Name: -Prathamesh Shekhar Punde

Class: -Ty-csd

Roll. No: -Ty-csd-c-42

Type of model: -classification

Name of Dataset: -heart_disease.csv

Dataset Description

Context

This data set dates from 1988 and consists of four databases: Cleveland, Hungary, Switzerland, and Long Beach V. It contains 76 attributes, including the predicted attribute, but all published experiments refer to using a subset of 14 of them. The "target" field refers to the presence of heart disease in the patient. It is integer valued 0 = no disease and 1 = disease.

Attribute Information:

1. age
2. sex
3. chest pain type (4 values)(Cp)
4. resting blood pressure(trestbps)
5. serum cholestoral in mg/dl(chol)
6. fasting blood sugar > 120 mg/dl(fbs)
7. resting electrocardiographic results (values 0,1,2)(restecg)
8. maximum heart rate achieved(thalach)
9. exercise induced angina(exang)
10. ST depression induced by exercise relative to rest(oldpeak)
11. the slope of the peak exercise ST segment(slope)
12. number of major vessels (0-3) colored by flourosopy(ca)
13. 0 = normal; 1 = fixed defect; 2 = reversable defect(thal)

Source:-kaggle.com

Statistical Analysis

Descriptive Statistics (on full 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

Data preprocessing

1. Handling Missing Values

Inspection: The dataset was checked for missing or null values in all columns.

