# Project Report: Assignment-

---

## 1. Introduction

Heart disease is a leading cause of death globally. Early prediction of heart disease can aid in making crucial decisions and potentially save lives. This project aims to develop a machine learning model to predict the presence of heart disease in a patient based on a set of medical attributes. We will use a classic dataset containing patient data and employ a Logistic Regression model for classification.

---

## 2. Dataset Description

The project utilizes the `cleaned_merged_heart_dataset.csv` https://www.kaggle.com/datasets/mfarhaannazirkhan/heart-dataset , which contains 14 attributes collected from patients. The primary goal is to predict the `target` variable, where **1 indicates the presence of heart disease** and **0 indicates its absence**.

The features included in the dataset are:

- `age`: Age of the patient (Numeric).
- `sex`: Gender of the patient (1 = male, 0 = female).
- `cp`: Chest pain type (0 = Typical angina, 1 = Atypical angina, 2 = Non-anginal pain, 3 = Asymptomatic).
- `trestbps`: Resting Blood Pressure (in mm Hg).
- `chol`: Serum Cholesterol level (in mg/dl).
- `fbs`: Fasting blood sugar > 120 mg/dl (1 = true, 0 = false).
- `restecg`: Resting electrocardiographic results (0 = Normal, 1 = ST-T wave abnormality, 2 = Left ventricular hypertrophy).
- `thalachh`: Maximum heart rate achieved.
- `exang`: Exercise-induced angina (1 = yes, 0 = no).
- `oldpeak`: ST depression induced by exercise relative to rest.
- `slope`: Slope of the peak exercise ST segment (0 = Upsloping, 1 = Flat, 2 = Downsloping).
- `ca`: Number of major vessels (0-4) colored by fluoroscopy.
- `thal`: Thalassemia (1 = normal; 2 = fixed defect; 3 = reversible defect).
- `target`: Presence of heart disease (1 = yes, 0 = no).

---

# 3. Exploratory Data Analysis (EDA)

Before building the model, we performed an exploratory data analysis to understand the relationships between different variables.

### Correlation Matrix

A correlation matrix was generated to visualize the linear relationships between all attributes in the dataset.

**(You would insert the image `image_7fe809.png` here)**

**Observations:**

- The `target` variable shows a notable positive correlation with `cp` (chest pain type), `thalachh` (maximum heart rate), and `slope`.
- The `target` variable has a strong negative correlation with `exang` (exercise-induced angina), `oldpeak`, `ca` (number of major vessels), and `sex`. This indicates that lower values for these features are associated with a higher likelihood of heart disease.

### Pair Plot

A pair plot was created to visualize the relationships between a subset of key features: `age`, `trestbps`, `chol`, `thalachh`, and the `target` variable.

**(You would insert the image `image_7fe82a.png` here)**

**Observations:**

- The distribution of patients with and without heart disease (`target` = 1 and 0, respectively) shows some separation across different features.
- For instance, patients with higher `thalachh` (maximum heart rate) are more likely to have heart disease. Conversely, older patients (`age`) appear to have a higher incidence of the disease.

---

# 4. Model Implementation

A Logistic Regression model was chosen for this classification task due to its simplicity, interpretability, and efficiency.

### Step 1: Importing Libraries and Loading Data

First, we import the necessary libraries and load the dataset into a pandas DataFrame.

Python
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import mean_squared_error, r2_score, accuracy_score

# Load the dataset
df = pd.read_csv('cleaned_merged_heart_dataset.csv')
```

## Step 2: Data Preparation

The dataset is split into features (`X`) and the target variable (`y`).

Python
```
# Separate features (X) and target (y)
X = df.drop('target', axis=1)
y = df['target']
```

## Step 3: Splitting the Data

The data is divided into training (80%) and testing (20%) sets to train the model and evaluate its performance on unseen data.

Python
```
# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

## Step 4: Model Training

A Logistic Regression model is instantiated and trained on the training data.

Python
```
# Initialize and train the Logistic Regression model
model = LogisticRegression(max_iter=1000) # Increased max_iter for convergence
model.fit(X_train, y_train)
```

## Step 5: Prediction

The trained model is used to make predictions on the test set. For accuracy calculation, we also define a classification prediction based on a 0.5 threshold.

Python

```
# Make predictions
y_pred = model.predict(X_test)
y_pred_class = (y_pred > 0.5).astype(int) # Thresholding for classification metrics
```

---

# 5. Results and Evaluation

The model's performance was evaluated using several metrics. While Mean Squared Error and $R^2$ are typically used for regression, they are included here as per the notebook. The most important metric for this classification task is **Accuracy**.

