


HEART-DISEASE PREDICTION

Final Report- Mini-Project
University of Lucknow (FoET)  [Github](#)

Sneha Singh (XXXXXXXXXXX111) BTech CSE (Sem-6)

Abstract

According to WHO (World Health Organisation), Cardiovascular diseases (CVDs) are the leading cause of death globally, taking an estimated 17.9 million lives each year. More than four out of five CVD deaths are due to heart attacks and strokes, and one-third of these deaths occur prematurely in people under 70 years of age. Therefore, to prevent patients from further damage, an accurate diagnosis of heart disease on time is essential. Machine learning has emerged as a powerful tool in the Healthcare Sector, offering the potential to analyze complex medical data and identify patterns that may elude traditional diagnostic methods. However, the prediction of heart disease through an ML Model is not simple. This project aims to predict the possibility of heart disease in the given patient record on basis of the model trained on the specific dataset. Secondly, we apply several evaluation metrics on the trained model to understand the impact of features on the end result.

Keywords: Heart Disease, Machine Learning, Evaluation Metrics

1. Introduction

Cardiovascular diseases (CVDs) are among the leading causes of death globally, necessitating early and accurate diagnosis. At this point, it is essential

to prevent disease and effectively utilize medical resources. Due to recent technological advancements, the field of medical sciences has seen a remarkable improvement over time. Talking especially about Machine Learning Domain with tons of raw electronic medical data gathered from low-cost wearable devices allows efficient heart disease diagnosis.

Machine Learning basically means capability of a machine to imitate intelligent human behavior without the need to be programmed. ML model starts with data — numbers, photos, or text, time series data from sensors, or reports as input which is then supplied to the chosen model to undergo mathematical optimizations to find patterns and provide desired predictions (e.g., disease, no disease, neutral).

While training a model large amount of data is needed to avoid overfitting but also keeping track of features so that our model does not suffer from Curse of Dimensionality.

The main features of this project are listed as follows:

- The Project uses a standard dataset from a UCI ML repository , where data is collected from Cleveland.
- ML classification models such as Logistic Regression (LR) is used on the datasets to identify the suitable model with higher accuracy.
- Evaluation Metrics such as Accuracy Score, Precision Score, Recall, F1 Score and Confusion Metrics are used to observe the impact of features on model and their performance.

- In the interest of reproducible science, the codebase for this article has been made accessible on GitHub.

2. Methodology

This section of the project focus on the dataset characteristics, Pre-Processing done on the data the project workflow. Figure 1 illustrate the workflow of the proposed methodology for the heart disease prediction

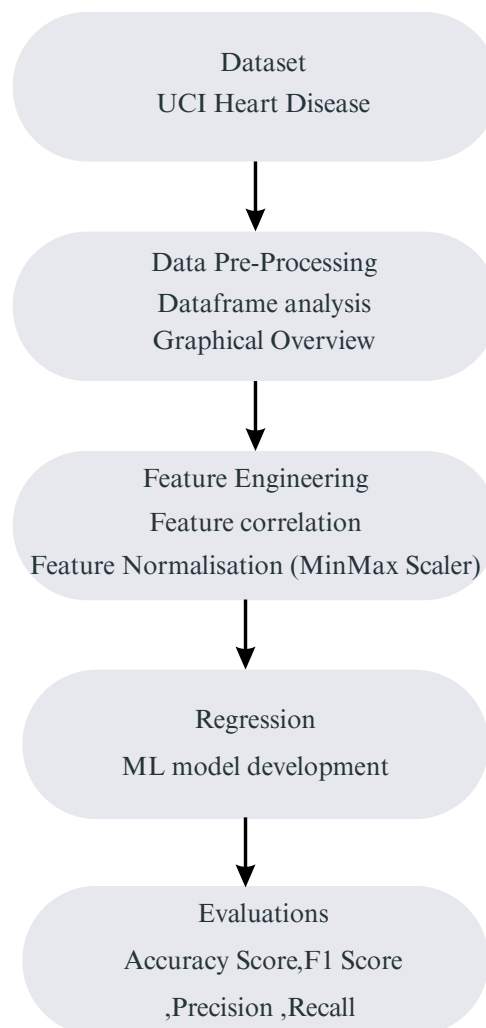


Figure 1: Flowchart of the proposed methodology describing each step for heart disease prediction

2.1 Dataset

In this Project, a dataset named UCI-Heart-Disease is used to predict the presence of heart disease based on diagnostic features. The Heart Disease UCI Dataset is a widely used benchmark dataset in healthcare analytics and machine learning. The dataset is sourced from the UCI Machine Learning Repository and is frequently used for classification tasks in medical AI research. (<https://archive.ics.uci.edu/>)

The Heart Disease UCI Dataset contains 14 key clinical and demographic features and 303 patient records .

