CODING &SOLUTIONING

Date	06 November 2022
Team ID	PNT2022TMID53213
Project Name	Visualizing and Predicting Heart Diseases with an Interactive Dash Board
Marks	10 marks

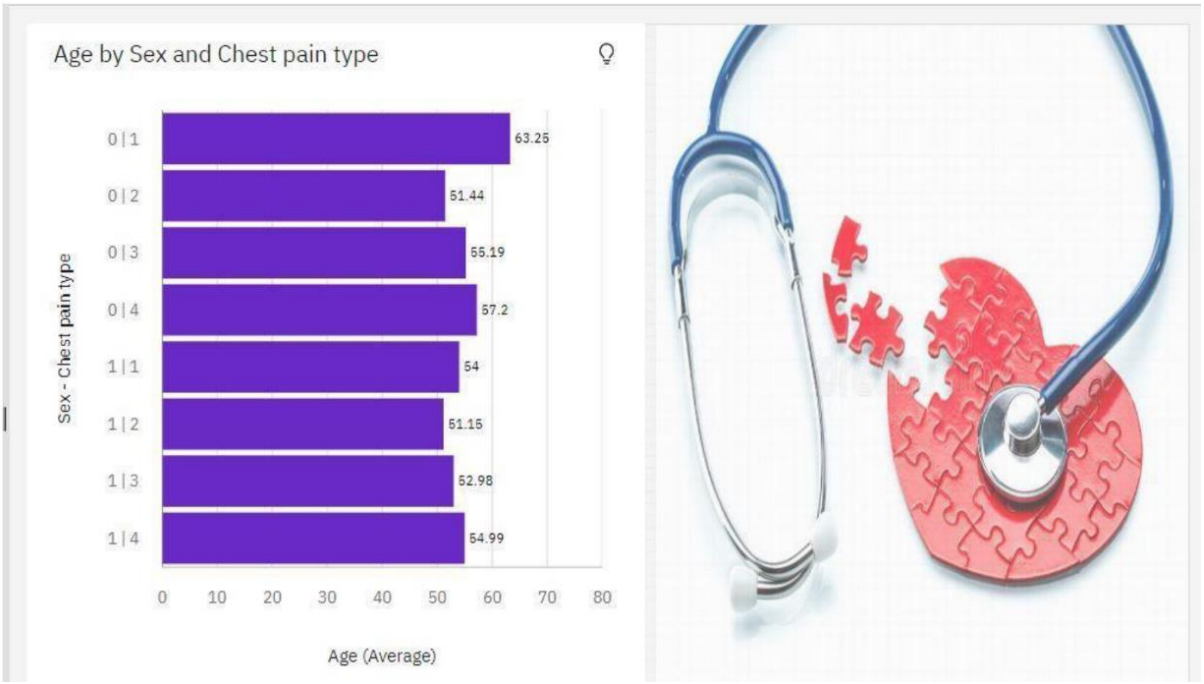
Features Implemented:

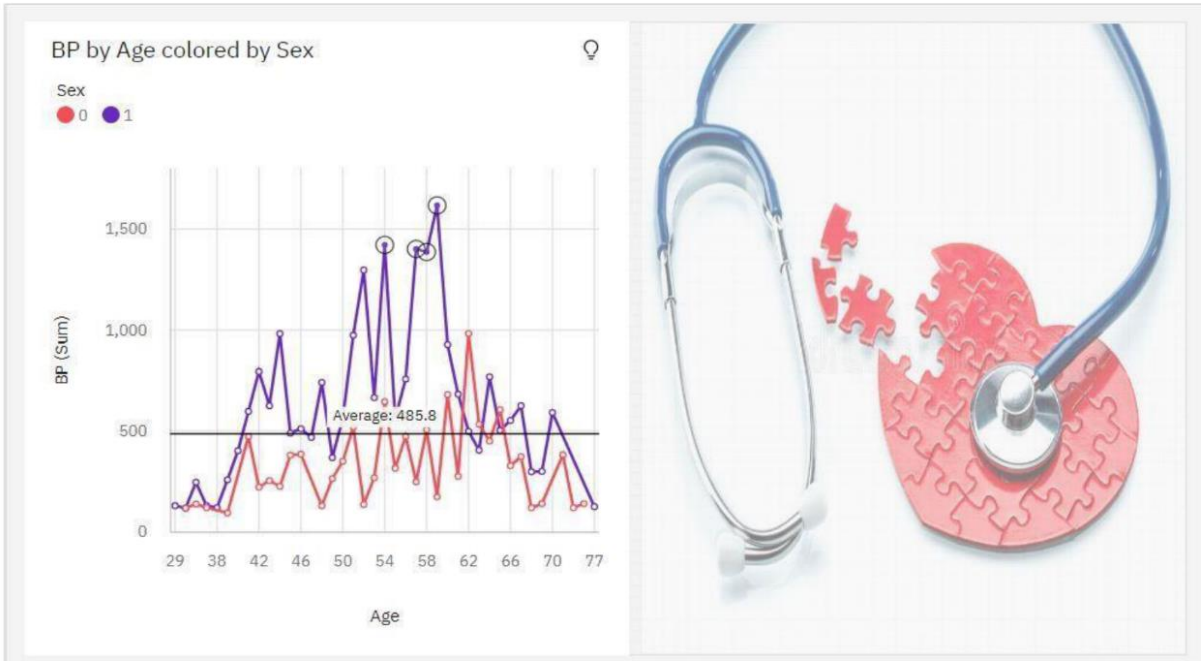
Visualizing various features of the data

