Name - Janhvi Saste
PRN - 22311825
Roll No - 282065
Year - SY B

Assignment No - 2

Analysis and Classification of Heart Disease Dataset

Problem Statement :

Heart disease remains a leading cause of mortality worldwide, making early detection and prediction critical for improving patient outcomes. The heart disease dataset provided contains 303 patient records with 15 features, including demographic attributes (e.g., Age, Sex), clinical measurements (e.g., RestBP, Chol, MaxHR), and a target variable (AHD) indicating the presence or absence of heart disease. However, real-world datasets often contain issues such as missing values, inconsistent data formats, and unscaled features, which can hinder effective analysis and modeling. The challenge is to preprocess this dataset, analyze its statistical properties, visualize feature distributions, and develop a classification model to predict heart disease accurately, thereby supporting clinical decision-making.

Objective :

The objectives of this assignment are to analyze the heart disease dataset through a structured machine learning pipeline. It begins with computing and displaying summary statistics—such as minimum, maximum, mean, range, standard deviation, variance, and percentiles—to understand the numerical properties of each feature. Feature distributions are intended to be visualized using histograms to reveal patterns and spread. The data is then cleaned by handling missing values and removing duplicates, ensuring overall data quality. Although limited to a single dataset, data integration ensures consistency throughout. Categorical variables are transformed into numerical formats using encoding techniques, making the data suitable for machine learning models. Finally, a logistic regression model is built and evaluated to predict the presence of heart disease, with performance measured through accuracy, classification metrics, and regression-based evaluation methods.

Software Used :

- Python 3.x
- Google Colab

Libraries and Packages Used :

The following Python libraries were utilized for data preprocessing and analysis:

- Pandas (For loading, manipulating, and analyzing tabular data)
- Numpy (For numerical computations and array operations)
- Seaborn (For generating plots and visualizations)
- Matplotlib (For advanced data visualization, such as heatmaps or distribution plots)
- Scikit-learn (For machine learning tasks)

Theory :

Methodology :

The assignment follows a standard machine learning pipeline for binary classification, consisting of the following steps:

- Load Data from a CSV file into a Pandas DataFrame.
- Explore Data using summary statistics.
- Clean Data by handling missing values, removing duplicates, and dropping irrelevant columns.
- Transform Data by encoding categorical features.
- Build Model using logistic regression and evaluate with accuracy and error metrics..

Main Function :

The main function used for classification is Logistic Regression from scikit-learn. Logistic Regression predicts the probability that a patient has heart disease based on input features. It uses the logistic (sigmoid) function to map predictions to a range of 0 to 1, then assigns a class (0 or 1) based on a threshold (default: 0.5). Key operations include:

- describe() for summary statistics
- isnull().sum() + fillna() for handling missing values
- drop_duplicates() to remove duplicates
- LabelEncoder to encode categorical variables
- train_test_split to split data
- LogisticRegression to train the model

Advantages :

- Clean preprocessing enhances model reliability
- Summary stats help understand data trends
- Logistic regression offers a simple, interpretable model
- Evaluation combines classification and regression metrics
- Scalable pipeline adaptable to larger or complex datasets

Disadvantages :

- Assumes Linearity: May not capture complex patterns (e.g., non-linear relationships).
- Sensitive to Outliers: Extreme values in features like cholesterol can affect performance.
- Needs Preprocessing: Requires scaling numeric features and encoding categorical ones.
- Limited for Complex Data: May underperform compared to advanced models like Random Forest for highly non-linear data..
- Convergence Warning: The logistic regression model failed to converge, indicating potential issues with feature scaling or insufficient iterations.

Applications with Example :

Logistic Regression is widely used in:

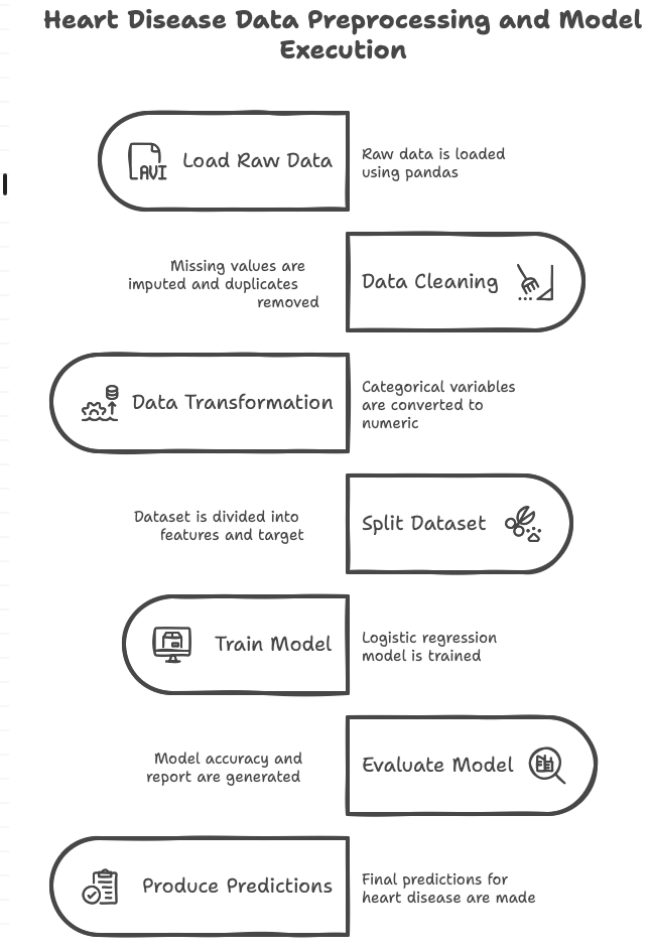
- Medical Diagnosis: Predicting diseases (e.g., heart disease, diabetes) based on patient data.
- Example: A hospital uses Logistic Regression to predict if a patient has heart disease based on age, cholesterol, and blood pressure, helping doctors prioritize tests.
- Credit Scoring: Determining if a customer will default on a loan.
- Marketing: Predicting if a customer will buy a product.
- Fraud Detection: Identifying fraudulent transactions.