Feature	Description
age	Age in years
sex	1 = male, 0 = female
cp	Chest pain type (1-4)
trestbps	Resting blood pressure (mmHg)
chol	Serum cholesterol (mg/dl)
fbs	Fasting blood sugar > 120 mg/dl (1 = yes, 0 = no)
restecg	Resting electrocardiographic results
thalach	Maximum heart rate achieved
exang	Exercise-induced angina (1 = yes, 0 = no)
oldpeak	ST depression induced by exercise
slope	Slope of peak exercise ST segment
ca	Number of major vessels colored by fluoroscopy
thal	Thalassemia (3 = normal, 6 = fixed defect, 7 = reversible defect)

Table 1: Description of features of UCI dataset

The datasets contained some main medical features like 'age', 'chest pain type', 'exercise-induced angina', 'blood pressure', 'cholesterol', 'ST depression' etc. which are closely related to the occurrence of disease and provides a great flexibility for heart disease analysis. Target Column represents diagnosis of heart disease 0 = No heart disease 1-4 = Presence of heart disease (varying severity).

2.2 Pre-Processing

Data pre-processing is one of the important phases of ML life cycle as it makes data analysis easy and increases the accuracy and speed of the ML algorithms. The Heart Disease UCI Dataset contains 14 key clinical and demographic features and 303 patient records. We check for missing values and columns datatype. In case missing values are found Imputation techniques can be used for handling the missing data, however, their usage in medical field is limited. Most of the times, researchers do not consider the observations with missing values and drop the incomplete cases intentionally, since the traditional data imputation methods are not sufficient to capture the missing data complexities in health care applications.

Furthermore, we check for class imbalance in target value to so that model did not achieve high accuracy by simply always predicting the majority class. Figure 2 for the reference which is a countplot for target value.

- Example: If 95% of patients are healthy (class 0), a model that always predicts "healthy" achieves 95% accuracy but is useless.

Models like Linear Regression, Logistic Regression, LDA assumes normality in the data so its crucial to check for Gaussian distribution in data to do that KDE Plots are one good way. Plotting KDE Plots for numerical columns such as 'chol', 'thalach', 'age' etc helps in data understanding.

Figure 3 which display the KDE Plots for the numerical columns.

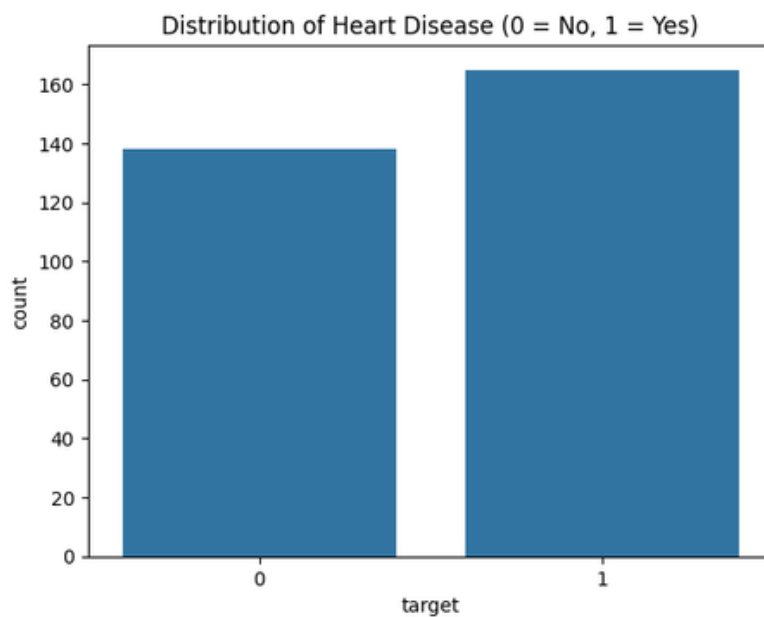
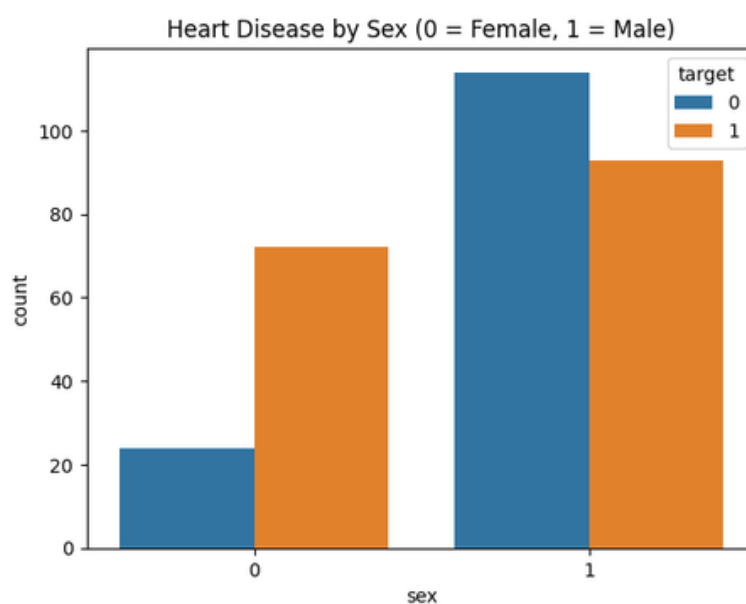