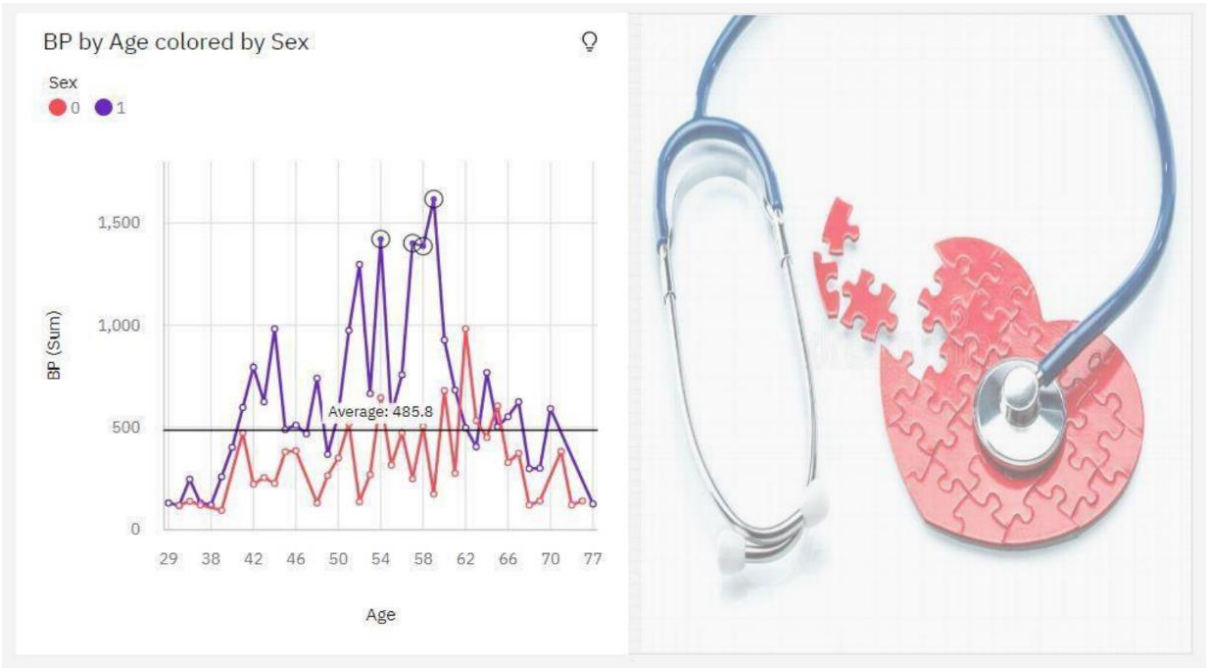
User Interactive Dashboard

User can predict whether they have heart disease or not

They can visualize and see in which category they falls




BP by Age colored by Sex
Sex
0
1
1,500
1,000
500
0
BP (Sum)
Average: 485.8
29
38
42
46
50
54
58
62
66
70
77
Age


BP by Age colored by Sex
Sex
0
1
1,500
1,000
500
0
BP (Sum)
Average: 485.8
29
38
42
46
50
54
58
62
66
70
77
Age

Max HR by heart disease and Exercise angina

VISUALIZATION DASHBOARD

Different machine learning algorithms are have been implemented

(i) Using Logistic Regression algorithm

Logistic regression is an example of supervised learning. It is used to calculate or predict the

probability of a binary (yes/no) event occurring.

```
[69] y=df['Heart Disease']
     x=df.drop('Heart Disease',axis=1)
```

```
X_train,X_test,Y_train,Y_test=train_test_split(x,y,test_size=0.20)
model=LogisticRegression()
model.fit(X_train,Y_train)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:818: ConvergenceWarning:

lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

LogisticRegression()
```

```
y=model.predict(X_test)
ya=np.array(Y_test)
ya
```

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

Accuracy score of Logistic Regression:

```
[ ]  scLR=model.score(X_test,Y_test)
     scLR
```

```
0.8333333333333334
```

(ii)Using Support vector machine algorithm

SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future.

