Cardiovascular Disease Prediction

1st Jaykumar Patel
Data Science
Stevens Institute of Technology
NJ, USA
jpatel5@stevens.edu

2nd Juilee Thakur
Data Science
Stevens Institute of Technology
NJ, USA
jthakur3@stevens.edu

3rd Samyak Upare
Data Science
Stevens Institute of Technology
NJ, USA
supare@stevens.edu

***Abstract*—The prevalence of cardiovascular disease is rising quickly every day, making it crucial and alarming to anticipate such illnesses. It is challenging to make this diagnosis since it must be done quickly and precisely. Based on a number of medical criteria, the main focus of this research is on which patients have a higher risk of developing cardiovascular disease. We have created a technique to predict the likelihood that a patient would be diagnosed with cardiovascular disease based on their medical history. In order to forecast and categorize cardiovascular disease patients, we applied a variety of machine learning methods, including logistic regression. We modified the model's application to enhance the predictability of a heart attack for each individual using a very helpful approach.**

keywords - cardiovascular, logistic regression, SVM, decision tree, KNN, Naive Bayes

I. Introduction

Cardiovascular disease has gotten a lot of attention in medical research among the various disorders that are life-threatening. A difficult and expensive is task, the diagnosis of cardiovascular disease. This program automatically forecasts the patient's heart status and ending allows for additional treatment. Signs, symptoms, and a physical examination of the patient are frequently used to make the diagnosis of cardiac disease. To anticipate the severity of the patient's chance of acquiring cardiovascular disease, we would like to gather data pertaining to every aspect of our field of study and train the data in accordance with the machine learning algorithms. It is frequently used to categorize whether a patient has cardiovascular disease or is healthy using data mining techniques like logistic regression. The suggested system can extract precise hidden data patterns, relationships, and patterns related to cardiovascular disease from historical cardiovascular disease databases. It can also provide complex solutions to problems involving cardiovascular disease diagnosis. Making wise clinical decisions could therefore be advantageous for healthcare professionals.

Obtaining high-quality service at a reasonable cost remains a challenge the most pressing and difficult issue in healthcare establishments. Normally, the healthcare sector entails a large amount of data concerning patients, various disease diagnoses, resource management, and etc. Human services must break down this information or data further. Patients' treatment records can be recorded on a computerized system, and crucial information and queries about the hospital can be obtained utilizing machine learning methods. Clinical data mining employs categorization approaches that are critical in predicting the risk of a heart attack before it occurs. The classification approach can be arranged and used to generate a prediction that determines an individual's perception of being triggered by coronary heart disease. Therefore, we would like to collect data related to all elements of our field of study and train the data according to the machine learning algorithm proposed by to predict the severity of the patient's likelihood of developing cardiovascular disease.

It is recommended, data mining techniques such as logistic regression is often used to classify whether a patient is normal or having heart disease. The proposed system can abstract accurate hidden data patterns, patterns and relationships related to cardiovascular disease, from past heart disease databases. It can also answer complex questions about diagnosing cardiovascular disease. Therefore, it may be helpful for healthcare professionals to make intelligent clinical decisions.

II. Related Work

Mohan et al in his paper states that in order to process the raw healthcare data and provide a fresh and original discernment towards heart disease, machine learning techniques were used. If the disease is discovered in its early stages and preventative measures are implemented as soon as possible, the mortality rate can be significantly reduced. The characteristics of the suggested hybrid Random Forest (RF) and Linear Method technique are combined (LM). When it came to predicting heart disease, HRFLM showed to be quite reliable [1].

When combined with PCA, alternating decision trees have demonstrated exceptional performance, however in some other situations, decision trees have demonstrated exceptionally poor performance, which may be caused by overfitting. Because they employ numerous algorithms to address the issue of overfitting, Random Forest and Ensemble models have fared very well. When combined with PCA, alternating decision trees have demonstrated exceptional performance, however in some other situations, decision trees have demonstrated exceptionally poor performance, which may be caused by overfitting. There is still much need for study about how to manage high dimensional data and overfitting, however Random Forest and Ensemble models have done quite well since they address the issue of overfitting by using many algorithms [2].

Srinivas et al states that based on the estimated significant weightage, an effective method was developed for the extraction of significant patterns from heart disease data warehouses for the valuable prediction of heart attack. The frequent patternsterns with a value larger than a predetermined threshold were selected. The definition of three mining objectives is based on data exploration. All of these models were capable of providing complicated predictions for heart attacks. [3].

III. Our Solution

This subsection elaborates on the solutions proposed by us to the problem.

A. Description of Dataset

The data set that we are planning to use is from UCI Machine Learning Repository that contains 14 physical attributes based on the physical testing of a patient. In addition to having blood drawn, the patient also undergoes a quick exercise test.

