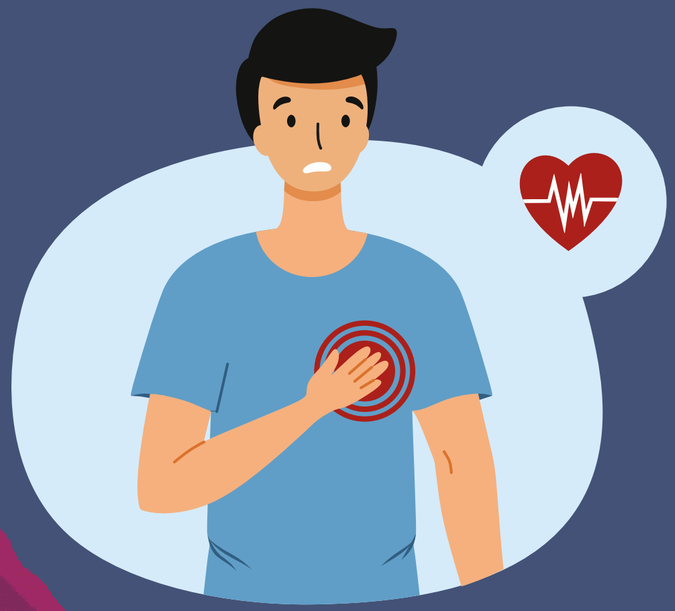
CARDIO VASCULAR DISEASE RISK PREDICTION



TEAM MEMBERS

Khyaat Punetha	21BCE10086
Swastik Yadav	21BCE10088
Ujesh Mishra	21BCE10183
Atharva Jaurkar	21BCE10300
Sumit Jha	21BCE11656

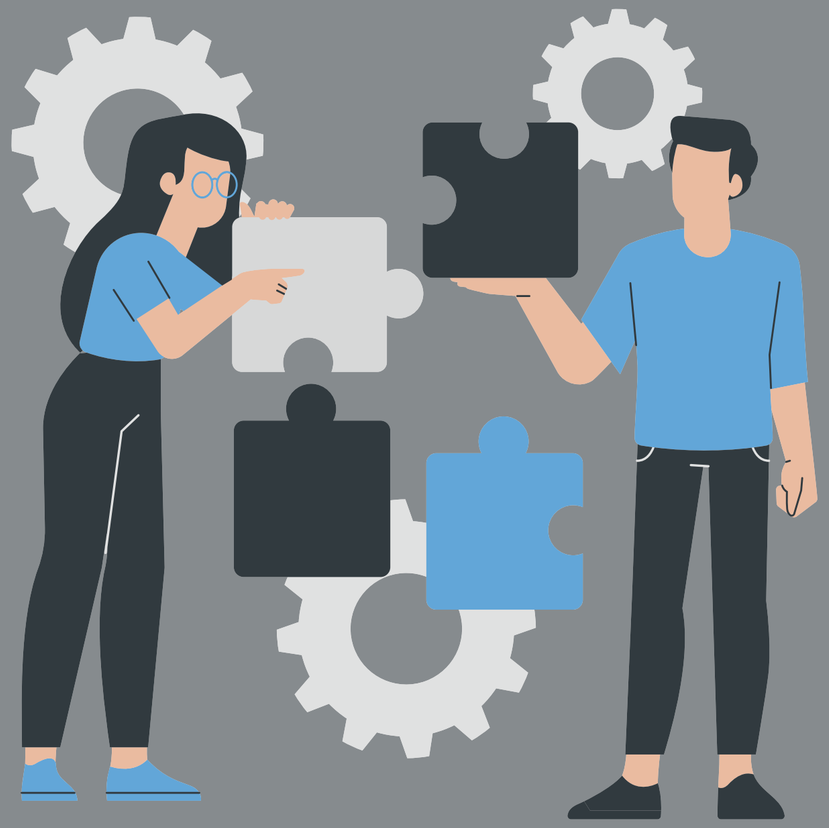
Guided by: Dr. Gopal Singh Tandel

INTRODUCTION

Cardiovascular disease is one of the most heinous disease, especially the silent heart attack, which attackes a person so abruptly that there's no time to get it theated and such disease is very difficult to be diagnosed. Various medical datamining and machine leaming techniques are being implementad to extract the valuable information regarding the heart disease prediction. Yet, the accuracy of the desired results are not satisfactory. This Model proposes heart attack prediction system using Machine learning techniques. Health care field has a vast amount of data, for processing those data certain techniques are used. Dataminingis one of the techniques often used. Heart diseases the Leading cause of death worldwide. This System predicts the arising possibilities of Heart Disease. The datasets used are classified in terms of medical parameters. This system evaluates those parameters using data mining dassification technique. The datasets are processed in python programming using two main Machine Leaming Algorithm namely Decision tree Algorithm and Naive Bayes Algorithm which shows the best algorithm among these two in terms of accuracy level of heart disease.

PROBLEM STATEMENT

We have a dataset that consists of patient's health details. From these details we are going to detect whether the patient will suffer from heart disease or not. We will attempt to develop a model that can determine whether a patient has this heart disease or not.



EXISTING WORK WITH LIMITATIONS

There are some model/application which can successfully predict if the person has cardio vascular disease or not.
But there is no such system that can predict the disease using multiple models of Machine Learning and compare the results of those multiple models and predict the disease with better accuracy.
Also there can be models which detects the disease accurately but our project can tell Specificity,Sensitivity,NPV,PPV,AUC also in percentage.

NOVELTY OF THE PROJECT

Our Project is novel as we are assisting the patients by trying to determine whether or not they have risk for any cardio vascular disease or not, with the best accuracy by taking the data of their medical reports in just couple of seconds. Also we are telling them what is their Specificity,Sensitivity,NPV,PPV,of disease in percentage so that they can have pre-knowledge of intensity of their disease. All of this will be informed by our model.

Last but not the least the detection of cardio vascular disease with the best accuracy by comparing with another models and hence this all makes our project novel.

