Heart disease

1. Introduction:

Overview of Heart Disease as a Global Health Issue:

Heart disease, also known as cardiovascular disease, encompasses a range of conditions that affect the heart and blood vessels. It is a leading cause of morbidity and mortality worldwide. The two primary types of heart disease are ischemic heart disease (narrowing of the coronary arteries) and heart failure (inability of the heart to pump blood effectively). Other conditions include arrhythmias, valvular heart diseases, and congenital heart defects.

Prevalence:

According to the World Health Organization (WHO), cardiovascular diseases are the leading cause of death globally, accounting for nearly 18 million deaths annually.

The prevalence of heart disease is influenced by factors such as age, genetics, lifestyle choices (diet, physical activity, smoking), and underlying health conditions (hypertension, diabetes).

Significance of Early Detection and Classification:

i. **Early Intervention and Treatment:**
   Early detection allows for timely intervention and treatment, which can significantly improve patient outcomes.
   For example, in cases of ischemic heart disease, early identification of risk factors (such as high blood pressure or elevated cholesterol levels) can lead to lifestyle modifications or medications to prevent progression.

ii. **Preventing Complications:**
   Detecting heart disease in its early stages can prevent complications such as heart attacks, strokes, and heart failure.
   Treatment strategies, such as medication management or surgical interventions, can be more effective when implemented early.

iii. **Risk Stratification and Personalized Treatment:**
   Classification of heart diseases aids in risk stratification, helping healthcare professionals tailor treatments based on the specific condition and its severity.
   This personalized approach improves the precision of interventions, optimizing the balance between benefits and potential side effects.

iv. **Reducing Healthcare Costs**:
   Early detection and classification can contribute to cost savings by preventing expensive emergency treatments and hospitalizations associated with advanced stages of heart disease.

v. **Patient Education and Lifestyle Modifications**:

Early diagnosis provides an opportunity for healthcare providers to educate patients about lifestyle modifications, including diet, exercise, and smoking cessation, which can mitigate disease progression.

vi. Monitoring and Follow-up:

Continuous monitoring and follow-up of patients with known heart conditions are crucial for managing the disease over time. This allows for adjustments in treatment plans as needed.

Heart disease remains a significant global health challenge. Early detection and accurate classification are paramount in improving patient outcomes by enabling timely and targeted interventions, reducing complications, and promoting a more personalized approach to treatment. These efforts not only enhance individual health but also contribute to the overall well-being of communities and healthcare systems.

2. Abstract:

Heart disease classification as a crucial research domain with substantial implications for public health.

The overarching goal in this field is to create methods that are both accurate and efficient in categorizing heart diseases.

The intricate nature of heart diseases, encompassing a diverse range of conditions and complexities, underscores the need for sophisticated classification methods. Accurate categorization allows healthcare professionals to discern between various cardiac conditions, such as coronary artery disease, heart failure, arrhythmias, and valvular disorders. This level of precision is crucial as each condition requires distinct treatment approaches.

The importance of accurate classification lies in its potential to facilitate early detection, prompt treatment, and ultimately enhance patient outcomes.

Heart disease classification is a pivotal research domain that holds the potential to significantly impact public health. The development of accurate and efficient classification methods is essential for early detection, prompt treatment, and improved patient outcomes. By advancing our ability to categorize heart diseases with precision, we can revolutionize the field of cardiology, leading to more effective healthcare practices and ultimately saving lives.

3. Objective:

The development of a predictive model for detecting the presence of heart disease in patients is a crucial undertaking with significant implications for proactive healthcare. This initiative involves leveraging a subset of 14 meticulously chosen attributes as input variables for the model. The binary nature of the target variable adds a clear and straightforward dimension to the problem, where a value of 0 signifies the absence of heart disease, while a value of 1 indicates the presence of the condition.

The careful selection of attributes is pivotal in ensuring the model's accuracy and relevance to the domain of heart disease. These attributes likely encompass a range of patient-specific factors, clinical measurements, and potentially relevant lifestyle indicators that collectively contribute to the predictive power of the model. This thoughtful curation helps in capturing the nuances of the disease and ensures that the model is trained on the most pertinent features for accurate predictions.

The ultimate goal of this predictive model is to be both robust and accurate. Robustness ensures that the model can generalize well to new, unseen data, while accuracy speaks to the model's ability to make correct predictions. Achieving this balance is essential for the model to be reliable in real-world scenarios and contribute meaningfully to healthcare practices.

