**Improving the accuracy of predicting the risk of heart diseases using ensemble learning techniques**

A Mini project submitted in partial fulfillment of the requirements for the degree of **Master of Computer Applications**

By

**SAHIL GOEL (205118062)**

****

**DEPARTMENT OF COMPUTER APPLICATIONS NATIONAL INSTITUTE OF TECHNOLOGY TIRUCHIRAPPALLI – 620015**

**DCEMBER-2020 (5th Semester)**

# CERTIFICATE

This is to certify that “**Improving the accuracy of predicting the risk of heart diseases using ensemble learning techniques.”** is a Mini project work successfully done bySAHIL GOEL (205118062).In partial fulfillment of the requirements for the awards of the degree of **Master of Computer Applications** of **National Institute of Technology, Tiruchirappalli** during the year 2020 (5th Semester).

**Dr. B. Balaji Dr. P. J. A. Alphonse**

**Project Guide Head of the Department**

Project viva-voce held on **10th Dec 2020**

# ABSTRACT

Machine learning requires artificial intelligence, and it is widely used in solving many problems which are native to data science. One common use case of machine learning is the predicting the outcome which is dependent upon existing data. The machine detects patterns from the existing dataset, and then applies them to an unknown dataset in order to predict the outcome.

Classification is a powerful machine learning technique that is commonly used for predicting a class. Some classification algorithms predict with good accuracy, whereas others exhibit a limited accuracy. This project implements a method termed ensemble classification, which is used for improving the accuracy of the classifiers with weak accuracy by combining them. Experiments with this method were performed using heart disease datasets.

A comparison is shown to determine how the ensemble techniques like stacking can be applied for improving prediction accuracy in heart disease. The focus of this project is to implement these techniques and determine if there is an increase in prediction accuracy. The results of the project indicate that with ensemble techniques such as stacking and by using feature selection we can improve the prediction accuracy of weak classifiers and exhibit satisfactory performance in identifying risk of heart disease.

# ACKNOWLEDGEMENTS

We express our deep sense of gratitude to **Dr. Mini Shaji Thomas**, Director, National Institute of Technology, Tiruchirappalli for giving us an opportunity to do this Mini project.

I wish to extend my sincere gratitude to **Dr. P. J. A. Alphonse**, Professor and Head of the Department, Department of Computer Applications, National Institute of Technology, Tiruchirappalli for supporting and encouraging me to carry out this project work and duly evaluating my progress.

I express my heartfelt gratitude to my internal guide **Dr. B. Balaji,** AssistantProfessor, Department of Computer Applications, National Institute of Technology, Tiruchirappalli for his immense support and guidance during the study.

I also thank the review panel for helping me identifying the shortages of the project and helping me to overcome those shortages by guiding me.

Finally, I would like to express my regards for all the faculty members of the Department of Computer Applications and others who have helped me develop this project directly or indirectly.

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **SERIAL NO** | **TITLE** | **PAGE NO** |
| **1** | **CERTIFICATE** | **I** |
| **2** | **ABSTRACT** | **II** |
| **3** | **ACKNOWLEDGEMENTS** | **III** |
| **4** | **TABLE OF CONTENTS** | **IV** |
| **5** | **CHAPTER-1 INTRODUCTION** | **1** |
| **6** | **CHAPTER-2 PROBLEM STATEMENT** | **2** |
| **7** | **CHAPTER-3 PLATFORM** | **3** |
| **8** | **CHAPTER-4 RELATED WORK** | **4** |
| **9** | **CHAPTER-5 METHODOLOGY USED** | **5-7** |
| **10** | **CHAPTER-6 DESIGN AND DEVELOPMENT** | **8-10** |
| **11** | **CHAPTER-7 OUTPUT AND RESULT** | **11-13** |
| **12** | **CHAPTER-8 CONCLUSION AND FUTURE WORK** | **14** |
| **13** | **CHAPTER-9 CODE IMPLEMENTATION** | **15-18** |
| **14** | **REFERENCES** | **19** |

**CHAPTER – 1 INTRODUCTION**

Cardiovascular Diseases (CVDs) amount to most deaths globally: more people die annually from CVDs as compared to any other cause. An estimated 17.9 million people died from CVDs in 2016, representing 31% of all global deaths. Of those deaths, 85% are because of heart attack and stroke. Over three quarters of CVD deaths take place in low- and middle-income countries. Out of the 17 million premature, death before the age of 70, due to noncommunicable diseases in 2015, 37% are caused by CVDs.