REAL TIME USAGE

The real time usage of our project is to predict the risk of CARDIO VASCULAR DISEASE of a patient with the best accuracy so that the patient can have a pre-knowledge of their disease and can take further steps to avoid the disease.

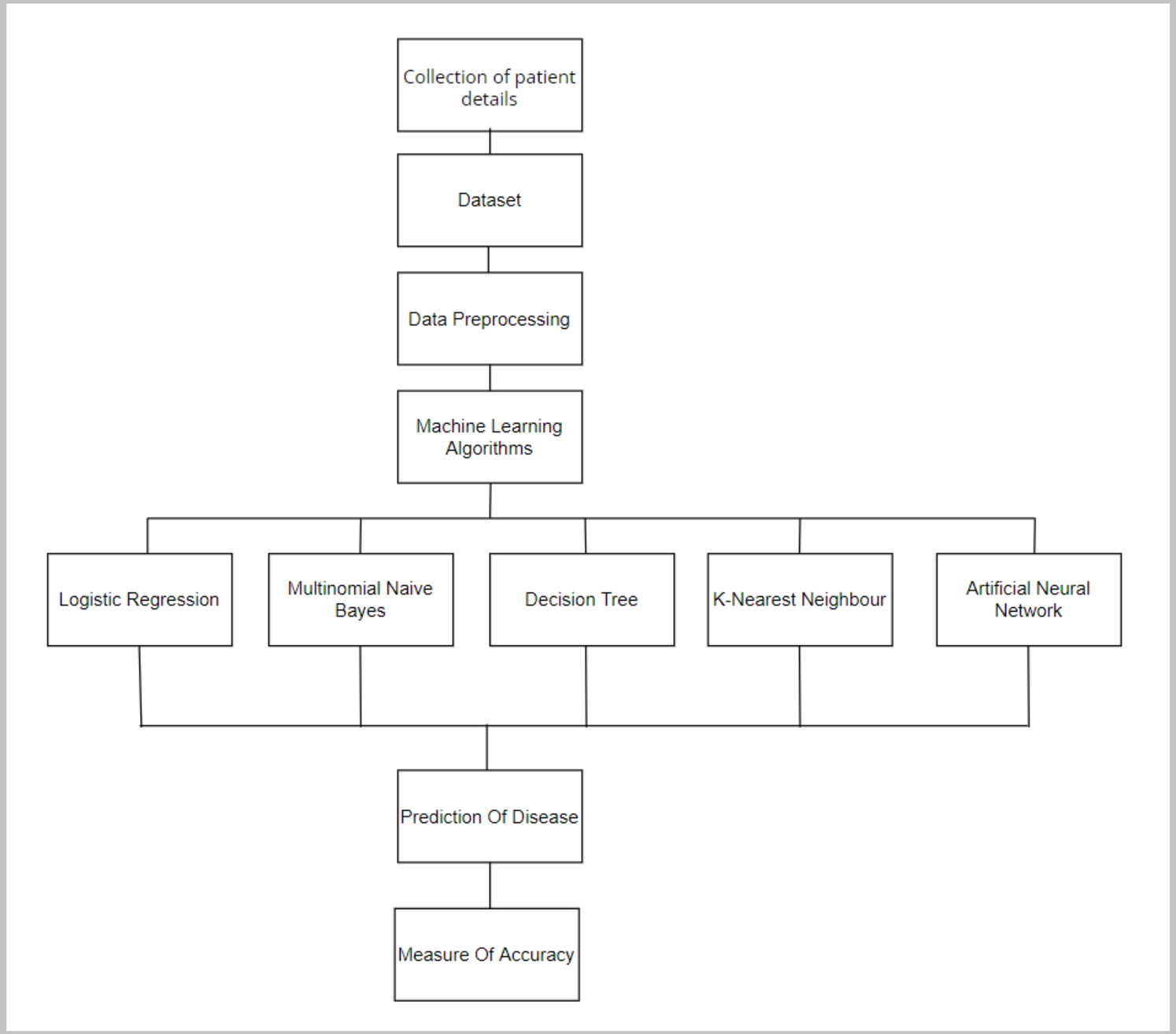
HARDWARE AND SOFTWARE REQUIREMENT

- Minimum 4 GB RAM
- OPERATING SYSTEM-WINDOWS 7+, UBUNTU, MAC OS CATALINA
- CPU REQUIREMENT – Minimum of intel core i3 7th gen processor
- MINIMUM OF 128 GB ROM

SOFTWARE REQUIREMENT-

- Google colab OR Jupyter Notebook
- MS EXCEL OR GOOGLE SHEETS for editing spreadsheets
- Python 3.11 and above

Overall system architecture diagram.



LITERATURE REVIEW

Cardiovascular disease (CVD) is a class of diseases that involve the heart or blood vessels. Cardiovascular diseases are the leading cause of death worldwide except Africa.[3] Together CVD resulted in 17.9 million deaths (32.1%) in 2015, up from 12.3 million (25.8%) in 1990. Numerous studies have been done that have focus on diagnosis of heart disease. They have applied different data mining techniques for diagnosis & achieved different probabilities for different methods.

We have seen about 10 research papers for the project from various Journals, we have scanned and compared the previously made models such as

Pal M, Parija S, Panda G, Dhama K, Mohapatra RK. Risk prediction of cardiovascular disease using machine learning classifiers.

M. Dhilsath Fathima, S. Justin Samuel, R. Natchadalingam, V. Vijeya Kaveri. (2022) Majority voting ensembled feature selection and customized deep neural network for the enhanced clinical decision support system.

And these researches and models have helped us a lot in our model making.

MODULES

Our project is divided among different stages or modules

1. **Data Finding** → Data is gathered from the Kaggle , a dataset finding site from here we have gathered a dataset containing all the attributes on which one's CV disease is predicted
2. **Data Preparation** → Data should be prepared such that there is no missing values in any of the columns for any of the row. For this we will use the pandas library of python.

