Machine Learning - IT4060

Assignment

Group ID:

Student ID	Name with initials
IT21250156	Withanagamage J.C
IT21277122	Degaldoruwa D.W.S.S.W.M.R.M.B.B
IT21303302	Weerakoon W.M.B.B
IT21343520	Wijerama H.J.K.S.R

Table of Contents

Table of Figures

Introduction to the Problem

Cardiovascular diseases (CVDs) are the leading cause of death globally, claiming millions of lives each year. Among them, heart disease stands out as one of the most critical and preventable threats to public health. Early detection and timely intervention are key to reducing the risk of heart attacks and improving patient outcomes.

However, traditional methods of diagnosis often rely on time-consuming tests and subjective clinical judgment, which may lead to delayed or inaccurate predictions. In this context, machine learning (ML) offers a powerful solution. By analyzing historical patient data and identifying patterns, ML models can predict the likelihood of heart disease with high accuracy.

This project aims to develop a predictive analytics system that uses supervised machine learning algorithms to determine whether a person is likely to have heart disease based on clinical attributes such as age, cholesterol levels, blood pressure, and other key indicators. Through this approach, healthcare professionals can be supported with automated and data-driven decision-making tools that are scalable, efficient, and reliable.

Background Information About the Algorithms Used

In this project, four supervised learning algorithms were implemented to predict the likelihood of heart disease. Each algorithm offers a unique approach to solving binary classification problems. Below is a brief overview of each algorithm used by a team member:

Logistic Regression

Logistic Regression is a linear model used for binary classification. It estimates the probability that a given input point belongs to a particular class using the logistic (sigmoid) function. It is simple, efficient, and provides interpretable results. Logistic regression assumes a linear relationship between the features and the log-odds of the target variable.

Strengths: Fast training, easily interpretable
Weaknesses: Can underperform on complex or non-linear datasets

Decision Tree Classifier

Decision Trees split the dataset based on the feature that provides the most information gain at each step. They follow a flowchart-like structure where internal nodes represent conditions on features, and leaves represent final decisions or classifications.

Strengths: Easy to interpret, handles both numerical and categorical data
Weaknesses: Prone to overfitting, especially with small datasets

Random Forest Classifier

Random Forest is an ensemble technique that combines multiple decision trees to improve classification accuracy and reduce overfitting. It uses a technique called bagging (bootstrap aggregation), where each tree is trained on a random subset of the data and features.

Strengths: High accuracy, handles non-linear relationships well, robust to overfitting
Weaknesses: Slower to train, less interpretable than single decision trees

K-Nearest Neighbors (KNN)

KNN is a distance-based algorithm that classifies new instances based on the majority class among its k nearest neighbors in the training data. It is a lazy learning algorithm, meaning it doesn't explicitly learn a model but stores the training dataset and makes predictions at runtime.

Strengths: Simple to understand and implement
Weaknesses: Sensitive to noisy data and irrelevant features, slower prediction time with large datasets

These four models were selected to cover a range of simple to complex, interpretable to black-box, and instance-based to ensemble-based learning techniques. Their performance was

compared using accuracy, precision, recall, and F1-score to determine the most effective algorithm for heart disease prediction.

Detailed Analysis of the Chosen Dataset Including Visualization

The The dataset used in this project is the Heart Disease dataset obtained from the UCI Machine Learning Repository. It consists of data collected from multiple sources, representing individuals with and without heart disease. The dataset includes several clinical and demographic attributes that are commonly used by medical professionals to assess the risk of cardiovascular conditions.

