Accurate Liver Disease Prediction with Extreme Gradient Boosting

Sivala Vishnu Murty, R Kiran Kumar

Abstract: ***Abstract-Machine learning is used extensively in medical diagnosis to predict the existence of diseases. Existing classification algorithms are frequently used for automatic detection of diseases. But most of the times, they do not give 100% accurate results. Boosting techniques are often used in Machine learning to get maximum classification accuracy. Though several boosting techniques are in place but the XGBoost algorithm is doing extremely well for some selected data sets. Building an XGBoost model is simple but improving the model by tuning the parameters is a challenging task. There are many parameters to the XGBoost algorithm and deciding what set of parameters to tune and the ideal values of these parameters is a cumbersome and time taking task. We, in this paper, tuned the XGBoost model for the first time for Liver disease prediction and got 99% accuracy by tuning some of the hyper parameters. It is observed that the model proposed by us exhibited highest classification accuracy compared to all other models built till now by machine learning researchers and some regularly used algorithms like Support Vector Machines (SVM), Naive Bayes (NB), C4.5 Decision tree, Random Belief Networks, Alternating Decision Trees (ADT) experimented by us.***

Key words: Machine Learning, Classification, Liver Disease, Prediction, Boosting,

I. INTRODUCTION

As per 2017 World Health Organization reports, deaths caused by Liver Diseases in India touched 259,749 and it is 2.95% of total deaths and stands 63rd rank in the world. In the past, liver diseases were caused frequently by hepatitis B and C but now days it is mostly because of alcohol and obesity. Every year approximately 10 lakh patients are diagnosed who are suffering with liver cirrhosis. Liver disease is the tenth most usual reason of deaths in India as stated by the studies of World Health Organization. Identifying Liver infections at an early stage is a difficult job as the Liver functions in a normally way, even when it is partially damaged In this scenario, automatic diagnosis tools will support the doctors a lot for timely detection of the disease and improves the patients' survival rate. Conventional classification methods are used frequently in many automatic medical diagnosis tools which are less accurate. Hence there is a need for sophisticated classifiers and techniques like boosting for more accurate prediction

In gradient boosting, the loss function is minimized using the gradients .In each round of training, a weak learner is constructed and the predictions are compared with the expected outcome. Eerror rate of the model is the difference between predicted outcome and expected outcome. Now the gradients or partial derivative of the loss function can be calculated by using the errors. So it represents the steepness of the error function. The parameters of the model can be altered in order to decrease the error in the next round of training by decreasing the gradient. In Gradient Boosting, we merge the predictions given by multiple models and the boosted model predictions are optimized. Thus, the gradients used in the current training by fitting the next tree to these values. Extreme Gradient Boosting is sequential ensemble technique. It calculates second partial derivatives of the loss function to comprehend the trends of the gradients and to obtain the minimum of the loss function. Normal gradient boosting methods use the loss function of the base decision tree as a substitute so as to minimize the error of the overall model where as for estimation; XGBoost uses the 2nd order derivative. It uses sophisticated L1 and L2 regularization technique to improve model generalization. Moreover, training XGBoost model is very fast and can be parallelized too.

II. RELATED WORK

Carmona P et.al [1] proposed a model by using extreme gradient boosting in order to predict bank failures in banking segment of United States of America. They identified important variables to forecast bank defaults. They collected annual data of 30 financial ratios for 156 national commercial banks of U.S.A during the period 2001 to 2015. Identifying important indicators for bank failure early is very important and helps bank managers to precautions before the financial institutions fail. Their findings show that lesser values of retained earnings to average equity, total risk-based capital ratio and pretax return on assets are primary reasons for bank failure. In addition to that very high yield from earning assets boosts the possibility of bank fiscal crisis.

Stefanos Georganos et.al [2] implemented Extreme Gradient Boosting (XGBoost). For Very High Resolution (VHR) object based estimation of urban land cover. They examined the sensitivity of Xgboost algorithm to a variety of sample sizes .and Feature Selection (FS) by applying correlation-based Feature Selection.

Revised Manuscript Received on October 30, 2019.

* Correspondence Author
Sivala Vishnu Murty, Dept. of CSE, Aditya Institute of Technology and Management, Tekkali (A.P),India. Email: Vishnu_212@yahoo.co.in
Dr. R Kiran Kumar, Dept of Computer Science, Krishna University, Machilipatnam - 521001,(A.P). India Email: kirankreddi@gmail.com

They compared the performance of XGBoost with some benchmark classifiers like Random Forest (RF) , Support Vector Machines (SVM) . The methods were applied to VHR images of the village of Vaihingen, Germany and two sub-Saharan cities of Dakarand. The results show that the Bayesian procedure parameterized Xgboost outperformed Random Forest and SVM did better for larger sample sizes

.