The 14 attributes are:

age: age in years

sex: sex (1 = male; 0 = female)

cp: chest pain type
Value 0: typical angina
Value 1: atypical angina
Value 2: non-anginal pain
Value 3 asymptomatic

trestbps: resting blood pressure (in mm Hg on admission to the hospital)

chol: serum cholesterol in mg/dl

fbs: (fasting blood sugar >120 mg/dl) (1 = true; 0 = false)

restecg: resting electr-cardiographic results
Value 0: normal
Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of >0.05 mV)
Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria

thalach: maximum heart rate achieved

exang: exercise induced angina (1 = yes; 0 = no)

oldpeak = ST depression induced by exercise relative to rest

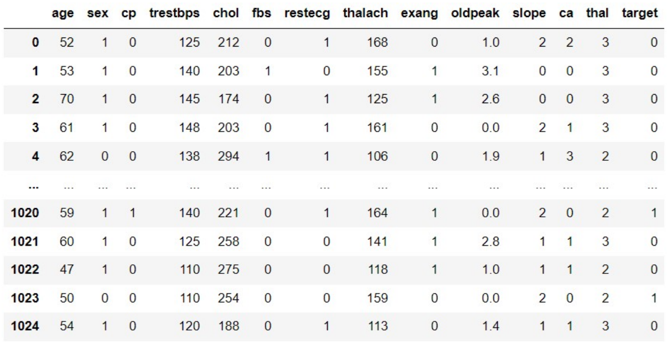
slope: the slope of the peak exercise ST segment
Value 0: upsloping
Value 1: flat
Value 2: downsloping

ca: number of major vessels (0-3) colored by flourosopy

thal: 0 = normal; 1 = fixed defect; 2 = reversable defect

target: 0 = no disease, 1 = disease

The website has data obtained from Hungarian Institute of Cardiology, University Hospital (Zurich), University Hospital (Basel) and V.A. Medical Centre, Long Beach and Cleveland Clinic Foundation.



	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	0
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1020	59	1	1	140	221	0	1	164	1	0.0	2	0	2	1
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3	0
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2	0
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2	1
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3	0

Fig. 1 Dataset

B. Machine Learning Algorithms

1) Logistic Regression:

This proposed system contains data that classifies whether a patient has heart disease or not according to some parameters. The proposed system could try to use this data to create a model that predicts whether a patient has this disease. The proposed system uses a logistic regression algorithm to calculate the score using the sklearn library. Logistic Regressions have proven to be the most popular and evolving field of machine learning in recent studies.

Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable in the provided dataset. Logistic regression is used to describe data and explain the relationship between dependent binary variables and one or more nominal, order, interval, or ratio-level independent variables. The logistic regression model does not conduct statistical classification (it is not a classifier); however, it can be used to create a classifier.

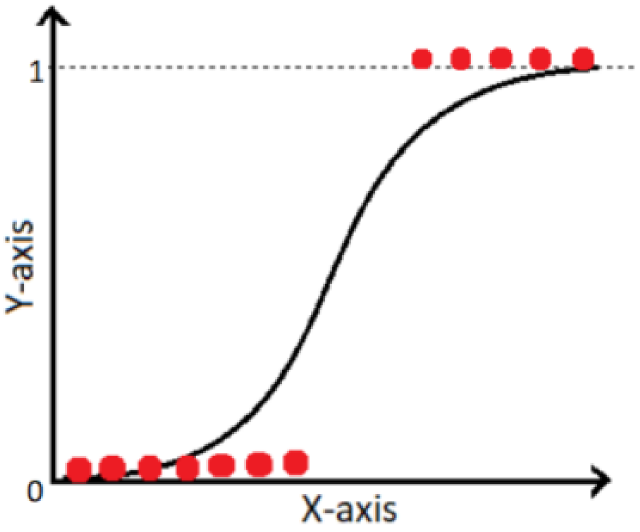


Fig. 2 Logistic Regression

2) Support Vector Machine (SVM):

Support Vector Machine is a very well-known managed AI technique (having a pre-characterized target variable) that can be utilized as a classifier just as an indicator. For characterization, it finds a hyper-plane in the element space that separates between the classes.

An SVM model addresses the preparing informative elements as focuses in the component space, planned in such a way that focuses having a place with discrete classes are isolated by an edge as wide as could really be expected. The test information focuses are then planned into that equivalent space and are arranged dependent on which side of the edge they fall.

SVMs can effectively do non-linear classification in addition to linear classification by implicitly mapping their inputs into high-dimensional feature spaces. This technique is known as the kernel trick.SVMs belong to a family of generalized linear classifiers and can be interpreted as an extension of the perceptron.The effectiveness of SVM depends on the selection of kernel, the kernel's parameters, and soft margin parameter λ.

Preprocessing (standardization) of data is strongly advised to improve classification accuracy. Standardization techniques include min-max, normalization by decimal scaling, and Z-score. SVM typically employs mean subtraction and variance division for each feature.

