintains high precision.

Table 14. Multiclass classification performance.

Model	Confusion Matrix Pattern	Observation
RF	Strong diagonal, very few off-diagonal values	High accuracy, minimal misclassifications.
DT	Diagonal presence but more off-diagonal values	Good classification, but higher misclassification than Random Forest.
SVM	Strong diagonal but some off-diagonal misclassifications	Performs well but struggles in high-dimensional classification.
LR	More misclassified instances compared to others	Worst performance, not suitable for this problem.
KNN	Strong diagonal dominance, very few off-diagonal values.	Minimal misclassifications, indicating strong performance.
GB	Diagonal pattern present, but with slightly more off-diagonal values.	Performs well, but slightly more misclassifications compared to Decision Tree.

Table 15. Performance analysis of models.

Model	Binary Classification	Multiclass Classification	Key Observations
RF	Perfect accuracy (TP = 421, TN = 431, FP = 0, FN = 0)	Strong diagonal, minimal misclassifications.	Best overall performer.
DT	Perfect accuracy	One of the best performers.	Excellent choice.
SVM	35 false negatives (weaker recall)	Slightly more misclassifications than others.	Good, but not as strong as RF, DT.
LR	Moderate false negatives (FN = 137)	More errors than RF, DT.	Moderate performer.
KNN	High false negatives (FN = 185) and false positives	More misclassifications than RF, DT.	Weak performance.
GB	High false positives and false negatives (Worst)	Most misclassifications.	Least recommended model.

From the ROC curves, the 70-30 split appears to yield better generalization for some models compared to the 80-20 split. While RF and DT maintained perfect classification (AUC = 1.00) in both cases, GB slightly improved from 0.95 (80-20) to 0.96 (70-30) which might indicate better generalization with more test data. SVM also improved from 0.65 to 0.69, and Logistic Regression slightly increased from 0.66 to 0.67.

This suggests that with a 70-30 split, the models are tested on a more diverse set, leading to more stable generalization. The 80-20 split might have been slightly overfitting for some models, as indicated by their lower AUC scores when tested on a smaller test set. This reinforces the importance of choosing an appropriate train-test split depending on dataset size and model complexity (Figures 21 and 22).

Final Model and Split Recommendation

Based on overall performance across both binary and multiclass classification, RF consistently achieved the highest 100% accuracy with minimal misclassifications. It demonstrated strong generalization, robustness against noise, and superior performance compared to other models, including DT, which, despite their strengths, may require more hyperparameter tuning.

- *Recommended Model:* RF – It is the most reliable choice, balancing accuracy, efficiency, and generalization, making it the optimal model for my dataset.

- *Recommended Split:* 70-30 – It ensures superior generalization and accuracy, making it the optimal choice for my dataset.