The significance of the predictive model lies in its potential to facilitate early diagnosis and intervention for individuals at risk of heart disease. By accurately identifying the presence of heart disease based on the selected attributes, the model can serve as a valuable tool for healthcare professionals in assessing the risk profile of patients. Early detection is instrumental in initiating timely interventions, enabling healthcare providers to implement appropriate treatment strategies, lifestyle modifications, or further diagnostic tests to mitigate the progression of the disease.

The model's application extends to preventive healthcare, allowing for targeted interventions and personalized care plans for individuals identified as at risk. This proactive approach aligns with the principles of precision medicine, tailoring medical interventions to individual characteristics and needs.

In conclusion, the development of a predictive model for detecting heart disease based on a subset of 14 carefully chosen attributes represents a significant step toward proactive healthcare. The binary nature of the target variable, coupled with the meticulous attribute selection, underscores the intention to create a model that is not only accurate but also applicable in real-world healthcare settings. The potential impact of such a model includes early diagnosis, timely intervention, and personalized care for individuals at risk of heart disease, contributing to improved patient outcomes and overall public health.

4. Real World Impact:

Here are several potential real-world impacts:

- **Disease Prediction Models:** The dataset provides a valuable resource for developing machine learning models to predict the presence or absence of heart disease. Researchers and data scientists can use this data to build predictive models that analyze various attributes and help identify patterns associated with heart disease.
- **Clinical Decision Support Systems:** Insights gained from the dataset could contribute to the development of clinical decision support systems. These systems could assist healthcare professionals in making more informed decisions by integrating patient data and predicting the likelihood of heart disease based on historical information.
- **Risk Assessment Tools:** The dataset may contribute to the creation of risk assessment tools for individuals. By analyzing the subset of attributes that are most relevant, these tools could provide personalized risk scores for patients, helping them and their healthcare providers make lifestyle and treatment decisions.
- **Early Detection and Prevention:** With accurate prediction models, healthcare providers can potentially identify individuals at risk of heart disease at an early stage. Early detection allows for timely intervention and preventive measures, potentially improving patient outcomes and reducing the overall burden on healthcare systems.
- **Research on Attribute Relevance:** The dataset's large number of attributes allows researchers to explore and identify the most critical factors contributing to heart disease. This could lead to a better understanding of the disease's underlying mechanisms and inform future research directions.
- **Global Comparisons:** Since the dataset includes information from different regions (Cleveland, Hungary, Switzerland, and Long Beach), researchers can analyze and compare patterns of heart disease across different populations. This global perspective could contribute to the development of more universally applicable diagnostic and preventive strategies.
- **Public Health Planning:** Insights from the dataset can inform public health initiatives and policies related to heart disease. Governments and healthcare organizations can use the information to tailor preventive measures and allocate resources more effectively based on the prevalence and risk factors identified in different populations.
- **Educational Purposes:** The dataset can be used for educational purposes, helping students and healthcare professionals understand the complexities of heart disease and the application of data-driven approaches in healthcare.

5. Dataset

This data set dates from 1988 and consists of four databases: Cleveland, Hungary, Switzerland, and Long Beach V. It contains 76 attributes, including the predicted attribute, but all published experiments refer to using a subset of 14 of them. The "target" field refers to the presence of heart disease in the patient. It is integer valued 0 = no disease and 1 = disease.

Content

Attribute Information:

a. age
b. sex
c. chest pain type (4 values)
d. resting blood pressure
e. serum cholesterol in mg/dl
f. fasting blood sugar > 120 mg/dl
g. resting electrocardiographic results (values 0,1,2)
h. maximum heart rate achieved
i. exercise induced angina
j. old peak = ST depression induced by exercise relative to rest
k. the slope of the peak exercise ST segment
l. number of major vessels (0-3) colored by fluoroscopy
m. thal: 0 = normal; 1 = fixed defect; 2 = reversable defect
n. The "target" field refers to the presence of heart disease in the patient. It is integer valued 0 = no disease and 1 = disease.

The names and social security numbers of the patients were recently removed from the database, replaced with dummy values.

