Problem

Heart disease remains a leading cause of death worldwide. Early and accurate detection of heart disease is crucial for timely intervention and improved patient outcomes. Delayed diagnosis can lead to serious complications and even mortality.

Goal

This project aims to develop an automated machine learning model capable of accurately detecting heart disease in individuals based on their health metrics, ultimately leading to better patient outcomes.

DATASET DESCRIPTION:

a) **Type of Data:** Numerical
b) **Size:** The dataset consists of [**1,025**] records, each with [**14**] health metrics.
c) **Attributes (Features):** The dataset comprises several numerical features, including:

Column Name	Description
age	Age of the patient
sex	Sex of the patient (1 = male, 0 = female)
cp	Chest pain type (1 = typical angina, 2 = atypical angina, 3 = non-anginal pain, 4 = asymptomatic)
trestbps	Resting blood pressure
chol	Serum cholesterol in mg/dL
fbs	Fasting blood sugar > 120 mg/dL (1 = true, 0 = false)
restecg	Resting electrocardiographic results (0 = normal, 1 = having ST-T wave abnormality, 2 = having QRS complex abnormalities)
thalach	Maximum heart rate achieved during exercise
exang	Exercise induced angina (1 = yes, 0 = no)
oldpeak	ST depression induced by exercise relative to rest
slope	The slope of the peak exercise ST segment (1 = upsloping, 2 = flat, 3 = downsloping)
ca	Number of major vessels (0-3) colored by fluoroscopy
thal	Thallium defect (3 = normal, 1 = fixed defect, 0 = reversible defect)
target	Presence of heart disease (1 = yes, 0 = no)

d) Statistical Presentation Of Dataset:

	count	mean	std	min	25%	50%	75%	max
age	1025.0	54.434146	9.072290	29.0	48.0	56.0	61.0	77.0
sex	1025.0	0.695610	0.460373	0.0	0.0	1.0	1.0	1.0
cp	1025.0	0.942439	1.029641	0.0	0.0	1.0	2.0	3.0
trestbps	1025.0	131.611707	17.516718	94.0	120.0	130.0	140.0	200.0
chol	1025.0	246.000000	51.592510	126.0	211.0	240.0	275.0	564.0
fbs	1025.0	0.149268	0.356527	0.0	0.0	0.0	0.0	1.0
restecg	1025.0	0.529756	0.527878	0.0	0.0	1.0	1.0	2.0
thalach	1025.0	149.114146	23.005724	71.0	132.0	152.0	166.0	202.0
exang	1025.0	0.336585	0.472772	0.0	0.0	0.0	1.0	1.0
oldpeak	1025.0	1.071512	1.175053	0.0	0.0	0.8	1.8	6.2
slope	1025.0	1.385366	0.617755	0.0	1.0	1.0	2.0	2.0
ca	1025.0	0.754146	1.030798	0.0	0.0	0.0	1.0	4.0
thal	1025.0	2.323902	0.620660	0.0	2.0	2.0	3.0	3.0
target	1025.0	0.513171	0.500070	0.0	0.0	1.0	1.0	1.0

By analyzing these statistics, we can gain insights into the distribution of different features in the dataset, which can be useful for understanding the underlying patterns and relationships between variables:

1-Age: The dataset includes patients aged between 29 and 77, with an average age of 54.4 years.

2-Sex: The dataset is relatively balanced between males and females.

3-Chest Pain Type (CP): The dataset includes various types of chest pain, with a mean value of 0.94.

4-Resting Blood Pressure (RestBP): The average resting blood pressure is 131.61 mm Hg, with a range from 94 to 200 mm Hg.

5-Cholesterol: The average cholesterol level is 246 mg/dl, with a range from 126 to 564 mg/dl.

6-Fasting Blood Sugar (FBS): A small proportion of patients had fasting blood sugar levels above 120 mg/dl.

7-Resting Electrocardiographic Result (RestECG): The dataset includes various resting electrocardiographic results.

8-Maximum Heart Rate Achieved (MaxHR): The maximum heart rate achieved by patients ranged from 71 to 202 bpm, with an average of 149.11 bpm.

9-Exercise-Induced Angina (ExAng): A significant proportion of patients experienced exercise-induced angina.

10-ST Depression Induced by Exercise Relative to Rest (ST_Depression): The average ST depression is 1.07 mm, with a range from 0 to 6.2 mm.

