Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 5 (2023)

Prognosis of Cardiovascular Disease using Machine Learning Approach

[1] Mrs Hamsa A S, [2] Mrs Keerthana M M
[1][2] Assistant Professor, Department of Computer Science and Engineering, ATME College of Engineering, Mysuru, Karnataka, India

Abstract: This paper presents a machine learning model that will detect cardiovascular disease at the early stage. Machine learning is an effective tool, assisting in making decision and predictions from the large quantity of data. The proposed machine learning algorithms-based prediction model works with different combination of features and known classification technique to analyze the dataset. The dataset consists of 11 attributes and a target attribute to performing the analysis, where the model begins with the pre-processing phase and selects the most relevant features in the dataset, it applies Random Forest algorithm and got the high accuracy compared to other popular classifiers, Also Proposed model uses multiple trees as a result there is no overfitting problem. And its training time is less and run efficiently on larger database.
Keywords—Random Forest Algorithm, Machine Learning, Python

1. Introduction

Machine learning for healthcare helps improve the efficiency and speed of medical services, which can lead to significant cost savings. Over three quarter of heart related diseases take place in low and middle-income countries. People living in low income and developing region often do not have the benefit of primary health care programmers for early detection. As a result, for many people in these sectors, detection is often late in the course of the disease and people die due to heart related diseases and other noncommunicable diseases, often in their younger age.

It is important to detect heart disease as early as possible so, that management with medicines can begin. The primary aim of this project is to analyses the various machine learning techniques and anticipate if someone in particular, given different individual attributes and indications, will get heart disease or not. The secondary aim is to produce high accuracy and build a model which takes less time compared to existing model.

Almost 32% of all deaths are due to heart related disease in all over the world.1 in 5 heart attack patients are younger than 40 years of age. The heart disease rate is increasing by 2% every year among young age. Early detection and treatment of several heart diseases is very complex, especially in developing countries, because of the lack of diagnostic centers and other resources that affect the accurate prognosis of heart disease. With this concern, this project makes use of computer technology and machine learning techniques to make medical aid software as a support system for early diagnosis of heart disease. Identification of any heart related illness at primary stage can reduce the death risk. Various ML techniques are used on medical data to understand the pattern of data and making prediction from them. The Machine Learning algorithms are designed to perform a large number of tasks such as prediction, classification, decision making etc. To learn the ML algorithms, training data is required. The available heart disease database consists of both numerical and categorical data. Before further processing, cleaning and filtering are applied on these records in order to filter the irrelevant data from the database. The processed data is given to different machine learning classifier for training purpose, this is also known as learning phase.

After the learning phase, a model is produced which is considered as an output of ML algorithm. The model is then tested and validated on a set of real time testing dataset. The final accuracy of the model is then compared with the actual value, which verifies the overall correctness of predicted result. Later, when the user provides his/her data to check about their health status the prediction is made accurately.

2. Existing System

In the existing system the implementation as not been found effective. The implementation was built using K-nearest neighbor, Support vector machine (SVM) and Logistic regression algorithm. The results generated were less accurate. The precision of decision tree was found be less around 80%. The dataset to train

the system was not adequate which affected the accuracy rate. The performance was noted to be low when new dataset was added to train the model. The preprocessing of data were another concern which led to overfitting and underfitting.

The existing system is implemented with minimal quantity of data which are insufficient in training and provided the required result. SVM is used to find an optimal boundary between the inputs, with inadequate data overlapping can be found while classification. SVM has many parameters that have to be met to achieve the best classification for the given problem. The logistic regression was not found suitable to obtain complex relationships between data.

Drawback:

In the existing system, the data collected as missing values which leads to incorrect model training and error in the detection of heart disease being present or normal. This practice leads to unwanted biases, errors and excessive medical costs which affects the quality of service provided to patients.

3. Proposed System

The proposed system has its main focus to build a dependable system to predict heart disease based on the user data. All the drawbacks of the previous systems will be solved using our proposed design. The data from research healthcare centers databases are used for training which are verified real time data. Any new data can be added to the existing dataset to train without any concern.

First, the data is collected and data preprocessing takes place. In data preprocessing, the missing values in the dataset are found. these may affect the accuracy hence; we replace these values with mean of column. Later, these numeric values of the dataset are changed to nominal to make it compatible with machine learning techniques. These datasets are then split into training and testing. The ratio is 80:20 where 80% is used for training purpose and the rest 20% is used for testing purpose.

