DOI 10.3233/SJI-190609
IOS Press

An overview of cardiovascular disease infection: A comparative analysis of boosting algorithms and some single based classifiers

Nureni Olawale Adeboye* and Olawale Victor Abimbola
Department of Mathematics and Statistics, Federal Polythechnic Ilaro, Nigeria

Abstract. Machine learning is a branch of artificial intelligence that helps machines learn from observational data without being explicitly programmed and its methods have been found to be very useful in the modern age for medical diagnosis and for early detection of diseases. According to the World Health Organization, 12 million deaths occur annually due to heart-related diseases. Thus, its early detection and treatment are of interest. This research introduces a better way of improving the timely prediction of cardiovascular diseases in suspected patients by comparing the efficiency of two boosting algorithms with four (4) other single based classifiers on cardiovascular official data. The best model was selected based on performances of 5 different evaluation metrics. From the results, Adaptive boosting is seen to outperform all other algorithms with a classification accuracy of 74.2%, closely followed by gradient boosting. However, gradient boosting was chosen as an acceptable technique because it trains faster than Adaboost with a better precision of 74.9% compared to 74.7% exhibited by Adaboost. Thus boosting algorithms are better predictors compared to single based classifiers with factors of age, systolic blood pressure, weight, cholesterol, height, and diastolic blood pressure as the major contributors to the model building.

Keywords: Cardiovascular diseases, ensemble, boosting algorithms, AdaBoost, gradient boosting

1. Introduction

Bishop defined machine learning as a branch of artificial intelligence which helps in making machines learn from observational data without being explicitly programmed [1]. Alternatively, machine learning is based on automated and self-training algorithms to learn from prior data in order to find the pattern, which exists within, and then help machines to make decisions in situations they have never seen. There are different approaches to machine learning which can be supervised learning which learns a function by mapping input to output e.g. classification and regression, unsupervised learning which doesn't have a label but finds a pattern in the dataset with a little supervision and reinforcement learning which learns from its environment i.e. the agent learns from its action and consequences without been explicitly taught. According to Magoulas et al., machine learning has been a useful tool in different diverse fields and most especially it has been used to solve many medical problems [2].

Cardiovascular disease generally refers to conditions that involve narrowed or blocked blood vessels that can lead to a heart attack, chest pain (angina) or stroke. Other heart conditions, such as those that affect the heart's muscle, valves or rhythm also are considered forms of heart disease. According to Badimon et al., cardiovascular disease can be referred to as different heart or blood vessel problems; the term is often used to describe damages of heart or blood vessels by atherosclerosis, a buildup of fatty plaques in the arteries [3]. The Mayo Clinic opined that plaque buildup thickens and stiffens artery walls, which can inhibit blood flow through the arteries to organs and tissues. Atherosclerosis is also the most common cause of cardiovascular

*Corresponding author: Nureni Olawale Adeboye, Department of Mathematics and Statistics, Federal Polythechnic Ilaro, Nigeria. E-mail: nureni.adeboye@federalpolyilaro.edu.ng.

disease [4]. It can be caused by correctable problems, such as an unhealthy diet, lack of exercise, being overweight and smoking. The World Health Organization has estimated that 12 million deaths occur worldwide, every year due to heart diseases. Half of the deaths in the United States and other developed countries occur due to cardiovascular diseases. It is also the chief reason for deaths in numerous developing countries. On the whole, it is regarded as the primary reason behind deaths in adults [5]. For many years, different techniques and approaches have been applied in diagnosing and predicting cardiovascular diseases and these approaches have yielded high accuracies. For instance, Andrea emphasized that researchers have used different data mining techniques in predicting cardiovascular diseases [6]. Yan also used multilayer perceptron in predicting heart diseases and ended up having an accuracy of about 63.6% [7] while Polat et al. used a fuzzy artificial immune recognition system and k-nearest neighbor in the detection of heart disease using the Cleveland Heart Disease Dataset [8]. Chau et al. did a comparison of bagging with the C4.5 algorithm and bagging with a naïve Bayes algorithm to diagnose the heart disease of the patient [9] while Rajkumar et al. investigated the compared naïve Bayes, k-nearest neighbor and decision tree in the diagnosis of heart disease patients [10]. Sitar-Taut and Raphia et al. developed a heart disease prediction system using three data mining techniques such as decision trees, naive Bayes, and neural network [11,12]. The results obtained after prediction using the Cleveland Heart disease database indicated that naive Bayes performed well followed by neural networks and decision trees. It was also observed that the relationship obtained between attributes using neural networks is more difficult to understand than that of the other models used. Mythili et al. showed a framework using combinations of support vector machines (SVM), logistic regression, and decision trees to arrive at an accurate prediction of heart disease [13]. Comparison between performance measures which include sensitivity, specificity, and accuracy proved SVM to be the best model with an accuracy of 90.5%, followed by a decision tree with 77.9%, and logistic regression with 73.9%.