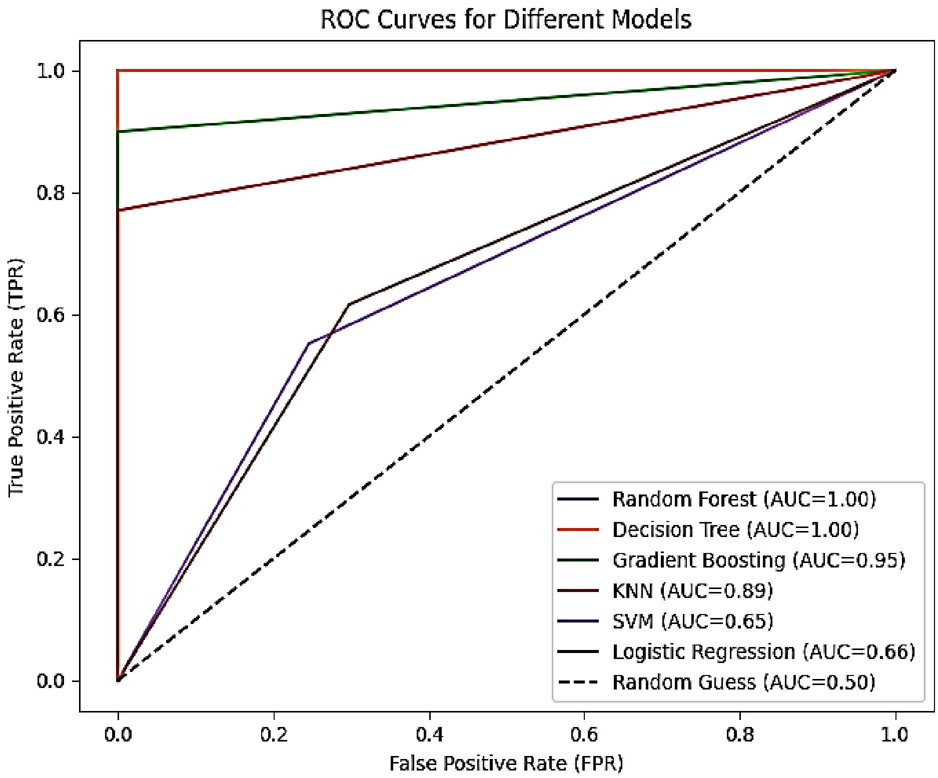


Figure 21. AUC-ROC on 80-20 dataset splitting.

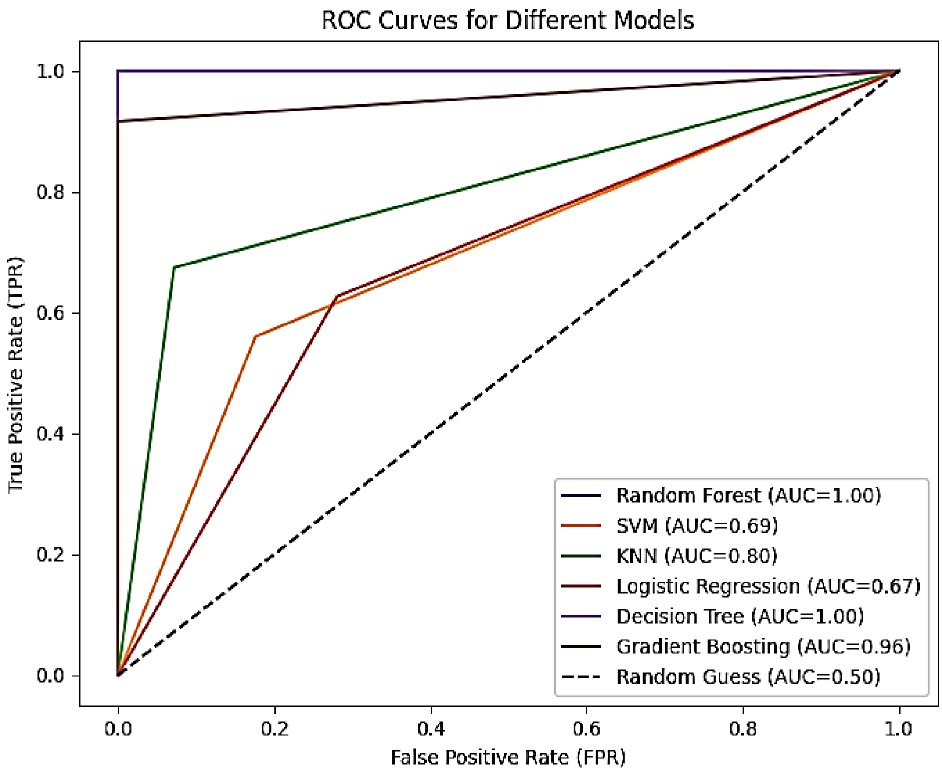


Figure 22. AUC-ROC on 70-30 dataset splitting.

CONCLUSIONS

The Automated ECG Prediction System offers a groundbreaking solution for automated, real-time diagnosis of cardiovascular conditions. By combining advanced signal processing techniques with RF and other classifiers, the system achieves unparalleled accuracy in both binary and multi-class classification tasks. The intuitive user interface further enhances its clinical applicability. This work covers the implementation for AI-driven innovations in cardiac care, promising improved diagnostic accuracy, efficiency, and patient outcomes. The future work might include real time prediction for diagnosis patient's CVD.