The classification model is applied on the training data and trained. The testing data is then given to check these classification model, the results of the training data of the ML classifier is verified using confusion matrix.

Features:

1. With the proposed system that integration of clinical decision support with computer-based patient records could reduce medical errors, enhance patient safety and improve outcome.
2. High performance and accuracy rate.
3. The reliability system which can accurately predict the result using 11 dataset attributes.

4. Methodology

PYTHON: Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library. Python is the major coding language since the system needs to analyses data and make predictions of data attributes. Python is utilized for data analytics, and we've employed it for machine learning.

FLASK: Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object- relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Flask is used in our project for the purpose of providing a web application service to the end user. Flask has been used to create a webpage with python classes for performing the needed processing on the provided data

HTML: The Hypertext Markup Language, or HTML is the standard markup language for documents

designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. The front end has been designed by making use of the HTML language, since it's easy to understand and we can build stable webpages we have chosen HTML.

CSS: Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file which reduces complexity and repetition in the structural content as well as enabling the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.CSS has been used in our project for the purpose of giving a proper look or design to the webpages.

VISUAL STUDIO: Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS.[10] Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality

CLASSIFICATION: Classification means classifying the data in different groups based on the similarities present in different data points. Here classification is used in the prediction of heart disease.

CONFUSION MATRIX: The confusion matrix is a matrix used to determine the performance of the classification models for a given set of test data. It can only be determined if the true values for test data are known. Since it shows the errors in the model performance in the form of a matrix, hence also known as an error matrix.

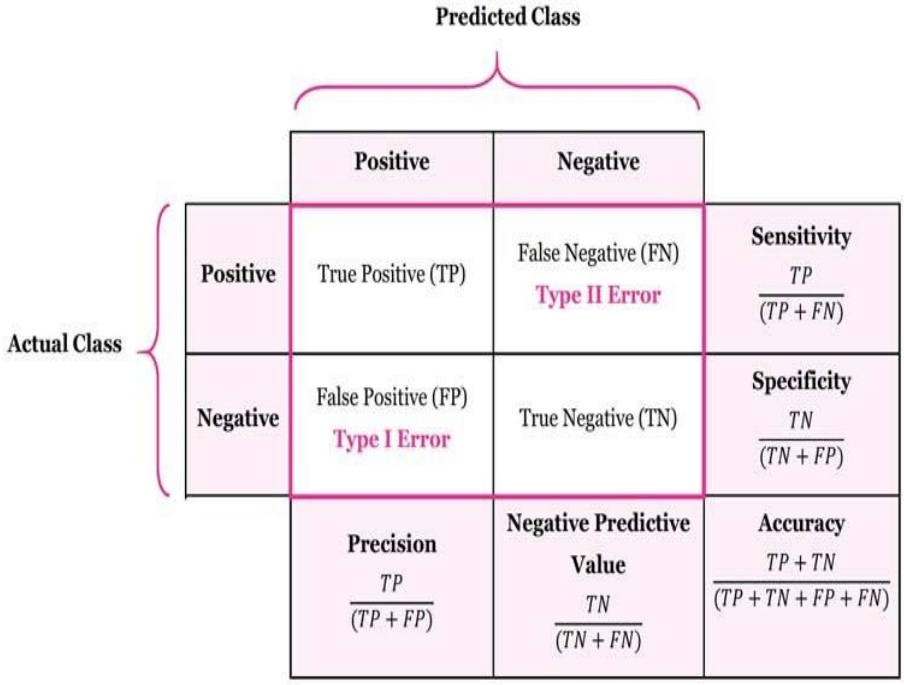


Fig 1 Confusion matrix

For the 2 prediction classes of classifiers, the matrix is of 2*2 table, for 3 classes, it is 3*3 table, and so on. The matrix is divided into two dimensions, that are predicted values and actual values along with the total number of predictions. Predicted values are those values, which are predicted by the model, and actual values are the true values for the given observations

The above table has the following cases:

1. True Negative: Model has given prediction No, and the real or actual value was also No.
2. True Positive: The model has predicted yes, and the actual value was also true.
3. False Negative: The model has predicted no, but the actual value was Yes, it is also called as Type-II error.
4. False Positive: The model has predicted Yes, but the actual value was No. It is also called a Type-I error.