Ensemble learning techniques have been a very great help when it comes to increasing the accuracy of a machine learning algorithm. Many researchers have applied different ensemble learning approaches for modeling cardiovascular diseases without actually emphasizing the superiority of any approach, or comparing machine learning results with results from single based classifiers. Thus, this study is focusing on comparing some selected machine learning classifiers with some ensemble boosting algorithms in order to substantiate the existing results on them to suggest the most suitable technique for timely prediction of cardiovascular disease.

2. Materials and methods

2.1. Materials

The data set used for this research was adapted from [14]. It is a cardiovascular disease dataset that was originated on 19th January, 2019 through factual information, results of medical examination and information given by patients. The dataset was code named as "cardio_train.csv" and the author is Svetlana Ulianova, a data science student at Ryerson University, Toronto, Ontario, Canada. It is a structured tabular dataset that was adopted concurrently based on the result of medical examination and information given by patients during a medical examination. It contains exactly 70,000 instances and 11 features, which are great determinants for predicting cardiovascular diseases. The features of the dataset are as explained in Table 1.

2.2. Methods

Ensemble methods combine several decision trees classifiers to produce better predictive performance than a single decision tree classifier. Thus this research considered an overview of two popular ensemble boosting algorithms (adaptive boosting and gradient boosting) and their comparison with some selected machine learning classifiers (logistic regression, support vector classifier, random forest, decision tree, and naïve Bayes). The aforementioned techniques were employed to model the rate of cardiovascular disease infection based on the adopted variables.

2.2.1. Boosting

Boosting is a sequential ensemble method known for converting weak learners' algorithms into strong learners' algorithms. Boosting is based on the question posed by Micheal and Valiant [15,16]. That is, can a set of weak learners create a single strong learner? The answer to this question was investigated by Schapire [17] and found to be positive. The two boosting algorithms employed in this research are as follows:

Adaptive boosting (AdaBoost)

AdaBoost is a combination of the words adaptive and

Table 1
Description of datasets

Name	Class	Label	Values
Age	Numeric	Age of patients in days	10798–21327
Gender	Integer	Gender of patients	1-Male, 2-Female
Height	Numeric	Height of patients (cm)	55 to 250
Weight	Numeric	Weight of patients (kg)	10-200
Ap_hi	Numeric	Systolic blood pressure	-150 to 16020
Ap_low	Numeric	Diastolic blood pressure	-70 to 11000
Cholesterol	Integer	Cholesterol level of patients	1-normal, 2-above normal, 3-well above normal
Gluc	Integer	Glucose level of the patients	1-normal, 2-above normal, 3-well above normal
Smoke	Integer	If patients smoke or not	0-No, 1-Yes
Alco	Integer	If patients take alcohol or not	0-No, 1-Yes
Active	Integer	If patients is active or not	0-No, 1-Yes
Cardio	Integer	If patients has the disease or not	0-No, 1-Yes

The data description gotten during medical examination and information provided by the patients. The data is available in https://www.kaggle.com/sulianova/cardiovascular-disease-dataset. It shows the class of each variable and the values in the dataset.

boosting, but it is not the only adaptive boosting algorithm. In addition, AdaBoost is a special case of gradient boosting, which becomes important when making claims about relative performance between models. It is a machine learning meta-algorithm which can be used alongside with many other types of learning algorithms to improve the performance of a model. AdaBoost can be applied to any classification algorithm, so it is really a technique that builds on top of other classifiers as opposed to being a classifier itself. AdaBoost refers to a particular method of training a boosted classifier. A boosting classifier is in the form

$$F_N(x) = \sum_{n=1}^{N} f_n(x) \tag{1}$$

Where each f_n is a weak learner that takes an object x as input and returns a value indicating the class of the object. Suppose we are given training data $\{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^K$ and $y_i \in \{-1, 1\}$. And suppose we are given a (potentially large) number of weak classifiers, denoted by $f_n(x) \in \{-1, 1\}$ and a 0 -1 loss function I is defined by:

$$I(f_n(x), y) = \begin{cases} 0 \text{ if } f_n(x_i) = y_i \\ 1 \text{ if } f_n(x_i) \neq y_i \end{cases} \tag{2}$$

After learning, the final classifier is based on a linear combination of the weak classifiers:

$$g(x) = sign\left(\sum_{n-1}^{M} \alpha_n f_n(x)\right) \tag{3}$$

Essentially, AdaBoost is an algorithm that builds up a "strong classifier" $(g(x))$ incrementally, by optimizing the weights for and adding one weak classifier at a time. This can be expressed as given in equation

$$H(x) = sign\left(\sum_{n=1}^{N} \alpha_n h_n(x)\right) \tag{4}$$

$h_n(x)$ is the output of weak classifier n for input x; α_n is the weight assigned to the classifier.

α_n is calculated as follows:

$$\alpha_n = 0.5 \times \ln\left(\frac{1-\varepsilon}{\varepsilon}\right)$$

The weight of the classifier is straightforward, it is based on the error rate ε. Initially, all the input training examples have equal weights. It is pertinent to mention that adaBoost is a special case of gradient boosting, which becomes important when making claims about relative performance between models.

Gradient boosting

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.