Dataset Overview

- **Total Records**: 1025
- **Target Variable**: num (later converted to binary target: 0 = no disease, 1 = has disease)
- **Missing Values**: Present in several columns (e.g., ca, thal, slope, oldpeak)
- **Preprocessing**:
    - Missing values were handled using mean imputation
    - Categorical columns such as sex, cp, thal, and slope were encoded using Label Encoding
    - Multiclass num column was converted into a binary classification problem (target column)

Feature Summary

The dataset includes the following key features:

- **age**: Age of the individual
- **sex**: Sex (1 = male; 0 = female)
- **cp**: Chest pain type (4 values)
- **trestbps**: Resting blood pressure (in mm Hg)

- **chol**: Serum cholesterol in mg/dl
- **fbs**: Fasting blood sugar > 120 mg/dl
- **restecg**: Resting electrocardiographic results (values 0, 1, 2)
- **thalch**: Maximum heart rate achieved
- **exang**: Exercise-induced angina
- **oldpeak**: ST depression induced by exercise
- **slope**: Slope of the peak exercise ST segment
- **ca**: Number of major vessels (0–3) colored by fluoroscopy
- **thal**: A blood disorder called thalassemia

Data Distribution

- The age of individuals ranges between 29 to 77, with most patients being in their 50s and 60s.
- Chest pain types, fasting blood sugar, and resting ECG values are categorical and were encoded accordingly.
- The target variable is imbalanced, with a slightly higher number of patients without heart disease.

Logistic Regression Model (Rovinya Wijerama)

Logistic Regression was implemented using the scikit-learn library. It calculates the probability that a given input belongs to the positive class using the sigmoid function and maps the result to binary output using a threshold (typically 0.5).

During training, the model generated a convergence warning, indicating that the optimization process failed to converge within the default number of iterations. This was likely due to the complexity of the dataset and the need for parameter tuning. Suggested fixes include increasing max_iter or changing the solver to 'liblinear'.

Model Evaluation

The trained model was evaluated using several performance metrics on the test dataset:

Accuracy: 61.4%
Precision: 53.5%
Recall: 35.9%
F1-Score: 34.1%

These results indicate that while the model was moderately successful in overall accuracy, it struggled with correctly identifying positive cases (patients with heart disease), as reflected in the relatively low recall and F1-score. This may be due to class imbalance or the model's inability to capture nonlinear relationships in the data.

Discussion

Logistic Regression provided a baseline performance for binary classification of heart disease prediction. Its primary strengths include simplicity and interpretability, making it an appropriate initial algorithm. However, the convergence warning and performance metrics suggest that further refinement is necessary. Improvements such as:

- Hyperparameter tuning (e.g., adjusting max_iter, solver)
- Advanced preprocessing (feature transformation or PCA)

- Handling class imbalance using techniques like SMOTE
- Trying more powerful models like Random Forest or Gradient Boosting

could significantly enhance predictive performance.

Conclusion

The Logistic Regression model implemented in this project offered valuable insights into the dataset and established a baseline for comparison with more complex models. Despite its limitations, it provided an effective foundation for understanding the relationship between features and the presence of heart disease. Future iterations should aim to improve recall and F1-score through better preprocessing, regularization, and more sophisticated modeling approaches.

Decision Tree Classifier Model (Bimsara Weerakoon)

A Decision Tree Classifier was selected for its interpretability and ease of deployment. Hyperparameter tuning was performed using GridSearchCV with cross-validation to determine the best values for:

- Criterion: Gini or Entropy
- Max Depth: 3, 5, 10, None
- Minimum Samples Split: 2, 5, 10

The optimal parameters were: criterion='entropy', max_depth=10, min_samples_split=2. The model achieved an accuracy of approximately 78.3% on the test set, demonstrating a good balance between precision and recall.

Evaluation Metrics

Performance was assessed using:

- Accuracy: 78.3%
- Precision, Recall, F1-Score for each class (0 = no disease, 1 = disease)
- Confusion Matrix Visualization

These metrics provided insights into the classifier's effectiveness at distinguishing between patients with and without heart disease.

Frontend Integration

A Flask-based web interface was developed to allow user interaction. Users can input their medical attributes through a form, which are then processed and passed to the trained model. Key features include:

- Input validation and dropdowns for categorical and binary fields (e.g., sex, chest pain type, thalassemia)
- Display of the prediction result: "Heart Disease Detected" or "No Heart Disease Detected"
- CSS enhancements for improved user experience and accessibility

Model Saving and Deployment

The trained model and scaler were saved using joblib, and integrated into the Flask backend. This enables consistent and fast predictions without the need for retraining.