3. **Data Training →** Hence to make the machine learn we will use the schikit-learn to distribute the data into two parts one is training data and one is testing data, hence the machine learning algorithm will be used here.
4. **Testing** → Now its time to test the data, we have some data left after training, hence it will be used to predict the outcome and the efficiency of the model.
5. If the model is of good efficiency,then last stage will be deploying the model and predicting if the person is healthy or not.

MODEL USED

1) Logistic Regression Model :-

Logistic regression is a process of modeling the probability of a discrete outcome given an input variable. The most common logistic regression models a binary outcome; something that can take two values such as true/false, yes/no, and so on. Logistic regression is a useful analysis method for classification problems, where you are trying to determine if a new sample fits best into a category. As aspects of cyber security are classification problems, such as attack detection, spam detection, fraud detection, emergency detection, logistic regression is a useful analytic technique.

MODEL USED

2) Multinomial Naive Bayes Model :-

The Multinomial Naive Bayes algorithm is a Bayesian learning approach popular in Natural Language Processing (NLP). The program guesses the tag of a text, such as an email or a newspaper story, using the Bayes theorem. It calculates each tag's likelihood for a given sample and outputs the tag with the greatest chance. The Naive Bayes classifier is made up of a number of algorithms that all have one thing in common: each feature being classed is unrelated to any other feature. A feature's existence or absence has no bearing on the inclusion or exclusion of another feature.

MODEL USED

3) Decision Tree Model :-

A decision tree is a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements. A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute

Decision trees are commonly used in operations research, specifically in decision analysis, to help identify a strategy most likely to reach a goal, but are also a popular tool in machine learning.

MODEL USED

4) K-Nearest Neighbour :-

The KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.

k-NN is a type of classification where the function is only approximated locally and all computation is deferred until function evaluation. Since this algorithm relies on distance for classification, if the features represent different physical units or come in vastly different scales then normalizing the training data can improve its accuracy dramatically.

MODEL USED

5)Artificial Neural Network :-

A brain's neural network is made up of around 86 billion neurons. The neurons are connected by what are called synapses. There are about 100 trillion synapses in the human brain. The neurons send signals to each other through the synapses.

To create a neural network we combine artificial neurons together so that the outputs of some neurons are inputs of other neurons. We will be working with feed forward neural networks which means that the neurons only send signals in one direction. In particular, we will be working with what is called a Multi-Layer Perceptron (MLP).

Work Flow

Heart Data

Data pre processing

Train Test split

Logistic Regression model

Trained Logistic Regression model

New data

Healthy
(or)
Heart Defect

Prediction

PERFORMANCE EVALUATION OF MODEL

Parameters	Description
True positive	Instances where we predicted yes (patient has the CVD), and it turned out to be correct
True negative	Instances where a patient does not have CVD and was predicted to not have CVD
False positive	Instances where a patient does not have CVD, but was predicted to have CVD
False negative	Instances where a patient have CVD and was predicted to not have CVD

- Accuracy: the ratio of the correctly classified individuals to the total number of individuals, i.e. $\frac{TP+TN}{TP+TN+FP+FN}$

- Sensitivity: the probability of predicting the individual's class as CVD risk hazardous when it truly is CVD risk hazardous, i.e. $\frac{TP}{TP+FN}$.

- Specificity: the probability of predicting the individual's class as non-CVD risk hazardous when it truly is non-CVD risk hazardous, i.e. . $\frac{TN}{TN+FP}$

- Area under the curve (AUC): is an important feature of the ROC curve that measures the ability of a classifier to distinguish between classes. The greater the AUC, the better the model's performance

Where TP = True positive
TN = True Negative
FP = False Positive
FN = False Negative

Precision is the percent of the model's positive predictions that are correct. We define it as follows:

$$\text{precision} = \frac{\#\text{ positives predicted correctly}}{\#\text{ positive predictions}} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

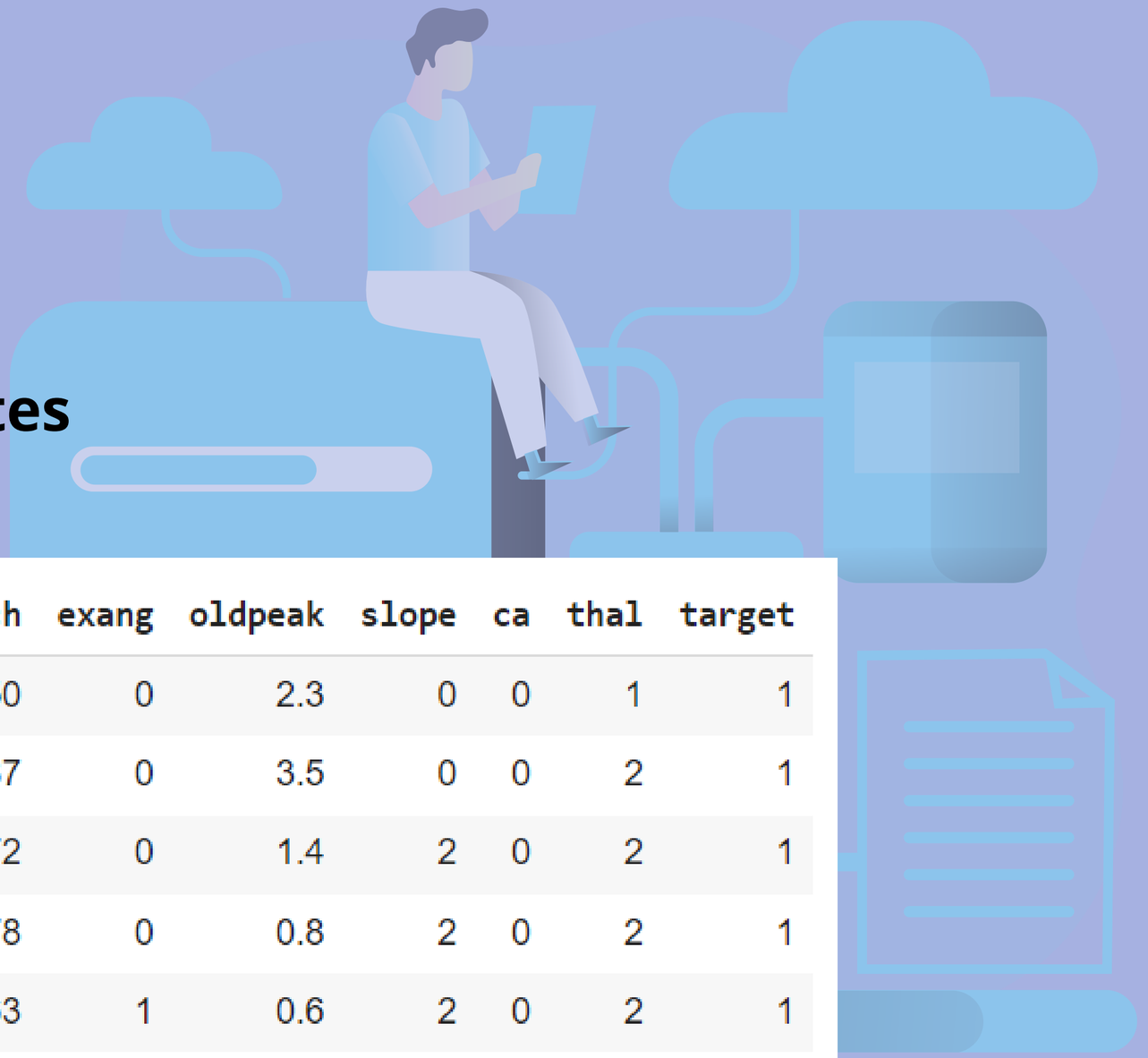
The sensitivity is another term for the recall, which is the true positive rate. Recall that it is calculated as follows:

$$\text{sensitivity} = \text{recall} = \frac{\#\text{ positives predicted correctly}}{\#\text{ positive cases}} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

The specificity is the true negative rate. It's calculated as follows.

$$\text{specificity} = \frac{\#\text{ negatives predicted correctly}}{\#\text{ negative cases}} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

DATASET



We have a dataset. It has 14 attributes and data of 300+ patients

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

```
heart_data["target"].value_counts()
```

```
1    165
0    138
Name: target, dtype: int64
```

Attributes

1. age
2. sex
3. cp - chest pain type (4 values)
4. trestbps - resting blood pressure
5. chol - serum cholestoral in mg/dl
6. fbs - fasting blood sugar > 120 mg/dl
7. restecg - resting electrocardiographic results (values 0,1,2)
8. thalach - maximum heart rate achieved
9. exang - exercise induced angina
10. oldpeak - ST depression induced by exercise relative to rest
11. slope - the slope of the peak exercise ST segment
12. ca - number of major vessels (0-3) colored by flourosopy
13. thal - 0 = normal; 1 = fixed defect; 2 = reversable defect
14. target - 0 = healthy ; 1 = heart defect

S No.	Source	No. of Patients	No. of Attributes	Data Size	Model Used	Performance			
						Accuracy (%)	sensitivity (%)	specificity (%)	AUC (%)
01	University of California Irvine	303	13	1200 lines	K-Nearest neighbour	73.77	94	74	86.21
02	LIMOSE Laboratory	400	14	1200 lines	Neural Network	84	97	64	93
03	ELSEVIER	329	13	1200	Support Vector Machine	92.1	91	66	95
04	National Library Of Medicine	410	14	1200	Decision Tree	79	90	66	89

S No.	Source	No. of Patients	No. of Attributes	Data Size	Model Used	Performance			
						Accuracy (%)	sensitivity	specificity	AUC (%)
05	BMC	300	13	1200 lines	Rain Forest	84	95	72	94
06	BMCF	400	14	1200 lines	HELLENIC-SCORE	85	97	70	93
07	HCGF	350	14	1200 lines	Gaussian Naive Bayes	81.48	95	73	91
08	KPU	380	14	1200 lines	LightGBM	82	79.9	79	90

S No.	Source	PPV(%)	NPV(%)	Model Used	Performance			
					sensitivity (%)	specificity (%)	P-Value	AUC(%)
09	Shihezi University School of China	23.5	96.2	Lasso-AdaBoost	73.09	74.10	0.09	79.8(78.2,81.3)
10	SUSC	27.4	96.5	FLR-L1-LR	73.49	78.86	0.17	81.7 (80.1, 83.2)
11	SUSC	23.0	97.0	FLR-RF	79.52	71.09	0.17	80.4(78.8,82.0)
12	SUSC	26.0	96.5	FLR-SVM	73.90	77.16	0.04	81.4(79.8,82.9)

The corresponding predicted CVD events/objective CVD events (P/O) values were 94.02, 94.00, 92.59 & 89.93 respectively

AUC, area under the receiver operating characteristic curve; PPV, positive predictive value; NPV, negative predictive value; Lasso-AdaBoost, AdaBoost with Lasso regression; FLR-L1-LR, L1 regularized Logistic regression with forward Partial Likelihood Estimation; FLR-RF, random forest with forward Partial Likelihood Estimation; FLR-SVM, support vector machine with forward Partial Likelihood Estimation.

