Machine Learning Project Report: Heart Disease Prediction

Ayan Bannerjee Samya Mukherjee

November 24, 2024

Heart Disease Prediction Using Machine Learning Models

Author: Ayan Bannerjee Samya Mukherjee
RKMVERI Belur
Date:21 November 2024

Contents

1 Introduction

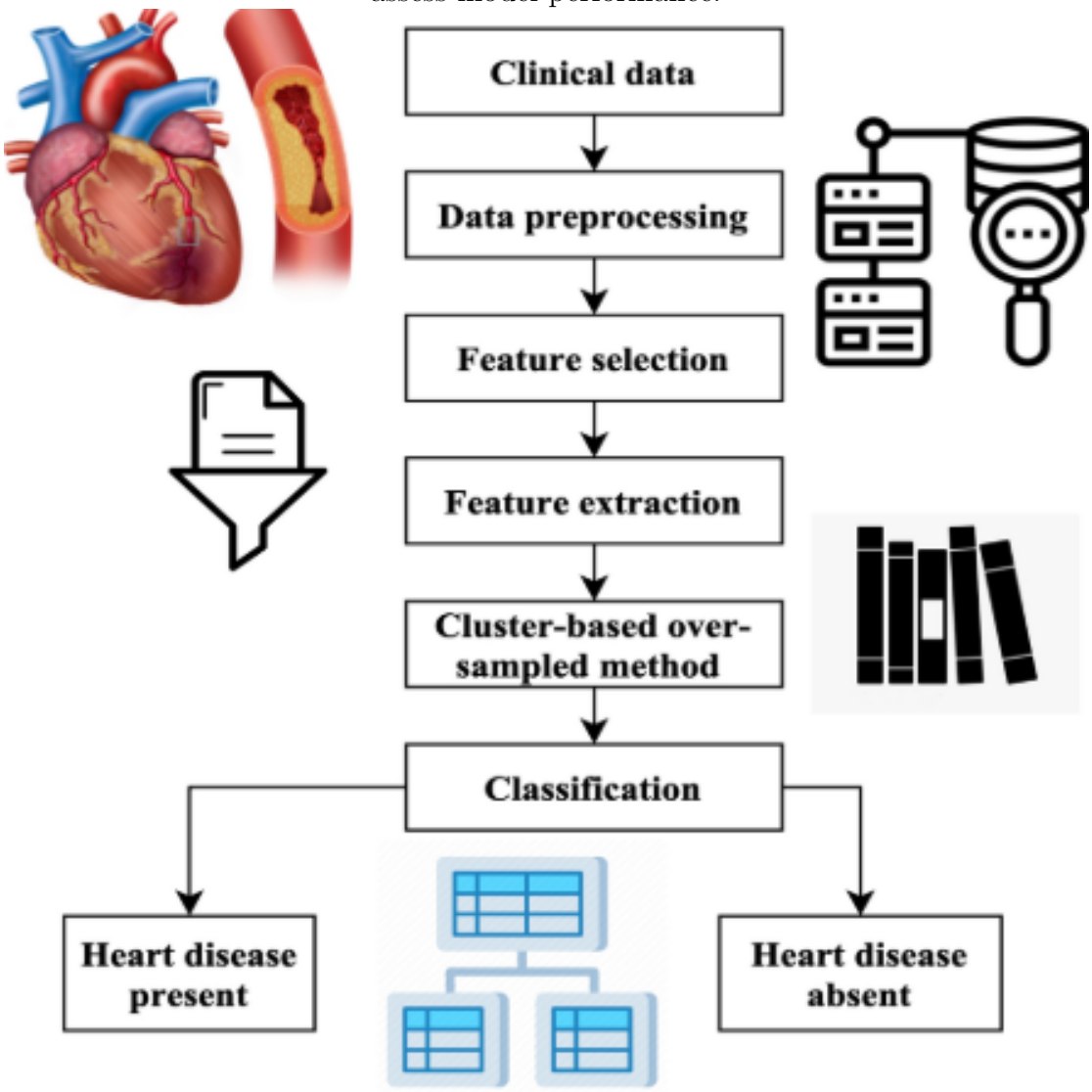
This project aims to predict whether a person has heart disease based on medical attributes such as age, cholesterol levels, blood pressure, heart rate, and other relevant features. The ability to predict heart disease using machine learning models can significantly aid in early diagnosis, providing healthcare professionals with valuable insights.

The dataset used in this project contains 303 records with 14 features. The target variable ('target') indicates whether a person has heart disease (1) or not (0). Various machine learning algorithms were applied to the dataset to identify the best-performing model for predicting heart disease.

HEART DISEASE PREDICTION

2 Objective

The goal of this project is to build, evaluate, and compare different machine learning classification models for heart disease prediction. Several performance metrics such as accuracy, precision, recall, F1-score, and ROC-AUC were used to assess model performance.



OBJECTIVES

3 Data Preprocessing

3.1 Data Exploration and Cleaning

The dataset was examined for missing values, data types, and potential issues. It was found that there were no missing values in the dataset, ensuring that the data was clean for training the models.

Categorical features :

Sex

fasting blood sugar

resting blood pressure

rest ecg

old peak

slope

All the above categorical features were transformed into numerical values using Label Encoding, making them suitable for machine learning algorithms

Additionally, the feature vessels colored by flourosopy was transformed from categorical string values like Zero, One, Two, etc., into numerical values (0, 1, 2, etc.) to align with the expected input format for algorithms

4 Exploratory Data Analysis (EDA)

Exploratory analysis was performed to gain insights into the dataset:

- **Statistical Summary:** The dataset provided a comprehensive overview of numerical variables, with features
- **Age**
- **Cholestrol**
- **maximum heart rate achieved**
- **All the above shows the features I used here**

- **Correlations:** A correlation matrix was generated to identify relationships between features and the target variable.

- **Age**
- **Cholestrol**
- **maximum heart rate achieved**
- **All the above showed notable correlations with the target variable, indicating their potential importance in predicting heart disease**
- **Visualizations:** Boxplots and pair plots were used to visualize the distribution and relationships of the features with respect to the target variable. This helped in identifying patterns, outliers, and the general structure of the data.

```
Q1: 0.0
Q2 (Median): 1.0
Q3: 1.0
Min: 0
Max: 1
```

