import pandas as pd

import pickle

from sklearn.preprocessing import StandardScaler

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

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix

from sklearn.ensemble import RandomForestClassifier

from sklearn.svm import SVC

from sklearn.linear\_model import LogisticRegression

from sklearn.neighbors import KNeighborsClassifier

from sklearn.tree import DecisionTreeClassifier

# loading and reading the dataset

heart = pd.read\_csv("C:\\Users\\Thinesh\\Desktop\\heart\_cleveland\_upload.csv")

# creating a copy of dataset so that will not affect our original dataset.

heart\_df = heart.copy()

# Renaming some of the columns

heart\_df = heart\_df.rename(columns={'condition':'target'})

print(heart\_df.head())

x= heart\_df.drop(columns= 'target')

y= heart\_df.target

# splitting our dataset into training and testing for this we will use

#train\_test\_split library.

x\_train, x\_test, y\_train, y\_test= train\_test\_split(x, y, test\_size= 0.25, random\_state=42)

#feature scaling

scaler= StandardScaler()

x\_train\_scaler= scaler.fit\_transform(x\_train)

x\_test\_scaler= scaler.fit\_transform(x\_test)

# creating K-Nearest-Neighbor classifier

model=RandomForestClassifier(n\_estimators=20)

model.fit(x\_train\_scaler, y\_train)

y\_pred=model.predict(x\_test\_scaler)

p =model.score(x\_test\_scaler,y\_test)

print(p)

print('Classification Report\n', classification\_report(y\_test, y\_pred))

print('Accuracy: {}%\n'.format(round((accuracy\_score(y\_test, y\_pred)\*100),2)))

# Save the trained model and scaler to pickle files

with open('heart\_model.pkl', 'wb') as model\_file:

pickle.dump(model, model\_file)

with open('scaler.pkl', 'wb') as scaler\_file:

pickle.dump(scaler, scaler\_file)

# HTML CODE

<!DOCTYPE html>

<html>

<head>

<title>Heart Disease Prediction</title>

<link rel="stylesheet" href="{{ url\_for('static', filename='css/styles.css') }}">

<style>

body {

font-family: Arial, sans-serif;

background-image: url("C:\\Users\\Downloads\\desktop-wallpaper-human-heart-10-

thumbnail.jpg");

background-size: cover;

background-position: center;

background-repeat: no-repeat;

margin: 0;

padding: 0;

}

.container {

width: 100%;

max-width: 600px;

margin: 50px auto; /\* Adjust the top margin as needed \*/

padding: 20px;

background-color: rgba(0, 0, 0, 0.5);

border-radius: 10px;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.3);

}

h1 {

text-align: center;

margin-bottom: 20px;

color: white; /\* Text color \*/

}

form {

margin-bottom: 20px;

}

label {

display: block;

margin-bottom: 5px;

font-weight: bold;

color: white; /\* Text color \*/

}

input[type="text"],

input[type="radio"] {

width: 100%;

padding: 10px;

margin-bottom: 10px;

border: 1px solid #ccc;

border-radius: 5px;

box-sizing: border-box;

}

input[type="submit"] {

width: 100%;

padding: 10px;

background-color: #ff4d4d; /\* Red color for button \*/

color: white;

border: none;

border-radius: 5px;

cursor: pointer;

opacity: 0.8;

}

input[type="submit"]:hover {

background-color: #ff3333; /\* Darker red on hover \*/

opacity: 1;

}

.result {

margin-top: 20px;

font-weight: bold;

color: white; /\* Text color \*/

}

</style>

</head>

<body>

<div class="container">

<h1>Heart Disease Prediction</h1>

<form method="POST" action="/">

<label for="age">Age (age):</label>

<input type="text" id="age" name="age" required>

<label for="sex">Sex (sex):</label><br>

<input type="radio" id="male" name="sex" value="1" required>

<label for="male">Male</label>

<input type="radio" id="female" name="sex" value="0">

<label for="female">Female</label>

<label for="cp">Chest Pain Type (cp):</label>

<input type="text" id="cp" name="cp" required>

<label for="trestbps">Resting Blood Pressure (trestbps):</label>

<input type="text" id="trestbps" name="trestbps" required>

<label for="chol">Serum Cholesterol (chol):</label>

<input type="text" id="chol" name="chol" required>

<label for="fbs">Fasting Blood Sugar (fbs):</label>

<input type="text" id="fbs" name="fbs" required>

<label for="restecg">Resting Electrocardiographic Results (restecg):</label>

<input type="text" id="restecg" name="restecg" required>

<label for="thalach">Maximum Heart Rate Achieved (thalach):</label>

<input type="text" id="thalach" name="thalach" required>

<label for="exang">Exercise Induced Angina (exang):</label>

<input type="text" id="exang" name="exang" required>

<label for="oldpeak">ST Depression Induced by Exercise (oldpeak):</label>

<input type="text" id="oldpeak" name="oldpeak" required>

<label for="slope">Slope of Peak Exercise ST Segment (slope):</label>

<input type="text" id="slope" name="slope" required>

<label for="ca">Number of Major Vessels Colored by Fluoroscopy (ca):</label>

<input type="text" id="ca" name="ca" required>

<label for="thal">Thalassemia (thal):</label>

<input type="text" id="thal" name="thal" required>

<input type="submit" value="Predict">

</form>

{% if result %}

<div class="result">

<h2>{{ result }}</h2>

</div>

{% endif %}

</div>

</body>

</html>