Our Evaluation

Serial Number	Model Used	PPV(%)	NPV(%)	sensitivity (%)	specificity (%)	Accuracy(%)
1	Linear Regression Model	90.24	66.0	68.51	89.18	76.92
2	Multinomial Naive Bayes Model	72.72	75.0	77.41	70.0	73.77
3	Decision Tree Model	92.68	66.0	69.09	91.66	78.02
4	K-Nearest Neighbour	90.24	76.0	75.51	90.47	82.41
5	Artificial Neural Network	94.35	85.22	90.0	91.46	83.51

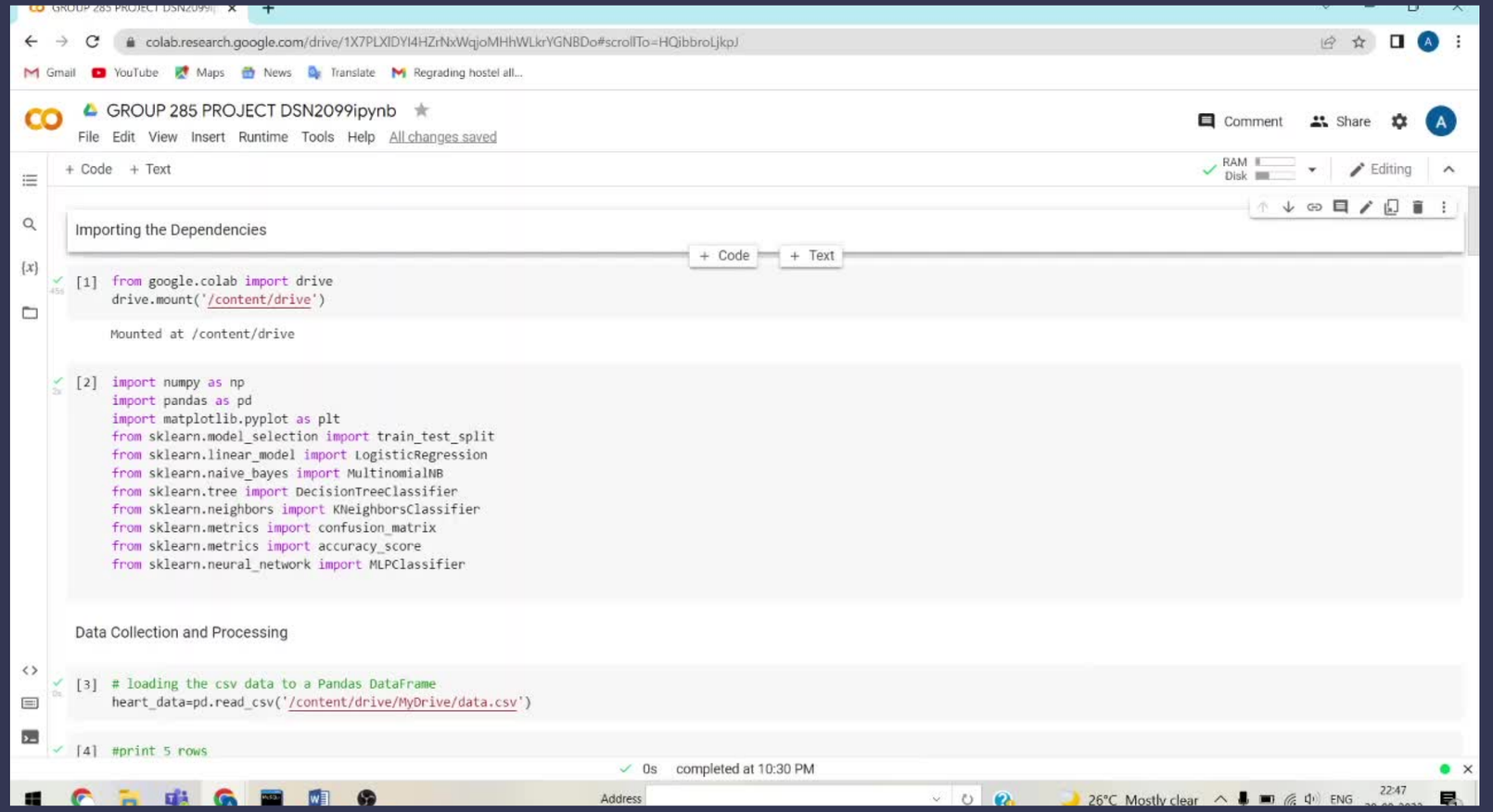
Comparison

Serial Number	Model Used	Existing Methods	Our Models
		Accuracy(%)	
1	Linear Regression Model	89.13	76.92
2	Multinomial naive bayes model	81.48	73.77
3	Decision Tree	79	78.02
4	K- Nearest Neighbour	73.77	82.41
5	Artificial Neural Network	84	83.51

(Data Size - 1200 lines in Research Paper)

(Data Size - 302 lines in our model)