Xuan-Phung Huynh et.al [3] proposed an approach for identifying drowsiness of the driver by examining facial manners like nodding, eye-closure and yawning. They used 3D Convolution Neural Network to pull out features in spatial-temporal domain, and for drowsiness categorization, they used gradient boosting .They applied semi-supervised learning to improve overall performance. Their model exhibited 87.46% accurateness.

Benjamin Murauer et.al [4] used extreme gradient boosting classifier to recognize musical genre from Audio on the Web in the Web Conference 2018. They used Convolution Neural Network (CNN) for spectrogram classification, Deep Neural Networks and ensemble methods using various numerical audio features to predict the genre of specified mp3 music files. They achieved most excellent results by using Extreme Gradient Boosting.

HaoWang, ChuyaoLiu & LeiDeng et.al [5] applied XGBoost and built a model called PredHS2 for detection of hot spots. Hotspots tiny segments of protein-protein interface residues contributing majority of the binding free energy. It was found that performance of PredHS2 was better compared to other machine learning algorithms .It also outperformed many contemporary hot spot forecast techniques equally on the training dataset and test set respectively. A number of new features like disorder scores ,second structure features and solvent exposure characteristics were found to be more helpful in identification of hot spots.

Ayumi, V. et.al [6] examined action identification by using Extreme Gradient Boosting. They compared the performance of XGBoost with some other machine learning classifiers like Support Vector Machine (SVM) and Naive Bayes (NB). Their investigational study on data sets related to human action shown that, XGBoost gave improved accuracy when compared to NB and SVM. although it took extra computational time, the XGBoost exhibited enhanced classification accuracy in action recognition.

Babajide Mustapha, et.al [7] proposed a model to predict the biological activity by using quantitative depiction of a compound’s molecular composition. They used seven mainly familiar datasets of the literature and investigational results revealed that Xgboost smashed Naïve Bayes (NB), Support Vector Machines (LSVM), Random Forest (RF) and Radial Basis Function Neural Network (RBFN) in predicting the biological behavior. Besides detecting minority activity classes’ imbalanced datasets, it also showed amazing performance on low as well as high diversity datasets. Their results show that XGBoost exhibited an accuracy of 94.47% and 98.49% on heterogeneous data and homogeneous data respectively.

Zhang, F.et.al [8] suggested Gradient Boosting Random Convolution Network (GBRCN) model for the purpose of classifying scenes .The GBRCN can efficiently combine a number of Deep Neural Networks. They applied this technique on the most two famous high-resolution data sets that are UC Merced data set which has 21 dissimilar categories of aerial scene with a sub meter resolution and a Sydney data which includes eight land-use categories with a 1.0-m spatial resolution. It was observed that GBRCN performed better than the high-tech techniques on the UC Merced data set and Sydney data set also.

Urraca, R. et.al [9] implemented a novel Gradient Boosting Machine (GBM) in order to predict global horizontal irradiation at some places, where pyranometer documentation was not available. They carried out the study with the data collected from almost 38 ground stations in Castilla-La Mancha during the years 2001 to 2013. They observed that that their model had good generalization capacity and obtained an average MAE of 1.63MJ/m2 in stations that are not used to attune the model .It performed better than all other statistical models that were existed in the Spain literature. A thorough investigation of the errors was done to figure out the distribution of errors according to the level of radiation and clearness index. In addition to that, the involvement of each input was also assessed.

Song, R. et.al [10] used a probabilistic approach for identifying individual users across diverse digital devices and compared the performance of diverse classification algorithms .They carried out a thorough study and expanded the attributes of data by studying the relationship among attributes. They added dummy variables to improve the efficiency of the models. They experimented on four datasets released by ICDM Challenge and the results show that, the eXtreme Gradient Boosting gave better accuracy and F1-score compared to all other algorithms.

Sheridan, R. P.et.al. [11] Compared eXtreme Gradient Boosting (XGBoost) with Single-Task Deep Neural Nets and Random Forest for Quantitative Structure-Activity Relationships on thirty in-house data sets. They identified some hyper parameters at which XGBoost gives good predictions. It performed better than Deep Neural Nets and Random Forest. Besides, to use Random Forest efficiently, we need to generate the trees on a cluster in parallel. Deep Neural Nets execute mostly on GPUs but XGBoost can run on a single CPU in less than a third of the wall-clock time of the other two methods.

