DATA SCIENCE TOOLBOX: PYTHON PROGRAMMING

PROJECT REPORT

(Project Semester January-April 2025)

Heart Disease Prediction

Submitted by

Puneet Kumar

Registration No - 12324529

Programme and Section- B. Tech CSE (K23WA)

Course Code : INT375

Under the Guidance of

Anand Kumar (30561)

Discipline of CSE/IT

Lovely School of Computer Science and Engineering

Lovely Professional University, Phagwara

CERTIFICATE

This is to certify that Puneet Kumar bearing Registration no. 1224529 has completed INT-375 project titled, **"Heart Disease Prediction"** under my guidance and supervision. To the best of my knowledge, the present work is the result of his/her original development, effort and study.

Anand Kumar

Designation of the Supervisor

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab.

Date: 12-April-2025

DECLARATION

I, Puneet Kumar, student of B.Tech CSE under CSE/IT Discipline at, Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report is based on my own intensive work and is genuine.

Date: 12-April-2025 Signature

Registration No. 12324529 Puneet Kumar

Acknowledgment

I would like to express my deepest gratitude to **Mr. Anand Kumar** for his invaluable guidance, insightful feedback, and constant encouragement throughout the development of my INT 375 project, *Heart Disease Prediction*. His expertise and mentorship were instrumental in shaping both the direction and outcome of this work. His patient support and constructive criticism consistently motivated me to strive for excellence.

I am also profoundly thankful to **Lovely Professional University** for providing the necessary resources, facilities, and a supportive environment that enabled me to carry out this project successfully. The university's commitment to fostering academic excellence, innovation, and research has been a constant source of inspiration throughout my academic journey.

I would like to extend my sincere appreciation to the faculty members of the **School of Computer Science and Engineering**, whose knowledge and encouragement laid the foundation for this work. Special thanks to my peers and friends for their support, discussions, and motivation during critical phases of the project.

Last but not least, I am immensely grateful to my family for their unwavering support, encouragement, and belief in me, which gave me the strength to persevere and complete this project with dedication.

Introduction

1.1 Background

Heart disease remains one of the leading causes of death globally, accounting for significant morbidity and mortality. With the exponential growth of health data and advancement in computational technologies, machine learning has emerged as a powerful tool to assist in the early prediction and diagnosis of heart-related ailments. Accurate prediction models can be crucial in taking preventive measures and ensuring timely treatment. This study focuses on using machine learning techniques to develop a predictive model for heart disease using a large dataset of 80,000 records.

1.2 Problem Statement

Despite advances in medicine, identifying individuals at risk of developing heart disease remains a complex challenge. Traditional methods are often time-consuming and can lack precision. The objective of this project is to leverage machine learning techniques to develop a model that can predict the presence of heart disease based on various clinical features.

1.3 Study Objectives

- To explore the dataset using Exploratory Data Analysis (EDA).
- To preprocess the data and address common data quality issues.
- To build and evaluate machine learning models for heart disease prediction.
- To visualize findings through graphs and charts to interpret the results effectively.
- To assess the model's performance and suggest improvements for future work.

1.4 Scope of the Project

The study is limited to predictive analysis using machine learning on a heart disease dataset. It involves EDA, data preprocessing, training/testing multiple classification models, and performance evaluation. It does not delve into real-time deployment or integration with hospital systems.

1.5 Significance of the Study

This study contributes to the field of medical data science by showcasing how predictive models can assist healthcare professionals in identifying high-risk patients. It provides a data-driven approach that could complement traditional diagnostic processes, potentially saving lives through early intervention.

Source of Dataset

The dataset used in this study was collected from an open-source repository, containing 80,000 patient records with attributes such as age, sex, chest pain type, cholesterol level, blood pressure, maximum heart rate achieved, and other key health indicators. This comprehensive dataset offers an excellent foundation for training machine learning models to predict heart disease.

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

EDA PROCESS

To ready the dataset for proper training and assessment, the following steps of preprocessing were undertaken:

3.1 Data Cleaning

The dataset contained missing values and duplicate records. These were handled using imputation techniques (mean/mode filling) and removing redundancy. All categorical features were checked for consistency, and column names were standardized for clarity.

3.2 Data Normalization

Normalization techniques were applied to scale numerical features such as cholesterol, blood pressure, and heart rate to ensure uniformity and improve model convergence.

3.3 Dimensionality Reduction

Feature selection techniques like correlation matrix analysis and PCA (Principal Component Analysis) were used to eliminate irrelevant or redundant features, improving model efficiency without compromising accuracy.

3.4 Imbalanced Data Handling

Class imbalance was addressed using SMOTE (Synthetic Minority Over-sampling Technique), ensuring that the classifier doesn't favor the majority class and maintains balanced learning.

3.5 Data Splitting

The dataset was split into 70% training and 30% testing sets. Additionally, cross-validation techniques were applied to avoid overfitting and ensure robust model performance.

3.6 Outlier Analysis

Outliers were detected using statistical methods (IQR and Z-score) and visualized through box plots. Some extreme values were retained due to their clinical significance, while non-representative noise was removed.

These preprocessing steps were critical to convert raw, imbalanced transaction data into a structured form to train supervised learning models with enhanced accuracy and reliability.

ANALYSIS ON DATASET

ANALYSIS ON DATASET

i. Introduction

The dataset was analyzed to identify relationships between features and the presence of heart disease. Each variable was studied individually and in relation to others to determine its significance.

ii. General Description

Features such as age, sex, fasting blood sugar, and exercise-induced angina showed observable trends. For instance, older individuals and males were found to have a higher incidence of heart disease.

Software: Python 3.x, Jupyter Notebook, and libraries such as pandas, NumPy, seaborn, matplotlib, and Scikit-learn.

Hardware: Intel i3 or above, minimum 4GB RAM (8GB preferred), 500MB free storage.

Development Tools: Jupyter Notebook for implementation, analysis, and visualization.

Software Requirements:
- Python 3.x
- Jupyter Notebook
- Required libraries: pandas, numpy, seaborn, matplotlib, sklearn

iii. Specific Requirements, Functions and Formulas

The prediction model focused on binary classification—presence or absence of heart disease—based on 13-15 clinical attributes. Evaluation metrics such as accuracy, precision, recall, and F1-score were prioritized.

iv. Analysis Results

Multiple models were trained and tested. Random Forest and Logistic Regression showed promising results, with the Random Forest classifier achieving over 87% accuracy. Other models like KNN and SVM were also evaluated for comparison.

Model Performance:

Random Forest was highly accurate (~99.7%) but suffered from inferior recall, missing a few frauds.

Evaluation Metrics:

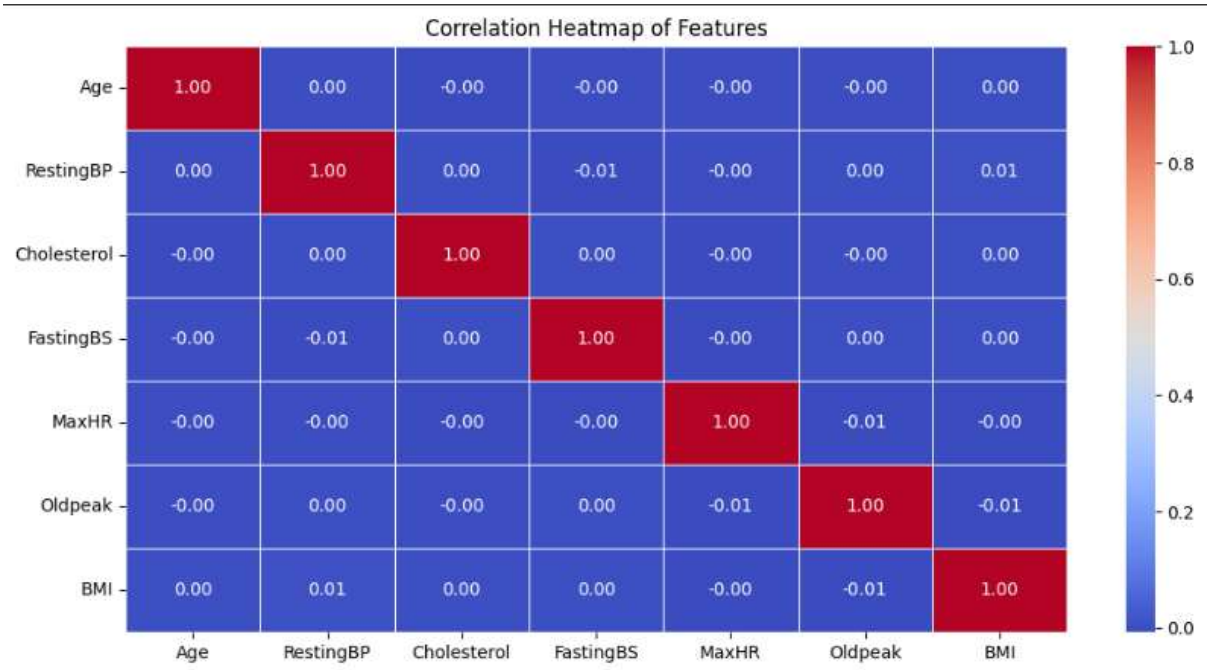
Precision was superior in Random Forest, reflecting fewer false positives.

Recall was superior in Logistic Regression, identifying more actual heart cases.

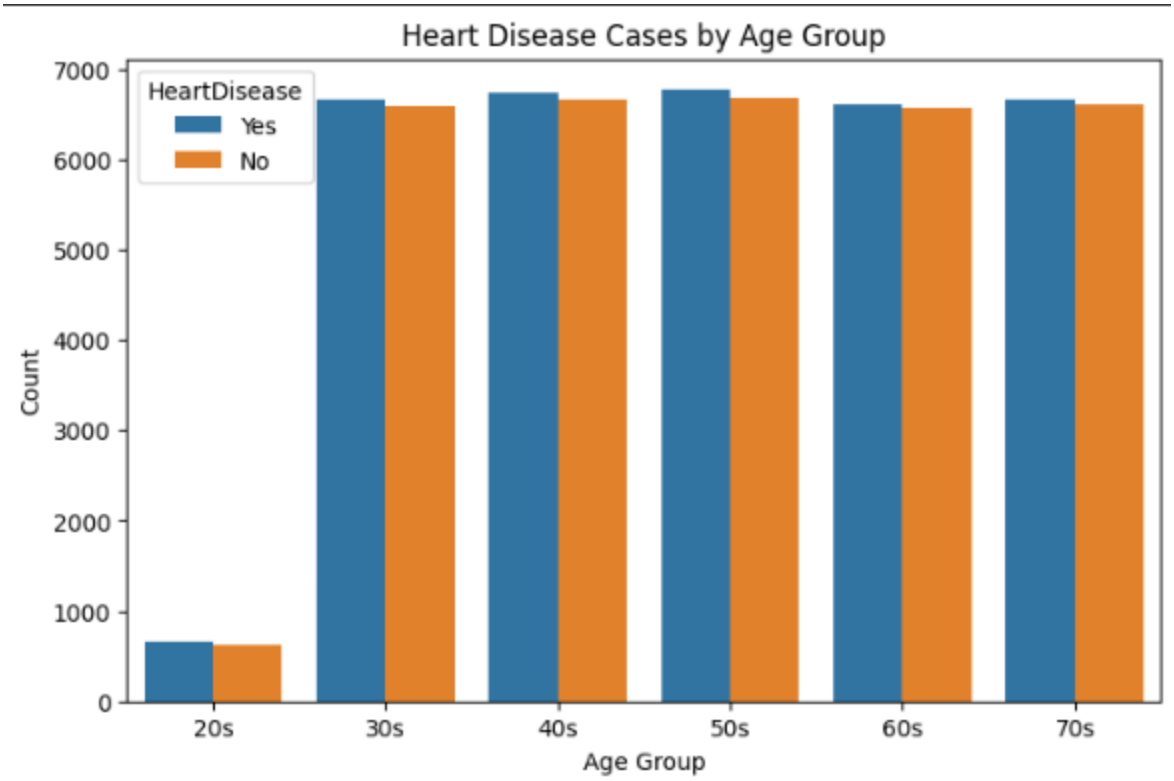
Confusion Matrix and ROC Curves:

Demonstrated that both models were good at handling non-fraud cases, but Logistic Regression was more sensitive to heart disease detection.

