heart2svm

April 22, 2023

Heart Disease Prediction Machine Learning Project with the implentation of the Support Vector Machine learning model.

Importing all the neccessary libraries and packages

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from pandas import *
from numpy import *
from matplotlib.pyplot import *
```

Reading the data and creating the dataframe

[3]:
```
data=read_csv("Heart_Disease_Prediction.csv")
data.head()
```

[3]:
```
   Age  Sex  Chest pain type   BP  Cholesterol  FBS over 120  EKG results  \
0   70    1                4  130          322             0            2
1   67    0                3  115          564             0            2
2   57    1                2  124          261             0            0
3   64    1                4  128          263             0            0
4   74    0                2  120          269             0            2

   Max HR  Exercise angina  ST depression  Slope of ST  \
0     109                0            2.4            2
1     160                0            1.6            2
2     141                0            0.3            1
3     105                1            0.2            2
4     121                1            0.2            1

   Number of vessels fluro  Thallium Heart Disease
0                        3         3      Presence
1                        0         7       Absence
2                        0         7      Presence
3                        1         7       Absence
4                        1         3       Absence
```

Checking the statistics of the dataset

[4]:
```
data.describe()
```

[4]:
```
              Age         Sex  Chest pain type          BP  Cholesterol  \
count  270.000000  270.000000       270.000000  270.000000   270.000000
mean    54.433333    0.677778         3.174074  131.344444   249.659259
std      9.109067    0.468195         0.950090   17.861608    51.686237
min     29.000000    0.000000         1.000000   94.000000   126.000000
25%     48.000000    0.000000         3.000000  120.000000   213.000000
50%     55.000000    1.000000         3.000000  130.000000   245.000000
75%     61.000000    1.000000         4.000000  140.000000   280.000000
max     77.000000    1.000000         4.000000  200.000000   564.000000

       FBS over 120  EKG results      Max HR  Exercise angina  ST depression  \
count    270.000000   270.000000  270.000000       270.000000      270.00000
mean       0.148148     1.022222  149.677778         0.329630        1.05000
std        0.355906     0.997891   23.165717         0.470952        1.14521
min        0.000000     0.000000   71.000000         0.000000        0.00000
25%        0.000000     0.000000  133.000000         0.000000        0.00000
50%        0.000000     2.000000  153.500000         0.000000        0.80000
75%        0.000000     2.000000  166.000000         1.000000        1.60000
max        1.000000     2.000000  202.000000         1.000000        6.20000

       Slope of ST  Number of vessels fluro    Thallium
count   270.000000               270.000000  270.000000
mean      1.585185                 0.670370    4.696296
std       0.614390                 0.943896    1.940659
min       1.000000                 0.000000    3.000000
25%       1.000000                 0.000000    3.000000
50%       2.000000                 0.000000    3.000000
75%       2.000000                 1.000000    7.000000
max       3.000000                 3.000000    7.000000
```

Checking if there is any missing value/ blank data in the dataset.. In case of null values we have to clean and filter the data

[5]:
```
data.isnull().sum()
```

[5]:
```
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
```

Checking the distribution of the target variable

```
[6]: data["Heart Disease"].value_counts()
```

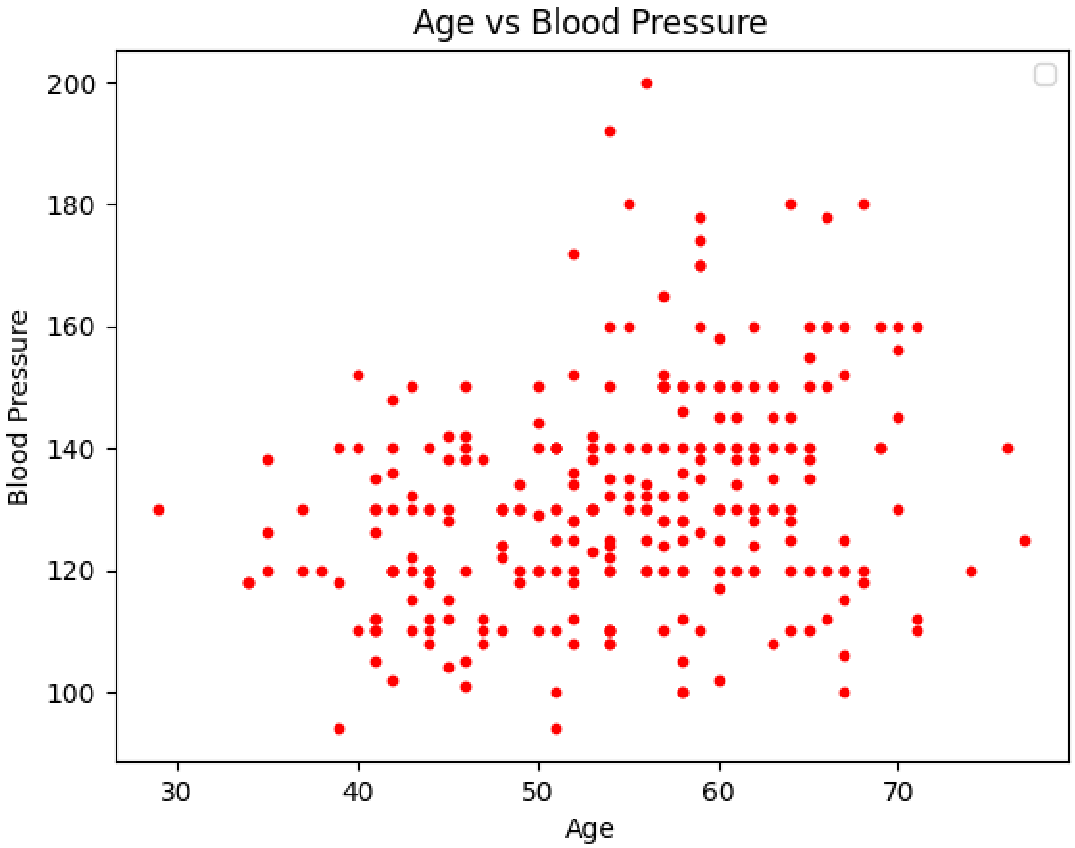
```
[6]: Absence     150
     Presence    120
     Name: Heart Disease, dtype: int64
```

Plotting the Age vs Blood Pressure Graph

