Visualizing and Predicting Heart Diseases with an Interactive Dash Board

Team I'd : PNT2022TMID10855

Submitted by,

Durga S
Kiruba T
Karpagalakshmi
Aiswarya Jeyabrinda S

ABSTRACT

Predicting and diagnosing heart disease is the biggest challenge in the medical industry and it is based on factors like physical examination, symptoms and signs of the patient. To overcome these problems, prediction of heart disease is being done by using machine learning algorithms and data mining techniques, it has become easy to perform automatic diagnosis in hospitals as they are playing vital role in this regard. Heart disease can be predicted by performing analysis on patient's different health parameters. Factors which influence heart diseases are cholesterol level of the body, smoking habit, and obesity, family history of diseases, blood pressure and working environment. In the project , the model experimented with Logistic Regression is proposed in Jupyter notebook and the data visualization is made with the help of IBM COGNOS and creating an Interactive Dashboard. Finally, the best results were obtained by Logistic Regression with the score of 83%.

TABLE OF CONTENTS

9. ADVANTAGES & DISADVANTAGES

10. FUTURE SCOPE

11. APPENDIX

Source Code
GitHub & Project Demo Link

1.INTRODUCTION

1.1 PROJECT OVERVIEW

Machine learning (ML) proves to be effective in assisting in making decisions and predictions from the large quantity of data produced by the healthcare industry. The dataset holds 270 records with 14 attributes such as age, chest pain type, blood pressure, blood glucose level, ECG in rest, heart rate and four types of chest pain. We'll be applying Machine Learning approaches(and eventually comparing them) for classifying whether a person is suffering from heart disease or not.

1.2 PURPOSE

Prediction is one of the area where machine learning plays a vital role, our topic is to predict heart diseases by processing patient's dataset and a data of patients i.e., user of whom we need to predict the chance of occurrence of a heart diseases. Finally we classify patients that are at risk of getting a heart disease or not and also this method is totally cost efficient.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM

1.The Efficient predication on Diseases : Data Analysis

Soumya Ranjan Jena ,July 2020

Healthcare industries generate massive amounts of data, known as big data, which contains hidden knowledge or patterns for decision making.

2.Heart Diseases prediction in machine leaning approach.

Dnyaneshwari Mahajan ,June-2020

Thehuman body Heart disease is the leading cause of death worldwide.Diseases. Some symptoms include chest pain, a faster heartbeat, and dizziness.

3.HEART DISEASES USING MACHINE LEARNING.

Yuvraj Nikhate , August 2020

Heart disease is one of the leading causes of death worldwide. Every year, approximately 17.9 million people die. Cardiovascular diseases are conditions that affect the heart and blood vessels.

4.HEART DISEASES PREICION USING DATA MININGAPPRAOCH.

M. Preethi , June 2020

This paper discusses data mining, big data, and machine learning models for predicting heart disease. Data mining and machine learning are important components in developing a model for a medical system to predict heart disease or cardiovascular disease.

5.HEART DISEASES PREDIACTION USING DATAANALYTCS

Mrs. Mehdi Khundmir Iliyas

Early detection of heart disease may reduce the death rate to some extent. This software aids in the early detection of heart disease. Nowadays, healthcare organizations generate massive amounts of data that are disorganized.

2.2 REFERENCE

- Dr.A.V.Senthil Kumar, *"Heart Disease Prediction Using Data Mining preprocessing and Hierarchical Clustering*", International Journal of Advanced Trends in Computer Science and Engineering, Volume-4, No.6, pp.07-18, 2015.
- Um a.K, M.Hanumathappa, "Heart Disease Prediction Using Classification Techniques with Feature Selection Method", Adarsh Journal of Information Technology, Volume-5, Issue-2, pp.22-29, 2016.
- Himanshu Sharma, M.A.Rizvi, *"Prediction of Heart Disease using Machine Learning Algorithms:A Survey",*International Journal on Recent and Innovation Trends in Computing and Communication,Volume5,Issue-8,pp.99-104, 2017.

- S.Suguna, Sakthi Sakunthala.N ,S.Sanjana, S.S.Sanjhana, "*A Survey on Prediction of Heart Disease using Big data Algorithms"*, International Journal of Advanced Research in Computer Engineering & Technology,Volume-6,Issue-3,pp.371-378,2017.

