NAME : Priyanka M. Narayankar

Heart Disease Prediction

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

#%matplotlib inline

import os

print(os.listdir())

import warnings

warnings.filterwarnings('ignore')

df =pd.read\_csv('/content/heart attack dataset.csv')

from sklearn.model\_selection import train\_test\_split

x = df.drop("target",axis=1)

y = df["target"]

dom\_state=0)

#### USING SVM

from sklearn.metrics import accuracy\_score

from sklearn import svm

sv = svm.SVC(kernel='linear')

sv.fit(X\_train, Y\_train)

Y\_pred\_svm = sv.predict(X\_test)

Y\_pred\_svm.shape

score\_svm = round(accuracy\_score(Y\_pred\_svm,Y\_test)\*100,2)

print("The accuracy score achieved using Linear SVM is: "+str(score\_svm)+" %")

##### USING DECISION TREEE

from sklearn.tree import DecisionTreeClassifier

max\_accuracy = 0

for x in range(200):

    dt = DecisionTreeClassifier(random\_state=x)

    dt.fit(X\_train,Y\_train)

    Y\_pred\_dt = dt.predict(X\_test)

    current\_accuracy = round(accuracy\_score(Y\_pred\_dt,Y\_test)\*100,2)

    if(current\_accuracy>max\_accuracy):

        max\_accuracy = current\_accuracy

        best\_x = x

dt = DecisionTreeClassifier(random\_state=best\_x)

dt.fit(X\_train,Y\_train)

Y\_pred\_dt = dt.predict(X\_test)

print(Y\_pred\_dt.shape)

score\_dt = round(accuracy\_score(Y\_pred\_dt,Y\_test)\*100,2)

print("The accuracy score achieved using Decision Tree is: "+str(score\_dt)+" %")

###### USING ARTIFICIAL NEURAL NETWORKS

from keras.models import Sequential

from keras.layers import Dense

model = Sequential()

model.add(Dense(11,activation='relu',input\_dim=13))

model.add(Dense(1,activation='sigmoid'))

model.compile(loss='binary\_crossentropy',optimizer='adam',metrics=['accuracy'])

model.fit(X\_train,Y\_train,epochs=2000)

Y\_pred\_nn = model.predict(X\_test)

Y\_pred\_nn.shape

rounded = [round(x[0]) for x in Y\_pred\_nn]

Y\_pred\_nn = rounded

score\_nn = round(accuracy\_score(Y\_pred\_nn,Y\_test)\*100,2)

print("The accuracy score achieved using Neural Network is: "+str(score\_nn)+" %")

scores = [score\_svm,score\_dt,score\_nn]

algorithms = ["Support Vector Machine","Decision Tree","Neural Network"]

for i in range(len(algorithms)):

    print("The accuracy score achieved using "+algorithms[i]+" is: "+str(scores[i])+" %")