

# **TEAM 5**

## Heart Disease prediction

TA: Salma Elgayar

### Members:

سهيله خالد عبد الرشيد محمد

2023170280

سما ايمن منير محمد

2023170267

جني وليد سعيد فهمي

2023170166

يوسف السيد حسنين السيد

2023170714

مروان صبحي محمد

2023170570

نوران اسامه محمد علي

2023170680

فرح محمد مصطفى يونس

2023170426

# Project Overview:

This project aims to develop an AI-based system to predict the presence of heart disease using machine learning models such as Logistic Regression, Decision Tree, and KNN,SVM. The system takes medical input data and provides prediction output to support early diagnosis.

## **Objectives:**

- Predict heart disease based on input data.
- Compare multiple algorithms.

# Dataset Description

Feature	Data Type	Missing values
age	float64	3
sex	float64	1
cp	float64	1
trestbps	float64	2
chol	object	5
fbs	float64	3
restecg	float64	1
thalach	float64	3
exang	int64	0
oldpeak	float64	1
slope	int64	0
ca	float64	2
thal	float64	2
target	int64	0

This project uses the **UCI Heart Disease Dataset**, which contains **310 records** and **14 attributes (columns)** related to patient health data.

Below is a summary of the dataset:

- . **Number of records (rows):** 310

- . **Number of features (columns):** 14

- . **Target variable:** target (1 = presence of heart disease, 0 = absence)

# **Data Preprocessing**

## **1) Conversion of Object Columns to Numeric:**

Some columns (like chol) were stored as object types due to formatting issues . These were converted to numeric for proper analysis.

## **2) Handling null , missing data and duplicated:**

### **-Duplicate rows:**

Duplicates rows were removed to avoid redundancy.

### **-Missing values Handling:**

- For numerical columns with potential outliers (trestbps, chol, thalach, oldpeak), missing values were filled using the median

to minimize the impact of skewed data. If still missing, the mean was used as a fallback.

- For categorical columns (sex, cp, fbs, exang, exang, slope, ca, thal), missing values were filled using the **mode** (most frequent value).

#### **-Outlier Clipping:**

The ca column was clipped to be within the valid range [0, 3] to handle abnormal values.

### **3) Outliers detection and Treatment:**

To identify and treat outliers in key numerical columns (trestbps, chol, thalach, oldpeak), the Interquartile Range (IQR) method was used.

#### **IQR Method:**

- For each column, the first (Q1) and third (Q3) quartiles were calculated.

- The normal range was defined as:  
Normal Range =  $Q1 - 1.5 * IQR$  TO  $Q3 + 1.5 * IQR$
- Any values outside this range were considered outliers.

### **Treatment:**

- Outliers were replaced with the respective boundary values:
  - Values lower than  $Q1 - 1.5 * IQR$  were set to the lower bound.
  - Values higher than  $Q3 + 1.5 * IQR$  were set to the upper bound.

## **4)Correlation Analysis with Target Variable:**

To understand the strength of the relationship between each feature and the target variable, we calculated the Pearson correlation coefficient.

- Features were classified based on their correlation with the target as follows:
  - High Correlation: Absolute correlation values between 0.7 and 0.95.

-Low Correlation: Absolute correlation values between 0.4 and 0.7.

Poor Correlation: Correlation values below 0.4 in absolute terms.

- This categorization helps in feature selection and

understanding which features significantly to predicting the target.

After completing all preprocessing steps, the dataset was clean, consistent, and ready for model training. The processed data was saved to a new file for reproducibility.

# **Model Training and Evaluation**

After preprocessing the dataset and selecting the most relevant features based on their correlation with the target variable, we proceeded to train and evaluate different machine learning models for heart disease prediction.

## **Input:**

- Features used: The top features selected based on correlation analysis (high correlation + low correlation)
- Target: target column (1 = presence of heart disease, 0 = absence)
- Data split: 80% training and 20% testing using `train_test_split` with `random_state=42` for reproducibility.



- Feature scaling: All features were scaled using StandardScaler to ensure consistent input for the models.

## **Models Used:**

1) logistic regression

2) SVM

3) Decision Tree

4) KNN

Each model was trained using the same training data, and evaluated on the test set.

### **1) Logistic Regression :**

We trained a Logistic Regression classifier using hyperparameter tuning via Randomized Search with 5-fold Stratified Cross-Validation. The search space included the regularization parameter C (sampled from a log-uniform distribution between 0.001 and 1000) and the maximum number of iterations (max\_iter).

The Best hyperparameters found were:

$C=0.016$ ,  $\text{max\_iter}=100$  .

Using these optimal parameters, the model was trained on the training set and evaluated on the test set.

### **Performance on the test set:**

- Accuracy: 0.92 (92%)
- ROC AUC Score: 0.95
- Precision, Recall, F1-score for class 0 and 1 all above 0.90
- Mean Squared Error (MSE): 0.082

The confusion matrix showed a balanced classification with only a few misclassifications.

### **Model coefficients and intercept:**

- Coefficients indicate the weight each selected feature contributes to the prediction.
- Intercept: 0.198

The Logistic Regression model shows strong predictive performance, indicating that the selected features are effective in discriminating between patients with and without heart disease

Accuracy of logistic regression classifier on test set: 0.92

Logistic Regression AUC: 0.95

	precision	recall	f1-score	support
0	0.93	0.90	0.91	29
1	0.91	0.94	0.92	32
accuracy			0.92	61
macro avg	0.92	0.92	0.92	61
weighted avg	0.92	0.92	0.92	61

Co-efficient of logistic regression: [[ 0.27229069 0.25008443 -0.28616747 -0.31597378 -0.32201112]]

Intercept of logistic regression model: [0.19828016]

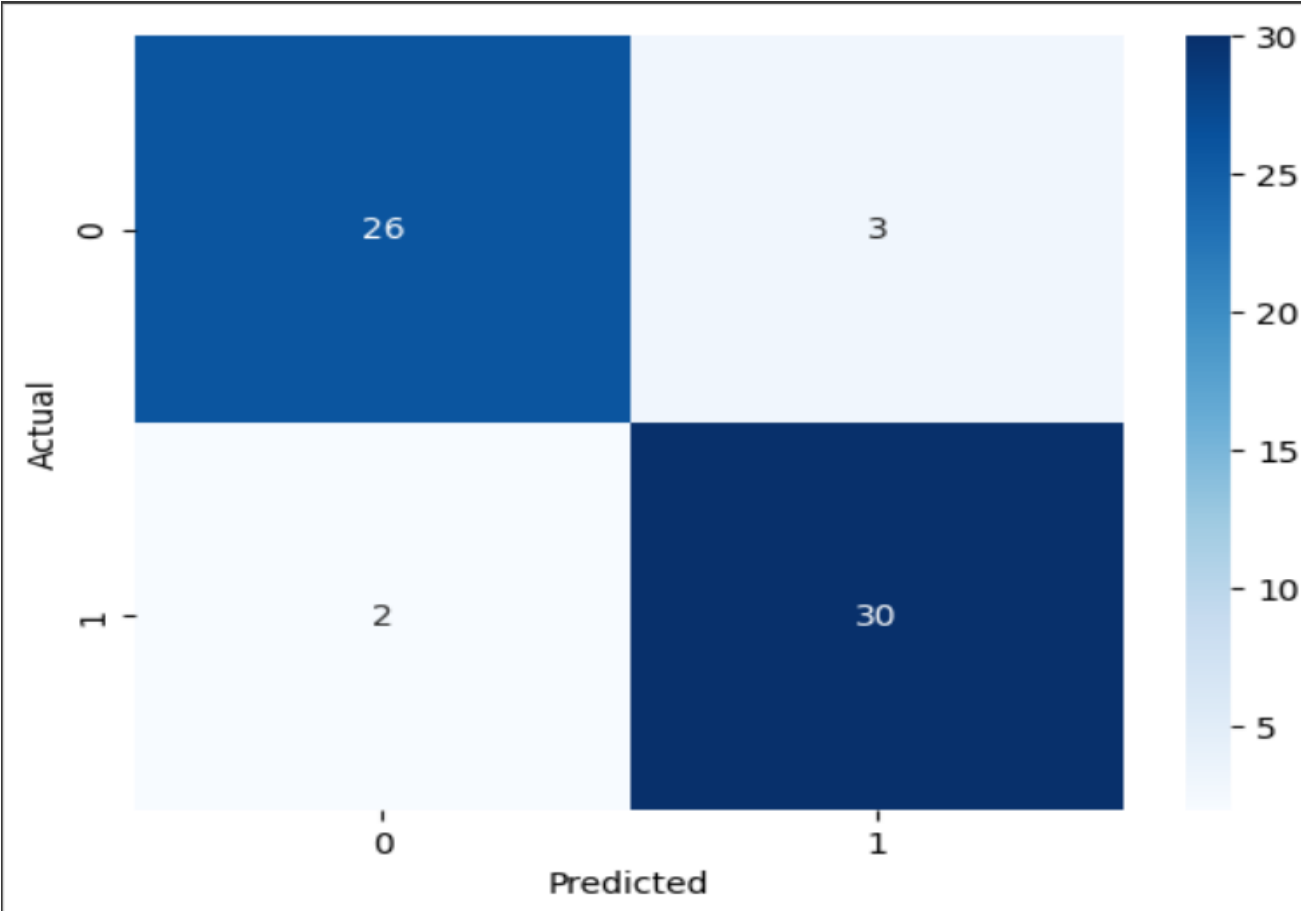
Mean Squared Error: 0.08196721311475409

True value for the first test sample: 0

Predicted value for the first test sample: 0

Classification Report:

	precision	recall	f1-score	support
0	0.93	0.90	0.91	29
1	0.91	0.94	0.92	32
accuracy			0.92	61
macro avg	0.92	0.92	0.92	61
weighted avg	0.92	0.92	0.92	61



## **2) Support Vector Machine (SVM):**

We trained an SVM classifier with a linear kernel and tuned the regularization parameter  $C$  using Grid Search combined with 5-fold Stratified Cross-Validation. The search space tested  $C$  values logarithmically spaced between 0.1 and 10.

The best hyperparameter found was:

$$C = 0.1$$

Using this optimal parameter, the model was trained on the training data and evaluated on the test set.

Performance on the test set:

- Accuracy: 0.92 (92%)
- ROC AUC Score: 0.94
- Precision, Recall, F1-score for both classes exceeded 0.90
- Mean Squared Error (MSE): 0.082

The confusion matrix confirmed balanced classification, with only a small number of misclassifications.

Model coefficients and intercept:

- Coefficients reflect the influence of each feature in the linear decision boundary.
- Intercept: 0.252

The SVM model demonstrated strong predictive ability, comparable to Logistic Regression, validating its effectiveness for heart disease classification using the selected features.

```
Accuracy of SVM classifier on test set: 0.92
SVM AUC: 0.94
```

	precision	recall	f1-score	support
0	0.93	0.90	0.91	29
1	0.91	0.94	0.92	32
accuracy			0.92	61
macro avg	0.92	0.92	0.92	61
weighted avg	0.92	0.92	0.92	61

```
Co-efficient of SVM (only for linear kernel): [[ 0.3449979  0.30165281 -0.39440212 -0.40499283 -0.48650336]]
```

```
Intercept of SVM model: [0.25213839]
```

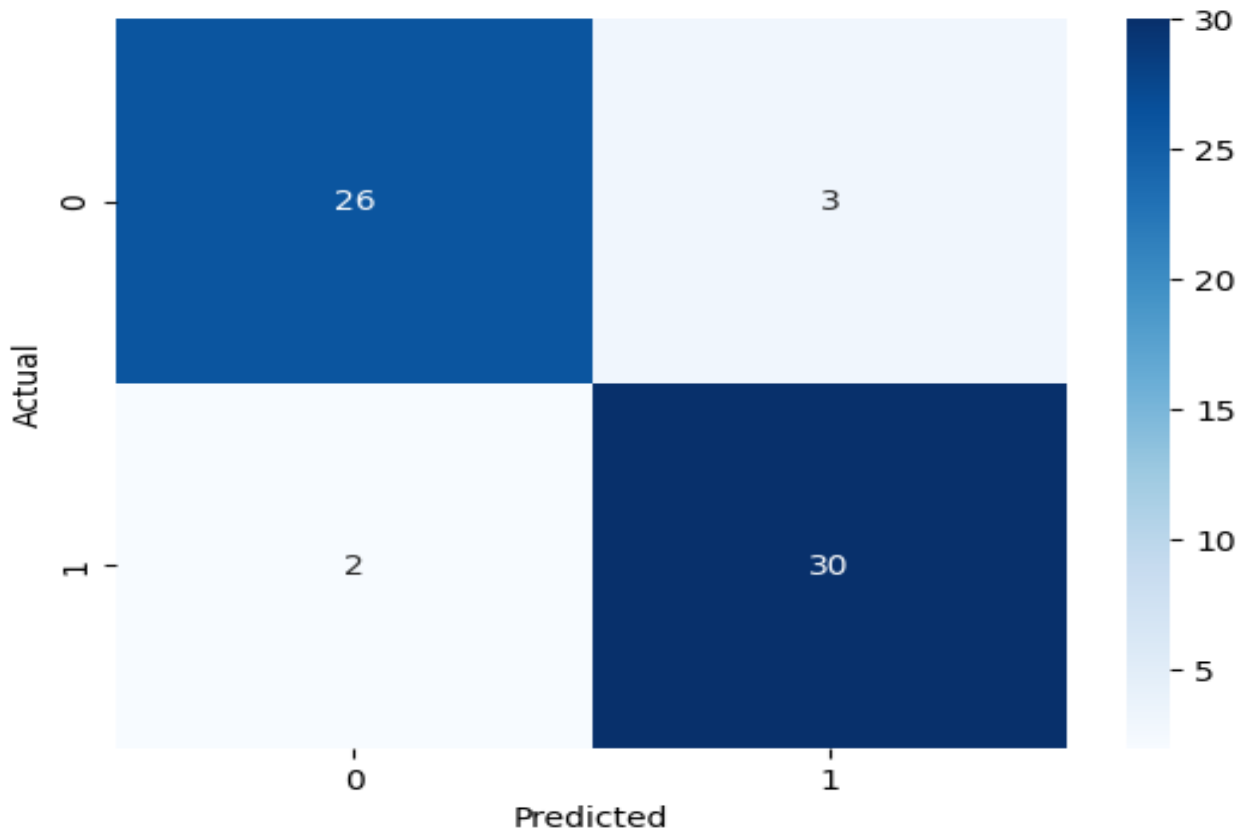
```
Mean Squared Error: 0.08196721311475409
```

```
True value for the first test sample: 0
```

```
Predicted value for the first test sample: 0
```

```
Classification Report:
```

	precision	recall	f1-score	support
0	0.93	0.90	0.91	29
1	0.91	0.94	0.92	32
accuracy			0.92	61
macro avg	0.92	0.92	0.92	61
weighted avg	0.92	0.92	0.92	61



### 3)Decision Tree:

We trained a Decision Tree classifier with hyperparameter tuning via Grid Search combined with 5-fold Stratified Cross-Validation. The search space included:

- Criterion: entropy
- Maximum tree depth: None, 3, 5, 7, 10, 15,1,2,20

- Minimum samples required to split a node: 2, 5, 10
- Minimum samples required at a leaf node: 1, 2, 4, 5, 8, 9, 10
- Maximum leaf node: None, 5, 7, 9, 10

**The best hyperparameters found were:**

criterion = 'entropy', max\_depth = None,  
min\_samples\_split = 2, min\_samples\_leaf = 8,  
max\_leaf\_nodes: 9

Using these optimal parameters, the model was trained on the training data and evaluated on the test set.

Performance on the test set:

- Accuracy: 0.8361 (84%)
- ROC AUC Score: 0.90
- Precision, Recall, and F1-score for both classes were 0.81 to 0.87, showing balanced classification
- Mean Squared Error (MSE): 0.163

The confusion matrix revealed a moderate number of misclassifications, with slightly better recall for class 0 (non-heart disease) than class 1.

