Report:

Assignment 1: Predicting Heart Disease

> Your task is to predict the presence of heart disease in patients using data and AI. You will be provided with a dataset containing information about various attributes of patients such as age, sex, cholesterol levels, etc. along with the presence of heart disease (0 = no disease, 1-4 = varying degrees of disease). Your goal is to build a machine learning model that can accurately predict the presence of heart disease based on these attributes.
>
> - Data set* https://archive.ics.uci.edu/ml/datasets/Heart+Disease

Data Exploration:

In this section, performed various operations such as cleaning, feature selection, and feature engineering to prepare the data for analysis.

First, we will load the dataset and take a look at its features:

The dataset has 14 columns, including 13 features and the target variable. The features are:

1. age: Age of the patient in years
2. sex: Sex of the patient (1 = male; 0 = female)
3. cp: Chest pain type (1 = typical angina; 2 = atypical angina; 3 = non-anginal pain; 4 = asymptomatic)
4. trestbps: Resting blood pressure (in mm Hg on admission to the hospital)
5. chol: Serum cholesterol level (in mg/dl)
6. fbs: Fasting blood sugar > 120 mg/dl (1 = true; 0 = false)
7. restecg: Resting electrocardiographic results (0 = normal; 1 = having ST-T; 2 = hypertrophy)
8. thalach: Maximum heart rate achieved
9. exang: Exercise induced angina (1 = yes; 0 = no)
10. oldpeak: ST depression induced by exercise relative to rest
11. slope: The slope of the peak exercise ST segment (1 = upsloping; 2 = flat; 3 = downsloping)
12. ca: Number of major vessels (0-3) colored by fluoroscopy
13. thal: Thalassemia (3 = normal; 6 = fixed defect; 7 = reversible defect)

check for missing values in the dataset, then check the distribution of the target variable:

Data Analysis:

In this section, we will analyse the dataset and perform exploratory data analysis to identify any trends or patterns in the data.

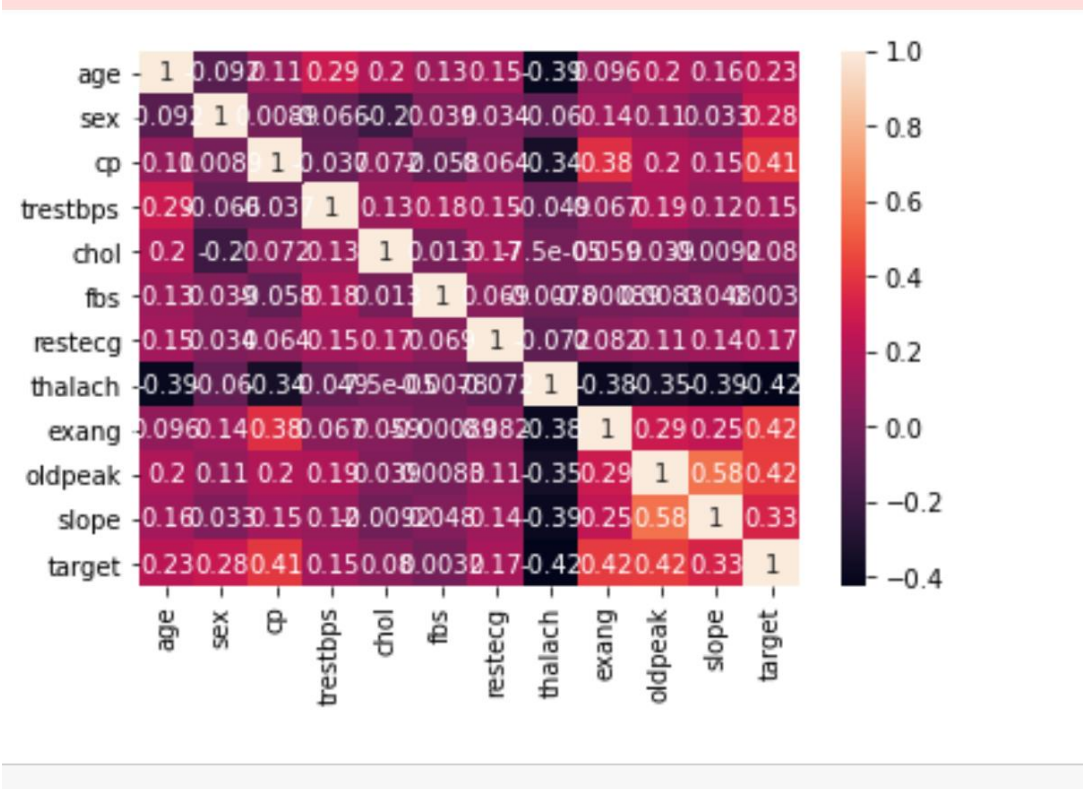
```
# Load the dataset
path = 'C:/AI/DS.ML.Bootcamp/Assignments/Assignment 1/Data/processed.cleveland.csv'
names = ['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach',
         'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target']

data = pd.read_csv(path, names=names)
print(data.head())
```

