A Report on developing a machine learning model to diagnose medical conditions based on patient symptoms, test results, and medical history.

1. Introduction:

1.1 Background:

Medical diagnosis plays a pivotal role in effective healthcare delivery. The conventional diagnostic process relies heavily on the expertise of medical professionals, but it can be challenging to interpret complex patterns in large datasets. The integration of machine learning presents a promising solution to augment diagnostic capabilities, leading to better decision-making and patient care.

1.2 Objectives:

The primary objective of this report is to detail the development of a machine learning model capable of diagnosing medical conditions using patient symptoms, test results, and medical history. This model aims to enhance diagnostic accuracy and assist medical professionals in providing personalized and timely treatments.

2. Methodology:

2.1 Data Collection:

Link to the dataset: -

https://www.kaggle.com/datasets/captainozlem/framingham-chd-preprocessed-data

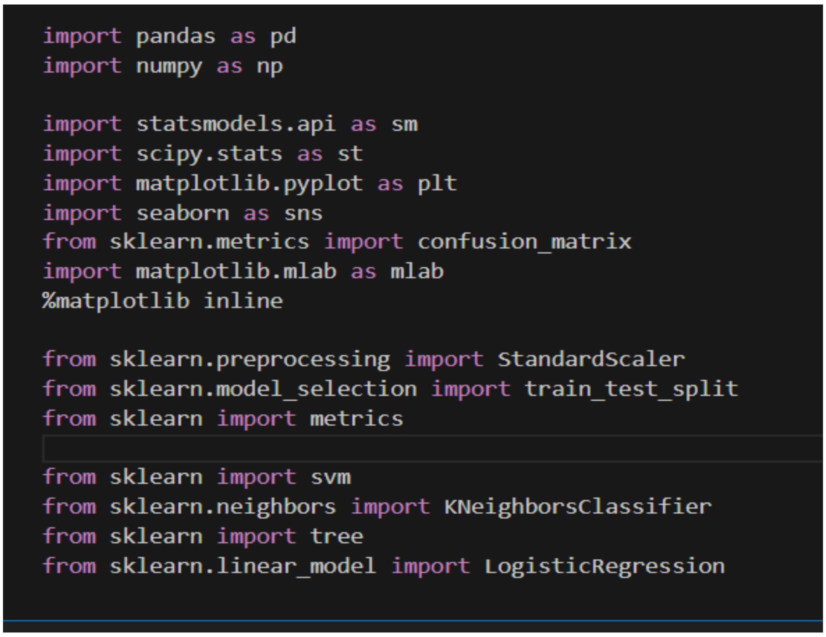
The dataset is in csv format. There is total 4134 numbers of rows and 16 columns.

Column Names Are: - Gender, Age,Education, CurrentSmoker, CigsPerDay, BPMeds, prevalentStroke, prevalentHyp, diabetes, totChol, sysBP, diaBP, BMI, heartrate, glucose, TenYearCHD

Categorical Columns: - Gender, Education, CurrentSmoker, BPMeds, prevalentStroke, prevalentHyp, diabetes, TenYearCHD

Numerical Column: - Age, CigsPerDay, totChol, sysBP, diaBP, BMI, heartrate, glucose.

Essential Python Libraries: -

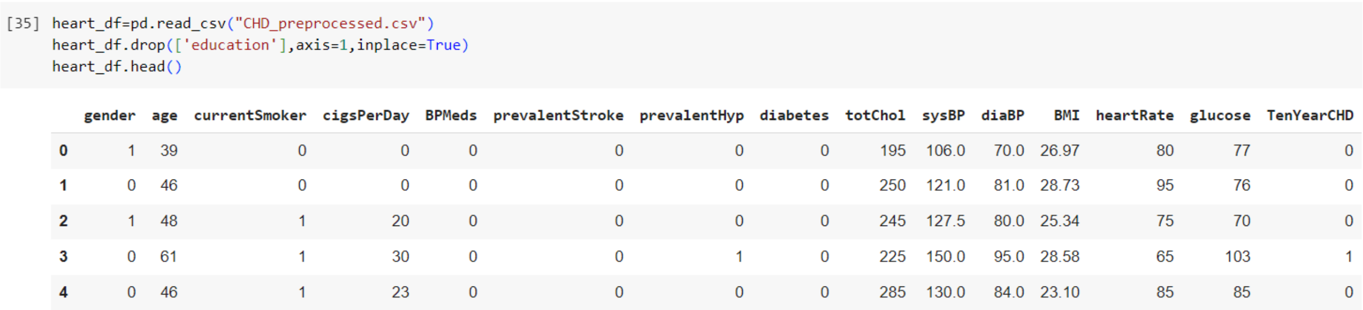


```
import pandas as pd
import numpy as np

import statsmodels.api as sm
import scipy.stats as st
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix
import matplotlib.mlab as mlab
%matplotlib inline

from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn import metrics

from sklearn import svm
from sklearn.neighbors import KNeighborsClassifier
from sklearn import tree
from sklearn.linear_model import LogisticRegression
```

A comprehensive dataset containing anonymized patient information, including symptoms, laboratory test results, and medical history, was obtained from electronic health records and relevant medical databases.