Figure 2: Target class distribution



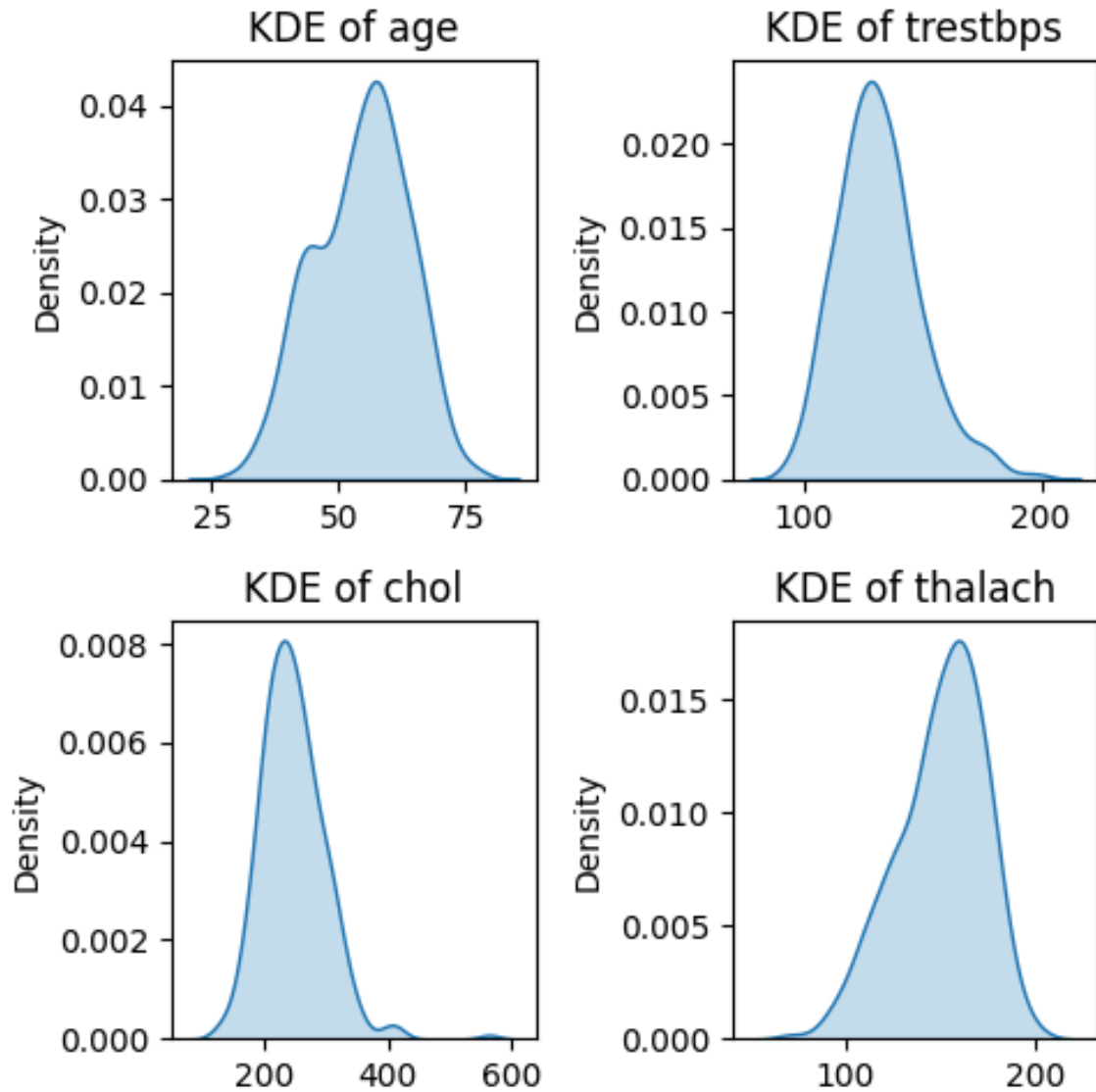


Figure 3: KDE Plots for chol ,thalach ,age & trestbps

Similarly for outlier detection and data variability we can plot box plots which help visualise flag outliers (data points beyond $1.5 \times \text{IQR}$ from the quartiles).