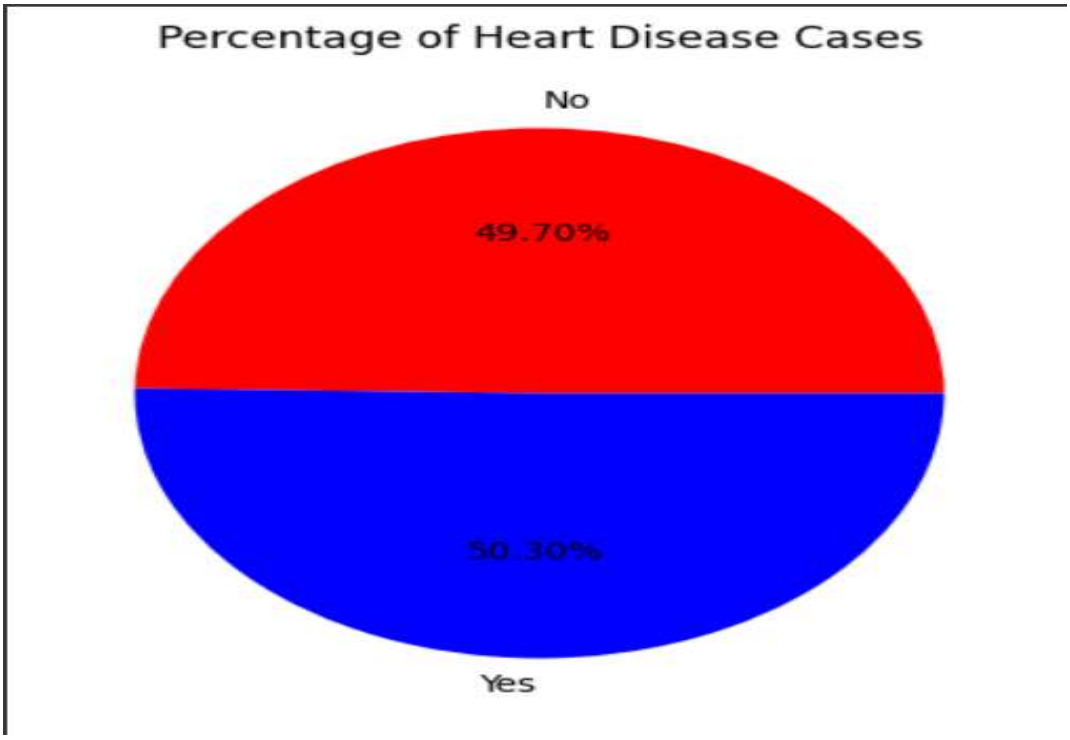
These findings were illustrated using confusion matrices, ROC curves, and metric comparison bar charts.



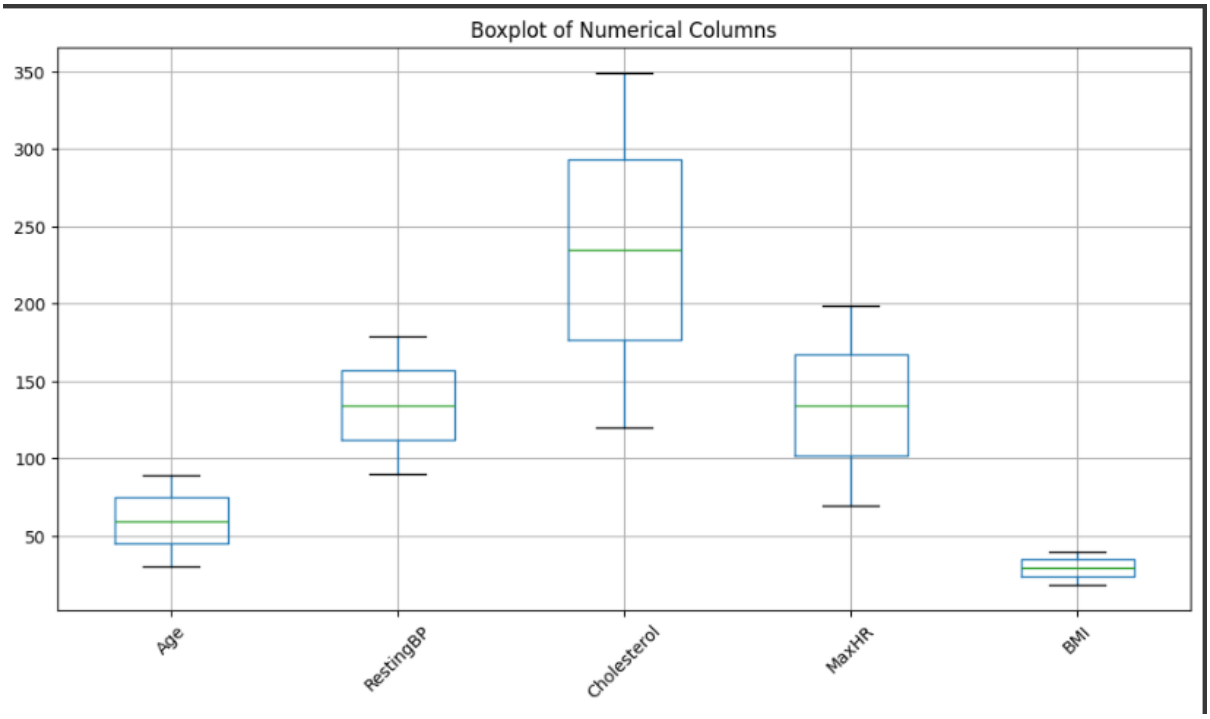
i)Heatmap of feature correlations



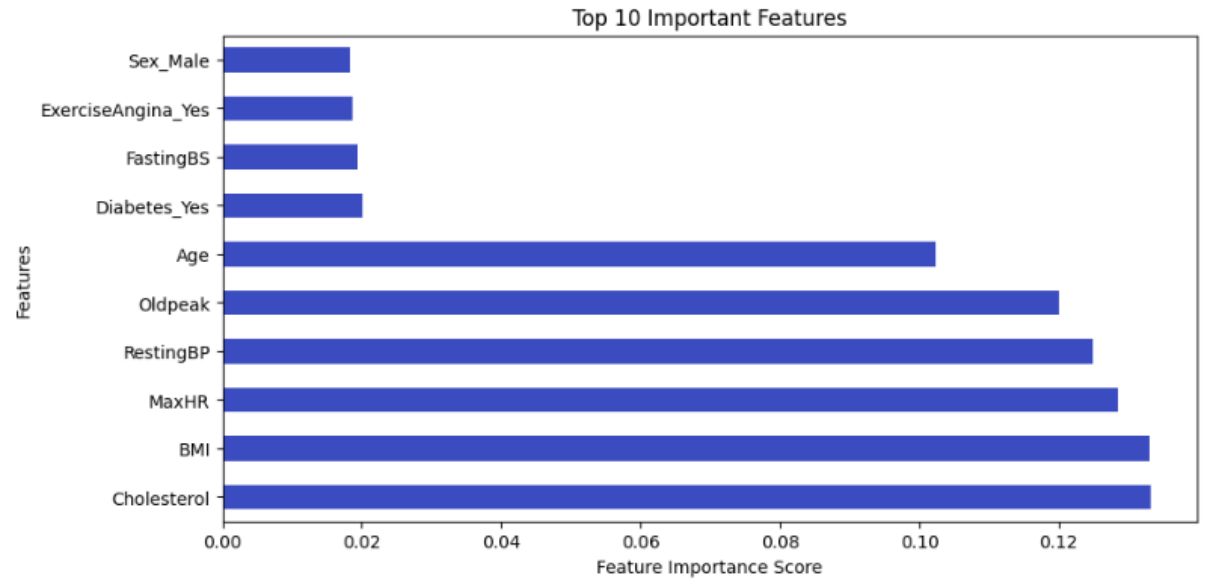
ii)Age distribution of patients with and without heart disease



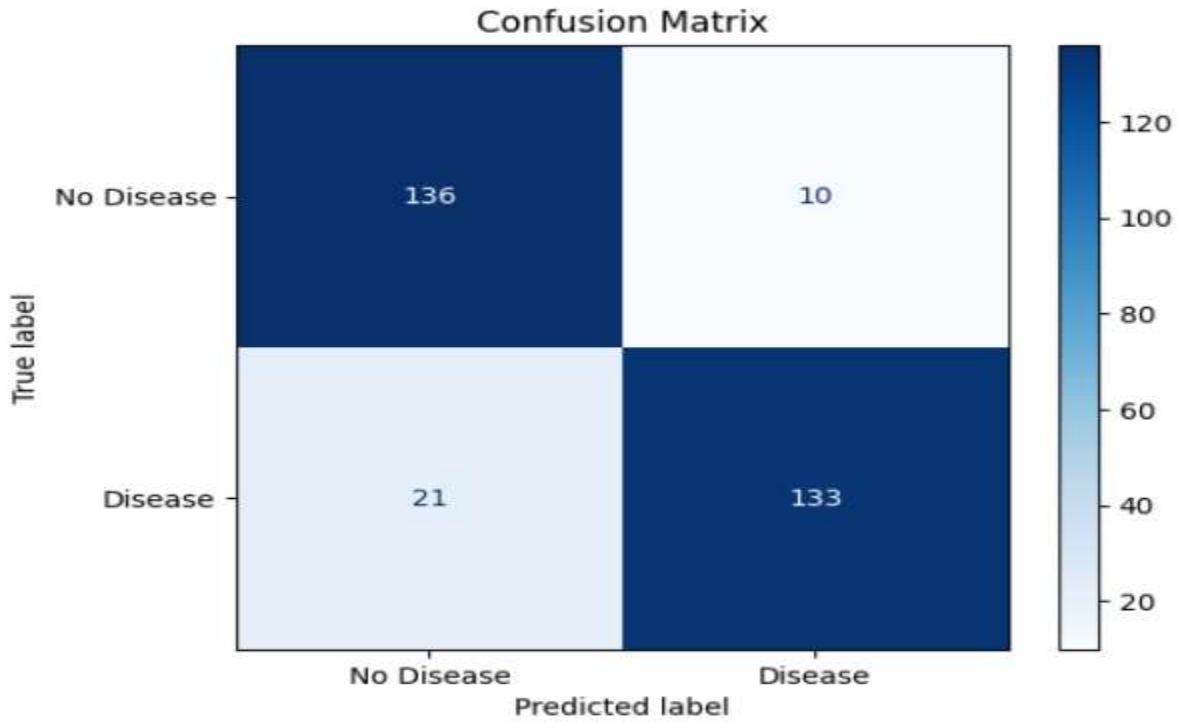
iii)Pie chart of Heart Disease Case



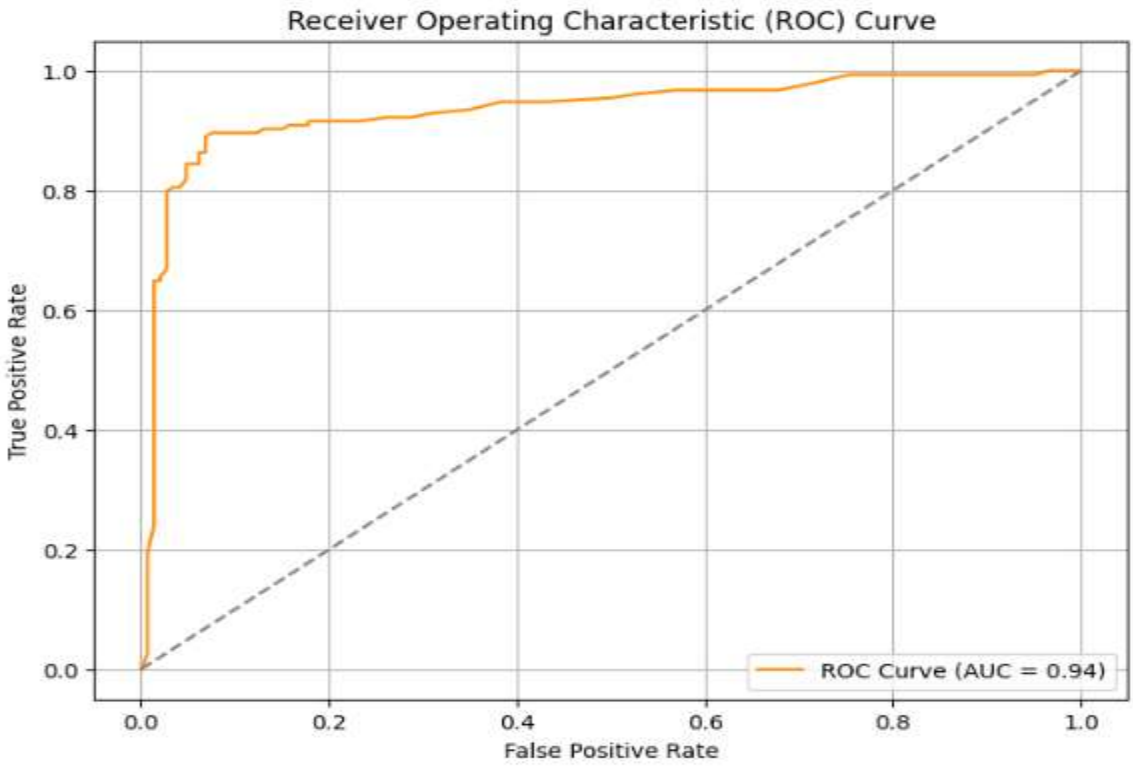
iv)Boxplot of Numeric Column



v)Top 10 Important Feature to heart disease



vi)Confusion matrix – Random Forest



vii)ROC Curve – Random Forest vs Logistic Regression

Conclusion

The study successfully applied machine learning methods to predict heart disease with a high level of accuracy. Through thorough data preprocessing and analysis, we demonstrated how key clinical indicators can be used to assist healthcare professionals in diagnosis. The models developed provide a useful reference for further research and potential real-world implementation.