Accuracy score of SVM:

```
[ ]  model1=SVC()
     model1.fit(X_train,Y_train)
     sc=model1.score(X_test,Y_test)
     sc
```

```
0.6481481481481481
```

(iii)Using Random Forest algorithm

Random forest is a supervised machine learning algorithm that is used widely in classification and regression problem. It builds decision tree on different samples and takes their majority vote for classification and average in case of regression

Accuracy score of Random forest

```
model2=RandomForestClassifier()
model2.fit(X_train,Y_train)
scrfc=model2.score(X_test,Y_test)
scrfc
```

```
0.8703703703703703
```

(iv)Using K Nearest Neighbors alogithm:

This algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K-NN algorithm

Accuracy score of KNN algorithm

```
[ ] model3=KNeighborsClassifier()
    model3.fit(X_train,Y_train)
    sckn=model3.score(X_test,Y_test)
    sckn
```

```
0.6111111111111112
```

(v)Using Gaussian Naïve Bayes algorithm

Gaussian naïve bayes algorithm based in applying bayes's theorem with strong independence assumptions.

Accuracy score of GaussianNB

```
model4=GaussianNB()
model4.fit(X_train,Y_train)
scnb=model4.score(X_test,Y_test)
scnb
```

```
0.8888888888888888
```

(vi)Using Linear Discriminant Analysis

Linear discriminant analysis is a dimensionality reduction technique that is commonly used for supervised classification problems. It is used for modelling difference in groups. i.e separating two or more classes.

Accuracy score of Linear Discriminant analysis

```
[ ]  model5=LinearDiscriminantAnalysis()
     model5.fit(X_train,Y_train)
     sclda=model5.score(X_test,Y_test)
     sclda

     0.8518518518518519
```

(vii)Using Ada Boosting algorithm

Ada Boosting is an ensemble learning method which was initially created to increase the efficiency of binary classifiers.

Accuracy score of Ada Boosting

```
[ ]  model6=AdaBoostClassifier()
     model6.fit(X_train,Y_train)
     scabc=model6.score(X_test,Y_test)
     scabc

     0.8333333333333334
```

10)The dataset is tested with all above models out of which random forest has given the good

accuracy 87% and this it is better than all the model for this dataset.

1.import the Random forest package from ensemble

```
[ ] from sklearn.ensemble import RandomForestClassifier

[ ] rf=RandomForestClassifier()

[ ] rf.fit(X_train,Y_train)

    RandomForestClassifier()

[ ] y_pred5=rf.predict(X_test)
```

2.Create a data for testing

```
[ ] new_data=pd.DataFrame({
        'Age':70,
        'Sex':1,
        'Chest pain type':4,
        'BP':130,
        'Cholesterol':322,
        'FBS over 120':0,
        'EKG results':2,
        'Max HR':109,
        'Exercise angina':0,
        'ST depression':2.4,
        'Slope of ST':2,
        'Number of vessels fluro':3,
        'Thallium':3,
    },index=[0])
```

```
[ ] new_data
```

	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
0	70	1	4	130	322	0	2	109	0	2.4	2	3	3

3. The above data is specifically tested to check the different testcases

```
p =rf.predict(new_data)
if p[0]==0:
  print("No disease")
else:
  print("disease")
```

```
disease
```

4.No Disease Prediction

```
new_data1=pd.DataFrame({
    'Age':57,
    'Sex':1,
    'Chest pain type':4,
    'BP':140,
    'Cholesterol':192,
    'FBS over 120':0,
    'EKG results':0,
    'Max HR':148,
    'Exercise angina':0,
    'ST depression':0.4,
    'Slope of ST':2,
    'Number of vessels fluro':0,
    'Thallium':6,
},index=[0])
```

```
[44] new_data1
```

	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
0	67	0	3	115	564	0	2	160	0	1.6	2	0	7

```
p1 =rf.predict(new_data1)
if p1[0]==0:
  print("disease")
else:
  print("No Disease")
```

```
No Disease
```