2.3 PROBLEM STATEMENT DEFINITION.

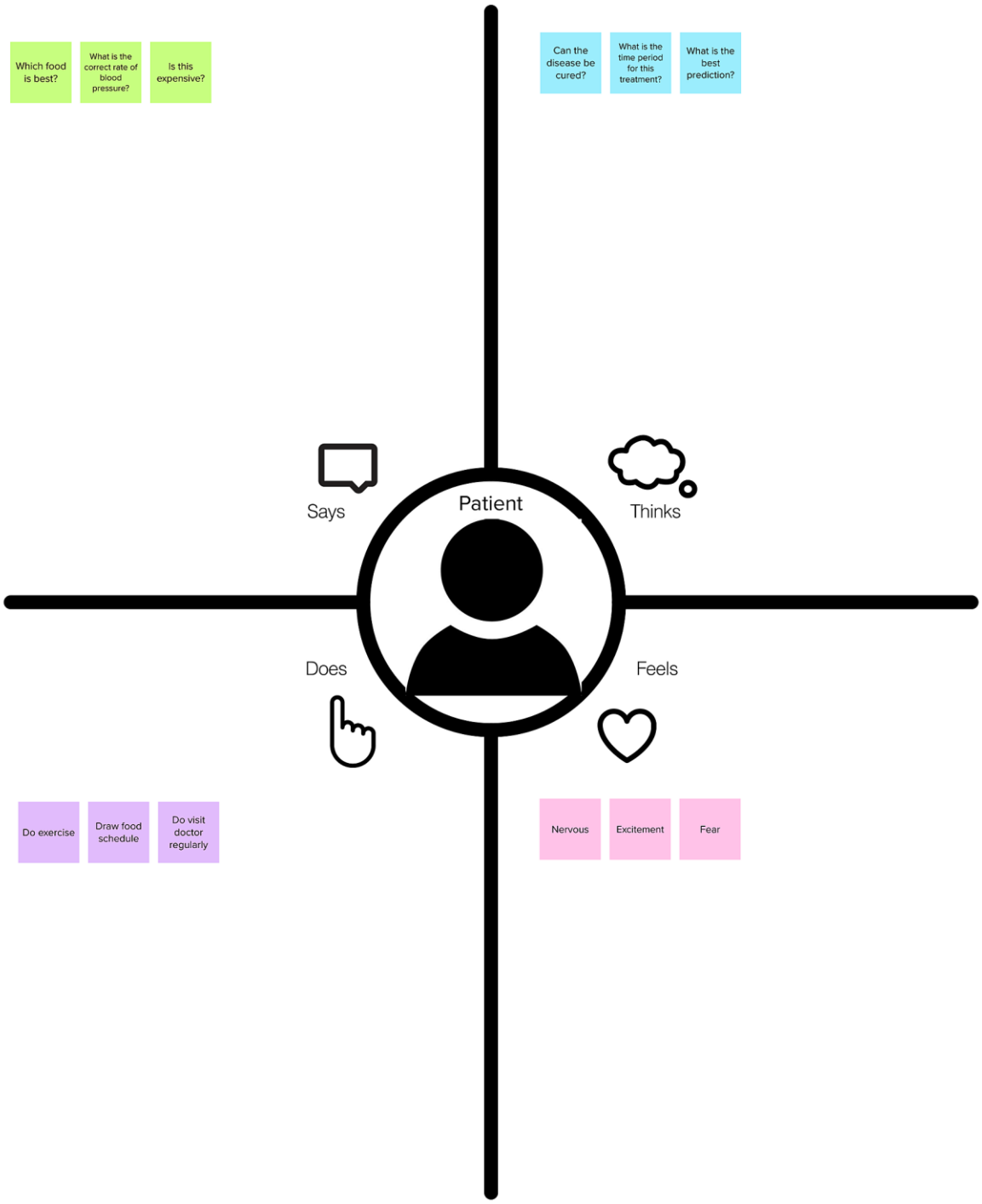
Heart Diseases remain the biggest cause of deaths for the last two decades. Recently computer technology and machine learning techniques are used to develop software to assist doctors in making appropriate decision of heart disease in an early stage. The diagnosis of heart disease depends on clinical and pathological data. Heart disease prediction system can assist medical professionals in predicting status of heart disease, based on the clinical data of patients.

Doctors may sometime fail to take an accurate decision in predicting heart disease risk level, therefore heart disease prediction systems are useful in such cases to get accurate results. There are many tools available for performing this task but all of them have some flaws. Most of the tools cannot handle big data and hence predicting heart disease would be a tedious task. In this project we are making an effort to predict the risk level of the huge datasets of patients.

3.IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP

It’s easy to jump straight into value proposition design. That is the core of your business and where the revenue or exchange of value will come from. However, trying to provide value to a misunderstood customer is very risky business. Do you have your blinkers on? Try using this canvas before you design your value proposition to make sure your offer nails exactly what your customer wants, needs, or may pleasantly surprise them! Keep asking yourself “why would they care?”. What problem are you solving? What opportunity are you creating? In this empathy map what customer think and feels. this map shows the pain and gain of the customer and what do their hear about the problem. this is the easy way to understand the problem statement.


