International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

Heart Failure Prognostication using Boosting Algorithms

Rachaell Nihalaani[1], Simran Mansharamani[2], Juhi Janjua[3]
[1]*Undergraduate Student, Department of Computer Engineering, Thadomal Shahani Engineering College, Mumbai-50, Maharashtra, India*
[2]*Graduate Student, Department of Computer Engineering, Thadomal Shahani Engineering College, Mumbai-50, Maharashtra, India*
[3]*Assistant Professor, Department of Computer Engineering, Thadomal Shahani Engineering College, Mumbai-50, Maharashtra, India*

Abstract: ***In the medical field, predicting a heart disease has become a very complicated and challenging task. So, in this contemporary lifestyle, there is an urgent need for a system that will help predict accurately the possibility of getting heart disease. This paper presents an observation-based comparison between four boosting algorithms namely Gradient boosting, XGBoost, ADAboost and CatBoost to predict heart failure efficiently. To do so, we have referred to the PLOS (Public Library of Science) Repository dataset. These algorithm's performances have been evaluated using metrics like Accuracy, F1 score, Recall and many more. All values obtained ensured the superiority of these boosting algorithms based on several performance measures.***

Keywords: Machine Learning, Binary Classification, Boosting Algorithm, Gradient boosting, XGBoost, AdaBoost, CatBoost.

I. INTRODUCTION

The number one cause of death globally, taking approximately 17.9 million lives every year, are cardiovascular diseases (CVDs). They include a number of disorders of the heart and blood vessels, including coronary heart disease, rheumatic heart disease, cerebrovascular disease and many such other conditions [1]. Heart failure is also a heart disorder that basically means that the heart is not pumping blood as well as it is supposed to, which in turn disrupts all major bodily functions. Also known as congestive heart failure, it should not be confused with when the heart stops beating. The heart keeps beating and working, but during heart failure the body's need for blood and oxygen is not being met adequately [2]. Like most CVDs, heart failure can be obviated by addressing behavioural risk factors like an unhealthy diet, obesity, tobacco and alcohol use, and physical inactivity, using population-wide strategies. However, if left untreated, heart failure could be substantially lethal. Humans diagnosed with heart failure and even those that are at high heart failure risk need detection and management at the earliest, and thus, building a machine learning model to make such predictions would be of great use. This literature discusses and compares four machine learning boosting techniques, Gradient Boosting, XGBoost, AdaBoost and CatBoost. Our research objective is to perform analysis and build a model to predict if a given set of symptoms would lead to heart failure. The performance of these techniques will be assessed by the evaluation metric of classification accuracy. In consequence, the results and conclusions of this literature should allow future researchers to select the most effective boosting algorithm out of the four, that can provide the best performance for future comparison.

Further, Section 2 reviews the study of related literature. Section 3 illustrates the theoretical considerations. Section 4 presents the experimental procedure and exploratory analysis of the referred dataset. Section 5 discusses the experimental results. Section 6 draws the conclusion of the paper.

II. LITERATURE REVIEW

There is an extensive variety of machine learning algorithms and techniques that have been used to predict an accurate model. [3] used five machine learning models to predict the heart disease using collected dataset namely, SVM, Random Forest, Decision tree, Logistic regression and Naïve bayes algorithm. According to their study, the accuracy of the decision tree model and SVM was the highest, while the performance of the Naïve bayes had shown the lowest accuracy. [4] worked on models for outcomes using machine learning techniques by first making use of claims-based predictors, and then adding recorded variables to the claims-based predictors. Logistic regression, Least Absolute Shrinkage, Random forests, and Selection Operator, Classification and Regression Tree and Gradient-boosted model were used. They concluded that machine learning methods offered limited scope of improvement over logistic regression in predicting key outcomes in HF. [5] makes a prediction by using various algorithms like Logistic regression, KNN and Random Forest Classifier, where the first two algorithms gain a maximum accuracy of 88.5%.