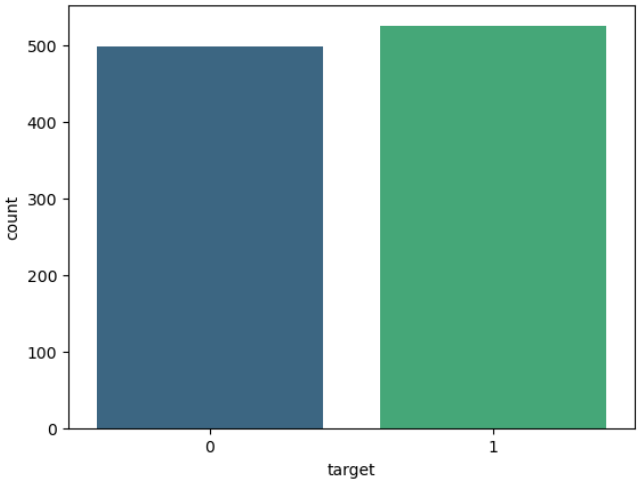


Figure 1: Target Variable

6. Metric selection and Reasoning :

Precision and Recall: Precision measures the proportion of true positive predictions out of all positive predictions, while recall measures the proportion of true positive predictions out of all actual positive cases. These metrics are particularly important if the business wants to balance the need to minimize false positives (approving bad loans) and false negatives (rejecting good loans).

F1 Score: The F1 score is the harmonic mean of precision and recall and provides a balanced measure of a model's performance. It can be useful when there is an uneven distribution between positive and negative cases in the dataset.

ROC Curve and AUC: Receiver Operating Characteristic (ROC) curve and Area Under the Curve (AUC) measure the model's ability to discriminate between good and bad credit risks across different decision thresholds. A high AUC indicates good model performance.