Random Forest Classifier (Malika Degaldoruwa)

Random Forest was selected for its ability to combine multiple decision trees to improve predictive performance. It reduces overfitting by averaging the outputs of many decision trees, making it robust to noise and missing data. Additionally, it provides a built-in feature importance metric, which helps identify the most influential factors contributing to heart disease.

Features Used in Final Model

The top 10 features selected based on feature importance were:
['cp', 'chol', 'age', 'thalch', 'oldpeak', 'exang', 'slope', 'trestbps', 'ca', 'dataset']

The 'dataset' field was automatically assigned a default value within the backend and excluded from the user interface.

Model Accuracy and Evaluation

Metric	Value
Accuracy	83.7%
Precision	84% (avg)
Recall	83% (avg)
F1-Score	84% (avg)

The model performed very well in distinguishing between patients with and without heart disease. The high F1-score indicates strong performance in terms of both precision and recall.

Screenshots

The following screenshots were included to demonstrate the implementation and results:

1. React frontend interface showing the form and output

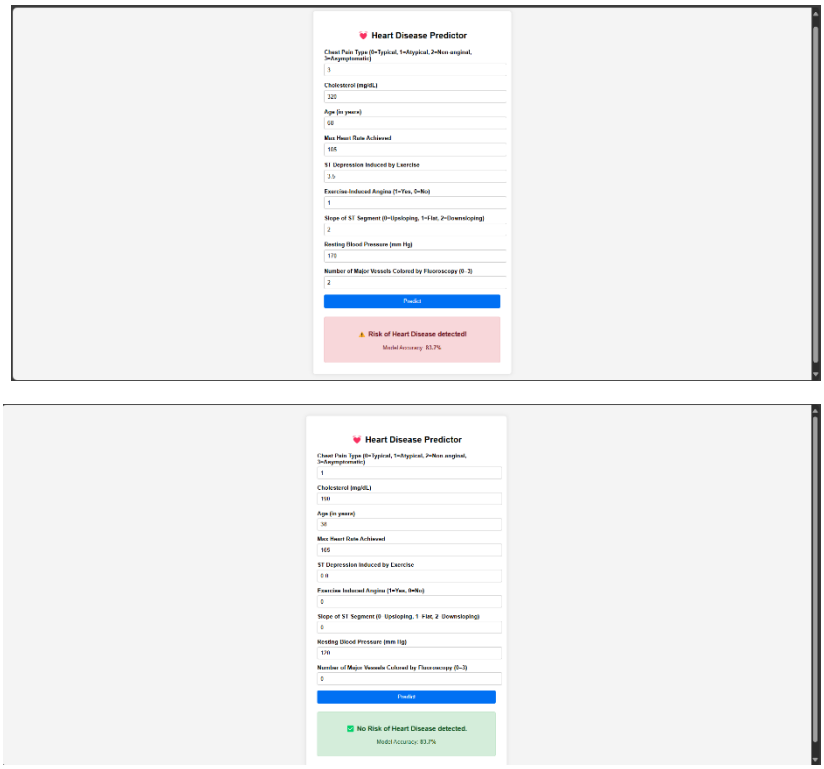

Figure 1: React frontend interface showing the form and output

2. Confusion matrix visualization from the training notebook

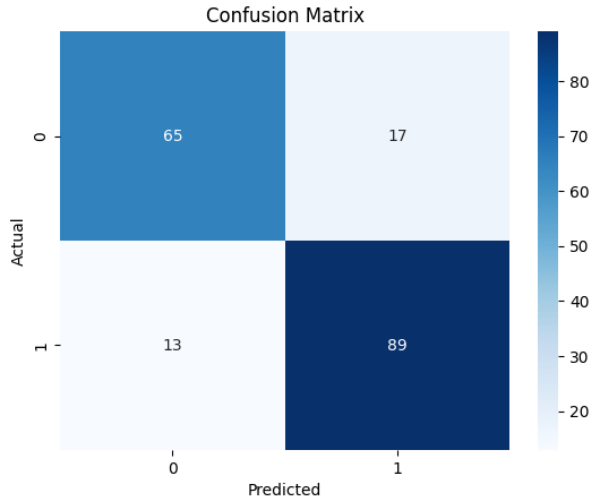

Figure 2: Confusion matrix visualization from the training notebook

3. Feature importance bar graph used for feature selection