Project Demo

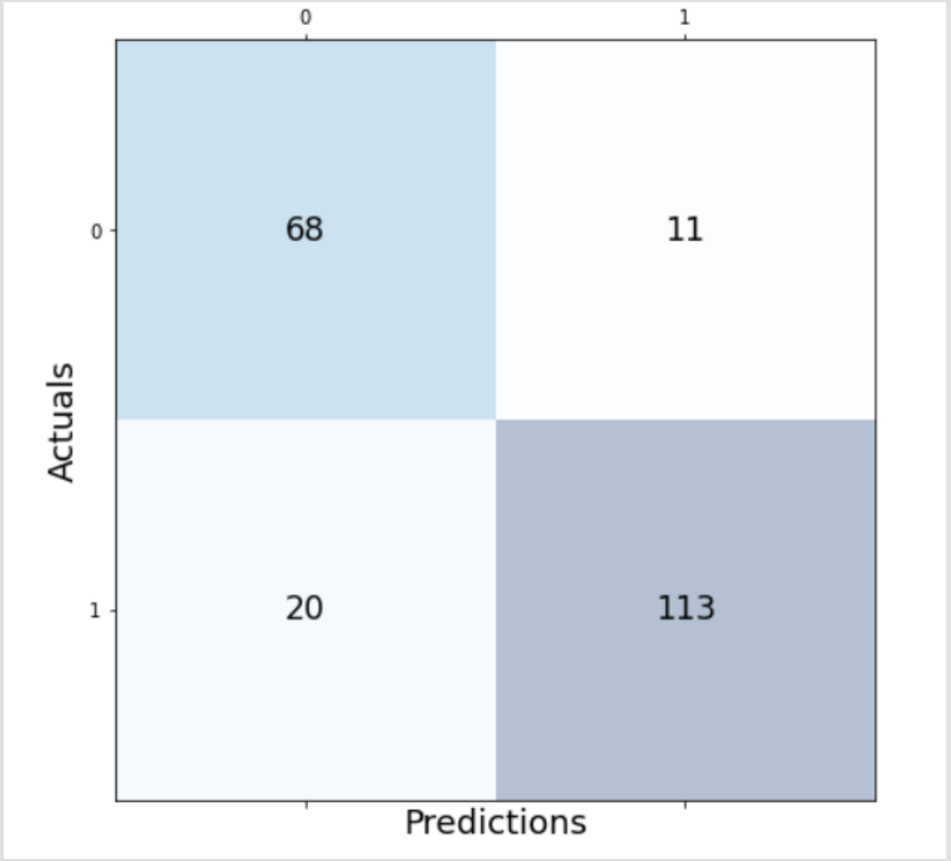


Please click the below link for watching the project demo

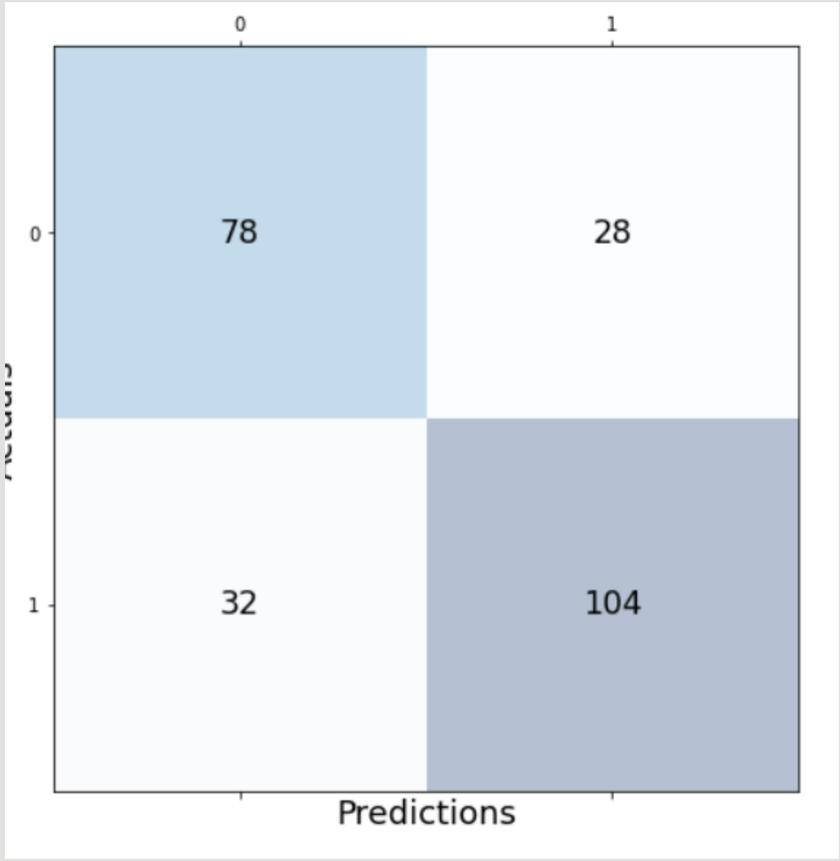
https://drive.google.com/drive/folders/1Fd6UCOn2Poij-TpdwzKe6jEpQrPIa16F?usp=sharing.