7. Libraries used

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import missingno as msno
```

Description of these libraries are as follows:-

- Pandas for Dataframe operations
- Numpy for Numeric operations
- Matplotlib, Seaborn and missingno are Data Visualisation libraries

8. Visualization:

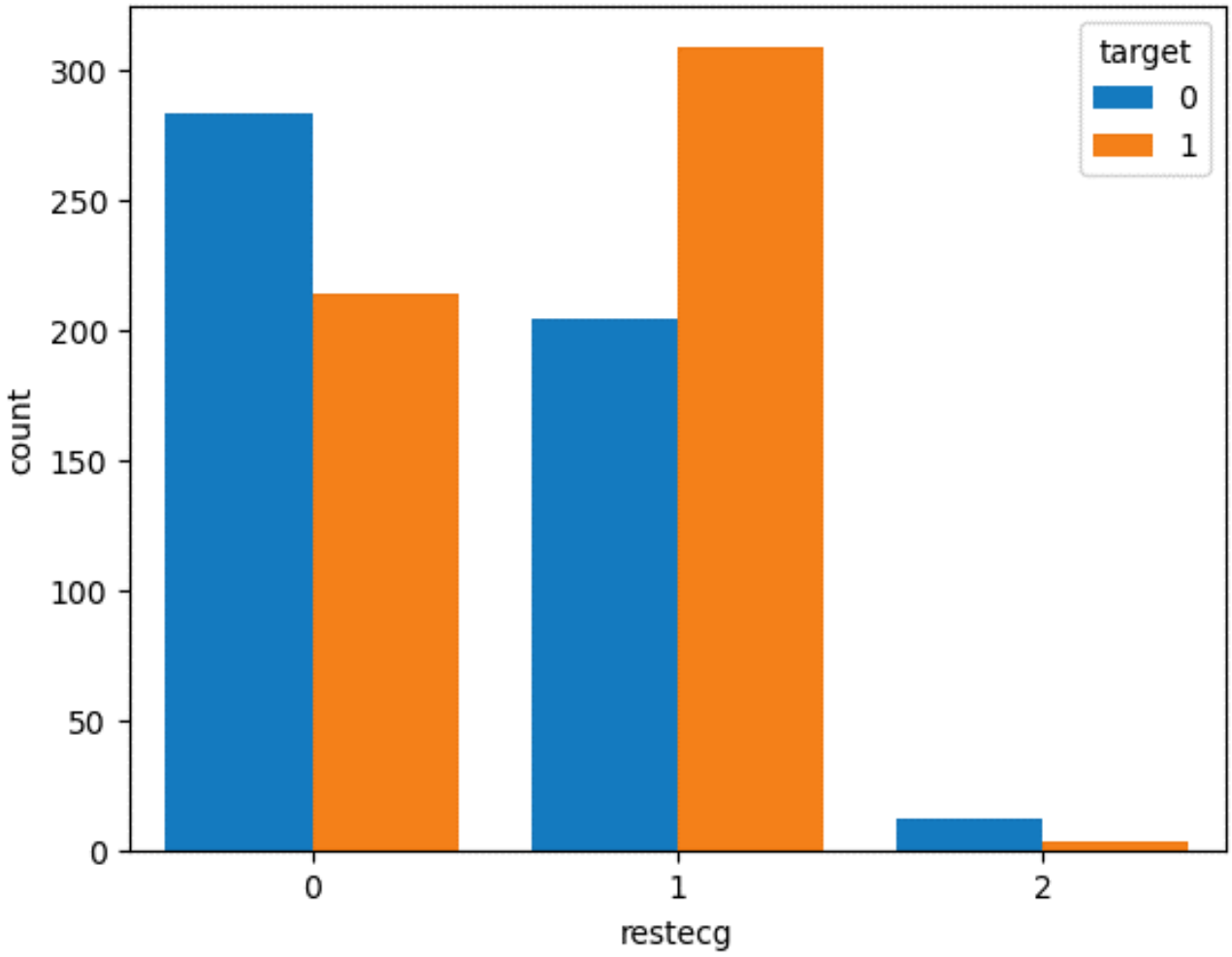


Figure 2: Resting electrocardiographic results (restecg) and (target)

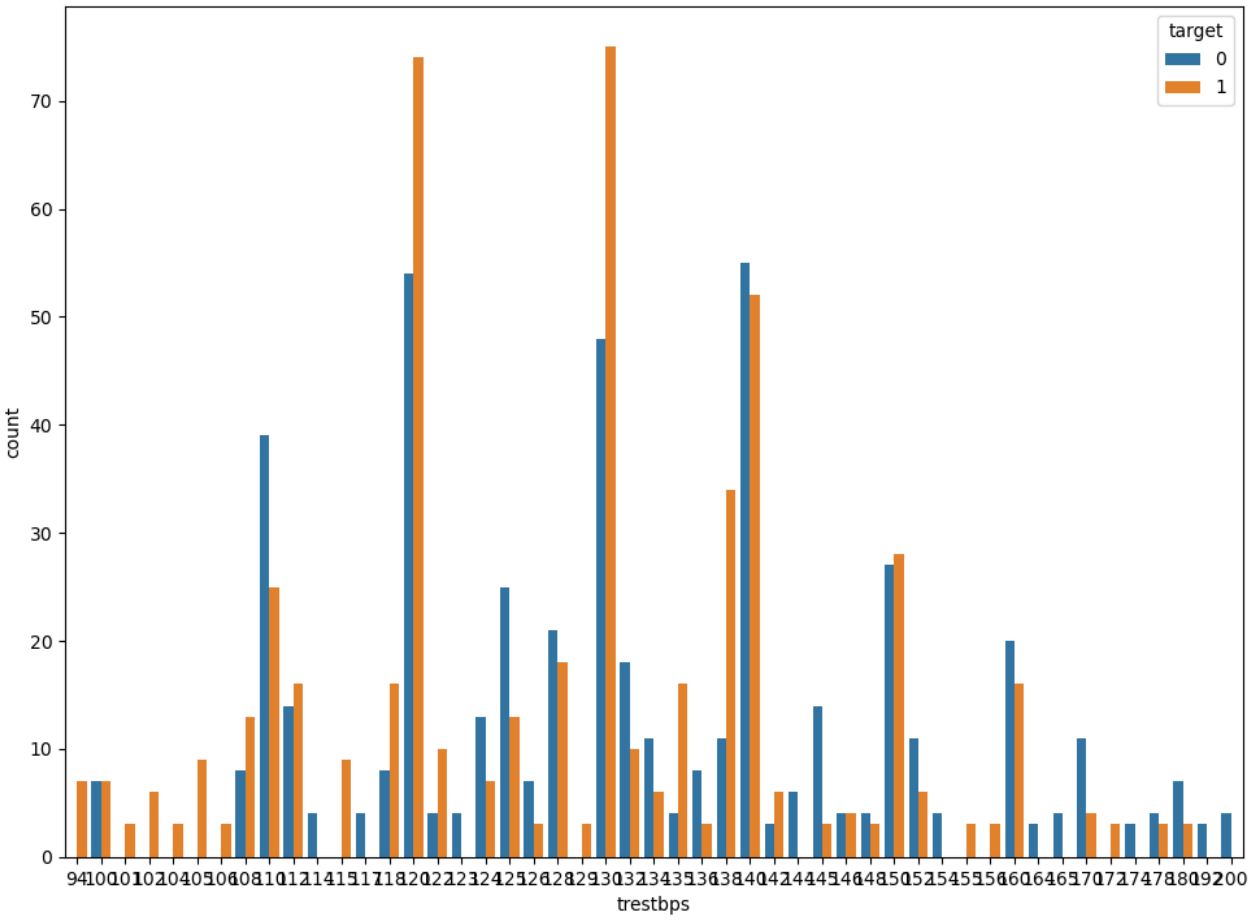


Figure 3: Count of resting blood pressure (trestbps)

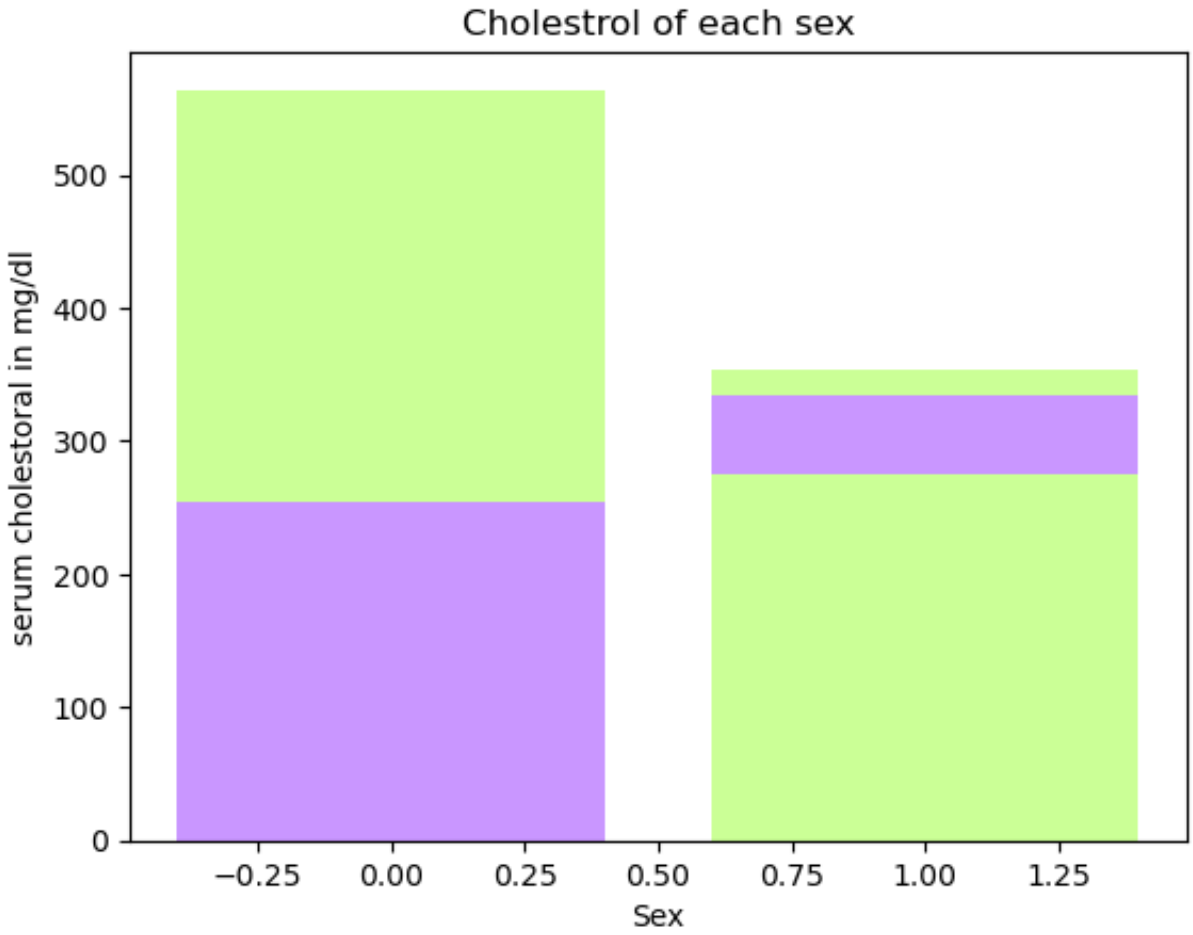


Figure 4L Cholesterol according to sex

9. Modeling:

9.1. Models used to predict

Below are models used:

a. Logistic Regression:
   Type: Supervised learning algorithm for binary and multiclass classification.
   Explanation: Logistic Regression models the probability that an instance belongs to a particular class. It uses the logistic function to squash the output between 0 and 1, making it suitable for binary classification problems.

b. K-Nearest Neighbors (KNeighborsClassifier):
   Type: Supervised learning algorithm for classification and regression.