```
[21]: plt.figure()
      plt.scatter(data["Age"],data["BP"],c="red",s=10)
      plt.title("Age vs Blood Pressure")
      plt.xlabel("Age")
      plt.ylabel("Blood Pressure")
      plt.legend()
```

```
No artists with labels found to put in legend.  Note that artists whose label
start with an underscore are ignored when legend() is called with no argument.
```

```
[21]: <matplotlib.legend.Legend at 0x20a42df67d0>
```



Separating the independent and the dependent variable ie spliting the dataframe into target dataframe and features dataframe

[8]:
```
X=data.drop("Heart Disease",axis=1) #this will have the independent features␣
  ↪variable
X
```

[8]:
```
     Age  Sex  Chest pain type   BP  Cholesterol  FBS over 120  EKG results  \
0     70    1                4  130          322             0            2
1     67    0                3  115          564             0            2
2     57    1                2  124          261             0            0
3     64    1                4  128          263             0            0
4     74    0                2  120          269             0            2
..   …    …               …  …          …            …           …
265   52    1                3  172          199             1            0
266   44    1                2  120          263             0            0
267   56    0                2  140          294             0            2
268   57    1                4  140          192             0            0
269   67    1                4  160          286             0            2
```

```
     Max HR  Exercise angina  ST depression  Slope of ST  \
0       109                0            2.4            2
1       160                0            1.6            2
2       141                0            0.3            1
3       105                1            0.2            2
4       121                1            0.2            1
..      …              …            …          …
265     162                0            0.5            1
266     173                0            0.0            1
267     153                0            1.3            2
268     148                0            0.4            2
269     108                1            1.5            2

     Number of vessels fluro  Thallium
0                          3         3
1                          0         7
2                          0         7
3                          1         7
4                          1         3
..                       …       …
265                        0         7
266                        0         7
267                        0         3
268                        0         6
269                        3         3

[270 rows x 13 columns]
```

[9]:

```
y=data["Heart Disease"] #this will have the dependent target variable
y
```

[9]:

```
0      Presence
1       Absence
2      Presence
3       Absence
4       Absence
         …
265     Absence
266     Absence
267     Absence
268     Absence
269    Presence
Name: Heart Disease, Length: 270, dtype: object
```

Importing the train_test_split module from the model selection package of the sklearn library

[22]:

```
from sklearn.model_selection import train_test_split
```

Spliting the data into training data and the testing data using the train test split module

[23]:

```
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2)
```

Checking the Split data

[25]:

```
print("The X training data \n",X_train)
print("The X testing data \n",X_test)
```

```
The X training data
      Age  Sex  Chest pain type   BP  Cholesterol  FBS over 120  EKG results  \
264   48    1                2  110          229             0            0
125   54    0                3  160          201             0            0
40    40    1                4  152          223             0            0
154   51    0                3  130          256             0            2
30    57    1                3  128          229             0            2
..   …  …              …  …          …           …          …
130   63    0                4  108          269             0            0
95    47    1                4  110          275             0            2
253   51    1                3  110          175             0            0
60    57    1                3  150          126             1            0
155   46    0                2  105          204             0            0