Basically, the main objective of any supervised learning algorithm is to define a loss function and to minimize the loss. Assuming we have a mean square error defined as:

$$\text{MSE} = \sum (y_i - \hat{y}_i)^2 \tag{5}$$

where y_i is the i^{th} target value, and $\hat{y}_i$ is the i^{th} prediction value.

Let $L(y_i, \hat{y}_i)$ be the loss function, our main aim is to build a model in which the loss function of the mean squared error is minimized. Using gradient descent and updating the prediction based on the learning rate, we can find the values where MSE is minimized.

$$\hat{y}_i = \hat{y}_i + \frac{\alpha \times \delta \sum (y_i - \hat{y}_i)^2}{\delta \hat{y}_i} \tag{6}$$

$\Rightarrow \hat{y}_i = \hat{y}_i + \alpha \times 2 \times \sum (y_i - \hat{y}_i)^2$, where α is the learning rate and $\sum (y_i - \hat{y}_2)^2$ is the sum of residuals.

Other machine learning classifiers

Logistic regression

Logistic regression is a specialized form of regression that is formulated to predict and explain a binary (two-group) categorical variable rather than a metric dependent measure. The form of the logistic regression variate is similar to the variate in multiple regression. The variate represents a single multivariate relationship, with regression-line coefficients indicating the relative impact of each predictor variable. Logistic regression is most commonly used when the data in question has binary output, so when it belongs to one class or another, or is either a 0 or 1. Because the dependent variable is not a continuous one, the goal of logistic regression is a bit different, because we are predicting the likelihood that Y is equal to 1 or 0. The logistic formulas are stated in terms of the probability that $Y = 1$, which is referred to as P. The probability that Y is 0 is $1 - P$.

$$\ln \left(\frac{P}{1 - P} \right) = a + bX \tag{7}$$

P can be computed from the regression equation also as

$$P = \frac{\exp(a + bX)}{1 + \exp(a + bX)} = \frac{e^{a+bX}}{1 + e^{a+bX}} \tag{8}$$

Random forest

Random forests are ensemble learning methods for classification, regression and other tasks that operate by constructing a multitude of decision trees and outputting the mode of the classes (classification) or mean prediction (regression) of the individual trees. In other words, random forests builds multiple decision trees and merge their predictions together to get a more accurate and stable prediction rather than relying on individual decision trees.

Random forest has nearly the same hyperparameters as a decision tree or bagging classifier. Fortunately, there's no need to combine decision trees with bagging classifiers because we can easily use the classifier-class of random forest. The main limitation of random forest is that large numbers of trees can make the algorithm too slow and ineffective for real-time predictions. In other words, algorithms are fast to train but quite slow and ineffective for real-time prediction.

Decision tree

Pouriyeh et al. defined decision tree (DT) as a tree-like structure that consists of a root node, branches and leaf nodes [18]. It is a non-parametric model that can efficiently deal with large and complex datasets without imposing complicated distributional assumptions. DT can be implemented in both classification and regression tasks. It is easy to interpret, robust to outliers and can also work in the presence of missing values without needing to resort to imputation. The main disadvantage of DT is that it can be subject to overfitting and underfitting when using a small data set according to Song and Ying [19].

Naive Bayes

Naive Bayes models perform probabilistic prediction with an assumption that there exists strong independence among predictors. Bayesian classification predicts the class of new sets of data, following the Bayes theorem.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \tag{9}$$

Evaluation metrics

Five (5) evaluation metrics adopted in this research are discussed as follows:

Mean Absolute Error (MAE)

MAE is computed by calculating the absolute difference between the target values and the predictions. This is a linear score which means that all the individual differences are weighted equally in the average. MAE is computed as below:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{10}$$

where y_i are the values of the target variable while $\hat{y}_i$ are the predicted values.

Classification accuracy

This is the ratio of the number of correct predictions to the total number of input samples. Formally, classification accuracy has the following definition:

$$\textit{Classification accuracy} = \frac{\textit{Total number of correct predictions}}{\textit{Total number of predictions}} \tag{11}$$

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

$$\textit{Classification accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