RANDOM FOREST: It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting

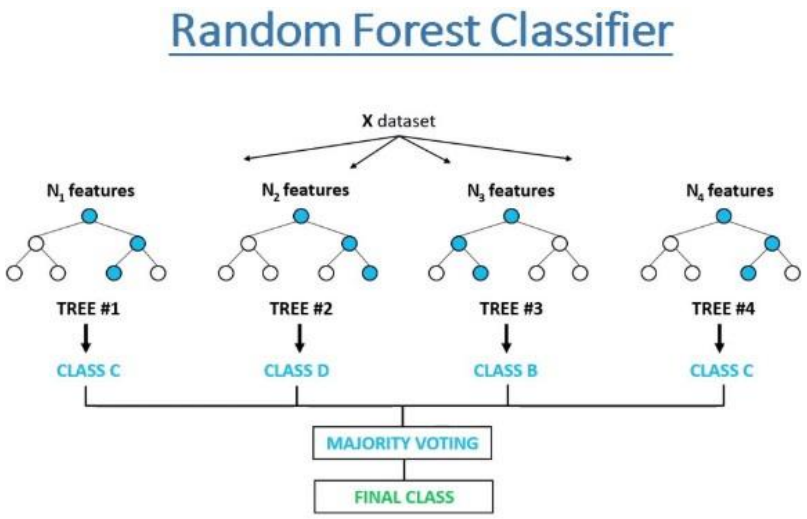
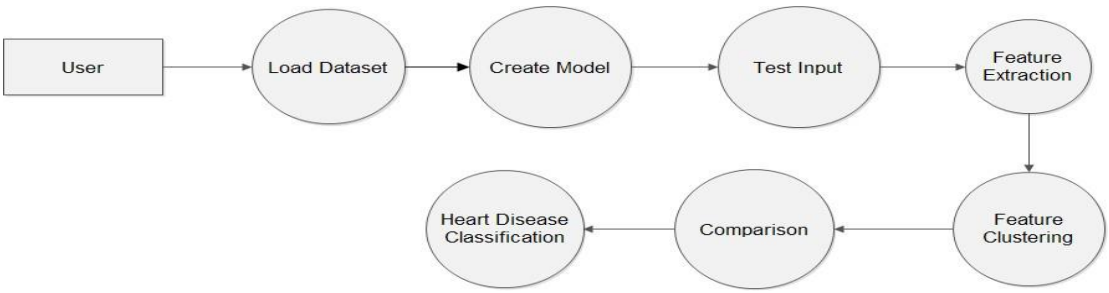


Fig.2 Random Forest Classifier

5. System Design

DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system. The visual representation makes it a good communication tool between User and System designer. Structure of DFD allows starting from a broad overview and expand it to a hierarchy of detailed diagrams.