Confusion Matrices

1) Linear Regression Model

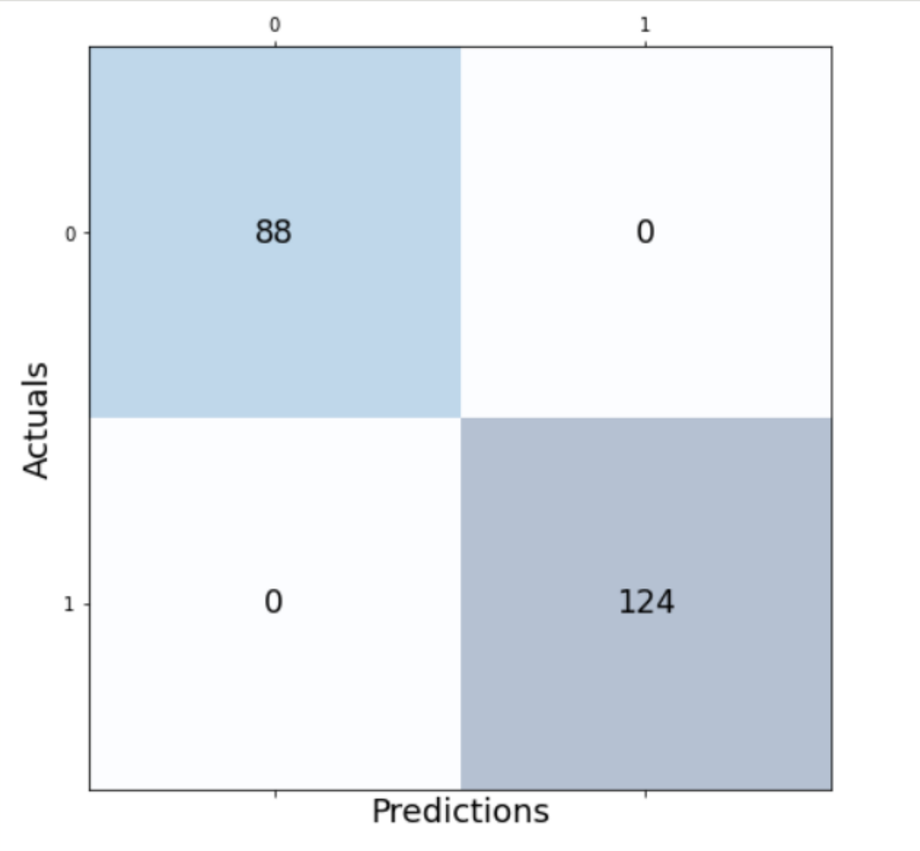


2) Multinomial Naive Bayes Model

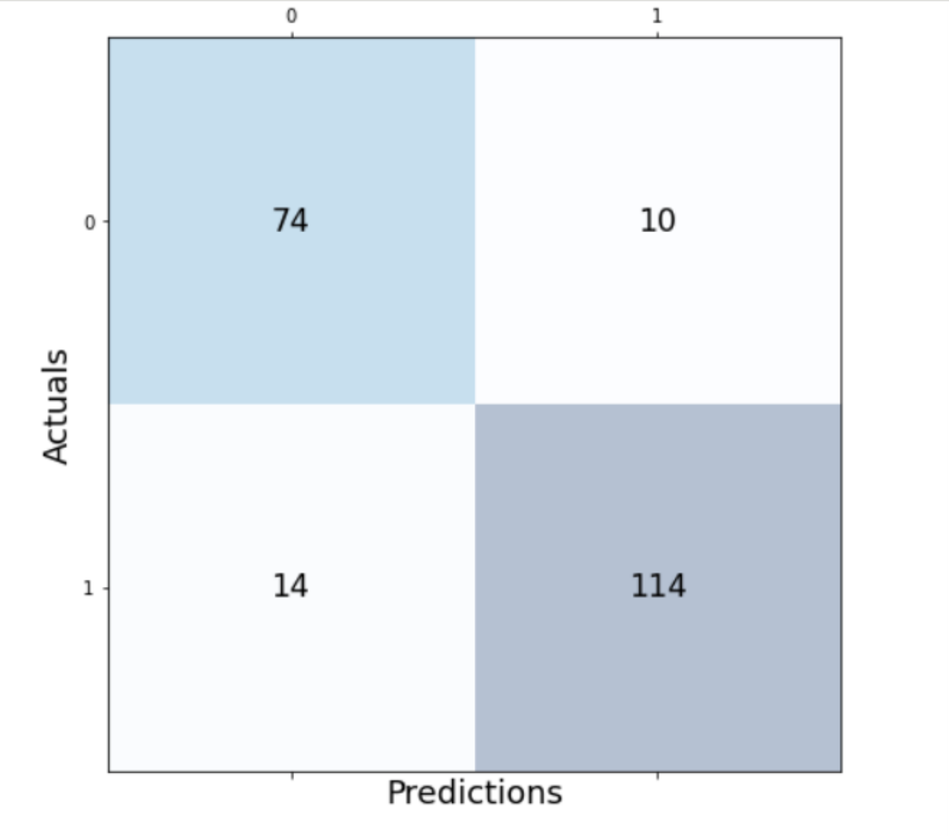


Confusion Matrices

3) Decision Tree Model

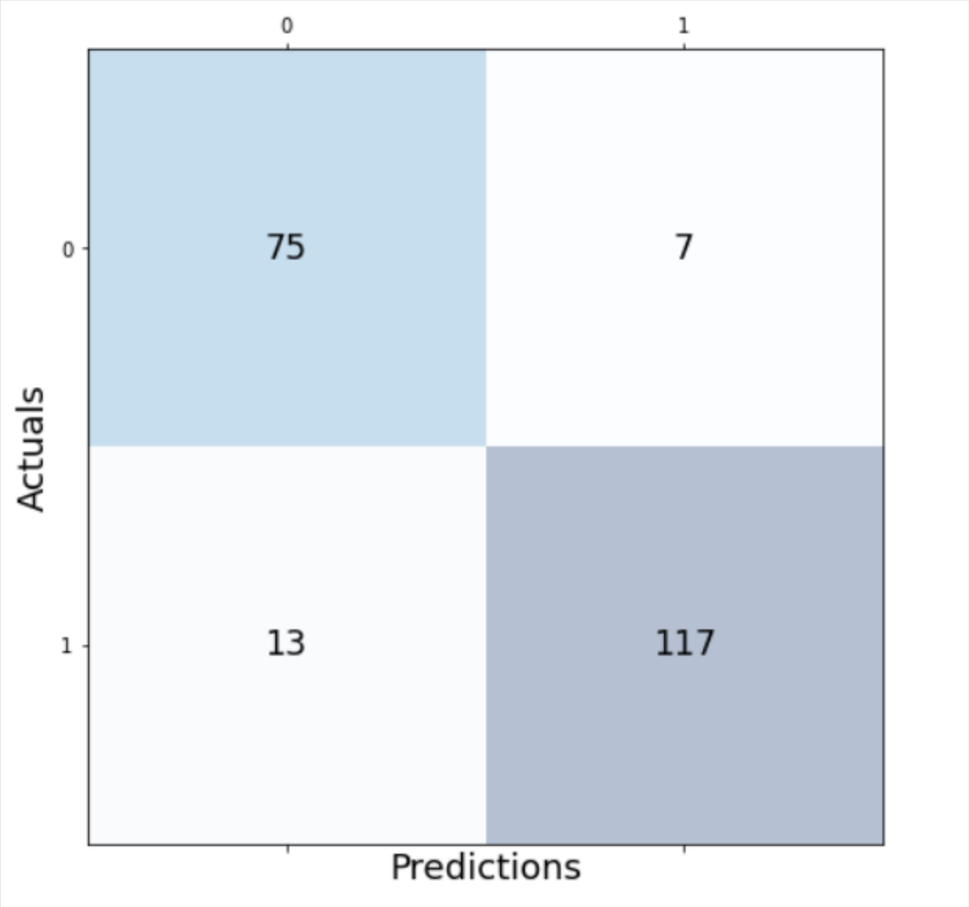


4) K-Nearest Neighbour



Confusion Matrices

5) Artificial Neural Network Model



Project-Snapshots

Importing the Dependencies

45s

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

2s

[2]

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import MultinomialNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
from sklearn.neural_network import MLPClassifier
```

Project-Snapshots

```
[3]  # loading the csv data to a Pandas DataFrame
     heart_data=pd.read_csv('/content/drive/MyDrive/data.csv')
```

```
[4]  #print 5 rows
     heart_data.head()
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