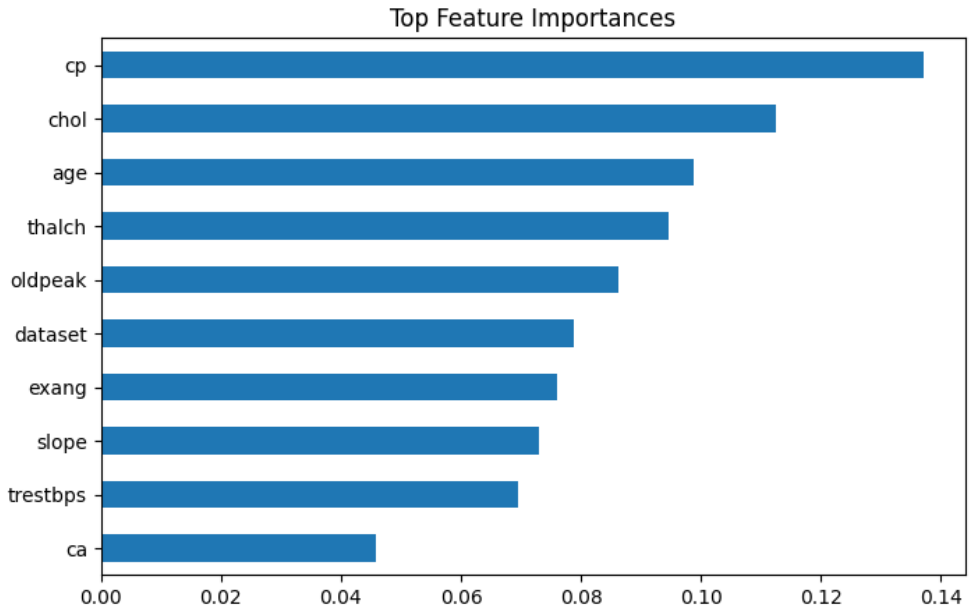

Figure 3: Feature importance bar graph used for feature selection

Tech Stack

- Frontend: React.js
- Backend: Python (Flask)
- Model: RandomForestClassifier (Scikit-learn)
- Data Processing: Pandas, Scikit-learn, Joblib
- Deployment (Local): Flask API served locally with a connected React interface

Conclusion

This component successfully demonstrates the practical application of supervised learning in medical diagnostics. The Random Forest classifier achieved high accuracy and offered a user-friendly real-time prediction tool through a connected web interface. Its ability to identify key health indicators and deliver reliable predictions makes it a valuable model in the context of heart disease risk assessment.

K-Nearest Neighbors (KNN) Model (Janith Withanagamage)

Before model training, exploratory data analysis was conducted to better understand the relationships within the dataset. Key findings included:

i. Class Distribution: The dataset contained a relatively balanced distribution of classes, with 411 negative cases (no heart disease) and 509 positive cases (presence of heart disease).
ii. Feature Correlation: Analysis revealed that 'cp' (chest pain type), 'thalach' (maximum heart rate achieved), and 'ca' (number of major vessels) had the strongest correlations with the target variable.
iii. Age Distribution: Patients with heart disease tended to be older, with a mean age of 56.5 years compared to 52.1 years for those without heart disease.