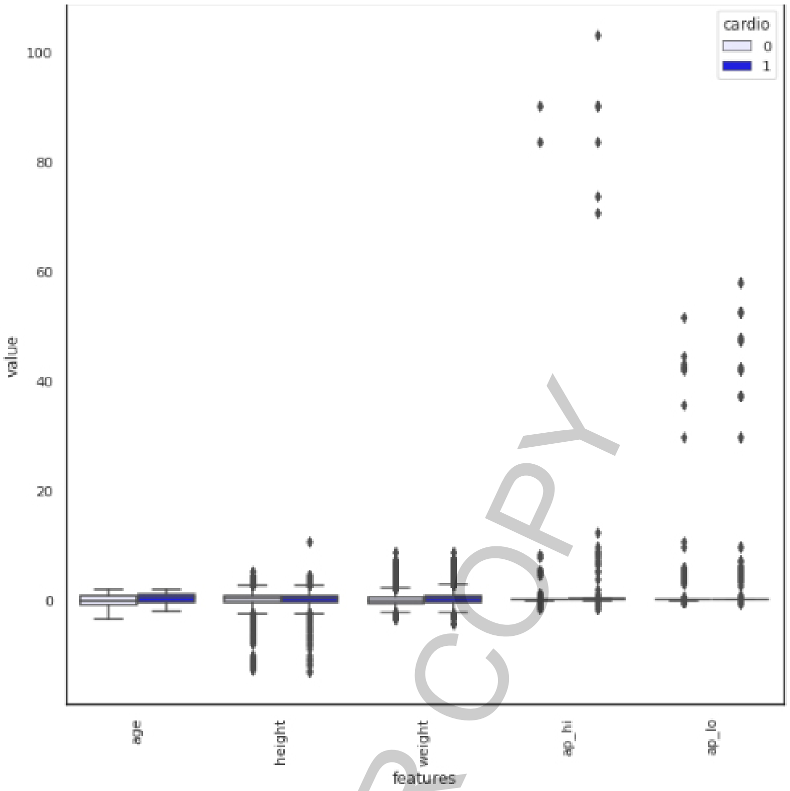

Fig. 1. Boxplot of the suspected features.

Recall

Recall measures the percentage of predictions that were correctly classified. Recall helps when the cost of false negatives is high and it is calculated as follows:

$$Recall = \frac{TP}{TP + FN} \tag{13}$$

Precision

Precision quantifies the number of positive class predictions that actually belong to the positive class. In other words, precision measures the portion of positive identifications in a classification set that was actually correct.

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

F1 score

F1 is an overall measure of a model's performance that combines precision and recall. It is the harmonic mean of precision and recall metrics. Therefore, this score takes both false positives and false negatives into account. F1 is usually more useful than accuracy. Accuracy works best if false positives and false negatives have a similar cost.

The computation is as below:

$$F1\ score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{15}$$

3. Results

Table 1 provides details of potentially predictive features in the dataset, their scale, and the units of measurement employed. Descriptive statistics are presented in Table 2 and Fig. 1 respectively. Additional tables and figures show two popular ensemble boosting algorithms and four other single based classification algorithms for the prediction of cardiovascular disease infection among patients. In addition, the results of the five evaluation metrics in selecting the best model are also presented.

Table 2 shows the descriptive statistics of the dataset after dropping the duplicates. We observed that the minimum height is 55 cm and that of the weight is 10 kg while the maximum height and weight are 250 cm and 200 kg respectively. It is observed that the minimum age is 10798 days which is approximately 29 years. These results show that there are estranged values in the dataset and these was reflected in the nature of boxplot

Table 2
Descriptive statistics table

	Age	Gender	Height	Weight	Ap_hi	Ap_lo	Cholesterol	Gluc	Smoke	Alco	Active	Cardio
Count	68771	68771	68771	68771	68771	68771	68771	68771	68771	68771	68771	68771
Mean	19464.4	1.348	164.362	74.124	126.616	81.365	1.364	1.226	0.088	0.053	0.803	0.494
Std	2468.1	0.476	8.185	14.332	16.766	9.728	0.679	0.572	0.283	0.225	0.397	0.499
Min	10798	1	55	11	60	20	1	1	0	0	0	0
Max	23713	2	250	200	240	190	3	3	1	1	1	1

The descriptive statistics of the variables used in the dataset which is showing the data count, mean standard deviation (Std), minimum value (Min) and maximum value (Max) of each variable used for the research.

Table 3
Comparison of algorithms efficiency

Model	Accuracy	Precision	Recall	MAE	F1 score
Adaptive boosting	**0.742**	0.747	**0.703**	**0.258**	**0.724**
Gradient boosting	0.741	0.749	0.695	0.259	0.721
Logistic regression	0.733	0.746	0.676	0.267	0.709
Random forest	0.732	0.735	0.695	0.268	0.714
Decision tree	0.721	0.729	0.669	0.279	0.698
Naive Bayes	0.717	**0.758**	0.607	0.283	0.674

The performance of the ensemble boosting algorithms (Adaptive Boosting and Gradient Boosting) and single based classifiers (Logistic Regression, Random Forest, Decision Tree and Naive Bayes) using a grid search of 5 fold cross-validation based on the model selection metrics (Accuracy, Precision, Recall, Mean Absolute Error (MAE) and F1 Score).

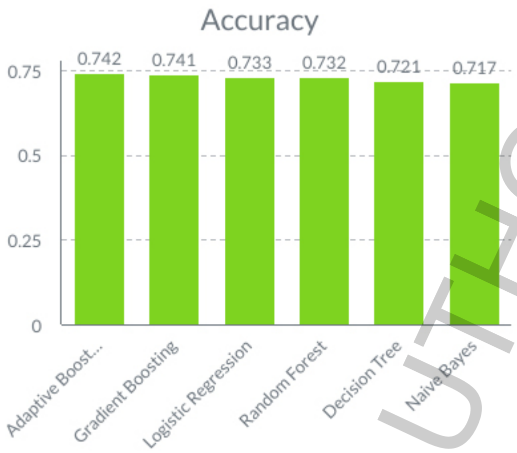

Fig. 2. Classification of algorithms efficiency.