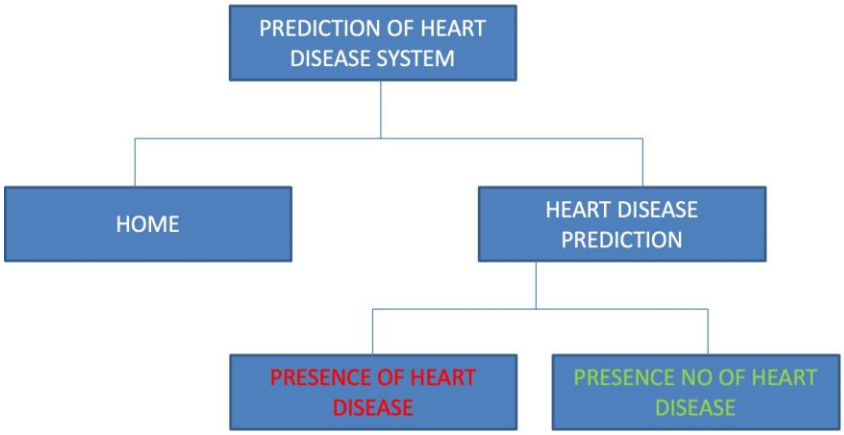


Fig. 3 System Architecture - 1

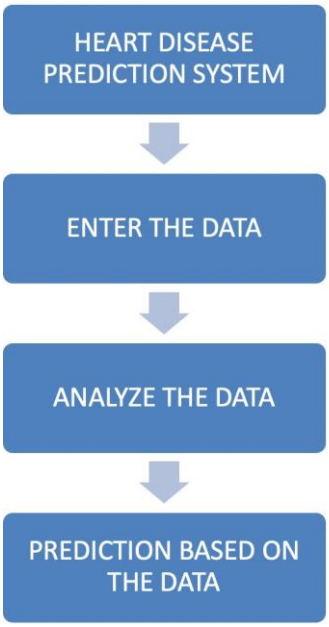


Fig. 4 System Architecture - 2

Dataset: Dataset from 5 different and popular heart disease dataset were combined to form one whole dataset. This dataset consists of 1190 entries,11 features and a target variable. It has6 nominal variables and 5 numeric variables.

The five-dataset used for this project are:

- Cleveland heart disease dataset
- Hungarian heart disease dataset
- Statlog heart disease dataset
- Switzerland cardiovascular disease dataset
- UK heart disease dataset

The detailed description of all the features are as follows:

NUMERIC:

- Age: Patients Age in years

- Resting bp: Level of blood pressure at resting mode in mm/HG
- Cholestrol: Serum cholestrol in mg/dl
- Max heart rate: Maximum heart rate achieved
- Oldpeak: Exercise induced ST-depression in comparison with the state of rest

NOMINAL:

- Sex: Gender of patient (Male - 1, Female - 0)
- Chest Pain Type: Type of chest pain experienced by patient categorized into 1 typical, 2 typical angina, 3 non- anginal pain, 4 asymptomatic
- Fasting blood sugar: Blood sugar levels on fasting > 120 mg/dl represents as 1 in case of true and 0 as false
- Resting ecg: Result of electrocardiogram while at rest are represented in 3 distinct values 0 : Normal 1: Abnormality in ST-T wave 2: Left ventricular hypertrophy
- Exercise angina: Angina induced by exercise 0 depicting NO 1 depicting Yes
- ST slope: ST segment measured in terms of slope during peak exercise 0: Normal 1: Upsloping 2: Flat 3: Downsloping.

6. Implementation

Data Aggregation:

Initially, the dataset for the heart disease prediction system is collected. After the collection of the dataset, the dataset is preprocessed before using it train and build the model,later we split the dataset into training data and testing data. The training dataset is used for prediction model learning and testing data is used for evaluating the prediction model. For this project, 70% of training data is used and 30% of data is used for testing.

Data Preprocessing:

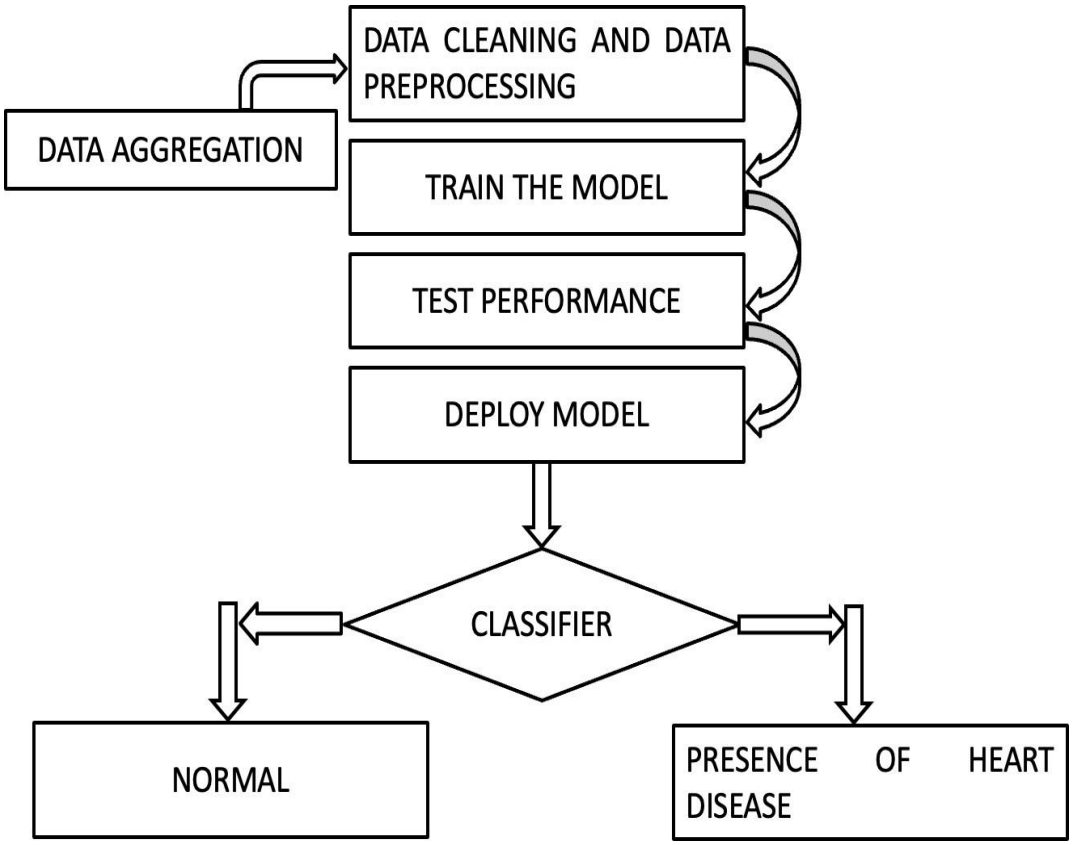
```
import pandas as pd
import numpy as np

# data visualization
import matplotlib.pyplot as plt
%matplotlib inline


from sklearn.model_selection import train_test_split

#model validation
from sklearn.metrics import log_loss,precision_score,f1_score,recall_score
from sklearn.metrics import classification_report,confusion_matrix,accuracy_score,matthews_corrcoef
from sklearn import metrics

from sklearn.ensemble import RandomForestClassifier
```