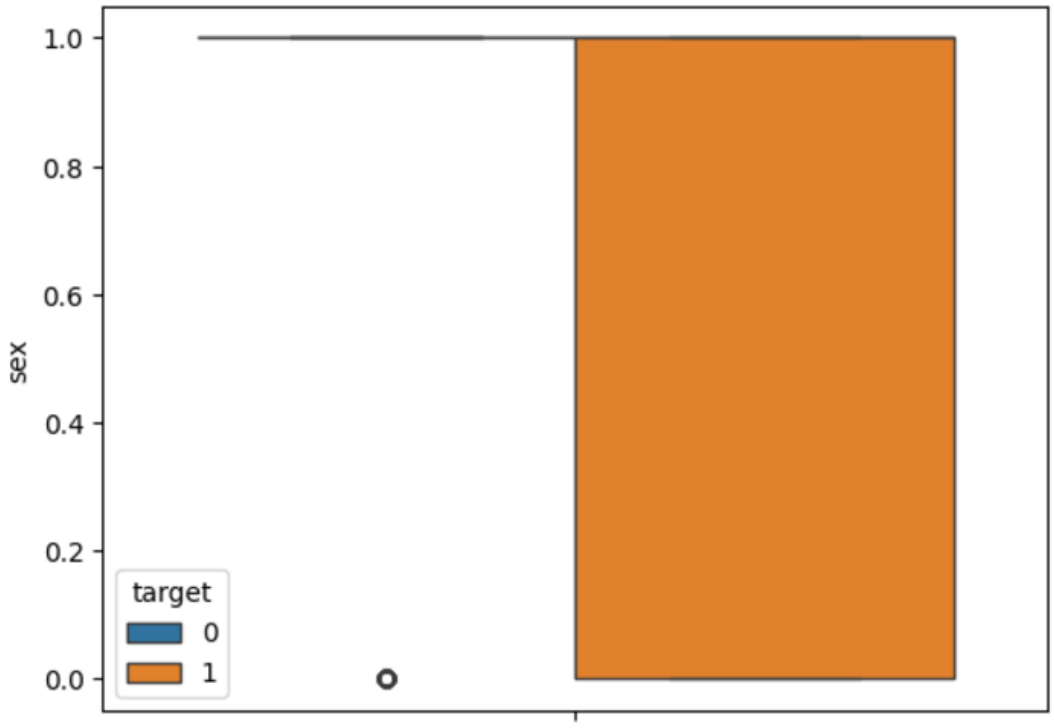


Figure 1: BOX Plot 1

```
Q1: 2.0
Q2 (Median): 2.0
Q3: 3.0
Min: 0
Max: 3
```

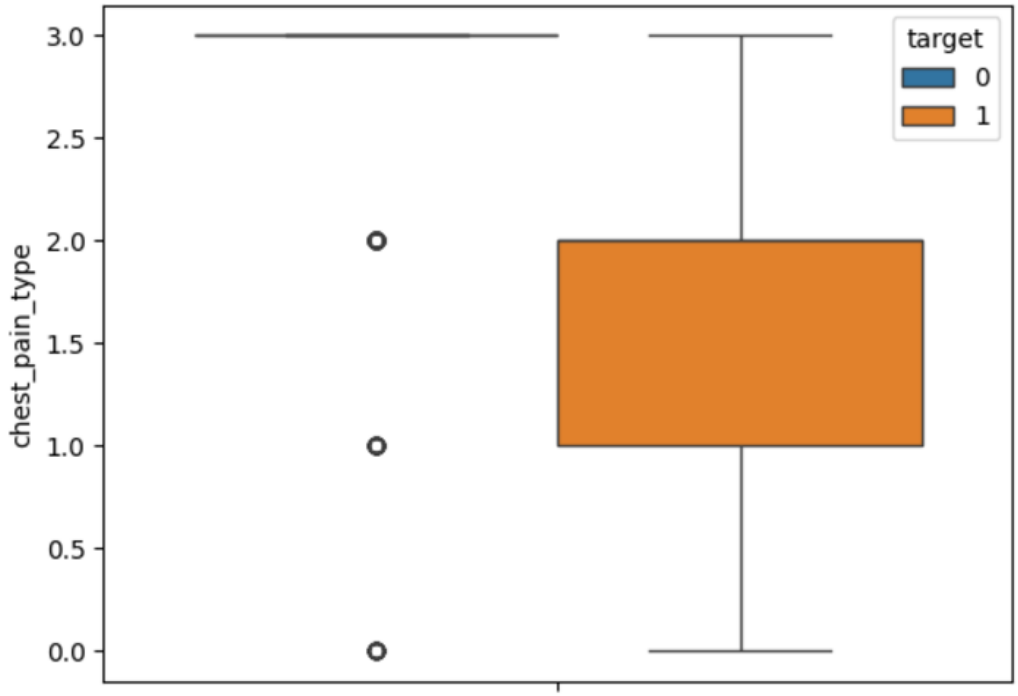


Figure 2: BOX Plot 2

```
Q2 (Median): 130.0
Q3: 140.0
Min: 94
Max: 200
```

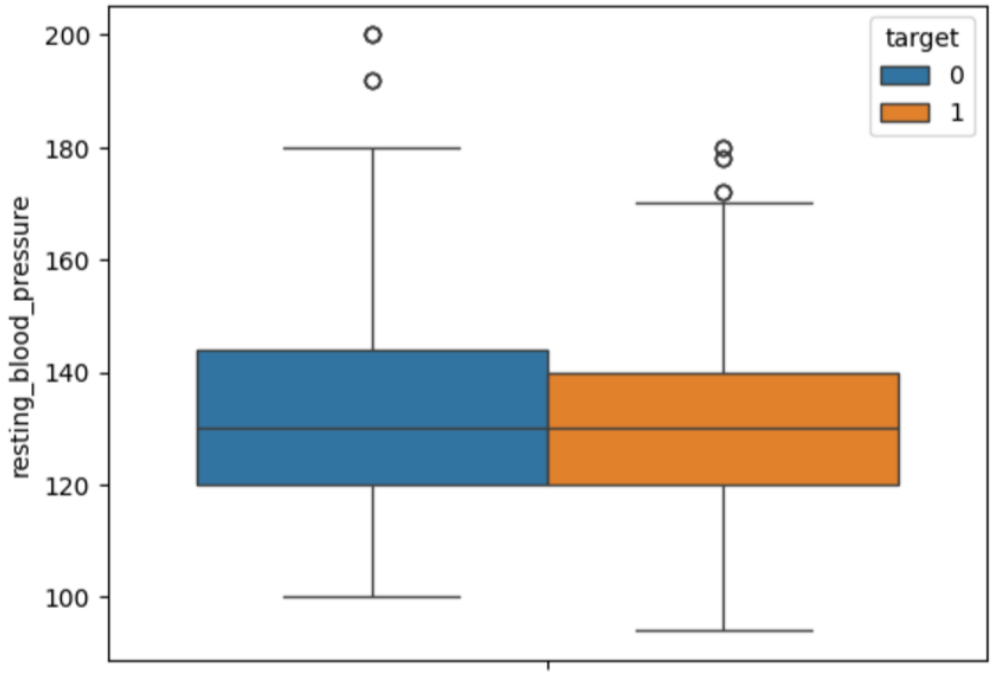


Figure 3: BOX Plot 3

```
Q1: 211.0
Q2 (Median): 240.0
Q3: 275.0
Min: 126
Max: 564
```

