ISSN No:-2456-2165

Ensemble Prediction Model on Cardio Vascular Disease

Vidyashree K ., Thippeswamy K
Department of PG Studies, VTU Mysore, India

Abstract:- Cardiovascular diseases, normally address all kinds of diseases associated with the heart and is being treated globally as the main cause of mortality. Numerous risks are associated with heart diseases, a need of the hour is to get worth, reasonable and accurate early diagnosis method and treatment. Machine Learning is being used in many areas to solve such problems. The aim of this project is to predict the heart disease in individuals. The use of many classification algorithms in machine learning (ML) on standard dataset has revealed that there is a need to improve the accuracy as taking risk in heart related diseases is not acceptable. The findings show that highest accuracy is achieved through Decision Tress Classifier (93%) and combining more than one method is being used on same dataset in all possible combinations and achieved an accuracy of 98.1% keeping in mind, the time complexity of final algorithm.

Keywords:- *Supervised Learning, Classifier, Cardio Vascular Disease, Classificiation, Confusion Matrix.*

I. INTRODUCTION

Cardiac disease has been detected because of diabetes, high cholesterol, irregular pulse and. due to many health problems such as high BP. Also, many more reasons are there for heart diseases. It's difficult to associate in particular which feature is related to heart disease when so many features contribute to it variably. The various stages of heart disease have been identified using various techniques. The Genetic Algorithm (GA), Decision Tree (DT), and Support Vector Classification algorithms [1, 2] are just a few of the methods used to categorise the severity. Complexity of cardiac disease points for treatment with higher caution. Accuracy achieved when we use any single method is not acceptable as the wrong decision in such cases may cause serious issue with patients. Hence, with the objective of improving the accuracy, multiple supervised methods are combined and called as Ensemble method. All the possible combinations reveal the best Ensemble method.

Analysis of Data is quite important for the prediction of heart diseases. Number of analysis are carried out in this project to develop a prediction model, not just using different ML methods but also combining every two techniques. More than two methods can also be combined but all machine learning algorithms take lot of processing time and merging many will result in longer time to compute. Hence, here only two methods are being merged.

Lot of studies are done, and number of ML models are deployed, multiple but two methods are merged to get the best accurate algorithm everything with the aim of precisely diagnosing and forecasting diagnoses of cardiac disease. ANN is used to predict cardiac disease using a multilayer back propagation perceptron (MLP). In comparison to other models in the field, the results are shown to be better [7]. DT, NN, SVMs, and NBs, among other tools, are used to check trends in data from UCI on patients with cardiac disease. Different algorithms' performance and accuracy are evaluated. The accuracy of the suggested hybrid strategy, which is 98.1 percent, is comparable to that of existing techniques. During the training phase, the CNN technique takes into account cardiac cycles with various start points determined from ECG data. During the "patient's testing phase," CNN is capable of producing features with shifting positions [9,10,11,12]. Many machine learning techniques have been discovered by Goland et al. that can be used to categorise cardiac disease [13]. Adaboost, GBC, K-Means, and KNN algorithms that can be used for classification have all been studied to determine their accuracy [13]. According to this study, DT provided the highest accuracy, followed by a mix of various approaches. A system that recommends data mining methods in conjunction with map reduction strategies was put forth by Nagamani et al. [14]. In comparison to the accuracy acquired using the traditional fuzzy ANN, a greater accuracy was attained for the 55 data sets in the test set. The method's accuracy was improved by using linear scales and dynamic schemes. By contrasting five various approaches, Alotaibi created a machine learning model [15]. We utilise a quick miner that performs more accurately than MATLAB and Weka. Here, the accuracy of the classification algorithms DT, LR, ABC, KNN, GBC, MLP, GNB, and SVM is compared. The DT algorithm that is most precise. A system that combines the Naive Bayes approach for data set classification and AES for trustworthy data transfer for illness prediction was created by Repaka et al. [16]. Performance metrics included mean absolute error, sum squared error, and root mean squared error. SVM has been shown to provide better accuracy than Naive Bayes [18]. KNN with SVM, KNN with DT, etc. to achieve better accuracy with index data analysis and complexity estimation. The use of ML combination of several methods is used.

The suggested system's goal is to build a diagnostic system with the aid of a computational infrastructure. All classification methods' accuracy levels were contrasted. DT, RF, and LR emerged as the best classification methods for predicting heart disease. This study is novel because we used several methods and achieved an accuracy of 98.1%,

higher than the previous claim (92%). Therefore, we aim to achieve the following goals with this work. Implementation of different ML algorithms on a standard dataset.

- Analysis of results obtained by all individual ML methods with every parameter related and finding the accuracy through confusion matrix.
- Combining two methods in every combination to achieve higher accuracy and analysing it.

II. DATASET

Though we got four different heart disease datasets, we used only the UCI Cleveland dataset [20]. There are 76 characteristics in total in this dataset, most works utilise only a subset of 14 dominating features [21]. As a result, we used the UCI Cleveland dataset that had already been processed and was available on the "Kaggle" website. The 14 characteristics utilised in the suggested work are described in Table 1.

A total of 165 cardiac databases and 138 non-cardiac databases are present in the target column. The target column is displayed in Figure 1.

Sl.No.	Attributes
1	"age: age in years"
2	"sex: sex (1 = male; 0 = female)"
3	"cp: chest pain type -- Value 0: typical angina -- Value 1: atypical angina -- Value 2: non-anginal pain -- Value 3: asymptomatic"
4	"trestbps: resting blood pressure (in mm Hg on admission to the hospital)"
5	"chol: serum cholestoral in mg/dl"
6	"fbs: (fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)"
7	"restecg: resting electrocardiographic 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"
8	"thalach: maximum heart rate achieved"
9	"exang: exercise induced angina (1 = yes; 0 = no)"
10	"oldpeak = ST depression induced by exercise relative to rest"
11	"slope: the slope of the peak exercise ST segment -- Value 0: upsloping -- Value 1: flat -- Value 2: downsloping"
12	"ca: number of major vessels (0-3) colored by flourosopy"
13	"thal: 0 = normal; 1 = fixed defect; 2 = reversable defect and the label"
14	"condition: 0 = no disease, 1 = disease"

Table 1 : Dataset Attributes

Three sections make up the remaining paper. Section 2 presents the approach and techniques, Section 3 presents the findings and analysis, and Section 4 presents the conclusion and future directions.

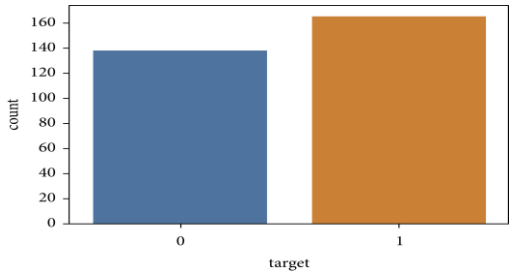

Fig. 1: Visualization of the target column

Generality of model will suffer, if we don't verify and handle the similar and null data. If duplicates are not handled effectively, then duplicates may appear in both the training & test datasets. So, elimination of all duplicate data is to be done during the pre-processing phase. Figure 2 confirms about no missing data in dataset considered. 0 values in above figure 2 is because of dataset for having no missing data.