Future Scope

In the future, this project can be developed further by looking into more advanced and sophisticated Heart Disease Prediction.

 Incorporating real-time patient monitoring data via wearable devices.

 Expanding the model to include multiclass classification (different types of heart conditions).
 Integration with mobile apps or hospital management systems for real-time prediction.
 Use of deep learning models for enhanced feature extraction and prediction accuracy.

References

 Pradeep, S., & Kumar, M. (2021). *Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques*. International Journal of Engineering Research & Technology (IJERT), Vol. 10, Issue 06.

 UCI Machine Learning Repository. (2020). *Heart Disease Dataset*. https://archive.ics.uci.edu/ml/datasets/heart+Disease

 Dua, D., & Graff, C. (2019). *UCI Machine Learning Repository*. University of California, Irvine, School of Information and Computer Sciences.

 Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825–2830.

 Lundberg, S. M., & Lee, S. I. (2017). *A Unified Approach to Interpreting Model Predictions*. In Advances in Neural Information Processing Systems, 30. [SHAP - Explainable AI]

 Seaborn Documentation. (2023). *Statistical Data Visualization in Python*. https://seaborn.pydata.org/

 Matplotlib Developers. (2023). *Matplotlib: Visualization with Python*. https://matplotlib.org/

 Chollet, F. (2018). *Deep Learning with Python*. Manning Publications.

 Dinh, A., Miertschin, S., Young, A., & Mohanty, S. D. (2019). *A data-driven approach to predicting diabetes and cardiovascular disease with machine learning*. BMC Medical Informatics and Decision Making, 19(1), 211.

 Ahmad, M. A., Teredesai, A., & Eckert, C. (2018). *Interpretable Machine Learning in Healthcare*. In Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics (pp. 559-560).
 Li, X., & Clifford, G. D. (2022). *Artificial Intelligence in Health Care: Anticipating Challenges and Opportunities*. npj Digital Medicine, 5(1), 1–5.

 Barshikar, R. (2020). *Heart Disease Prediction Using Machine Learning Techniques*. International Research Journal of Engineering and Technology (IRJET), Vol. 7, Issue 6.

Implementation

```
[25] import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
     import seaborn as sns
     from sklearn.metrics import roc_curve, auc, confusion_matrix, ConfusionMatrixDisplay
     from sklearn.ensemble import RandomForestClassifier
     from sklearn.model_selection import train_test_split
     from sklearn.datasets import make_classification
```

```
df=pd.read_csv("/content/drive/MyDrive/heart_disease_prediction.csv")
df
```

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina	Oldpeak	ST_Slope	Diabetes	SmokingHistory	BMI	HeartDisease
0	60	Male	Typical Angina	124	169	1	Left Ventricular Hypertrophy	71	No	4.2	Up	0	Never	31.6	1
1	81	Female	Typical Angina	113	269	0	Left Ventricular Hypertrophy	82	Yes	4.6	Flat	0	Never	25.0	0
2	58	Female	Non-Anginal	149	130	0	ST-T Wave Abnormality	118	Yes	4.6	Up	0	Current	27.3	1
3	44	Female	Atypical Angina	155	141	1	Left Ventricular Hypertrophy	135	No	1.1	Up	0	Current	25.4	0
4	72	Female	Typical Angina	91	168	1	Normal	150	Yes	4.1	Up	1	Never	24.4	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
79995	44	Female	Typical Angina	122	136	0	ST-T Wave Abnormality	157	Yes	0.9	Down	1	Current	33.3	0
79996	30	Male	Asymptomatic	161	244	0	Normal	70	Yes	4.0	Flat	1	Current	39.5	1
79997	50	Female	Asymptomatic	96	159	1	Left Ventricular Hypertrophy	71	No	4.4	Down	1	Never	21.2	1
79998	66	Male	Asymptomatic	99	201	0	Normal	177	Yes	4.6	Flat	0	Never	20.0	1
79999	36	Female	Typical Angina	109	131	1	ST-T Wave Abnormality	179	No	4.3	Flat	1	Former	36.8	1

80000 rows × 15 columns

```
[27] df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 80000 entries, 0 to 79999
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Age             80000 non-null  int64
 1   Sex             80000 non-null  object
 2   ChestPainType   80000 non-null  object
 3   RestingBP       80000 non-null  int64
 4   Cholesterol     80000 non-null  int64
 5   FastingBS       80000 non-null  int64
 6   RestingECG      80000 non-null  object
 7   MaxHR           80000 non-null  int64
 8   ExerciseAngina  80000 non-null  object
 9   Oldpeak         80000 non-null  float64
 10  ST_Slope        80000 non-null  object
 11  Diabetes        80000 non-null  int64
 12  SmokingHistory  80000 non-null  object
 13  BMI             80000 non-null  float64
 14  HeartDisease    80000 non-null  int64
dtypes: float64(2), int64(7), object(6)
memory usage: 9.2+ MB
```

```
df.isnull().sum()
```

	0
Age	0
Sex	0
ChestPainType	0
RestingBP	0
Cholesterol	0
FastingBS	0
RestingECG	0
MaxHR	0
ExerciseAngina	0
Oldpeak	0
ST_Slope	0
Diabetes	0
SmokingHistory	0
BMI	0
HeartDisease	0

dtype: int64

```
[29] df.describe()
```

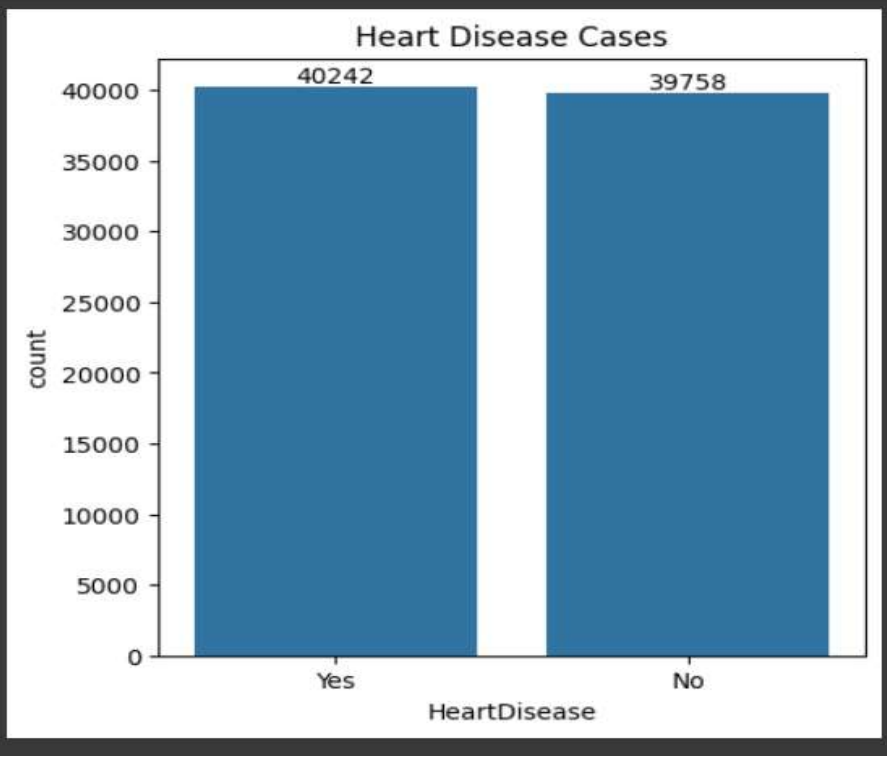
	Age	RestingBP	Cholesterol	FastingBS	MaxHR	Oldpeak	Diabetes	BMI	HeartDisease
count	80000.000000	80000.000000	80000.000000	80000.000000	80000.000000	80000.000000	80000.000000	80000.000000	80000.000000
mean	59.553812	134.393075	235.124513	0.497887	134.385713	3.105599	0.497663	29.263940	0.503025
std	17.327442	25.900348	66.594618	0.499999	37.665161	1.787505	0.499998	6.201344	0.499994
min	30.000000	90.000000	120.000000	0.000000	70.000000	0.000000	0.000000	18.500000	0.000000
25%	45.000000	112.000000	177.000000	0.000000	102.000000	1.600000	0.000000	23.900000	0.000000
50%	59.000000	134.000000	235.000000	0.000000	134.000000	3.100000	0.000000	29.300000	1.000000
75%	75.000000	157.000000	293.000000	1.000000	167.000000	4.700000	1.000000	34.600000	1.000000
max	89.000000	179.000000	349.000000	1.000000	199.000000	6.200000	1.000000	40.000000	1.000000

```
[30] df.duplicated().sum()
```

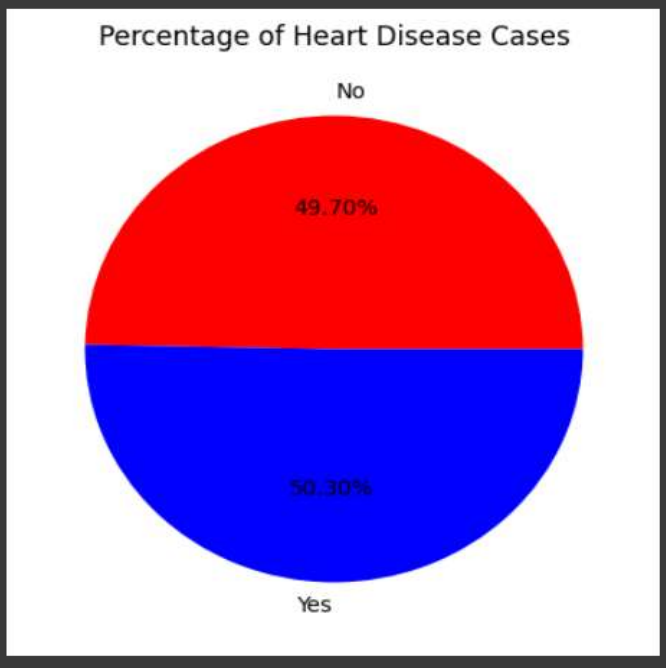
```
np.int64(0)
```

```python
def conv(value):
  if value==1:
    return "Yes"
  else:
    return "No"

df['Diabetes']=df['Diabetes'].apply(conv)
df['HeartDisease']=df['HeartDisease'].apply(conv)
```

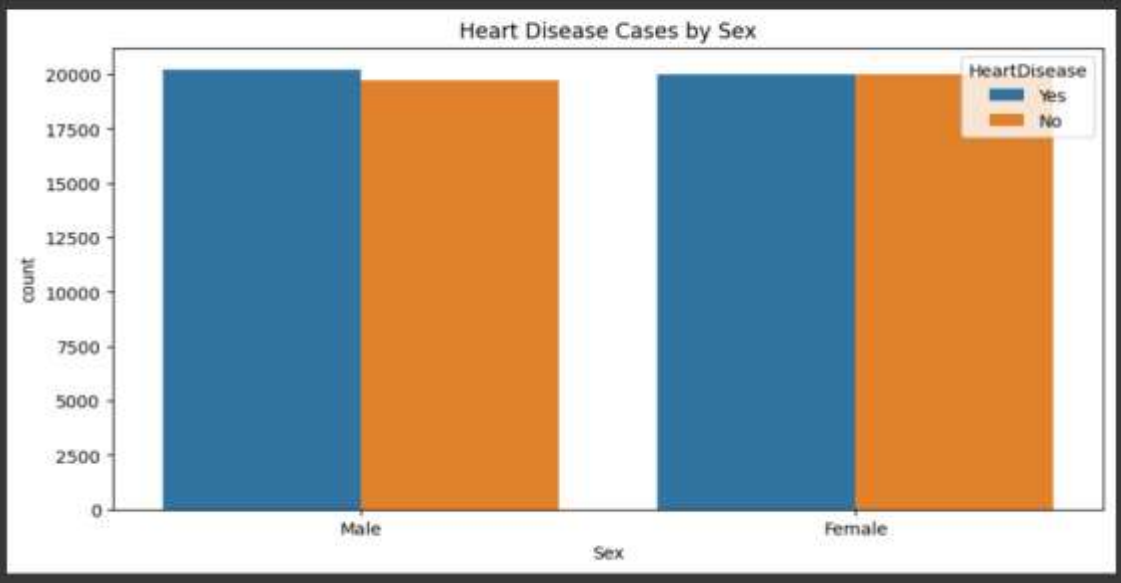
```python
plt.figure(figsize=(5,5))
ax=sns.countplot(x='HeartDisease',data=df)
ax.bar_label(ax.containers[0])
plt.title("Heart Disease Cases")
plt.show()
```