Machine learning algorithms used here have been efficient in various other models and fields. [6] used the semi-supervised Tri-CatBoost algorithm and compared it with Random Forest, Decision tree, multilayer perceptron, and GBDT achieving higher macro precision and macro recall with CatBoost. [7] predicts traffic flow using the ADAboost algorithm and gained promising results on both simulations and real data. The Naive Bayes predictor in comparison to their model gives a higher error rate. [8] compared with the traditional models to the XGB model for predicting evaporation duct height and found significant improvement and excellent prediction results. It showed that, for EDH prediction, learning and prediction ability of XGBoost algorithm are better than feedforward deep neural network.

III. THEORITICAL CONSIDERATIONS

In any pattern recognition problems, classification plays a vital role. It uses machine learning algorithms which learn how to assign a class label to examples from the problem domain. This literature illustrates a binary classification problem. Typically, binary classification involves two classes – the first class is the normal state having label 0 and the other is the abnormal state having label 1. [9]. 'Boosting' refers to a set of algorithms, which converts learners from weak to strong, with the aim to increase accuracy [10]. A weak learner or classifier is a model that performs better than random guessing or a naïve prediction model whereas a strong learner or classifier is a model that performs really well compared to random guessing or a naïve prediction model. Boosting is an ensemble method, which is an algorithm that builds a set of classifiers and then takes a vote of their predictions to classify new data points.

This is to improve the predictions of the model of any given learning algorithm. The idea is to train the weak learners sequentially, each trying to rectify its predecessor [11]. In simple terms, to classify a particular set of data, we have rules. Since these rules are not strong enough, they individually cannot be classified, they are called weak learners. This weak learner is then converted to a strong learner by combining each and every weak learner using normal or weighted average, or by taking into account the highest voted prediction.

A. Gradient Boosting

In gradient boosting, with every iteration, the overall model improves sequentially. In other words, base learners are sequentially generated such that the current base learner is always more effective than the previous base learner. The central idea in gradient boosting is to overcome the prediction of the previous learner by optimising its loss function, which is done by adding a new adaptive model that adds weak learners to reduce the loss function. There are three main components. First, the loss function which helps in reducing errors. Second, we need weak learners to compute predictions and to form strong learners. Third is the additional model that helps in regularising the loss function from the previous learner. Gradient boosting is an algorithm that can be used for both regression and classification problems [11].

B. XGBoost

A more advanced version of Gradient boosting method called the XGBoost or Extreme gradient boosting comes under distributed machine learning. It focuses on model efficiency and computational speed. Where the older model's sequential analysis of datasets takes a long time, there is a need for this advanced algorithm to boost the performance of the model. To boost performance, it creates decision trees parallelly and implements distributed computing methods to evaluate complex models. The use of out of core computing helps analyse huge datasets and implements cache optimisation to utilize all resources and hardware [11].

C. AdaBoost

Adaptive boosting (most commonly seen in classification problems) combines multiple weak learners into one strong learner. First, equal weightage is assigned to all the data points and a decision stump is drawn out for a single input feature. Then, it analyses the obtained results and for any misclassified observations, it assigns higher weights to them. A new decision stump is then drawn which gives importance to the higher weights.

Again, the miss-classified observations get the higher weights. This process keeps running in a loop till all observations are in the right place i.e all data points are classified correctly. AdaBoost is capable of being used in both regression and classification problems [11].

D. CatBoost

CatBoost also known as Category boosting is a machine learning algorithm developed to be open source. This algorithm offers a unique feature that is the integration to work with diverse data types which helps in solving a wide range of data problems faced by innumerable businesses. It also offers accuracy just like the other algorithms in the tree family [12]. It introduces advancement in two critical algorithms - a permutation-based alternative to the classic algorithm which is known as ordered boosting and an innovative algorithm to process categorical features. Random permutations of the training examples are being used by both, to fight the prediction shift which is caused by a special kind of target leakage present in all existing implementations of gradient boosting algorithms [13]. Unlike some other machine learning algorithms, CatBoost performs well with a small data set. It improves the performance of a model while reducing overfitting and also the time spent on tuning. Datasets where categorical features play an important role, such as the *Internet* datasets and *Amazon*, the improvement is significant and undeniable [13].