11-Slope of the Peak Exercise ST Segment (Slope): The dataset includes various slopes of the peak exercise ST segment.

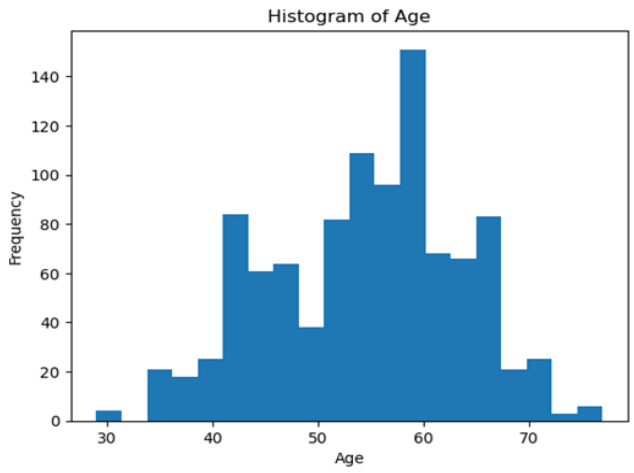
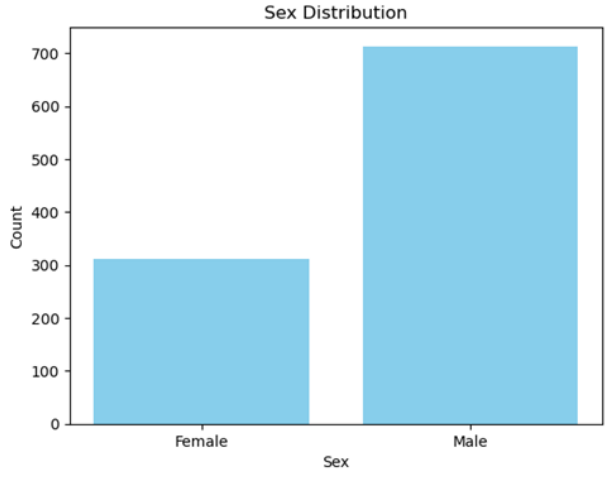
12-Number of Major Vessels (0-3) Colored by Flourosopy (Num_Major_Vessels): The number of major vessels colored by fluoroscopy ranges from 0 to 4, with an average of 0.75.

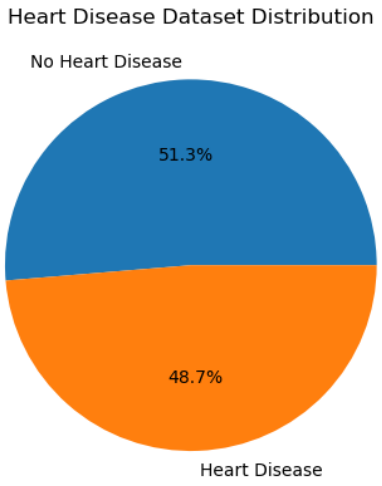
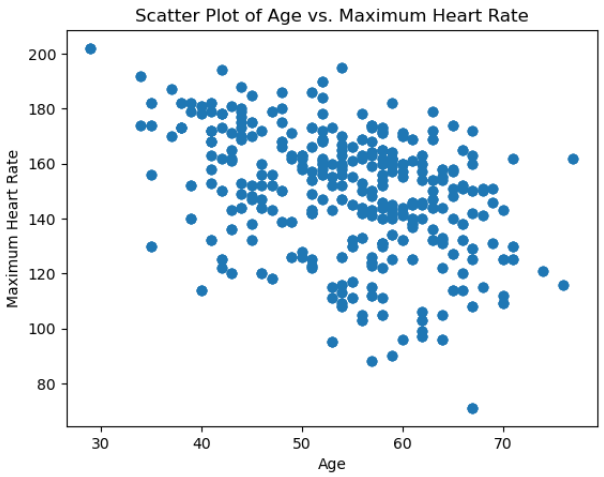
13-Thallium Stress Result (Thal): The dataset includes various thallium stress results.