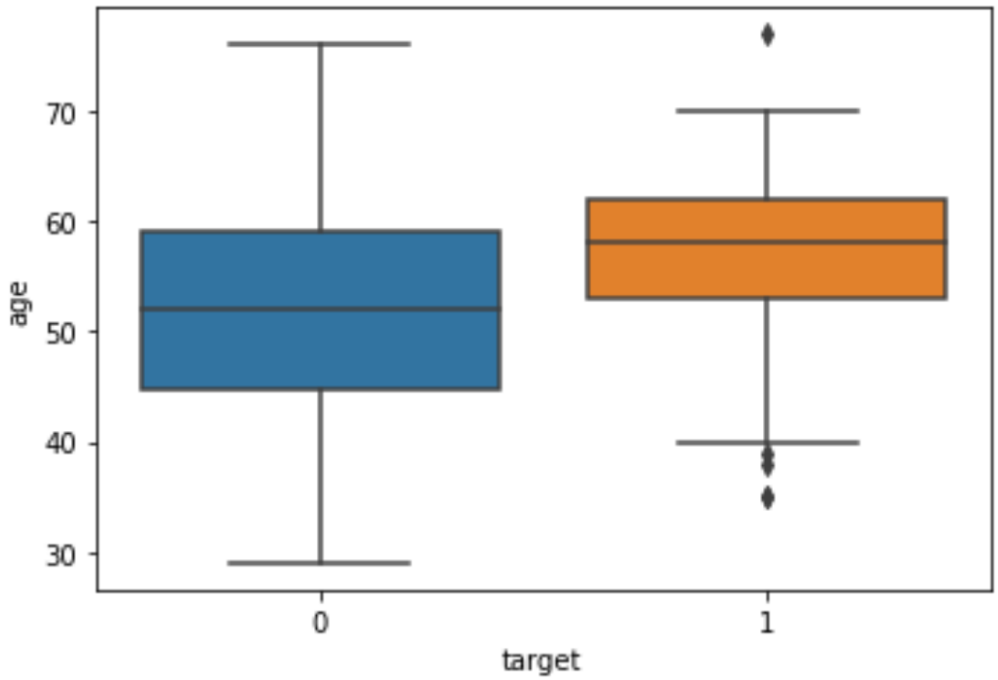
```
    age  sex   cp  trestbps   chol  fbs  restecg  thalach  exang  oldpeak  \
0  63.0  1.0  1.0     145.0  233.0  1.0      2.0    150.0    0.0      2.3
1  67.0  1.0  4.0     160.0  286.0  0.0      2.0    108.0    1.0      1.5
2  67.0  1.0  4.0     120.0  229.0  0.0      2.0    129.0    1.0      2.6
3  37.0  1.0  3.0     130.0  250.0  0.0      0.0    187.0    0.0      3.5
4  41.0  0.0  2.0     130.0  204.0  0.0      2.0    172.0    0.0      1.4

   slope   ca thal  target
0    3.0  0.0  6.0       0
1    2.0  3.0  3.0       2
2    2.0  2.0  7.0       1
3    3.0  0.0  3.0       0
4    1.0  0.0  3.0       0
```