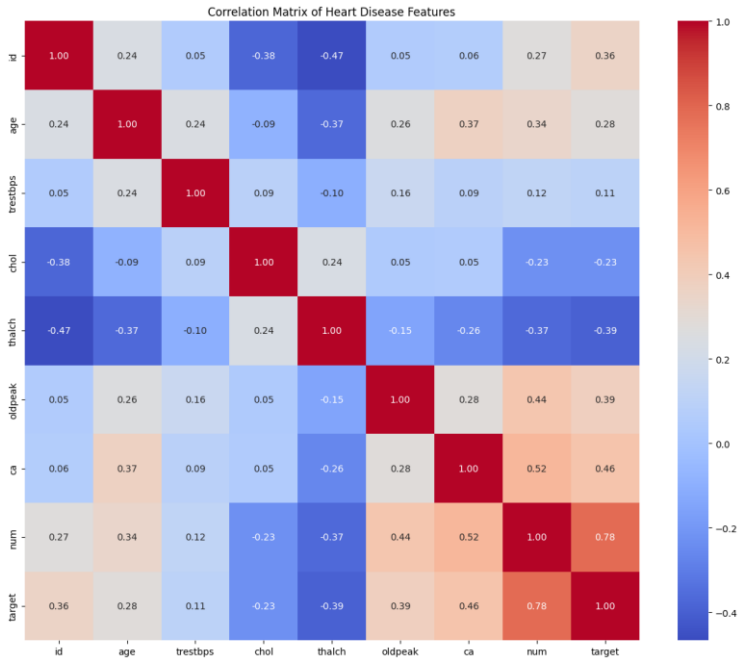

Figure 4: Confusion matrix visualization Features for KNN

Implementation

The initial KNN model was implemented with k=5, a commonly used starting value. The model was built using scikit-learn's KNeighborsClassifier:

python

CopyEdit

knn = KNeighborsClassifier(n_neighbors=5)

knn.fit(X_train, y_train)

Hyperparameter Tuning

To determine the optimal k value, two approaches were used:

i. Error Rate Analysis: The error rate on the test set was computed for k values from 1 to 20.
ii. Cross-Validation: 5-fold cross-validation was used to evaluate model performance more robustly.

The optimal value was determined to be k=9 based on cross-validation results, offering the best tradeoff between bias and variance. Lower k values tended to overfit, while higher values showed reduced sensitivity to local patterns.

Distance Metric Evaluation

Three distance metrics were tested:

i. Euclidean distance (default)
ii. Manhattan distance
iii. Minkowski distance

Euclidean distance yielded the best results with this dataset, achieving the highest average accuracy across cross-validation runs.

Results and Evaluation

The final model (k=9, Euclidean distance) achieved the following metrics on the test set:

- Accuracy: 85.5%
- Precision (for heart disease cases): 88%
- Recall (for heart disease cases): 78%