## Evaluation Code

Python
```
# Evaluation
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
accuracy = accuracy_score(y_test, y_pred_class)
```

## Model Performance

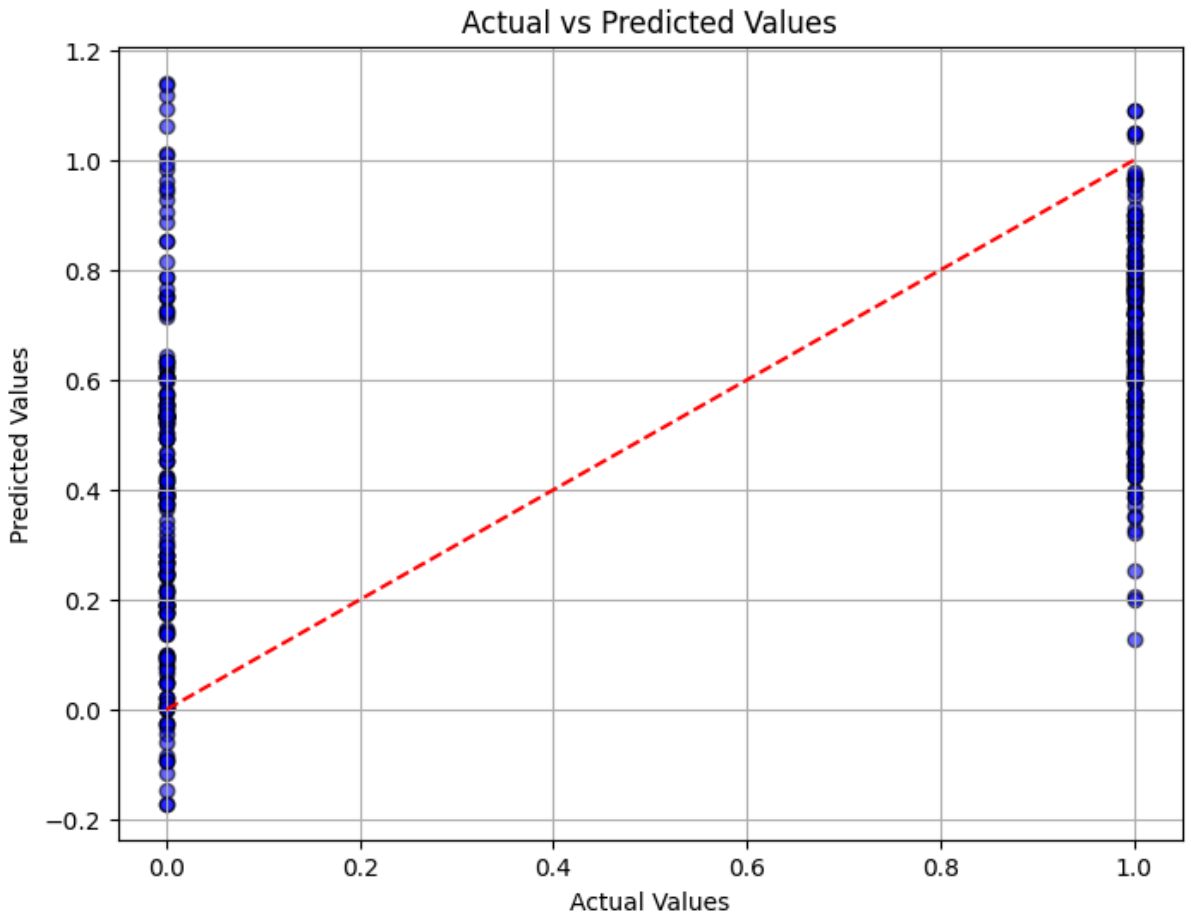
The following output was generated after running the evaluation code.