14-Target (Heart Disease Presence): The dataset is relatively balanced between patients with and without heart disease.

e) Graphical Presentation Of Dataset:

Here is a graphical presentation of some attributes in dataset:

Age Distribution	Sex Distribution
	

Heart Disease Distribution	Age vs. Maximum Heart Rate
	

COMPARISON:

1- Project original code (Before Adding ML):

This's the original project code, contains a single machine learning method, specifically a Decision Tree Classifier, to detect heart disease in individuals based on various health metrics.

a-The original code:

```
from sklearn.tree import DecisionTreeClassifier

dt= DecisionTreeClassifier()
dt.fit(X_train,y_train)

# Make predictions on the testing set
y_pred = dt.predict(X_test)

# Evaluate the model's performance
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy :", accuracy)
```

b- Accuracy: 98.54%

c-Output:

```
Accuracy : 0.9853658536585366
```

d-Drawbacks of relying only on a Decision Tree model:

1. While Decision Trees are a powerful tool for classification tasks, relying solely on this method may not be sufficient to ensure the most accurate results.

2. Decision Trees may not capture complex relationships between features, potentially missing important contributing factors to heart disease.

2- Project modified code (After Adding ML methods):

For the previous reasons, we will add three more models to the project, **Linear Regression, Logistic Regression**, and **Random Forest Classifier**.

By evaluating the performance of these models alongside the existing Decision Tree Classifier, we aim to improve the overall accuracy and reliability of heart disease detection project.

a-Here we imported needed libraries to add our methods:

```
#Importing the basic librarires for building model - classification

from sklearn.impute import SimpleImputer
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score,r2_score

from sklearn.linear_model import LinearRegression
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier

from sklearn.preprocessing import LabelEncoder
from sklearn.inspection import permutation_importance
```

b-Then, we implemnted machine learning models:

```
li = [LinearRegression(), LogisticRegression(), DecisionTreeClassifier(), RandomForestClassifier()]
d = {}
for i in li:
    i.fit(X_train, y_train)
    if isinstance(i, LinearRegression):              # add Linear Regression ML method
        ypred = i.predict(X_test)
        r2 = r2_score(y_test, ypred) * 100
        print(f"Linear Regression: {round(r2, 2)}%")
        d.update({str(i): round(r2, 2)})

    elif isinstance(i, LogisticRegression):          # add Logistic Regression ML method
        ypred = i.predict(X_test)
        acc = accuracy_score(y_test, ypred) * 100
        print(f"Logistic Regression: {round(acc, 2)}%")
        d.update({str(i): round(acc, 2)})

    elif isinstance(i, DecisionTreeClassifier):      # Decision Tree Classifier, from the original code
        ypred = i.predict(X_test)
        acc = accuracy_score(y_test, ypred) * 100
        print(f"Decision Tree Classifier: {round(acc, 2)}%")
        d.update({str(i): round(acc, 2)})

    elif isinstance(i, RandomForestClassifier):      # add Random Forest Classifier ML method
        ypred = i.predict(X_test)
        acc = accuracy_score(y_test, ypred) * 100
        print(f"Random Forest Classifier: {round(acc, 2)}%")
        d.update({str(i): round(acc, 2)})
```

c- The output:

```
Linear Regression: 40.96%
Logistic Regression: 78.54%
Decision Tree Classifier: 98.54%
Random Forest Classifier: 98.54%
```

By comparing the performance of these models, we can gain insights into the underlying patterns in the data and select the most suitable models for heart disease detection.

ML METHODS:

a-Linear Regression:

Linear regression is a type of supervised machine learning algorithm that computes the linear relationship between the dependent variable and one or more independent features by fitting a linear equation to observed data.

Why it underperformed ?

1.Nature of Linear Regression:

- Linear Regression is designed for continuous numerical predictions rather than classification tasks. In this case, the target variable represents the presence or absence of heart disease (a binary variable), making it unsuitable for linear regression.

2.Complex Relationships:

- The relationships between features (e.g., age, cholesterol, maximum heart rate) and the target might be nonlinear or involve interactions. Linear Regression cannot model such complexities effectively.

3.Metric Mismatch:

- Classification tasks typically optimize for metrics like recall, precision, or F1-score. Linear Regression minimizes the mean squared error, which doesn't align with the project's goals of maximizing recall for detecting heart disease.