Overall, the Decision Tree classifier provided good interpretability with competitive performance, although

Accuracy of Decision Tree classifier on test set: 0.84

Decision Tree AUC: 0.89

	precision	recall	f1-score	support
0	0.81	0.86	0.83	29
1	0.87	0.81	0.84	32
accuracy			0.84	61
macro avg	0.84	0.84	0.84	61
weighted avg	0.84	0.84	0.84	61

Feature Importances of the Decision Tree: [0.41402963 0.02679186 0.15826027 0.17770161 0.22321664]

Mean Squared Error: 0.16393442622950818

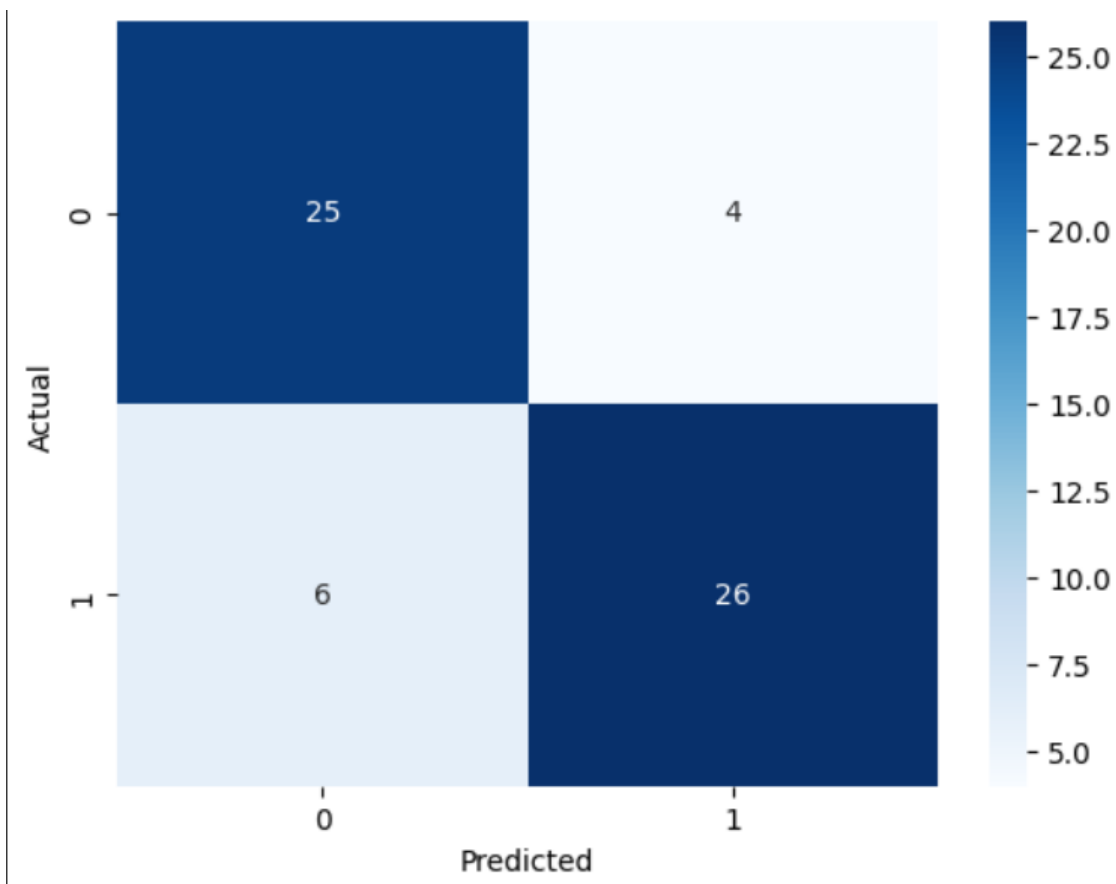
True value for the first test sample: 0

Predicted value for the first test sample: 0

Classification Report:

	precision	recall	f1-score	support
0	0.81	0.86	0.83	29
1	0.87	0.81	0.84	32
accuracy			0.84	61
macro avg	0.84	0.84	0.84	61
weighted avg	0.84	0.84	0.84	61

slightly less accurate than Logistic Regression and SVM on this dataset.





#### **4) K-Nearest Neighbors (KNN) :**

We trained a KNN classifier with hyperparameter tuning via Grid Search using 5-fold Stratified Cross-Validation. The tuning focused on the number of neighbors (`n_neighbors`) with the search space including [1, 2, 3, 5, 7, 9, 14, 20].

The best hyperparameter found was:

`n_neighbors = 14`

Using this optimal parameter, the model was trained on the training set and evaluated on the test set.

#### **Performance on the test set:**

- Accuracy: 0.90 (90%)
- ROC AUC Score: 0.93
- Precision, Recall, and F1-score for both classes were balanced around 0.90, reflecting strong classification performance
- Mean Squared Error (MSE): 0.098

The confusion matrix showed only a few misclassifications, with a slight edge in recall for class 0 (non-heart disease).

Overall, the KNN classifier demonstrated robust

```
Accuracy of KNN classifier on test set: 0.90
KNN AUC: 0.93

              precision    recall  f1-score   support

     0       0.87       0.93       0.90         29
     1       0.93       0.88       0.90         32

 accuracy          0.90         0.90         0.90         61
 macro avg         0.90         0.90         0.90         61
weighted avg         0.90         0.90         0.90         61

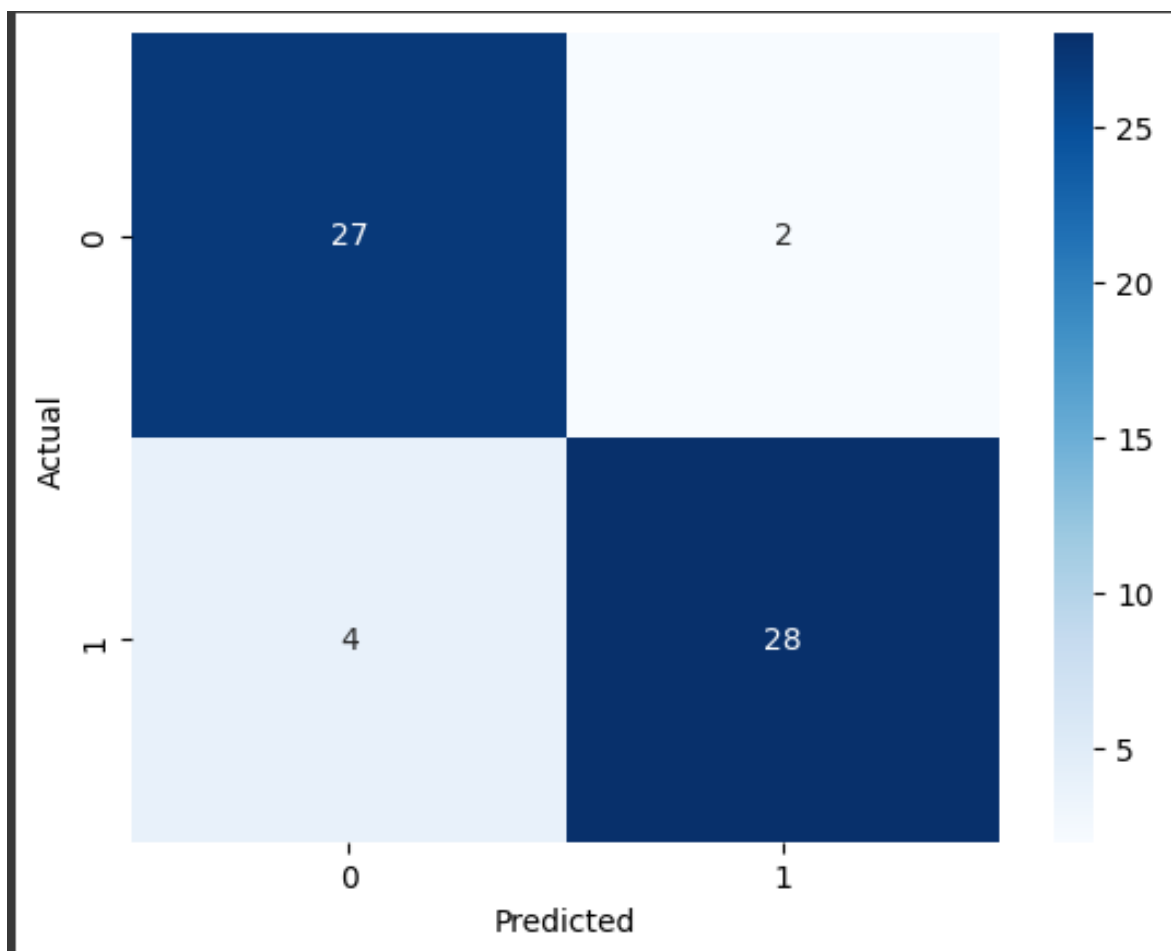
Mean Squared Error: 0.09836065573770492
True value for the first test sample: 0
Predicted value for the first test sample: 0
Classification Report:

              precision    recall  f1-score   support

     0       0.87       0.93       0.90         29
     1       0.93       0.88       0.90         32

 accuracy          0.90         0.90         0.90         61
 macro avg         0.90         0.90         0.90         61
weighted avg         0.90         0.90         0.90         61
```

predictive ability, performing closely behind Logistic Regression and SVM in this heart disease classification task.



After evaluating all four models Logistic Regression, SVM, Decision Tree, and KNN based on accuracy, ROC AUC score, classification metrics, and error rate:

- Logistic Regression and SVM showed the highest performance with accuracy ~92% and AUC > 0.95, making them the top-performing models.