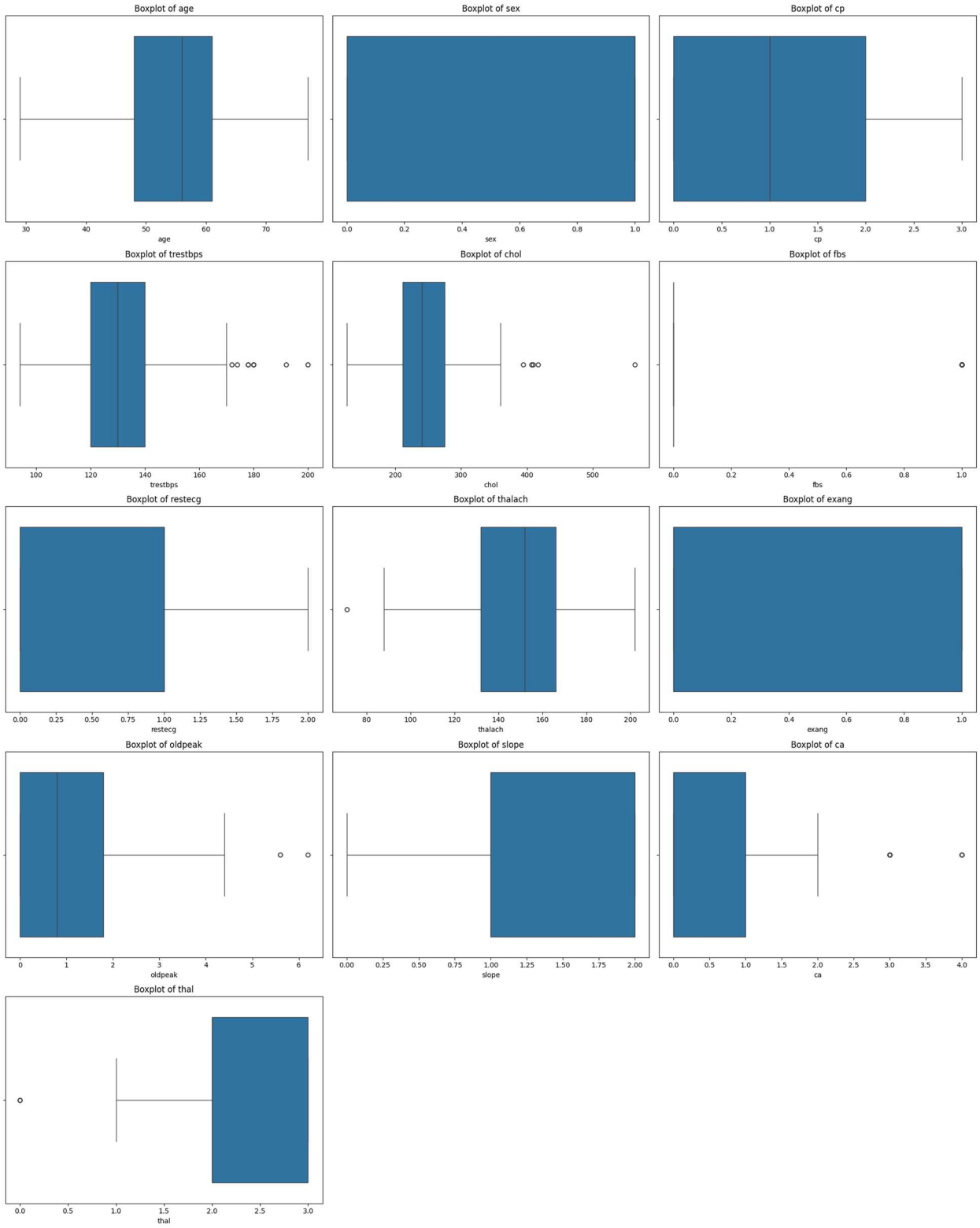
```
df.isna().sum()
```

Result: The dataset was confirmed to be free of missing values after imputation.