Most cardiovascular diseases can be prevented if we keep track of behavioral risk factors like smoking, unhealthy diet, obesity, tobacco use etc. People having cardiovascular disease or who are at high cardiovascular risk (due to the presence of risk factors involving hypertension, diabetes, hyperlipidemia or already established disease) need early detection and management using counselling and medicines, as appropriate.

It is not easy to manually determine the chances of getting heart disease based on risk factors. Machine Learning techniques can help to fulfill this need. Hence, this project uses one such machine learning technique called classification for predicting heart disease risk from the risk factors. It is trying to show how ensemble techniques can be used to increase the accuracy of predicting risks.

# 

# CHAPTER – 2

# PROBLEM STATEMENT

If we can detect cardiovascular diseases early, we can save a lot of lives. If an individual is aware of the signs of CVD, it can be detected early. When a person has risk factors mainly because of behavioral issues, their doctor can refer them to a specialist for further testing. Some metrics such as Blood Pressure, glucose level, age, BMI value, physical exercise, smoking and alcohol history can be used to predict the risk of heart diseases for an individual.

# CHAPTER – 3 PLATFORM

**Hardware:**

* Processor: intel core i5/i7
* Operating system: Windows 10 or Linux
* Space: 2GB

**Software:**

* PyCharm
* Web Browser

**Web Framework:**

* Flask

**Library:**

* Sci-kit learn
* Matplotlib
* NumPy
* Pandas
* Pickle

**CHAPTER – 4**

**RELATED WORK**

Studies on classification techniques have shown that neural networks, decision trees, Naïve Bayes and associative classification are very good in predicting heart diseases. Associative classification is highly accurate and gives us strong flexibility as compared to traditional classifiers.

Analysis has shown that decision tree classifiers are simple as well as accurate. Among the techniques Naïve Bayes was found the best algorithm closely followed by neural networks and decision tree. Artificial neural networks are also used for detecting the disease by training them using Back-Propagation Algorithm.

Most of the research that produces high accuracy uses a hybrid method which include classification algorithms.

**CHAPTER – 5**

**METHODOLGY USED**

Methodologies used in this project are:

* **Stacking –** Stacking is an ensembling technique in which the training data is passed through various classifiers and the output of these classifiers is passed to another classifier which predicts the final class based on the outputs of original classifiers.

**Algorithm**

Let D= {d1, d2, d3, ... dn} be the given dataset

E = {E1, E2, E3, ...En}, the set of ensemble classifiers

C = {c1, c2, c3, ...cn}, the set of classifiers

X = the training set, X∈D

Y = the test set, Y∈D

K = meta level classifier

L = n(D)

for i =1 to L do

M(i) = Model trained using E(i) on X

Next i

M=M∪K

Result = Y classified by M

* **Bagging –** Bagging is a technique in which all the classifiers are trained on a subset of original data and the result is calculated based on the class predicted by the classifiers based on majority count.

**Algorithm**

Let D = {d1, d2, d3, ... dn} be the given dataset

E = {}, the set of ensemble classifiers

C = {c1, c2, c3, ...cn}, the set of classifiers

X = the training set, X∈D

Y = the test set, Y∈D

L = n(D)

for i =1 to L do

S(i) = {Bootstrap sample I with replacement} I⊂X

M(i) = Model trained using C(i) on S(i)

E = E∪C(i)

next i

for i = 1 to L

R(i) = Y classified by E(i)

next i

Result = max(R(i): i=1,2, ..., n)

* **Boosting –** Boosting is a technique in which weights are given to each record and it is increased for those records which are classified incorrectly. The classifier will adjust the parameters to classify the records which have more weight.