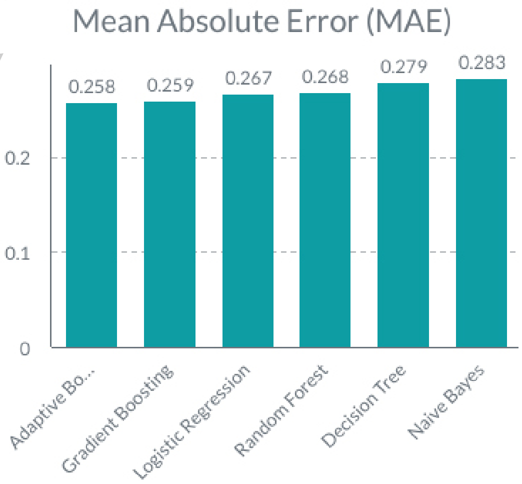

Fig. 3. MAE evaluation metric.

in Fig. 1. It was also discovered that there exists outliers in the ap_hi (systolic blood pressure) and ap_low (diastolic blood pressure) which contain values that are medically impossible, we found out values that are negative and very high. According to Madell, the normal range value for systolic blood pressure is between (90 and 180), and the normal range for diastolic blood pressure is between (80 and 120) [20]. This led to the removal of records with those values which are not biologically feasible. The box plots were inspected and outliers were identified for the height and weight as values outside the ranges of 153 cm–195 cm and 50 kg–95 kg respectively, while that of ap_hi and ap_low are identified outside the ranges of 90 mmHg–170 mmHg and 65 mmHg–105 mmHg respectively. It is pertinent to note that the selection of the height and weight ranges were highly influenced by the minimum and maximum ages.

After a proper data purification, out of 70000 rows of the dataset, the leftover of 68771 rows was split into training and validation sets. 80% of the processed data were used for model training and 20% of the remainder was used for model evaluation. The focus is on the fitting of two popular ensemble boosting algo-

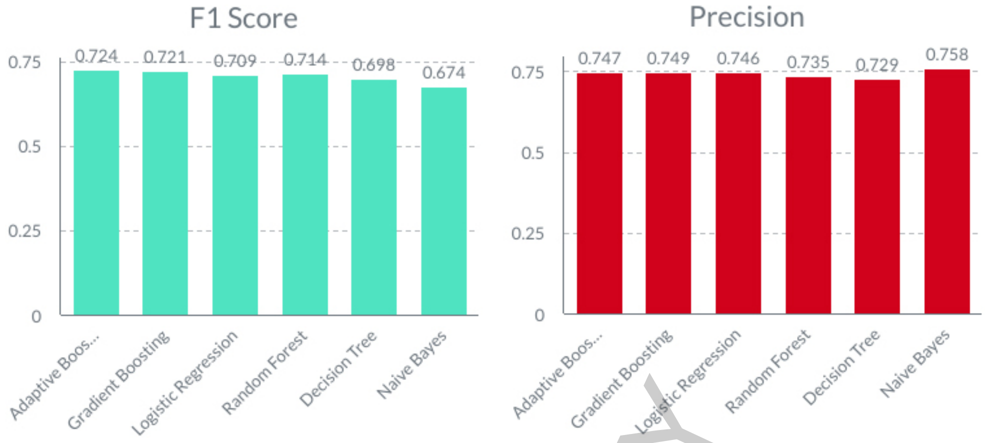

Fig. 4. F1 score and precision evaluation metrics.

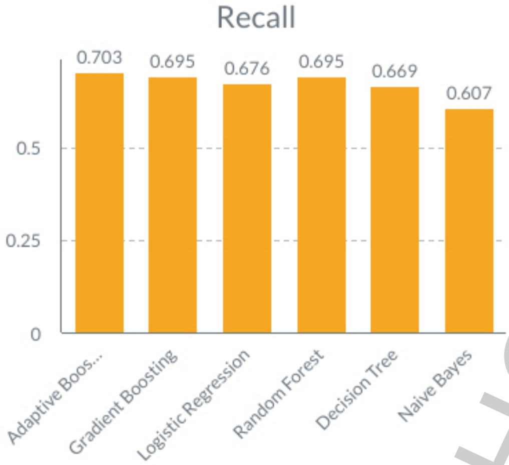

Fig. 5. Recall evaluation metrics.

rithms and four other single based classification algorithms on Python 3.6 [21] with Scikit learn library [22]. Grid search of 5fold cross-validation was carried out using multiple hyperparameters for each model. The grid search identified the best hyperparameters and the adopted evaluation metrics were computed based on the results of that analysis. The best hyperparameters selected with grid search used in fitting the models were listed in Appendix 1.