Figure 4 represents box plots for age and thalach against heart disease

Figure 5 represents box plots for chol and trestbps against heart disease

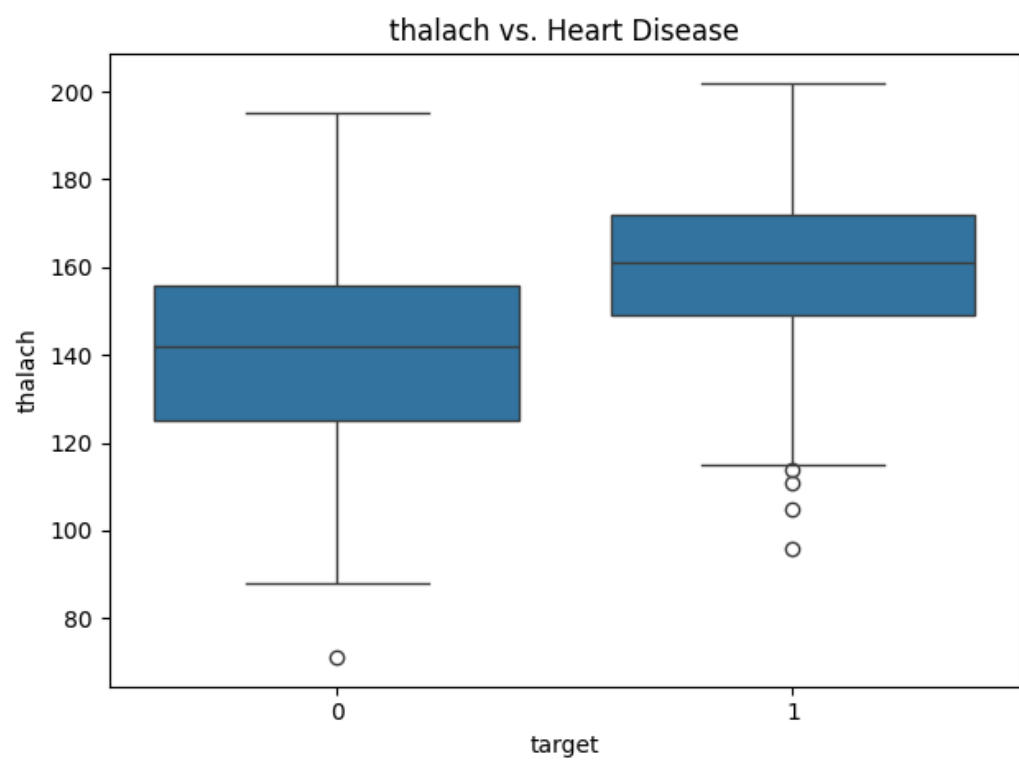
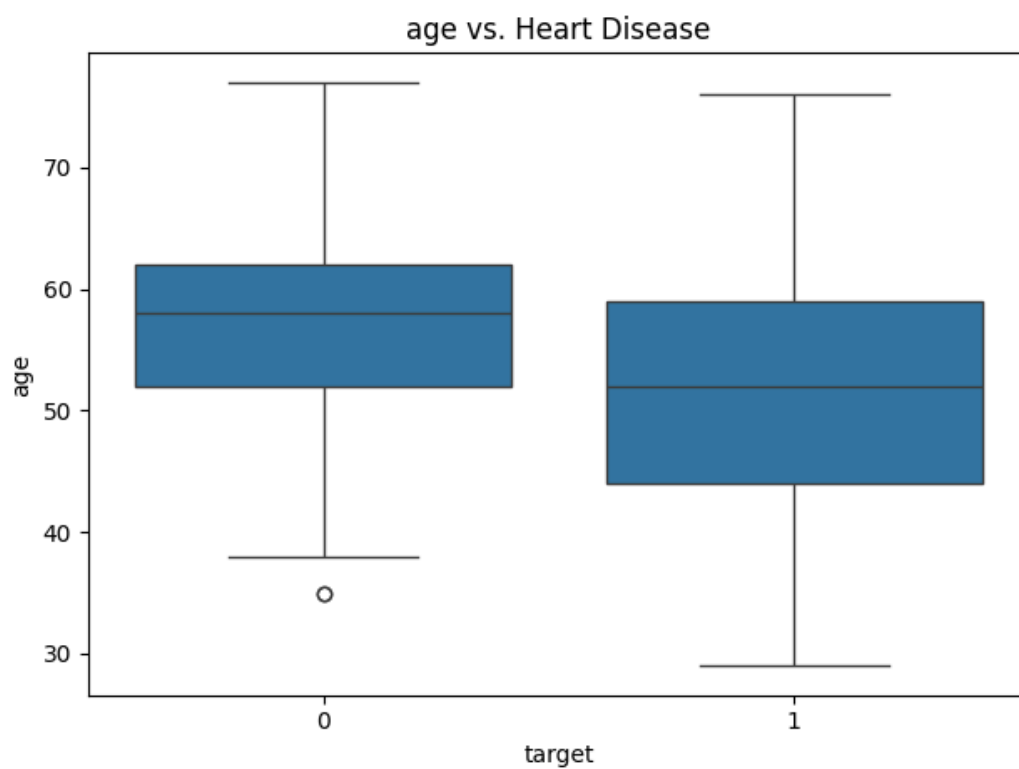