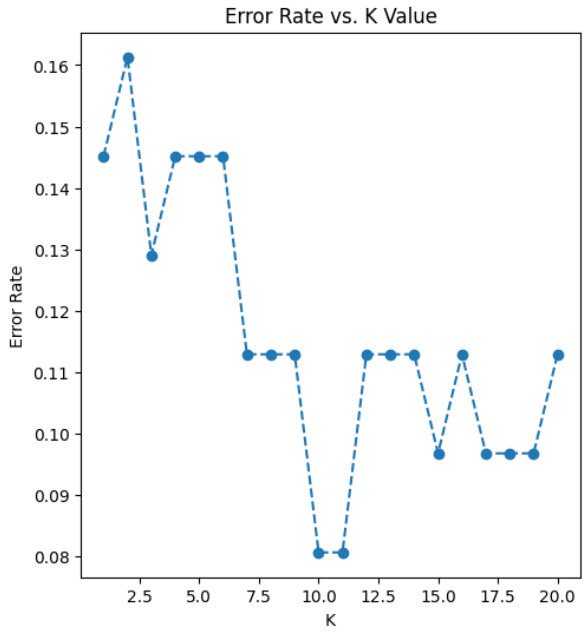

Figure 5: Error Rate vs K Value

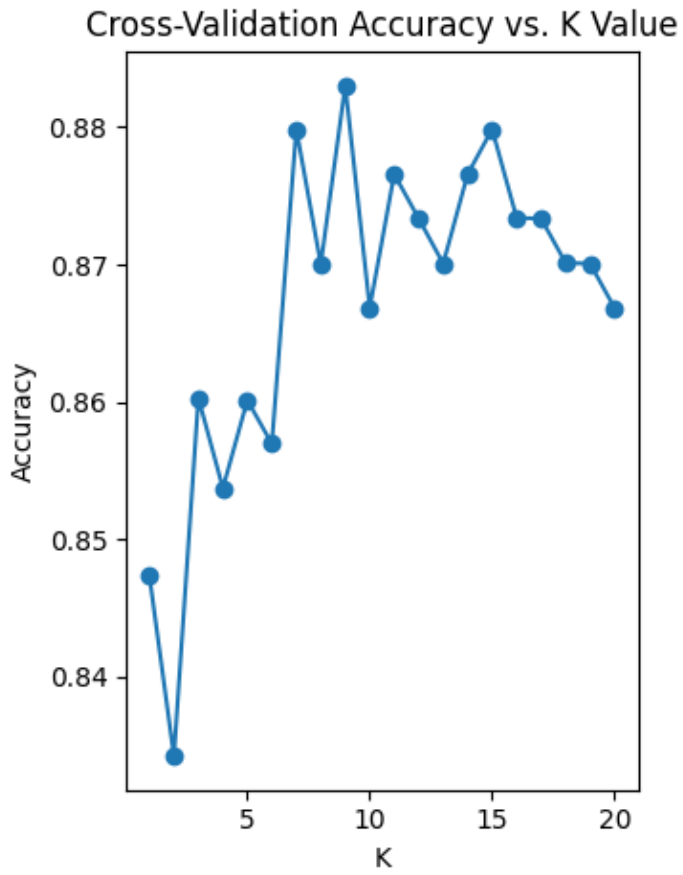

Figure 6: CVA vs K. Value

Confusion Matrix Analysis

The confusion matrix revealed:

- True Positives (TP): 22 – correctly identified heart disease cases
- True Negatives (TN): 31 – correctly identified no heart disease cases
- False Positives (FP): 3 – predicted heart disease when it wasn't present
- False Negatives (FN): 6 – missed heart disease cases

False negatives are a key concern in medical diagnostics. Although the model showed a relatively low false negative rate, reducing this further is an area for future improvement.