Johnson, N. E. et.al [12] used gradient boosting regression to foretell municipal concrete waste generation throughout the New York City. The model training was done using historical data collected during 2005 to 2011 and they did validation both spatially and temporally.

Their model could precisely predict weekly generation of MSW in tonnages in all the 232 regions in New York City across three waste streams that are paper , refuse and metal/-glass/plastic. Significantly, their model could forecast frequency of waste production in urban areas and capable of capturing variations during special events, holidays, weather related events and seasonal variations also. This research showed the trends in waste generation of New York City and the significance of abundant data collection so as to accurately forecast waste generation.

Aler, R. et.al [13] used machine learning with gradient boosting to divide components of diffuse and direct solar radiation. They applied XGBoost with ensembles of both linear and non-linear weak prediction models. The predictions given by 140 models were combined using XGBoost in order to reduce the randomly generated errors in the predictions by individual models at all of the validation sites. The least prediction error was exhibited by a mixture of 26 models out of the 140 models. Most of these 26 models used a minimum of three inputs besides clearness index. Out of 24 probable inputs that were used in the original 140 models, merely 14 inputs were found significant. It was observed that when validation and training datasets were not collocated, RMSD of the predictions increased by 2%. By and large, their results indicate that a data-driven ML proposal by merging a restricted number of existing models significantly reduce the huge random errors with such models if used independently at high temporal frequency.

Manning, B. (2017) applied eXtreme Gradient Boosting for classification of a variety of users by using HCI based behavioral biometrics for identifying insider hacks in a system.. This method could be used to authenticate users after they have gained entry into a protected system using data that is as human-centric as other biometrics, but less insidious. For training and testing, they used extreme boosting algorithm on dataset consisting information of keystroke dynamics. The ultimate predictive model gave an accuracy of 0.941 with a Kappa value of 0.942 proving that HCI-based behavioral biometrics in the form of keystroke dynamics can be used to recognize the users of a system.

Ajit, P et.al [15] applied Extreme Gradient Boosting (XGBoost) with regularization to predict Employees turnover in organizations .They wanted to replace traditional over fitted and inaccurate machine learning models due to noise with their new model. They used HRIS data of global retailers' .It was observed that XGBoost with regularization procedure gave higher accuracy in predicting employees turnover compared to six historically used supervised classifiers.

Dhaliwal, S et.al [16] designed a model to compute diverse parameters like accuracy, precision, confusion matrix of data in a network to in order study the security aspects of data in the network. They applied XGBoost classifier on the data set called NSL-KDD (Network Socket Layer-Knowledge Discovery In Databases) to learn about the integrity of data, attain the essential results and obtain better prediction accuracy on the dataset. Their motto was to minimize the quantity of harmful data that is floating in a network so as to make the network a secure place for sharing information, control hacking and unauthorized modifications. Studying and analyzing patterns and quantity of data helps to the materialization of superb Intrusion Detection Systems (IDS) as a result the network becomes a healthy and safe place to share confidential information

Mesut Gumus et.al [17] used XGBoost to identify the parameters that are affecting the cost of crude oil. Variations of crude oil prices play a critical role in budget planning and treasury of many crude oil importing countries. Forecasting the price of crude oil and proper plan will save lots of money in many government and corporate economies. This kind of estimation is vital in countries where crude oil production is very low and depends mainly on crude oil import.

Chen, T., & Guestrin, C et.al [18] recommended a new sparsity conscious algorithm for weighted quantile plan and sparse data for learning rough tree. They demonstrated an end-to-end scalable tree boosting technique called XGBoost. More significantly, they focused on data compression, cache access patterns and sharding to build their scalable tree boosting scheme. By merging these techniques, XGBoost scaled ahead of billions of examples with lesser resources when compared to existing systems

Zheng, H., Yuan, J, et al [19] constructed a model for efficient functioning of a power system by short term load forecasting They constructed a hybrid algorithm called SD-EMD-LSTM by combining Empirical Mode Decomposition (EMD), Similar Days (SD) selection and Long Short-Term Memory (LSTM) Neural Networks. Extreme gradient boosting using weighted k-means method was used to assess the resemblance between historical and forecasting days. They used EMD technique to divide the Similar Days load into a number of intrinsic mode functions (IMFs) and residual. Separated LSTM Neural Networks were employed for predicting each residual and IMF . Finally, they reconstructed the forecasted values by each LSTM model. Mathematical testing revealed that their technique could accurately estimate the electric load