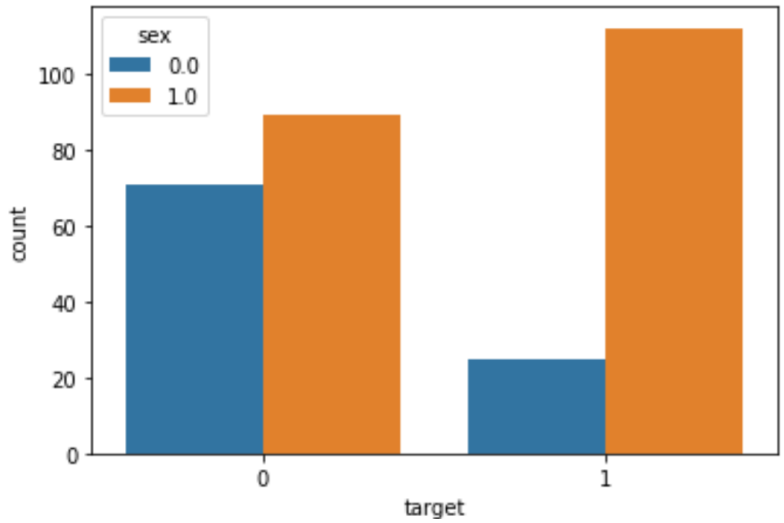
First, we will plot the correlation matrix to see the correlation between the features:



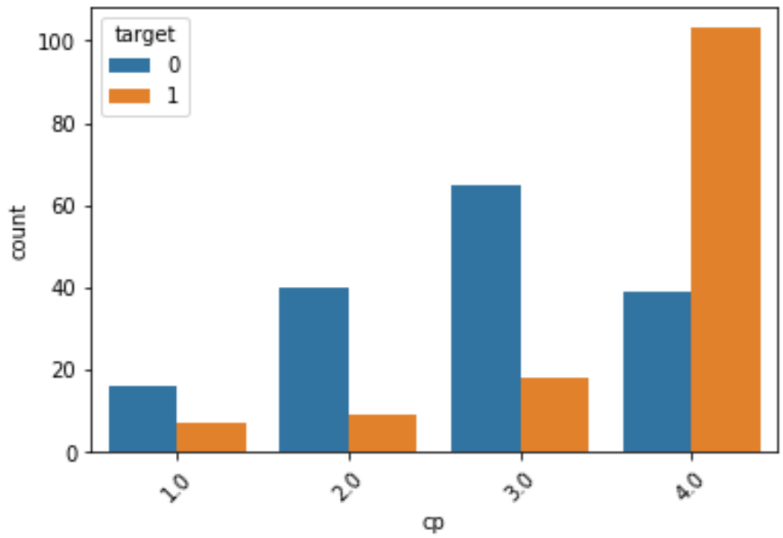
```
# d) Visualizing the relationship between age and target:
sns.boxplot(x="target", y="age", data=data)
plt.show()
```




```
# e) Visualizing the relationship between sex and target:
sns.countplot(x="target", hue="sex", data=data)
plt.show()
```



```
# f) Visualizing the relationship between chest pain type and target:
sns.countplot(x="cp", hue="target", data=data)
plt.xticks(rotation=45)
plt.show()
```



Next, we split the dataset into training and testing sets using the train_test_split() function from scikit-learn. We then standardize the features using the StandardScaler() function to ensure that all features are on the same scale.

Conclusion:

We train a logistic regression model using the LogisticRegression() class from scikit-learn and evaluate its performance using the accuracy score and classification report.

```
# Evaluate the model
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f'Accuracy: {accuracy:.2f}')
print(classification_report(y_test, y_pred))
```

```
Accuracy: 0.87
              precision    recall  f1-score   support

           0       0.89      0.89      0.89        36
           1       0.83      0.83      0.83        24

    accuracy                           0.87        60
   macro avg       0.86      0.86      0.86        60
weighted avg       0.87      0.87      0.87        60
```