**Algorithm:**

Let D = {d1, d2, d3, ... dn} be the given dataset

E = {}, the set of ensemble classifiers

C = {c1, c2, c3, ...cn}, the set of classifiers

X = the training set, X∈D

Y = the test set, Y∈D

L = n(D)

Let init = 1

S(init)=A random subset of X; S(init)⊂X

M (0) = {}

For i =1 to L do

if i > 1

s(i) = Set of incorrectly classified instances of M(i-1) + S(i)

M(i) = Model trained using C(i) on S(i)

E = E∪C(i)

end if

next i

for i = 1 to L

R(i) = Y classified by E(i)

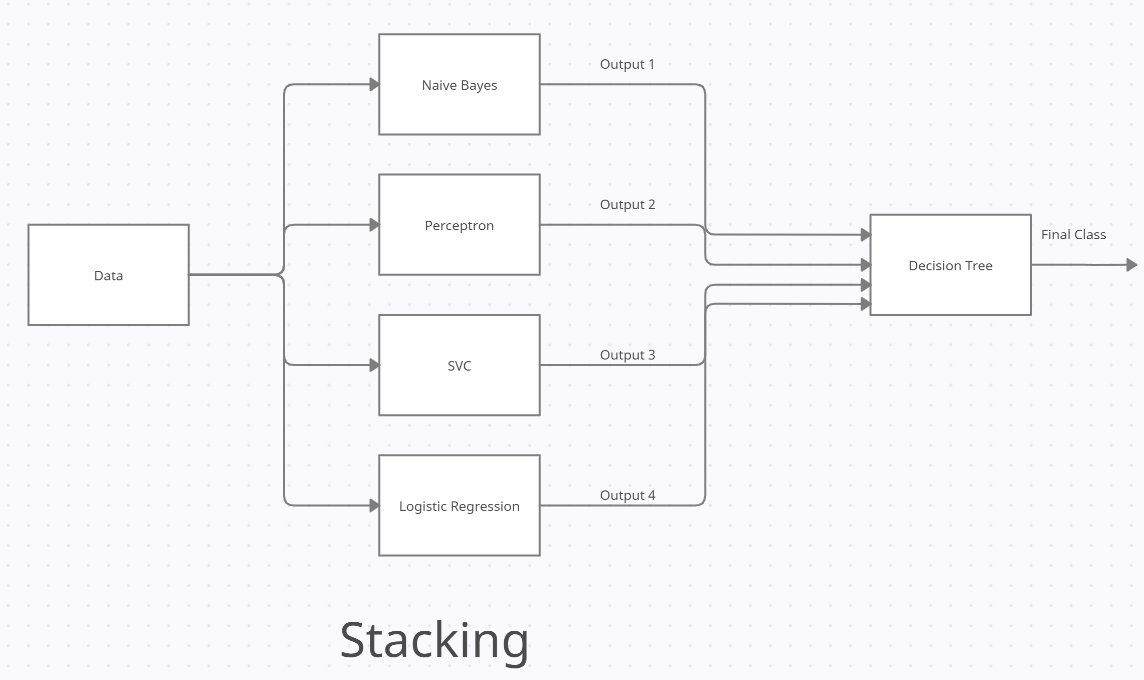
next i

Result = max(R(i): i=1,2, ..., n)