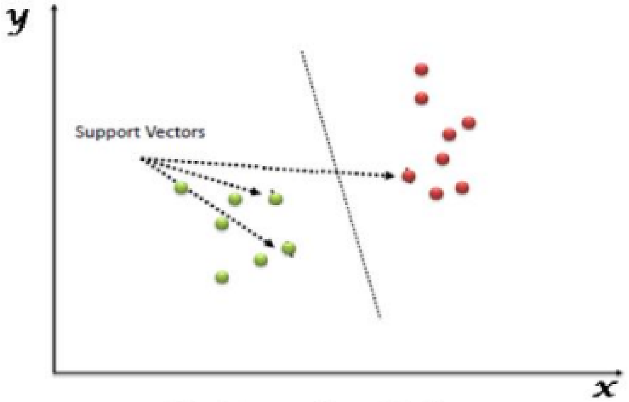


Fig. 1: Support Vector Machine

Fig. 3 SVM

3) K-Nearest Neighbor:

In 1951, one of the most popular technique for pattern classification known as KNN was introduced. K-Nearest Neighbor strategy is one of the most rudimentary yet exceptionally viable characterization strategies. It makes no presumptions about the information and is for the most part be utilized for grouping undertakings when there is exceptionally less or no earlier information regarding the information appropriation. This calculation includes tracking down the k closest informative elements in the preparation set to the item for which target esteem is inaccessible and doling out the normal worth of the observed information focuses to it.

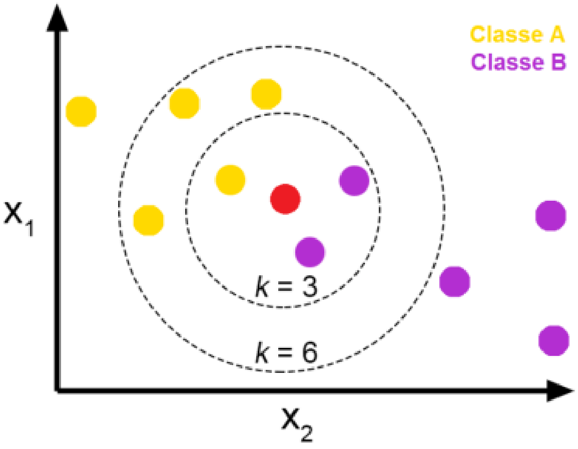


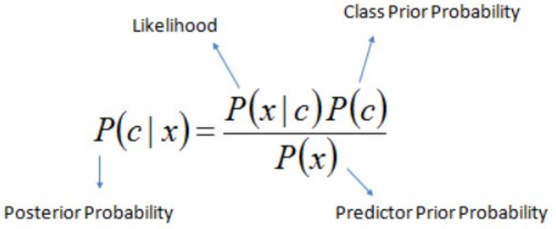
Fig. 4 KNN

4) Naive Bayes Algorithm:

It is a characterization strategy dependent on Bayes' Theorem with a presumption of freedom among indicators. In basic terms, a Naive Bayes classifier expects that the presence of a specific component in a class is inconsequential.

For instance, an organic product might be viewed as an apple on the off chance that it is red, round, and around 3 crawls in width. Regardless of whether these highlights rely upon one another or upon the presence of different elements, these properties freely add to the likelihood that this organic product is an apple and that is the reason it is known as 'Gullible'.

Innocent Bayes model is not difficult to assemble and especially helpful for exceptionally huge informational collections. Alongside effortlessness, Naive Bayes is known to beat even exceptionally complex characterization techniques. Bayes hypothesis gives a method of computing back likelihood P(c$| x) fromP(c), P(x) andP(x| c)$.


Likelihood
Class Prior Probability
$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$
Posterior Probability
Predictor Prior Probability

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Fig. 5 Naive Bayes

- $P(c|x)$ is the posterior probability of *class* (*target*) given *predictor* (*attribute*).
- $P(c)$ is the prior probability of *class*.
- $P(x|c)$ is the likelihood which is the probability of *predictor* given *class*.
- $P(x)$ is the prior probability of *predictor*.

5) **Decision Tree Classifier**:

Decision tree classifiers are supervised machine learning models. This means that they use prelabelled data in order to train an algorithm that can be used to make a prediction. Decision trees can also be used for regression problems. Much of the information that you'll learn in this tutorial can also be applied to regression problems.

Decision tree classifiers work like flowcharts. Each node of a decision tree represents a decision point that splits into two leaf nodes. Each of these nodes represents the outcome of the decision and each of the decisions can also turn into decision nodes. Eventually, the different decisions will lead to a final classification.

The diagram below demonstrates how decision trees work to make decisions. The top node is called the root node. Each of the decision points are called decision nodes. The final decision point is referred to as a leaf node.