Import Datasets:

```
df = pd.read_csv('heart.csv')
```

```
df.head()
```

	age	sex	chest pain type	resting bp s	cholesterol	fasting blood sugar	resting ecg	max heart rate	exercise angina	oldpeak	ST slope	target
0	40	1	2	140	289	0	0	172	0	0.0	1	0
1	49	0	3	160	180	0	0	156	0	1.0	2	1
2	37	1	2	130	283	0	1	98	0	0.0	1	0
3	48	0	4	138	214	0	0	108	1	1.5	2	1
4	54	1	3	150	195	0	0	122	0	0.0	1	0

```
df.tail()
```

	age	sex	chest pain type	resting bp s	cholesterol	fasting blood sugar	resting ecg	max heart rate	exercise angina	oldpeak	ST slope	target
1185	45	1	1	110	264	0	0	132	0	1.2	2	1
1186	68	1	4	144	193	1	0	141	0	3.4	2	1
1187	57	1	4	130	131	0	0	115	1	1.2	2	1
1188	57	0	2	130	236	0	2	174	0	0.0	2	1
1189	38	1	3	138	175	0	0	173	0	0.0	1	0

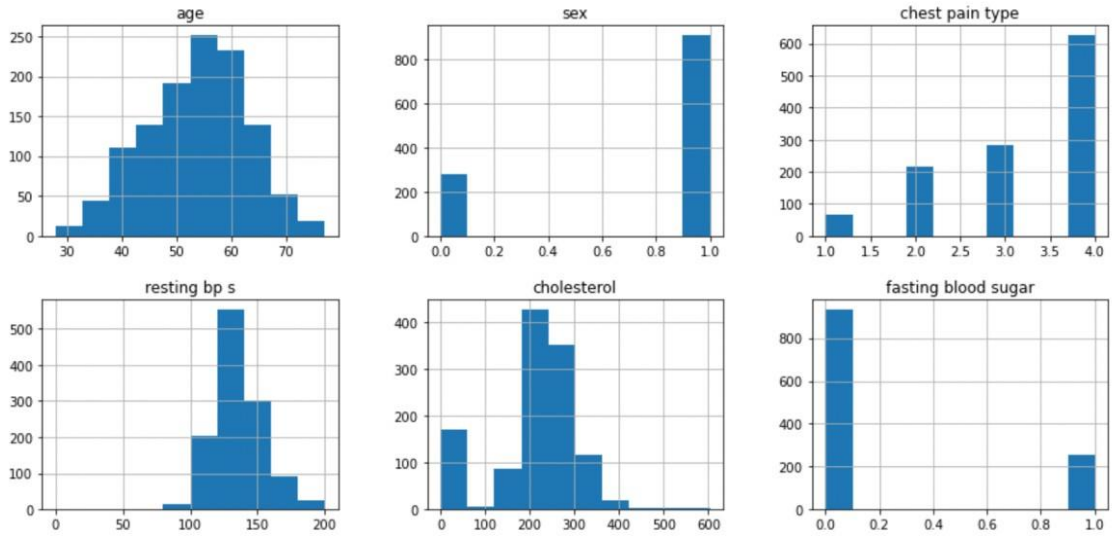
Finding Null Values:

Attributes Visualization:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1190 entries, 0 to 1189
Data columns (total 12 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   age                  1190 non-null   int64
 1   sex                  1190 non-null   int64
 2   chest pain type      1190 non-null   int64
 3   resting bp s         1190 non-null   int64
 4   cholesterol          1190 non-null   int64
 5   fasting blood sugar  1190 non-null   int64
 6   resting ecg          1190 non-null   int64
 7   max heart rate       1190 non-null   int64
 8   exercise angina      1190 non-null   int64
 9   oldpeak              1190 non-null   float64
 10  ST slope             1190 non-null   int64
 11  target               1190 non-null   int64
dtypes: float64(1), int64(11)
memory usage: 111.7 KB
```

```
df.hist(figsize=(15,15))
plt.savefig('featuresplot')
```



Renaming Attributes:

```
dt.columns = ['age', 'sex', 'chest_pain_type', 'resting_blood_pressure', 'cholesterol', 'fasting_blood_sugar', 'rest
       'exercise_induced_angina', 'st_depression', 'st_slope','target']
```

```
dt['chest_pain_type'][dt['chest_pain_type'] == 1] = 'typical angina'
dt['chest_pain_type'][dt['chest_pain_type'] == 2] = 'atypical angina'
dt['chest_pain_type'][dt['chest_pain_type'] == 3] = 'non-anginal pain'
dt['chest_pain_type'][dt['chest_pain_type'] == 4] = 'asymptomatic'


dt['rest_ecg'][dt['rest_ecg'] == 0] = 'normal'
dt['rest_ecg'][dt['rest_ecg'] == 1] = 'ST-T wave abnormality'
dt['rest_ecg'][dt['rest_ecg'] == 2] = 'left ventricular hypertrophy'


dt['st_slope'][dt['st_slope'] == 1] = 'upsloping'
dt['st_slope'][dt['st_slope'] == 2] = 'flat'
dt['st_slope'][dt['st_slope'] == 3] = 'downsloping'

dt["sex"] = dt.sex.apply(lambda  x:'male' if x==1 else 'female')
```

```
dt['chest_pain_type'].value_counts()
```

```
asymptomatic        625
non-anginal pain    283
atypical angina     216
typical angina       66
Name: chest_pain_type, dtype: int64
```

Value Count:

```
dt['rest_ecg'].value_counts()
```

```
normal                          684
left ventricular hypertrophy    325
ST-T wave abnormality           181
Name: rest_ecg, dtype: int64
```