E. Terminologies

1) *Confusion Matrix:* A matrix for summarizing.performance of a classification algorithm. It depicts how the model is confused while making predictions, i.e. it summarises the number of correct and incorrect predictions, with count values and are broken down by each class. This breakdown gives a better insight than just using only the accuracy value [14].
2) *Accuracy:* The fraction of predictions that the model gets right [15].
3) *Precision:* The fraction of relevant information among the extracted information. High precision indicates that more relevant instances are being extracted by the model [15]
4) *Recall / Sensitivity:* The fraction of number of relevant instances with both the known and predicted result are positive [15].
5) *F1 Score:* The weighted average of precision and sensitivity. It considers false positives and negatives, and is more beneficial than accuracy in the case of class distributions being uneven [15]
6) *Specificity*: The fraction of number of relevant instances with both the known and predicted result are negative [16].
7) *False Discovery Rate:* The fraction of number of relevant instances with a positive prediction result for which the known result is negative [16].
8) *False Omission Rate:* It is the fraction of number of relevant instances with a negative prediction result for which the known result is positive [16].
9) *Matthews Correlation Coefficient (MCC):* It is a statistical rate generating a high score if and only if the estimate of all the four categories of confusion matrix achieves great results, in proportion to the size of positive and negative elements in the dataset. The closer MCC is to 1, the higher the possibility of the model being a pure binary classifier.

IV. PROPOSED FRAMEWORK

A. Methodology

This section explains the general process of binary classification and offers a detailed explanation of the flow of the experiment performed.

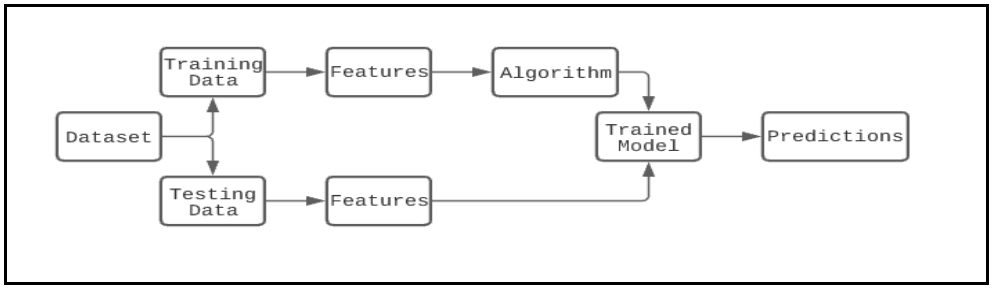

Fig. 1 Binary Classification Process

The first step is data collection, we have used an already existing dataset [17]. Our second step is to familiarise ourselves with the data. Exploratory analysis includes reading and cleaning our dataset, filling any missing values, and exploring the shape and features of the dataset. The third step is data pre-processing and visualisation. We convert all non-numeric values to numeric values as machine learning models can only work with numeric values. We visualise the correlation between features of our dataset to get a better idea of what we are working with. A thorough understanding of the data, we are manipulating as it helps us make more informed decisions.

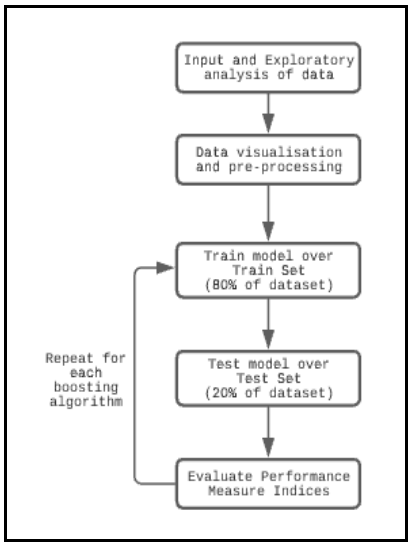

Fig. 2 Flowchart for our experimental process

The fourth step is splitting our data into training and testing subsets, in a 80-20 ratio respectively. We are performing binary classification using 4 boosting algorithms namely Gradient Boosting, XGBoost (eXtreme Gradient Boosting), AdaBoost (Adaptive Boosting), and CatBoost. For each algorithm, we perform the same 3 steps. First, we train our model over 80% of our dataset. Then, we test our model over the remaining 20% of the dataset and observe its performance, and lastly we calculate certain performance measure indices over which our models can be compared and the most accurate model can be identified.