Figure 4: Box Plots for age ,thalach

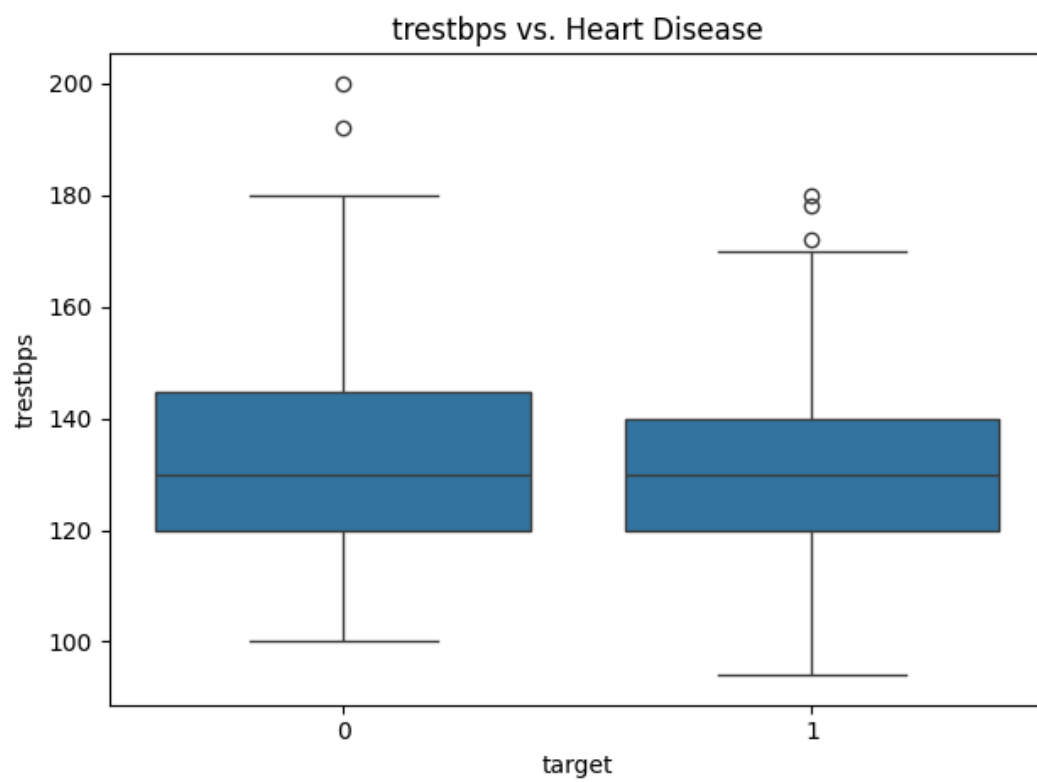
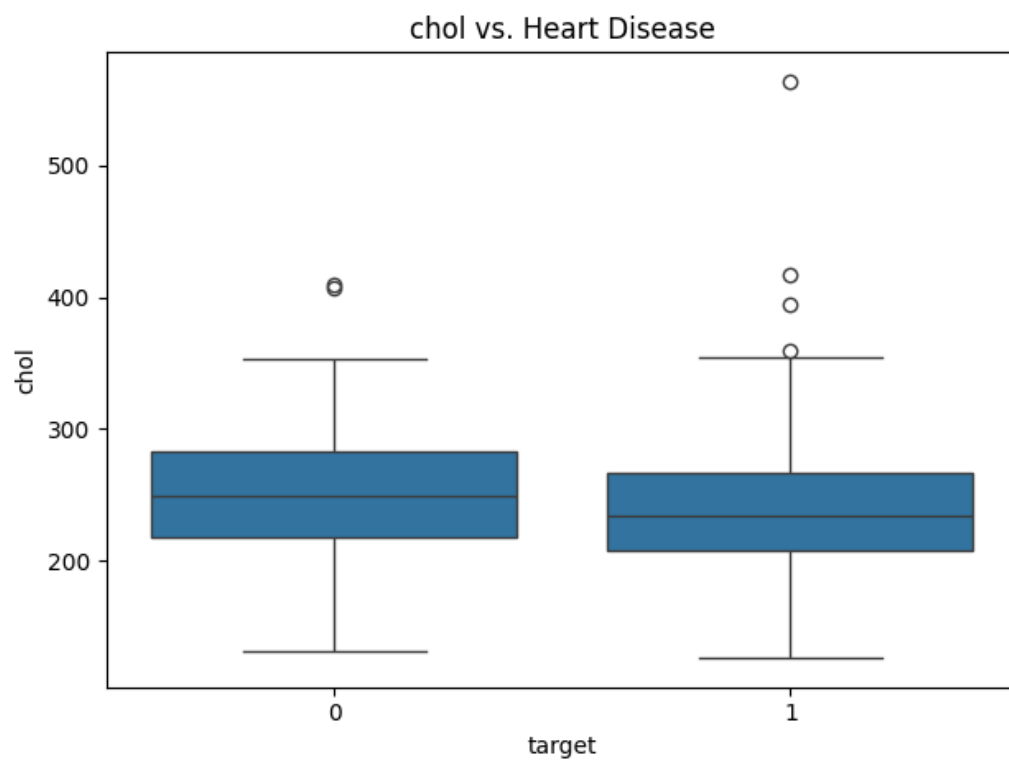


Figure 5: Box Plots for chol ,trestbps

2.3 Feature Engineering

Feature Engineering is a crucial step in the machine learning pipeline. Feature engineering involves 4 main steps :

- Feature Transformation
- Feature Construction
- Feature Selection
- Feature Extraction

Machine learning algorithms learn from the patterns in your data. The features you provide directly influence what patterns the algorithm can discover. Well-engineered features can:

- Improve model accuracy: By highlighting important relationships and structures in the data that might be hidden in the original features.
- Speed up training: Models can converge faster when provided with more relevant and informative features.
- Enhance model interpretability: Sometimes, newly created features can offer more intuitive insights into the underlying phenomena.

The project approach is to implement Feature Transformation specifically Normalisation.

2.3.1 Normalisation

The primary goal of normalization (or scaling) is to ensure that all features contribute equally to the model training process, preventing features with larger values from dominating those with smaller values .

1. Min-Max Scaling (Normalization):

This method scales the features to a specific range, typically between 0 and 1.

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

2.3.2 Feature Correlation Analysis

Feature correlation is a method which helps in understanding the underlying relationships between various data features present in a dataset. Feature Correlation help in identifying highly correlated features can help in feature selection as we might keep one and discard another, it also provides insights into the relationships between different attributes of your data. Computed correlation values between different medical features are shown in Figure 6.