- KNN followed closely with 90% accuracy and AUC ~0.93, showing solid generalization but slightly less robustness.
- Decision Tree showed decent performance (80% accuracy) but was outperformed by the other models in terms of overall metrics and generalization.

## Final Model Evaluation Summary:

Across all models, we performed accurate evaluation using both feature selection and recursive feature elimination Here's the comparison:

Model	Accuracy	Recall	F1Score	AUC	Precision
Logistic Regression	0.9180	0.9375	0.9231	0.9450	0.9091
SVM	0.9180	0.9375	0.9231	0.9429	0.9091
KNN	0.9016	0.8750	0.9032	0.9343	0.9333
Decision Tree	0.8361	0.8125	0.8387	0.9000	0.8667

# Data visualization

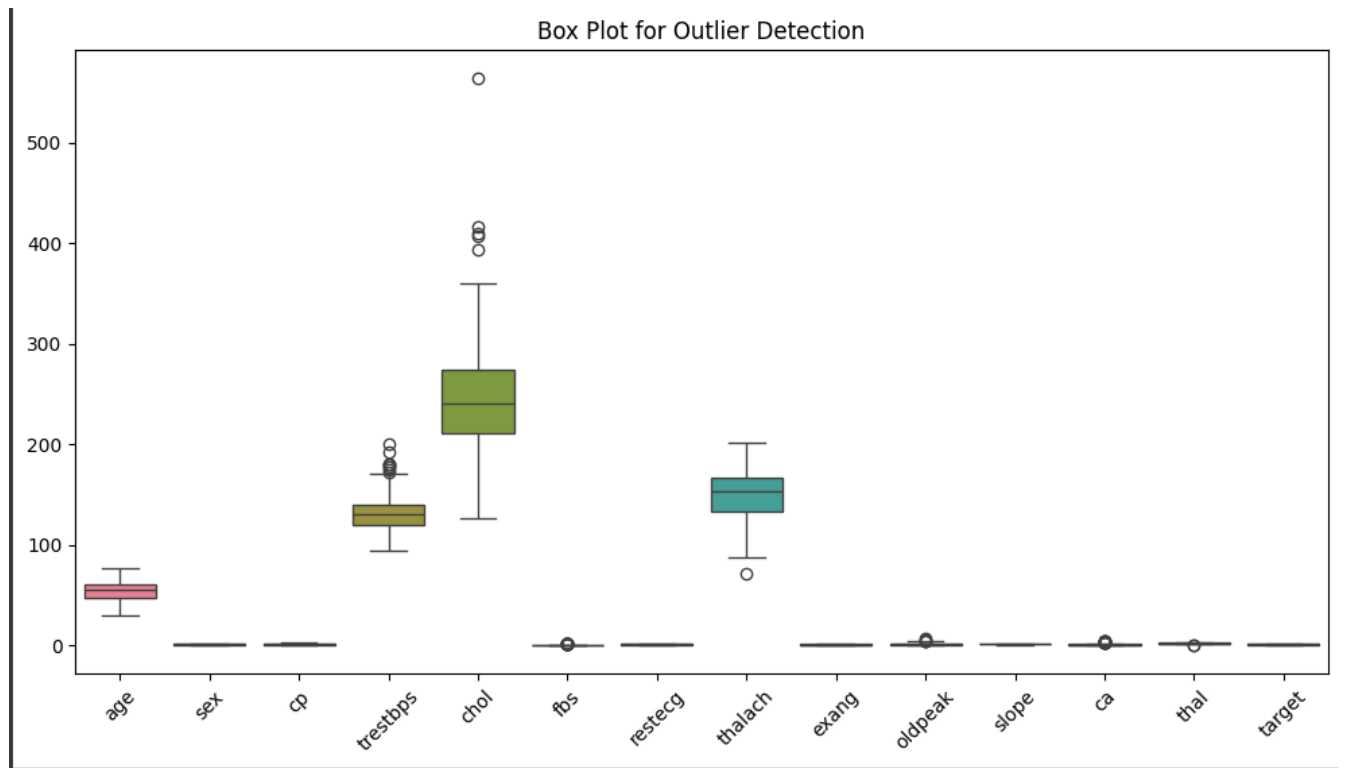
## 1)Box plot for outliers Detection:

Box plots help detect outliers in the dataset by showing the interquartile range (IQR), median, and extreme values.

Each box displays:

- The 25th percentile (Q1) and 75th percentile (Q3) as the box boundaries.
- The median as a horizontal line inside the box.
- Whiskers extending to show the range within  $1.5 \times \text{IQR}$ .
- Points outside the whiskers are considered outliers.

Features like chol, trestbps, and thalach showed multiple outliers.



## 2)Histogram and Normal Distribution curves:

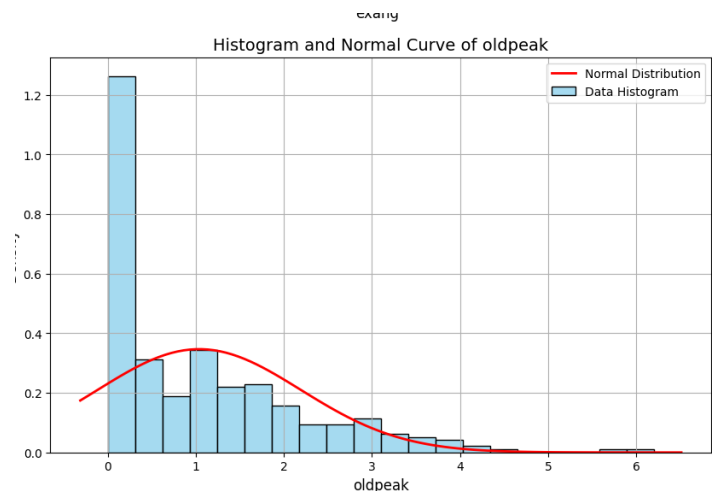
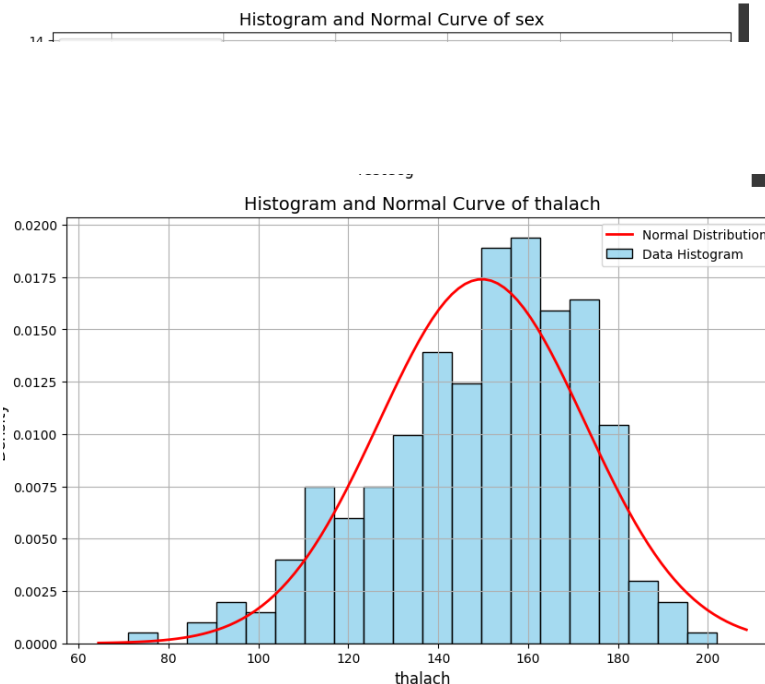
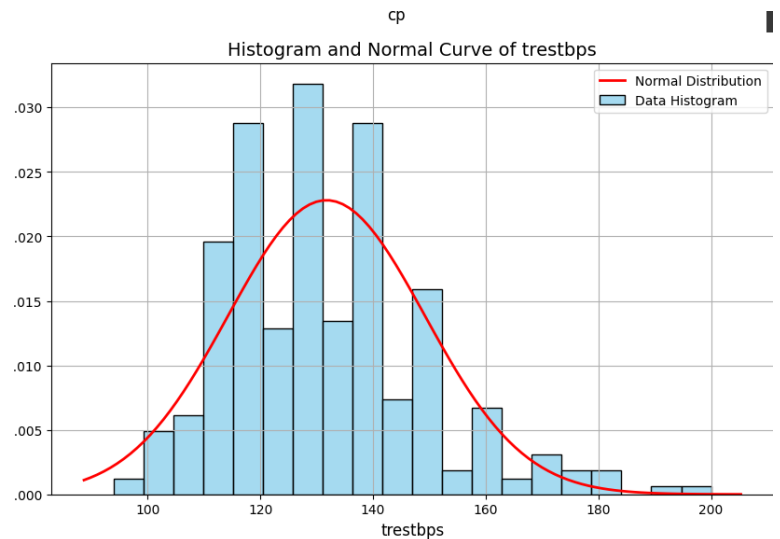
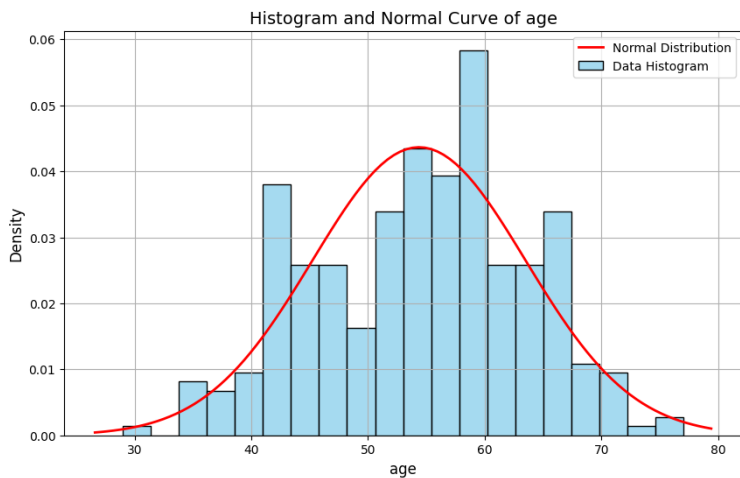
To analyze the distribution of each numerical feature, histograms were plotted alongside the normal distribution curve for each variable.