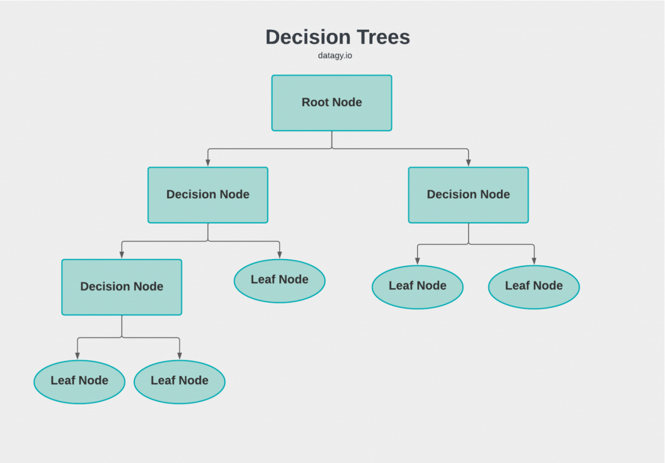
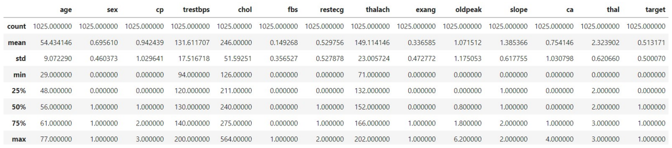


Fig. 6 Decision Tree

C. *Implementation Details*

This project was implemented in total 4 parts. First comes the pre-processing step, where all the missing and irrelevant data was removed. Missing values were causing incorrect visualization of some features. Considering this problem, all the missing or null values were removed successfully.

To get more information about the data we used the describe menthod to get some statistical information about the data set as below:



	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.00000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000
mean	54.434146	0.695610	0.942439	131.611707	246.00000	0.149268	0.529756	149.114146	0.336585	1.071512	1.385366	0.754146	2.323902	0.513171
std	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.527878	23.005724	0.472772	1.175053	0.617755	1.030798	0.620660	0.500070
min	29.000000	0.000000	0.000000	94.000000	126.00000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	48.000000	0.000000	0.000000	120.000000	211.00000	0.000000	0.000000	132.000000	0.000000	0.000000	1.000000	0.000000	2.000000	0.000000
50%	56.000000	1.000000	1.000000	130.000000	240.00000	0.000000	1.000000	152.000000	0.000000	0.800000	1.000000	0.000000	2.000000	1.000000
75%	61.000000	1.000000	2.000000	140.000000	275.00000	0.000000	1.000000	166.000000	1.000000	1.800000	2.000000	1.000000	3.000000	1.000000
max	77.000000	1.000000	3.000000	200.000000	564.00000	1.000000	2.000000	202.000000	1.000000	6.200000	2.000000	4.000000	3.000000	1.000000

Fig. 7 Dataset - Statistical Information

Some of the independent variables were irrelevant for further implementation of algorithms. Then after we analyze our features in the given dataset. We have some continuous data also available in the dataset. So, we apply normalization techniques to convert features into some standard scale. (We only applied normalization techniques to continuous features). We plot some graphs over the data which can be analyzed as follows.

First, we removed correlation between the variables and plotted a heatmap to check the dependancy between the variables.

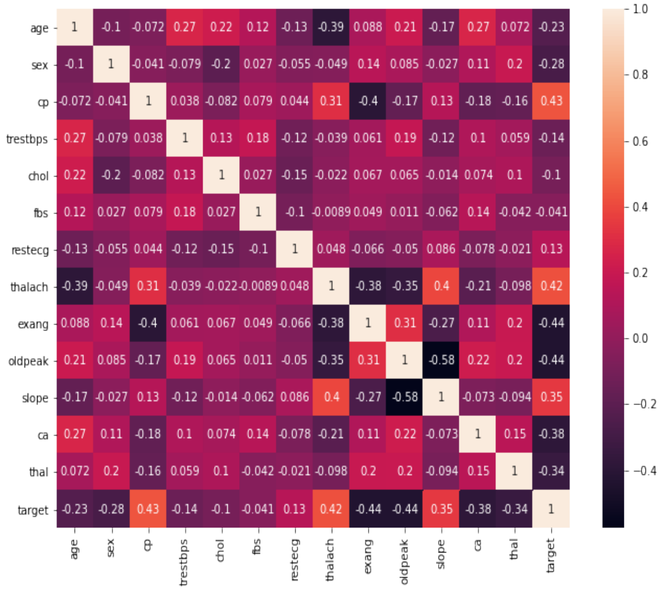


Fig. 8 Correlation Heatmap

Here, we could see clearly which variables have dependancy and how correlated each of them are.

We then plotted histograms of each column to see the frequency of data values and analyse the same.