     Max HR  Exercise angina  ST depression  Slope of ST  \
264     168                0            1.0            3
125     163                0            0.0            1
40      181                0            0.0            1
154     149                0            0.5            1
30      150                0            0.4            2
..      …              …            …          …
130     169                1            1.8            2
95      118                1            1.0            2
253     123                0            0.6            1
60      173                0            0.2            1
155     172                0            0.0            1

     Number of vessels fluro  Thallium
264                        0         7
125                        1         3
40                         0         7
154                        0         3
30                         1         7
..                       …       …
130                        2         3
95                         1         3
253                        0         3
60                         1         7
155                        0         3
```

```
[216 rows x 13 columns]
The X testing data
      Age  Sex  Chest pain type   BP  Cholesterol  FBS over 120  EKG results   \
102   49    0                4  130          269             0            0
115   49    0                2  134          271             0            0
233   52    1                4  128          255             0            0
235   62    0                4  160          164             0            2
141   59    1                4  138          271             0            2
127   52    0                3  136          196             0            2
200   68    1                3  118          277             0            0
51    52    1                2  128          205             1            0
47    44    1                4  110          197             0            2
118   66    0                1  150          226             0            0
169   65    1                1  138          282             1            2
52    65    0                3  140          417             1            2
114   42    1                2  120          295             0            0
136   67    0                3  152          277             0            0
215   41    0                2  130          204             0            2
227   43    0                4  132          341             1            2
134   54    1                3  150          232             0            2
71    57    0                4  120          354             0            0
76    45    1                4  104          208             0            2
245   60    1                4  130          253             0            0
231   39    1                4  118          219             0            0
252   44    1                4  112          290             0            2
57    60    0                3  120          178             1            0
191   70    1                4  145          174             0            0
26    46    0                4  138          243             0            2
251   44    1                2  130          219             0            2
39    48    1                4  122          222             0            2
24    54    0                2  132          288             1            2
201   58    1                4  125          300             0            2
128   52    1                2  134          201             0            0
260   58    0                3  120          340             0            0
268   57    1                4  140          192             0            0
214   29    1                2  130          204             0            2
226   62    0                3  130          263             0            0
11    53    1                4  142          226             0            2
135   46    0                3  142          177             0            2
144   54    1                2  192          283             0            2
220   54    1                4  110          239             0            0
43    46    1                2  101          197             1            0
256   61    1                3  150          243             1            0
218   54    1                3  120          258             0            2
133   64    1                4  120          246             0            2
180   42    1                3  120          240             1            0
239   52    1                2  120          325             0            0
```

```
240   68    1                3  180          274             1            2
185   43    1                3  130          315             0            0
221   65    1                4  135          254             0            2
1     67    0                3  115          564             0            2
197   54    0                3  110          214             0            0
126   62    1                4  120          267             0            0
59    62    1                2  120          281             0            2
216   63    0                3  135          252             0            2
42    44    1                3  130          233             0            0
139   57    1                4  132          207             0            0

     Max HR  Exercise angina  ST depression  Slope of ST  \
102     163                0            0.0            1
115     162                0            0.0            2
233     161                1            0.0            1
235     145                0            6.2            3
141     182                0            0.0            1
127     169                0            0.1            2
200     151                0            1.0            1
51      184                0            0.0            1
47      177                0            0.0            1
118     114                0            2.6            3
169     174                0            1.4            2
52      157                0            0.8            1
114     162                0            0.0            1
136     172                0            0.0            1
215     172                0            1.4            1
227     136                1            3.0            2
134     165                0            1.6            1
71      163                1            0.6            1
76      148                1            3.0            2
245     144                1            1.4            1
231     140                0            1.2            2
252     153                0            0.0            1
57       96                0            0.0            1
191     125                1            2.6            3
26      152                1            0.0            2
251     188                0            0.0            1
39      186                0            0.0            1
24      159                1            0.0            1
201     171                0            0.0            1
128     158                0            0.8            1
260     172                0            0.0            1
268     148                0            0.4            2
214     202                0            0.0            1
226      97                0            1.2            2
11      111                1            0.0            1
135     160                1            1.4            3
```

```
144     195                0            0.0            1
220     126                1            2.8            2
43      156                0            0.0            1
256     137                1            1.0            2
218     147                0            0.4            2
133      96                1            2.2            3
180     194                0            0.8            3
239     172                0            0.2            1
240     150                1            1.6            2
185     162                0            1.9            1
221     127                0            2.8            2
1       160                0            1.6            2
197     158                0            1.6            2
126      99                1            1.8            2
59      103                0            1.4            2
216     172                0            0.0            1
42      179                1            0.4            1
139     168                1            0.0            1

     Number of vessels fluro  Thallium
102                        0         3
115                        0         3
233                        1         7
235                        3         7
141                        0         3
127                        0         3
200                        1         7
51                         0         3
47                         1         3
118                        0         3
169                        1         3
52                         1         3
114                        0         3
136                        1         3
215                        0         3
227                        0         7
134                        0         7
71                         0         3
76                         0         3
245                        1         7
231                        0         7
252                        1         3
57                         0         3
191                        0         7
26                         0         3
251                        0         3
39                         0         3
24                         1         3
```