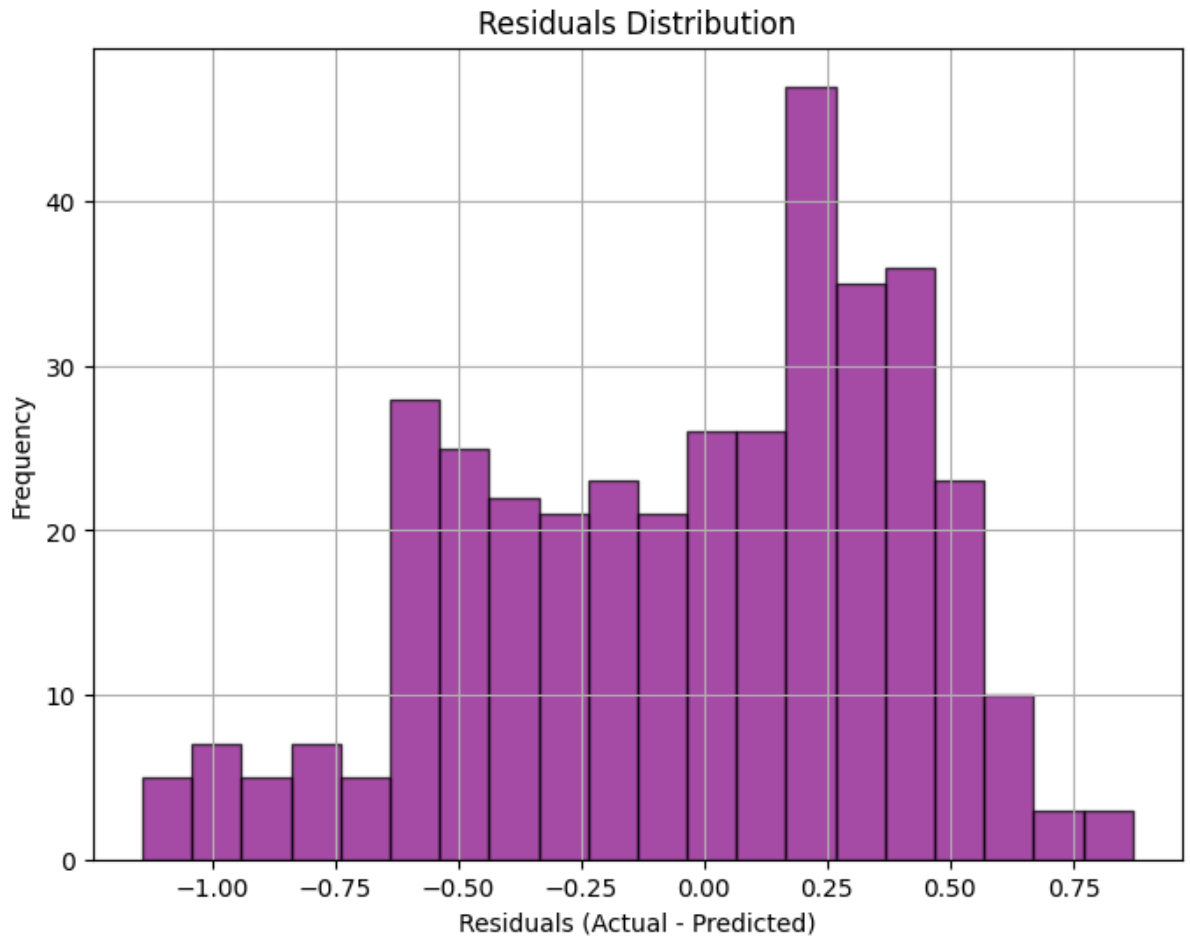
Plaintext
```
Mean Squared Error: 0.14634146341463414
R² Score: 0.407843137254902
Accuracy: 0.8536585365853658
```

**Analysis of Results:**

- **Accuracy:** The model achieved an accuracy of approximately **85.4%**. This means that the model correctly predicted the presence or absence of heart disease for about 85% of the patients in the test set. This is a strong result for a baseline model.
- **Mean Squared Error (MSE):** The MSE is 0.146. In a classification context, this reflects the average squared difference between the predicted class (0 or 1) and the actual class.
- **$R^2$ Score:** The $R^2$ score is 0.408. This metric is not standard for classification but suggests that the model explains about 41% of the variance in the target variable, which is moderate.

---


Actual vs Predicted Values
Predicted Values
Actual Values
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
0.0
0.2
0.4
0.6
0.8
1.0




# 6. Conclusion

This project successfully demonstrated the development of a machine learning model for heart disease prediction. The Logistic Regression classifier achieved a high accuracy of **85.4%**, indicating its effectiveness in distinguishing between patients with and without heart disease.

The exploratory data analysis revealed significant relationships between the target variable and features like chest pain type, maximum heart rate, and age.

**Future work** could involve exploring more complex models (e.g., Random Forest, Gradient Boosting), performing more extensive feature engineering, and tuning hyperparameters to potentially improve prediction accuracy further.