Which food is best?
What is the correct rate of blood pressure?
Is this expensive?
Can the disease be cured?
What is the time period for this treatment?
What is the best prediction?
Says
Patient
Thinks
Does
Feels
Do exercise
Draw food schedule
Do visit doctor regularly
Nervous
Excitement
Fear

3.2 IDEATION & BRAINSTORMING

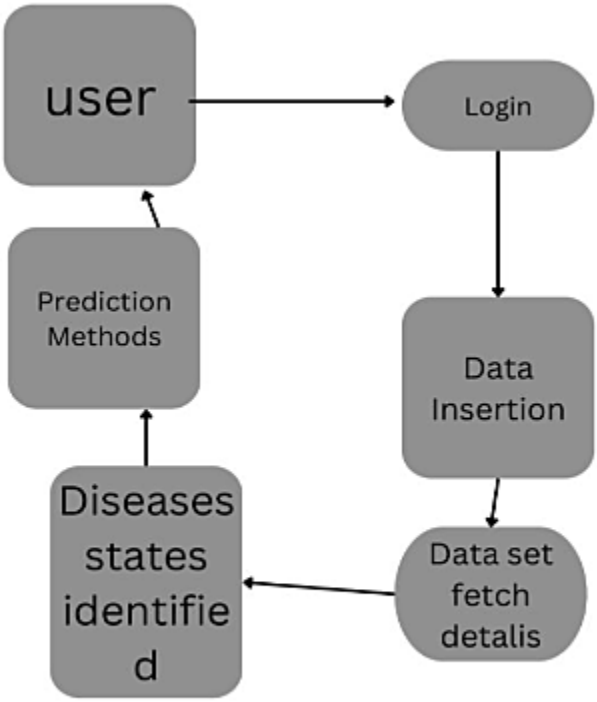
Brainstorming is a method of generating ideas and sharing knowledge to solve a particular commercial or technical problem, in which participants are encouraged to think without interruption. Brainstorming is a group activity where each participant shares their ideas as soon as they come to mind.

3.3 PROPOSED SOLUTION

To over come this we are implementing regression in order to achieve accurate results in less time . Machine learning is given a major priority in modern life in many application and in healthcare sector . Prediction is one of areas where machine learning plays a vital role , Our topic is to predict heart diseases by processing patient's dataset and a data of patients i.e., user of whom we need to predict the chances of occurrence of the heart diseases.A csv file is given as input . After the successful completion of operation the result is predicted and displayed.

4.PROJECT DESIGN

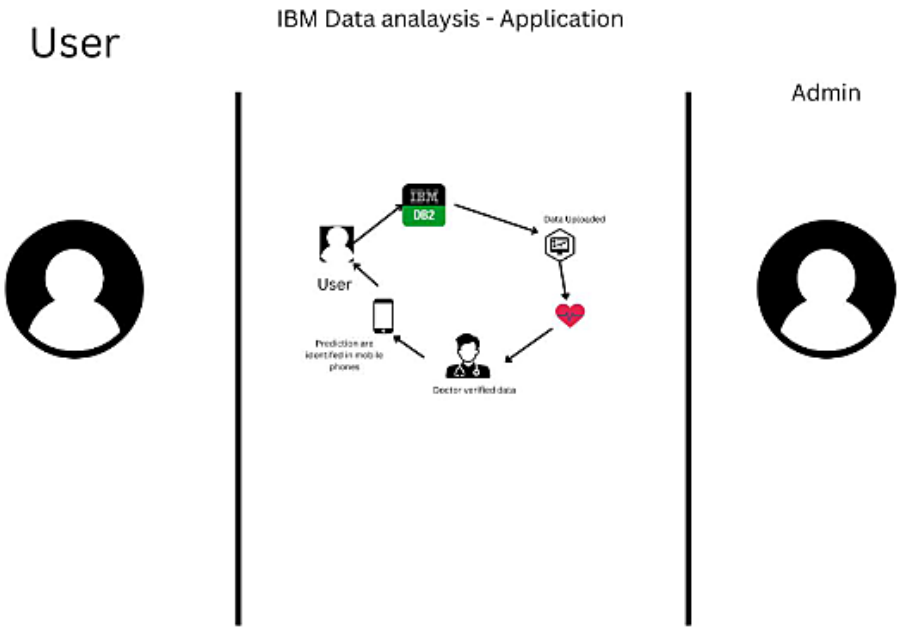
4.1 DATA FLOW DIAGRAM



A data flow diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement.

4.2 SOLUTION AND TECHNICAL ARCHITECTURE

A solution architecture (SA) is an architectural description of a specific solution. SAs combine guidance from different enterprise architecture viewpoints (business, information and technical), as well as from the enterprise solution architecture (ESA).



4.3 USER STORIES

User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	login		Can login and enter the data to know the details	Can receive the confirmation	high	Sprint-1
Patient details	View in data set		As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
Dataset insertion			As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
Disease are verified			As a user, I can register for the application through Gmail		Medium	Sprint-1
predictied			As a user, I can log into the application by entering email & password		High	Sprint-1

5.PROJECT PLANNING & SCHEDULING

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring technical papers,research publications etc.	26 September 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	19 September 2022
Ideation	List the ideas by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	19 September 2022
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	1 October 2022
Problem Solution Fit	Prepare problem - solution fit document.	1 October 2022
Solution Architecture	Prepare a solution architecture document.	22 October 2022