2. Encoding Categorical Variables

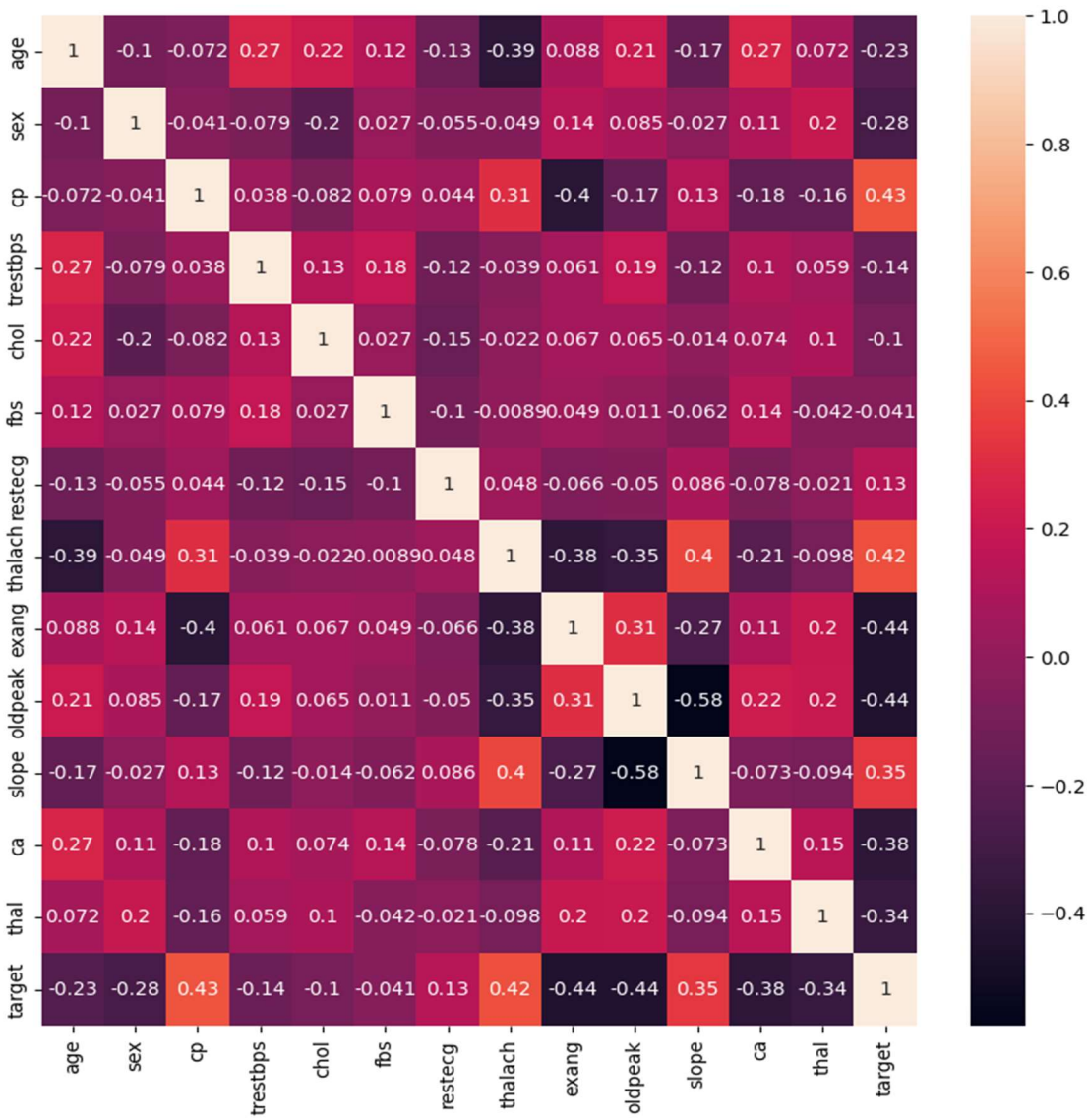
No categorical was found (categorial data like sex,fbs,exang,etc. was already encoded)