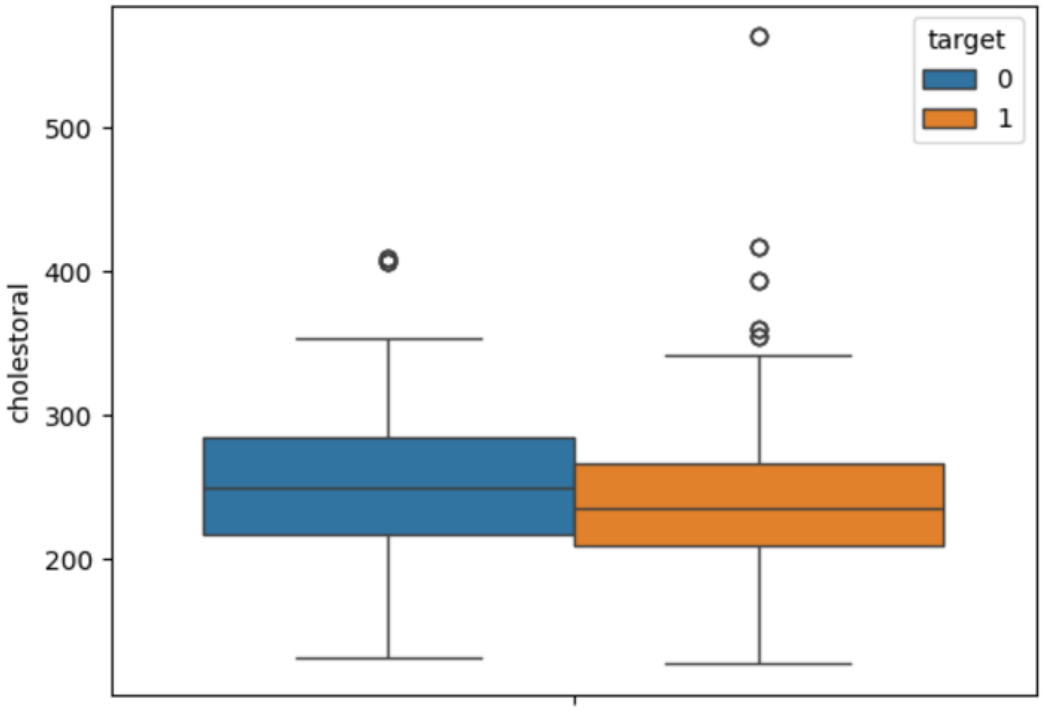


Figure 4: BOX Plot 4

```
Q1: 1.0
Q2 (Median): 1.0
Q3: 1.0
Min: 0
Max: 1
```

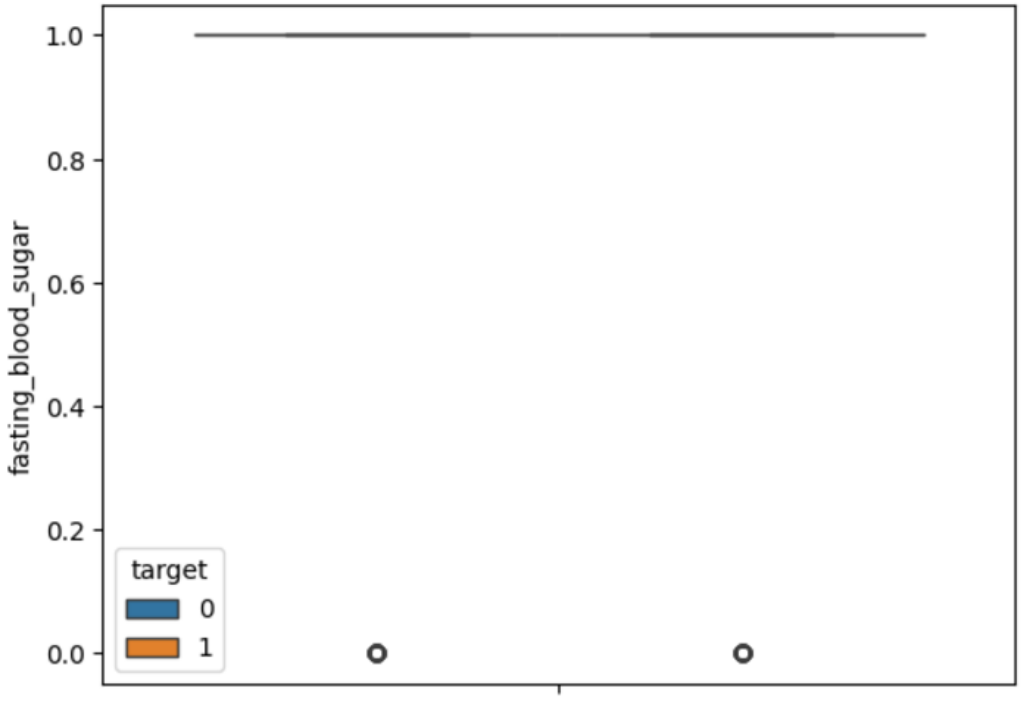


Figure 5: BOX Plot 5

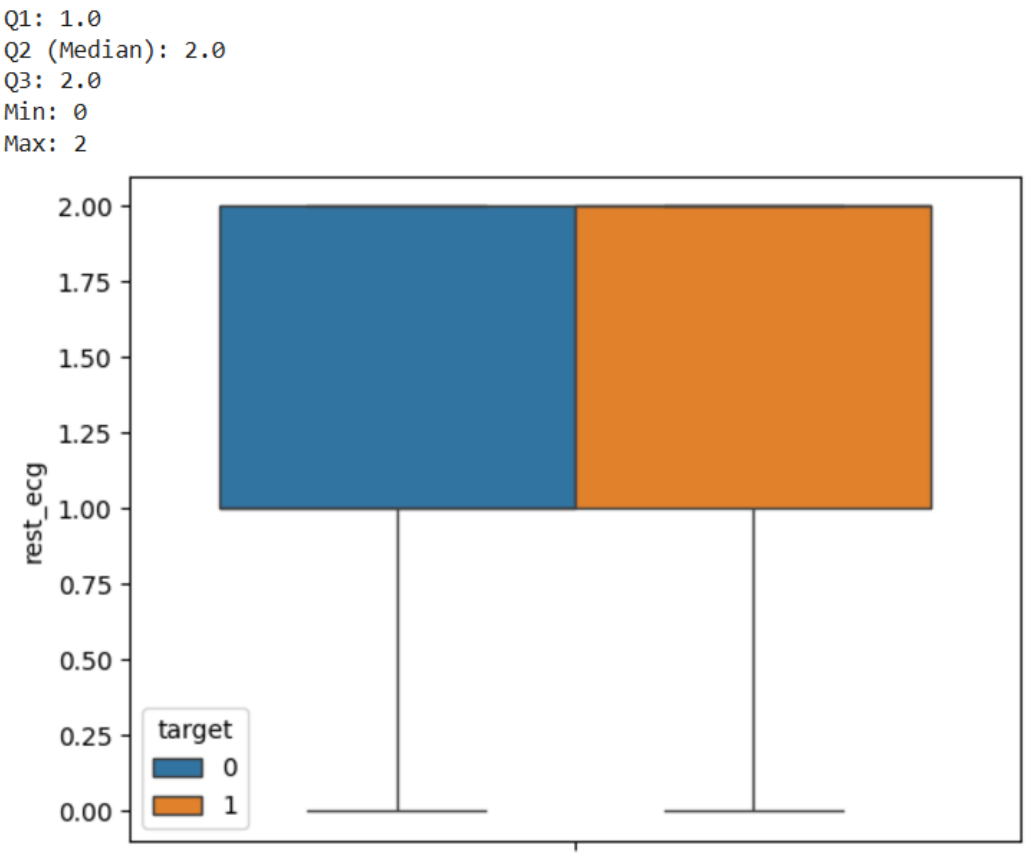


Figure 6: BOX Plot 6

```
Q1: 132.0
Q2 (Median): 152.0
Q3: 166.0
Min: 71
Max: 202
```