III. METHODOLOY

The working of Xgboost is shown in the subsequent figure. In the process of boosting, we build competent models from a set of individual weak learners in an iterative way. Initially all the samples in the data set have same weight and the first model is trained by randomly picking some samples from the data set ,where every sample has equal chance to participate in training. Every model is tested on all the samples and weight of the wrongly classified samples is updated so that they are picked for the training of the next model. It builds a number of models in a sequential manner. When a test sample is to be predicted, then the predictions of majority of the models are considered and that will be the final prediction.

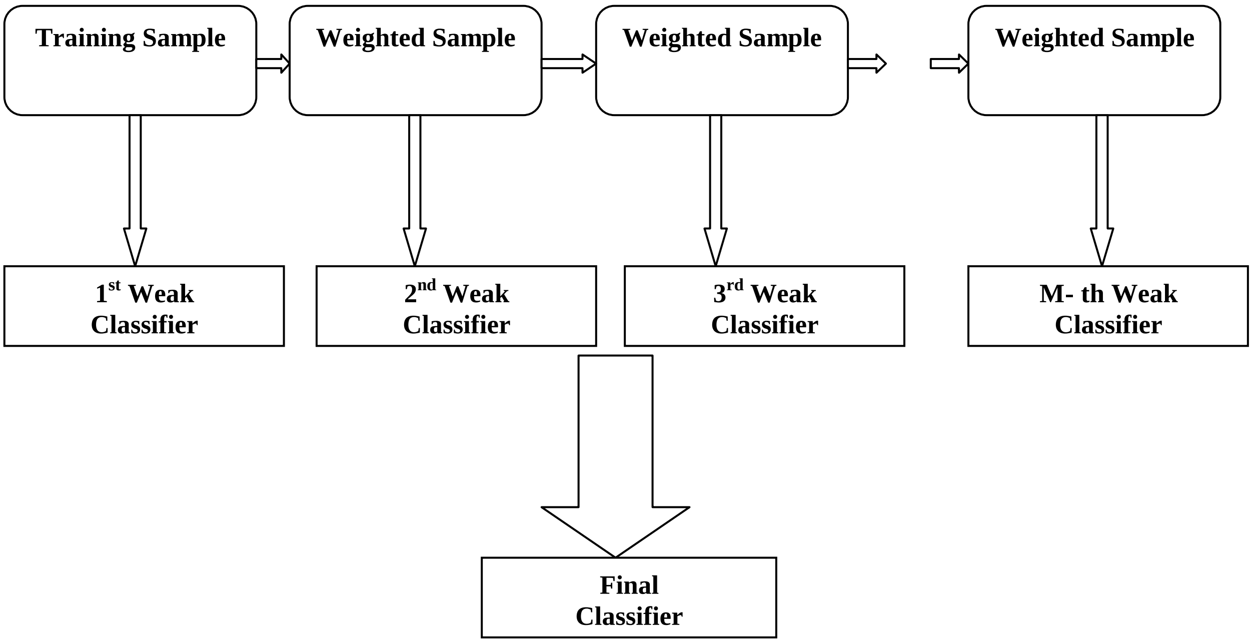

Fig.1: Working Model of Xgboost Classifier

The main problem with the majority boosting methods is that they don't take regularization issues very seriously. They allow grow many very similar trees that can be also at times somewhat shaggy. In Gradient Boosting Technique, there are some regularization parameters like minimum samples per leaf, maximum depth learning rate etc using which we can control the structure of the tree. It can be improved even more by using extreme gradient boosting. It is a more regularized edition of Gradient Boosted Trees.

We applied L2 regularization of XGBoost to get a more generalized model. Regularization technique makes minute modifications to the learning algorithm as a result, the model generalizes better. Regularization tries to push the weights of several variables to zero .In this process, it performs variable selection also. Selection of variables plays a vital in high dimensional problems. This, in sequence improves the model's performance on the unseen data too.L1 and L2 are widely used regularization techniques. For regularization of the model, another term called regularization term is added to the cost function as stated below

Cost function = Loss + Regularization term

As a result, the values of weight matrices decrease. It is inherent that a Neural Network with smaller weight matrix gives simpler models and also reduces over fitting to a large extent. If we set a big value for regularization lambda, then to minimize the cost, the network will set the weight matrices W to be reasonably close to zero for a bunch of hidden units. The impact of many hidden units is minimized in this process and results in a much smaller Neural Network.