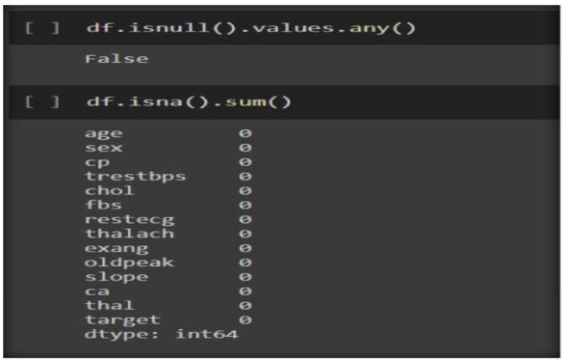

Fig. 2: Zero missing data

III. SYSTEM DIAGRAM IN SCHEMATIC

The above-mentioned classification techniques are examined in the proposed study, together with performance analysis, to identify heart disease. This test's objective is to correctly forecast whether or not a patient has cardiac disease. The information was fed into a model that foresaw the likelihood of developing heart disease. The system's schematic is shown in

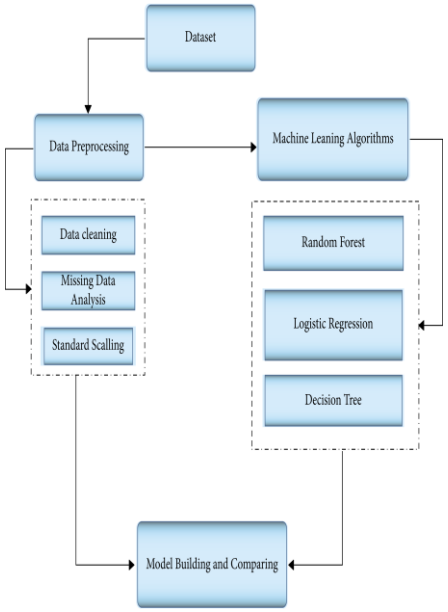

Fig. 3: System's Schematic Diagram

"The properties shown in Figure 3 are random forest, decision tree, logistic regression, etc. used as input for classification methods such as input dataset divided into 90% training dataset and 10% testing dataset. A training dataset is a collection of data used to train a model. The test data set is also used to evaluate the performance of the trained model. The performance of each method is determined and analyzed using several measures, including precision, accuracy, recall, and F1 score, as described below."

A. Machine Learning Algorithms

Let us look at different algorithms of Machine Learning used here. Following nine ML methods are considered for implementation in this work.

- "Random Forest
- Decision Tree
- Logistic Regression
- K-Nearest Neighbour
- AdaBoost classifier
- Gaussian Naive Bayes
- Gradient Boosting
- MLP Classifier
- Support Vector Classifier

There are two steps to generate a random forest: the first is generating a random forest, and the second is clustering for prediction. Figure 4 shows a diagram of the random forest algorithm."

Random forest is also a tree-based decision method. A single decision tree is more precise and trustworthy than a joint one. Edge of DT, random forest. Although DT is more accurate, RF is better against optimization. It uses DT as RF pocket model.

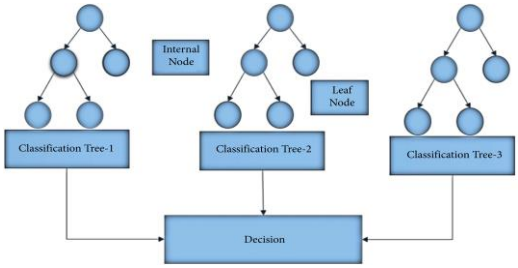

Fig. 4: Schematic diagram of Random Forest

The decision tree method is described as a flow chart showing the database attributes of the main point and the outer branches representing the results. It was chosen for its speed, reliability and simple reading and less data preparation. In DT, the root of the tree is the starting point for predicting class labels. Depending on the outcome of the comparison, the comparison branch is further analyzed to determine the value that will be provided to the next button. A schematic representation of the decision tree method is shown in Figure 5.

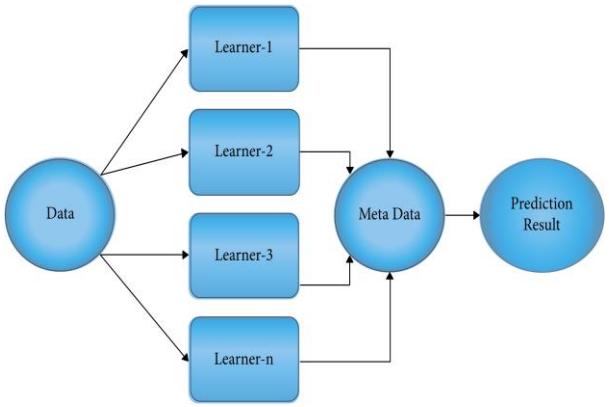

Fig. 5: Schematic diagram of decision tree

The accuracy of the decision tree depends on the strategic allocation. Different decision criteria are used by them. The growth of sub-points increases its homogeneity. As the target variable increases, the point purity increases. DT can handle category and numerical data and is easy to grasp.

A statistical method for addressing binary classification issues is logistic regression. Instead of using a hyperplot or a straight line, logistic regression (LR) categorises the result of a linear equation by limiting it to a range between 0 and 1, using 13 independent variables. A conceptual representation of the logistic regression technique is shown in Figure 6.

ISSN No:-2456-2165

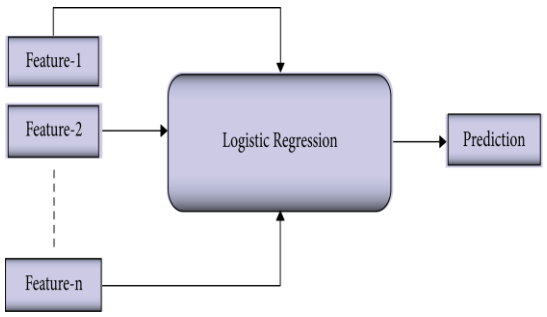

Fig. 6: Schematic diagram of Logistic Regression

One of the simplest machine learning methods for supervised learning is nearest neighbour. The K-NN method takes into account how similar new state/data are to old state/data, and new state is more similar to old category. Problems involving classification can be solved using the K-NN method. The non-parametric K-NN method makes no assumptions on the underlying data. Because it keeps a database and performs classification operations on the database rather than learning directly from the training set, this method is also known as a lazy learner.

The AdaBoost classifier is a meta-estimator that starts with the actual implementation of the classifier and then matching additional instances of the classifier to the same database, but with the weight of misclassified cases adjusted, the classifiers then focus on more difficult cases.