They help to visually inspect:

- The shape of the distribution (e.g., normal, skewed, bimodal).

Columns like age and chol showed normal distributions.

Other features such as thalach and oldpeak showed skewness, indicating deviation from normality.



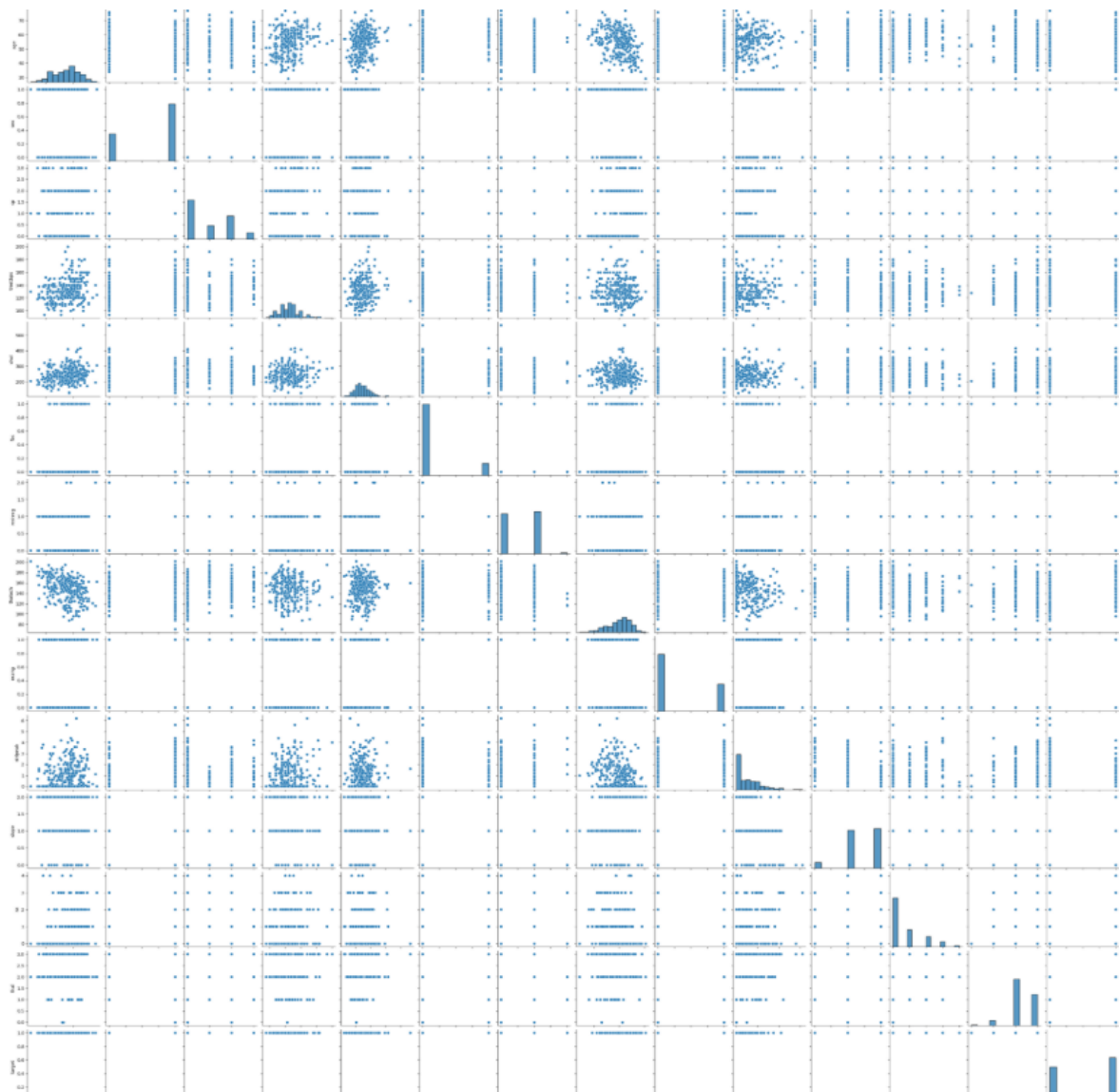


### 3)Pair plot:

A pair plot displays a grid of scatter plots for each pair of numerical variables.

The diagonal of the plot contains histograms (or KDEs) showing the distribution of individual features.

It allows us to visually inspect correlations, clusters, and potential linear/nonlinear patterns between variables.



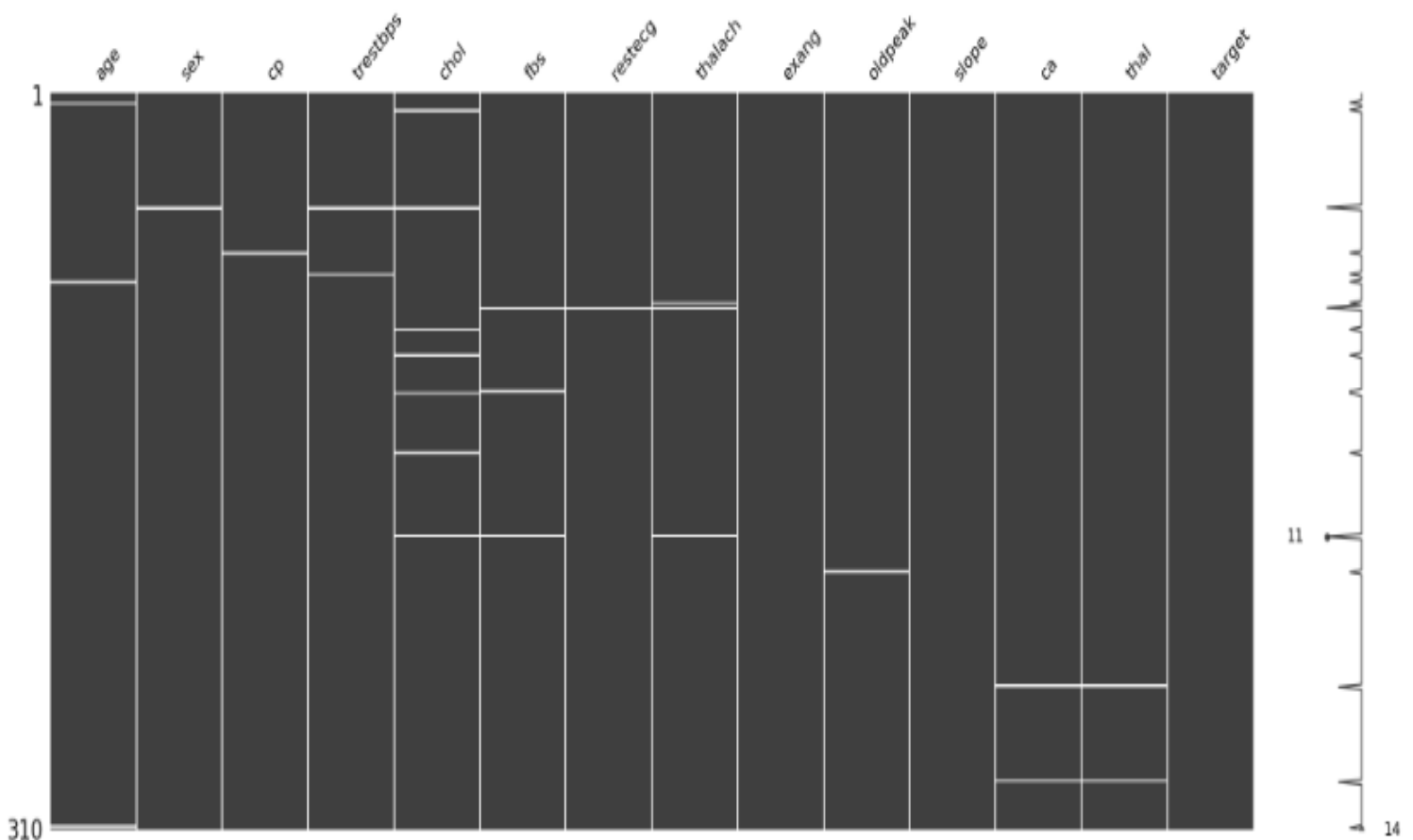
## 4)Missing Data Visualization – Matrix Plot :