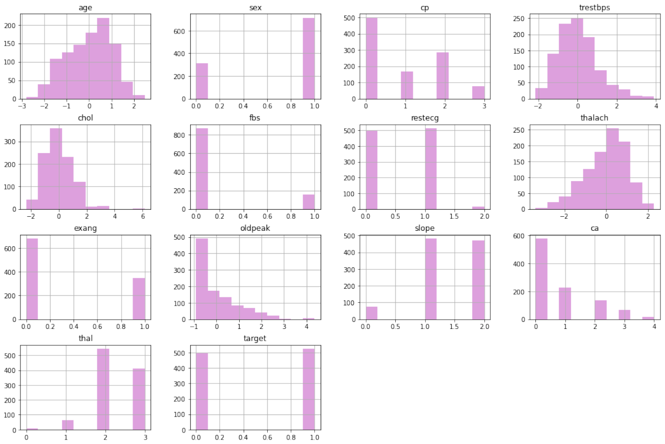


Fig. 9 Histograms of All Attributes

We then plotted two more graphs For Chest Pain and Sex Attributes so as to get a gist of the data values and also the changes on important attributes.

The graph for sex vs count below represents the count of male and female affected by the disease. Here,
— 0 represents the gender Female
— 1 represents the gender Male.

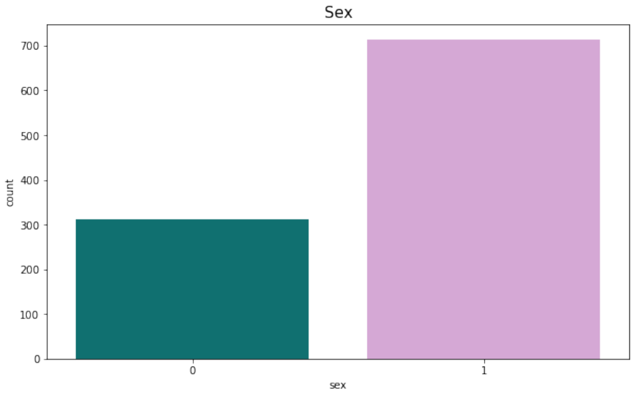


Fig. 10 Attribute Sex - Count

This graph is plotted based on the type of chest pain and the count of people suffering through it. It can be analyzed that the maximum number of people have chest pain type 0.

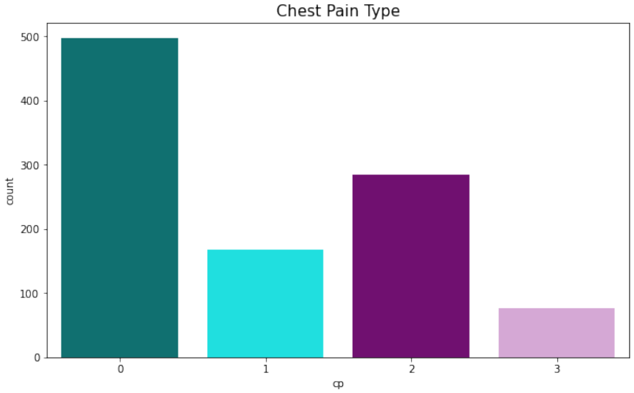


Fig. 11 Attribute Chest pain - Count

The below is a density vs age graph

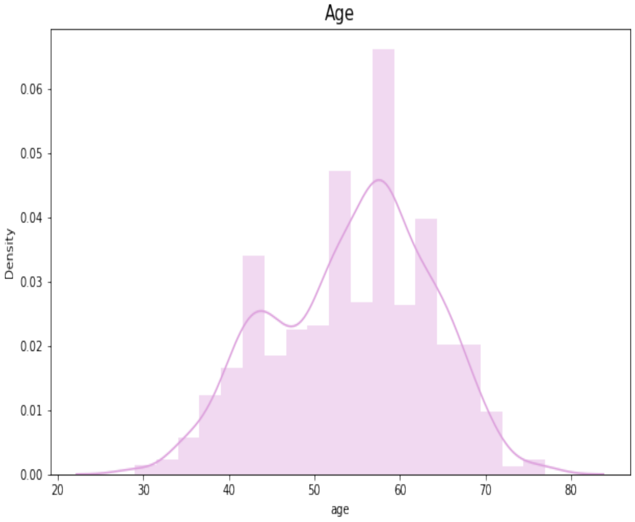


Fig. 12 Density vs Age

And here, this scattered line chart shows the analysis of Attack vs Age.