Table 3 presents the comparative results of the ensemble and classifiers algorithms' efficiency in predicting cardiovascular disease infection. From results, it was observed that adaptive boosting (Adaboost) outperformed gradient boosting and other single classification algorithms based on accuracy. Adaboost has an accuracy of 74.2% which is the highest value compared to the accuracy values of the other algorithms. However, Adaboost and gradient boosting have almost the same classification accuracy closely followed by logistic regression, random forest, decision tree and naive Bayes with classification accuracies of 74.1%, 73.3%, 73.2%, 72.1% and 71.7% respectively. These values for accuracy confirmed the effectiveness of all the chosen metrics. These results are depicted graphically in Fig. 2. However, considering the times of 25.9 minutes and 26.3 seconds both Adaboost and gradient boosting techniques took to train respectively, the gradient boosting technique will be a better choice than Adaboost given the fact that it trains faster with a higher percentage of precision as equally presented in table 3 results. Adaboost had a precision of 74.7% while that of gradient boosting is 74.9%.

These performances were shown graphically in Fig. 1, while Figs 2–5 described the results of MAE, F1, recall and precision metrics respectively in a more descriptive form. The latter figures actually helped in confirming if classification accuracy is really sufficient enough in selecting the best algorithm for predicting cardiovascular disease infection. Figures 6 and 7 showed respectively the importance of each of the features for predicting cardiovascular diseases with the gradient boost and Adaboost algorithms. For gradient boosting, the most important feature is systolic blood pressure (ap_hi) followed by age while with Adaboost, age is the most important feature closely followed by systolic blood pressure (ap_hi). Systolic blood pressure (ap_hi) variable indicates that the higher the blood pressure, the more risk such a person has to cardiovascular disease infection while the age factor shows that the older a patient, the more risk such a person has to cardiovascular disease infection.

According to Hastie, accuracy should be avoided for evaluating the utility of clinical models, because it does

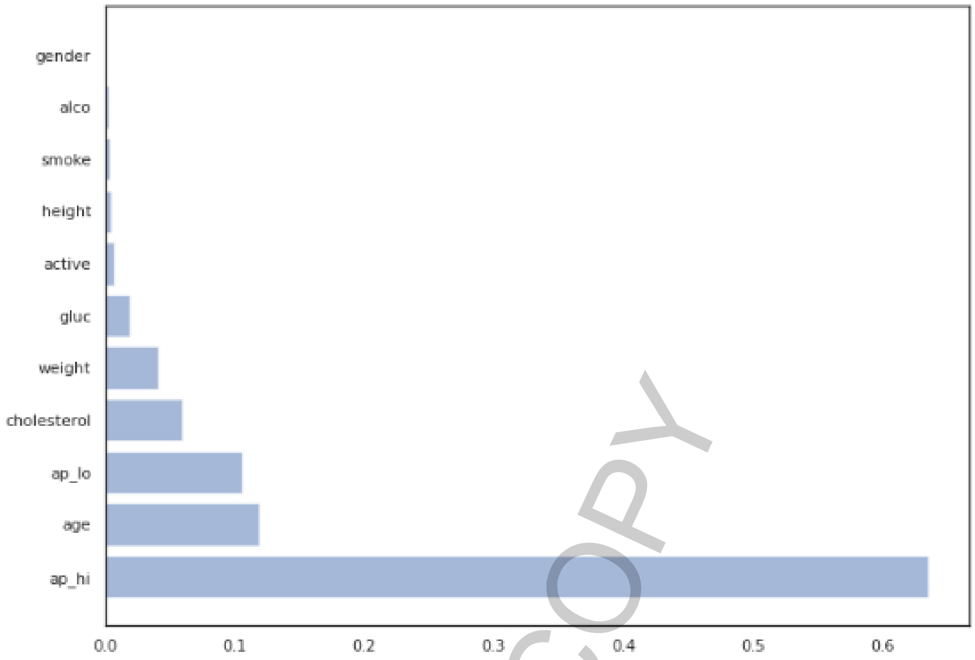

Fig. 6. Variables predictive of cardiovascular disease using gradient boosting. The importance of each of the features for predicting cardiovascular diseases with the Gradient Boosting algorithm. The most important variable is the Systolic blood pressure (ap_hi) followed by Age. Other variables are ap_lo, cholestrol, weight, glucose, active, height, smoke, alcohol, and gender.