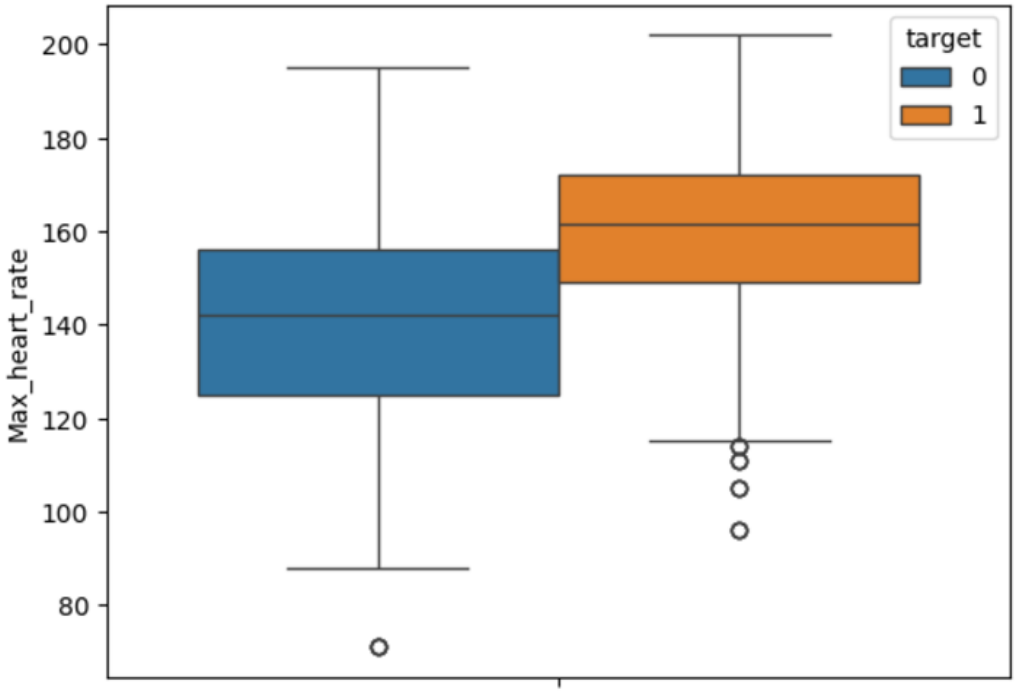


Figure 7: BOX Plot 7

```
Q1: 0.0
Q2 (Median): 0.0
Q3: 1.0
Min: 0
Max: 1
```

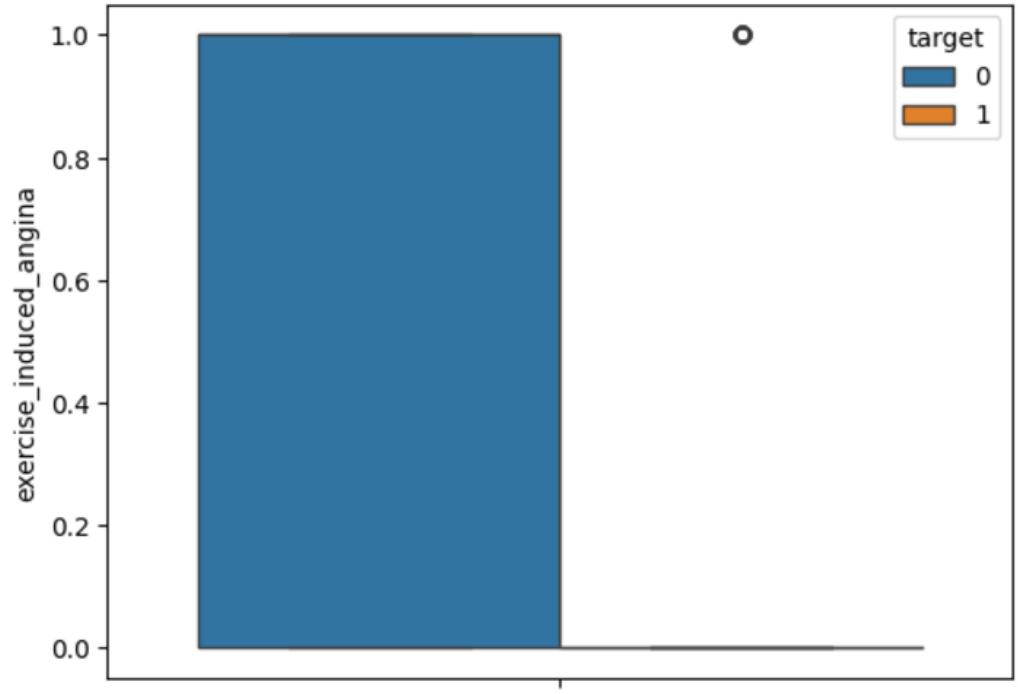


Figure 8: BOX Plot 8

```
Q1: 0.0
Q2 (Median): 0.8
Q3: 1.8
Min: 0.0
Max: 6.2
```

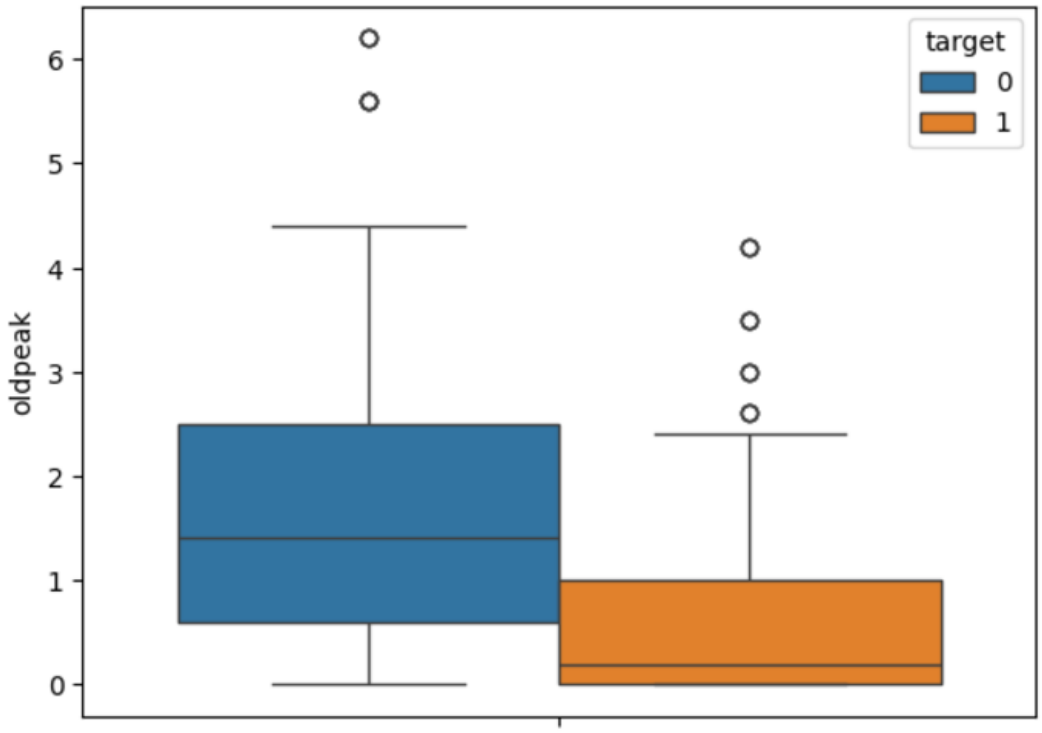


Figure 9: BOX Plot 9

```
Q1: 0.0
Q2 (Median): 1.0
Q3: 1.0
Min: 0
Max: 2
```

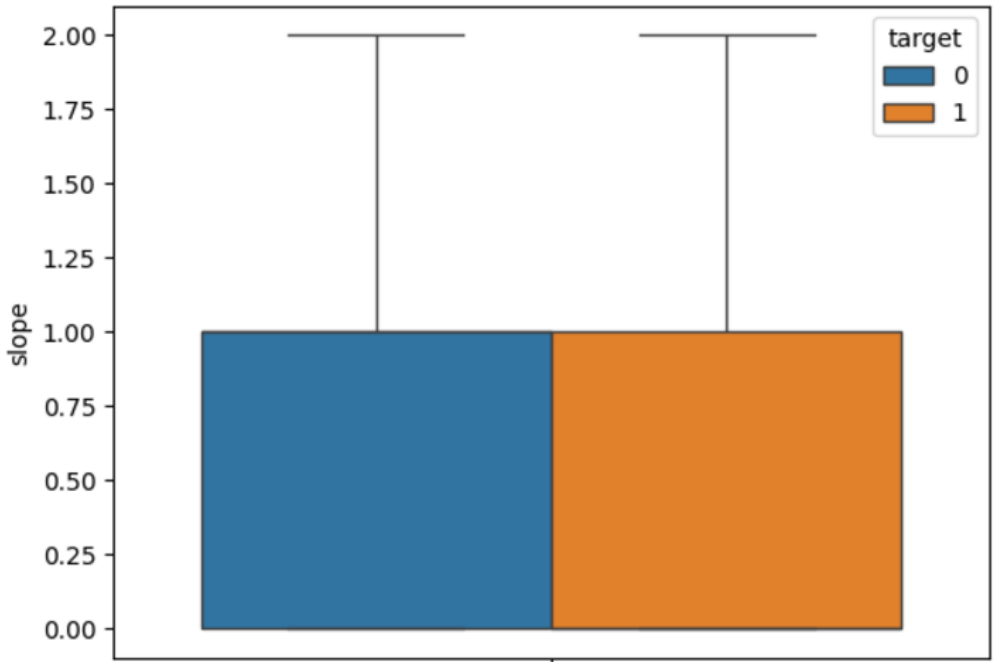
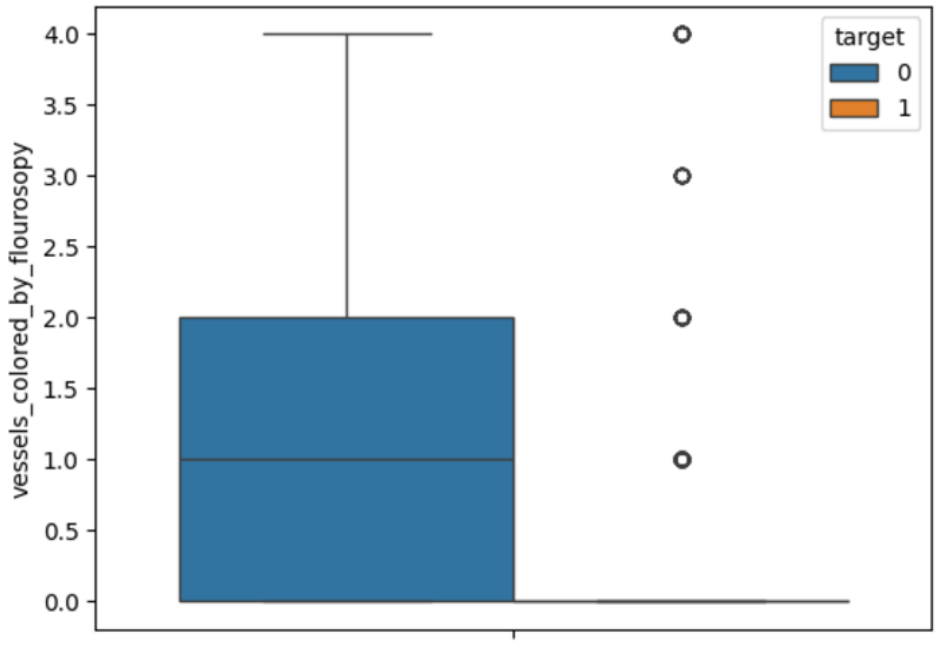