```
201                        2         7
128                        1         3
260                        0         3
268                        0         6
214                        0         3
226                        1         7
11                         0         7
135                        0         3
144                        1         7
220                        1         7
43                         0         7
256                        0         3
218                        0         7
133                        1         3
180                        0         7
239                        0         3
240                        0         7
185                        1         3
221                        1         7
1                          0         7
197                        0         3
126                        2         7
59                         1         7
216                        0         3
42                         0         3
139                        0         7
```

[26]:
```
print("The Y training data \n",y_train)
print("The Y testing data \n",y_test)
```

```
The Y training data
 264    Presence
125     Absence
40     Presence
154     Absence
30     Presence
         …
130    Presence
95     Presence
253     Absence
60      Absence
155     Absence
Name: Heart Disease, Length: 216, dtype: object
The Y testing data
 102     Absence
115     Absence
233    Presence
235    Presence
```

```
141     Absence
127     Absence
200     Absence
51      Absence
47     Presence
118     Absence
169    Presence
52      Absence
114     Absence
136     Absence
215     Absence
227    Presence
134     Absence
71      Absence
76      Absence
245    Presence
231    Presence
252    Presence
57      Absence
191    Presence
26      Absence
251     Absence
39      Absence
24      Absence
201    Presence
128     Absence
260     Absence
268     Absence
214     Absence
226    Presence
11      Absence
135     Absence
144    Presence
220    Presence
43      Absence
256     Absence
218     Absence
133    Presence
180     Absence
239     Absence
240    Presence
185     Absence
221    Presence
1       Absence
197     Absence
126    Presence
59     Presence
216     Absence
```

```
42      Absence
139     Absence
Name: Heart Disease, dtype: object
```

Importing the learning model. I this case we are importing the Support Vector Classifier(SVC) module from the Support Vector Machine(SVM) package of the Sklearn Library.

```
[30]: from sklearn.svm import SVC
```

Fit the training data into the model

```
[35]: classifier = SVC(kernel="rbf")
      classifier.fit(X_train,y_train)
```

```
[35]: SVC()
```

Now make predictions on the testing set

```
[37]: y_pred=classifier.predict(X_test)
      print(y_pred)
```

```
['Absence' 'Absence' 'Absence' 'Absence' 'Absence' 'Absence' 'Absence'
 'Absence' 'Absence' 'Presence' 'Absence' 'Presence' 'Absence' 'Absence'
 'Absence' 'Presence' 'Absence' 'Presence' 'Absence' 'Absence' 'Absence'
 'Absence' 'Presence' 'Absence' 'Absence' 'Absence' 'Absence' 'Absence'
 'Absence' 'Absence' 'Absence' 'Absence' 'Absence' 'Presence' 'Presence'
 'Absence' 'Absence' 'Presence' 'Absence' 'Presence' 'Absence' 'Presence'
 'Absence' 'Absence' 'Presence' 'Absence' 'Presence' 'Presence' 'Absence'
 'Presence' 'Presence' 'Absence' 'Absence' 'Absence']
```

Making a dataframe to compare the actual data and the predicted data

```
[38]: df= DataFrame({"Actual":y_test,"Prediction":y_pred})
      df.head(10)
```

```
[38]:       Actual Prediction
      102   Absence    Absence
      115   Absence    Absence
      233  Presence    Absence
      235  Presence    Absence
      141   Absence    Absence
      127   Absence    Absence
      200   Absence    Absence
      51    Absence    Absence
      47   Presence    Absence
      118   Absence   Presence
```

Importing the accuracy score and the confusion matrix modules from the metrics package of sklearn library inorder to evalute the performance of the model.

```
[39]: from sklearn.metrics import␣
       ↪accuracy_score,confusion_matrix,classification_report
      accuracy=accuracy_score(y_test,y_pred)
      print("Accuracy : ",accuracy)
```

```
Accuracy :  0.6851851851851852
```

```
[43]: cnfmatrix=confusion_matrix(y_test,y_pred)
      print("The confusion matrix :\n",cnfmatrix)
```

```
The confusion matrix :
 [[29  7]
 [10  8]]
```

```
[44]: report=classification_report(y_test,y_pred)
      print("The classification report is :\n",report)
```

```
The classification report is :
               precision    recall  f1-score   support

     Absence       0.74      0.81      0.77        36
    Presence       0.53      0.44      0.48        18

    accuracy                           0.69        54
   macro avg       0.64      0.62      0.63        54
weighted avg       0.67      0.69      0.68        54
```

This model has been made by Aditya Kundu, Arnab Bera, Arnab Manna, Debojjo Talukdar, Biraj Naskar