Data Flow Diagrams	Draw the data flow diagrams and submitfor review.	15 OCTOBER 2022
Technology Architecture	Architecture diagram.	16 OCTOBER 2022
Prepare Milestone & Activity List	Prepare the milestones &activity list of the project.	24 OCTOBER 2022
Project Development - Delivery of Sprint-1, 2, 3 & 4	Develop & submit thedeveloped code by testing it.	5 NOVEMBER 2022

6.CODING & SOLUTIONING

Importing Necessary Libraries

```
In [3]:
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
```

Loading Data

```
In [71]:
data=pd.read_csv("C:/Users/admin/Downloads/Heart_Disease_Prediction (1).csv")
data.head()
```

Out[71]:

	Age	Sex	Chest pain type	BP	Cholesterol	FBS over 120	EKG results	Max HR	Exercise angina	ST depression	Slope of ST	Number of vessels fluro	Thallium	Heart Disease
0	70	1	4	130	322	0	2	109	0	2.4	2	3	3	Presence
1	67	0	3	115	564	0	2	160	0	1.6	2	0	7	Absence
2	57	1	2	124	261	0	0	141	0	0.3	1	0	7	Presence
3	64	1	4	128	263	0	0	105	1	0.2	2	1	7	Absence
4	74	0	2	120	269	0	2	121	1	0.2	1	1	3	Absence

```
In [5]:
data.info()
```

```
<class
'pandas.core
.frame.DataF
rame'>
RangeIndex:
270 entries,
0 to 269
Data columns
(total 14
columns):
 #   Column                    Non-Null Count Dtype
---  ------                    -------------- -----
 0   Age                       270 non-       int64
                               null
 1   Sex                       270 non-       int64
                               null
 2   Chest pain type           270 non-       int64
                               null
 3   BP                        270 non-       int64
                               null
 4   Cholesterol               270 non-       int64
                               null
 5   FBS over 120              270 non-       int64
                               null
 6   EKG results               270 non-       int64
                               null
 7   Max HR                    270 non-       int64
                               null
 8   Exercise angina           270 non-       int64
                               null
 9   ST depression             270 non-       float
```

```
                                7   null         64
                                0
  1    Slope of ST              2   non-         int64
  0                             7   null
                                0
  1    Number of vessels        2   non-         int64
  1    fluro                    7   null
                                0
  1    Thallium                 2   non-         int64
  2                             7   null
                                0
  1    Heart Disease            2   non-         objec
  3                             7   null         t
                                0
dtypes:
float64(1),
int64(12),
object(1)
memory usage:
29.7+ KB
```

```
In [8]: data.describe()
```

Out[8]:

	Age	Sex	Chest pain		FBS over	EKG	Exercise	ST	Slope of

Exploratory Data Analysis

```
In [7]: data.shape
Out[7]: (270, 14)
```

		type	BP	Cholesterol	120	results	Max HR angina	depression	ST	2
count	270.000000 270.000000		270.000000 270.000000	270.000000 270.000000		270.000000	270.000000	270.000000	270.000000	
mean	54.433333	0.677 778	3.174074 131.344444	249.659259	0.14 8148	1.0222 22	149.677778	0.32 9630	1.05000	1.5851 85
std	9.109067	0.468 195	0.950090 17.8616 08	51.686237	0.355 906	0.9978 91	23.165717	0.470 952	1.14521	0.6143 90
min	29.000000	0.000 000	1.000000 94.0000 00	126.000000	0.00 0000	0.0000 00	71.000000	0.000 000	0.00000	1.0000 00
25%	48.000000	0.000 000	3.000000 120.000000	213.000000	0.00 0000	0.0000 00	133.000000	0.00 0000	0.00000	1.0000 00
50%	**55.000000 0.000000**	**1.000000**	**0.80000**	**3.000000 2.000000**	**130.000000**	**245.000000**	**0.000000**	**2.000000**	**153.500000**	
75%	**61.0000 00**	1.000 000	4.000000 140.000000	280.0000 00	0.000 000	2.0000 00	166.0000 00	1.0000 00	1.60 000	2.0000 00

max	**77.000000**	1.000000	4.000000	200.000000	564.000000	1.000000	2.000000	202.000000	1.000000	6.200000	3.000000

In [9]:
```
data.isnull().sum()
```

Out[9]:
```
Age                       0
Sex                       0
Chest pain type           0
BP                        0
Cholesterol               0
FBS over 120              0
EKG results               0
Max HR                    0
Exercise angina           0
ST depression             0
Slope of ST               0
Number of vessels fluro   0
Thallium                  0
Heart Disease             0
dtype: int64
```