Working/Algorithm :

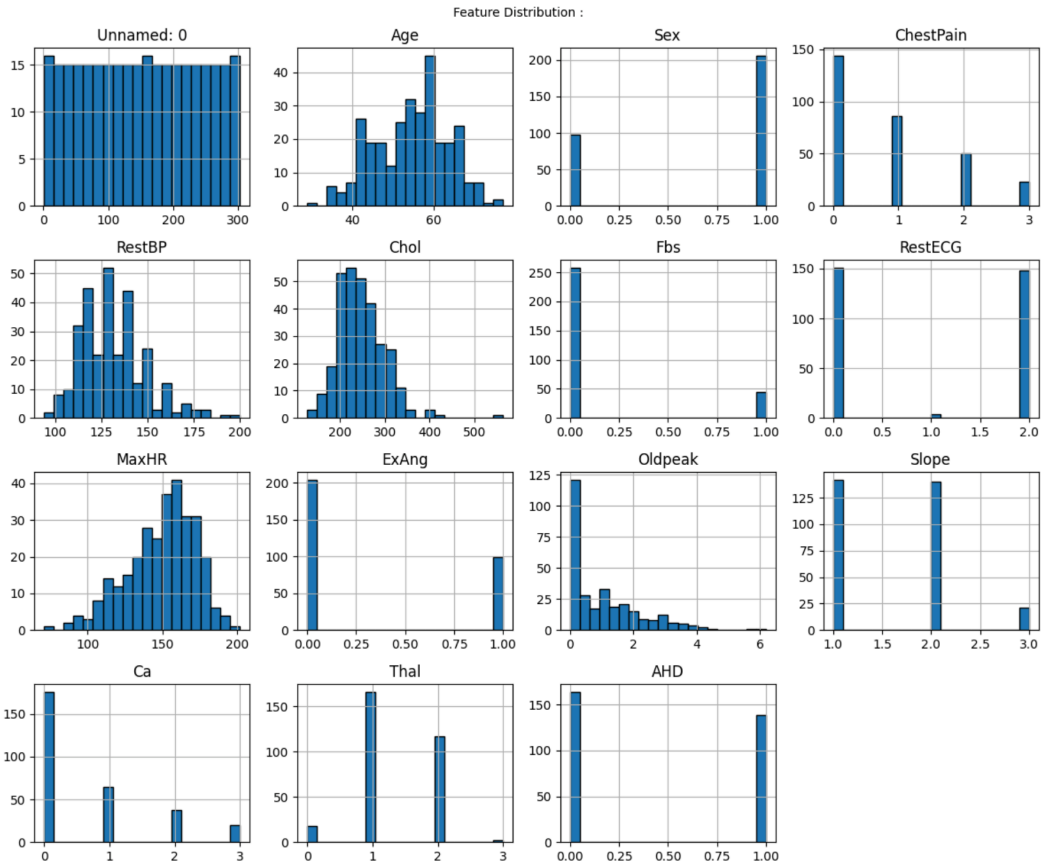
- Load heart disease dataset, clean missing values, remove duplicates, and encode categorical variables.
- Split data into features and target, create training and testing sets (80/20 split).
- Build logistic regression classifier on training data.
- Test model performance with 86.89% accuracy, generating precision, recall, and F1-scores.

In this assignment, the logistic regression model predicts whether a patient has heart disease (AHD = Yes/No) based on features like ChestPain, Thal, and MaxHR, achieving an accuracy of approximately 86.89%.

Diagram :


Heart Disease Data Preprocessing and Model Execution
Load Raw Data
Raw data is loaded using pandas
Missing values are imputed and duplicates removed
Data Cleaning
Data Transformation
Categorical variables are converted to numeric
Dataset is divided into features and target
Split Dataset
Train Model
Logistic regression model is trained
Model accuracy and report are generated
Evaluate Model
Produce Predictions
Final predictions for heart disease are made

Result :


Feature Distribution :
Unnamed: 0
Age
Sex
ChestPain
RestBP
Chol
Fbs
RestECG
MaxHR
ExAng
Oldpeak
Slope
Ca
Thal
AHD

```python
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

print("\nRegression Report:")
print(f"Mean Absolute Error (MAE): {mean_absolute_error(y_test, y_pred)}")
print(f"Mean Squared Error (MSE): {mean_squared_error(y_test, y_pred)}")
rmse = mean_squared_error(y_test, y_pred) ** 0.5
print(f"Root Mean Squared Error (RMSE): {rmse}")
print(f"R-squared Score (R²): {r2_score(y_test, y_pred)}")
```

```
Regression Report:
Mean Absolute Error (MAE): 0.13114754098360656
Mean Squared Error (MSE): 0.13114754098360656
Root Mean Squared Error (RMSE): 0.3621429841700741
R-squared Score (R²): 0.47413793103448265
```

Conclusion :

The assignment demonstrates core ML tasks on heart disease data. It includes summary stats, data cleaning, and encoding. While histograms were missing, the logistic regression model achieved 86.89% accuracy with balanced precision and recall. Despite a moderate R^2 (0.474) and no feature scaling, the pipeline is solid. Future improvements could add histograms, scaling, and advanced models like Random Forest.