**CHAPTER - 6**

**DESIGN AND DEVELOPMENT**

* **Stacking**

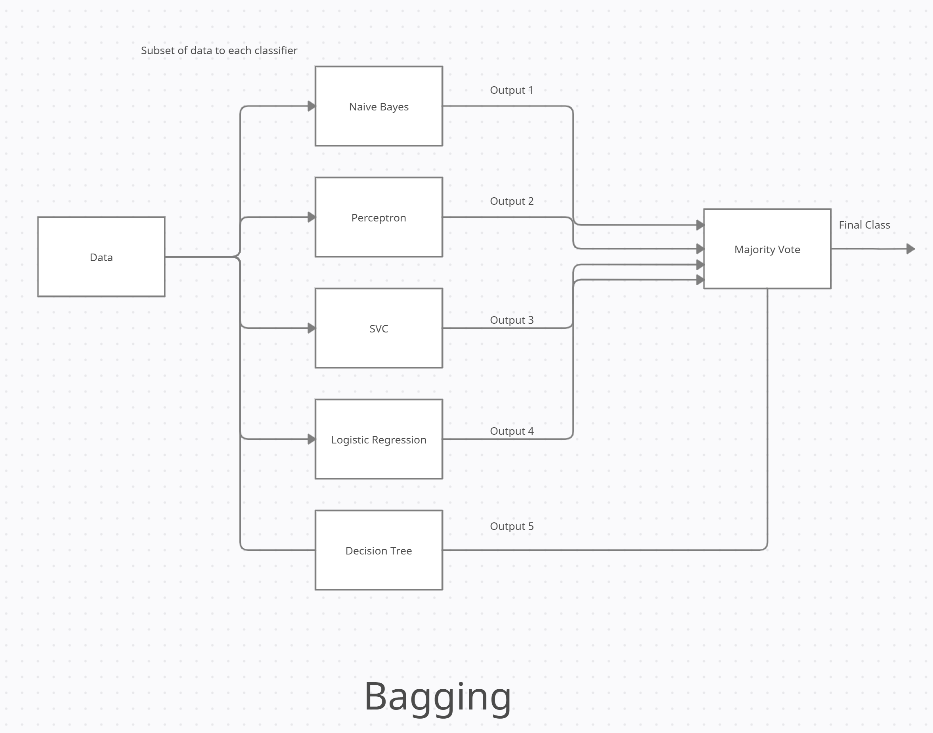


Classifiers used as base classifiers are:

* Naïve Bayes
* Multi-layer Perceptron
* Support Vector Classifier
* Logistic Regression

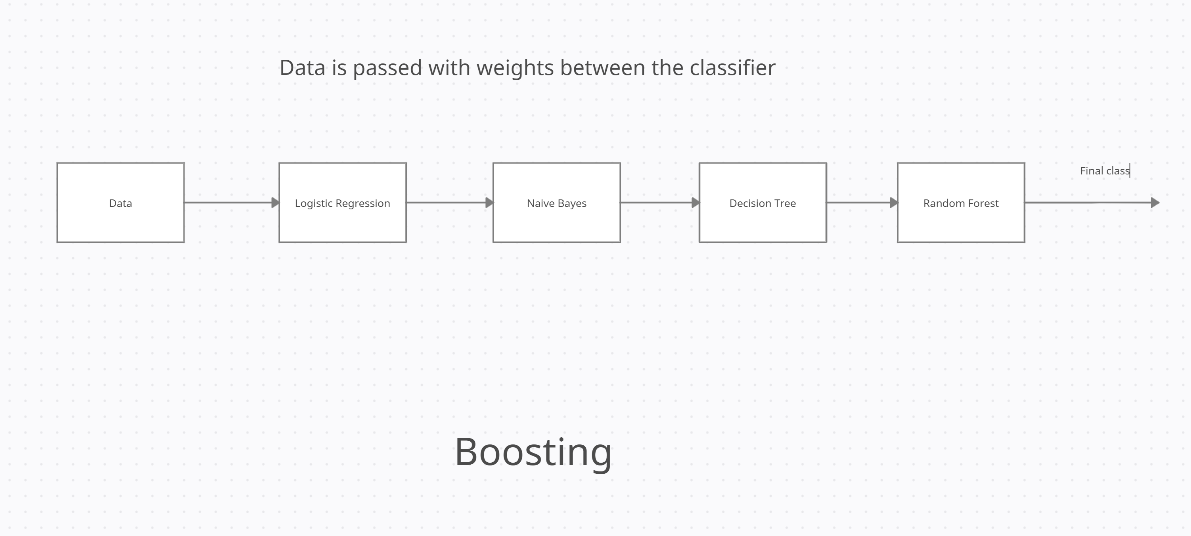
Final Classifier used:

* Decision Tree
* **Bagging**



Classifiers used:

* Naïve Bayes
* Multi-layer Perceptron
* Support Vector Classifier
* Logistic Regression
* Decision Tree
* **Boosting**