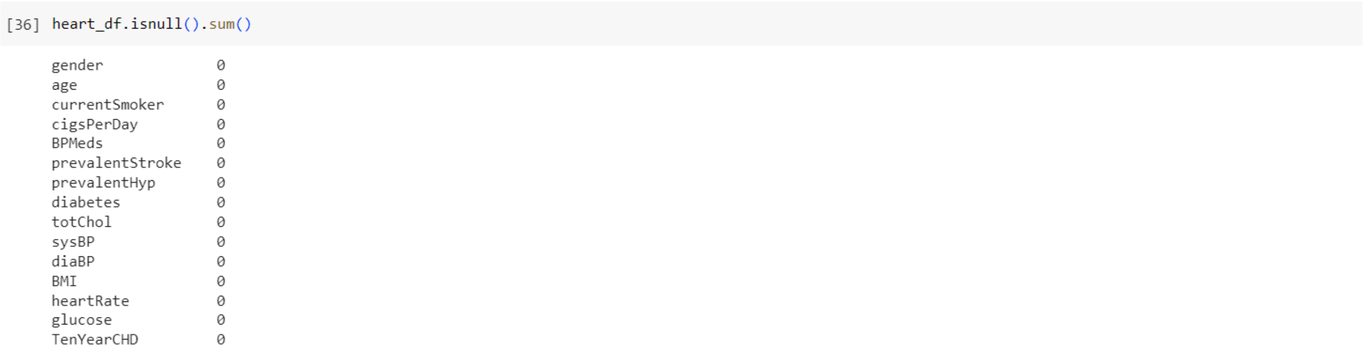
2.2 Data Preprocessing:

The collected data underwent rigorous preprocessing steps, including data cleaning and standardization to ensure uniformity and enhance the model's performance, since no null value was present, we farther proceed with the dataset. The column – education was irrelevant in this evaluation process so we drop that.



```
[35] heart_df=pd.read_csv("CHD_preprocessed.csv")
     heart_df.drop(['education'],axis=1,inplace=True)
     heart_df.head()
```

	gender	age	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose	TenYearCHD
0	1	39	0	0	0	0	0	0	195	106.0	70.0	26.97	80	77	0
1	0	46	0	0	0	0	0	0	250	121.0	81.0	28.73	95	76	0
2	1	48	1	20	0	0	0	0	245	127.5	80.0	25.34	75	70	0
3	0	61	1	30	0	0	1	0	225	150.0	95.0	28.58	65	103	1
4	0	46	1	23	0	0	0	0	285	130.0	84.0	23.10	85	85	0

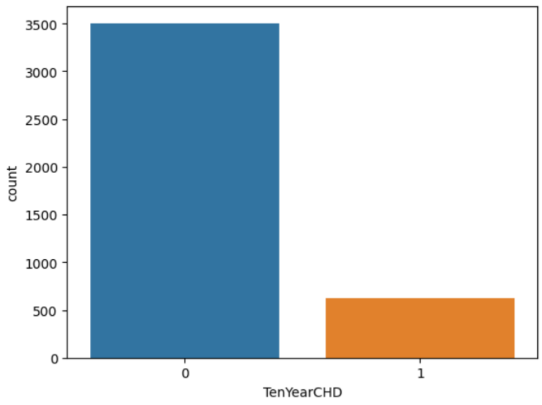


```
[36] heart_df.isnull().sum()

     gender             0
     age                0
     currentSmoker      0
     cigsPerDay         0
     BPMeds             0
     prevalentStroke    0
     prevalentHyp       0
     diabetes           0
     totChol            0
     sysBP              0
     diaBP              0
     BMI                0
     heartRate          0
     glucose            0
     TenYearCHD         0
```

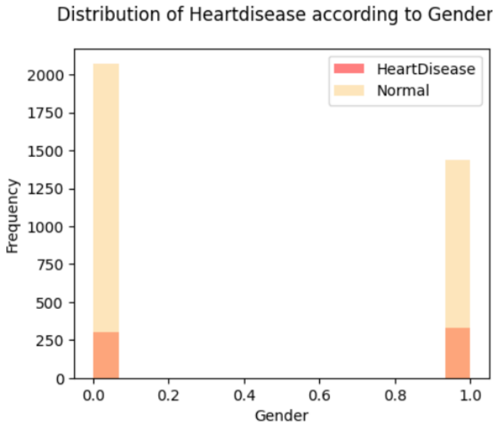
2.3 Exploratory Data Analysis (EDA):