A matrix plot visually represents which values are missing and where they appear across the dataset.

Each column is a variable; each row is a record.

White lines indicate missing values, and gray bars show existing (non-null) values.

A bar chart on the right summarizes the total number of non-null entries per column.



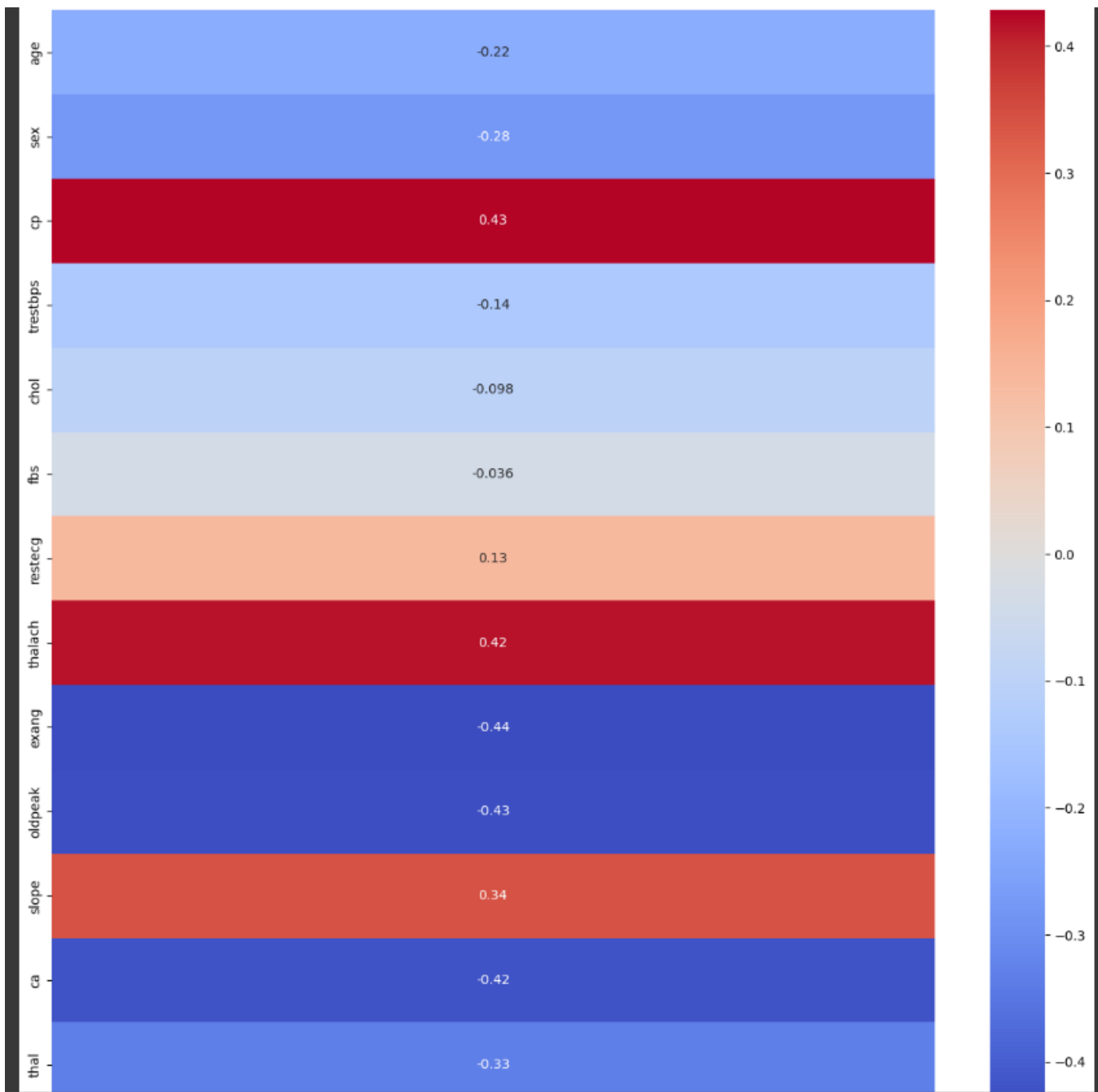
## 4)Correlation Visualization :

- **Heatmap:**

To analyze the relationships between the numerical features, a correlation matrix was visualized using a heatmap.

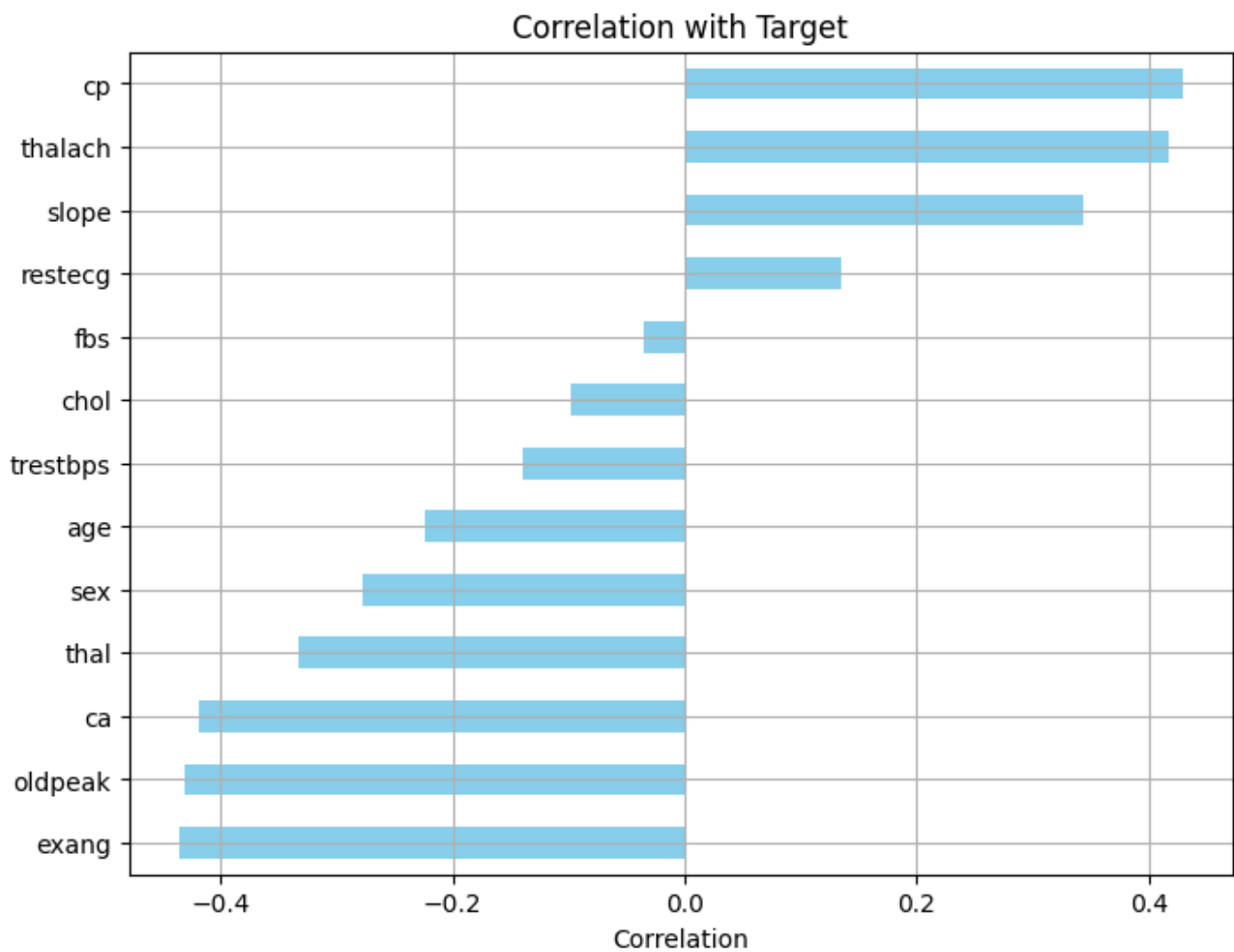
- Helped in identifying highly correlated features, which may lead to multicollinearity in models like logistic regression.
- Allowed detection of redundant variables that could be dropped or combined.