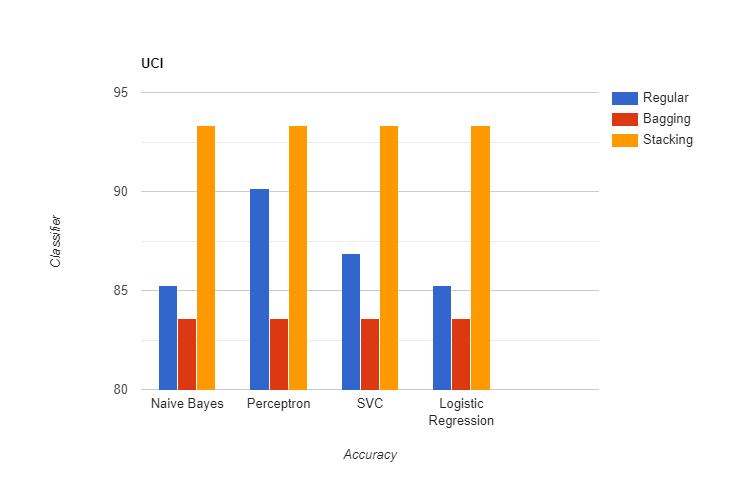
Classifiers used are:

* Logistic Regression
* Naïve Bayes
* Decision Tree
* Random Forest

**CHAPTER-7**

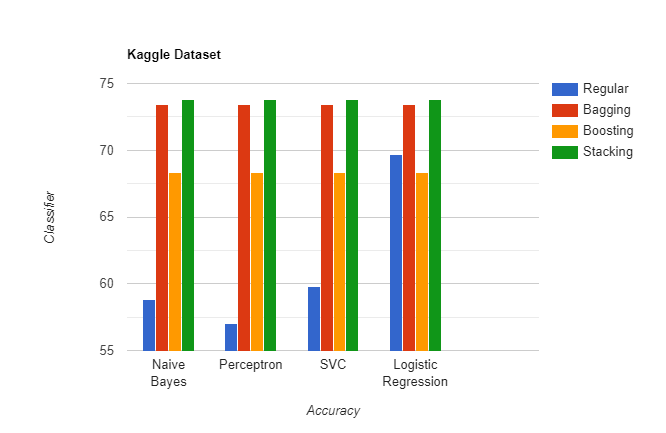
**OUTPUT AND RESULT**

* **UCI Dataset Accuracy:**



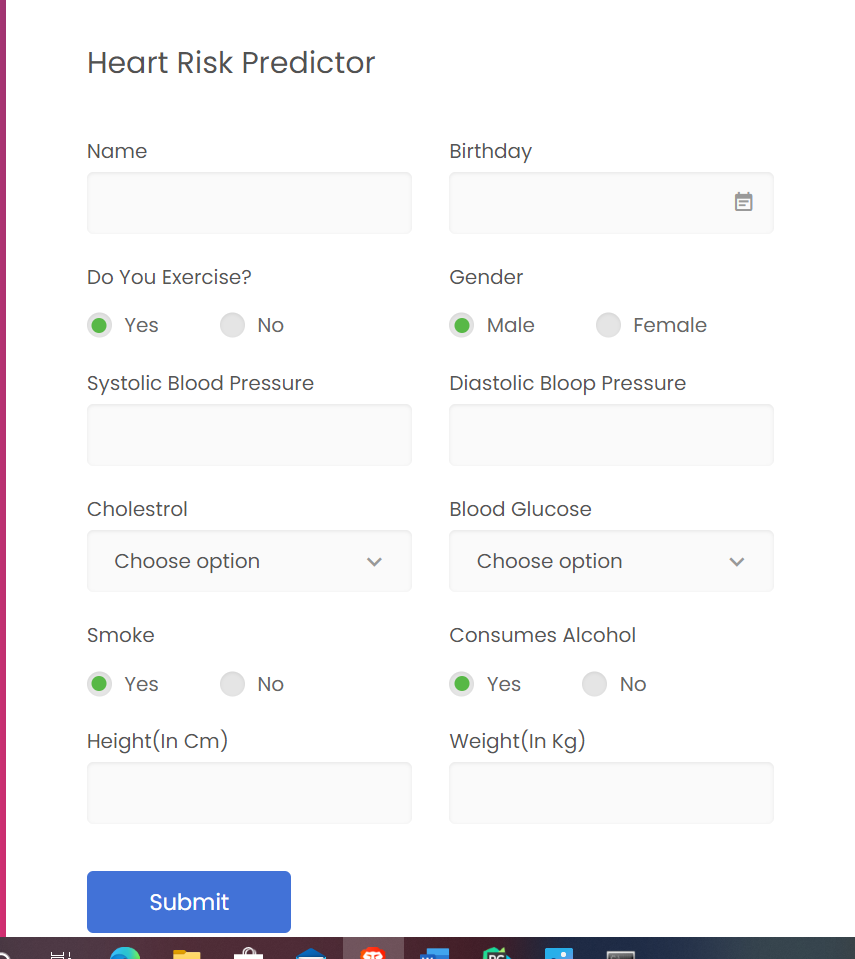
|  |  |
| --- | --- |
| **CLASSIFIER** | **ACCURACY** |
| Naïve Bayes | 85.24% |
| Multilayer Perceptron | 90.16% |
| Support Vector Machine | 86.88% |
| Logistic Regression | 85.25% |
| Stacking | 93.33% |
| Bagging | 83.60% |