Data Visualization:

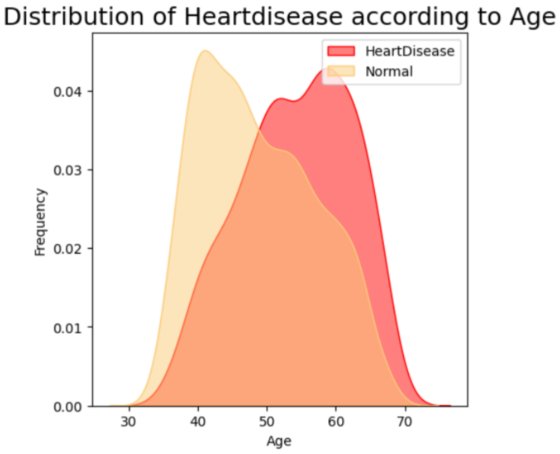
2.3.1



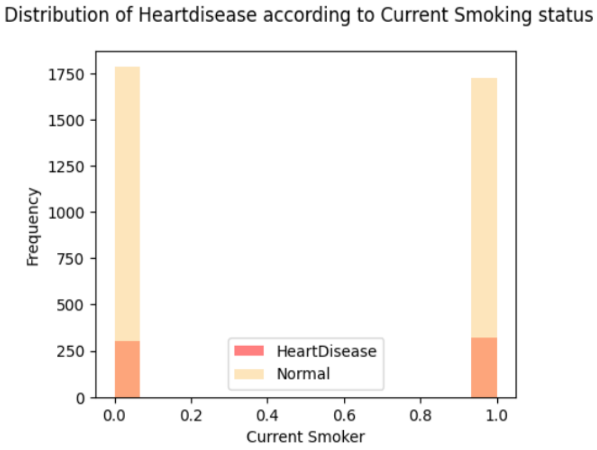
2.3.2



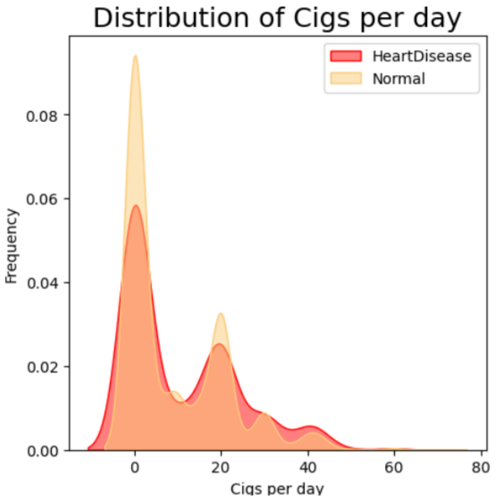
2.3.3



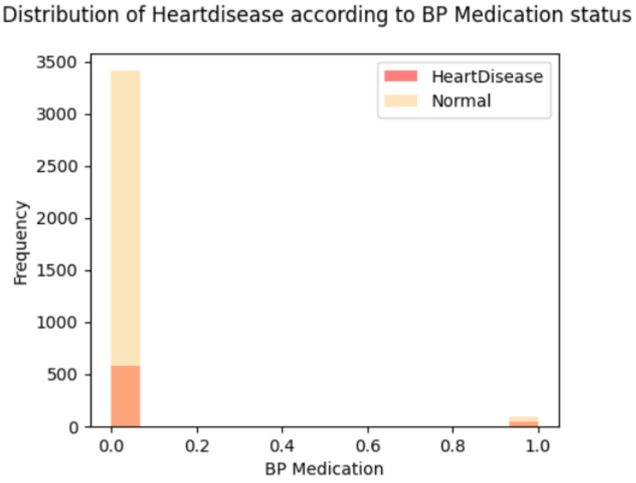
2.3.4



2.3.5

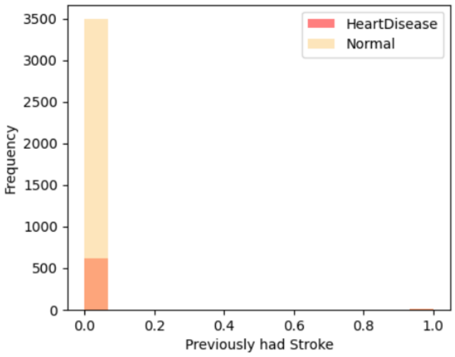


2.3.6



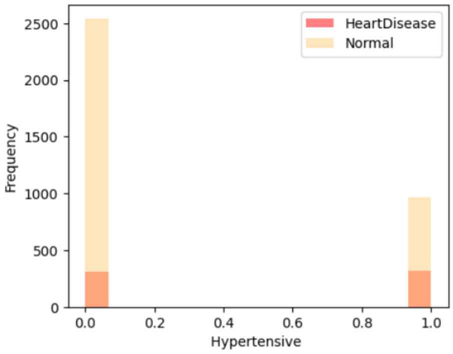
2.3.7

Distribution of Heartdisease according to Previously had Stroke record



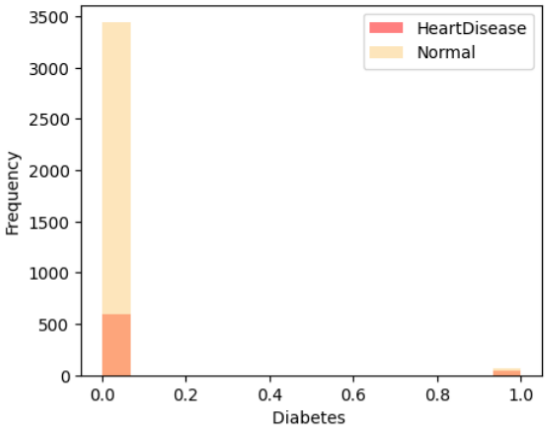
2.3.8

Distribution of Heartdisease according to Hypertensiveness status