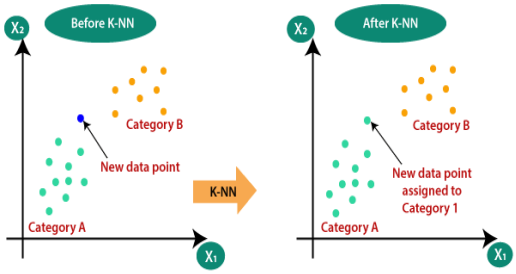

Fig. 7: Schematic diagram of KNN

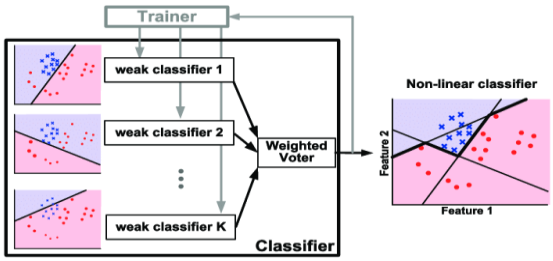

Fig. 8: Schematic diagram of Adaboost

Properties and continuous-valued models that make use of a Gaussian (normal) distribution are supported by Gaussian Naive Bayes. Assuming that the data are characterised by a Gaussian distribution with no variations between observations is a straightforward method for building models (independent measurements). The effectiveness of the Gaussian Naive Bayes (GNB) classifier is seen in Figure 9. The distance from the class divided by the class's standard deviation is used to calculate the z-score interval for each data point, which is the distance between the point and each class.

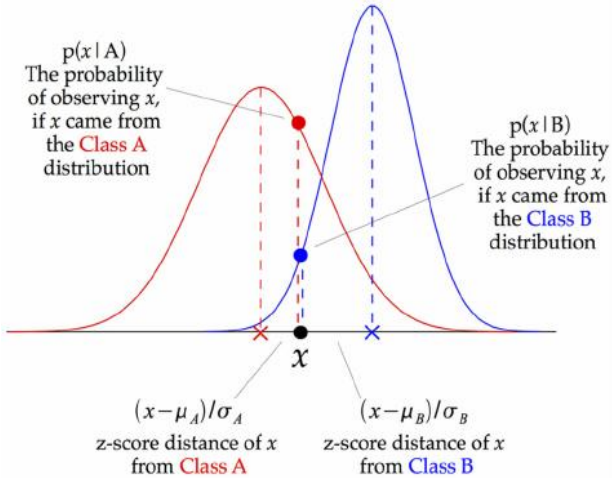

Fig. 9: Working of Gaussian Naive Bayes Classifier

Gradient Boosting is an iterative functional gradient algorithm, which is an algorithm that reduces the loss function by repeatedly choosing a function oriented to a negative gradient; weak hypothesis. AdaBoost and Gradient Boost continue to learn from weak learners. Strong pupils are formed from additive models of such weak pupils. The main goal here is to learn from mistakes at each step of the iteration.

AdaBoost requires the user to define a set of weak learners. It increases the weight of incorrectly predicted events and decreases correctly predicted events. A weak learner tends to focus more on difficult situations. After training, weaker students are paired with stronger ones based on their performance (called alpha weights). The better he performs, the higher the contribution of the strong student.

Increasing the gradient will not change the sampling distribution. Conversely, weak students learn the remaining mistakes from strong students. He seems to give more importance to difficult situations. In each iteration, the pseudo-residual is calculated and the student's weakness is fitted to this pseudo-residual. Then, the contribution of weak learners to strong ones is not calculated based on their performance in the new distributed model using the gradient descent optimization process. The calculated contribution is the one that minimizes the total power learned error. Adaboost is more about "boosting weights and gradients" than "adding gradient optimization".

A multi-layer Perceptron classifier connected to a neural network is called an MLP classifier. Instead of using

a support vector or naive bayes classifier, it uses an underlying neural network to carry out the classification operation.

The Support Vector Machine (SVM) algorithm's goal is to produce the optimal line or decision boundary that can categorise the n-dimensional space, making it simple to later assign fresh data points to the appropriate groups. A hyperplane is the name given to this decision limit.

SVM chooses the highest vector or point that contributes to the creation of the hyperplane. The approach is known as a support vector machine since this extreme instance is known as a support vector.

B. Diagram of the Confusion Matrix in Blocks

The "confusion matrix is a way to describe the performance of a classification method. The number of correct and false guesses is added and displayed by the score value. This is the key to understanding precision through the matrix. A block diagram of this matrix is shown in Figure 10.

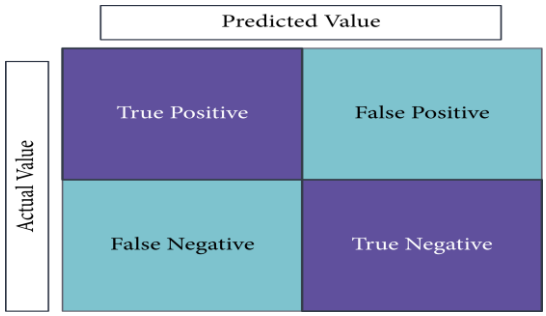

Fig. 10: Block diagram of the confusion matrix

This clarifies not only the error made by the classifier, but also the type of error made. The expected row and the predicted column for the class contain the total number of correct predictions."

IV. RESULT AND DATA ANALYSIS

Here, we discuss modeling ability, assumptions, findings, & conclusions. With the exception of old peaks, most continuous variables exhibit minor skews (to the left or right) and are near to the Gaussian distribution. Varied attributes have different consequences on CVD. Once more, there are cholesterol-related standouts that merit further investigation.

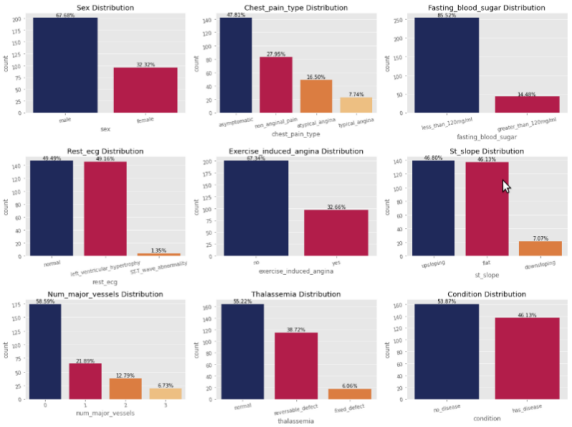

Fig. 11: Features impact on CVD

A. Data Visualization

A histogram distributes repetitions with infinite classes. This is a figure that crosses class boundaries and has an area related to the frequency of the comparison class. The relative class frequency and the repetition density for various classes are related to the squared value. The distribution of age, blood pressure, cholesterol, heart rate, and peak age is shown in Figure 12. The distribution of age, blood pressure, cholesterol, heart rate, and old peak is shown in Figure 13.