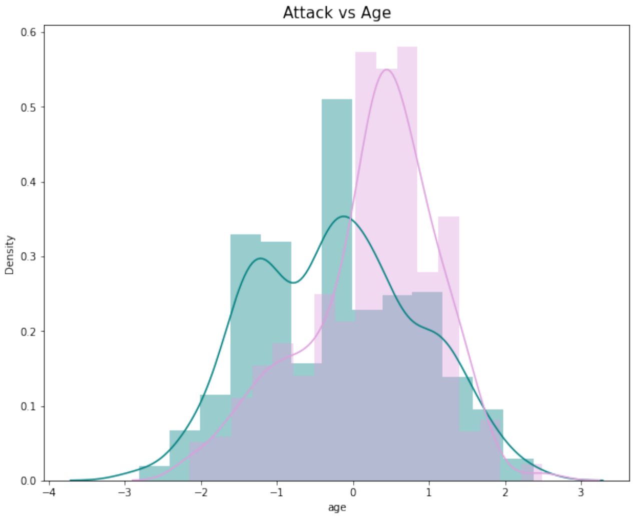


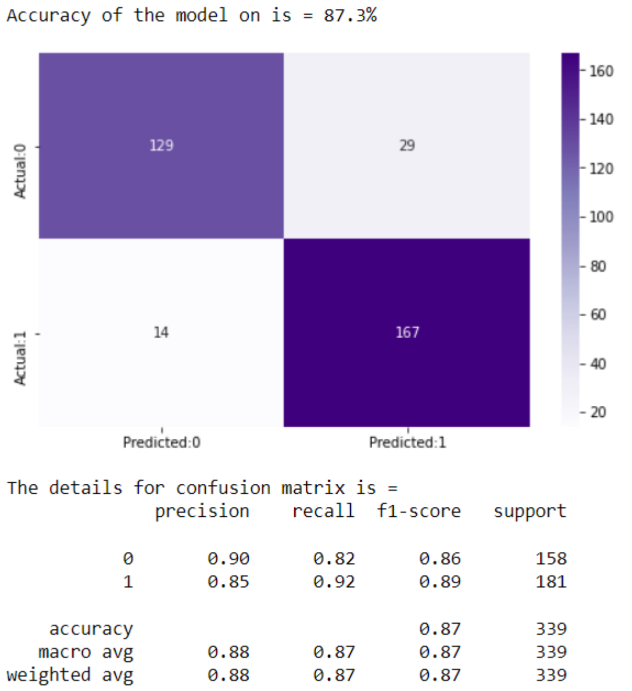
Fig. 13 Heart Attack vs Age

Now, we determine the outputs generated by using particular models:

1. Logistic Regression

According to our Logistic Regression model we get an accuracy of 87.3%.

Below is the output showing accuracy percentage and confusion matrix of the same and the classification report giving as valuable insight on the performance of the model


```
Accuracy of the model on is = 87.3%
```

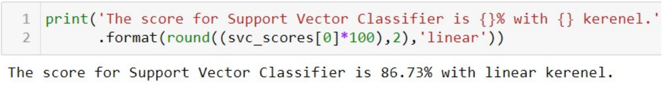
```
The details for confusion matrix is =
              precision    recall  f1-score   support

           0       0.90      0.82      0.86       158
           1       0.85      0.92      0.89       181

    accuracy                           0.87       339
   macro avg       0.88      0.87      0.87       339
weighted avg       0.88      0.87      0.87       339
```

Fig. 14 Output of Logistic regression

2. Support Vector Machines

Here, we have different choices accessible with portion like,”straight”, ”rbf”,”poly” and others (default esteem is

”rbf”). where we’ve utilized straight portionon component of heart dataset.



```
print('The score for Support Vector Classifier is {}% with {} kerenel.'
      .format(round((svc_scores[0]*100),2),'linear'))
```

```
The score for Support Vector Classifier is 86.73% with linear kerenel.
```

Fig. 15 Output of SVM

By applying the SVM algorithm we achieved an accuracy of 86.73%.

3. K-Nearest Neighbor

Picking the right incentive for K To choose the K that is ideal for your information, we run the KNN calculation a few times with various upsides of K and pick the K that diminishes the quantity of mistakes we experience while keeping up with the calculations capacity to precisely make the forecasts when it’s given information it hasn’t seen previously.

Here are a few things to remember: As we decline the worth of k to 1, our forecasts become less steady. Simply think briefly, envision k=1 and we have a question point encompassed by a few reds and one green (I’m contemplating the upper left corner of the hued plot above), however the green is the single closest neighbor. Sensibly, we would think the inquiry point is no doubt red, but since K=1, KNN inaccurately predicts that the question point is green. Contrarily, as we increment the worth of K, our expectation becomes steadier because of greater part casting a ballet/averaging, and hence, bound to make more precise forecast (in a limited way).

In the end, we start to observe an expanding number of mistakes. It is now we realize that we have driven the worth of k excessively far. In situations where we are taking a greater part vote (for example picking the mode in an arrangement issue) among names, we normally make K an odd number to have a sudden death round.