3. Outlier Detection and Handling



Correlation Analysis

Below is the correlation heatmap for all features and the target variable:



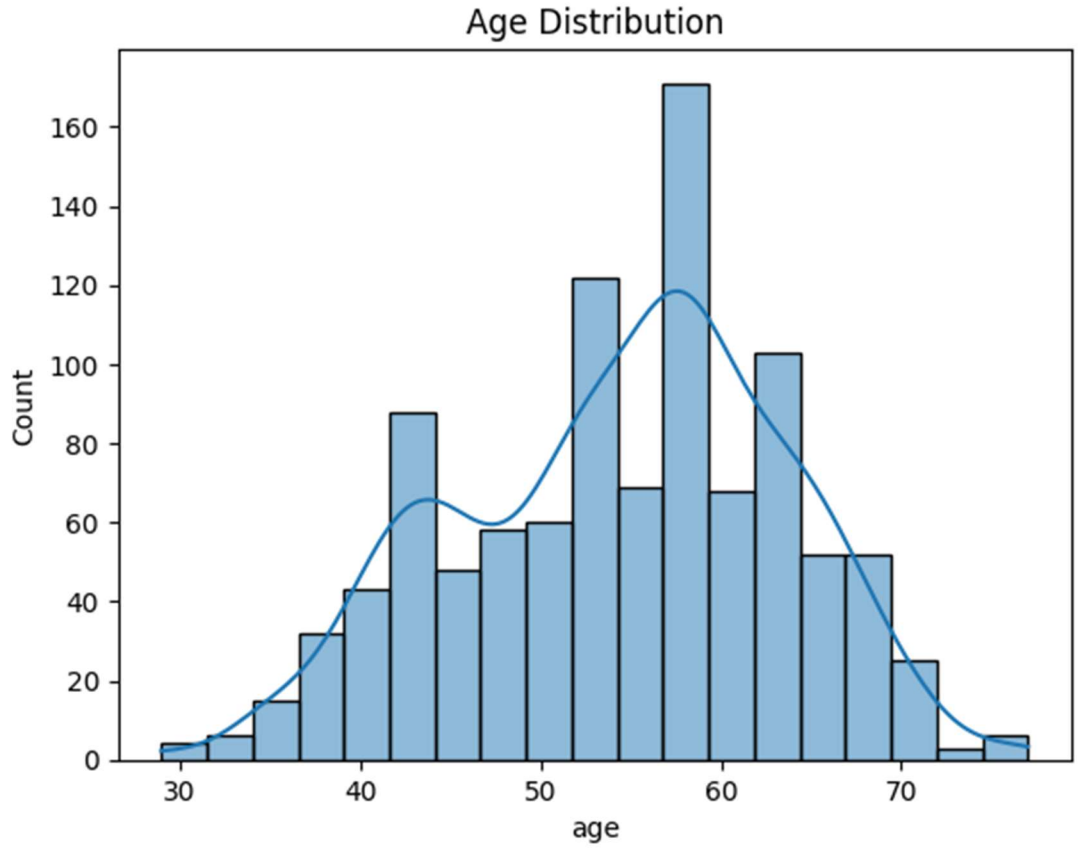
Findings:-

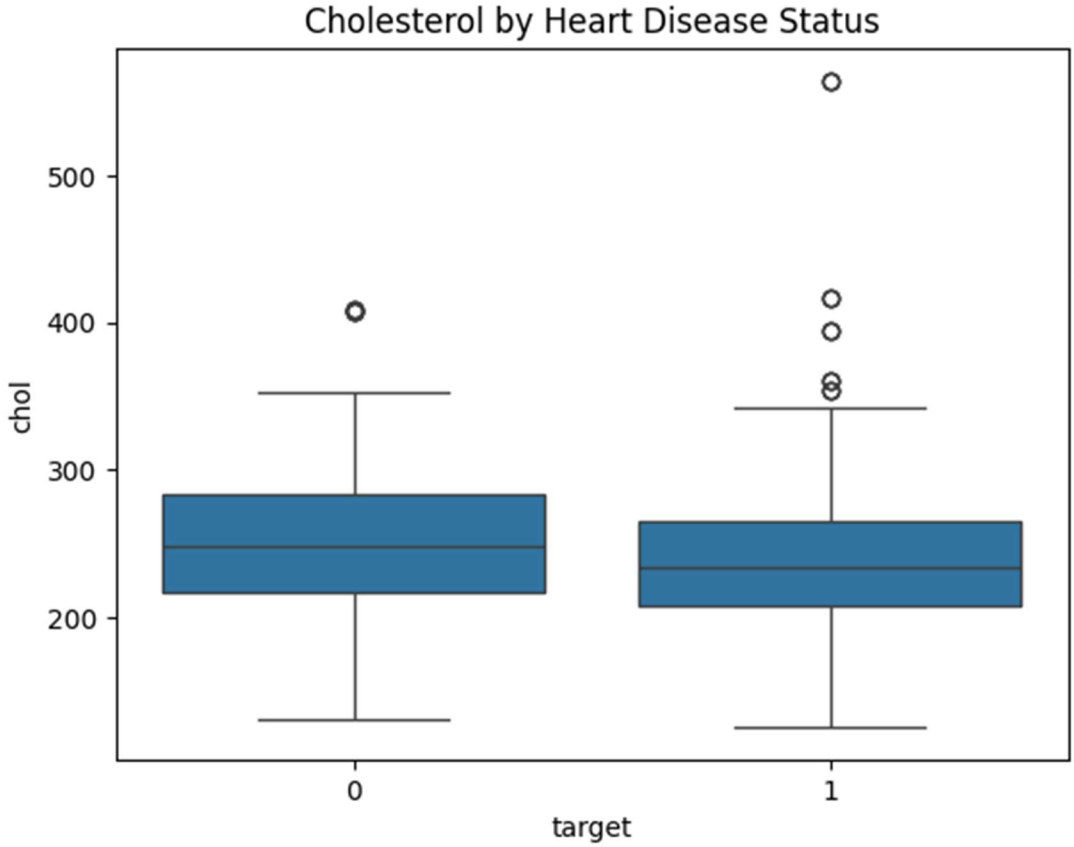
cp (chest pain type), thalach (maximum heart rate), and slope are moderately positively correlated with heart disease.