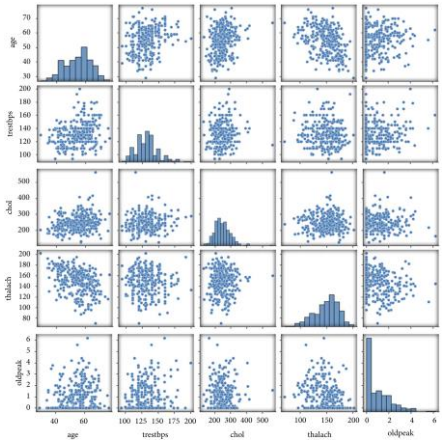

Fig. 12: Distribution of the main features of the dataset

Figure 13 shows the heart health of persons of different ages and demonstrates that cardiovascular illness is not present in those under the age of 35. As you become older, your risk of acquiring cardiovascular disease increases. The presence of heart illness is indicated by Target 0. The gender-specific disease status is shown in Figure 14.

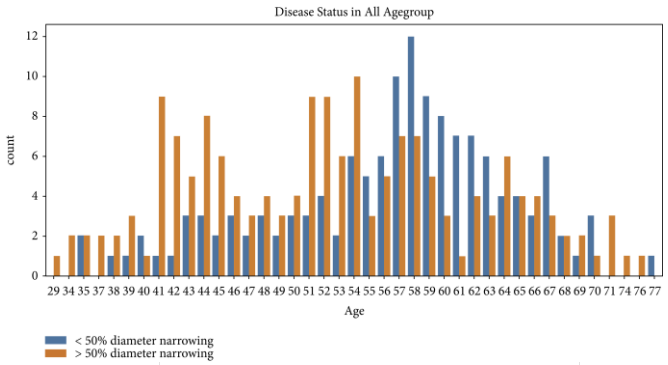

Fig. 13: Cardiac state of people of various ages.

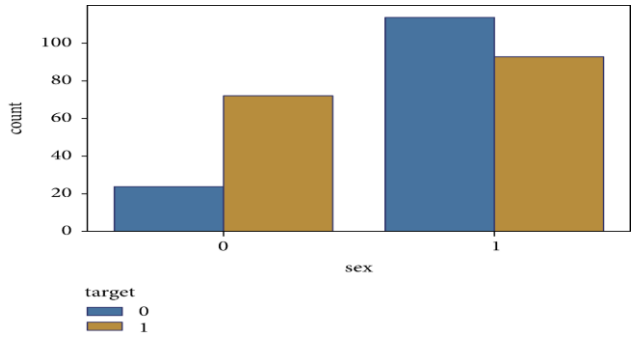

Fig. 14: Disease status by Gender

The graph shows that males have a higher risk of developing cardiovascular disease than do women. Figure 14 displays the probability distribution for four different types of features.

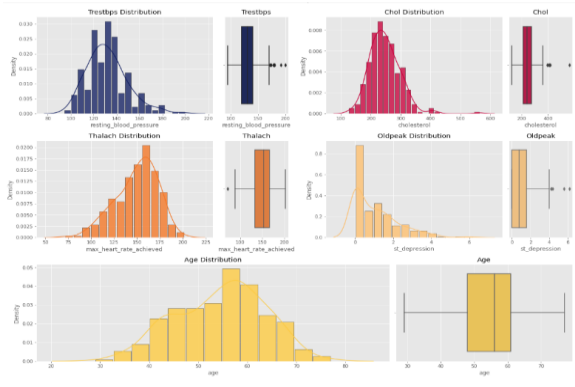

Fig. 15: Probability density of features.

B. Bivariate Analysis

The most basic type of quantitative analysis is bivariate analysis. In order to ascertain their empirical relationship, two variables must be analysed. Simple hypothesis testing can benefit from bivariate analysis.

In this part we are going to take our variables and compare them against our target condition which is if the observed patient has a disease or not.

Here we can presume that:

- Males are much more likely for heart diseases.
- Asymptomatic chest pain is the most common illness result, and the sort of chest pain is extremely subjective and has little bearing on the outcome.
- The condition is not directly impacted by blood sugar.
- Rest ECG results showing no direct results but having normal ECG is pretty good sign. If you have an ST-T wave irregularity, even if it appears to be rather uncommon in the data, you are three times more likely to have heart disease.
- Exercise-induced angina is a rather significant indication of heart disease; people are about three times more likely to get the condition if they have it. In the meanwhile, it's less than half without it.
- Patients who had flat slope distribution are more likely to have disease.
- Number of major vessels observed seems on similar levels for patients who have disease but 0 observations is good sign for not having disease.
- Having defected thalium test results is powerful indicator for heart disease.

"Figure 16 shows that blood pressure levels, cholesterol levels, age and peak heart rate are not uniformly distributed. This will need to be resolved to prevent data insufficiency or redundancy. In addition, cholesterol is an important factor in the study of heart disease."

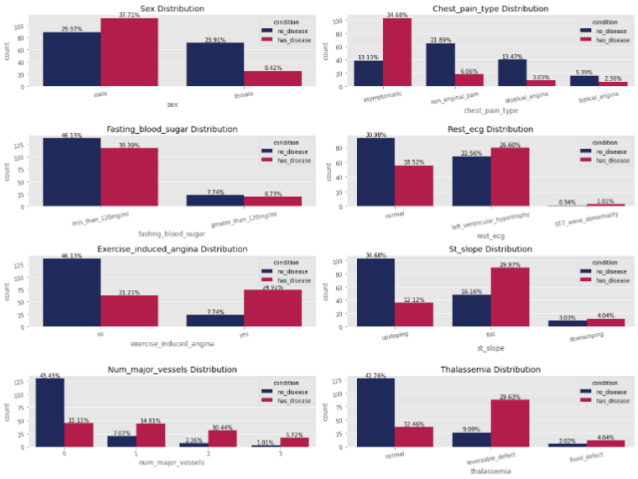

Fig. 16: Numerical Data vs Target

Following assumptions are made:

- Having higher resting blood pressure shows you are little bit more likely to have heart disease.
- Again, same for Cholesterol, it's not strong indicator but patients are little bit more likely to have disease with high cholesterol. There's is also one outlier there with no disease, pretty interesting.
- Find max heart rate distribution a bit interesting, expecting the other way around but it might be due to testing conditions and if you have normal results on ECG while exercising instructors might be increasing your exercise density?
- It's pretty clear that heart disease likelihood increases with ST depression levels...

ISSN No:-2456-2165

- Finally, older patients are more likely to have heart disease.

C. *Correlation Analysis*

Use Pearson correlation as in Figure 17 to find linear relations between features. Heatmap is decent way to show these relations as shown in Figure 18 & 19.