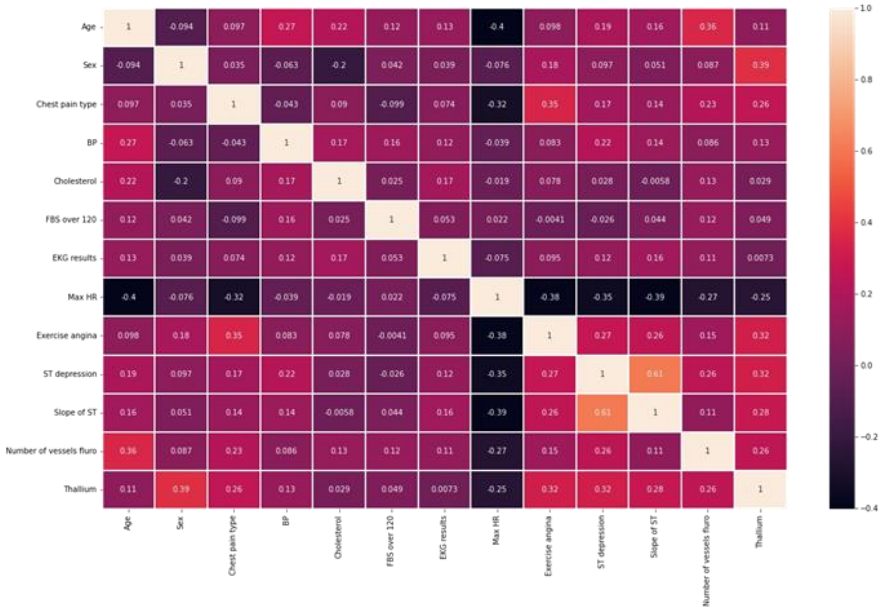
In [10]:
```
data['Heart Disease'].value_counts()
```

Out[10]:
```
Absence     150
Presence    120
Name: Heart Disease, dtype: int64
```

Correlation Matrix

In [27]:
```
plt.figure(figsize=(20,12))
sns.heatmap(data.corr(),annot=True,linewidth =2)
```

Out[37]:
```
<AxesSubplot:>
```

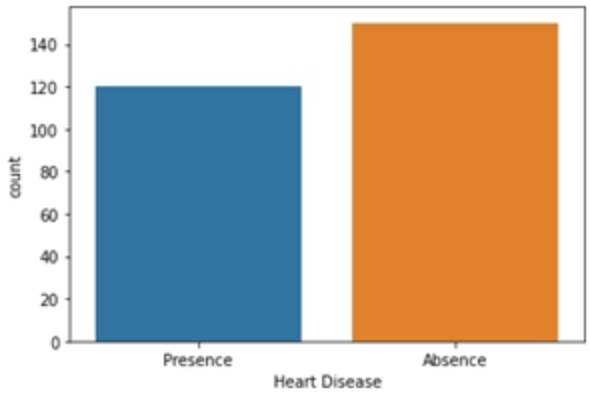


Data Visualization

In [22]:
```
sns.countplot(data['Heart Disease'])
```

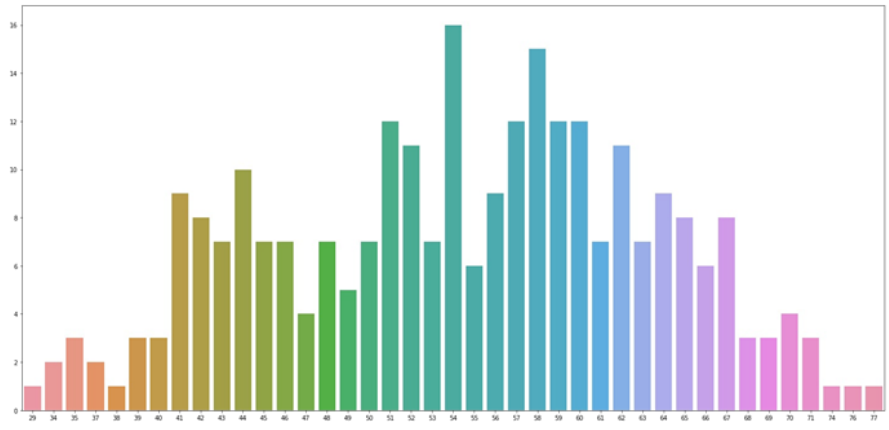
```
C:\Users\admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable
as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other argu
ments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[22]: <AxesSubplot:xlabel='Heart Disease', ylabel='count'>



In [40]:
```
plt.figure(figsize=(25,12))
sns.barplot(x=data.Age.value_counts().index,y=data.Age.value_counts().values)
```

Out[40]: <AxesSubplot:>



In [42]:
```
minAge=min(data.Age)
maxAge=max(data.Age)
meanAge=data.Age.mean()
print('Min Age :',minAge)
print('Max Age :',maxAge)
print('Mean Age :',meanAge)
```

MinAge:29
MaxAge:77

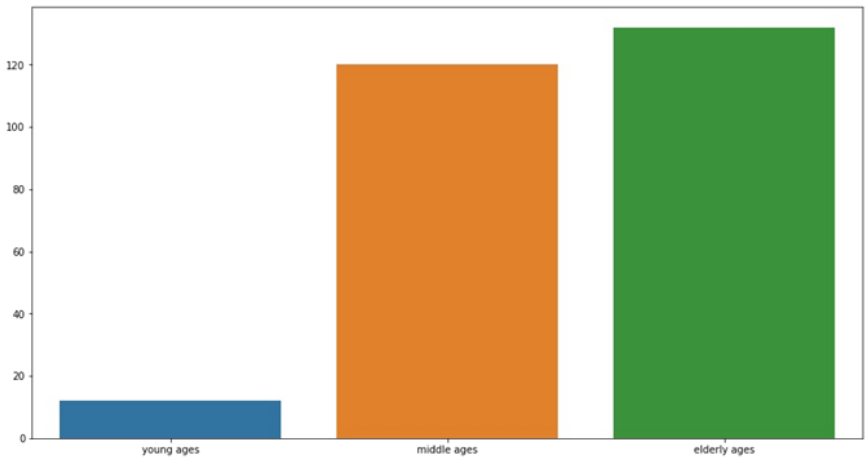
Mean Age : 54.43333333333333

In [49]:
```
Young = data[(data.Age>=29)&(data.Age<40)]
```

Out[53]:


```
Middle = data[(data.Age>=40)&(data.Age<55)]
Elder = data[(data.Age>55)]
plt.figure(figsize=(15,8))
sns.barplot(x=['young ages','middle ages','elderly ages'],y=[len(Young),len(Middle),len(Elder)])
```

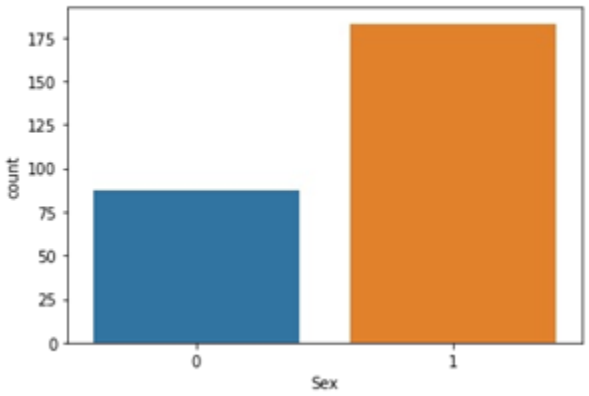
Out[49]: <AxesSubplot:>



In [24]:
```
sns.countplot(data.Sex)
```

```
C:\Users\admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable
as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other argu
ments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[24]: <AxesSubplot:xlabel='Sex', ylabel='count'>



In [53]:
```
sns.countplot(data['Chest pain type'])
```

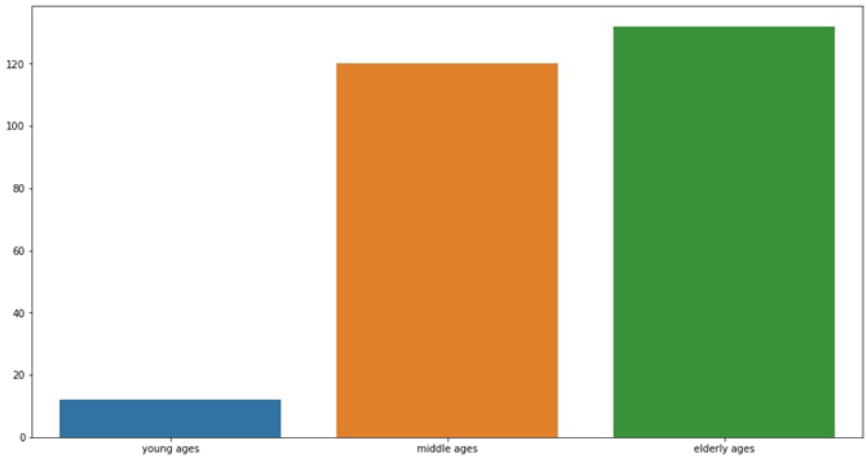
```
C:\Users\admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable
as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other argu
ments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Chest pain type', ylabel='count'>
```

Out[53]:


```
Middle = data[(data.Age>=40)&(data.Age<55)]
Elder = data[(data.Age>55)]
plt.figure(figsize=(15,8))
sns.barplot(x=['young ages','middle ages','elderly ages'],y=[len(Young),len(Middle),len(Elder)])
```