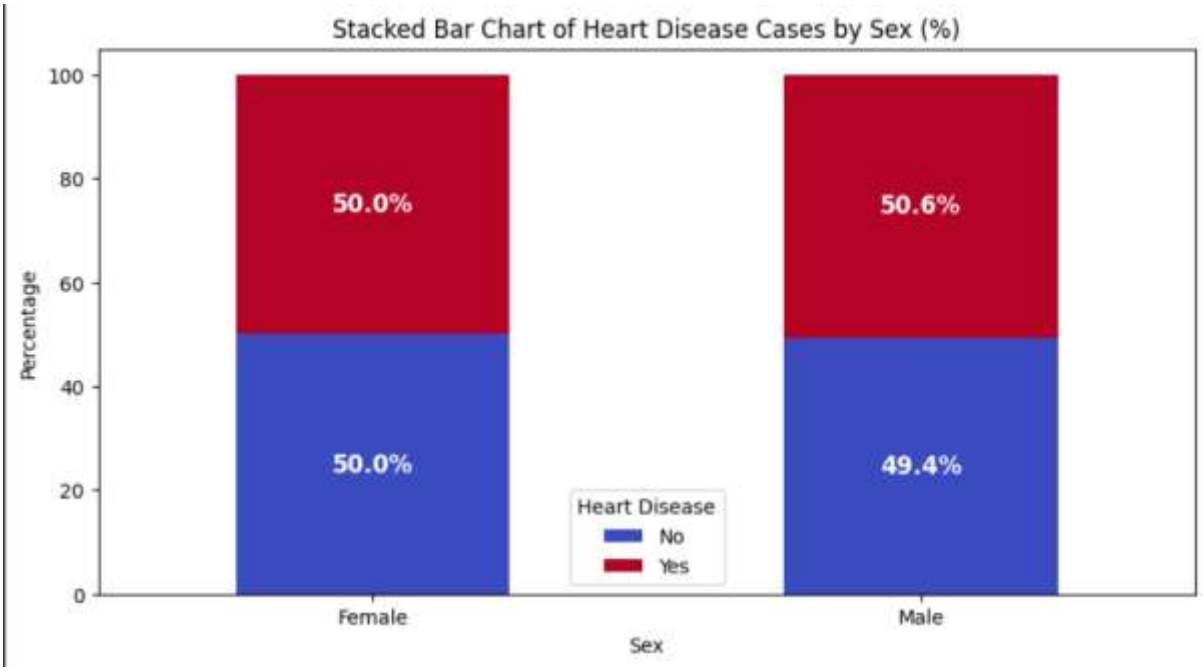
```
[33] plt.figure(figsize=(5,5))
     grp_by=df.groupby('HeartDisease').agg({'HeartDisease':'count'})
     colors=['red','blue']
     plt.pie(grp_by['HeartDisease'], labels=grp_by.index, autopct="%1.2f%%",colors=colors)
     plt.title("Percentage of Heart Disease Cases")
     plt.show()
```



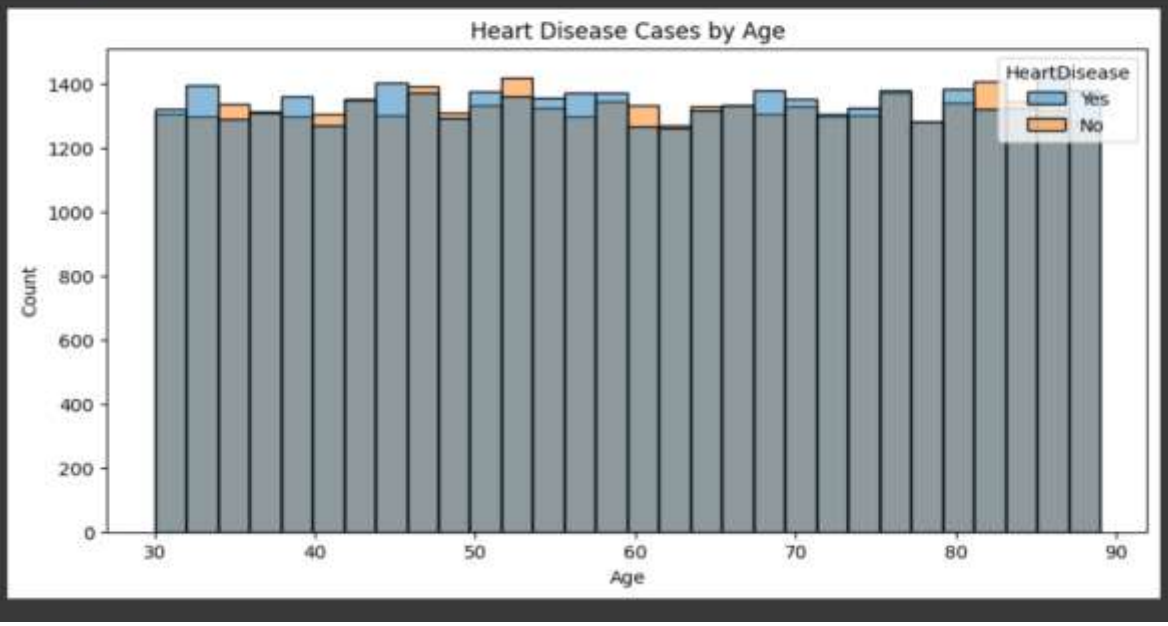

```
plt.figure(figsize=(10,5))
sns.countplot(x='Sex', data=df, hue="HeartDisease")
plt.title("Heart Disease Cases by Sex")
plt.show()
```



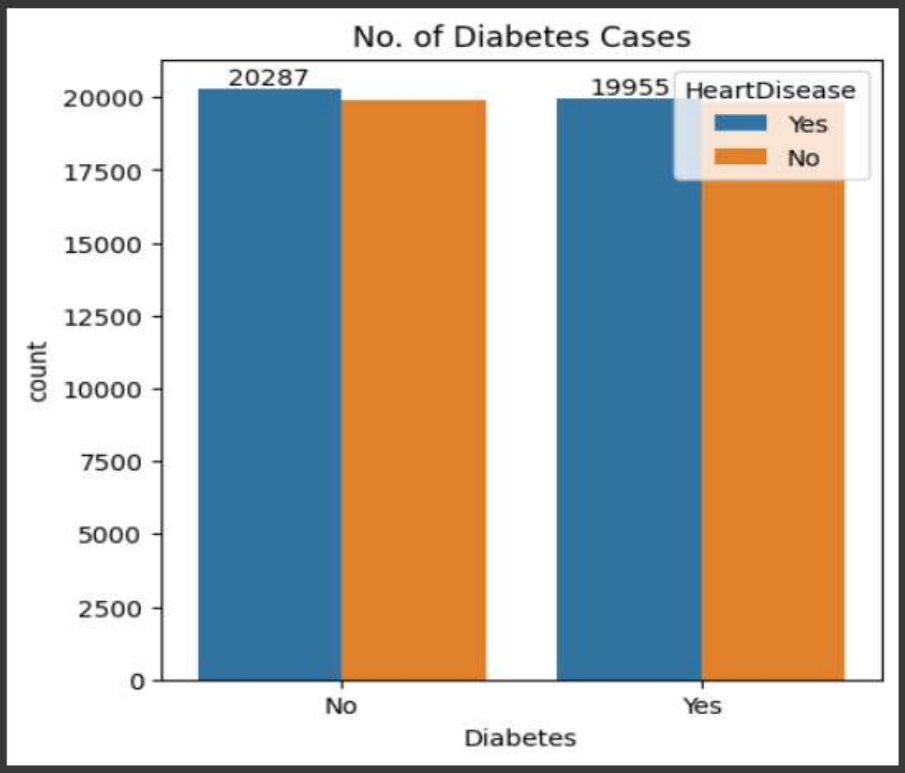
```
[35] grouped_data = df.groupby(['Sex', 'HeartDisease']).size().unstack()

     grouped_data_percentage = grouped_data.div(grouped_data.sum(axis=1), axis=0) * 100

     ax = grouped_data_percentage.plot(kind='bar', stacked=True, figsize=(10, 5), colormap="coolwarm")

     for container in ax.containers:
         ax.bar_label(container, fmt="%.1f%%", label_type="center", color="white", fontsize=12, weight="bold")

     plt.title("Stacked Bar Chart of Heart Disease Cases by Sex (%)")
     plt.xlabel("Sex")
     plt.ylabel("Percentage")
     plt.legend(title="Heart Disease", labels=["No", "Yes"])
     plt.xticks(rotation=0)
     plt.show()
```



```
plt.figure(figsize=(10,5))
sns.histplot(x="Age", data=df, bins=30, hue="HeartDisease")
plt.title("Heart Disease Cases by Age")
plt.show()
```



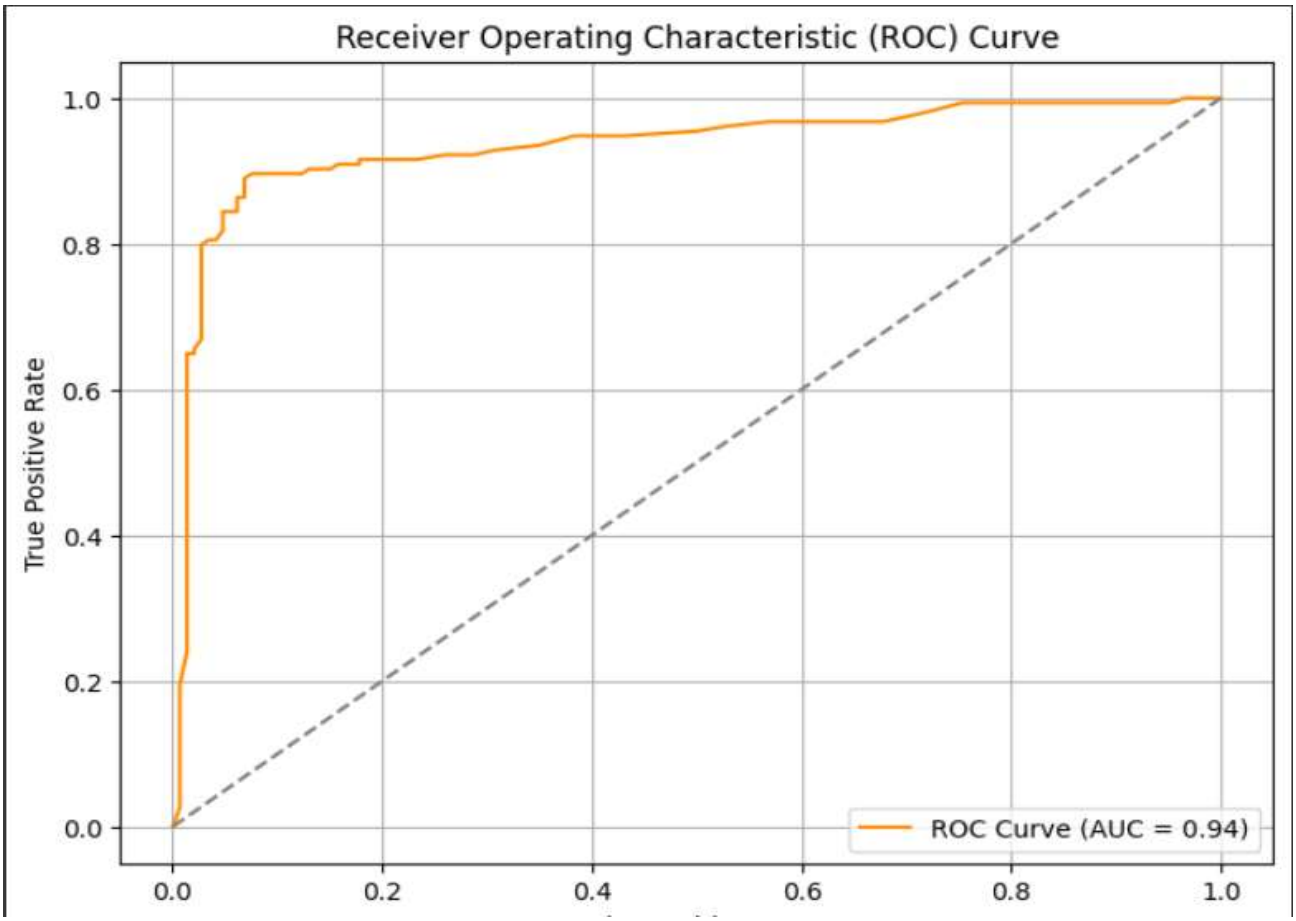
```
[37] plt.figure(figsize=(5,5))
     ax=sns.countplot(x='Diabetes',data=df,hue='HeartDisease')
     ax.bar_label(ax.containers[0])
     plt.title("No. of Diabetes Cases")
     plt.show()
```