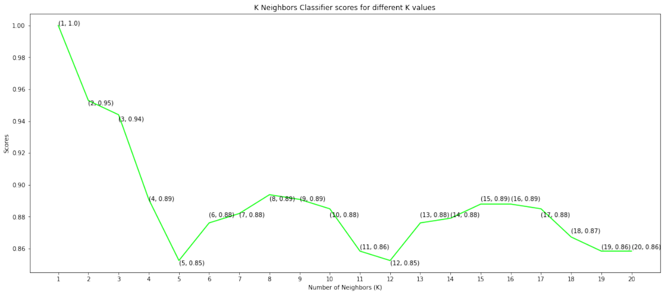
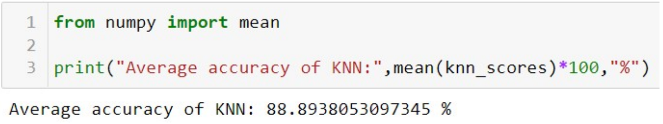


Fig. 16 Output of KNN

We calculated the average accuracy of all knn score and got an accuracy of 88.89% as shown below:



```
from numpy import mean

print("Average accuracy of KNN:",mean(knn_scores)*100,"%")
```

```
Average accuracy of KNN: 88.8938053097345 %
```

4. Naïve Bayes Algorithm

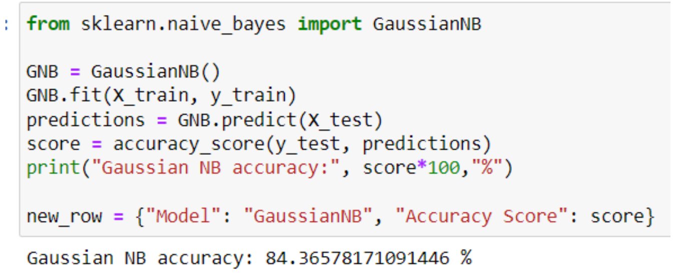
Performing the algorithm as below.

Stage 1: Convert the informational collection into a recurrence table

Stage 2: Create Likelihood table by observing the probabilities like Overcast likelihood = 0.29 and likelihood of playing is 0.64.

Step 3: Now, use Naïve Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

For our model, first we import all the dependencies Then, we use the classic prediction function on our testing dataset. Then, we implement our model using the sklearn Gaussiannm library to predict out accuracy.



```
from sklearn.naive_bayes import GaussianNB

GNB = GaussianNB()
GNB.fit(X_train, y_train)
predictions = GNB.predict(X_test)
score = accuracy_score(y_test, predictions)
print("Gaussian NB accuracy:", score*100,"%")

new_row = {"Model": "GaussianNB", "Accuracy Score": score}
```

```
Gaussian NB accuracy: 84.36578171091446 %
```

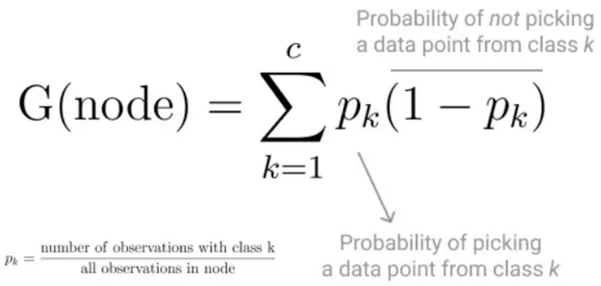
Fig. 17 Output of Naive Bayes Algorithm

5. Decision Tree Classifier

Decision trees work by splitting data into a series of binary decisions. These decisions allow you to traverse down the tree based on these decisions. You continue moving through the decisions until you end at a leaf node, which will return the predicted classification.

The algorithm uses a number of different ways to split the dataset into a series of decisions. One of these ways is the method of measuring Gini Impurity.

Gini Impurity refers to a measurement of the likelihood of incorrect classification of a new instance of a random variable if that instance was randomly classified according to the distribution of class labels from the dataset.



$$G(\text{node}) = \sum_{k=1}^{c} p_k(1 - p_k)$$

$$p_k = \frac{\text{number of observations with class k}}{\text{all observations in node}}$$

Probability of *not* picking a data point from class *k*
Probability of picking a data point from class *k*

Gini Impurity of a node.

Similarly to Gini Impurity, Entropy is a measure of chaos within the node. And chaos, in the context of decision trees, is having a node where all classes are equally present in the data.

Using Entropy as loss function, a split is only performed if the Entropy of each the resulting nodes is lower than the

$$\text{Entropy}(\text{node}) = -\sum_{i=1}^{c} p_k \log(p_k)$$

$$p_k = \frac{\text{number of observations with class k}}{\text{all observations in node}}$$

Probability of picking a data point from class k

Entropy of a node.

Entropy of the parent node. Otherwise, the split is not locally optimal.

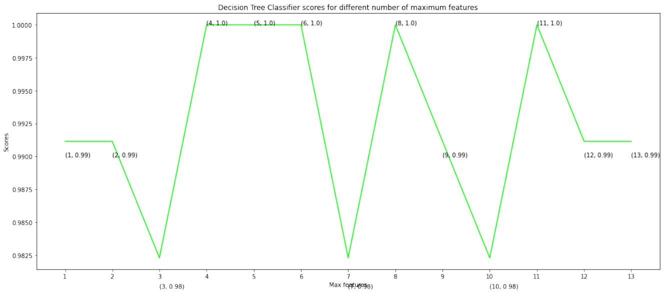


Fig. 18 Output of Decision Tree Classifier

Thus, here too we calculated the average of the decision tree classifier scores and obtained an accuracy of 99.25%.