REFERENCES

1. Sumalatha U, Prakasha KK, Prabhu S, Nayak VC. Deep learning applications in ECG analysis and disease detection: An investigation study of recent advances. IEEE Access. 2024;12. doi: 10.1109/ACCESS.2024.3447096.
2. Pan J, Tompkins WJ. A real-time QRS detection algorithm. IEEE Trans Biomed Eng. 1985;32(3):230–236. doi: 10.1109/TBME.1985.325532.
3. Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PC, Mark RG, et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation. 2000;101(23):e215–e220. doi: 10.1161/01.cir.101.23.e215.
4. Breiman L. Random forests. Mach Learn. 2001;45(1):5–32. doi: 10.1023/A:1010933404324.
5. Acharya UR, Fujita H, Lih OS, Adam M, Tan JH, Chua CK. Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network. Inf Sci. 2017;405(C):81–90. doi: 10.1016/j.ins.2017.04.012.
6. Moody GB, Mark RG. The impact of the MIT-BIH arrhythmia database. IEEE Eng Med Biol Mag. 2001;20(3):45–50. doi: 10.1109/51.932724.
7. Kalmady SV, Salimi A, Sun W, Sepehrvand N, Nademi Y, Bainey K, et al. Development and validation of machine learning algorithms based on electrocardiograms for cardiovascular diagnoses at the population level. NPJ Digit Med. 2024;7(1):133. doi: 10.1038/s41746-024-01130-8.
8. Aziz S, Ahmed S, Alouini MS. ECG-based machine-learning algorithms for heartbeat classification. Sci Rep. 2021;11(1):18738. doi: 10.1038/s41598-021-97118-5.
9. Pham H, Egorov K, Kazakov A, Budennyy S. Machine learning-based detection of cardiovascular disease using ECG signals: Performance vs. complexity. Front Cardiovasc Med. 2023;10:1229743. doi: 10.3389/fcvm.2023.1229743.
10. Ayano YM, Schwenker F, Dufera BD, Debelee TG. Interpretable machine learning techniques in ECG-based heart disease classification: A systematic review. Diagnostics (Basel). 2022;13(1):111. doi: 10.3390/diagnostics13010111.
11. Abubaker MB, Babayiğit B. Detection of cardiovascular diseases in ECG images using machine learning and deep learning methods. IEEE Trans Artif Intell. 2022;4(2):373–382. doi: 10.1109/TAI.2022.3159505.
12. Gondane R, Rajhans R, Susheela Devi V. Classification using probabilistic random forest. In: 2015 IEEE Symposium Series on Computational Intelligence. Piscataway: IEEE; 2015. pp. 174–179. doi: 10.1109/SSCI.2015.35.
13. Abubaker MB, Babayiğit B. Detection of cardiovascular diseases in ECG images using machine learning and deep learning methods. IEEE Trans Artif Intell. 2023;4(2):373–82. doi:10.1109/TAI.2022.3159505.
14. Venkatesan P, Karthigaikumar S, Paul A, Satheeskumaran S, Kumar R. ECG signal preprocessing and SVM classifier-based abnormality detection in remote healthcare applications. IEEE Access. 2018;6:9767–9773. doi: 10.1109/ACCESS.2018.2794346.
15. Bazi Y, Alajlan N, AlHichri H, Malek S. Domain adaptation methods for ECG classification. In: 2013 IEEE International Conference on Computer Medical Applications (ICCMA). Piscataway: IEEE; 2013. pp. 1–4. doi: 10.1109/ICCMA.2013.6506156.
16. Pachiyannan M, Alsulami D, Alsadie AKJ, Saudagar M, AlKhathami M, Poonia RC. A novel machine learning-based prediction method for early detection and diagnosis of congenital heart

disease using ECG signal processing. Technologies. 2024;12(1):4. doi: 10.3390/technologies12010004.
17. Maleki M, Haeri F. Identification of cardiovascular diseases through ECG classification using wavelet transformation. arXiv [Preprint]. 2024. doi: 10.48550/arXiv.2404.09393.
18. Mincholé A, Camps J, Lyon A, Rodríguez B. Machine learning in the electrocardiogram. J Electrocardiol. 2019;57S:S61–S64. doi: 10.1016/j.jelectrocard.2019.08.008.
19. De S, Chakraborty B. Disease Detection System (DDS) using machine learning technique. In: Dey N, Bhateja V, Hassanien AE, editors. International Conference on Machine Learning with Health Care Perspective. Singapore: Springer; 2020. pp. 107–132.
20. Fatima M, Pasha M. Survey of machine learning algorithms for disease diagnostic. J Intell Learn Syst Appl. 2017;9(1):1–16. doi: 10.4236/jilsa.2017.91001.
21. Richens JG, Lee CM, Johri S. Improving the accuracy of medical diagnosis with causal machine learning. Nat Commun. 2020;11(1):3923. doi: 10.1038/s41467-020-17419-7.
22. Babu MG, Shankar G. An exploration of ECG signal feature selection and classification using machine learning techniques. Int J Innov Technol Explor Eng. 2020;9(3):1917–1922. doi: 10.35940/ijitee.C8728.019320.