B. Dataset

For the purpose of this research, we have used the dataset from the PLOS (Public Library of Science) Repository [17]. With 299 entries, this dataset has 13 features that indicate clinical, body and lifestyle information. It includes features like age, creatinine phosphokinase, ejection fraction and more. All the features are described in Table 1.

TABLE I
Features of Heart Failure Dataset

Dataset Feature	What it indicates	Range in dataset	Unit of measurement
Age	Patient's age	40 - 95	Years
Anaemia	Whether or not patient has a health condition where haemoglobin concentration or number of red blood cells is less than normal	0, 1	Boolean
High blood pressure	Whether or not patient has hypertension	0, 1	Boolean
Creatinine phosphokinase (CPK)	Patient's CPK enzyme level in blood	23 - 7861	mcg/L (micrograms per liter)
Diabetes	Whether or not patient has diabetes	0, 1	Boolean
Ejection fraction	Patient's blood percentage leaving the heart with each contraction	14 - 80	Percentage

Sex	Male/ Female	0, 1	Binary
Platelets	Patient's blood platelet count	25.01 - 850.00	kiloplatelets/mL
Serum creatinine	Patient's blood creatinine level	0.50 - 9.40	mg/dL
Serum sodium	Patient's blood sodium level	114 - 148	mEq/L (milliequivalents per litre)
Smoking	Whether or not the patient smokes	0, 1	Boolean
Time	Follow-up period	4 - 285	Days
Death Event (target variable)	Whether or not the patient died during the follow-up period	0, 1	Boolean

The dataset has been divided into a 80:20 ratio for training and testing respectively i.e. 80% data will be used to train our models whereas 20% data will be used to test our models.

C. Performance Measure Indices

We have used 8 indices to measure the performance of the implemented models. A confusion matrix is formed, for actual and predicted class, consisting of TP (True Positive/ Correctly Identified), FP (False Positive/ Correctly Rejected), TN (True Negative/ Incorrectly Identified), FN (False Negative/ Incorrectly Rejected) to assess the parameters.

The formulas used to measure the performance are:

$$\text{Accuracy} = \frac{TN+TP}{TN + TP + FN + FP} \quad (1)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

$$\text{Recall or Sensitivity} = \frac{TP}{TP+FN} \quad (3)$$

$$\text{F1 Score} = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (4)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (5)$$

$$\text{False Discovery Rate (FDR)} = \frac{FP}{FP+TP} \quad (6)$$

$$\text{False Omission Rate (FOR)} = \frac{FN}{FN + TN} \quad (7)$$

$$\text{Matthews Correlation Coefficient} = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (8)$$

V. EXPERIMENTAL RESULTS

This section discusses the tables and graphs obtained on performing binary classification on the chosen dataset [17] using Gradient Boosting, XGBoost, AdaBoost and CatBoost algorithms.

We visualise the data using a heat map, as shown in Fig. 3, specifically, we are checking the correlation between attributes. The blue boxes indicate a negative correlation i.e. one increases and the other decreases, the yellow and green boxes indicate only a moderate correlation and the red boxes indicate that the attributes are correlated with each other.