- **Horizontal Bar plot of Feature correlation with Target:**

A horizontal bar chart was generated showing the Pearson correlation coefficients of all features with the target.



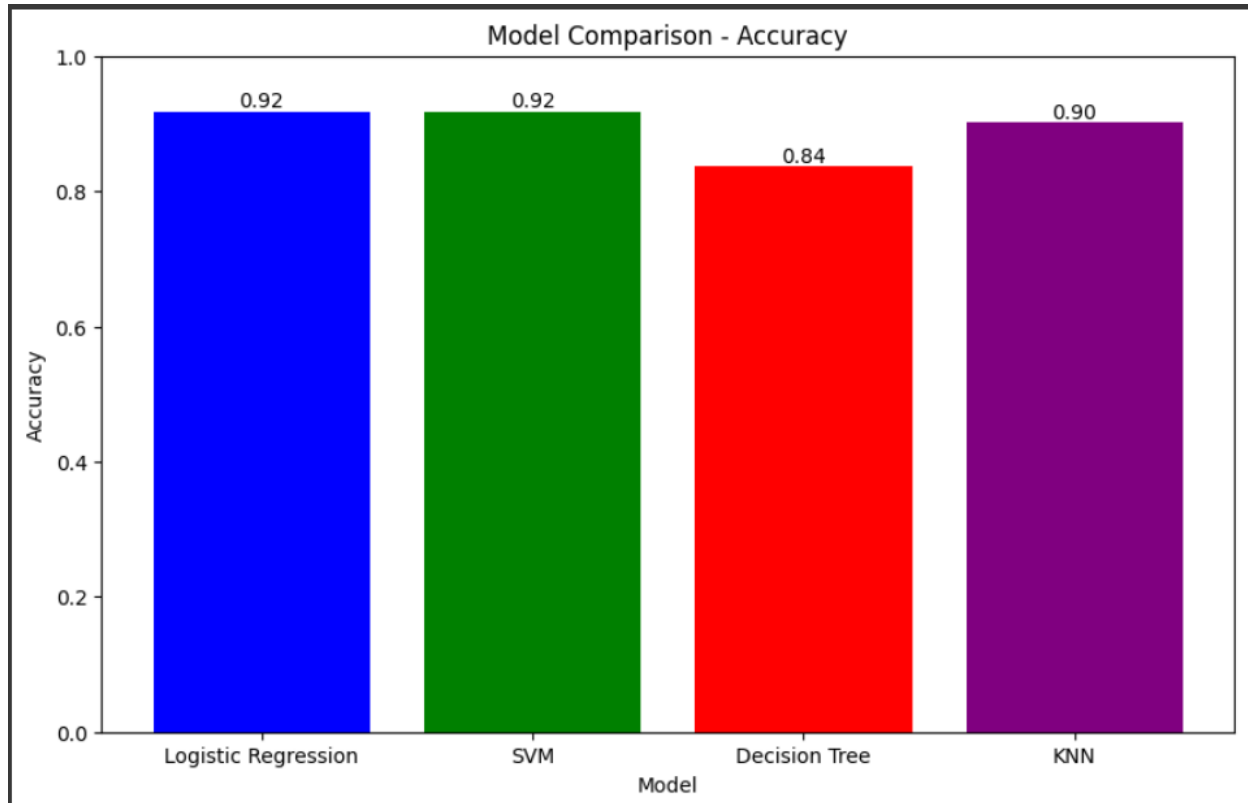
#### 4) Models Visualization:

- **Bar chart :**

X-axis represents the model names (e.g., Logistic Regression, SVM, Decision Tree, KNN).

Y-axis represents the accuracy values (ranging from 0 to 1).

The model(s) with the highest bar(s) represent those with better predictive performance.

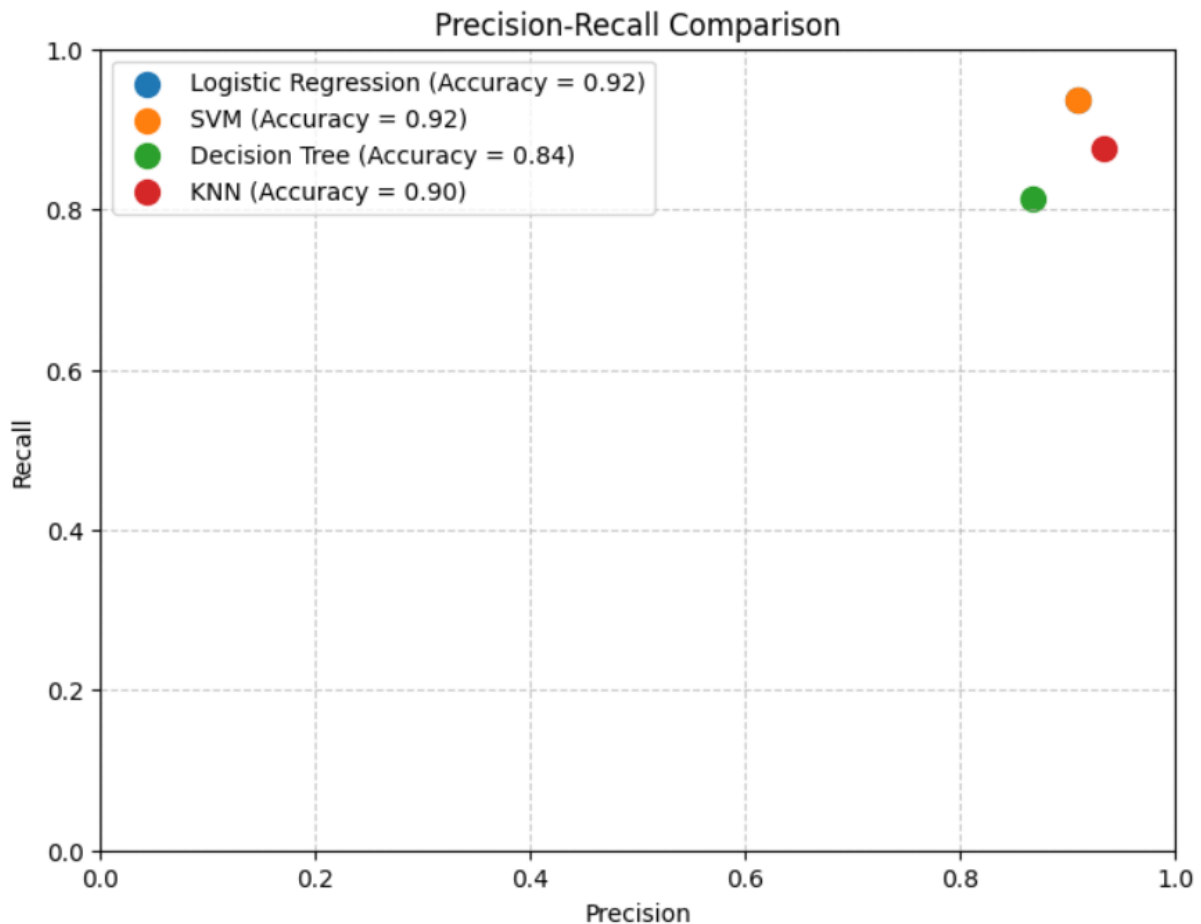


- **Scatter plot :**

A scatter plot was generated where:

- Each point represents a model.
- The x-axis shows the precision score.
- The y-axis shows the recall score.

Models in the upper-right quadrant (high precision and high recall) are ideal.