Implementation

Spyder (Python 3.9)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\SSN

C:\Users\SSN\Documents\SEMESTER 7\IBM\backend.py

temp.py | backend.py - IBM | frontpage.html - IBM\templates | backend.py - SEMESTER 7\...\Medical-Diagnosis-Web-App-Using-ML-master | frontpage.html - SEMESTER

```python
from flask import Flask,render_template,request
import numpy as np
import pickle

app = Flask(__name__)
@app.route('/')

def front_page():
    return render_template('frontpage.html')
       #project-id_PNT2022TMID53213

@app.route('/hearttt',methods=['POST'])
def heartt():
    
        age = request.form['age']
        sex = request.form['sex']
        chest = request.form['chest']
        trestbps = request.form['trestbps']
        chol = request.form['chol']
        fbs = request.form['fbs']
        restecg = request.form['restecg']
        thalach = request.form['thalach']
        exang = request.form['exang']
        oldpeak = request.form['oldpeak']
        slope = request.form['slope']
        ca = request.form['ca']
        thal = request.form['thal']
        #project-id_PNT2022TMID53213

        model2 = pickle.load(open('./static/heart_model.pkl','rb'))
        input_data = [age,sex,chest,trestbps,chol,fbs,restecg,thalach,exang,oldpeak,slope,ca,thal]
        for i in range(len(input_data)):
            input_data[i]=float(input_data[i])
        print(input_data)
        input_data_as_numpy_array= np.asarray(input_data)
        input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)
        prediction = model2.predict(input_data_reshaped)
        senddata=""
        if (prediction[0]== 0):
            print("According to the given details person does not have Heart Disease")
            senddata='According to the given details person does not have Heart Disease'
            print(senddata)
        else:
            print("According to the given details person does not have Heart Disease")
            senddata='According to the given details chances of having Heart Disease are High, So Please Consult a Doctor'

        return render_template('result.html',resultvalue=senddata)
if __name__ == '__main__':
    app.run()
```

Spyder (Python 3.9)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\SSN

C:\Users\SSN\Documents\SEMESTER 7\IBM\templates\frontpage.html

temp.py | backend.py - IBM | frontpage.html - IBM\templates | backend.py - SEMESTER 7\...\Medical-Diagnosis-Web-App-Using-ML-master | frontpage.html - SEMESTER 7\...\templates

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8" />
    <title>Health Diagnosis ML</title>
    <link
      rel="stylesheet"
      href="{{ url_for('static',filename = 'style.css')}}"
    />
  </head>
  <body>
    <header>
      <h1 style=margin-left:"100px">Health Disease Prediction</h1>

    </header>
    <div class="frontpageclass">
      <h2>Welcome!</h2>

      <h2>About us</h2>
      <p>
        The major challenge in heart disease is its detection. There are instruments available which can
predict heart disease but either they are expensive or are not efficient to calculate chance of heart
disease in human. Early detection of cardiac diseases can decrease the mortality rate and overall
complications. However, it is not possible to monitor patients every day in all cases accurately and
consultation of a patient for 24 hours by a doctor is not available since it requires more sapience,
time and expertise.So here is the solution you can predict disease by this web easily by entering the details
      </p>

      <h2>Heart Disease</h2>


      <h3>Overview</h3>
      <p>
        Heart disease describes a range of conditions that affect your heart.
        Diseases under the heart disease umbrella include blood vessel diseases,
        such as coronary artery disease; heart rhythm problems (arrhythmias);
        and heart defects you're born with (congenital heart defects), among
```

```
            <input type="text" name="age" /><br />
            <label>Sex(1:Male,0:Female) :</label>
            <input type="text" name="sex" /><br />
            <label>Chest Pain Type :</label>
            <input type="text" name="chest" /><br />
            <label>BP :</label>
            <input type="text" name="trestbps" /><br />
            <label>Serum cholestoral in mg/dl :</label>
            <input type="text" name="chol" /><br />
            <label>Fbs :</label>
            <input type="text" name="fbs" /><br />

            <label>EKG :</label>
            <input type="text" name="restecg" /><br />
            <label>Max hr :</label>
            <input type="text" name="thalach" /><br />
            <label>Exercise angina :</label>
            <input type="text" name="exang" /><br />
            <label>St depression :</label>
            <input type="text" name="oldpeak" /><br />
            <label>Slope of st:</label>
            <input type="text" name="slope" /><br />
            <label>Number of vessels fluro :</label>
            <input type="text" name="ca" /><br />
            <label>Thallium :</label>
            <input type="text" name="thal" /><br />
                #project-id_PNT2022TMID53213
            <input type="submit" />
          </div>
        </form>