IV. Comparison

We compare the accuracy of all the algorithms in order to judge their performance as below:

Sr No.	ML Algorithm	Accuracy
1	Logistic Regression	87.3%.
2	SVM	86.73%
3	KNN	88.89%
4	Naive Bayes	84.36%
5	Decision Tree	99.25%

Thus, we can clearly state that Decision tree performs way better than any another algorithm. But as we further analyse we get following insights too.

The "Loss Function" defines the best result in different loss functions. When we look at the optimization problem of linear SVM and Logistic Regression, they are very similar.

This means that they only differ in loss functions - SVM minimizes hinge loss while logistic minimizes logistic loss.

Now, Logistic loss diverges faster than hinge loss. So, in general, it will be more sensitive to outliers. Logistic loss does not go to zero even if the point is classified sufficiently confidently. This might lead to minor degradation in accuracy.

So, this theory states that we can expect the SVM to generate a more efficient accuracy as compared to logistic regression, which we achieved.

However, for larger datasets, we know that SVM gives higher accuracy for unstructured and semi-structured data while logistic regression gives better results for already identified independent variables.

Advantages and Disadvantages:

Logistic Regression : As we can see the advantage of using logistic regression is it is easier to implement, interpret and very efficient to train but we know that it can only be utilized to foresee discrete capacities. Thus, the reliant variable of Logistic Regression is bound to the discrete numberset.

SVM: The advantage of using SVM in our data set was it works well with a clear margin of separation, and it is effective in high dimension spaces. However, we know that the model doesn't perform well when we have a large dataset because it has a high required training time.

KNN : The best advantage associated with KNN is that the algorithm is versatile, it does not need several assumptions nor tuning of the model is required to run. However, the algorithm gets significantly slower as the number of examples increase.

Naive Bayes Algorithm: Naïve Bayes Algorithm is simple and quick to foresee class of test informational index. It likewise performs well in multi class expectation.

At the point when suspicion of autonomy holds, a Naive Bayes classifier performs better contrast with different models like calculated relapse and you want less preparing information.

If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as "Zero Frequency". To solve this, we can use the smoothing technique. One of the simplest smoothing techniques is called Laplace estimation.

On the other side naive Bayes is also known as a bad estimator, so the probability outputs from predict proba are not to be taken too seriously.

Another limitation of Naive Bayes is the assumption of independent predictors. In real life, it is almost impossible that we get a set of predictors which are completely independent.

Decision Tree Classifier: It can be used for both classification and regression problems: Decision trees can be used to predict both continuous and discrete values i.e. they work well in both regression and classification tasks.

An advantage of the decision tree algorithm is that it does not require any transformation of the features if we are dealing with non-linear data because decision trees do not take multiple weighted combinations into account simultaneously.

On the other hand, The time complexity right for operating this operation is very huge keep on increasing as the number of records gets increased decision tree with to numerical variables takes a lot of time for training.

A decision tree generally needs the overfitting of data. In the overfitting problem, there is a very high variance in output which leads to many errors in the final estimation and can show highly inaccuracy in the output. Achieve zero bias (overfitting), which leads to high variance.

V. Future Directions

As we have done Cardiovascular disease prediction, similar prediction system can be built for various other fatal disease such as cancer, diabetes etc. using recent technological advancement like machine learning algorithms, image processing and a variety of other. Also, new algorithms can be proposed to achieve more accuracy and reliability such as decision tree, Linear Discriminant Analysis as well as Real time patient monitoring system using random forest for disease prediction. Hidden naive bayes can and deep learning algorithms such as LSTM and RNN also be proposed in order to improve accuracy and dependability. Also, big data technologies such as Hadoop Distributed File System can be used to store large amounts of data from all users across the world and to manage the data or reports of those users. Other tools like cloud,mongo DB can also be employed.

VI. Conclusion

We can conclude that among the machine learning algorithms such as logistic regression, SVM and KNN, NB and decision tree, we found out that decision tree had the best accuracy considering the performance which is an accuracy of 99.25%. Cardiovascular disease prediction system is proposed to identify the risk of heart disease accurately. Also, the algorithm gives the nearby reliable output based on the input provided by the users. Hence, if the number of people using the system increases, then the awareness about their current heart status will be known and the rate of people dying due to heart diseases will reduce eventually.

References

[1] Senthilkumar Mohan, Chandrasegar Thirumalati, Gautam Srivastava,"Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques", IEEE Access.

[2] V.V. Ramalingam, Ayantan Dandapath, M Karthik Raja, "Heart disease prediction using machine learning techniques: a survey", International Journal of Engineering Technology, 7 (2.8) (2018) 684-687

[3] K.Srinivas, Dr.G.Raghavendra Rao, Dr. A.Govardhan, "Analysis of Coronary Heart Disease and Prediction of Heart Attack in Coal Mining Regions Using Data Mining Techniques", The 5th International Conference on Computer Science Education Hefei, China, August 24–27, 2010.