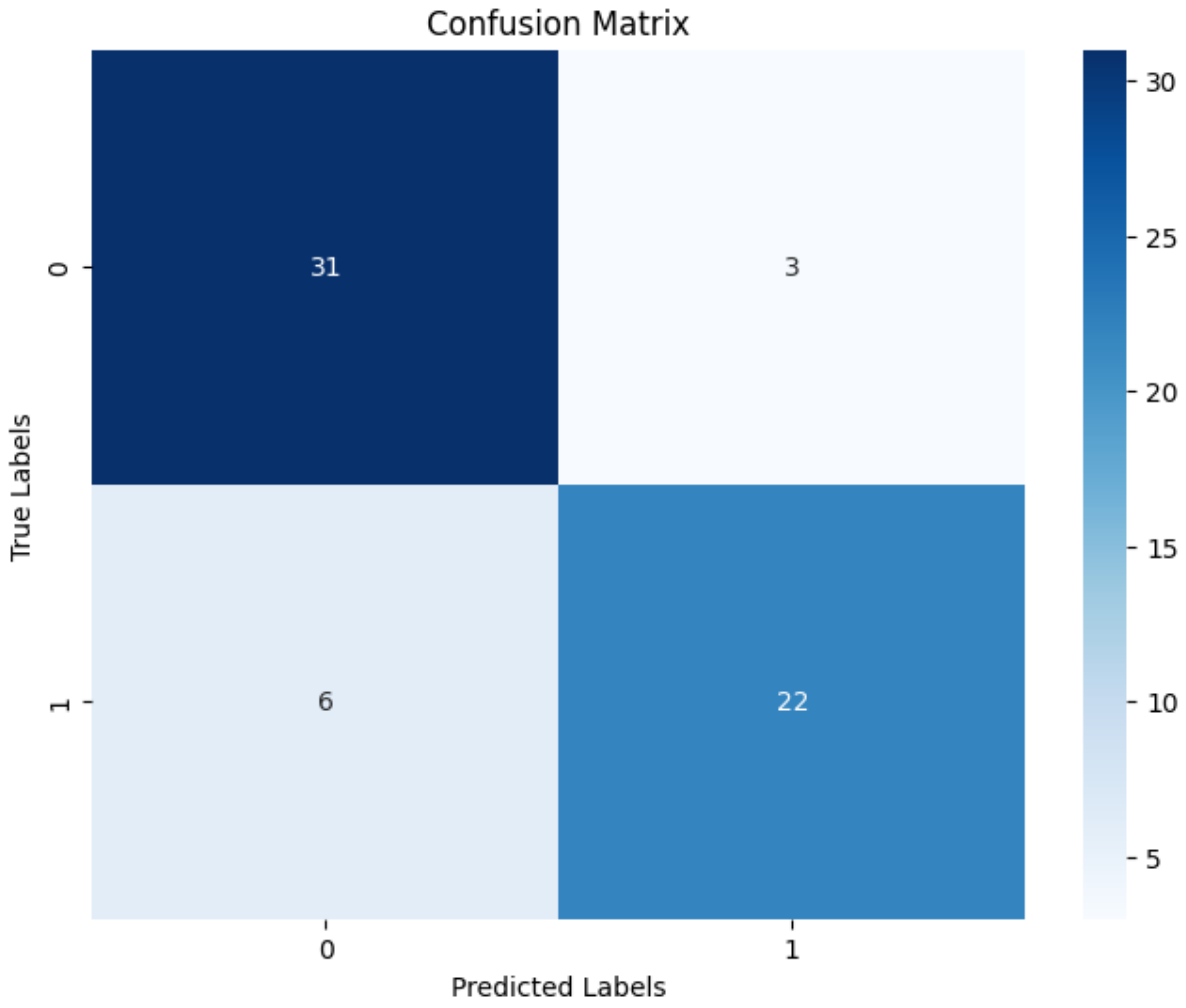

Figure 7: Confusion matrix KNN

ROC Curve and AUC

The model's Receiver Operating Characteristic (ROC) curve showed strong performance, with an Area Under the Curve (AUC) of 0.93. This value indicates

that the model is highly effective in distinguishing between the two classes. An AUC of 1.0 represents perfect classification, while 0.5 indicates no discriminative power.