   Explanation: K-Nearest Neighbors classifies data points based on the majority class of their k-nearest neighbors. It's a simple and intuitive algorithm, but its performance may be sensitive to the choice of distance metric and the value of k.

c. Decision Tree Classifier:
   Type: Supervised learning algorithm for classification and regression.
   Explanation: Decision Trees make decisions by recursively splitting the data based on features. Each split is determined to maximize information gain or Gini impurity, leading to a tree-like structure that can be used for classification.

d. Random Forest Classifier:
   Type: Ensemble learning algorithm for classification and regression.
   Explanation: Random Forest builds multiple decision trees and combines their predictions. It introduces randomness during the training process, both in the data used for training each tree and the features considered for each split, which often improves generalization performance.

e. Support Vector Classification (SVC):
   Type: Supervised learning algorithm for classification.
   Explanation: Support Vector Classification separates classes by finding the hyperplane that maximizes the margin between them. It works well in high-dimensional spaces and is effective when the data is not linearly separable by transforming it into a higher-dimensional space.

f. Naive Bayes:
   Type: Probabilistic supervised learning algorithm for classification.
   Explanation: Naive Bayes is based on Bayes' theorem and assumes that features are conditionally independent given the class label. Despite its "naive" assumption, it often performs well in practice and is particularly suitable for text classification.

g. XGBoost:
   Type: Ensemble learning algorithm, specifically a gradient boosting framework.

Explanation: XGBoost is an efficient and scalable implementation of gradient boosting. It builds a series of weak learners (usually decision trees) sequentially, with each tree correcting the errors of the previous ones. It's known for its speed, accuracy, and regularization techniques to prevent overfitting.

9.2. Libraries for Modeling:

```
from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeRegressor
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score,f1_score
from sklearn.svm import SVC
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report, fbeta_score
import xgboost as xgb
```

9.3. Scaling:

Standard scaling, also known as Z-score normalization, involves transforming the features of a dataset to have a mean of 0 and a standard deviation of 1.
This is achieved by subtracting the mean of each feature and dividing by its standard deviation.
Standard scaling is less sensitive to the presence of outliers compared to other scaling methods.
Outliers can have a significant impact on the mean and range of a feature, but their influence is reduced when using the Z-score.

Code snippet:

```
from sklearn.preprocessing import StandardScaler
# Standardize the features (important for KNN)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

9.4. Creating models:

Code snippet:

```
def check():
    for m in models:
        model_name = m.__class__.__name__
        model=m
        model.fit(X_train,y_train)
        y_pred = model.predict(X_test)
        print('Model Name: ',model_name)
        print("Accuracy score is: ",accuracy_score(y_test,y_pred)*100)
        print(confusion_matrix(y_test, y_pred))
        print(classification_report(y_test, y_pred))
        print()
check()
```

9.5. Precision, recall, support,f1-score and accuracy of each model

```
Model Name:  LogisticRegression
Accuracy score is:  81.81818181818183
[[119  40]
 [ 16 133]]
              precision    recall  f1-score   support

           0       0.88      0.75      0.81       159
           1       0.77      0.89      0.83       149

    accuracy                           0.82       308
   macro avg       0.83      0.82      0.82       308
weighted avg       0.83      0.82      0.82       308
```

```
Model Name:  KNeighborsClassifier
Accuracy score is:  71.42857142857143
[[111  48]
 [ 40 109]]
              precision    recall  f1-score   support

           0       0.74      0.70      0.72       159
           1       0.69      0.73      0.71       149

    accuracy                           0.71       308
   macro avg       0.71      0.71      0.71       308
weighted avg       0.72      0.71      0.71       308
```

```
Model Name:  DecisionTreeClassifier
Accuracy score is:  97.07792207792207
[[159   0]
 [  9 140]]
              precision    recall  f1-score   support

           0       0.95      1.00      0.97       159
           1       1.00      0.94      0.97       149

    accuracy                           0.97       308
   macro avg       0.97      0.97      0.97       308
weighted avg       0.97      0.97      0.97       308
```

```
Model Name:  RandomForestClassifier
Accuracy score is:  99.02597402597402
[[159   0]
 [  3 146]]
              precision    recall  f1-score   support

           0       0.98      1.00      0.99       159
           1       1.00      0.98      0.99       149

    accuracy                           0.99       308
   macro avg       0.99      0.99      0.99       308
weighted avg       0.99      0.99      0.99       308
```

```
Model Name:  SVC
Accuracy score is:  67.53246753246754
[[100  59]
 [ 41 108]]
              precision    recall  f1-score   support

           0       0.71      0.63      0.67       159
           1       0.65      0.72      0.68       149

    accuracy                           0.68       308
   macro avg       0.68      0.68      0.68       308
weighted avg       0.68      0.68      0.67       308
```

```
Model Name:  GaussianNB
Accuracy score is:  81.4935064935065
[[118  41]
 [ 16 133]]
              precision    recall  f1-score   support

           0       0.88      0.74      0.81       159
           1       0.76      0.89      0.82       149

    accuracy                           0.81       308
   macro avg       0.82      0.82      0.81       308
weighted avg       0.82      0.81      0.81       308
```

```
Model Name:  XGBClassifier
Accuracy score is:  98.05194805194806
[[159   0]
 [  6 143]]
              precision    recall  f1-score   support

           0       0.96      1.00      0.98       159
           1       1.00      0.96      0.98       149

    accuracy                           0.98       308
   macro avg       0.98      0.98      0.98       308
weighted avg       0.98      0.98      0.98       308
```

So we get the highest accuracy score that is 99 and 98.05 in Random Forest and XG Boost respectively

9.6. Hyper parameter tuning:

Define hyper parameter grids for each model:

```
param_grids = [
    {'C': [0.001, 0.01, 0.1, 1, 10, 100], 'penalty': ['l1', 'l2']},
    {'n_neighbors': [3, 5, 7], 'weights': ['uniform', 'distance']},
    {'max_depth': [None, 10, 20, 30], 'min_samples_split': [2, 5, 10]},
    {'n_estimators': [50, 100, 200], 'max_depth': [None, 10, 20, 30]},
    {'C': [0.001, 0.01, 0.1, 1, 10], 'kernel': ['linear', 'rbf']},
    {'var_smoothing': [1e-9, 1e-10, 1e-11]},  # Adjust parameters for GaussianNB
    {'learning_rate': [0.01, 0.1, 0.2], 'max_depth': [3, 5, 7]}
]
```