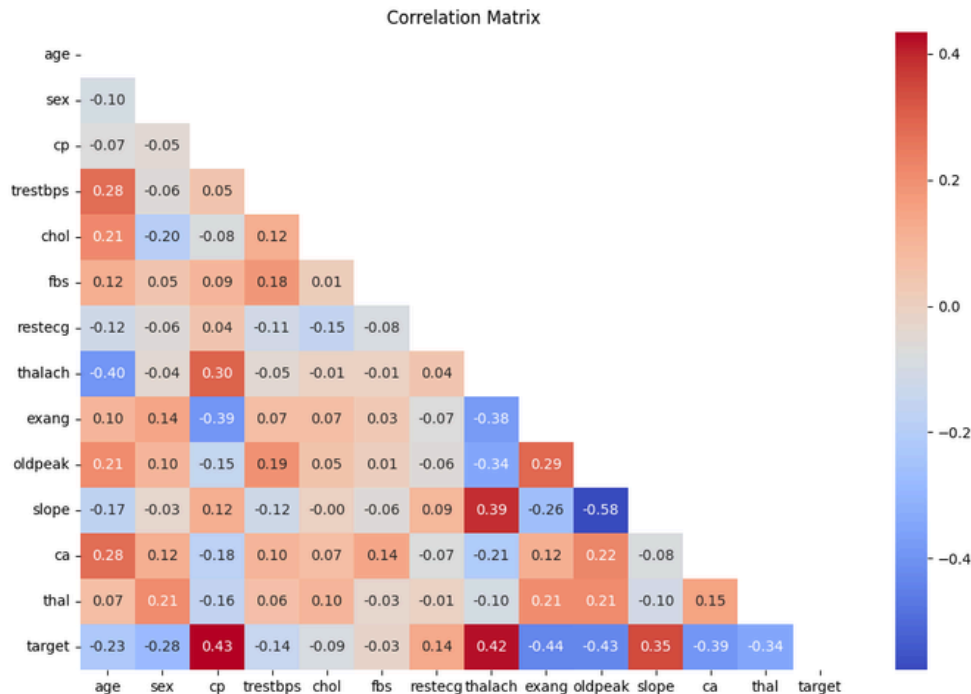


Figure 6: The correlation values for each medical feature

As we can observe from Figure 6, of the 13 features of UCI heart disease dataset 4 features are having a positive correlation with the target feature *i.e* “target” and 9 features have a negative correlation with the target feature . Features such as “cp” , “restecg” ,”thalach” and ”slope” are having values 0.43 , 0.14 , 0.42 and 0.35 when correlated with target showing a significant correlation for all three features except restecg which is negligible.

Features such as “cp” ,”thalach” has high positive correlation of 0.43 which indicates that certain chest pain type and increased heart rate strongly relate to heart condition .

Features like “fbs” has a very weak correlation of -0.03 indicating its lower or near - zero relevance for this model

Correlation between “sex” and “target” is -0.028 suggesting that males are at higher risk then females. Like wise correlation between “age” and “thalach” is -0.40 suggesting that younger patients have higher max heart rate which aligns with the medical concept.

Features like “oldpeak” and “slope” have very high negative correlation and affect model due to multicollinearity .

As per medical research findings, with aging, major changes can be observed in the heart and blood vessels. For example, the heartbeat rate is not as fast during any physical activity as it could when you are younger which can be indicated by “exang” and its correlation with target value .

3. Evaluation Matrices / Scores

For this project we have used some popular evaluation metrics *i.e* , Accuracy ,Confusion matrix, Precision, Recall and F1-score to evaluate the performance of our ML classification Model.The accuracy metric in machine learning classification which answers a simple question that is "How often is our model getting the predictions right?"

Accuracy Score:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

The confusion matrix gives us a much clearer picture of our model's performance than just a simple percentage of correct predictions (accuracy). For a binary classification task involving a positive and a negative class, the confusion matrix is typically structured as a 2x2 table:

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

Figure 7: Confusion matrix

this 2x2 matrix is described as (1) True Positive (T P) test result that correctly classify the presence of heart disease in patient, (2) True Negative (T N) test result that correctly classify the absence of heart disease in patient, (3) False Negative (F N) test result that wrongly classify that a particular patient does not have heart disease and (4) False Positive (F P) test result which wrongly classify that a particular patient has heart disease.