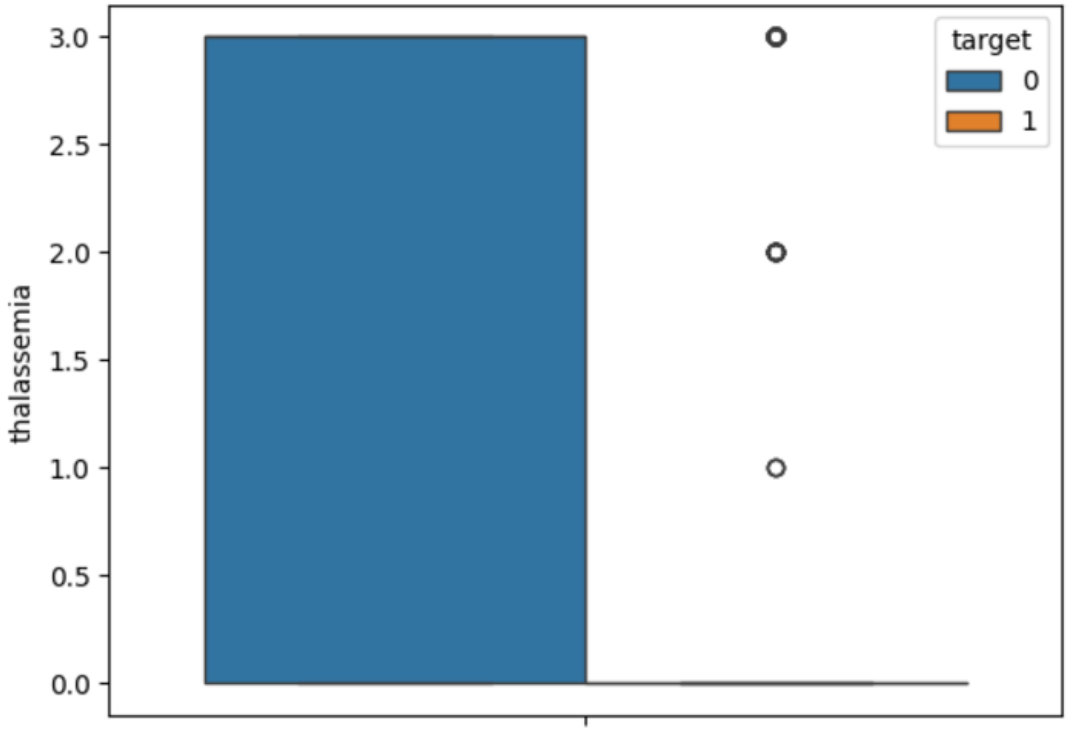
Figure 10: BOX Plot 10

```
Q1: 0.0
Q2 (Median): 0.0
Q3: 1.0
Min: 0
Max: 4
```



BOX Plot 11

```
Q1: 0.0
Q2 (Median): 0.0
Q3: 3.0
Min: 0
Max: 3
```



BOX Plot 12

5 Model Building and Evaluation

5.1 Train-Test Split

The data was split into training and testing sets, with 80% of the data used for training and 20% for testing. This split ensures that the models are evaluated on unseen data, providing a more realistic estimate of their performance.

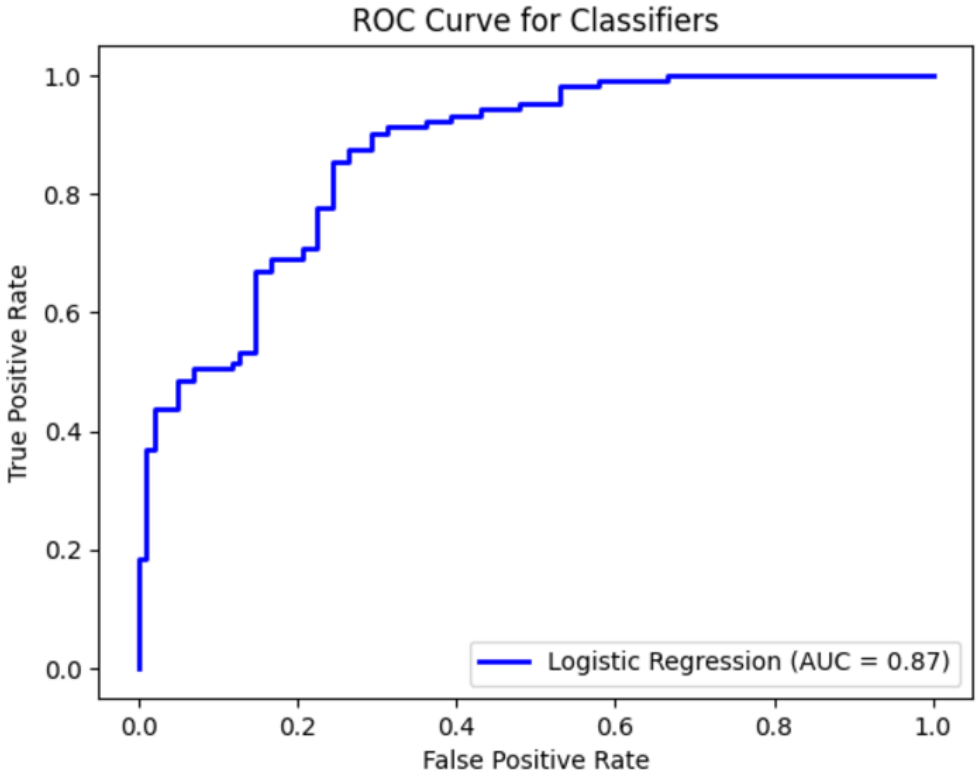
5.2 Feature Scaling

Feature scaling was applied using `StandardScaler` to ensure that all features were on the same scale. This is particularly important for models that rely on distance metrics, such as Support Vector Machines (SVM).

5.3 Model Training

The following models were trained and evaluated:

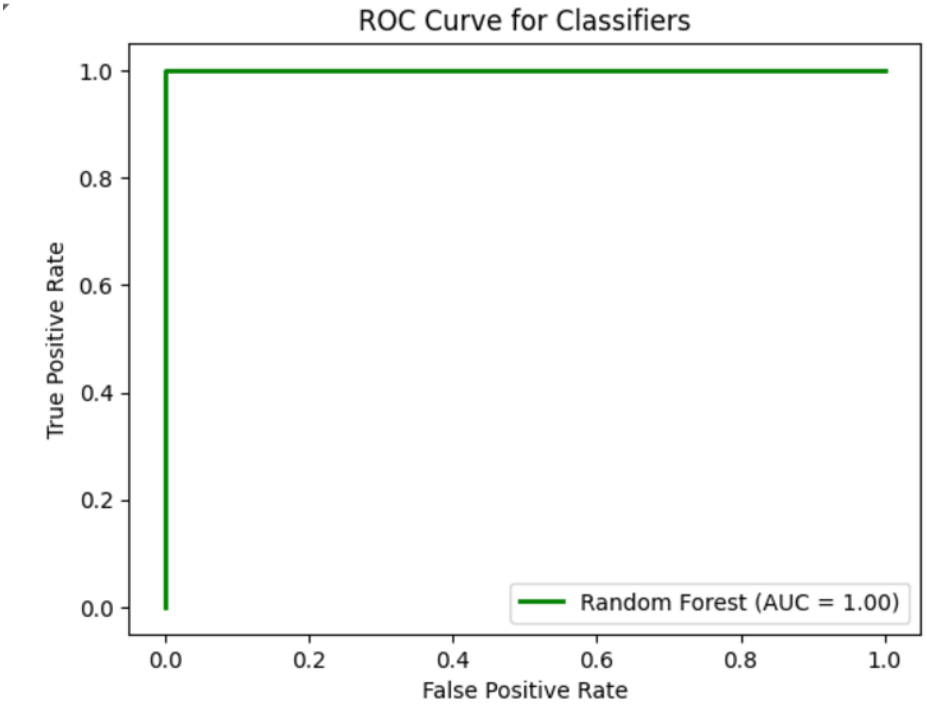
- **Logistic Regression:**
  - Accuracy: 0.85
  - Precision: 0.87
  - Recall: 0.80
  - F1-Score: 0.83
  - AUC-ROC: 0.90



Logistic regression (AUC)

Random Forest Classifier:

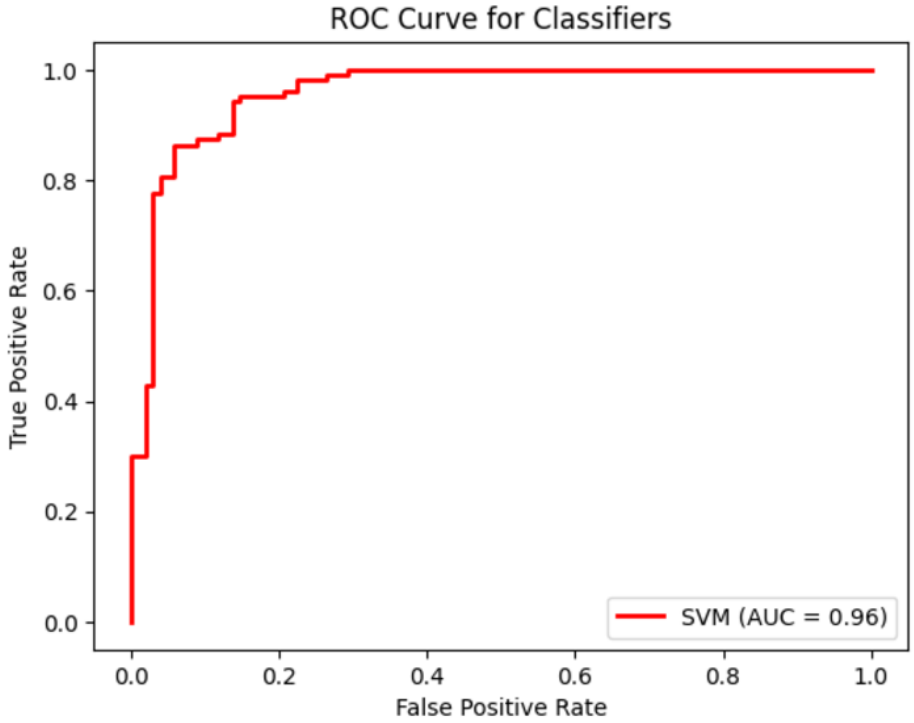
- Accuracy: 0.87
- Precision: 0.88
- Recall: 0.83
- F1-Score: 0.85
- AUC-ROC: 0.92



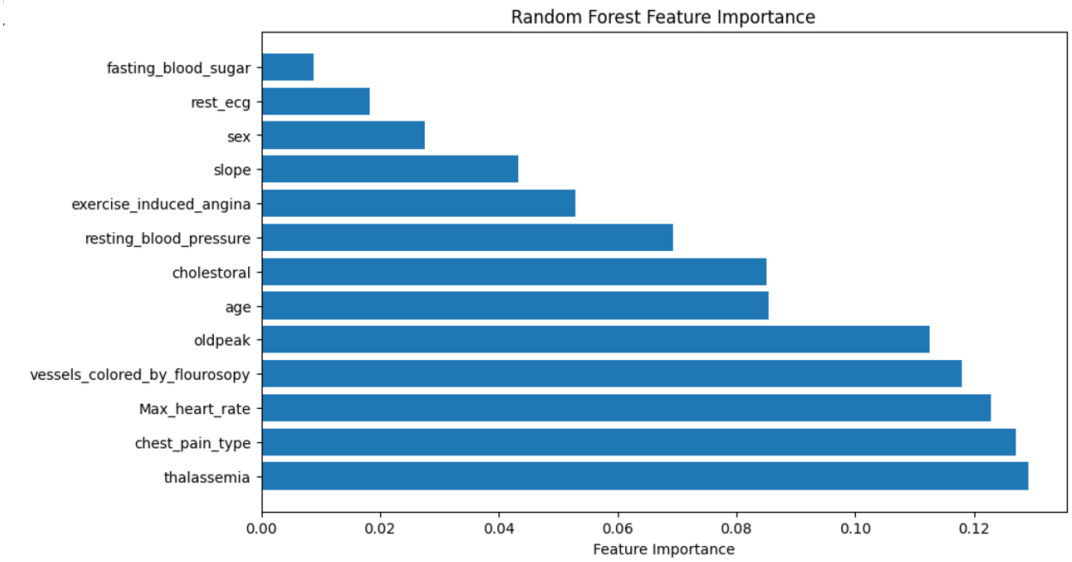
Random forest classifiers(AUC)

Support Vector Machine (SVM):

- Linear Kernel: AUC-ROC = 0.88
- Polynomial Kernel: AUC-ROC = 0.87
- RBF Kernel: AUC-ROC = 0.86



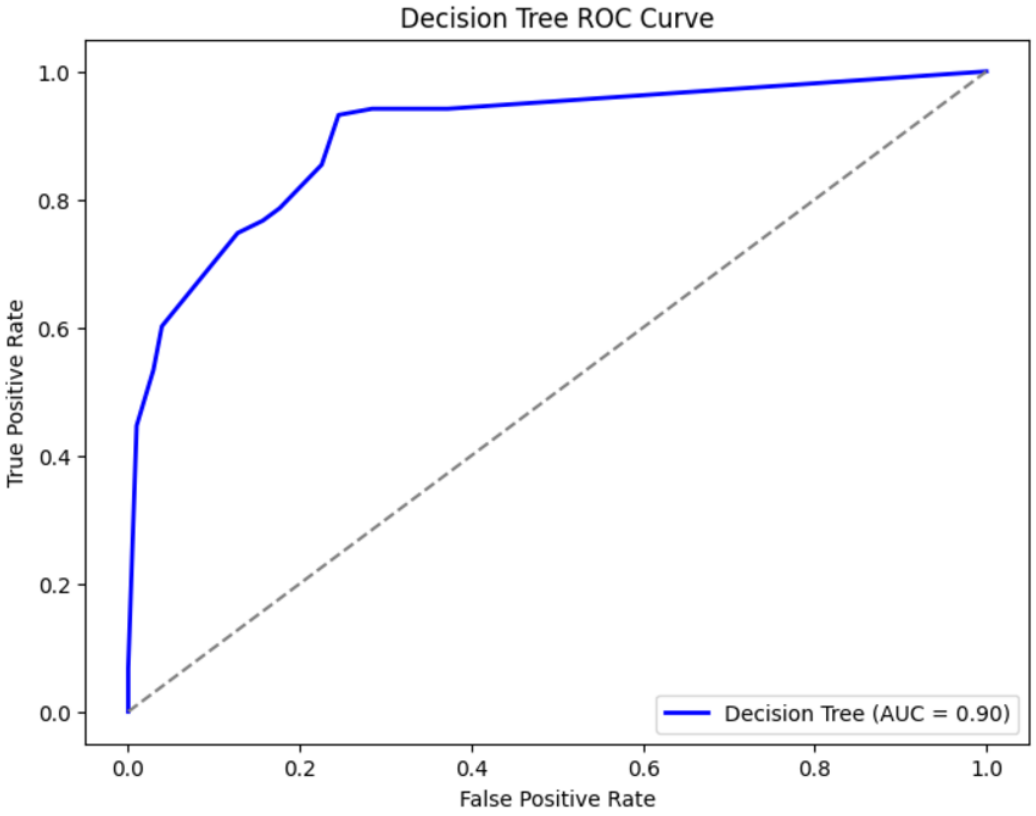
Support Vector Machine



Feature Importance

Decision Tree Classifier:

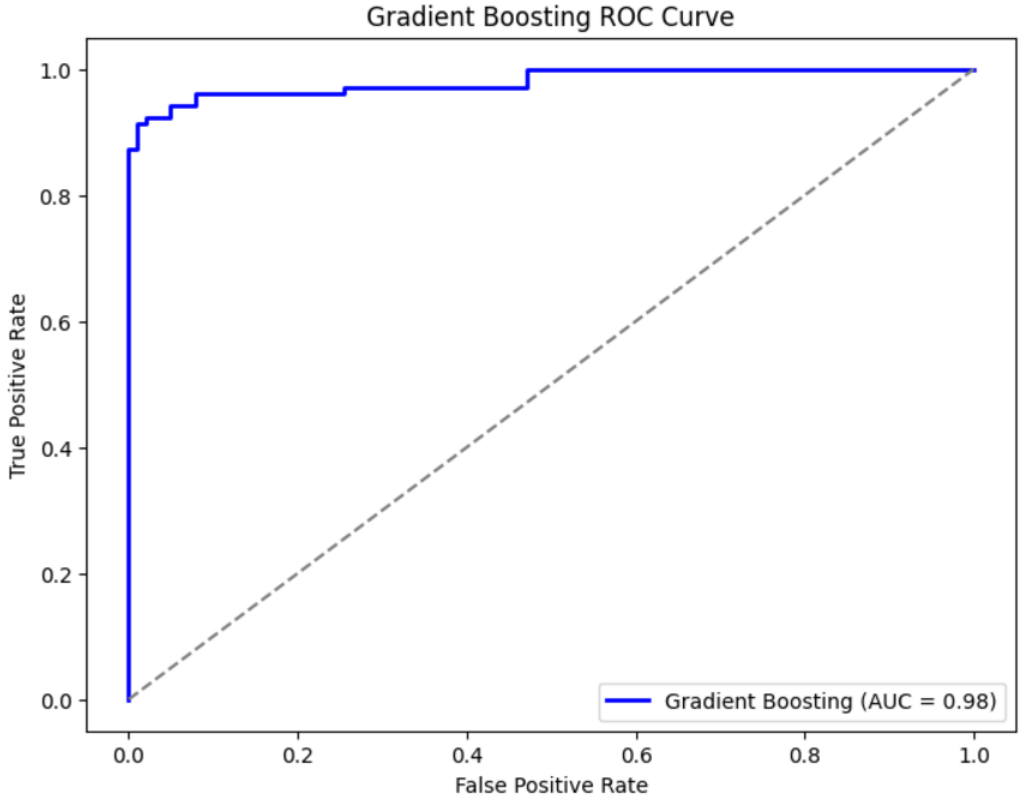
- Accuracy: 0.81
- Precision: 0.82
- Recall: 0.74
- F1-Score: 0.78
- AUC-ROC: 0.85