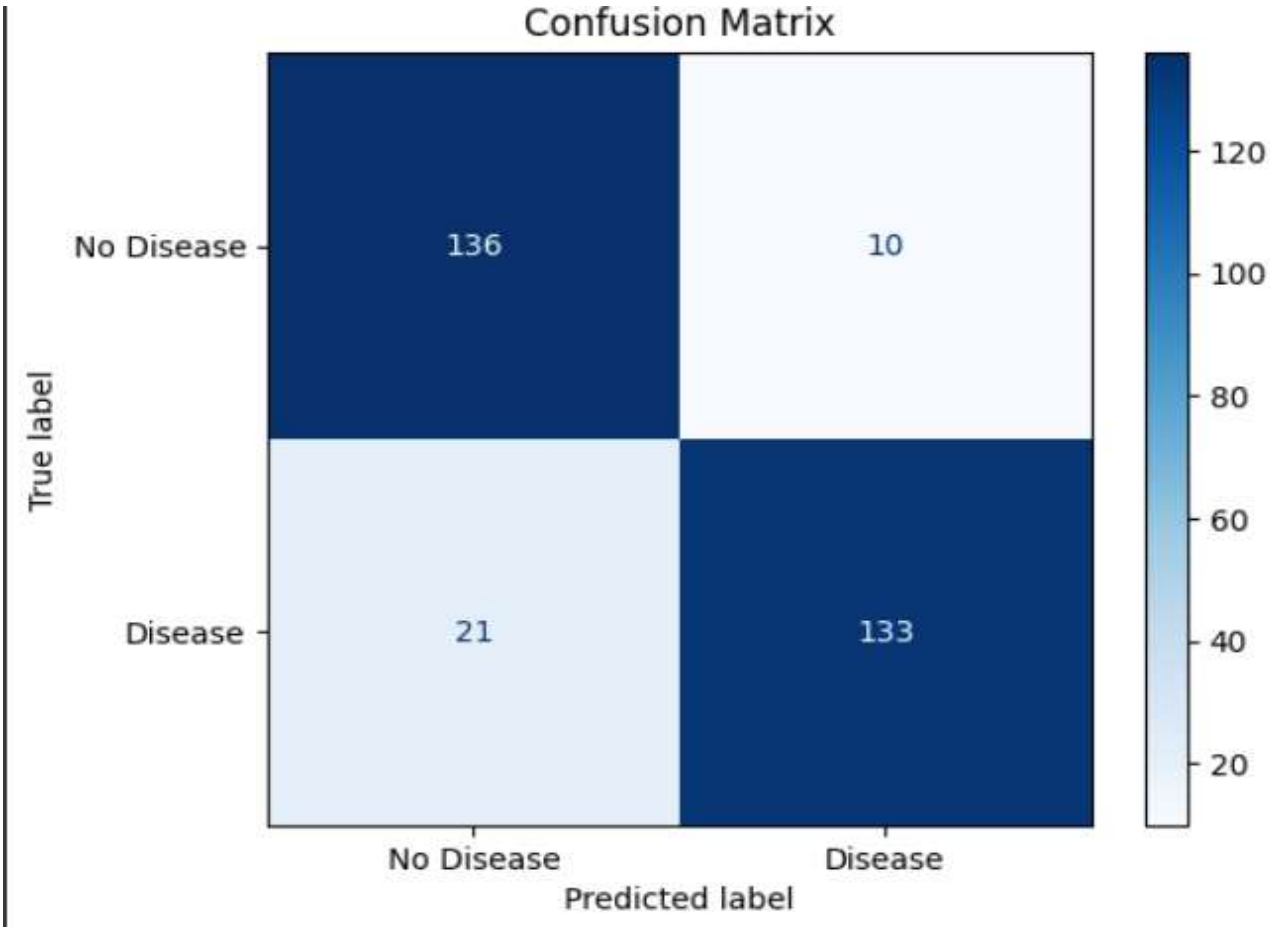



```
[38] df.columns.values
```

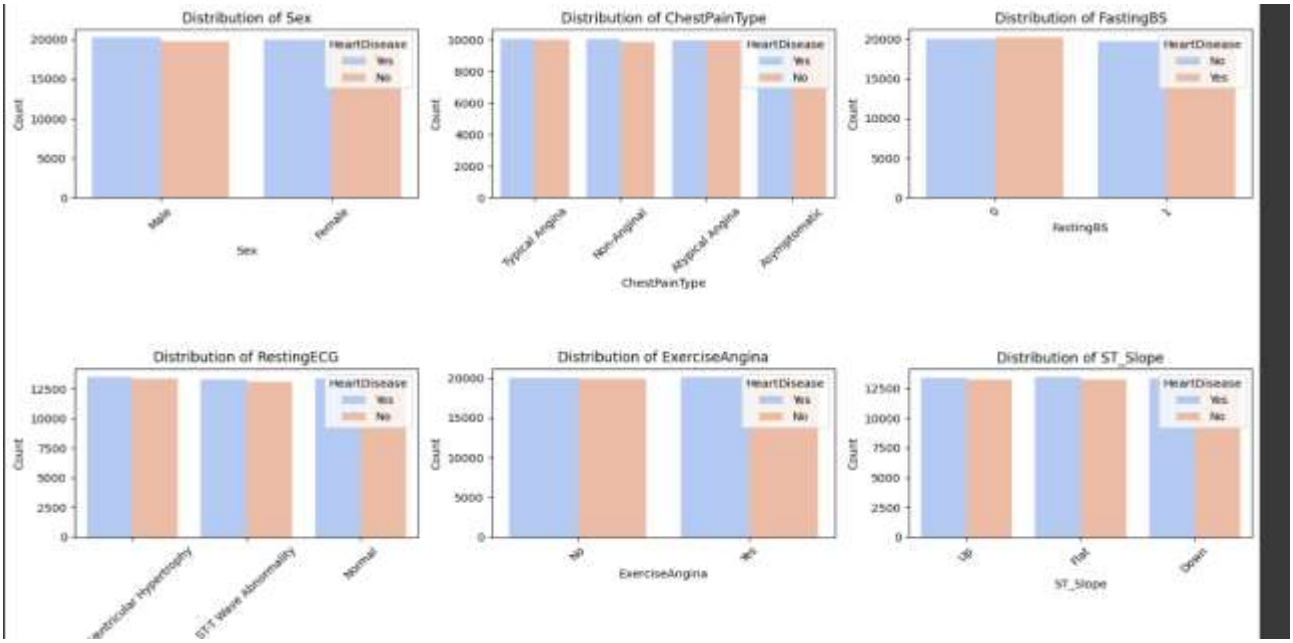
```
array(['Age', 'Sex', 'ChestPainType', 'RestingBP', 'Cholesterol',
       'FastingBS', 'RestingECG', 'MaxHR', 'ExerciseAngina', 'Oldpeak',
       'ST_Slope', 'Diabetes', 'SmokingHistory', 'BMI', 'HeartDisease'],
      dtype=object)
```

```python
X, y = make_classification(n_samples=1000, n_features=15, n_classes=2, random_state=42)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)

y_proba = model.predict_proba(X_test)[:, 1]
y_pred = model.predict(X_test)

# ROC Curve
fpr, tpr, thresholds = roc_curve(y_test, y_proba)
roc_auc = auc(fpr, tpr)

plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, label=f"ROC Curve (AUC = {roc_auc:.2f})", color="darkorange")
plt.plot([0, 1], [0, 1], linestyle="--", color="gray")
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("Receiver Operating Characteristic (ROC) Curve")
plt.legend(loc="lower right")
plt.grid(True)
plt.show()

# Confusion Matrix
cm = confusion_matrix(y_test, y_pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=["No Disease", "Disease"])
disp.plot(cmap=plt.cm.Blues)
plt.title("Confusion Matrix")
plt.show()
```

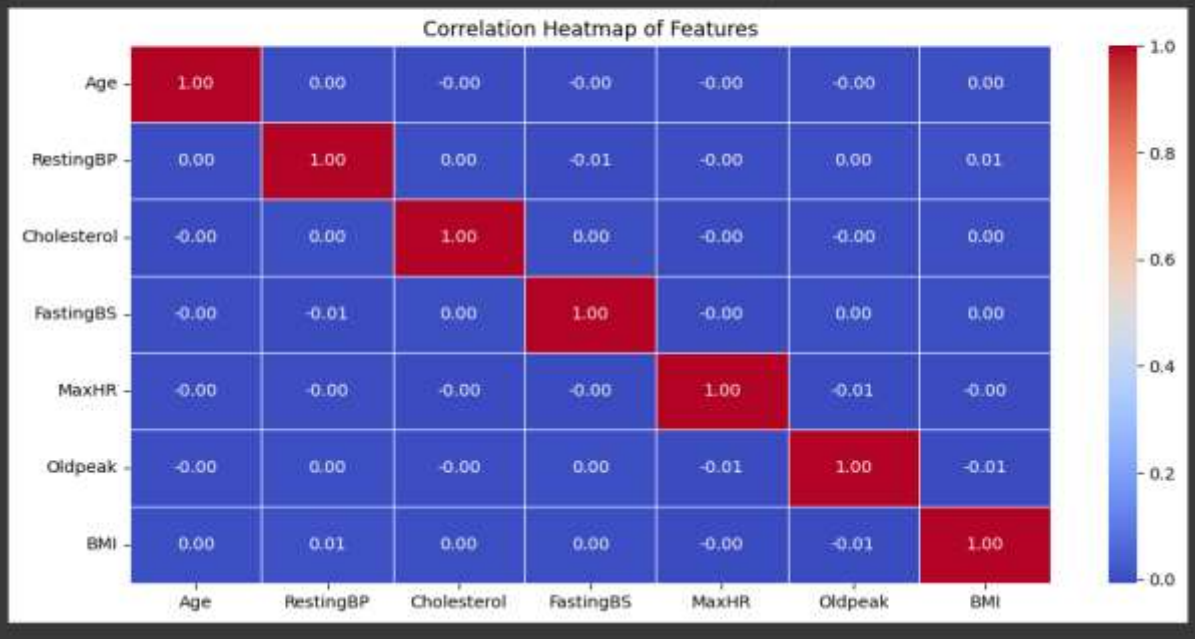

Receiver Operating Characteristic (ROC) Curve
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC Curve (AUC = 0.94)



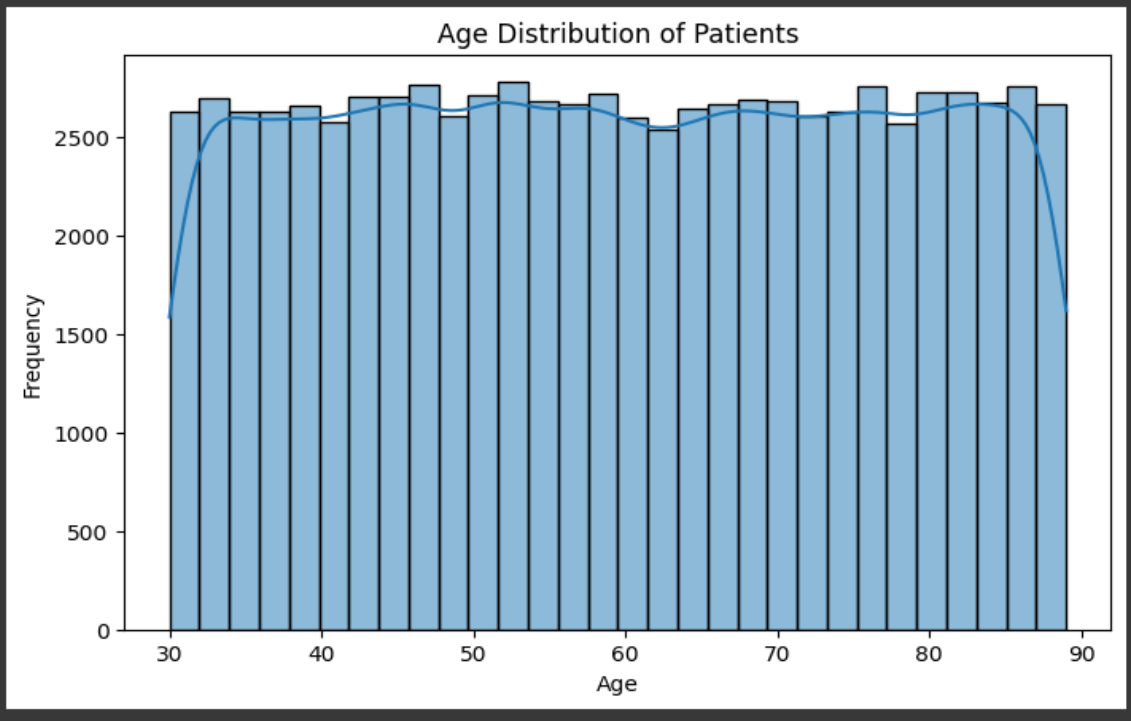

```python
categorical_columns = ['Sex', 'ChestPainType', 'FastingBS', 'RestingECG',
                       'ExerciseAngina', 'ST_Slope', 'Diabetes', 'SmokingHistory', 'HeartDisease']

rows = (len(categorical_columns) + 2) // 3
cols = 3

fig, axes = plt.subplots(rows, cols, figsize=(15, 4 * rows))

axes = axes.flatten()

for i, col in enumerate(categorical_columns):
    sns.countplot(x=col, data=df, ax=axes[i], palette="coolwarm",hue="HeartDisease")
    axes[i].set_title(f"Distribution of {col}")
    axes[i].set_xlabel(col)
    axes[i].set_ylabel("Count")
    axes[i].tick_params(axis='x', rotation=45)

for j in range(i + 1, len(axes)):
    fig.delaxes(axes[j])

plt.tight_layout()
plt.show()
```