```
dt['st_slope'].value_counts()
```

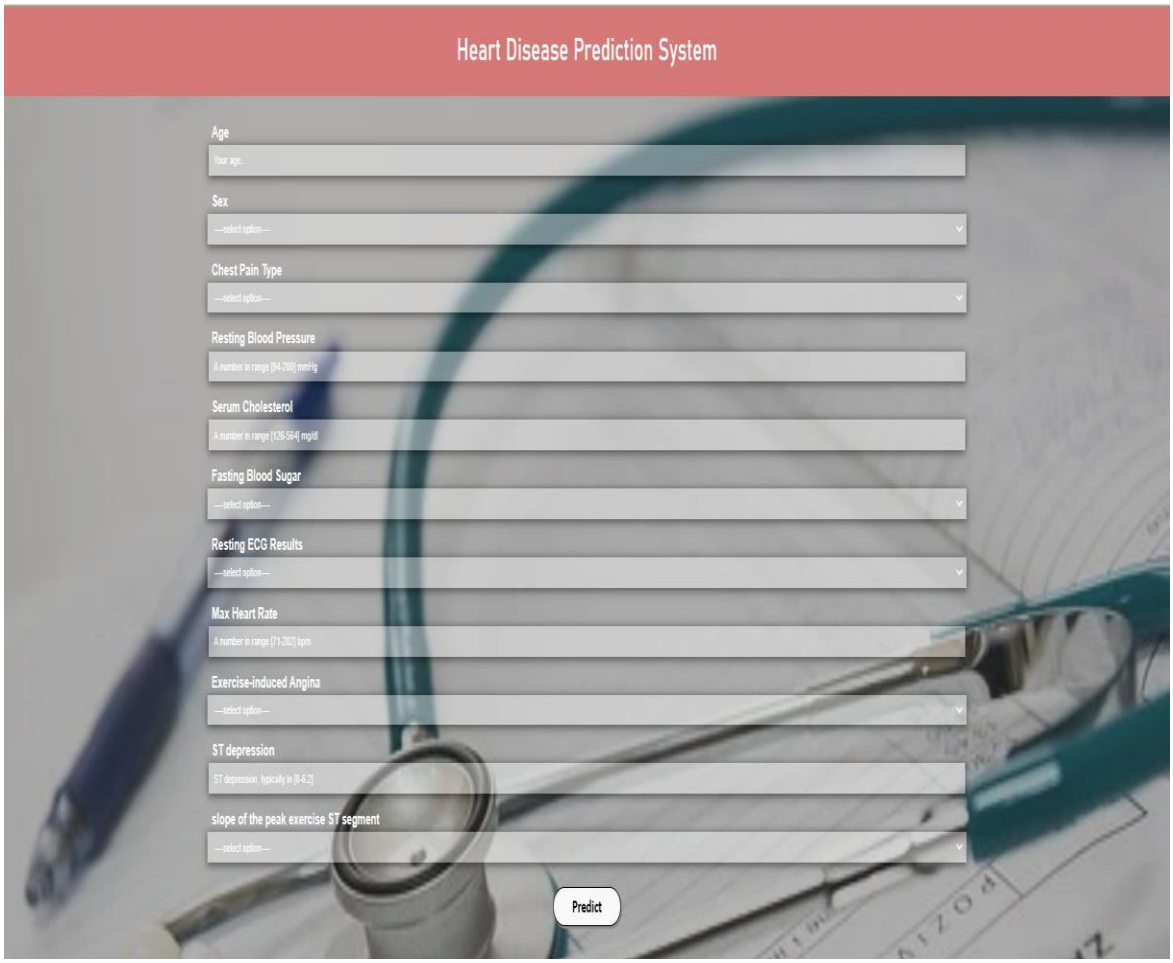
```
flat           582
upsloping      526
downsloping     81
0                1
Name: st_slope, dtype: int64
```