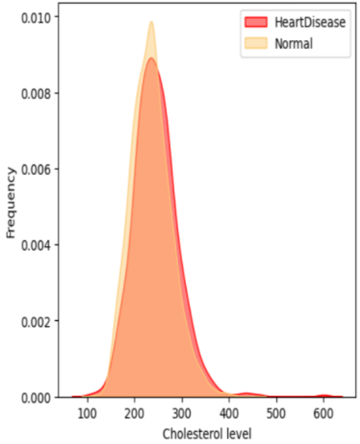
2.3.9

Distribution of Heartdisease according to Diabetes



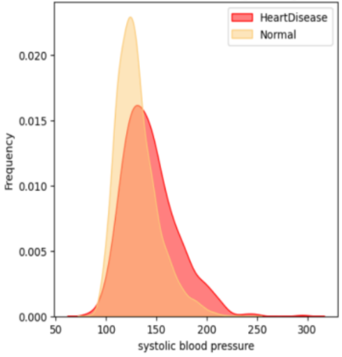
2.3.10

Distribution of Heartdisease according to Cholesterol level



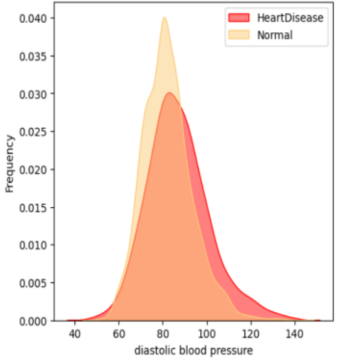
2.3.11

Distribution of Heartdisease according to systolic blood pressure

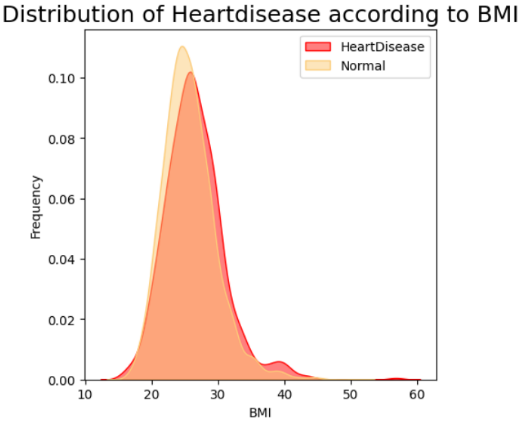


2.3.12

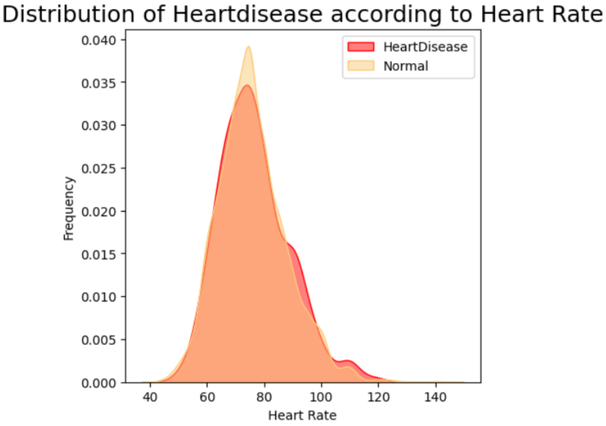
Distribution of Heartdisease according to diastolic blood pressure