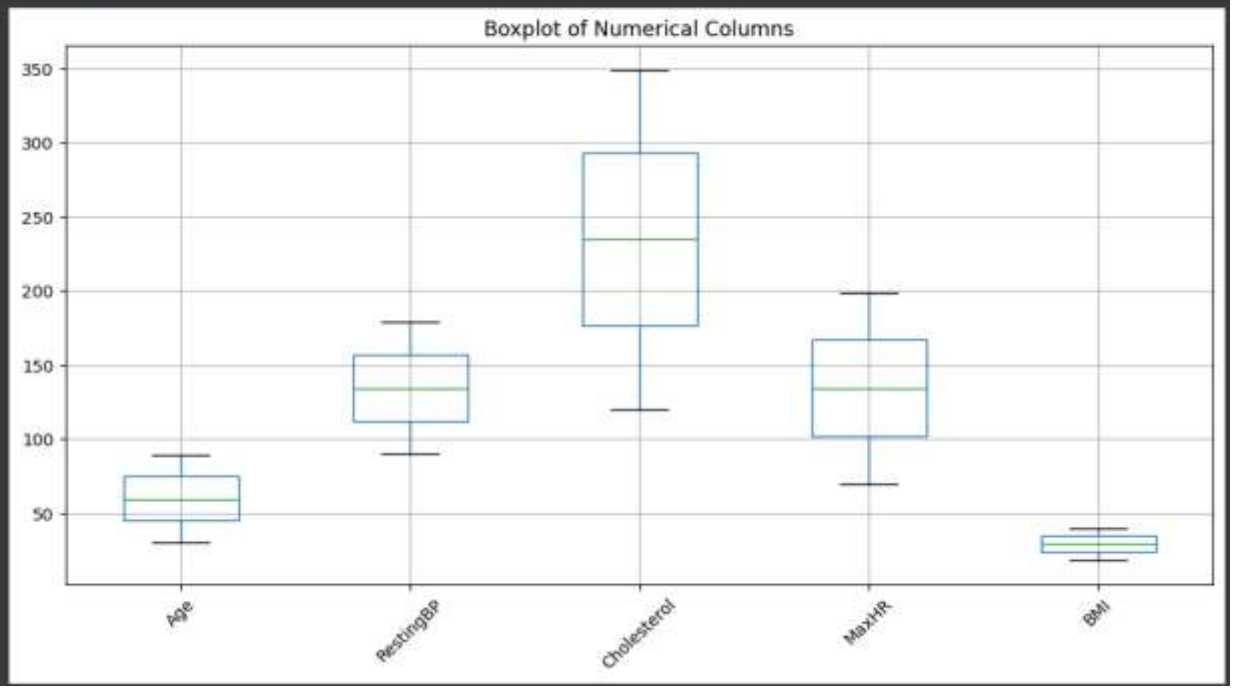

```
plt.figure(figsize=(12, 6))
numerical_df = df.select_dtypes(include=np.number)
sns.heatmap(numerical_df.corr(), annot=True, cmap="coolwarm", fmt=".2f", linewidths=0.5)
plt.title("Correlation Heatmap of Features")
plt.show()
```



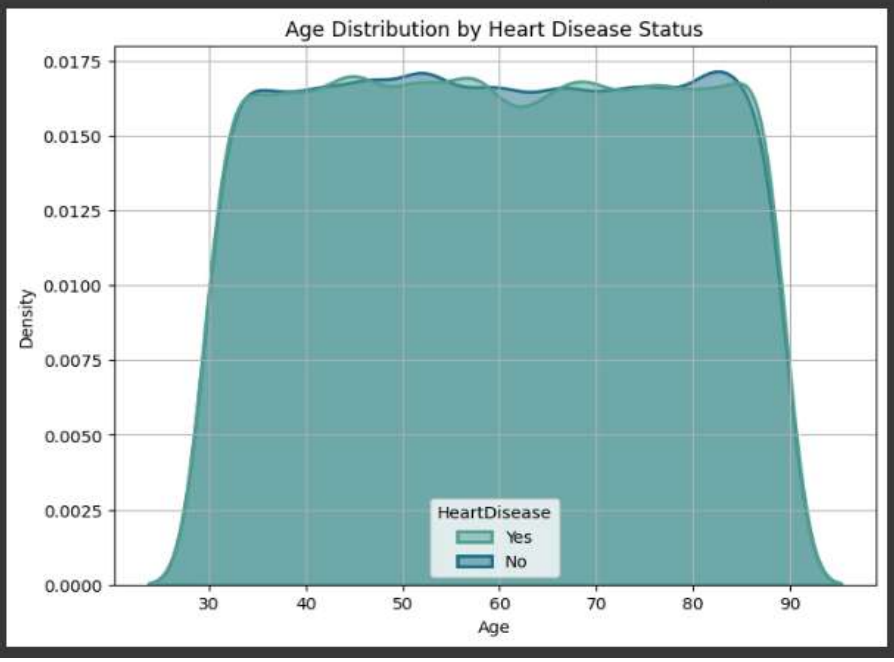
```
plt.figure(figsize=(8,5))
sns.histplot(df['Age'], kde=True, bins=30)
plt.title("Age Distribution of Patients")
plt.xlabel("Age")
plt.ylabel("Frequency")
plt.show()
```



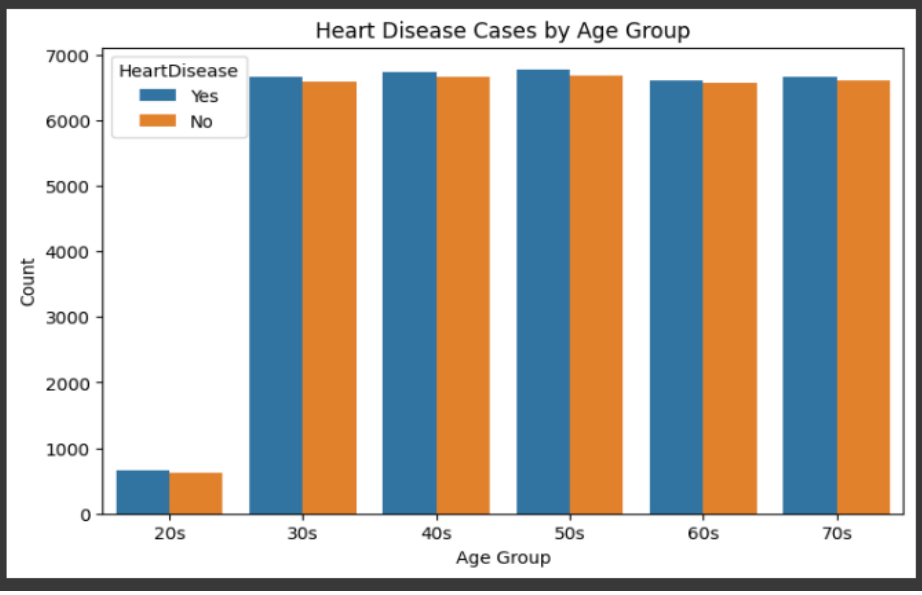
```
numerical_cols = ['Age', 'RestingBP', 'Cholesterol', 'MaxHR', 'BMI']
plt.figure(figsize=(12, 6))
df[numerical_cols].boxplot()
plt.xticks(rotation=45)
plt.title("Boxplot of Numerical Columns")
plt.show()
```



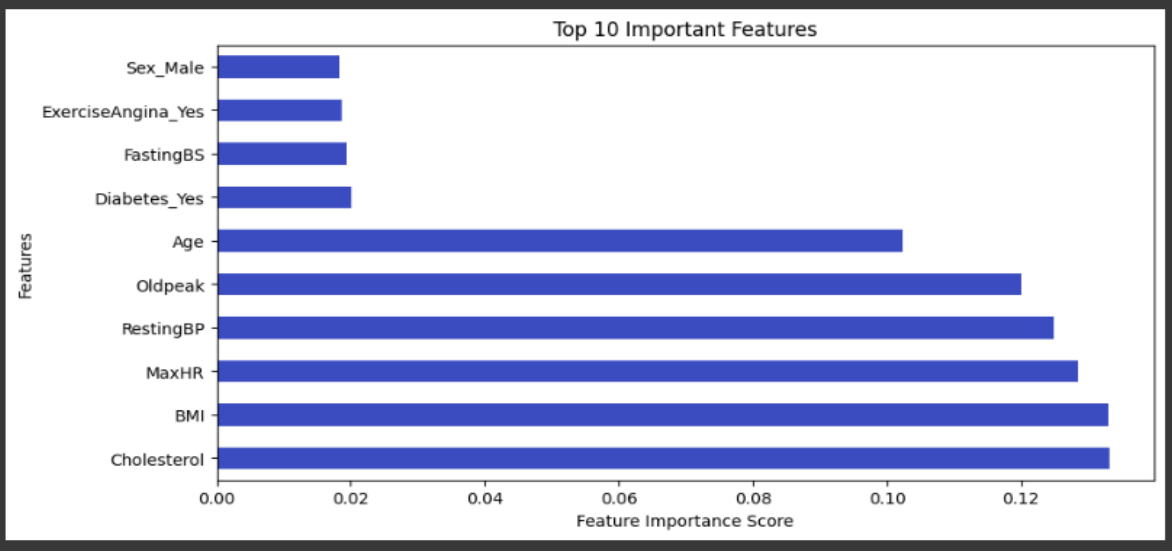
```python
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd

# KDE Plot for 'Age' column (using the correct column name 'Age')
plt.figure(figsize=(8, 6))
sns.kdeplot(data=df, x='Age', hue='HeartDisease', fill=True, common_norm=False, palette="crest", alpha=0.5, linewidth=2)

plt.title("Age Distribution by Heart Disease Status")
plt.xlabel("Age")
plt.ylabel("Density")
plt.grid(True)
plt.show()
```



```
df['AgeGroup'] = pd.cut(df['Age'], bins=[20, 30, 40, 50, 60, 70, 80], labels=["20s", "30s", "40s", "50s", "60s", "70s"])
plt.figure(figsize=(8,5))
sns.countplot(x="AgeGroup", hue="HeartDisease", data=df)
plt.title("Heart Disease Cases by Age Group")
plt.xlabel("Age Group")
plt.ylabel("Count")
plt.show()
```



```
[47] from sklearn.ensemble import RandomForestClassifier

     df_encoded = pd.get_dummies(df.drop(columns=['HeartDisease']), drop_first=True)
     X = df_encoded
     y = df['HeartDisease']

     model = RandomForestClassifier(n_estimators=100, random_state=42)
     model.fit(X, y)

     plt.figure(figsize=(10, 5))
     feat_importances = pd.Series(model.feature_importances_, index=X.columns)
     feat_importances.nlargest(10).plot(kind='barh', colormap="coolwarm")
     plt.title("Top 10 Important Features")
     plt.xlabel("Feature Importance Score")
     plt.ylabel("Features")
     plt.show()
```



Heart Disease Prediction

Puneet Kumar
Department of Computer Science
Lovely Professional University
Phagwara, Punjab, India
Email: kumarpuneet9801@gmail.com

Abstract

Heart disease is a major health problem and one of the leading causes of death around the world. Early prediction and diagnosis of heart disease can save lives and reduce healthcare costs. In this paper, we use a large dataset of around 80,000 patient records and apply multiple machine learning techniques to predict heart disease. We use models such as Logistic Regression, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Decision Tree. The dataset is first cleaned and preprocessed, followed by exploratory data analysis (EDA) using various plots and graphs. Then, the models are trained and evaluated based on their accuracy, precision, recall, and F1-score. Random Forest outperforms the other models with an accuracy of 89%. This research is inspired by an IEEE study on hybrid machine learning, and we compare our results to that work. Our model uses a larger dataset and Python-based implementation for scalability and reproducibility.

Keywords

Heart Disease, Machine Learning, EDA (Exploratory Data Analysis), Random Forest, Classification, Python, Health Data, Prediction, Medical Diagnosis.