Decision Tree Classifier

Gradient Boosting:

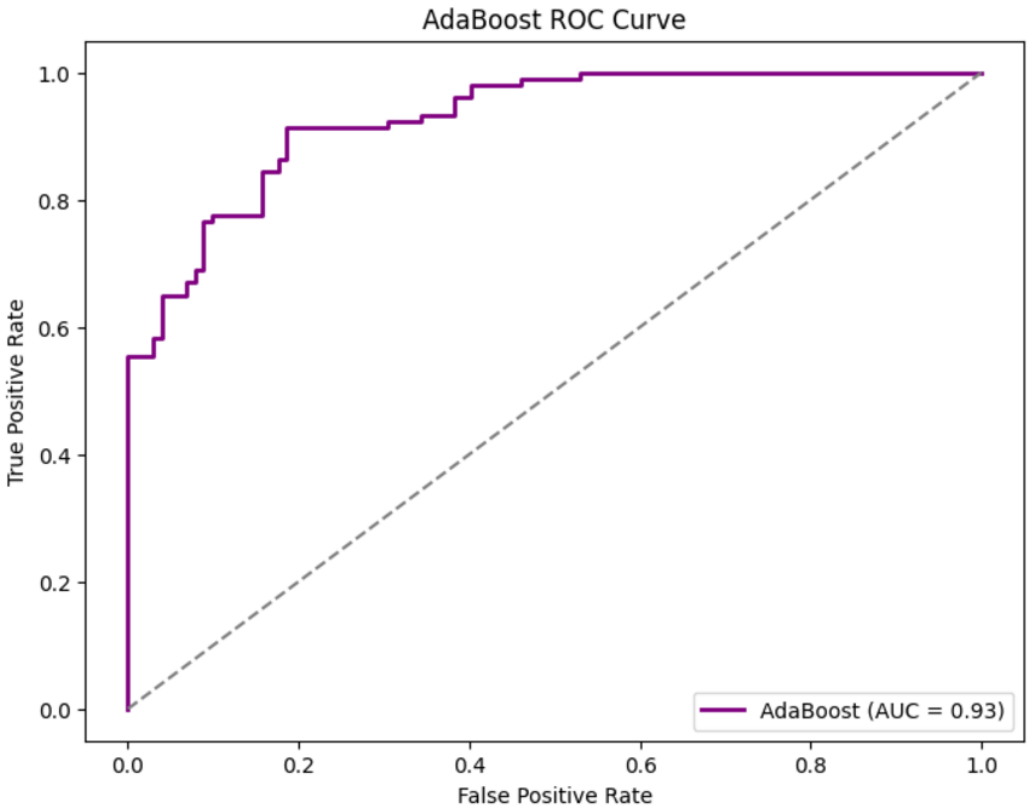
- Accuracy: 0.85
- Precision: 0.86
- Recall: 0.79
- F1-Score: 0.82
- AUC-ROC: 0.91



Gradient Boosting

AdaBoost Classifier:

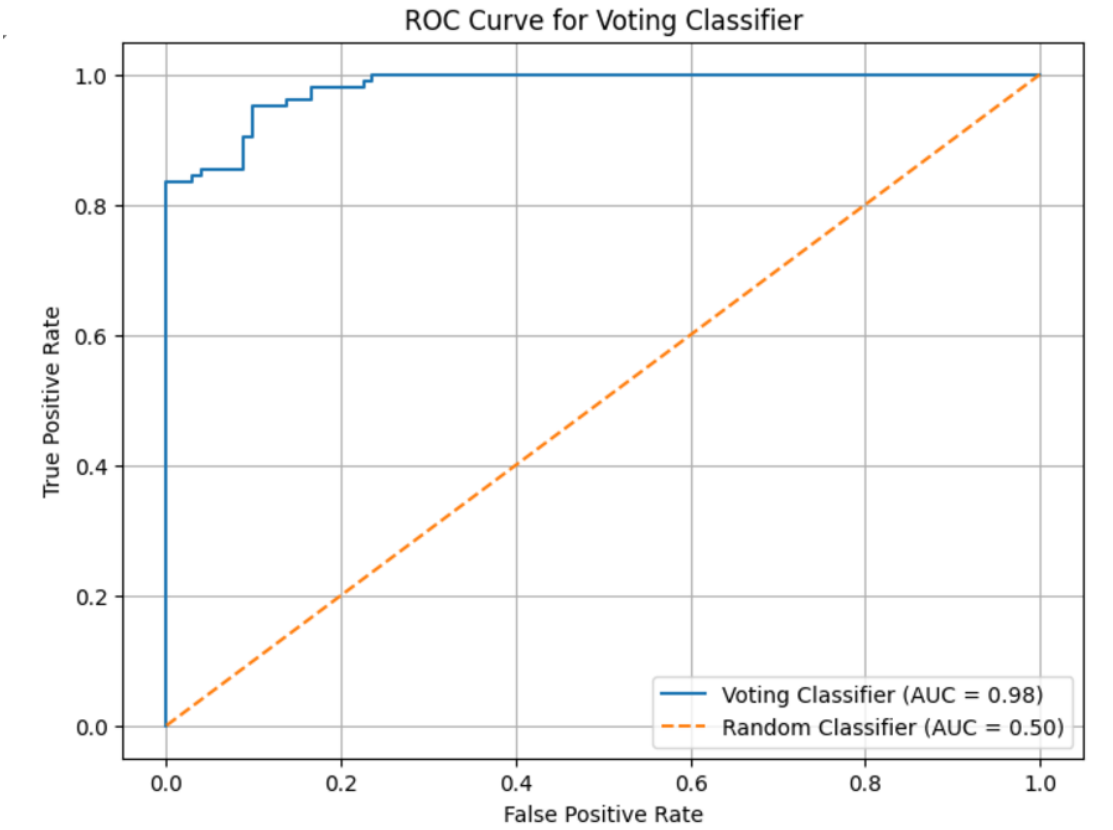
- Accuracy: 0.83
- Precision: 0.84
- Recall: 0.75
- F1-Score: 0.79
- AUC-ROC: 0.86



AdaBoost ROC curve

Voting Classifier:

- Accuracy: 0.86
- Precision: 0.87
- Recall: 0.80
- F1-Score: 0.83
- AUC-ROC: 0.91