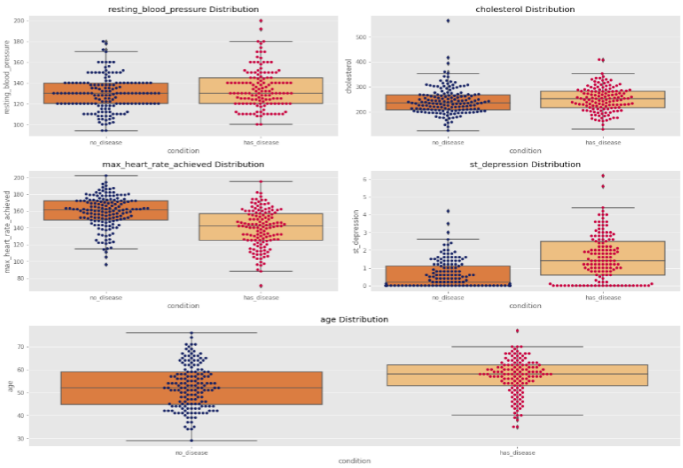

Fig. 17: Pearson Correlation to find linear relations between features

V. MODELING

We start by loading our train data and labels as X and y's and we get dummy variables for categorical data using one hot encoding [OHE].

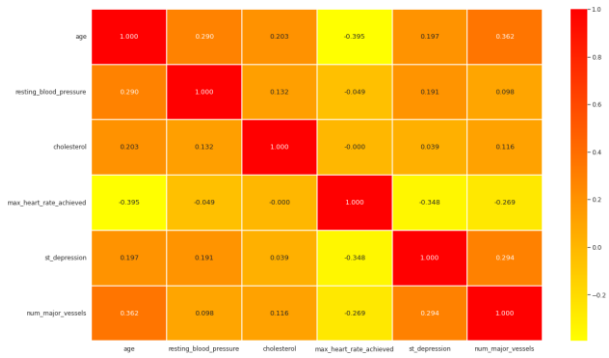

Fig. 18: Heatmap

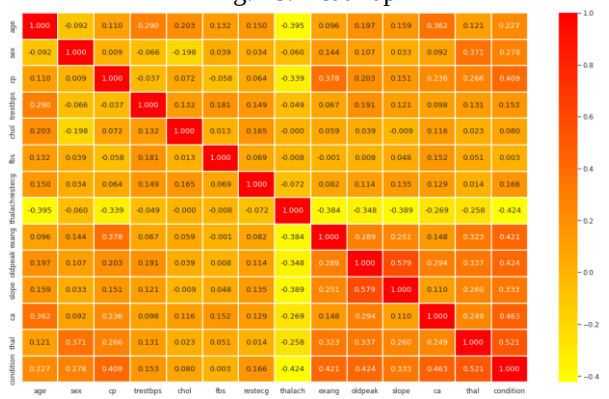

Fig. 19: Heatmap on all features

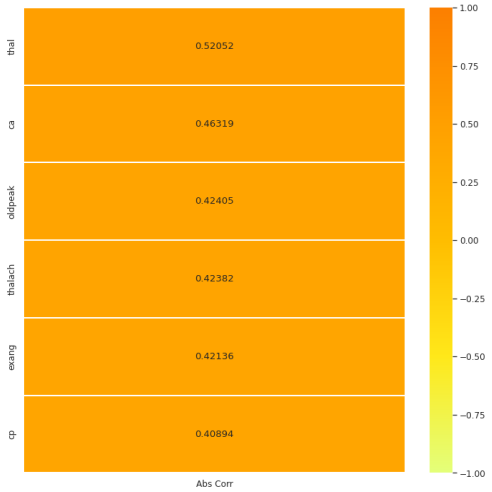

Fig. 20: Top Correlated Variables

A. *Classifiers*

Gradient Boosting Classifier: Gradient boosting is a generalisation of rising variable loss functions. It is also known as gradient boosted decision trees (GBDT). Regression and classification issues may be solved using GBDT, an effective and precise method, in a variety of domains, including online search ranking.

K-Nearest Neighbor Classifier: Neighbor-based classification is a form of instance-based learning, often known as non-generalization learning, because it merely maintains examples of training data rather than attempting to create a general internal model. The classification is determined by calculating the simple majority of each node's closest neighbours; the query node specifies the class of data that most accurately describes the point's closest neighbour.

Decision Tree Classification: "A non-parametric supervised learning approach used for classification and regression," according to the Wikipedia entry on decision trees (DT). The goal is to create a model that predicts the value of the target variable by learning simple decision rules inferred from data features. Decision trees learn from data to compute sine curves with different decision rules. Decision rules are more complex as the depth of the tree.

Support Vector Machines: Support Vector Machines (SVM) are a set of highly sophisticated learning techniques used for classification, regression, and outlier detection. Advantages of SVM:

- Effective in high dimensional spaces.
- More efficient if the number of samples is less than the number of dimensions.
- It is also memory efficient because it uses only a fraction of learning points in the decision function.
- Multidimensional: different kernel functions can be defined for the decision function." Standard cores are provided, but custom cores can also be specified.

Two averaging strategies based on random decision trees are included in the Ensemble module: the random forest algorithm and the additive tree approach. Any two

ISSN No:-2456-2165

ML techniques can be combined to create an ensemble model. Both algorithms are combinatory and perturbing methods created especially for trees. As a result, while building classifiers, a variety of classifiers are created using random input. The ensemble prediction provides the average prediction made by each classifier.

AdaBoost Classifier: A meta-estimator that emphasises upcoming classifiers is the AdaBoost classifier., starting from fitting a classifier to a real database, fitting additional instances of the classifier in the same database, but adjusting the weight for cases of misclassification. difficult situation.

Multiple-Layer Perceptron Classifier The stochastic gradient descent or LBFGS method is used in this model to optimise the log loss function.

Gaussian NB: using the Gaussian Naive Bayes classification technique. They presumptively believe that the feature's probability is Gaussian.

VI. RESULT DISCUSSION

- We have many metrics but decided to sort them by F1 score since precision and recall is important in this case. Looking at our first result, Random Forest Classifier is the best performing one in the list, followed by MLP and Gradient Boosting classifiers.
- But we can see most of our decision tree-based models are overfitting, that's something we should take a look at soon...

	Model Name	Train Roc/AUC Mean	Test Roc/AUC Mean	Test Roc/AUC Std	Train Accuracy Mean	Test Accuracy Mean	Test Acc Std	Train F1 Mean	Test F1 Mean	Test F1 Std	Time
4	RandomForestClassifier	1.000000	0.900139	0.048001	1.000000	0.831095	0.052795	1.000000	0.812917	0.056677	0.331019
6	MLPClassifier	0.943029	0.895704	0.053869	0.880705	0.810192	0.056530	0.848979	0.796919	0.059191	0.372652
0	GradientBoostingClassifier	0.999972	0.885312	0.053211	0.995791	0.814872	0.063174	0.995487	0.799866	0.062559	0.225396
7	GaussianNB	0.901929	0.893079	0.032982	0.836702	0.811695	0.052721	0.814794	0.790126	0.053867	0.006470
5	AdaBoostClassifier	0.908810	0.864971	0.044481	0.829309	0.787053	0.070837	0.822778	0.763340	0.074706	0.111552
2	DecisionTreeClassifier	1.000000	0.731797	0.058374	1.000000	0.730395	0.059089	1.000000	0.706855	0.053496	0.011588
1	KNeighborsClassifier	0.847549	0.694880	0.058835	0.770177	0.649700	0.057291	0.745321	0.593375	0.060607	0.008158
3	SVC	0.781350	0.736896	0.077651	0.670453	0.668036	0.052064	0.577783	0.562750	0.061458	0.010158