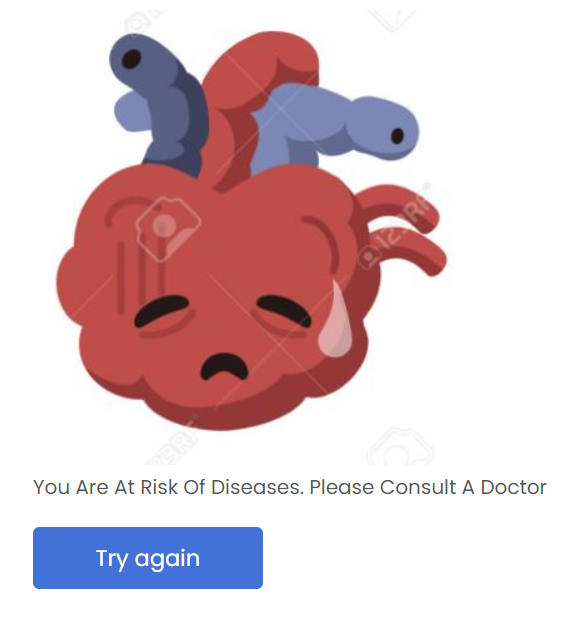
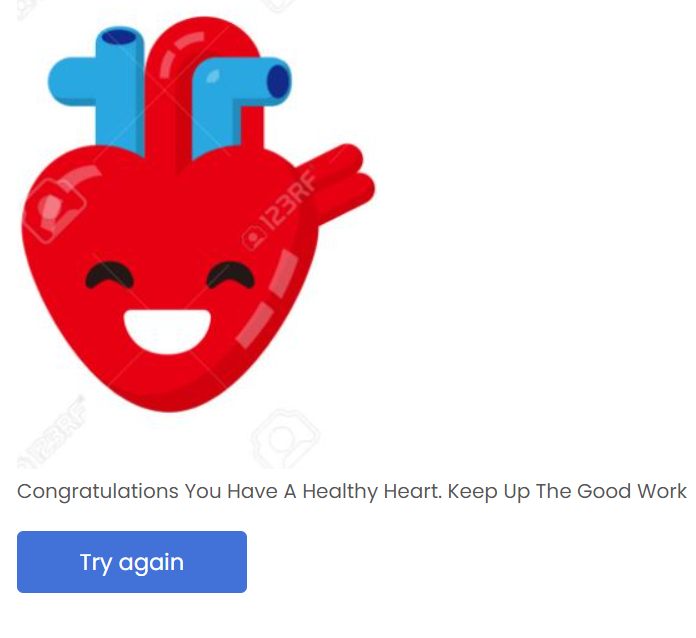
* **Kaggle Dataset**



|  |  |
| --- | --- |
| **CLASSIFIER** | **ACCURACY** |
| Naïve Bayes | 58.85% |
| Multilayer Perceptron | 50.86% |
| Support Vector Classifier | 59.84% |
| Logistic Regression | 69.68% |
| Bagging | 73.49% |
| Boosting | 67.62% |
| Stacking | 73.55% |

* **Webpage**



**CHAPTER-8**

**CONCLUSION AND FUTURE WORK**

**Conclusion:**

From the result obtained it is clear that ensemble techniques do help in increasing the accuracy. With the help of this project, an individual can easily know whether he is at risk of disease. Although it is not a substitute for professional medical opinion, it can act as an indicator.