In the field of Healthcare, FN category is most harmful predictions because this can lead to death to the patient.

For the given dataset different calculations for evaluation report will be:

$$Accuracy = (T P + T N) / (T P + F P + F N + T N)$$

$$Precision = T P / (T P + F P)$$

$$Recall = T P / (T P + F N)$$

F1-score is the harmonic mean of Precision and Recall.

$$F1 - Score = 2(Precision \times Recall) / (Precision + Recall)$$

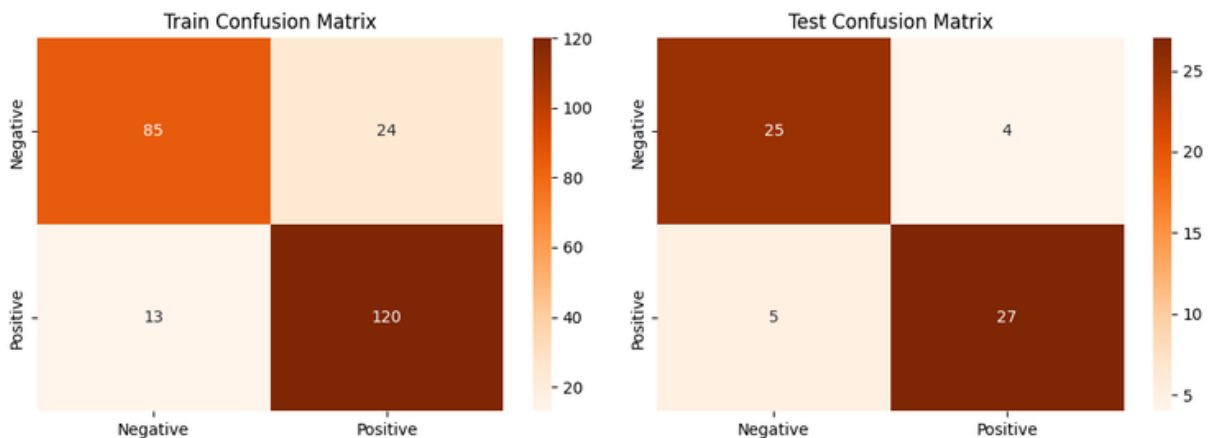


Figure 8: Confusion matrix Plot

Train Classification Report:				
	precision	recall	f1-score	support
0	0.867347	0.779817	0.821256	109
1	0.833333	0.902256	0.866426	133
accuracy	0.847107	0.847107	0.847107	0.847107
macro avg	0.85034	0.841036	0.843841	242
weighted avg	0.848654	0.847107	0.846081	242

Test Classification Report:				
	precision	recall	f1-score	support
0	0.833333	0.862069	0.847458	29
1	0.870968	0.84375	0.857143	32
accuracy	0.852459	0.852459	0.852459	0.852459
macro avg	0.852151	0.852909	0.8523	61
weighted avg	0.853076	0.852459	0.852538	61

Figure 9: Classification Report for train and test data

Score	
precision_train	0.848654
precision_test	0.853076
recall_train	0.847107
recall_test	0.852459
acc_train	0.847107
acc_test	0.852459
F1_train	0.846081
F1_test	0.852538

Figure 10: Evaluation Scores for the model (considered weighted avg)

4. Results

In this section, we will discuss the performance of the selected classification models from different perspectives.

The logistic regression model demonstrate consistent and reliable performance across both training and test datasets, indicating its suitability for predicting heart disease in the UCI dataset.

Key takeaways include:

- Cross-Validation Score: 80% (mean accuracy), confirming generalizability.
- Test Performance:
 - Precision: 0.853 (minimized false positives).
 - Recall: 0.852 (captured 85.2% of true positives).
 - F1-Score: 0.853 (balanced precision-recall tradeoff)
 - Accuracy: 0.852 (aligned with cross-validation).

Talking about models relevance for clinical use it can flag high-risk patients, though further tuning is requiered to reduce FN.

For Benchmarking , compared to baseline models such as Random Forest (RF), Naive Bayes (NB), Decision Trees (DT) logistic regression provides interpretabilty on cost of accuracy.

4.1 Future Works

Future work could integrate SHAP values for explainability or test ensemble methods for marginal gains.

Integrating ensemble methods will help gain accuracy and better performance.