Fig. 21: Comparison of ML Methods

- Since our decision tree-based models overfitted, wanted to look which features mostly effected these decisions, sampled two of the tree-based models as presented below:

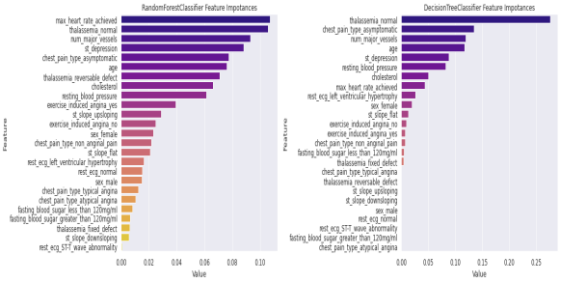

Fig. 22: Results of ML methods

A. *Automatic Outlier Detection*

Before fine tuning our models, get rid of some outliers. We have pretty small database and we can actually remove them by hand or more basic methods.

B. *Isolation Forest*

- An "isolation forest" randomly chooses a split value between the minimum and maximum value of the feature that was initially chosen in order to "isolate" observations.
- The number of partitions necessary to isolate an instance is equal to the length of the route from the root node to the end node since the recursive partition may be determined by the tree topology.
- In a forest of random trees like this, the average path length is a function of our choice and serves as a gauge of normalcy.
- Short pathways are made for anomalies by random splits. Therefore, it is likely anomalous when a forest of random trees collectively delivers the shortest path length for a particular sample.
- Contamination rate of our data was set to 10% and dropped them using masks. It didn't do great on the results and removing some datasets reduces model performances.

C. *Model Accuracy*

"LR outperformed other algorithms in terms of accuracy. RF also performs well in terms of accuracy. On the other hand, the performance of DT is really bad. Precision, recall, F1 score, and precision of the random forest algorithm are 77%, 87%, 82%, and 80%, respectively. Furthermore, the precision, recall, F1-score, and precision of the logistic regression algorithm were 92%, 92%, 92%, and 92% respectively."

D. *Confusion Matrix*

The confusion matrix for the RF classifier is seen in Figure 23. This classifier succeeds in achieving an accuracy rate of 80%."

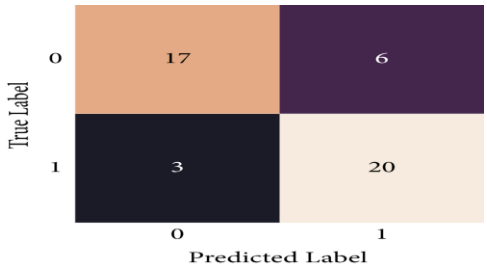

Fig. 23: Confusion matrix of Random Forest Classifier.

Figure 23 displays the 37 data points that the RF classifier predicted correctly and the 9 data points that it predicted erroneously.

The receiver operating characteristic (ROC) prediction curve is displayed in Figure 24. AUC (accuracy under the curve) for random forest classification is 88 percent. Figure 25 depicts the decision tree algorithm's confusion matrix."

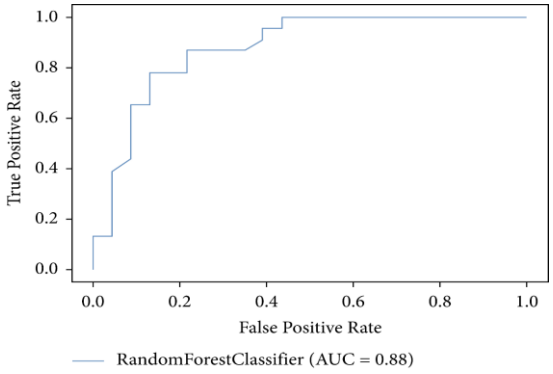

Fig. 24: AUC for Random Forest.

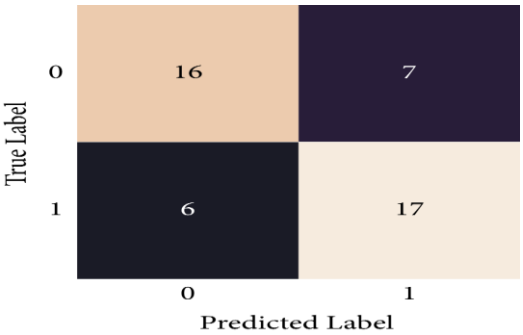

Fig. 25: Confusion matrix of decision tree classifier.

Figure 25 shows that the DT classifier predicted 33 data points correctly and 13 data points erroneously.

Figure 26 depicts the AUC's form for the forecast. The skewness accuracy of the DT classifier is 72%. The confusion matrix of the LR method is shown in Figure 27."

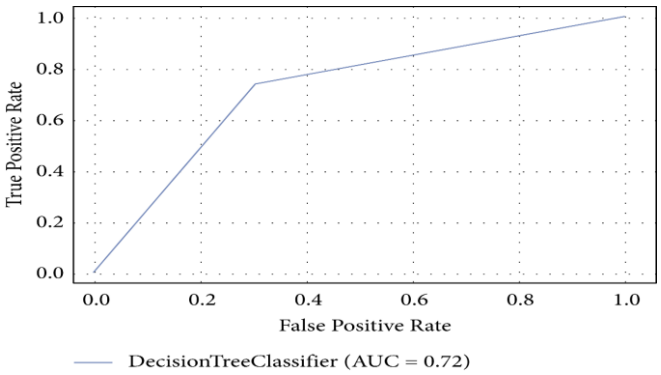

Fig. 26: AUC of decision tree

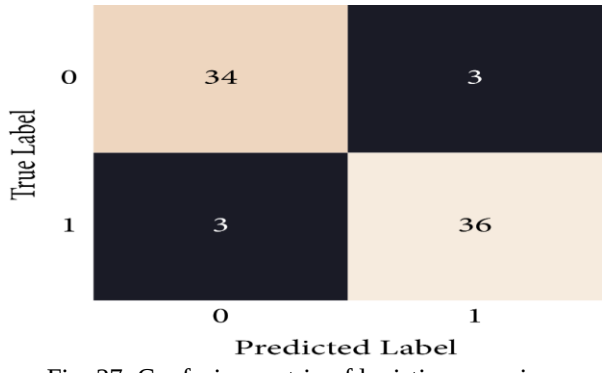

Fig. 27: Confusion matrix of logistic regression.

The logistic regression classifier accurately predicted 70 data points, whereas 6 data points were wrongly forecasted, as shown in Figure 27.

Figure 28 depicts the AUC of the prediction's form. The LR classifier has a 95% accuracy under the curve. It demonstrates unequivocally that among the several framework models, logistic regression is the best one. Its degree of precision is greater."