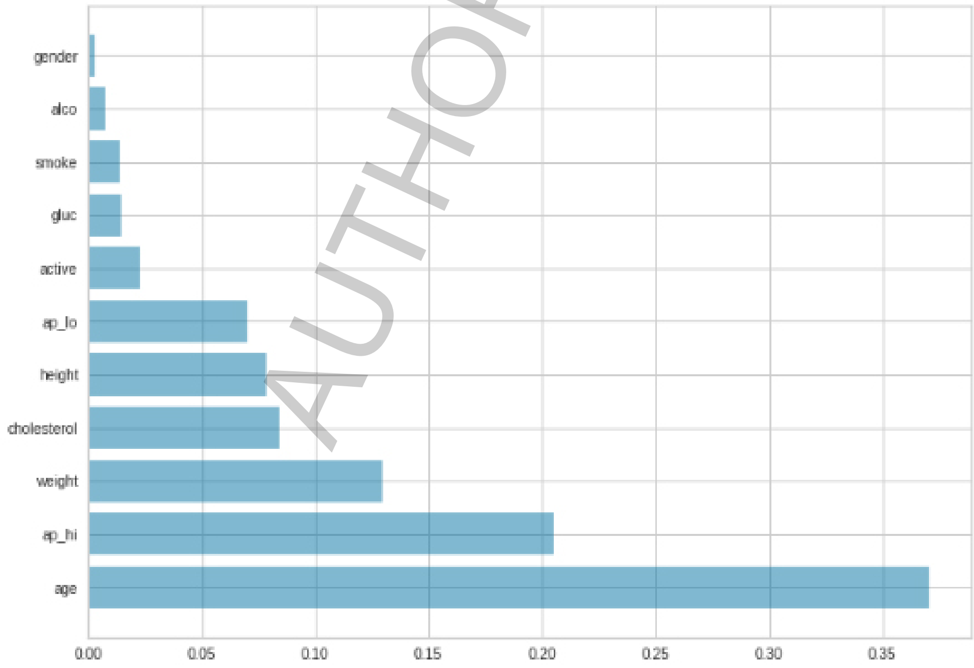

Fig. 7. Variables predictive of cardiovascular disease using adaboost boosting. The importance of each of the variables for predicting cardiovascular diseases with the Adaboost algorithm. The most important variable is Age closely followed by Systolic blood pressure (ap_hi). Other variables are weight, cholesterol, height, ap_lo, active, glucose, smoke, alcohol, and gender.

not take into account clinically relevant information, hence the need to examine more evaluation metrics [23]. The MAE or linear score measures model efficiency in terms of their weights. If all the individual differences are weighted equally, the higher the MAE score the worse the model. Based on the results from Table 3, in this study, naive Bayes has the highest MAE, and Adaboost has the lowest closely followed by gradient boost. Following the selection criteria, the Adaboost and gradient boost are the better algorithms in predicting cardiovascular disease infection. The graphical representation is shown in Fig. 3.

F1 is usually a powerful metric in measuring the performance of a model. Adaboost has the highest F1 score of 72.4% while naive Bayes has the lowest of about 67.4%. Based on the F1 score selection criteria, the higher the F1 score the better the model and from Table 3, it was seen that Adaboost equally outperformed other classifications based on the F1 score. Naive Bayes has the highest precision followed by gradient boosting and Adaboost Algorithm with 75.8%, 74.9% and 74.7% evaluations respectively. While the Adaboost has the highest value for recall, naive Bayes has the lowest.

F1 score and precision metrics are displayed graphically in Fig. 4 while the recall metric is displayed in Fig. 5.

On the overall, the gradient boost technique is adjudged to give the best predictive model closely followed by Adaboost, for the timely prediction of cardiovascular diseases in suspected cardiovascular patients. Thus, the important features of the Gradient boost and Adaboost are as presented in the plot displayed graphically in Figs 6 and 7 respectively. It is readily observed that age, systolic blood pressure, weight, cholesterol, height and diastolic blood pressure are the major contributing factors to the model building.

4. Conclusion

This article presented a comprehensive evaluation of ensemble boosting algorithms and some other machine learning classifiers to determine the best algorithm capable of predicting cardiovascular diseases. The single machine learning classifiers used were “logistic regression, random forest, decision tree and naive Bayes. The results from the analysis and the trained hyperparameters show that Adaboost outperformed other algorithms including gradient boosting with a classification accuracy of about 74.2% and favorable values for recall, F1 score and MAE. However, gradient boosting was identified as an acceptable technique for this research because it trains faster than Adaboost and has a slightly better precision of 74.9% compared to 74.7% exhibited by Adaboost. The accuracy of 74.2% attributed to Adaboost and of 74.1% attributed to gradient boosting both approximately imply that if the model is used for predicting or detecting cardiovascular diseases, 74 out of 100 predictions will be correct.

Similarly, the F1 score which is known as the overall measure of model performance and which tells us how perfect our precision and recall is, justifies this argument. The F1 scores of both the Adaboost and gradient boosting techniques are approximately the same with values 0.724 and 0.721 respectively. All these were achieved using a grid search with a ‘k’ fold cross-validation of 5.

This shows that gradient boosting is a better algorithm in predicting cardiovascular diseases with factors of age, systolic blood pressure, weight, cholesterol, height, and diastolic blood pressure as the major contributors to the model building; and that boosting algorithms are better predictors compared to single based classifiers.

With this justification, we strongly conclude that Gradient boosting algorithm will generate a correct prediction of cardiovascular diseases if used on a new dataset and the implementation of this model will go a long way in the early detection of cardiovascular diseases among patients. Though Adaboost may equally perform excellently as indicated by the closeness of its estimated metrics to that of gradient boosting, especially when it trains better. This inference has also served as justification that Boosting Algorithms Perform better than single based classifiers in classification and model predictive ability.

References

[1] Bishop, C.M. *Pattern recognition and machine learning*. Springer. 2016

[2] Magoulas, G.D., and Prentza, A. Machine learning in medical applications. *Machine Learning and its Applications*. 2001; 300–307. doi: 10.1007/3-540-44673-7_19.