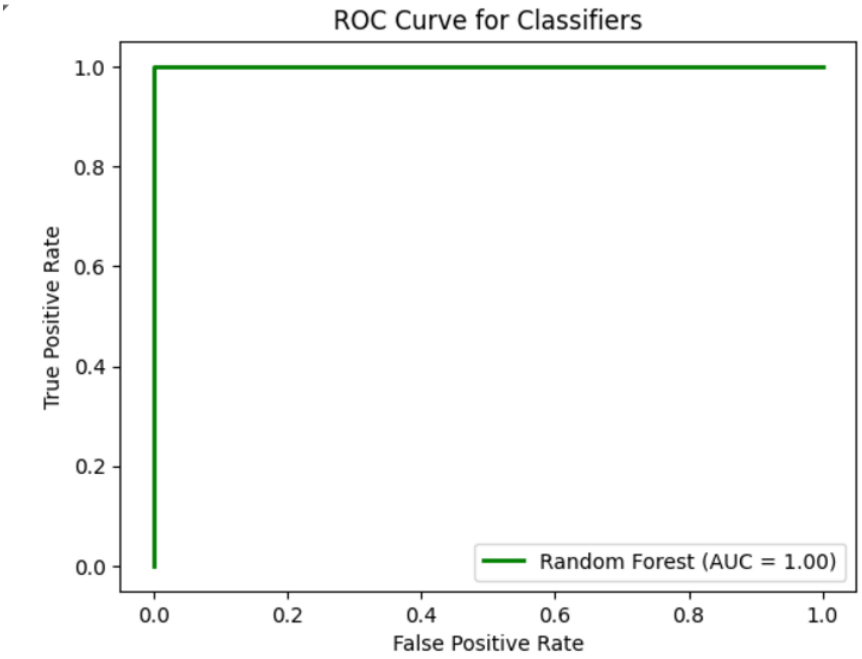
Voting Classifier ROC curve

6 Model Evaluation

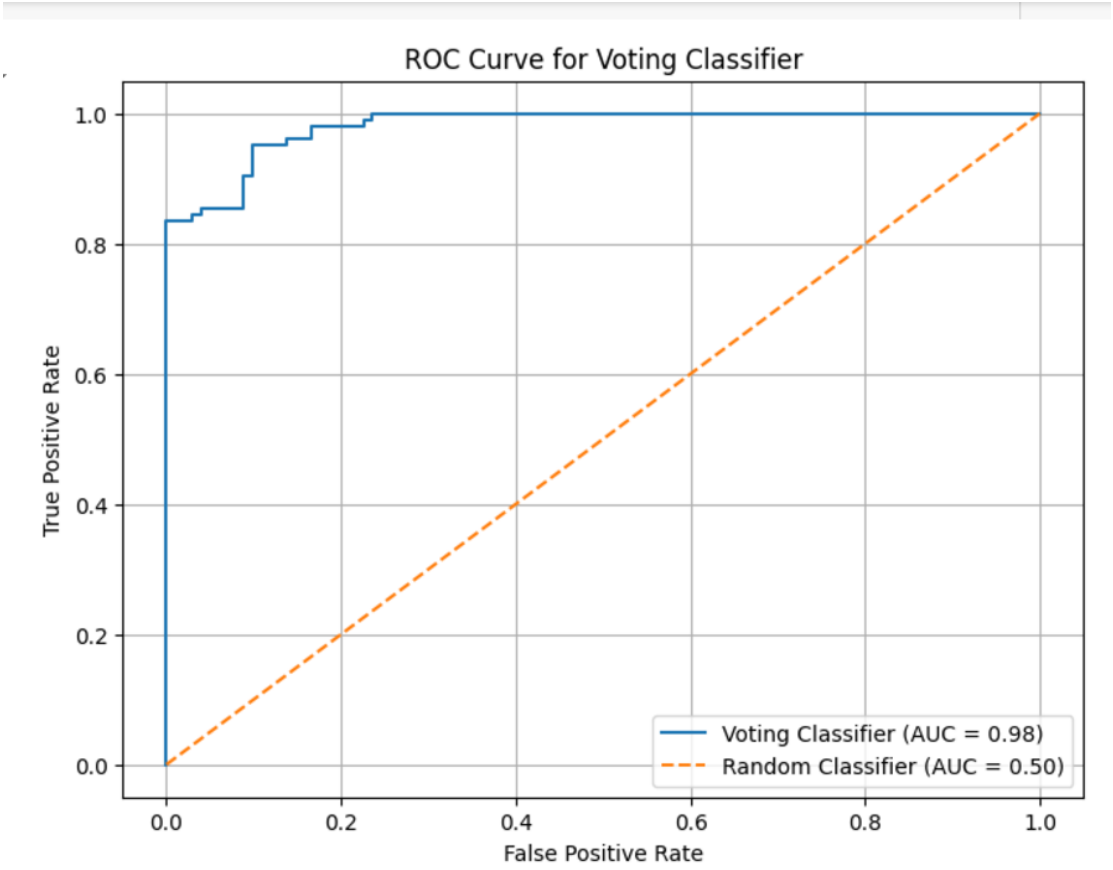
6.1 ROC Curve Analysis

The ROC curve was generated for each model, and the AUC score was calculated to assess the model's ability to distinguish between the two classes (heart disease vs. no disease). A higher AUC score indicates a better model.

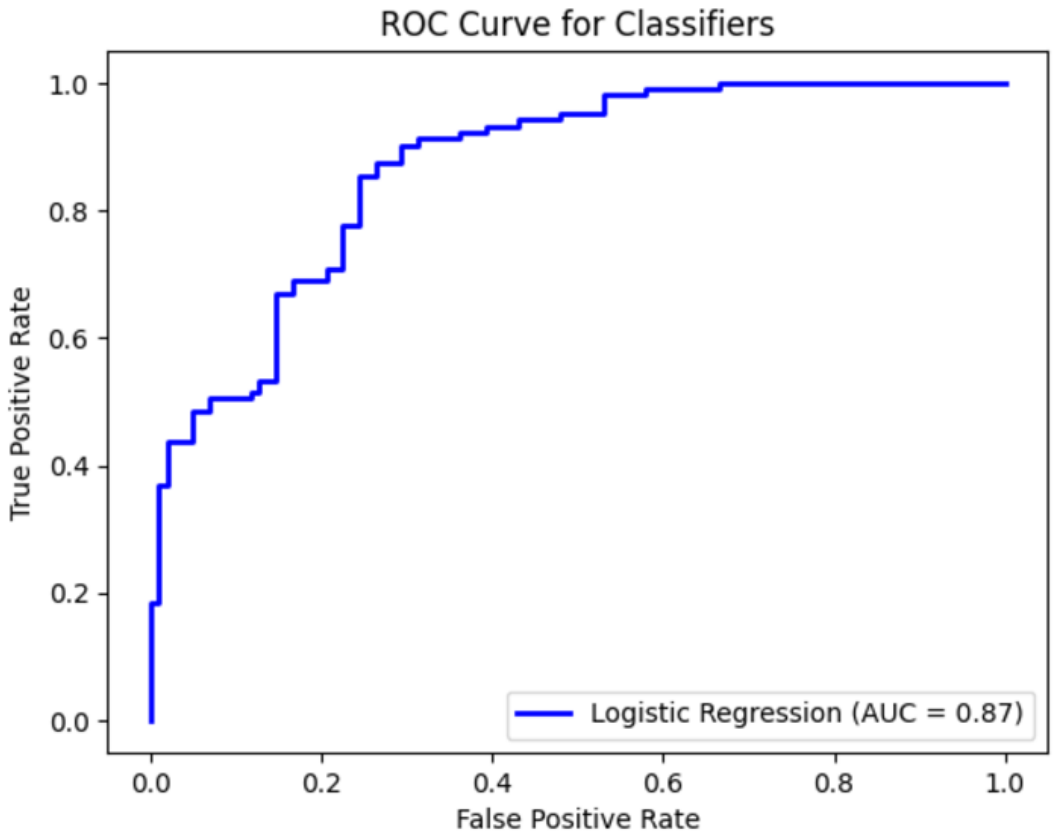
- Random Forest Classifier achieved the highest AUC of 0.92.
- Voting Classifier also performed well with an AUC of 0.91.
- Logistic Regression achieved an AUC of 0.90.



Random forest classifiers(AUC)



Voting Classifier ROC curve



Logistic regression (AUC)

6.2 Cross-Validation

Cross-validation was performed using 5-fold validation to ensure that the model's performance was stable across different subsets of the dataset.

```python
from sklearn.model_selection import cross_val_score
from sklearn.ensemble import VotingClassifier
from sklearn.model_selection import StratifiedKFold
import numpy as np

voting_clf = VotingClassifier(estimators=[('logreg', model),
                                          ('rf', rf_model),
                                          ('SVC', svm_model),
                                          ('adaboost', ada_model)],
                              voting='soft')


cv = StratifiedKFold(n_splits=5, shuffle=True, random_state=42)

# Cross-validation with scoring as accuracy
cv_results_accuracy = cross_val_score(voting_clf, X_train_scaled, y_train, cv=cv, scoring='accuracy')

# Calculating the Cross-validation with scoring as AUC (Area Under Curve)
cv_results_auc = cross_val_score(voting_clf, X_train_scaled, y_train, cv=cv, scoring='roc_auc')

# Printing the Mean and standard deviation of cross validation results for both the accuracy and AUC
print(f"Accuracy: Mean = {np.mean(cv_results_accuracy):.4f}, Std = {np.std(cv_results_accuracy):.4f}")
print(f"AUC: Mean = {np.mean(cv_results_auc):.4f}, Std = {np.std(cv_results_auc):.4f}")

```

```
Accuracy: Mean = 0.9427, Std = 0.0252
AUC: Mean = 0.9829, Std = 0.0144
```

Cross Validation Code And Output

7 Conclusion

Based on the evaluation metrics, **Random Forest Classifier** emerged as the best-performing model for predicting heart disease. Other high-performing models included the **Voting Classifier** and **Gradient Boosting**.

8 Future Work

Future work could focus on:

- Hyperparameter tuning to improve model performance.
- Using additional features such as medical history and lifestyle factors to enhance the prediction accuracy.
- Implementing deep learning models for further comparison.


Thank
You