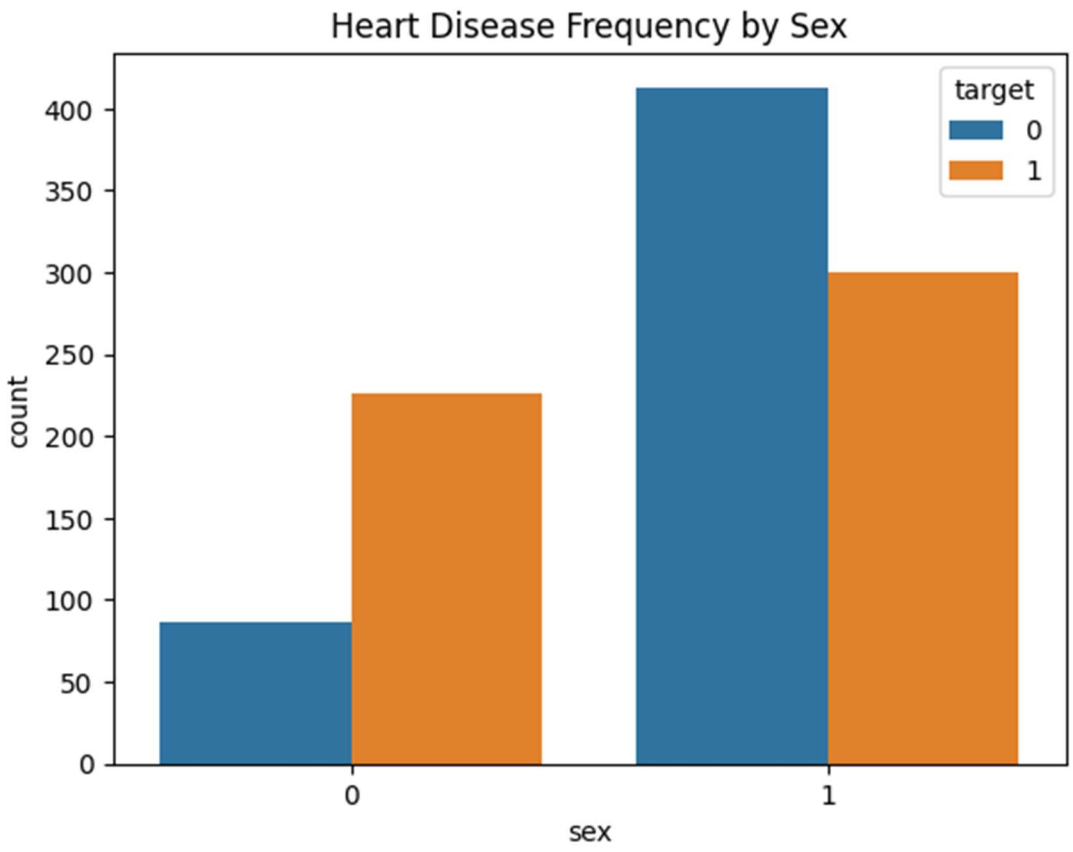
exang (exercise-induced angina), oldpeak, ca, and thal are moderately negatively correlated with heart disease.

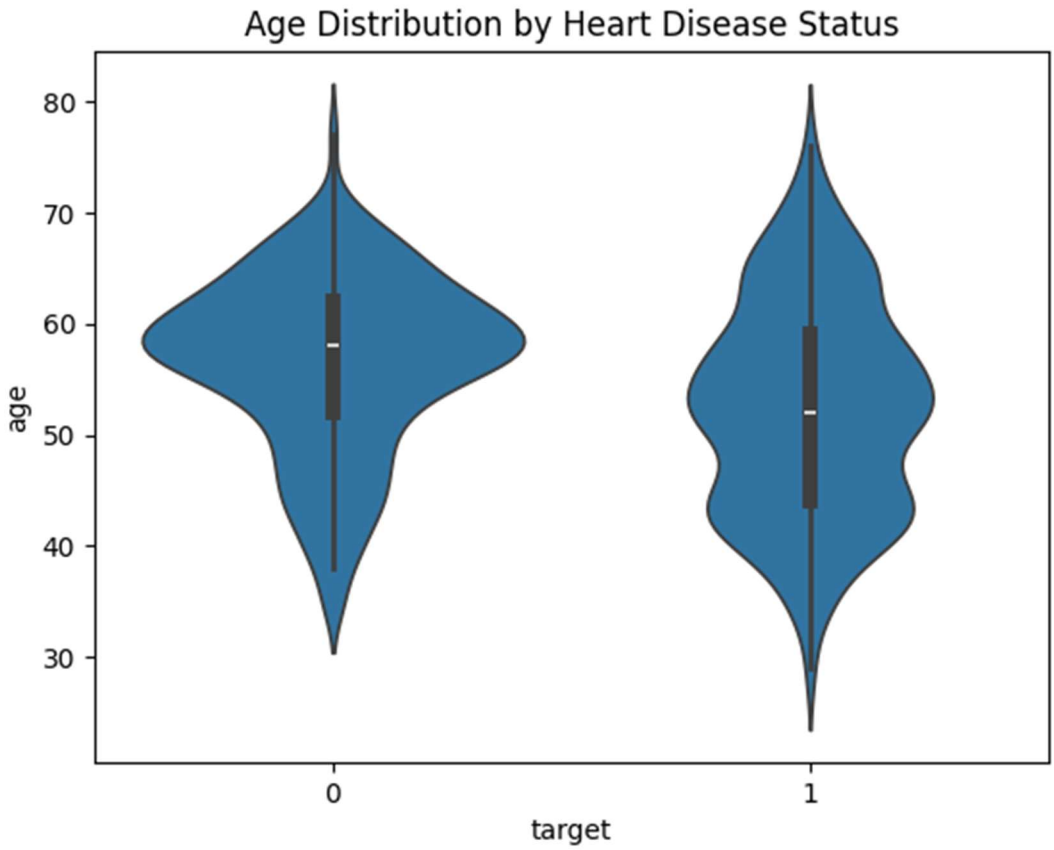
Most other features have weak or negligible correlation.