      </div>

    </body>
</html>
```

C:\Users\SSN\Documents\SEMESTER 7\IBM\templates\result.html

age.html - IBM\templates | backend.py - SEMESTER 7\...\Medical-Diagnosis-Web-App-Using-ML-master | frontpage.html - SEMESTER 7\...\templates | result.html

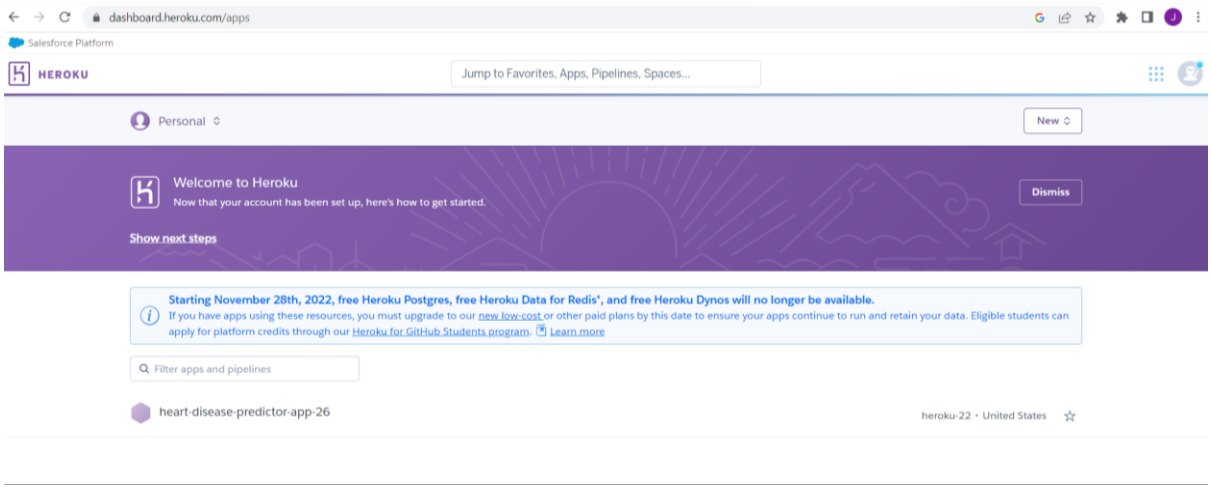
```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8" />
    <title>Heart Disease Detection Website</title>
    <link
      rel="stylesheet"
      href="{{ url_for('static',filename = 'style.css')}}"
    />
  </head>
  <body>
    <header>
      <h1>Home Page</h1>

    </header>
    <div class="resultclass">{{resultvalue}}</div>

  </body>
</html>

```

Cloud deployment



Webapp deployed in Heroku

Health Disease Prediction

Welcome!

About us

The major challenge in heart disease is its detection. There are instruments available which can predict heart disease but either they are expensive or are not efficient to calculate chance of heart disease in human. Early detection of cardiac diseases can decrease the mortality rate and overall complications. However, it is not possible to monitor patients every day in all cases accurately and consultation of a patient for 24 hours by a doctor is not available since it requires more sapience, time and expertise.So here is the solution you can predict disease by this web easily by entering the details

Heart Disease

Overview

Heart disease describes a range of conditions that affect your heart. Diseases under the heart disease umbrella include blood vessel diseases, such as coronary artery disease; heart rhythm problems (arrhythmias); and heart defects you're born with (congenital heart defects), among others. The term "heart disease" is often used interchangeably with the term "cardiovascular disease." Cardiovascular disease generally refers to conditions that involve narrowed or blocked blood vessels that can lead to a heart attack, chest pain (angina) or stroke. Other heart conditions, such as those that affect your heart's muscle, valves or rhythm, also are considered forms of heart disease. Many forms of heart disease can be prevented or treated with healthy lifestyle choices.

Symptoms

- Chest pain, chest tightness, chest pressure and chest discomfort (angina)
- Shortness of breath
- Pain, numbness, weakness or coldness in your legs or arms if the blood vessels in those parts of your body are narrowed
- Pain in the neck, jaw, throat, upper abdomen or back

To check whether you are having heart disease or not use the below link

Click here to visualize in the dashboard">

Predict whether you are having disease or not

Age :
Sex(1:Male,0:Female) :
Chest Pain Type :
BP :
Serum cholestoral in mg/dl :
Fbs :
EKG :
Max hr :
Exercise angina :
St depression :
Slope of st:
Number of vessels fluro :
Thallium :
#project-id_PNT2022TMID53213 Submit

Debugging & Traceability:

Each Feature is handled and any exceptions are reported with a user interface. Exception handling involves handling the unexpected situation and thus performing actions