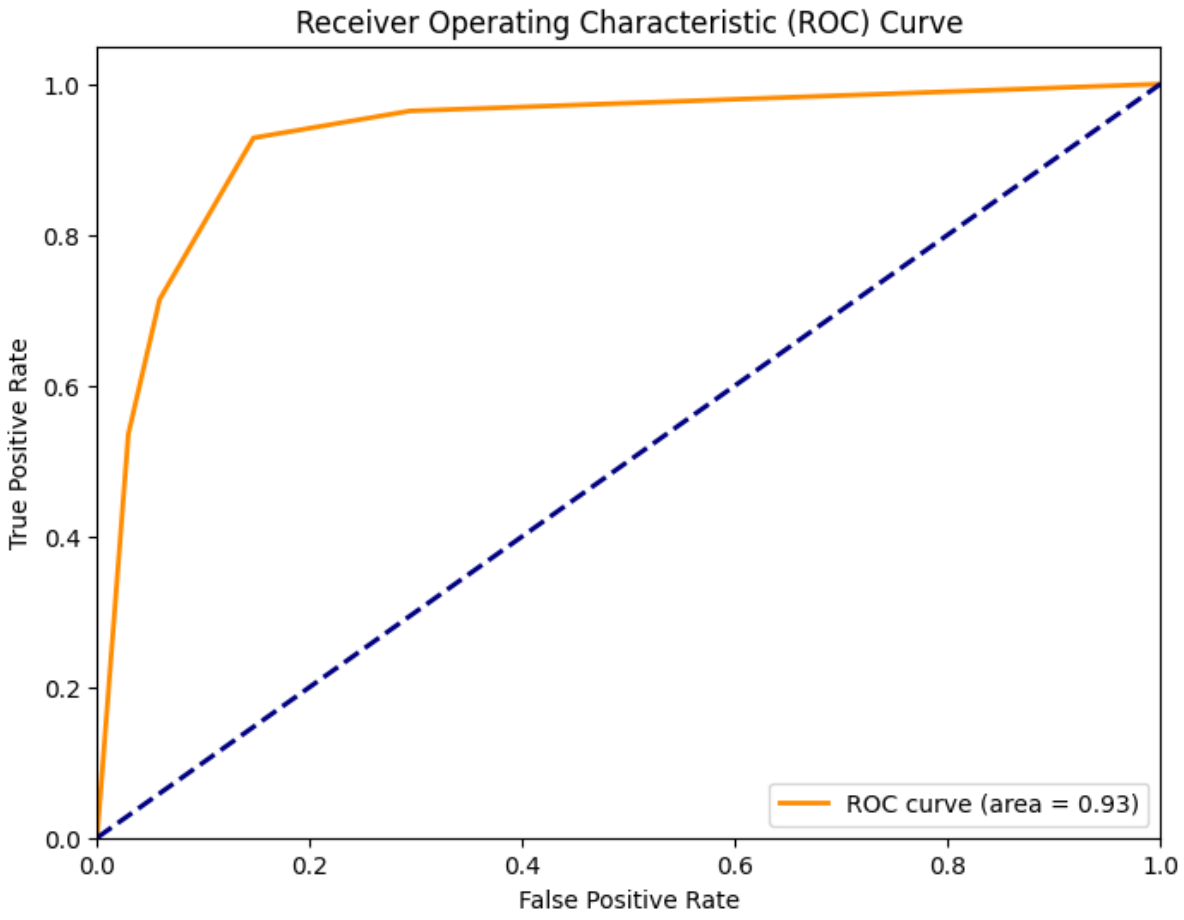

Figure 8: ROC Curve

Results and Comparison

Each of the four supervised learning models was evaluated on the same preprocessed heart disease dataset using accuracy, precision, recall, and F1-score as key metrics. The objective was to determine which model provided the most reliable and clinically useful predictions for heart disease classification.

Below is a comparison of the results obtained from all four models:

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**AUC**
Logistic Regression	61.4%	53.5%	35.9%	34.1%	—
Decision Tree Classifier	78.3%	80%	77%	78%	—
Random Forest Classifier	83.7%	84%	83%	84%	—
K-Nearest Neighbors	**85.5%**	88%	78%	82%	0.93

All models were able to provide meaningful predictions, but their effectiveness varied depending on the complexity of the algorithm and how well it handled the underlying relationships in the data.

- **Logistic Regression**, while simple and interpretable, struggled with convergence and failed to capture complex, non-linear patterns in the data.
- **Decision Tree** showed balanced performance but was more prone to overfitting.
- **Random Forest**, an ensemble method, significantly improved generalization, providing high accuracy and robust feature importance analysis.

- **K-Nearest Neighbors (KNN)** achieved the **highest accuracy (85.5%)** and **highest AUC (0.93)**, suggesting strong discriminative power and reliability in classification.

Best Performing Model

Based on the evaluation metrics, **K-Nearest Neighbors (KNN)** emerged as the best-performing model in this study. It offered the best combination of accuracy, precision, and discriminative ability while maintaining reasonable interpretability and low false negative rates. These characteristics are particularly important in a healthcare setting, where missing a positive diagnosis (false negative) can have serious consequences.