2.3.13



2.3.14

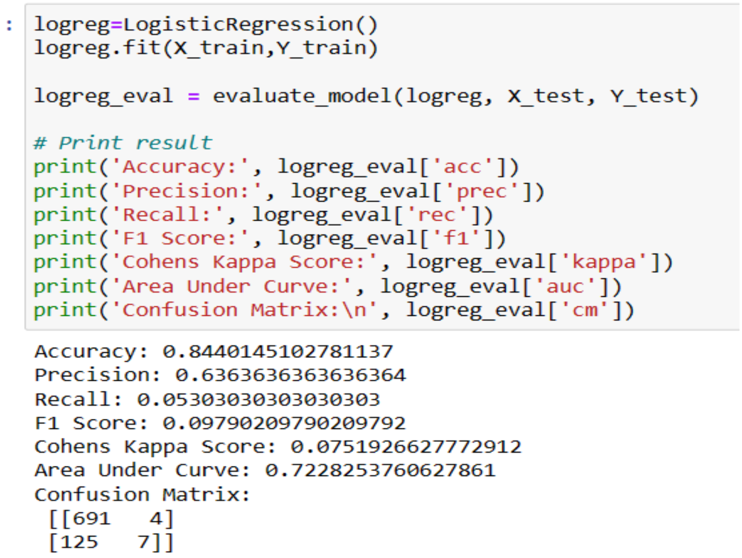


2.4 Model Selection:

Several machine learning algorithms, including Decision trees, K-Nearest Neighbour and Logistic Regression, were considered and evaluated to identify the most suitable model for the given task.

2.5 Model Training:

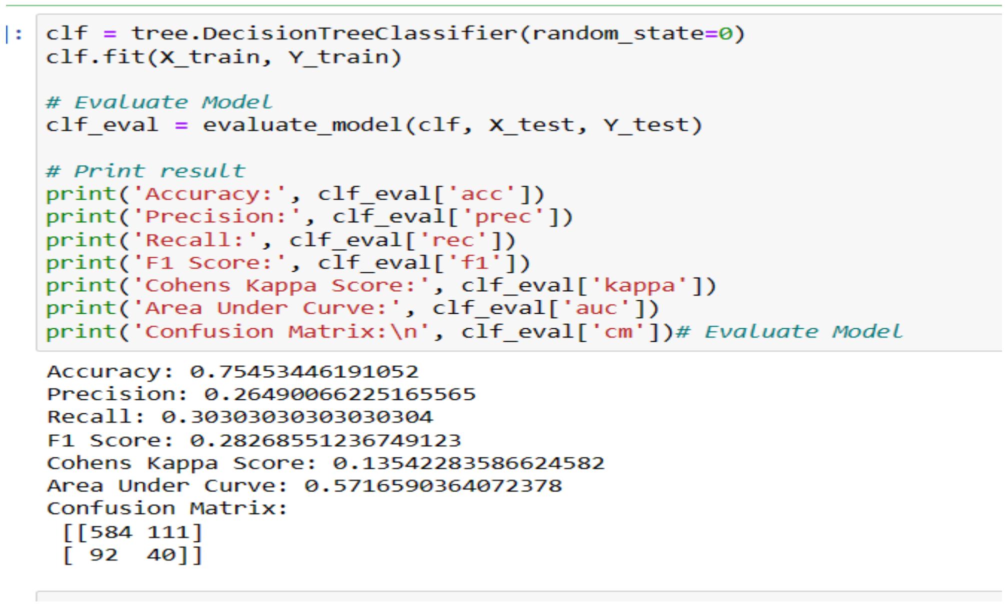
Logistic Regression: -



```
logreg=LogisticRegression()
logreg.fit(X_train,Y_train)

logreg_eval = evaluate_model(logreg, X_test, Y_test)

# Print result
print('Accuracy:', logreg_eval['acc'])
print('Precision:', logreg_eval['prec'])
print('Recall:', logreg_eval['rec'])
print('F1 Score:', logreg_eval['f1'])
print('Cohens Kappa Score:', logreg_eval['kappa'])
print('Area Under Curve:', logreg_eval['auc'])
print('Confusion Matrix:\n', logreg_eval['cm'])
```

```
Accuracy: 0.8440145102781137
Precision: 0.6363636363636364
Recall: 0.05303030303030303
F1 Score: 0.09790209790209792
Cohens Kappa Score: 0.0751926627772912
Area Under Curve: 0.7228253760627861
Confusion Matrix:
 [[691   4]
 [125   7]]
```