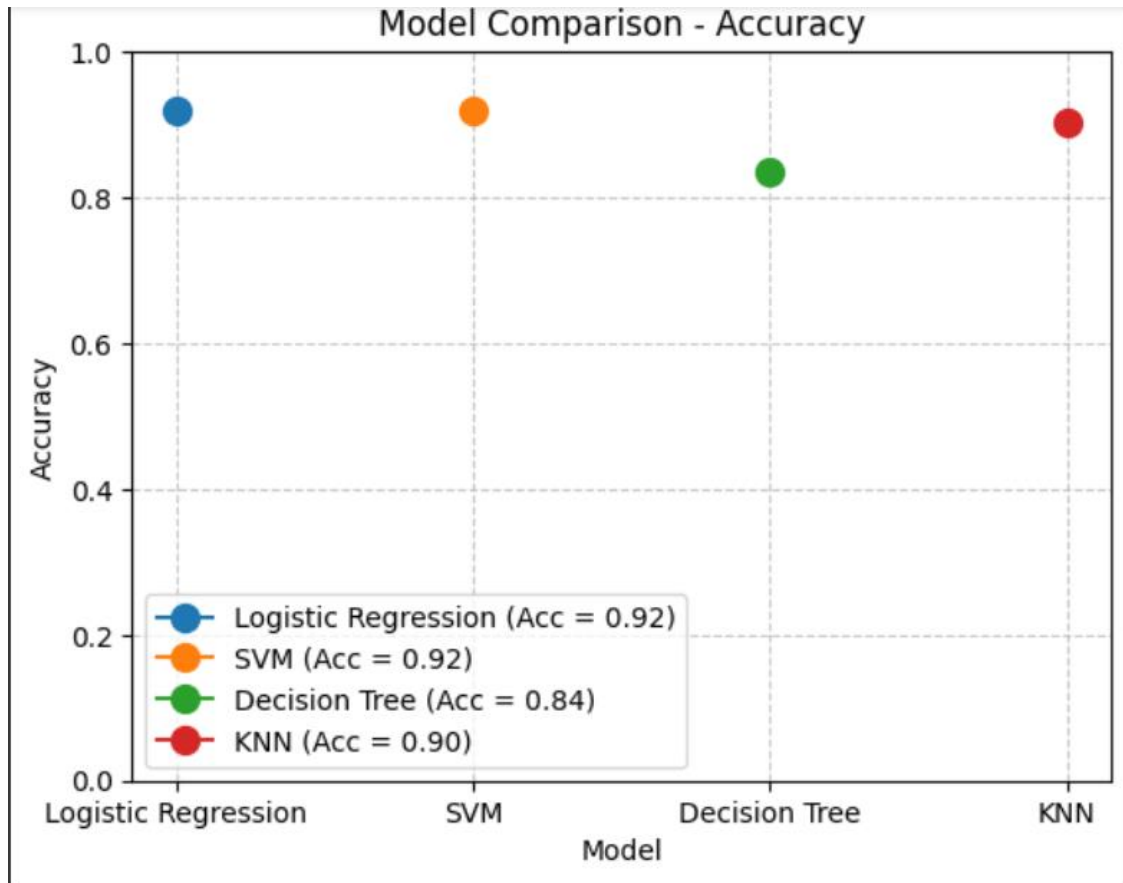


- **Line plot:**

The x-axis represents the different machine learning models.

The y-axis shows the accuracy score for each model.

The model with the highest point on the y-axis demonstrates the best performance in terms of accuracy.



- **ROC Curve:**

Each line on the plot represents the ROC curve for a model:

X-axis: False Positive Rate (FPR)

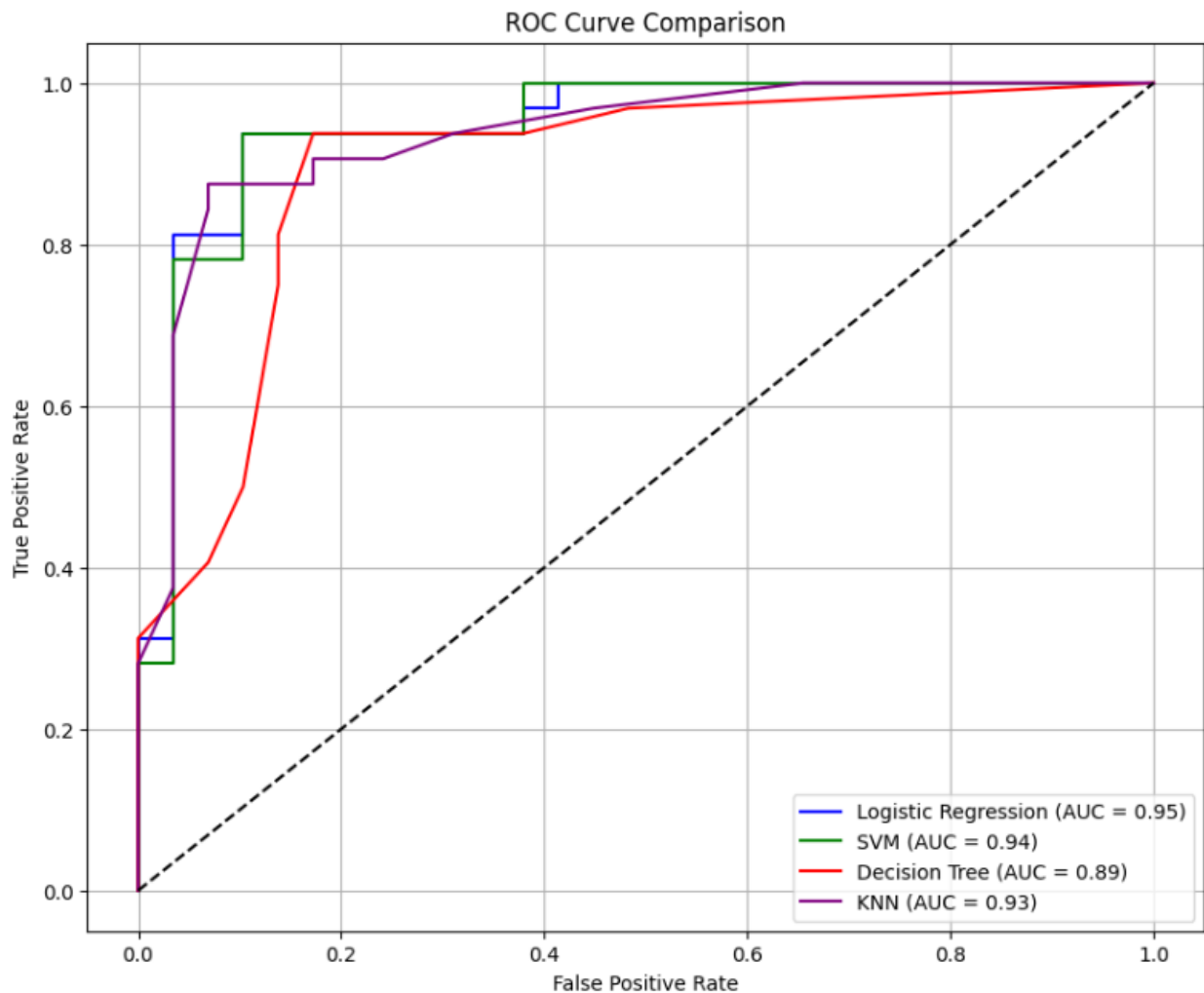
Y-axis: True Positive Rate (TPR)

The diagonal dashed line represents random guessing (baseline).





The closer the curve is to the top-left corner, the better the model is at distinguishing between classes.


The AUC value is included in the legend for each model, quantifying its performance (higher is better).



# GUI

 Heart Disease Prediction System

 **Heart Disease Prediction**

 **Enter All Clinical Features**

age (Age in years):  
sex (0=Female, 1=Male):  
cp (Chest Pain Type (0-3)):  
trestbps (Resting Blood Pressure):  
chol (Serum Cholesterol):  
fbs (Fasting Blood Sugar > 120 (0/1)):  
restecg (Resting ECG (0-2)):  
thalach (Max Heart Rate):  
exang (Exercise Angina (0/1)):  
oldpeak (ST Depression):  
slope (Slope of ST (0-2)):  
ca (No. of vessels (0-3)):  
thal (Thalassemia (1=normal, 2=fixed defect, 3=reversible defect)):

Enter age

0 - Female

0 - Typical Angina

Enter trestbps

Enter chol

0 - False

0 - Normal

Enter thalach


0 - No

Enter oldpeak


0 - Upsloping


0

1 - Normal

 **Choose Model:**

Logistic Regression

 **Predict**

 **Clear**