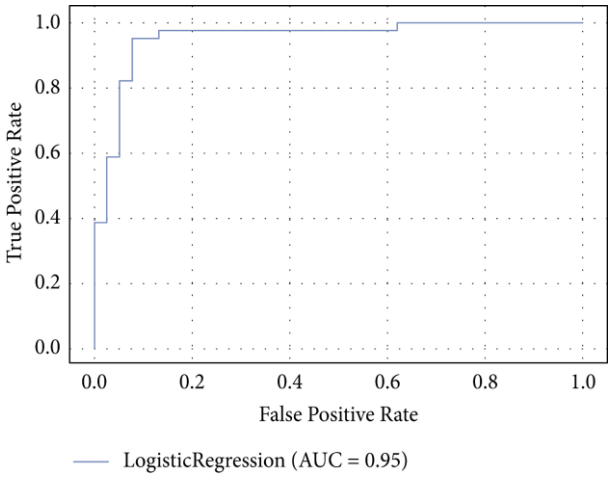

Fig. 28: ROC curve for logistic regression

VII. HYPER PARAMETER TUNING

A mathematical model with several parameters that can learn from data is referred to as a machine learning model. However, there are some parameters known as "Hyperparameters" that cannot be directly verified. Before beginning the real training, people frequently take a certain course or practise and exam. By enhancing the model's functionality, such as complexity or training speed, these parameters demonstrate their significance. Finding the ideal set of parameters can be viewed as a research topic because a model may contain several hyper-parameters.

Every ML approach uses these parameters. Given that SVM has several hyper-parameters, finding the best one is a difficult process (such the C value or gamma utilised). However, it may be done by considering all the possible combinations and identifying the traits that are problematic.

The GridSearchCV function takes a dictionary describing the parameters that can be tested in the model to train it. An array of parameters is defined as a dictionary, where the key is the parameter and the value is the setting to test.

The application of the GridSearchCV search method to find hyper-optimal parameters and thus improve the accuracy/prediction results is tested and the results are verified."

A. Data Acquisition

We will use the CVD dataset constructed from the Cleveland database. We will split the data into a 90:10 ratio between the grass and the test set. Train a "support vector

classifier without hyper-parameter tuning. First, we will train our model by making standard calls.

Run SVC () without setting hyperparameters and see the classification and confusion matrix."

```
""# train the model on train set
model = SVC ()
model.fit(X_train, y_train)
# print prediction results
predictions = model.predict(X_test)
print(classification_report(y_test, predictions))""
```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	66
1	0.61	1.00	0.76	105
accuracy			0.61	171
macro avg	0.31	0.50	0.38	171
weighted avg	0.38	0.61	0.47	171

Fig. 29: Results of SVC Before Hyperplane

Although recall and accuracy are usually zero for class 0, we achieved a precision of 61%. This indicates that everything is consistently assigned to class 1, which is the classifier's only class! This indicates that we must change the model's parameters.

The advantages of Grid Search then become useful. Grid Search allows us to look for parameters!

B. Use Grid search CV

The meta-evaluator in Grid Search CV is one of its best features. Create a new estimator that functions similarly to an estimator like SVC, in this instance acting as a classifier. Refit=Yes should be included, then select a password number. The longer the number, the more detailed (the phrase means only textual output describing the process). Grid search CV import.

```
"# defining parameter range
param_grid = {'C': [0.1, 1, 10, 100, 1000],
'gamma': [1, 0.1, 0.01, 0.001, 0.0001],
'kernel': ['rbf']}
```

	KNN	Decision Tree Classifier	Random Forest Classifier	Gradient Boosting Classifier	AdaBoost Classifier	Gaussian NB	SVC	MLP Classifier
KNN	-	94.2	93.1	91.7	92.5	93.7	94.6	95.6
Decision Tree Classifier	94.2	-	96.2	95.1	94.7	97.2	98.1	97.7
Random Forest Classifier	93.1	96.2	-	95.5	96.7	97.1	97.3	97.6
Gradient Boosting Classifier	91.7	95.1	95.5	-	94.3	94.3	95.8	95.8
AdaBoost Classifier	92.5	94.7	96.7	94.3	-	94.8	95.8	96.3
Gaussian NB	93.7	97.2	97.1	94.3	94.8	-	97.2	97.4
SVC	94.6	98.1	97.3	95.8	95.8	97.2	-	97.8
MLPClassifier	95.6	97.7	97.6	95.8	96.3	97.4	97.8	-

```
grid = GridSearchCV(SVC(), param_grid, refit = True,
verbose = 3)
# fitting the model for grid search
grid.fit(X_train, y_train)"
```

The fit is slightly larger than usual. First, it works by cross-validating the same loop to find the best combination of parameters. Because it's the perfect combination, that is

It refits all given data to build a single new model (without cross-validation) using the best parameter settings.

The best parameters found by GridSearchCV are the

```
{'C': 10, 'gamma': 0.0001, 'kernel': 'rbf'}
SVC(C=10, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma=0.0001, kernel='rbf',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
```

best estimators in the best_params_ attribute and the best_estimator_ attribute:

```
"# Print the best parameters after setting
```

	precision	recall	f1-score	support
0	0.95	0.91	0.93	66
1	0.94	0.97	0.96	105
accuracy			0.95	171
macro avg	0.95	0.94	0.94	171
weighted avg	0.95	0.95	0.95	171

```
print(grid.best_params_)

# print how our model looks after hyper-parameter tuning

print(grid.best_estimator_)"
```

We can then run the prediction again and see the classification report on these mesh objects as in the normal model.

```
"grid_predictions = grid.predict(X_test)
# print classification report
print(classification_report(y_test, grid_predictions))"
```

Fig. 30: Results of SVC After Hyperplan**e**

We have got almost **95% prediction** result.

Like this, when we merge different ML models, we note following accuracy.

When two different machine learning classifiers are merged using hyperplane method, we get different accuracies as given in above table. Best combination is SVC with Decision Tree Classifier which gives 98.1% accuracy."

Fig. 31: Results of Ensemble Models

VIII. CONCLUSION AND FUTURE ENHANCEMENTS

In this project, various machine learning methods are presented and their comparative value is explained. The purpose of this project "is to determine which machine learning classifier will be more effective in predicting heart failure based on the" dataset used, and combine two ML algorithms to achieve high speed. By combining the decision tree and SVC, we can get 98.1% accuracy so we can't switch to a combination of three methods. The outputs of eight "classifiers" were compared. Accuracy, specificity, sensitivity, and confounding matrix are a few of the comparative techniques employed. The LR classifier outperformed the ML approach for 14 of the sample's variables. With an accuracy of 92 percent, the logistic regression technique fared better than the other two classifiers. The DT classifier's accuracy is 72%, compared to the RF classifier's accuracy of 80%. This concept has the potential to revolutionise the medical industry. This technique can help lower the death rate by identifying heart disease patients early. Writing records in the database that are based on the standard model is not costly. As a consequence, many patients may receive these high-quality tests, and many more individuals can benefit from them. Future advancements in machine learning algorithms will lead to an increase in this sort of diagnosis. The model may be improved and altered if more patient data are utilised. Results are more exact and accurate when larger data sets are used. This is crucial since making a medical diagnosis demands a high level of accuracy and precision and is a difficult process. Future web apps that combine these methods and work with bigger data sets than those utilised in this study may be created. As a result, healthcare providers can predict and treat heart problems more accurately and efficiently. This will increase the reliability and visibility of the frame.