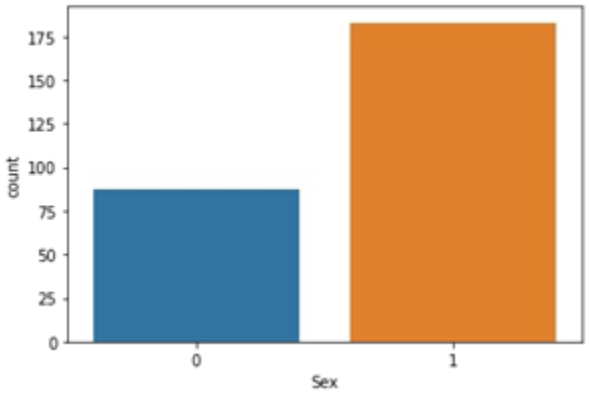
Out[49]:
```
<AxesSubplot:>
```



In [24]:
```
sns.countplot(data.Sex)
```

```
C:\Users\admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable
as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other argu
ments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[24]:
```
<AxesSubplot:xlabel='Sex', ylabel='count'>
```



In [53]:
```
sns.countplot(data['Chest pain type'])
```

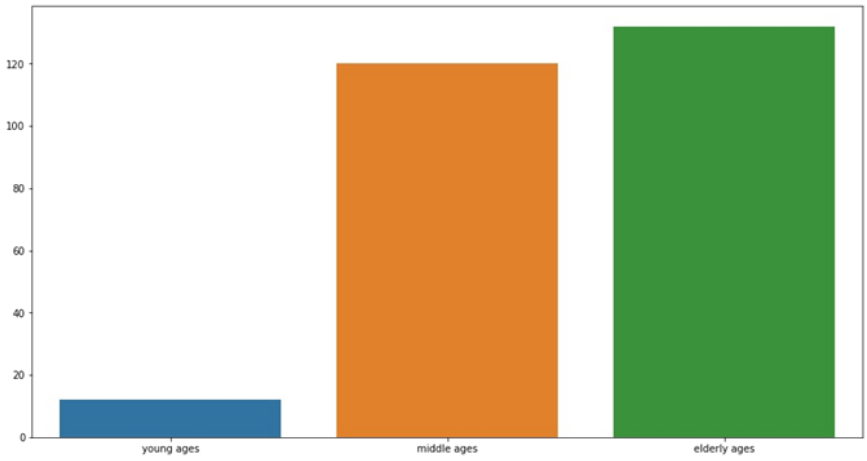
```
C:\Users\admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable
as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other argu
ments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Chest pain type', ylabel='count'>
```

Out[53]:


```
Middle = data[(data.Age>=40)&(data.Age<55)]
Elder = data[(data.Age>55)]
plt.figure(figsize=(15,8))
sns.barplot(x=['young ages','middle ages','elderly ages'],y=[len(Young),len(Middle),len(Elder)])
```

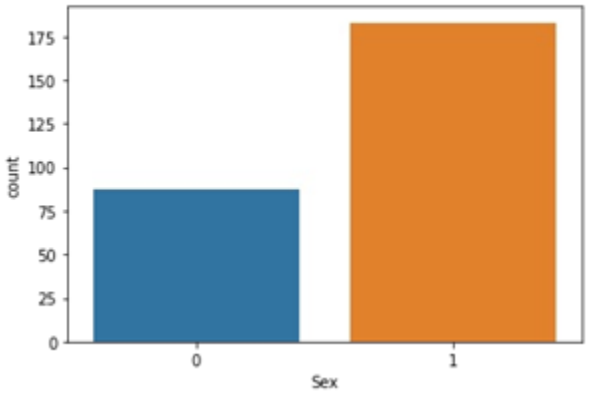
Out[49]:
```
<AxesSubplot:>
```



In [24]:
```
sns.countplot(data.Sex)
```

```
C:\Users\admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable
as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other argu
ments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[24]:
```
<AxesSubplot:xlabel='Sex', ylabel='count'>
```



In [53]:
```
sns.countplot(data['Chest pain type'])
```

```
C:\Users\admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable
as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other argu
ments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

<AxesSubplot:xlabel='Chest pain type', ylabel='count'>

In [98]:
```
model.fit(train_x,train_y)
```

Out[98]:
```
LogisticRegression()
```

In [99]:
```
pred_y=model.predict(test_x)
pred_y
```

Out[99]:
```
array(['Presence', 'Absence', 'Absence', 'Absence', 'Absence', 'Presence',
       'Absence', 'Absence', 'Absence', 'Absence', 'Absence', 'Presence',
       'Presence', 'Presence', 'Absence', 'Absence', 'Absence', 'Absence',
       'Presence', 'Absence', 'Presence', 'Absence', 'Absence', 'Absence',
       'Absence', 'Presence', 'Presence', 'Presence', 'Absence',
       'Absence', 'Absence', 'Absence', 'Presence', 'Absence', 'Absence',
       'Presence', 'Presence', 'Absence', 'Presence', 'Absence',
       'Absence', 'Absence', 'Absence', 'Absence', 'Absence', 'Absence',
       'Absence', 'Presence', 'Presence', 'Absence', 'Absence', 'Absence',
       'Absence', 'Presence', 'Absence', 'Presence', 'Absence',
       'Presence', 'Absence', 'Absence', 'Presence', 'Presence',
       'Presence', 'Presence', 'Presence', 'Absence', 'Absence',
       'Absence', 'Absence', 'Absence', 'Absence', 'Absence', 'Absence',
       'Absence', 'Absence', 'Absence', 'Absence', 'Presence', 'Absence',
       'Absence', 'Absence'], dtype=object)
```

In [100]:
```
model.score(test_x,test_y)
```

Out[100…
```
0.8395061728395061
```

Confusion Matrix

In [111…
```
from sklearn.metrics import confusion_matrix
```

In [112…
```
confusion_matrix(test_y,pred_y)
```

Out[112…
```
array([[46,  3],
       [10, 22]], dtype=int64)
```

In [ ]:

7.TESTING

7.1 TEST CASES

A test case is nothing but a series of step executed on a product, using a predefined set of input data, expected to produce a pre-defined set of outputs, in a given environment. It describes “how” to implement those test cases. Test case specifications are useful as it enlists the specification details of the items. The purpose of testing is to discover errors . Testing is the process of trying to discover every conceivable fault or weakness in a work product . It provide a way to check the functionality of component , sub assemblies , assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the software system meets its requirement and user expectation and does not fail in an unacceptable manner. There are various types of testing. Each test type addressing a specific testing requirement.
The testing report are submitted in github account.

7.2 USER ACCEPTANCE TESTING

User acceptance testing is a critical phase of any project and requires significant participant by the end user. It also ensure that the system meets the functional requirement.

8. RESULTS

8.1 PERFORMANCE METRICS

Confusion Matrix