**Future Work:**

* Live learning can be used in which the model can be trained with new data as it is entered. It will help in adjusting the model with new trends in data.
* An application with database can be made through which the user can keep track of the previous records and keep track of the changes.

**CHAPTER-9**

**CODE IMPLEMENTATION**

**Form.py**

from flask import Flask, render\_template, flash, request, redirect, url\_for,session

from wtforms import Form, TextField, TextAreaField, validators, StringField, SubmitField, DateField, RadioField

import numpy as np

import pandas as pd

import pickle

import datetime

from os import path

import matplotlib.pyplot as plt

# App config.

DEBUG = True

app = Flask(\_\_name\_\_)

app.config.from\_object(\_\_name\_\_)

app.config['SECRET\_KEY'] = '7d441f27d441f21567d441f2b6176a'

class ReusableForm(Form):

@app.route("/", methods=['GET', 'POST'])

def hello():

form = ReusableForm(request.form)

if session.get('result') is None:

session['result']=-1

return render\_template('index.html',results=int(session['result']), form=form)

@app.route("/check",methods=['POST'])

def check():

gender = request.form['gender']

dateofBirth = request.form['birthday']

systolic = request.form['sysbp']

diastolic = request.form['diabp']

cholestrol = request.form['cholestrol']

smoke = request.form['smoker']

alcohol = request.form['alcohol']

glucose = request.form['glucose']

exercise = request.form['phe']

height = request.form['height']

weight = request.form['weight']

session['result'] = predictor(dateofBirth,gender,systolic,glucose,diastolic,cholestrol,smoke,alcohol,exercise, height,weight)

return redirect('/')

@app.route("/clear", methods=['POST'])

def clear():

session.clear()

return redirect('/')

if \_\_name\_\_ == "\_\_main\_\_":

app.run()

def predictor(date\_entry,gender,systolic,glucose,diastolic,cholestrol,smoke,alcohol,exercise,height,weight):

columns = ['age', 'gender', 'ap\_hi', 'ap\_lo', 'cholesterol', 'gluc', 'smoke', 'alcohol', 'active', 'bmi']

user\_data = []

day, month, year = map(int, date\_entry.split('/'))

birth\_date = datetime.date(year, month, day)

curr\_date = datetime.date.today()

user\_data.append((curr\_date - birth\_date).days)

user\_data.append(int(gender))

user\_data.append(int(systolic))

user\_data.append(int(diastolic))

user\_data.append(int(cholestrol))

user\_data.append(int(glucose))

user\_data.append(int(smoke))

user\_data.append(int(alcohol))

user\_data.append(int(exercise))

height = int(height)

weight = int(weight)

user\_data.append(weight / (height / 100) \*\* 2)

individual\_predictions = pd.DataFrame()

data = pd.DataFrame([user\_data], columns=columns)

fileName = 'Logistic Regression Stacking.sav'

logClassifier = pickle.load(open(fileName, 'rb'))

individual\_predictions['Logistic'] = logClassifier.predict(data)

fileName = 'Naive bayes Stacking.sav'

nb = pickle.load(open(fileName, 'rb'))

individual\_predictions['Naive Bayes'] = nb.predict(data)

fileName = 'Perceptron Stacking.sav'

perc = pickle.load(open(fileName, 'rb'))

individual\_predictions['Perceptron'] = perc.predict(data)

fileName = 'SVC Stacking.sav'

svc = pickle.load(open(fileName, 'rb'))

individual\_predictions['SVC'] = svc.predict(data)

fileName = 'Stacking.sav'

stacking = pickle.load(open(fileName, 'rb'))

output = pd.DataFrame()

output['answer'] = stacking.predict(individual\_predictions)

return int(output.at[0,'answer'])

# REFERENCES

1. https://www.kaggle.com/sulianova/cardiovascular-disease-dataset
2. https://www.sciencedirect.com/science/article/pii/S235291481830217X#:~:text=The%20results%20of%20the%20study,identifying%20risk%20of%20heart%20disease.
3. https://www.researchgate.net/publication/337063133\_A\_Review\_on\_Heart\_Disease\_Prediction\_Using\_Machine\_Learning\_Techniques
4. https://www.kaggle.com/ronitf/heart-disease-uci