Project-Snapshots

Model Training

Logistic Regression Model

[ ] ↳ 25 cells hidden

Multinomial Naive Bayes Model

[ ] ↳ 25 cells hidden

Decision Tree Model

[ ] ↳ 24 cells hidden

K Nearest Neighbour Model

[ ] ↳ 24 cells hidden

Artifical Neural Network Model

[ ] ↳ 21 cells hidden

```
[117] print("Accuracy On Training Data : ", trainig_data_accuracy*100)

      Accuracy On Training Data :  85.37735849056604

[118] #accuracy on testing data
      X_test_prediction=LR.predict(X_test_std)
      testing_data_accuracy= accuracy_score(X_test_prediction,y_test)

[119] print("Accuracy On Testing Data : ", testing_data_accuracy*100)

      Accuracy On Testing Data :  76.92307692307693
```

Checking Accuracy using confusion matrix.

```
[120] from sklearn.metrics import confusion_matrix
      cm = confusion_matrix(X_train_prediction,y_train, labels=[0, 1])

      print(cm)

      [[ 68  11]
       [ 20 113]]
```

Splitting the Data into Training Data and Test Data

```
[193] X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,stratify=y,random_state=3)

[194] NB=MultinomialNB()

[195] NB.fit(X_train,y_train)

      MultinomialNB()
```

Modal Evaluation

Accuracy Score

```
[196] X_train_prediction=NB.predict(X_train)
      trainig_data_accuracy= accuracy_score(X_train_prediction,y_train)

[197] print("Accuracy On Training Data : ", trainig_data_accuracy*100)

      Accuracy On Training Data :  75.20661157024794
```

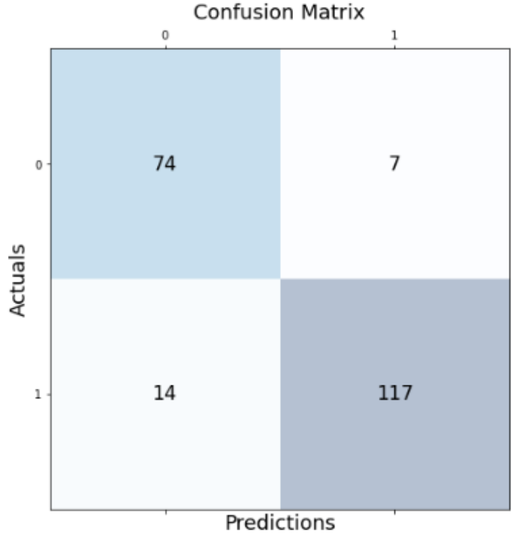
Input Data to Predict Risk

```
input_data=(21,1,1,130,210,0,0,200,0,1.8,2,0,0)

#change input data to a numpy array
input_data_as_numpy_array=np.asarray(input_data)

#reshape the numpy array as we are predicting for only one instance
input_data_reshaped=input_data_as_numpy_array.reshape(1,-1)
prediction1=LR.predict(input_data_reshaped)
print("Using Linear Regression Model")
print(prediction1)
if (prediction1[0]==0):
  print('The person does not have Heart Disease')
  print()
else:
  print('The person has Heart Disease')
  print()
```

```
Text(0.5, 1.05, 'Confusion Matrix')
```




```
tn, fp, fn, tp = confusion_matrix(list(X_train_prediction_nn), list(y_train_nn), labels=[0, 1]).ravel()

print('True Positive', tp)
print('True Negative', tn)
print('False Positive', fp)
print('False Negative', fn)
```

```
True Positive 117
True Negative 74
False Positive 7
False Negative 14
```

Results

We give our final verdict to that target which is supported by maximum number of models

```
Using Linear Regression Model
[1]
The person has Heart Disease

Using Multinomial Naive Bayes Model
[1]
The person has Heart Disease

Using Decision Tree Model
[0]
The person does not have Heart Disease

Using  K Nearest Neighbour Model
[1]
The person has Heart Disease

Using Artificial Neural Network Model
[1]
The person has Heart Disease

----------------------------------------------------------------
***FINAL VERDICT: The person has Heart Disease***
----------------------------------------------------------------
```

Citations

[1] Pal M, Parija S, Panda G, Dhama K, Mohapatra RK. Risk prediction of cardiovascular disease using machine learning classifiers. Open Med (Wars). 2022 Jun 17;17(1):1100-1113. doi: 10.1515/med-2022-0508. PMID: 35799599; PMCID: PMC9206502.

[2] A. Gavhane, G. Kokkula, I. Pandya and K. Devadkar, "Prediction of Heart Disease Using Machine Learning," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2018, pp. 1275-1278, doi: 10.1109/ICECA.2018.8474922.

[3] Latha, C. B. C., & Jeeva, S. C. (2019). Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques. Informatics in Medicine Unlocked, 16, 100203. https://doi.org/10.1016/j.imu.2019.100203

[4] Qian, X., Li, Y., Zhang, X., Guo, H., He, J., Wang, X., Yan, Y., Ma, J., Ma, R., & Guo, S. (2022). A Cardiovascular Disease Prediction Model Based on Routine Physical Examination Indicators Using Machine Learning Methods: A Cohort Study. Frontiers in Cardiovascular Medicine. https://doi.org/10.3389/fcvm.2022.854287

[5] Dimopoulos, A.C., Nikolaidou, M., Caballero, F.F. et al. Machine learning methodologies versus cardiovascular risk scores, in predicting disease risk. BMC Med Res Methodol 18, 179 (2018). https://doi.org/10.1186/s12874-018-0644-1

[6] M. Dhilsath Fathima, S. Justin Samuel, R. Natchadalingam, V. Vijeya Kaveri. (2022) Majority voting ensembled feature selection and customized deep neural network for the enhanced clinical decision support system. International Journal of Computers and Applications 0:0, pages 1-11.

SUMMARY

- The ability to forecast cardiac disease is difficult and crucial in the medical industry. However, if the condition is identified at an early stage and preventive treatments are implemented as soon as feasible, the mortality rate can be significantly reduced.

- The proposed LR model approach is proved to be quite accurate in the prediction of heart disease.

Thank you!