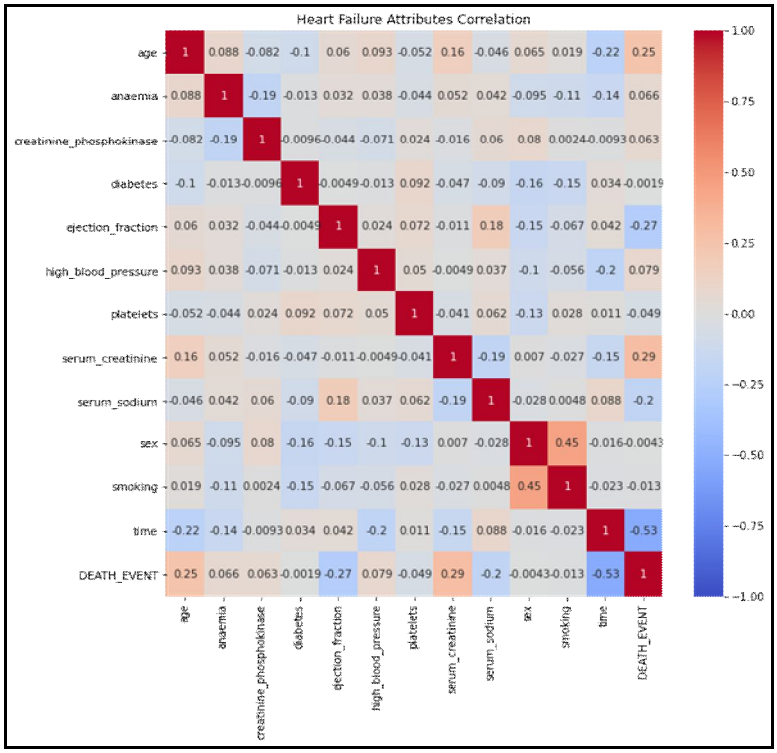

Fig. 3 Failure Attribute Correlation

We split our dataset into train and test sets in a 80:20 ratio. That means, we used 80% of our dataset for training the model and 20% of it for testing the model. We performed testing on all the four boosting algorithms, Gradient Boosting, XGBoost, AdaBoost and CatBoost.

The values of a confusion matrix helps us evaluate the performance of the boosting classification algorithms that we have used. Tables 2-5 depict the confusion matrices for Gradient Boosting, XGBoost, AdaBoost and CatBoost respectively.

TABLE II
Result Confusion Matrix obtained using Gradient Boosting

		Predicted Class	
		P	N
Actual Class	P	43	0
	N	3	14

The confusion matrix for Gradient Boosting shows that out of 60 test set entries, this model accurately predicts 43 cases as positive (no death caused by heart failure) and 14 cases as negative (death event caused by heart failure). On the other hand, it incorrectly predicts 3 cases as positive and 0 cases as negative.

TABLE III
Result Confusion Matrix obtained using XGBoost

		Predicted Class	
		P	N
Actual Class	P	42	1
	N	5	12

The confusion matrix for XGBoost shows that out of 60 test set entries, this model accurately predicts 42 cases as positive (no death caused by heart failure) and 12 cases as negative (death event caused by heart failure). On the other hand, it incorrectly predicts 5 cases as positive and 1 case as negative.

TABLE IV
Result Confusion Matrix obtained using AdaBoost

		Predicted Class	
		P	N
Actual Class	P	40	3
	N	4	13

The confusion matrix for AdaBoost shows that out of 60 test set entries, this model accurately predicts 40 cases as positive (no death caused by heart failure and 13 cases as negative (death caused by heart failure). On the other hand, it incorrectly predicts 4 cases as positive and 3 cases as negative.

TABLE V
Result Confusion Matrix obtained using CatBoost

		Predicted Class	
		P	N
Actual Class	P	43	0
	N	4	13

The confusion matrix for CatBoost shows that out of 60 test set entries, this model accurately predicts 43 cases as positive (no death caused by heart failure) and 13 cases as negative (death event caused by heart failure). On the other hand, it incorrectly predicts 4 cases as positive and 0 cases as negative.

Using the values shown in Tables 2-5, performance measure indices for the models are computed to understand the model performed. We calculate the Accuracy, Precision, Sensitivity, Specificity, F1 Score, False Discovery Rate, False Omission Rate and Matthews Correlation Coefficient. In Table 6, if the value of Matthews Correlation Coefficient is ~1, that means that there is a high likelihood for the respective model to be a pure binary classifier.