```
dt.drop(dt[dt.st_slope ==0].index, inplace=True)
dt['st_slope'].value_counts()
```

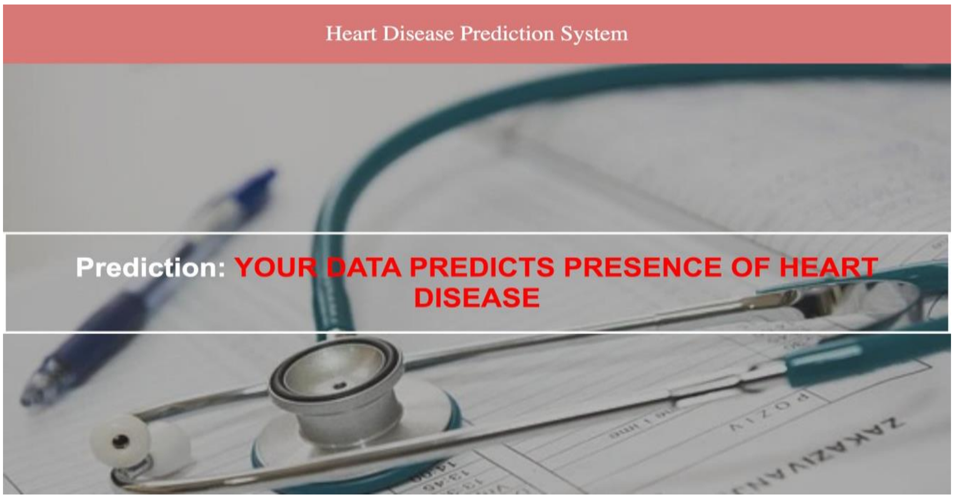
```
flat           582
upsloping      526
downsloping     81
Name: st_slope, dtype: int64
```

Results

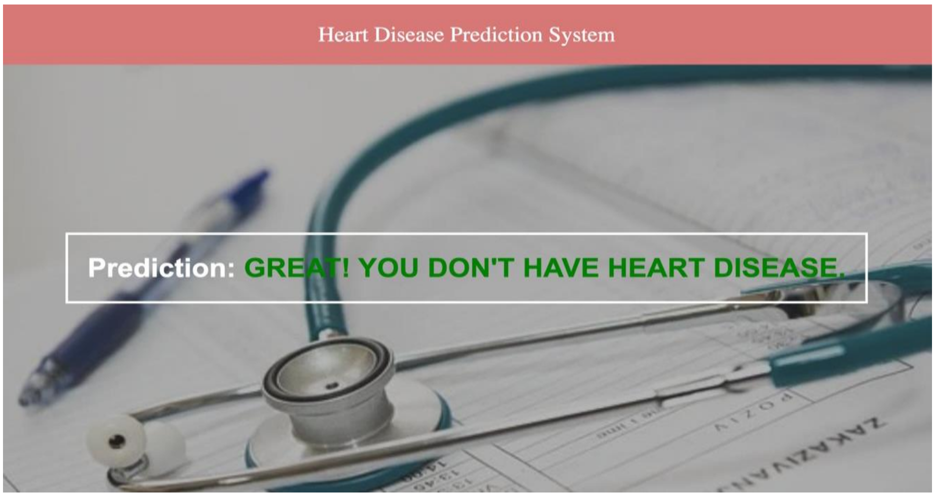
Heart disease prediction system



Presence of heart disease (case-i)



Normal heart (case-ii)



Confusion matrix and Accuracy

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    JUPYTER

2   37   1        2    130    283        0    1     98     0   0.0    1    0
3   48   0        4    138    214        0    0    108     1   1.5    2    1
4   54   1        3    150    195        0    0    122     0   0.0    1    0
0.9453781512605042
Classification Report
               precision    recall  f1-score   support

           0       0.96      0.92      0.94       107
           1       0.93      0.97      0.95       131

    accuracy                           0.95       238
   macro avg       0.95      0.94      0.94       238
weighted avg       0.95      0.95      0.95       238

Accuracy: 94.54%

[[ 98   9]
 [  4 127]]
(base) neeraj@Neerajs-MacBook-Air old file %
(base) neeraj@Neerajs-MacBook-Air old file %
```

7. Conclusion

Heart disease prediction is essential as well as challenging work in the medical field. Mortality rate can be reduced if the disease is recognized at the initial stages, which leads to proper treatment as soon as possible. Due to recent advancement in technology, the use of data and data analytics is used for every section of the society, the use of data analysis can be used in healthcare to make life saving decisions. The proposed system uses ECG parameters and random forest algorithm to train and build the model. The current accuracy of the model is 94.54% and is reliable.

Reference

[1] IEEE PAPERS – Vijeta Sharma,Manjari Gupta,Shrinkhala Yadav,"Heart disease prediction using Machine Learning",2020 2nd International Conference on Advancesin Computing.

[2] IEEE PAPERS – Archana Singh,Rakesh Kumar,"Heart disease prediction using Machine Learning algorithms",(ICE3-2020).

[3] IEEE PAPERS – Dr.Shailendra Narayan Singh,"Prediction of heart disease using Machine Learning algorithms",International Conference on Smart Technologies in Computing – 2020.

[4] IEEE PAPERS – Atharv Nikam,Sanket Bhandari,Aditya Mhaske,Shamla Mantri,"Cardiovascular disease prediction using Machine Learning Model", 2020 IEEE Pune Section International Conference (PuneCon) Vishwakarma Institute of Technology, Pune India. Dec 16-18, 2020.

[5] https://www.geeksforgeeks.org/machine-learning/

[6] https://www.who.int/health-topics/cardiovascular-diseases#tab=tab_1

[7] https://www.tutorialspoint.com/machine_learning_with_python/index.htm

[8] https://towardsdatascience.com/understanding-random-forest-58381e0602d2

[9] https://ieee-dataport.org/open-access/heart-disease-dataset