```
In [111…
from sklearn.metrics import confusion_matrix

In [112…
confusion_matrix(test_y,pred_y)

Out[112…
array([[46,  3],
       [10, 22]], dtype=int64)
```

Accuracy 83%

9.ADVANTAGE & DISADVANTAGE

ADVANTAGE

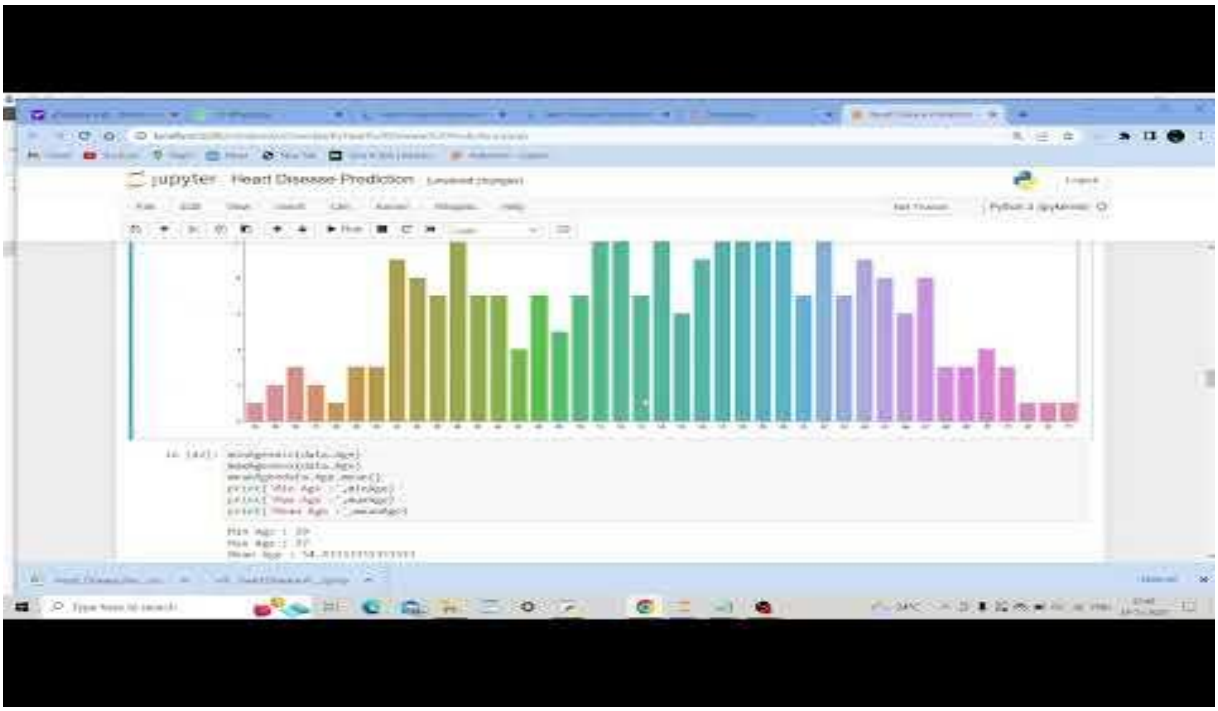
1.The advantage of this model are high performance and accuracy rate.
2.It is very flexible and high rates of success are achieved.
3.The application when implemented using random forests has more accuracy rate when compare to other algorithm. In this system, we achieve around 83%.

10. FUTURE SCOPE

In the future, various other metrics like throughput, average time, resources utilizing, waiting time, etc. can be considered. In the future, author will work to optimize the cloud resources further and enhance cloud-based application performance, such as considering more SLA (service level agreement) parameters. For example, the algorithm will be tested based on the number of violation and the migration count for better performance. Also, the algorithm will be comprehensively compared to other existing algorithm in the literature.

PROJECT DEMO LINK

heart prediction


jupyter Heart Disease Prediction