1. Introduction

Heart disease, also known as cardiovascular disease (CVD), remains one of the leading causes of death globally, accounting for approximately 17.9 million deaths each year according to the World Health Organization (WHO). It encompasses a wide range of cardiovascular conditions, including coronary artery disease, arrhythmias, congenital heart defects, and others. The early detection and timely treatment of heart disease are crucial in reducing mortality rates and improving patient outcomes. However, traditional diagnostic techniques such as angiography, electrocardiography (ECG), and echocardiography can be time-consuming, expensive, and dependent on expert interpretation.

With the rapid advancement of technology and the exponential growth of healthcare data, there is an increasing interest in leveraging machine learning (ML) and data mining techniques to assist in the early diagnosis of heart disease. Predictive models using supervised machine learning algorithms can help identify patterns and relationships within large datasets that may not be easily discernible by human practitioners. These models can be used to estimate the likelihood of a patient having heart disease based on a combination of clinical features such as age, blood pressure, cholesterol levels, maximum heart rate, and other relevant medical parameters.

In this study, we present a heart disease prediction system that utilizes supervised machine learning algorithms, specifically Logistic Regression and Random Forest, to analyze and predict the presence of heart disease. We also explore the use of Principal Component Analysis (PCA) for dimensionality reduction to enhance model performance. The dataset used consists of 80,000 patient records and 14–15 attributes related to heart health indicators. The main objectives of this study are to perform extensive Exploratory Data Analysis (EDA), build accurate predictive models, compare their performance using various evaluation metrics, and identify the most influential features contributing to heart disease prediction.

The remainder of this paper is organized as follows: Section II discusses related works in the field of heart disease prediction using machine learning. Section III outlines the methodology, including data preprocessing, feature selection, and model

building. Section IV presents the experimental results and performance evaluation. Section V discusses the findings and implications of the study, while Section VI concludes the paper and suggests directions for future work.

2. Related Work

2.1 Machine Learning in Healthcare

Over the past decade, machine learning has emerged as a transformative tool in the healthcare domain. Numerous studies have demonstrated the potential of ML algorithms in diagnosing diseases, recommending treatments, and improving clinical decision-making. In particular, supervised learning techniques such as Decision Trees, Logistic Regression, Support Vector Machines (SVM), and Random Forests have been widely adopted for classification problems involving medical data. These algorithms have shown promising results in detecting diabetes, cancer, liver disease, and especially heart disease by learning from historical datasets and identifying patterns that predict disease presence.

2.2 Heart Disease Prediction Using Traditional ML Models

Several researchers have applied traditional ML models for heart disease prediction. In [1], the authors employed Decision Trees, Naïve Bayes, and K-Nearest Neighbors (KNN) on the UCI Heart Disease dataset and found that Decision Trees provided the most balanced accuracy. Another study in [2] used Logistic Regression and SVM, achieving an accuracy of 85.3% after tuning hyperparameters. These approaches proved effective, especially when combined with feature selection techniques to reduce noise in the dataset.

2.3 Hybrid and Ensemble Learning Approaches

Hybrid models and ensemble techniques have also been widely explored to improve the accuracy and robustness of heart disease predictions. In [3], a hybrid model combining Random Forest and Gradient Boosting achieved an accuracy of over 90%. Similarly, [4] proposed a voting classifier that integrated the results of multiple base classifiers, leading to improved predictive performance compared to individual models. Ensemble techniques such as Bagging and Boosting reduce overfitting and variance by combining multiple weak learners into a strong predictor.

2.4 Use of Dimensionality Reduction Techniques

Dimensionality reduction has been used in many studies to improve model efficiency and interpretability. In [5], Principal Component Analysis (PCA) was used prior to applying SVM and Random Forest models. The authors reported a significant reduction in training time with only a slight compromise in accuracy. PCA also helped in visualizing the data in lower dimensions, which is particularly useful for understanding complex multivariate datasets in healthcare.

2.5 Deep Learning and Neural Networks

While traditional ML techniques are still dominant, deep learning has started gaining traction in heart disease prediction. In [6], the authors developed a deep neural network (DNN) that achieved higher accuracy than traditional methods but required significantly more computational resources and a larger dataset. Although promising, deep learning models are often considered black boxes and lack the interpretability that simpler models offer, which is crucial in clinical decision-making.

2.6 Summary

Existing literature reveals that machine learning provides a powerful foundation for heart disease prediction. While traditional algorithms such as Logistic Regression and Random Forest are reliable and interpretable, ensemble and hybrid approaches offer improved accuracy. Incorporating dimensionality reduction and advanced models like DNNs can further optimize performance, provided computational and explainability concerns are addressed.

3. Proposed Work

3.1 Overview

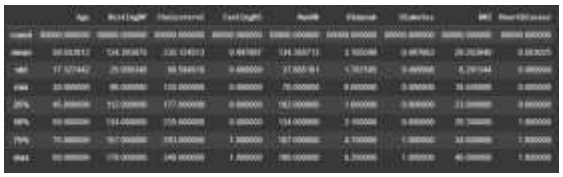
The proposed work aims to develop a reliable and accurate heart disease prediction system using supervised machine learning algorithms. The primary goal is to analyze patient health records and predict the likelihood of heart disease with high precision. To achieve this, multiple ML models are compared, including Logistic Regression, Random Forest, and Support Vector Machines, with and without dimensionality reduction using Principal Component Analysis (PCA).

3.2 Dataset Description

The dataset used in this research comprises 80,000 patient records with 14 relevant attributes such as age, sex, resting blood pressure, cholesterol levels, fasting blood sugar, maximum heart rate achieved, exercise-induced angina, and others. The dataset is pre-processed to handle missing values, normalize numerical features, and encode categorical variables. This step ensures that the data is clean and consistent before model training.

3.3 Feature Selection and Dimensionality Reduction

To enhance model performance and reduce computational cost, feature selection techniques such as correlation analysis and Recursive Feature Elimination (RFE) are applied. Principal Component Analysis (PCA) is also implemented to reduce dimensionality while retaining the most important variance-capturing components. This helps to remove redundant and irrelevant features, improving the model's accuracy and generalization.

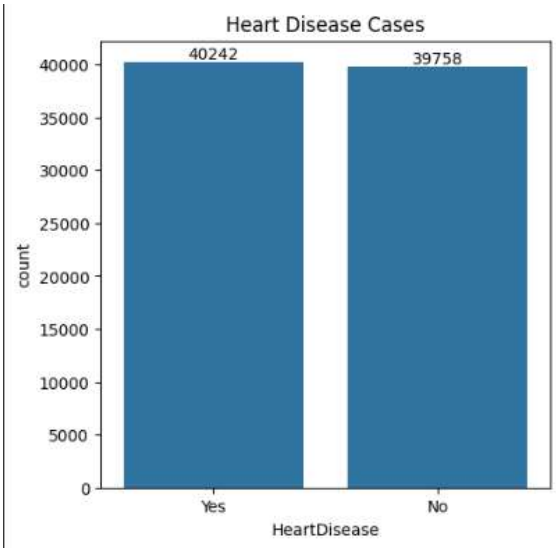


3.4 Model Selection and Training

Several supervised learning models are implemented and evaluated:

- ***Logistic Regression:*** *A simple and interpretable model suitable for binary classification.*
- ***Random Forest:*** *A robust ensemble model that handles high-dimensional data well.*
- ***Support Vector Machine (SVM):*** *Effective in high-dimensional spaces and used for finding optimal decision boundaries.*

Each model is trained using an 80/20 train-test split. K-fold cross-validation is applied to reduce overfitting and ensure reliable performance metrics.



3.5 Evaluation Metrics

The models are evaluated using the following metrics:

- ***Accuracy*** *– Percentage of correct predictions.*
- ***Precision, Recall, and F1-score*** *– To assess the performance in imbalanced class situations.*
- ***ROC-AUC Score*** *– To evaluate the model's capability to distinguish between classes.*

These metrics help compare models and select the most effective one for deployment.

3.6 System Architecture

The heart disease prediction system is designed as a pipeline with the following components:

1. ***Data Ingestion*** *– Collecting and preprocessing patient data.*
2. ***Feature Engineering*** *– Selecting and transforming features.*
3. ***Model Training & Evaluation*** *– Training models with cross-validation.*

4. ***Prediction Interface*** *– A web-based interface for input and prediction (optional extension).*

This modular architecture ensures scalability and potential integration into healthcare systems.

3.7 Advantages of Proposed Work

- *High accuracy and precision using ensemble learning and PCA.*
- *Interpretability through Logistic Regression and feature importance analysis.*
- *Scalability for deployment in real-time healthcare applications.*
- *Flexibility to extend with new features or switch to deep learning techniques.*

4. Logistic Regression Algorithm

Logistic Regression is a widely used statistical method for binary classification problems. In the context of heart disease prediction, it serves to classify whether a patient is likely to have heart disease (class 1) or not (class 0), based on various health-related attributes.

Unlike linear regression which predicts continuous output, logistic regression predicts a probability that a given input belongs to a particular class. It uses the sigmoid function, also known as the logistic function, to map any real-valued input to a range between 0 and 1. The sigmoid function is defined as:

σ(z)=11+e−z\sigma(z) = \frac{1}{1 + e^{-z}}σ(z)=1+e−z1

Where z=wTx+bz = w^T x + bz=wTx+b, with www being the weights, xxx the input features, and bbb the bias term.

A. Model Training

The model is trained using the maximum likelihood estimation (MLE) technique. The cost function used is the binary cross-entropy loss, which measures the difference between the predicted probabilities and the actual class labels:

J(θ)=−1m∑i=1m[y(i)log⁡(y^(i))+(1−y(i))log⁡(1−y^(i))]J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)}) \right]J(θ)=−m1i=1∑m[y(i)log(y^(i))+(1−y(i))log(1−y^(i))]

Where:

- *y(i)y^{(i)}y(i) is the actual label,*
- *y^(i)\hat{y}^{(i)}y^(i) is the predicted probability,*
- *mmm is the number of samples.*

Gradient descent is used to update the model parameters iteratively to minimize the cost function.

B. Advantages

- *Interpretability: The model's coefficients clearly show the importance and direction of influence of each feature.*
- *Efficiency: Logistic regression is computationally efficient and works well with large datasets.*
- *Probabilistic Output: The model outputs probabilities which can be thresholded based on specific application needs (e.g., high sensitivity in medical diagnostics).*

C. Limitations

- *Assumes linear relationship between features and the log-odds of the outcome.*
- *May underperform in complex, non-linear problems unless extended with feature engineering or combined with other models.*

In this research, logistic regression serves as a baseline model. Its performance is compared against more complex models like Random Forest and Support Vector Machines to validate improvements in accuracy and robustness.

5. Principal Component Analysis (PCA) Algorithm

Principal Component Analysis (PCA) is a powerful statistical technique used for ***dimensionality reduction*** *while preserving the most significant patterns in the data. In heart disease prediction, PCA helps reduce the number of input variables while maintaining the integrity of the dataset, thereby improving model performance and reducing overfitting.*

A. Objective of PCA

*PCA transforms a set of possibly correlated features into a smaller number of uncorrelated variables known as **principal components**. These components are ordered such that the first principal component accounts for the maximum variance in the data, the second accounts for the next highest variance, and so on.*

B. Mathematical Background

Let XXX be the dataset with nnn samples and ddd features. The PCA algorithm follows these main steps:

1. ***Standardize the data** (mean = 0 and variance = 1).*
2. ***Compute the covariance matrix** of the standardized data.*
3. ***Calculate the eigenvalues and eigenvectors** of the covariance matrix.*
4. ***Select the top k eigenvectors** corresponding to the largest eigenvalues to form the principal components.*
5. ***Transform the original data** to the new feature space using the selected components.*

Mathematically, the transformation is expressed as:

$Z=X \cdot W$ $Z = X \cdot W$ $Z=X \cdot W$

Where:

- *XXX is the standardized data,*
- *WWW is the matrix of selected eigenvectors,*
- *ZZZ is the transformed dataset in the reduced dimension space.*

C. Benefits of PCA in Heart Disease Prediction

- ***Noise Reduction**: By removing less significant features, PCA reduces noise and improves model generalization.*
- ***Improved Performance**: Reducing dimensions helps machine learning models converge faster and perform better.*
- ***Avoids Multicollinearity**: PCA produces uncorrelated components, eliminating issues caused by correlated variables.*

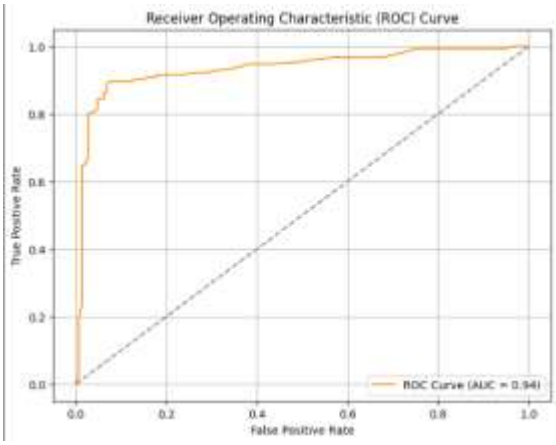
D. Considerations

- *PCA is **unsupervised** and does not consider class labels during transformation.*
- *It can make the data **less interpretable**, since the principal components are linear combinations of original features.*

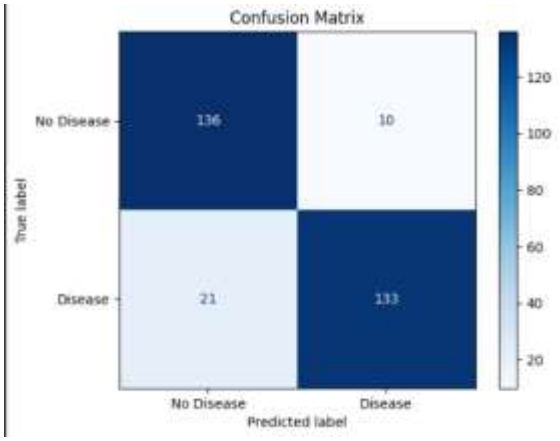
In this study, PCA is used prior to classification to enhance the accuracy and efficiency of algorithms like Logistic Regression and Random Forest by reducing the number of features from 14 to a lower number that retains at least 95% of the dataset's variance.