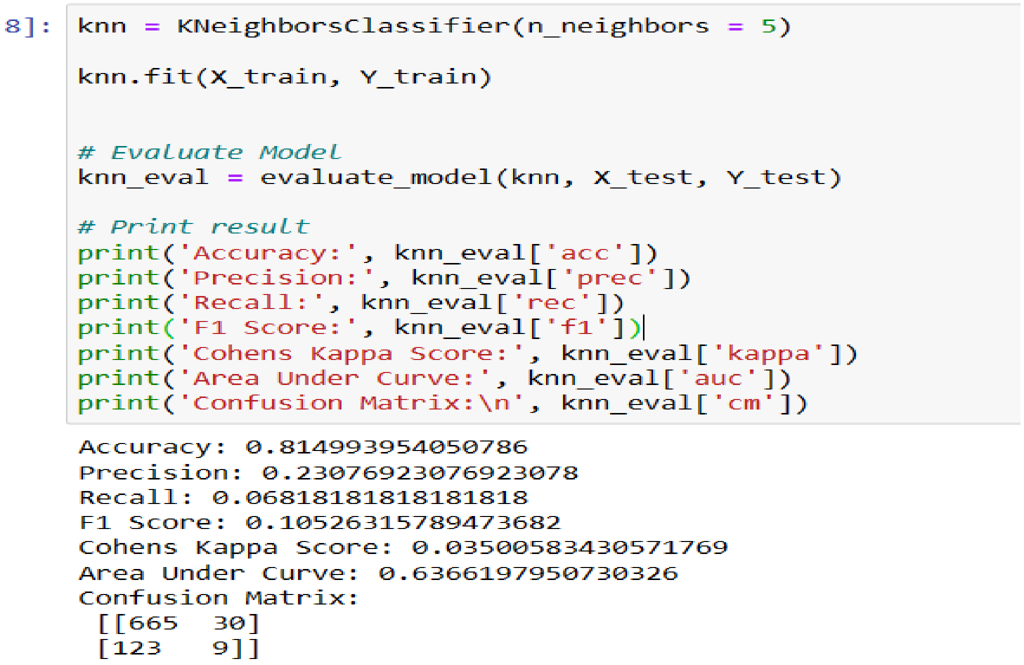
Decision Tree:-



```
clf = tree.DecisionTreeClassifier(random_state=0)
clf.fit(X_train, Y_train)

# Evaluate Model
clf_eval = evaluate_model(clf, X_test, Y_test)

# Print result
print('Accuracy:', clf_eval['acc'])
print('Precision:', clf_eval['prec'])
print('Recall:', clf_eval['rec'])
print('F1 Score:', clf_eval['f1'])
print('Cohens Kappa Score:', clf_eval['kappa'])
print('Area Under Curve:', clf_eval['auc'])
print('Confusion Matrix:\n', clf_eval['cm'])# Evaluate Model
```

```
Accuracy: 0.75453446191052
Precision: 0.26490066225165565
Recall: 0.30303030303030304
F1 Score: 0.28268551236749123
Cohens Kappa Score: 0.13542283586624582
Area Under Curve: 0.5716590364072378
Confusion Matrix:
 [[584 111]
 [ 92  40]]
```