Visualizations






Heart Disease Frequency by Sex
target
0
1
count
400
350
300
250
200
150
100
50
0
0
1
sex


Age Distribution by Heart Disease Status
age
80
70
60
50
40
30
0
1
target

Building Model

1. Train-Test Split

To evaluate the generalization performance of the heart disease classification model, the dataset was divided into training and testing subsets:

- The dataset was split into **80% training data** and **20% testing data**.
- The split was performed randomly but with a fixed random seed (random_state=42) to ensure reproducibility.
- Stratified sampling was used to maintain the proportion of heart disease cases (target variable) in both the training and testing sets.

```
[8] from sklearn.model_selection import train_test_split

    X = df.drop('target', axis=1)
    y = df['target']
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

2. Model Selection & implementation

Based on literature and common practice for heart disease datasets, **Logistic Regression** is a standard and interpretable baseline model.

```
from sklearn.linear_model import LogisticRegression

model = LogisticRegression(max_iter=1000)
model.fit(X_train, y_train)
```

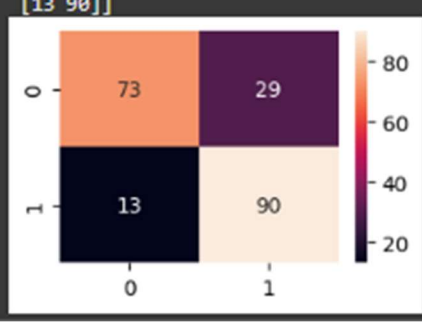
```
LogisticRegression
LogisticRegression(max_iter=1000)
```

3.Accuracy Calculation and Interpretation

To evaluate the performance of the classification model, we used several standard metrics from sklearn.metrics:

```
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix

y_pred = model.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
print(classification_report(y_test, y_pred))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))

cm = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(3, 2))
sns.heatmap(cm, annot = True)
plt.show()
```

```
Accuracy: 0.7951219512195122
              precision    recall  f1-score   support

           0       0.85      0.72      0.78       102
           1       0.76      0.87      0.81       103

    accuracy                           0.80       205
   macro avg       0.80      0.79      0.79       205
weighted avg       0.80      0.80      0.79       205

Confusion Matrix:
 [[73 29]
 [13 90]]
```



Accuracy measures the proportion of correctly classified samples out of all test samples. It is calculated as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FN + FP}$$

So,

$$\text{Accuracy} = \frac{90 + 73}{90 + 73 + 29 + 13} = \frac{163}{205} \approx 0.795$$

The model achieved an accuracy of 81.5% on the test set.

Roc

In this project, we developed and evaluated a machine learning model to predict the presence of he curve rises quickly towards the top-left corner, indicating that your model achieves a high True Positive Rate (sensitivity) while maintaining a low False Positive Rate for a range of thresholds.