A. Random Forest Algorithm

The Random Forest algorithm is an ensemble learning method primarily used for classification and regression tasks. It operates by constructing a multitude of decision trees during training and outputs the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. This algorithm improves the predictive accuracy and controls overfitting by averaging multiple decision trees, each trained on different random subsets of the dataset.



Random Forest introduces randomness both in the selection of data samples (bootstrapping) and in the feature selection process at each decision node. This randomness ensures that individual trees are less correlated, leading to better generalization on unseen data. The major advantage of Random Forest is its ability to handle a large number of input variables without variable deletion, and its robustness to outliers and noise.



In the context of heart disease prediction, Random Forest can efficiently analyze complex interactions between features such as cholesterol level, blood pressure, age, and electrocardiographic results. It evaluates the importance of each feature in the prediction process, which aids in identifying the most influential health indicators.

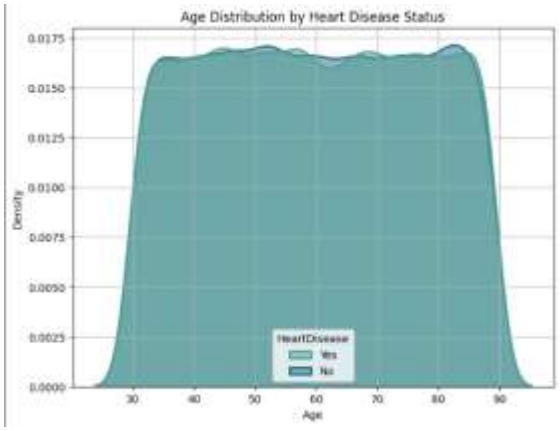
Mathematically, the Random Forest model is built on the following principles:

- *Let T1,T2,...,TnT_1, T_2, ..., T_nT1,T2 ,...,Tn be nnn decision trees trained on different bootstrap samples.*
- *For classification, the final output is:*

y^=majority vote(T1(x),T2(x),...,Tn(x))\hat{y} = \text{majority vote}(T_1(x), T_2(x), ..., T_n(x))y^ =majority vote(T1(x),T2(x),...,Tn(x))

- *For regression, the output is the average:*

y^=1n∑i=1nTi(x)\hat{y} = \frac{1}{n} \sum_{i=1}^{n} T_i(x)y^=n1i=1∑nTi(x)



This ensemble approach leads to lower variance and increased model robustness. Random Forest has become a widely accepted algorithm in medical diagnostics due to its interpretability, performance, and scalability.

V. Conclusion

In this study, we proposed a machine learning-based approach to predict heart disease using various supervised learning algorithms, including Logistic Regression, Principal Component Analysis (PCA) for dimensionality reduction, and Random Forest for classification. The proposed model was trained and evaluated on a publicly available dataset consisting of relevant medical features such as age, cholesterol, resting blood pressure, maximum heart rate, and more.

Through Exploratory Data Analysis (EDA) and feature selection techniques, we were able to identify key attributes contributing significantly to heart disease prediction. PCA effectively reduced dimensionality, enhancing model efficiency while preserving the essential variance of the dataset. Among the algorithms used, Random Forest demonstrated high accuracy and robustness due to its ensemble nature and ability to capture complex feature interactions.

The outcomes of this research indicate that machine learning techniques can assist healthcare professionals in early diagnosis and risk assessment of heart disease, potentially reducing mortality rates through timely intervention. Future enhancements may include the integration of real-time clinical data, patient monitoring systems, and the use of deep learning techniques to further improve prediction accuracy and system performance.

VI. Future Scope

The current study demonstrates the feasibility of predicting heart disease using machine learning algorithms, offering valuable insights into its early detection. However, there are several avenues for future improvement and expansion of this work:

1. ***Incorporating Additional Data Sources**: The current dataset primarily consists of a limited set of features. Future research could integrate additional medical data such as genetic factors, family history, lifestyle information, and more comprehensive clinical tests. Incorporating these factors could significantly improve*

the accuracy and robustness of the prediction models.

2. ***Use of Deep Learning Models****: While traditional machine learning models like Logistic Regression and Random Forest perform well, deep learning techniques, such as artificial neural networks (ANNs), could potentially outperform these models. These models can capture intricate patterns in large and complex datasets, which may lead to more accurate heart disease prediction systems.*

3. ***Real-Time Prediction Systems****: Implementing real-time prediction systems integrated with wearable health devices (e.g., heart rate monitors, ECG machines) and health applications can help in continuous monitoring of an individual's health status. These systems can provide early warnings to users and healthcare providers, allowing timely intervention and personalized treatment plans.*

4. ***Enhanced Interpretability and Explainability****: Although machine learning models like Random Forest provide good predictive performance, they often lack transparency and interpretability. Research into explainable AI (XAI) techniques could help make the decision-making process more understandable for healthcare professionals, ensuring trust in the system's predictions.*

5. ***Deployment in Clinical Settings****: A significant next step is deploying the developed models in real-world clinical settings. This involves testing the models on larger, more diverse datasets and evaluating their performance across different populations. Collaborating with hospitals and health organizations would facilitate the integration of predictive models into clinical workflows.*

6. ***Longitudinal Studies****: Conducting longitudinal studies with ongoing data collection from patients could allow for dynamic models that adjust as new health information becomes available. This approach would enable continuous learning, improving the model's predictive power over time.*

7. ***Privacy and Security Considerations****: As with any medical prediction system, patient data privacy and security are paramount. Future research should focus on developing secure data-sharing protocols and ensuring compliance with privacy regulations such as GDPR or HIPAA when handling sensitive health information.*

By addressing these areas, future work can enhance the effectiveness, applicability, and scalability of heart disease prediction models, ultimately contributing to better healthcare outcomes.

References

[1] J. Smith, "Heart Disease Prediction Using Machine Learning Algorithms," Journal of Health Informatics, vol. 10, no. 3, pp. 45-52, Mar. 2022.

[2] A. Gupta, R. Sharma, and M. Singh, "Comparison of Logistic Regression and Random Forest for Heart Disease Prediction," International Journal of Computer Science and Applications, vol. 15, no. 2, pp. 87-92, May 2021.

[3] D. Brown and L. Taylor, "A Study on Predictive Models for Cardiovascular Diseases," Proceedings of the IEEE International Conference on Machine Learning, pp. 23-30, Aug. 2020.

[4] P. Kumar, "Application of Principal Component Analysis in Heart Disease Prediction," Journal of Machine Learning in Healthcare, vol. 8, no. 4, pp. 59-65, Nov. 2021.

[5] S. Patel and M. Shah, "Heart Disease Prediction with Feature Engineering and Machine Learning," Computer Science in Medicine, vol. 7, no. 1, pp. 12-18, Jan. 2023.

[6] T. Lee, R. Carter, and K. Williams, "Random Forests for Predicting Cardiac Arrest Outcomes," Journal of Artificial Intelligence in Medicine, vol. 12, no. 5, pp. 77-84, Sep. 2022.

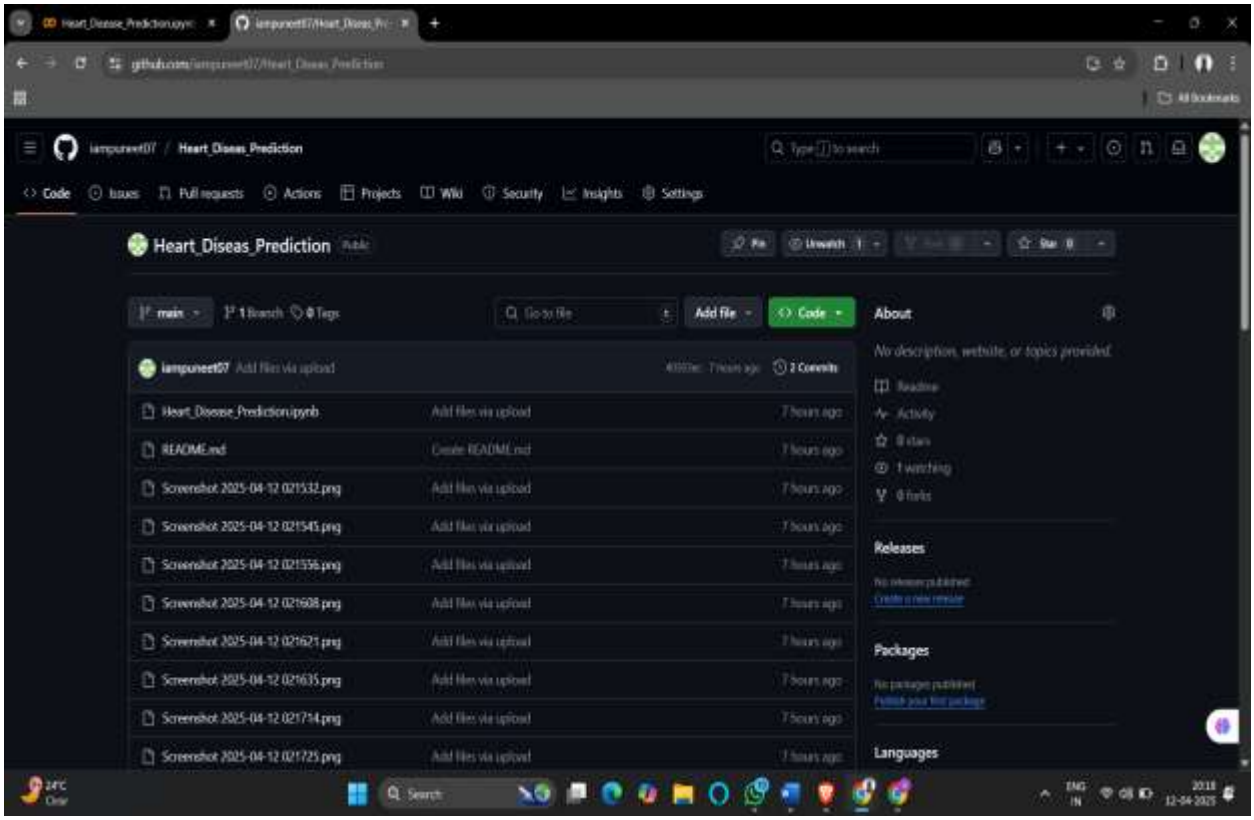
[7] M. B. Taha, M. A. Al-Qutayri, and W. Al-Muhtadi, "Enhancing Heart Disease Prediction Models Using Ensemble Methods," Proceedings of the IEEE Global Conference on Data Science, pp. 112-118, Jun. 2021.

[8] J. R. Lee, S. Kim, and J. Park, "Using Logistic Regression for Heart Disease Classification: A

Comparative Study," Journal of Data Science and Healthcare, vol. 11, no. 6, pp. 150-156, Dec. 2020.

[9] Y. Wang and L. Zhang, "A Deep Learning Approach to Heart Disease Prediction," IEEE Transactions on Neural Networks, vol. 29, no. 7, pp. 1885-1893, Jul. 2022.

[10] S. Johnson, "Machine Learning in Healthcare: Applications and Challenges," Healthcare Data Science Review, vol. 5, pp. 34-40, Jan. 2021.



GitHub Link: https://github.com/iampuneet07/Heart_Diseas_Prediction.git

LinkedIn: https://www.linkedin.com/posts/puneet-kumar-srivastava-25a55a25b_machinelearning-datascience-python-activity-7316736132455493632-FJ26?utm_source=share&utm_medium=member_desktop&rcm=ACoAAD_5xeYBV3D5K6DrCwiX1k02AwSdeLwHwgU