4.Comparison with Other Models:

- Models like Logistic Regression or Random Forest are more appropriate for classification tasks. They inherently support binary outputs and can handle complex patterns, which Linear Regression cannot.

b-Logistic Regression:

Logistic Regression is a machine learning algorithm used primarily for classification tasks. It predicts the probability of an instance belonging to a particular class, typically in scenarios with two possible outcomes, such as "yes" or "no" (binary classification).

Why it underperformed ?

Logistic Regression assumes a linear relationship between features and the target class. If the relationship is non-linear, as is often the case in medical datasets, it struggles. It doesn’t account for interactions unless explicitly modeled, limiting its effectiveness for complex data.

c-Decision Tree Classifier:

One machine learning model for classification and regression applications is the decision tree. It creates a tree structure by splitting the data according to feature values, with the leaves representing the final prediction and each node representing a decision based on a feature.

It is employed for several reasons:

1. Interpretability:
    - It is simple to see and comprehend decision trees. In the medical field, this is crucial because doctors have to analyze the model's conclusions and determine which characteristics—such as age and cholesterol level—are most essential in predicting heart disease.
2. Manages Mixed Data Types:
    - The dataset contains both category (such as the type of chest pain) and numerical (such as age and cholesterol) data. Both categories are easily handled by decision trees.
3. Non-linear Relationships:
    - Complex, non-linear relationships between features may be present in heart disease data.
4. Feature Selection:
    - The algorithm simplifies the modeling process and concentrates on the most important factors by automatically choosing the most crucial features for prediction.

d-Random Forest Classifier:

Random Forest is a machine learning algorithm used for classification and regression tasks. It works by building multiple decision trees during training and combining their outputs to make more accurate predictions.

How Random Forest Works?

- It creates many decision trees using random subsets of the dataset and features.
- Each tree gives its prediction (e.g., whether a patient has heart disease or not).
- The final prediction is based on the majority vote of all trees.

Why It Outperformed?

1. Better Accuracy:
    - Combining multiple trees reduces the errors that might occur with a single Decision Tree.
2. Robustness:
    - It performs well with noisy data and avoids overfitting by using random subsets of the dataset and features.
3. Generalisation:
    - It's effective at making predictions on new, unseen data, ensuring reliable performance on the test dataset.

RESULTS AND CONCLUSION:

Output:

```
Linear Regression: 40.96%
Logistic Regression: 78.54%
Decision Tree Classifier: 98.54%
Random Forest Classifier: 98.54%
```

-Linear Regression

Linear Regression performed poorly, with an accuracy of 40.96%. This method is not designed for binary classification tasks, making it unsuitable for heart disease detection.

-Logistic Regression

Logistic Regression achieved a moderate accuracy of 78.54%. This result is due to its assumption of linearity between features and the target variable, which limits its ability to model the complex, non-linear interactions inherent in the dataset.

-Decision Tree Classifier

The Decision Tree Classifier achieved the highest accuracy (98.54%) by effectively handling the non-linear and hierarchical relationships in the dataset. Its ability to split data based on feature importance allows it to adapt to the complexity of medical indicators.

-Random Forest Classifier

Random Forest matched the performance of the Decision Tree Classifier, with an accuracy of 98.54%. As an ensemble method, Random Forest builds multiple Decision Trees and combines their outputs, improving the model's robustness, reducing overfitting, and enhancing generalization on unseen data.

Why Decision Tree and Random Forest Are the Most Suitable ?

- **Non-linear Relationships:** They capture complex patterns and interactions between medical features, such as age, cholesterol levels, and blood pressure.

- **Threshold-based Decisions:** They split the data based on critical thresholds, reflecting how certain health indicators affect heart disease risk.

- **Interpretability and Reliability:** While Decision Trees are easy to interpret, Random Forest offers higher reliability by reducing overfitting through ensemble learning.

Conclusion

Given the dataset's complexity, Random Forest and Decision Tree Classifiers provide the most accurate and reliable predictions. These methods are ideal for heart disease detection, where capturing intricate relationships between features is crucial for making precise and impactful medical decisions.

Sources & References:

-The original project link:
https://www.kaggle.com/code/syedali110/heart-disease-detection/notebook

-Dataset link from the original project:
https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset

-The notebook of modified code link:
https://drive.google.com/file/d/1MRjAfCwOd-VAdlT5NXbdnWtCYrV9GT3d/view?usp=drive_link

or a copy to show from the code:

..\Downloads\Heart_Disease_Prediction.html