TABLE VI
Comparison of calculated performance measure metrics

Performance Measure Indices	Gradient Boosting	XGBoost	AdaBoost	CatBoost
Accuracy	0.95	0.90	0.88	0.93
Precision	1.00	0.97	0.93	1.00
Sensitivity	0.93	0.89	0.90	0.91
F1 Score	0.96	0.93	0.91	0.95
Specificity	1.00	0.92	0.81	1.00
False Omission Rate	0.00	0.02	0.06	0.00
False Discovery Rate	0.17	0.29	0.23	0.23
Matthews Correlation Coefficient	0.87	0.74	0.70	0.83

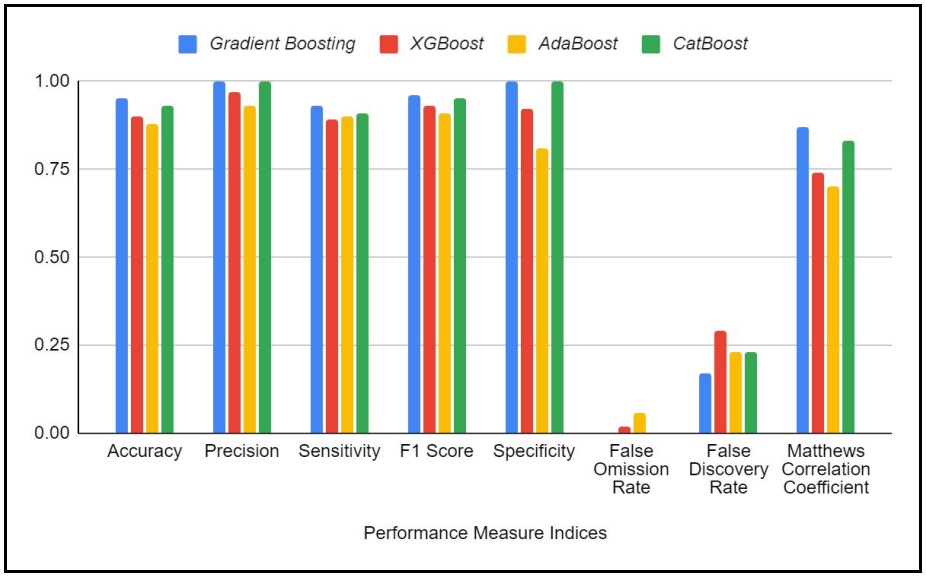

Fig. 4 Comparison of calculated Performance Measure Indices

VI. CONCLUSION

In this section, we discuss the inference of the charts and tables discussed in the previous section. The aim of this experiment was to present a comparative study by classifying the likelihood of the death of a patient by heart failure, based on certain medical factors, using four boosting algorithms, Gradient Boosting, XGBoost, AdaBoost and CatBoost. This study involved analysis based on different metrics like accuracy, precision, sensitivity, f1 score, specificity, false omission and discovery rates, and Matthews Correlation Coefficient which helped in the comparison between the four models. Pertaining to classification analysis, we found that Gradient Boosting algorithm proved to be the most accurate model, on our dataset, amongst the four, with an accuracy of 95%. It also showed the maximum value of Matthews Correlation Coefficient, ~0.87, which proves that it is the purest binary classifier amongst the four models. In addition, it shared the highest values of precision and specificity with CatBoost, but had the highest values of f1 score and sensitivity all by itself. It also shared the minimum value of false omission rate with CatBoost, and had the lowest false discovery rate. CatBoost proved to be the second most accurate model, with an accuracy of 93%.

VII. ACKNOWLEDGMENT

We would like to express our gratitude to our professor Juhi Janjua for her immense support and guidance. We would also like to express our gratitude to our Head of Department, Dr. Tanuja Sarode and our Principal, Dr. G.T. Thampi.