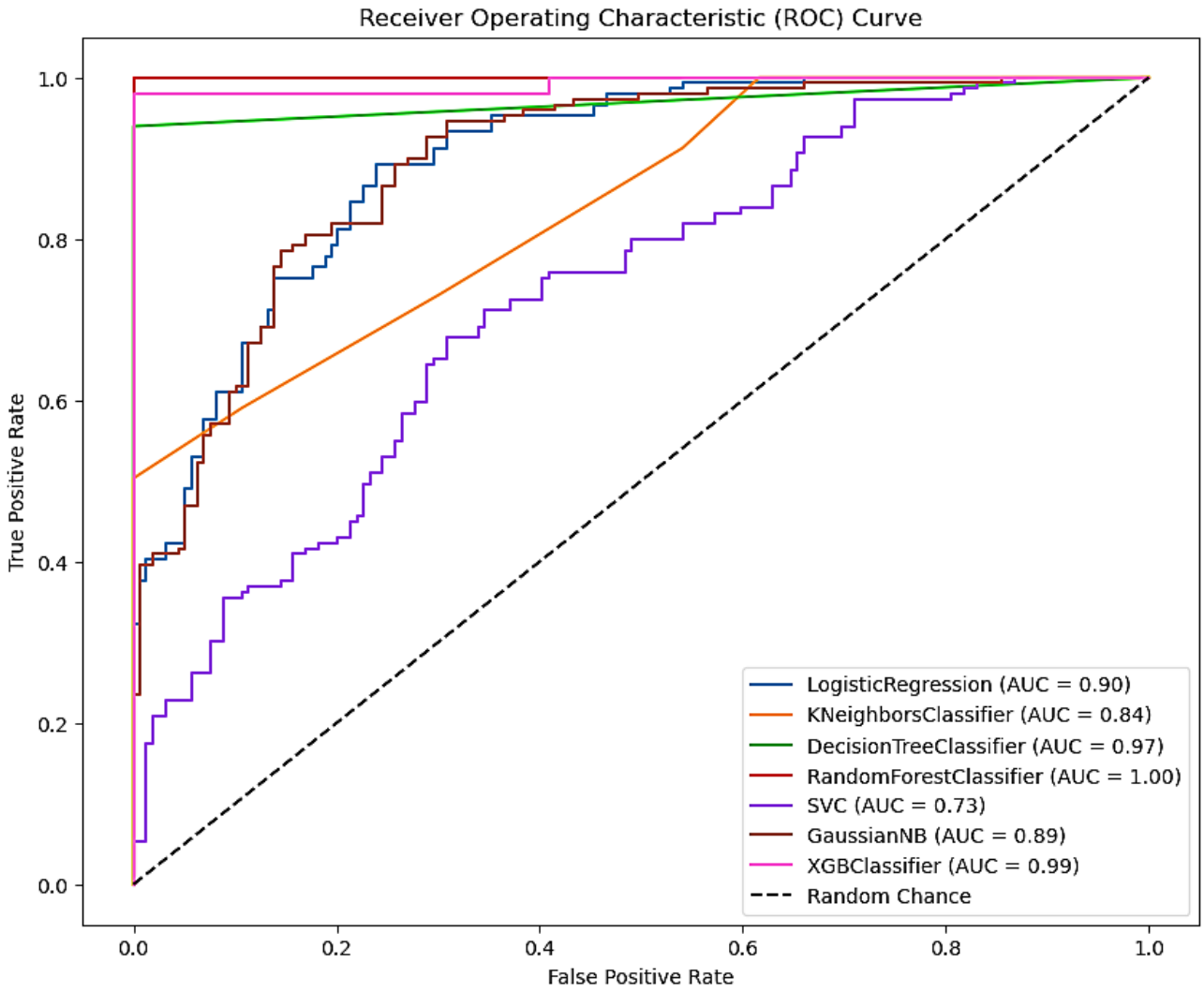
Perform hyper parameter tuning for each model:

```
for model, param_grid in zip(models, param_grids):
    model_name = model.__class__.__name__
    print(f"Hyperparameter tuning for {model_name}:-")

    grid_search = GridSearchCV(model, param_grid, cv=5, scoring='accuracy', n_jobs=-1)
    grid_search.fit(X_train, y_train)
    # Make sure to replace X_train and y_train with your data

    print(f"Best parameters: {grid_search.best_params_}")
    print(f"Best accuracy: {grid_search.best_score_:.4f}\n")
```

After hyper parameter tuning we get:

```
Best parameters: {'C': 100, 'penalty': 'l2'}
Best accuracy: 0.8466

Hyperparameter tuning for KNeighborsClassifier:-
Best parameters: {'n_neighbors': 7, 'weights': 'distance'}
Best accuracy: 0.9526

Hyperparameter tuning for DecisionTreeClassifier:-
Best parameters: {'max_depth': 10, 'min_samples_split': 2}
Best accuracy: 0.9735

Hyperparameter tuning for RandomForestClassifier:-
Best parameters: {'max_depth': 10, 'n_estimators': 200}
Best accuracy: 0.9665

Hyperparameter tuning for SVC:-
Best parameters: {'C': 1, 'kernel': 'linear'}
Best accuracy: 0.8563

Hyperparameter tuning for GaussianNB:-
Best parameters: {'var_smoothing': 1e-09}
Best accuracy: 0.8396

Hyperparameter tuning for XGBClassifier:-
Best parameters: {'learning_rate': 0.1, 'max_depth': 5}
Best accuracy: 0.9665
```

9.7.Ploting the Receiver Operating Characteristic (ROC) Curve



10. Conclusion:

After conducting model training, testing, hyper parameter tuning, and analyzing the ROC curve, the Random Forest classifier emerged as the most suitable model.

The decision to proceed with it is based on its superior performance and ability to handle the given task effectively.

In summary, the recommendation to use the Random Forest classifier is grounded in a thorough analysis of its performance across various aspects of model development and evaluation.

11. Reference:

https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset/

https://archive.ics.uci.edu/dataset/45/heart+disease