[3] Badimon, L., Casani, L., and Vilahur, G. *Animal Models for the Study of Human Disease*. Academic Press, ScienceDirect. 2013; 221–239. doi: 10.1016/B978-0-12-415894-8.00010-5.

[4] Mayo Clinic, Health Diseases Symptoms and Causes, https://www.mayoclinic.org/diseases-conditions/heart-disease/symptoms-causes/syc-20353118.

[5] Worlds Health Organization. Cardiovascular Diseases (CVDs), https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds); 2018.

[6] Andrea D'Souza. Heart disease prediction using data mining techniques. *International Journal of Research in Engineering and Science (IJRES)*. 2015; 3(3), 74–77.

[7] Yan, H. Development of a decision support system for heart disease diagnosis using multilayer perceptron. *Proceedings of the 2003 International Symposium*. 2003; 5, 709–712.

[8] Polat, K., Sahan, S., and Gunes, S. Automatic detection of heart disease using an artificial immune recognition system (AIRS) with fuzzy resource allocation mechanism and k-nn (nearest neighbour) based weighting preprocessing. *Expert Systems with Applications*. 2007; 625–631.

[9] Chau, T., Shin, D., and Shin, D. Effective Diagnosis of Heart Disease through Bagging Approach. In: *2nd International Conference on Biomedical Engineering and Informatics*. 2009.

[10] Rajkumar, A., and Reena, G.S. Diagnosis of heart disease using data mining algorithm. *Global Journal of Computer Science and Technology*. 2010; 10(10).

[11] Sitar-Taut, V.A. Using machine learning algorithms in cardiovascular disease risk evaluation. Journal of Applied Computer Science & Mathematics, 2009.

[12] Rafiah, A., and Palaniappan, S. Intelligent heart disease prediction system using data mining techniques. *IJCSNS International Journal of Computer Science and Network Security*. August 2008; 8(8).

[13] Mythili, T., Dev, M., Nikita, P., and Abhiram, N. A heart disease prediction model using SVM-decision trees-logistic regression (SDL). *International Journal of Computer Applications*. 2013; 68(16).

[14] Kaggle Dataset Repository, https://www.kaggle.com/sulianova/cardiovascular-disease-dataset.

[15] Michael, K. Thoughts on Hypothesis Boosting. Unpublished manuscript (Machine Learning class project, December 1988). 1988.

[16] Michael, K., and Leslie, V. Cryptographic limitations on learning Boolean formulae and finite automata. *Symposium on Theory of Computing*. ACM 1989; 21, 433–444.

[17] Schapire, R.E. The strength of weak learnability. *Machine Learning*. 1990; 5(2), 197–227.

[18] Pouriyeh, Seyedamin, Sara Vahid, Giovanna Sannino, Giuseppe De Pietro, Hamid Arabnia, and Juan Gutierrez. A comprehensive investigation and comparison of Machine Learning Techniques in the domain of heart disease. In: *2017 IEEE Symposium on Computers and Communications (ISCC)*. 2017. doi: 10.1109/iscc.2017.8024530.

[19] Song, Y., and Lu, Y. Decision tree methods: applications for classification and prediction. PubMed Central (PMC). 2015.

[20] Madell, Robin, and Kristeen Cherney. Blood Pressure Readings Explained. Healthline. Accessed July 8, 2020. https://www.healthline.com/health/high-blood-pressure-hypertension/blood-pressure-reading-explained.

[21] Python Release Python 3.6.0. Python.org. Accessed July 8, 2020. https://www.python.org/downloads/release/python-360/.

[22] Support Vector Machines – scikit-learn 0.20.2 documentation. Archived from the original on 2017-11-08. Retrieved 2017-11-08.

[23] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (PDF) (Second ed.). New York: Springer. 2008; 134.

Appendix 1: Train times of boosting algorithms and single based classifiers

Model	Best hyperparameter	Time taken
Adaptive boosting	algorithm = 'SAMME.R', Criterion = gini, max_depth = 2, min_samples_leaf = 1, min_splits_leaf = 2, learning_rate = 1.0, n_estimators = 50.	25.9 min
Gradient boosting	criterion = 'friedman_mse', learning_rate = 0.6, loss = 'deviance', max_depth = 5, max_features = 'sqrt', min_samples_leaf = 1, min_samples_split = 2, n_estimators = 50, validation_fraction = 0.1	26.3 sec
Logistic regression	Penalty = l1, Solver = liblinear, max_iter = 100	6.5 min
Random forest	bootstrap = True, criterion = 'gini', max_features = 'auto', min_samples_leaf = 1, min_samples_split = 2, n_estimators = 100	11.3 min
Decision tree	criterion = 'gini', min_samples_leaf = 1, min_samples_split = 2, presort = 'deprecated', splitter = 'best'	2.0 min
Naive Bayes	priors = None, var_smoothing = 1e-09	20.5 sec

This is the best hyperparameter selected via Grid search using a cross validation of 5fold: Note!!! If a particular hyper parameter is not included in the table, it is majorly because they are default.