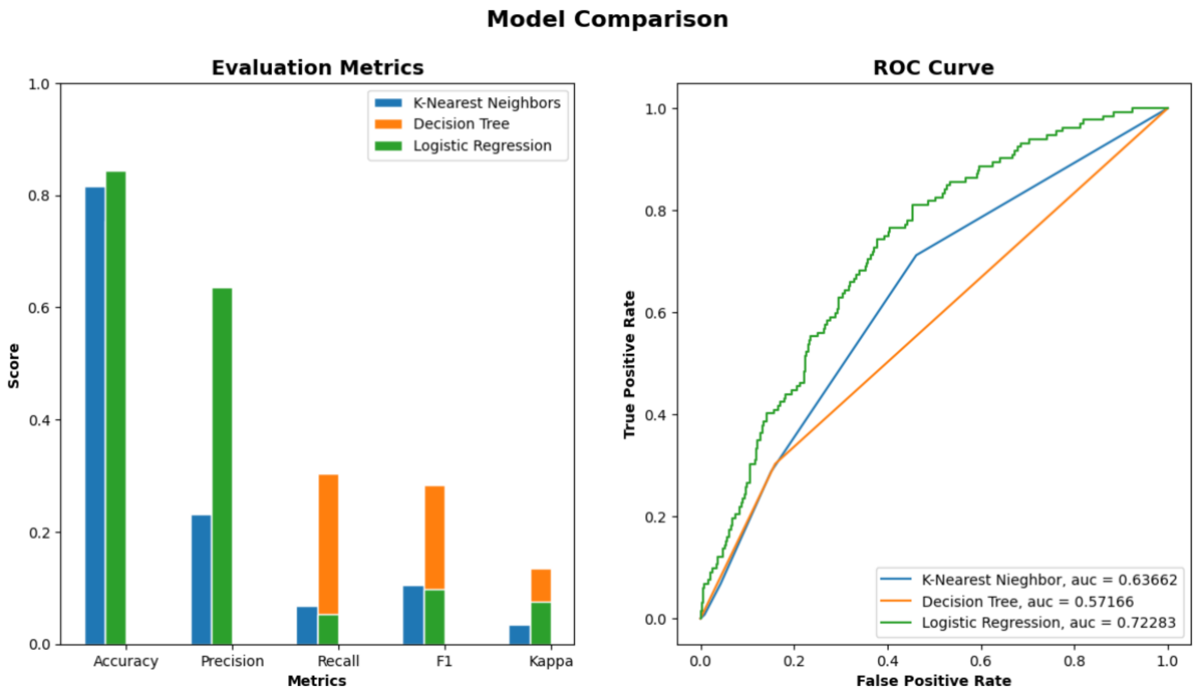
KNN:-



```
knn = KNeighborsClassifier(n_neighbors = 5)

knn.fit(X_train, Y_train)


# Evaluate Model
knn_eval = evaluate_model(knn, X_test, Y_test)

# Print result
print('Accuracy:', knn_eval['acc'])
print('Precision:', knn_eval['prec'])
print('Recall:', knn_eval['rec'])
print('F1 Score:', knn_eval['f1'])
print('Cohens Kappa Score:', knn_eval['kappa'])
print('Area Under Curve:', knn_eval['auc'])
print('Confusion Matrix:\n', knn_eval['cm'])
```

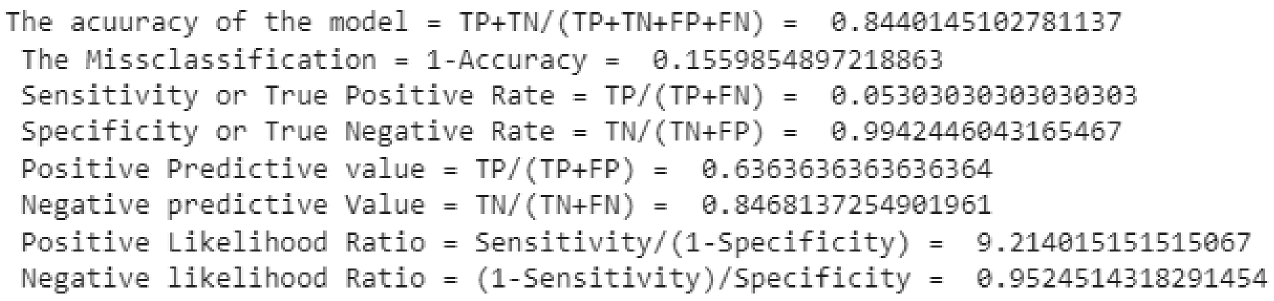
```
Accuracy: 0.814993954050786
Precision: 0.23076923076923078
Recall: 0.06818181818181818
F1 Score: 0.10526315789473682
Cohens Kappa Score: 0.03500583430571769
Area Under Curve: 0.6366197950730326
Confusion Matrix:
 [[665  30]
 [123   9]]
```

2.6 Model Accuracy & Validation:



Looking at the comparative study of the accuracy of 3 different machine learning algorithm, Logistics Regression model happens to have the highest accuracy of 84.4%.

The selected model was trained on a labelled dataset, and performance was validated using appropriate metrics such as accuracy, precision, recall, and F1-score.