Dataset:- ""The data used to support the findings of this study are available online at https://www.kaggle.com/ronitf/heart-disease-uci."""

REFERENCES

[1.] M. Durairaj and V. Revathi, "Prediction of Heart Disease Using Back Propagation MLP Algorithm" Information Communication and Embedded Systems, vol. 4, no. 08, pp. 235–239, 2015.View at: Google Scholar

[2.] M. Gandhi, "Predictions in heart disease using techniques of data mining," in Proceedings of the International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), pp. 520–525, Noida, India, February 2015.View at: Google Scholar

[3.] S. Abdullah, "A Data mining Model for predicting the Coronary Heart Disease using Random Forest Classifier," International Journal of Computer Application, vol. 22, 2012.View at: Google Scholar

[4.] S. F. Weng, J. Reps, J. Kai, J. M. Garibaldi, and N. Qureshi, "Can machine learning improve cardiovascular risk prediction using routine clinical data?" PLoS One, vol. 12, no. 4, Article ID e0174944, 2017.View at: Google Scholar

[5.] V. V. Ramalingam, A. Dandapath, and M. K. Raja, "'Heart disease prediction using machine learning techniques: a survey," International Journal of Engineering & Technology, vol. 7, no. 2.8, pp. 684–687, 2018.View at: Google Scholar

[6.] L. Baccour, "Amende d fuse d TOPSIS-VIKOR for classification (ATOVIC) applied to some UCI data sets R," Expert Systems With Applications, vol. 99, pp. 115–125, 2018.View at: Publisher Site | Google Scholar

[7.] R. Das, I. Turkoglu, and A. Sengur, "Expert Systems with Applications Effective diagnosis of heart disease through neural networks ensembles," Expert Systems with Applications, vol. 36, no. 4, pp. 7675–7680, 2009.View at: Publisher Site | Google Scholar

[8.] Cheng and H. Chiu, "An artificial neural network model for the evaluation of carotid artery stenting prognosis using a nationalwide database," in Proceedings of the 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 2566–2569, Jeju Island, Republic of Korea, July 2017.View at: Google Scholar

[9.] J. Nahar, T. Imam, and K. S. Tickle, "Expert Systems with Applications Association rule mining to detect factors which contribute to heart disease in males and females," Expert Systems with Applications, vol. 40, no. 4, pp. 1086–1093, 2013.View at: Publisher Site | Google Scholar

[10.] S. Zaman and R. Toufiq, "Codon based back propagation neural network approach to classify hypertension gene sequences," in Proceedings of the 2017 International Conference on Electrical, Computer and Communication Engineering (ECCE), pp. 443–446, Cox's Bazar, Bangladesh, February 2017.View at: Publisher Site | Google Scholar

[11.] K. Ravish, K. J. Shanthi, N. R. Shenoy, and S. Nisargh, "Heart function monitoring, prediction and prevention of heart attacks: using artificial neural networks," in Proceedings of the 2014 International Conference on Contemporary Computing and Informatics (IC3I), pp. 1–6, Mysore, India, November 2014.View at: Publisher Site | Google Scholar

[12.] W. Zhang and J. Han, "Towards heart sound classification without segmentation using convolutional neural network," in Proceedings of the 2017 Computing in Cardiology (CinC), pp. 1–4, Rennes, France, September 2017.View at: Publisher Site | Google Scholar

[13.] Golande and T. P. Kumar, "Heart disease prediction using effective machine learning techniques," International Journal of Recent Technology and Engineering, vol. 8, pp. 944–950, 2019.View at: Google Scholar

[14.] T. Nagamani, S. Logeswari, and B. Gomathy, "Heart disease prediction using data mining with mapreduce algorithm," International Journal Of Innovative

Technology And Exploring Engineering, vol. 8, no. 3, 2019.View at: Google Scholar
[15.] F. S. Alotaibi, "Implementation of machine learning model to predict heart failure disease', (IJACSA)," International Journal of Advanced Computer Science and Applications, vol. 10, no. 6, 2019.View at: Google Scholar
[16.] N. Repaka, S. D. Ravikanti, and R. G. Franklin, "Design and implementing heart disease prediction using naives bayesian," in Proceedings of the 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), pp. 292–297, Tirunelveli, India, April 2019.View at: Publisher Site | Google Scholar
[17.] J. Thomas and R. T. Princy, "Human heart disease prediction system using data mining techniques," in Proceedings of the 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), pp. 1–5, Nagercoil, India, March 2016.View at: Publisher Site | Google Scholar
[18.] M. N. Lutimath, C. C. Basavaraj, and S. Pol, "Prediction of heart disease using machine learning," International Journal of Recent Technology and Engineering, vol. 8, pp. 474–477, 2019.View at: Google Scholar
[19.] S. T. Noor, S. T. Asad, and M. M. Khan, "Predicting the risk of depression based on ECG using RNN," Computational Intelligence and Neuroscience, vol. 2021, Article ID 1299870, 12 pages, 2021.View at: Publisher Site | Google Scholar
[20.] "Heartdisease UCI," https://www.kaggle.com/ronitf/heart-disease-uci.View at: Google Scholar
[21.] B. Rjeily, G. Badr, and E. Hassani, "Medical data mining for heart diseases and the future of sequential mining in medical field," in Machine Learning Paradigms, pp. 71–99, Springer, Cham, Switzerland, 2019.View at: Google Scholar
[22.] K. Srinivas, G. Raghavendra Rao, and A. Govardhan, "Analysis of coronary heart disease and prediction of heart attack in coal mining regions using data mining techniques," in Proceedings of the 2010 5th International Conference On Computer Science & Education, pp. 1344–1349, IEEE, Hefei, China, August 2010.View at: Google Scholar
[23.] R. Bharti, A. Khamparia, M. Shabaz, G. Dhiman, S. Pande, and P. Singh, "Prediction of heart disease using a combination of machine learning and deep learning," Computational Intelligence and Neuroscience, vol. 2021, Article ID 8387680, 11 pages, 2021.View at: Publisher Site | Google Scholar
[24.] R Deepu, S Murali, Vikram Raju, A mathematical model for the determination of distance of an object in a 2D image, Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV),2013, The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).
[25.] G. Guidi, M. C. Pettenati, P. Melillo, and E. Iadanza, "A machine learning system to improve heart failure patient assistance," IEEE Journal of Biomedical and Health Infor-Matics, vol. 18, no. 6, pp. 1750–1756, 2014.View at: Publisher Site | Google Scholar