REFERENCES

[1] "Cardiovascular Disease." World Health Organisation. https://www.who.int/health-topics/cardiovascular-diseases/#tab=tab_1 (accessed Jun. 5, 2021).
[2] "What is Cardiovascular Disease." American Heart Association. https://www.heart.org/en/health-topics/consumer-healthcare/what-is-cardiovascular-disease (accessed Jun. 5, 2021).
[3] F Saleh. (2019). Implementation of Machine Learning Model to Predict Heart Failure Disease. Presented at International Journal of Advanced Computer Science and Applications(IJACSA), Volume 10 Issue 6. [Online]. Available:___https://thesai.org/Downloads/Volume10No6/Paper_37-Implementation_of_Machine_Learning_Model.pdf
[4] R. J. Desai, S. V. Wang, M. Vaduganathan, T. Evers and S. Schneeweiss. (Jan. 2020). Comparison of Machine Learning Methods With Traditional Models for Use of Administrative Claims With Electronic Medical Records to Predict Heart Failure Outcomes. Presented at Jama Network Open. [Online]. Available: https://jamanetwork.com/journals/jamanetworkopen/article-abstract/2758475
[5] H. Jindal, S. Agrawal, R. Khera, R. Jain and P. Nagrath. (2021). Heart disease prediction using machine learning algorithms. Presented at OP Conf. Series: Materials Science and Engineering. [Online]. Available: https://iopscience.iop.org/article/10.1088/1757-899X/1022/1/012072/meta
[6] W. Liu, K. Deng, X. Zhang, Y. Cheng, Z. Zheng, F. Jiang and J. Peng. (Jan. 2020). A semi-supervised Tri-CatBoost Method for Driving Style Recognition. Presented at Symmetry. [Online]. Available: https://www.mdpi.com/2073-8994/12/3/336/htm
[7] G. . Leshem, and Y. Ritov. (Jan. 2007). Traffic Flow Prediction using Adaboost Algorithm with Random Forests as a Weak Learner. Presented at World Academy of Science, Engineering and Technology. [Online]. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.131.8200&rep=rep1&type=pdf
[8] W. Zhao, J. Li, J. Zhao, D. Zhao, J. Lu and X. Wang. (Apr. 2018). XGB Model: Research on Evaporation Duct Height Prediction Based on XGBoost Algorithm. Presented at Radio Engineering, Vol. 29, No. 1. [Online]. Available: https://dspace.vutbr.cz/bitstream/handle/11012/186937/20_01_0081_0093.pdf?sequence=1
[9] J. Brownlee. "4 Types of Classification Tasks in Machine Learning." Machine Learning Mastery. https://machinelearningmastery.com/types-of-classification-in-machine-learning/ (accessed May. 13,2021)
[10] S. Ray. "Quick Introduction to Boosting Algorithms in Machine Learning." Analytics Vidya. https://www.analyticsvidhya.com/blog/2015/11/quick-introduction-boosting-algorithms-machine-learning/ (accessed Jun. 6, 2021).
[11] Z. Lateef. "A Comprehensive Guide To Boosting Machine Learning Algorithms." Edureka. https://www.edureka.co/blog/boosting-machine-learning/ (accessed Jun. 6, 2021).
[12] S. Adebayo. "How CatBoost algorithm works in machine learning." Data Aspirant. https://dataaspirant.com/catboost-algorithm/ (accessed Jun. 6, 2021).
[13] T. Peretz. "Mastering the new generation of gradient boosting." Towards Data Science. https://towardsdatascience.com/https-medium-com-talperetz24-mastering-the-new-generation-of-gradient-boosting-db04062a7ea2(accessedJun. 6, 2021).
[14] J. Brownlee. "What is a Confusion Matrix in Machine learning." Machine Learning Mastery. https://machinelearningmastery.com/confusion-matrix-machine-learning/ (accessed June. 6, 2021).
[15] K. P. Shung. "Accuracy, Precision, Recall or F1?." Towards Data Science. https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9 (accessed Jun. 6, 2021).
[16] "Binary Diagnostic Tests – Single Sample." NCSS statistical software. https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Binary_Diagnostic_Tests-Single_Sample.pdf (Accessed May. 14, 2021).
[17] T. Ahmad, A. Munir, S. Bhatti, M. Aftab and M. A. Raza. "Data Minimal." July 20, 2017. Distributed by Public Library of Science. https://plos.figshare.com/articles/dataset/Survival_analysis_of_heart_failure_patients_A_case_study/5227684/1