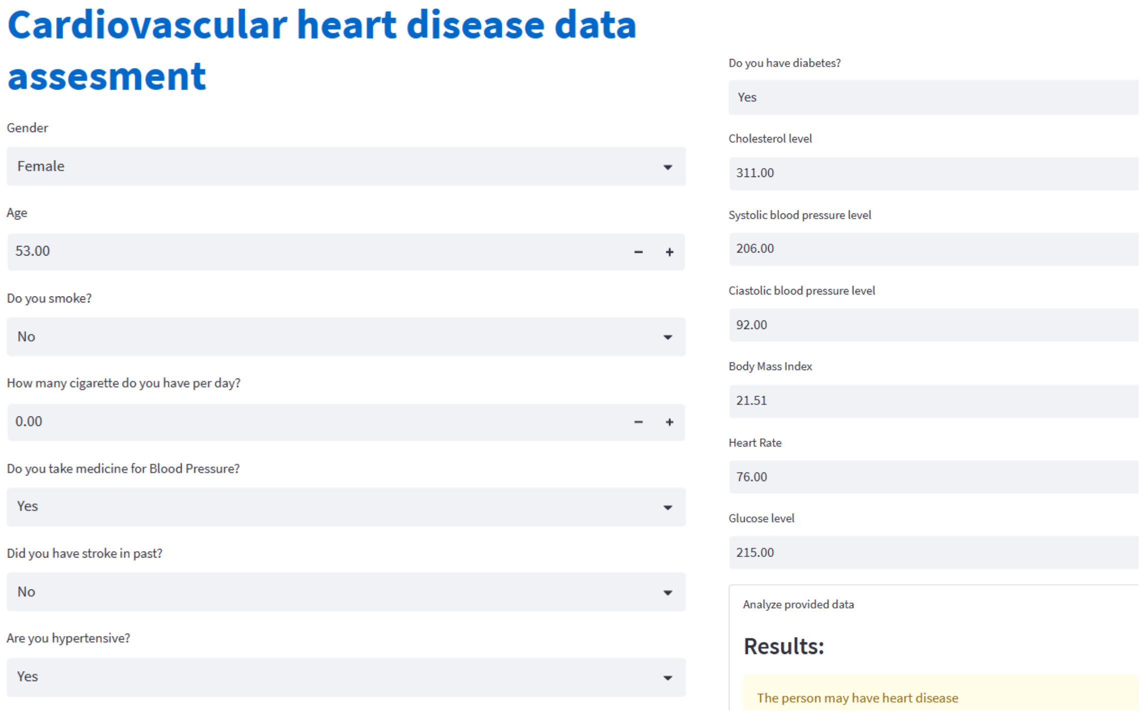


```
The acuuracy of the model = TP+TN/(TP+TN+FP+FN) =  0.8440145102781137
 The Missclassification = 1-Accuracy =  0.1559854897218863
 Sensitivity or True Positive Rate = TP/(TP+FN) =  0.05303030303030303
 Specificity or True Negative Rate = TN/(TN+FP) =  0.9942446043165467
 Positive Predictive value = TP/(TP+FP) =  0.6363636363636364
 Negative predictive Value = TN/(TN+FN) =  0.8468137254901961
 Positive Likelihood Ratio = Sensitivity/(1-Specificity) =  9.214015151515067
 Negative likelihood Ratio = (1-Sensitivity)/Specificity =  0.9524514318291454
```

3. Deployment:

Model deployment in Streamlit:

Based on the appropriate machine learning model we develop a FORM on streamlit platform where patient will be able to take assessment on the information provided by them. The app works as such- there are total 14 fields from where it takes input from patient. The information includes patient symptoms, test results, and medical history. Processing on the provided data the model predicts whether the person may have cardiovascular heart disease (which includes probability of having coronary heart disease, cerebrovascular disease, peripheral arterial disease, rheumatic heart disease, congenital heart disease etc) in future.



Cardiovascular heart disease data assesment

Gender: Female

Age: 53.00

Do you smoke?: No

How many cigarette do you have per day?: 0.00

Do you take medicine for Blood Pressure?: Yes

Did you have stroke in past?: No

Are you hypertensive?: Yes

Do you have diabetes?: Yes

Cholesterol level: 311.00

Systolic blood pressure level: 206.00

Ciastolic blood pressure level: 92.00

Body Mass Index: 21.51

Heart Rate: 76.00

Glucose level: 215.00

Results:

The person may have heart disease

4.1 Advantages and Limitations:

The advantages of the machine learning model include enhanced diagnostic accuracy, time efficiency, and the ability to handle large datasets. However, the limitations and potential biases of the model must be acknowledged, along with its dependency on the quality and representativeness of the training data.

4.2 Ethical Considerations:

The ethical implications of deploying machine learning models in clinical settings were carefully considered. Patient privacy, transparency in decision-making, and the potential biases introduced by the model were addressed to ensure ethical and responsible use.

5. Results:

The performance of the developed machine learning model was assessed based on various evaluation metrics. The model demonstrated high accuracy and robustness in diagnosing medical conditions, showcasing its potential as an effective diagnostic tool.

6. Conclusion:

The development of a machine learning model for medical condition diagnosis represents a significant step towards improving healthcare outcomes. The model's ability to leverage patient symptoms, test results, and medical history enables more accurate and timely diagnoses, assisting medical professionals in providing optimal care to patients. Nevertheless, continued research, validation, and collaboration between data scientists and medical experts are essential to further enhance the model's performance and realize its full potential in transforming medical practice.