In L2 regularization function is:

$$Cost\ function = Loss + \frac{\lambda}{2m} * \sum \| w \|^2 \qquad (1)$$

In the equation above, lambda is the regularization parameter and m indicates number of classes. The equation above states that cost function is loss plus $\frac{\lambda}{2m}$ times the sum of norm of weight matrix w ,squared. Lambda is a hyper parameter whose value can be optimized for better results.

Table I: Best parameters found for the XGBoost model for liver data set.

General Parameters		Tree learner		Regularization		Sub sampling	
Booster	gbtree	Base Score	0.5	Reg_alpha	0	Colsample by level	1
n_jobs	1	Learning rate	0.07	Reg_lambda	1	Colsample by tree	1
Objective	binary:logostic	Max-depth	3			Sub sample	1
		Min child weight	1				
		n_estimators	50				
		Random state	43				
		Scale_pos_weight	1				

Booster is a **kind** of model which is built at every iteration of the algorithm. We used gradient boosting trees in our model. The number of cores to be used for parallel processing is indicated by n_jobs .In this case it is set to 1.Objective defines the loss function which is to be decreased. We used logistic loss function as ours is a binary classification problem, so that it returns predicted probability and not the class. The equations below illustrate calculating log loss for one observation. When estimating a model on a dataset, the log loss score is the average log loss score of the entire observations.

In the equation below , N denotes number of observations, M is the possible number of class labels, log implies the natural logarithm, y is 0 or 1 indicating whether class label c is the right classification for observation o, p is the predicted probability by the model that the observation o belongs to class c.

In binary classification (M=2), the logistic loss formula equals:

$$-(y\log(p)+(1-y)\log(1-p)) \qquad (2)$$

For illustration, given a class label of 1 and a predicted probability of .25, using the formula above, we can calculate the log loss as

$$-(1\log(.25)+(1-1)\log(1-.25))$$
$$-(\log(.25)+0\log(.75))$$
$$-(\log(.25)$$
$$-(\log(.25)$$

The model can be made more robust by adjusting the learning rate parameter **which** minimizes the weights on each step. We took 0.07 as the learning rate value. Maximum depth of the tree is specified by the parameter the max_depth. It is set to 1 so as to control over-fitting because higher depth values will make the model teach relations that are extremely specific to a particular sample. The value of Min child weight is set to 1, which is the minimum total of weights of all observations needed in a child. Higher values of this lead to over fitting and too high values can cause under fitting also. The value of n estimators is set to 50, which specifies the number of trees to be generated. Random state is set to same value so as to validate the results for multiple runs.. It is used for initializing the internal random number generator, which will choose to split data into train and test samples. The value of Scale position weight **to** is set 1**.** If the class imbalance is high in the data set, **a** value greater than 0 should be used for faster convergence.

The value of Reg_lambda is set to 1 which L2 regularization term on weights. Gamma value decides whether to split node or not. **It** specifies the required minimum loss reduction to make a split. A node is split only when the resulting split gives a positive decrease in the loss function

The subsample fraction of columns for each split, in each level is set to 1. Column sample by tree value indicates the portion of columns to be randomly sampled for each successive tree and it is given as 1.Sub sample indicate the portion of observations to be arbitrarily sampled for each tree. Smaller values prevent over fitting but make the algorithm more unadventurous however too small values are also not suggested as it might cause under-fitting.

IV. RESULTS AND DISCUSSIONS

The liver dataset used in this paper is collected from Amrutha Group of Hospitals, Srikakulam, and Andhra Pradesh, India. This liver dataset consists of 11 attributes, out of which 1-10 attributes are considered as input attributes and 11th attribute is considered as target class attribute which is having 0(non-diseased) or 1 class (diseased).

Table II: Attributes Description:

S. No	Attribute	Description
1	Gender	Gender of the patient
2	Age	Patient's Age
3	TB	Total Bilirubin
4	DB	Direct Bilirubin
5	SGOT	Aspartate Aminotransferase
6	SGPT	Alamine Aminotransferase
7	ALP	Alkaline Phosphotase
8	ALB	Albumin
9	GLB	Globulin
10	A/G Ratio	Albumin and Globulin Ratio
11	Class label	Diseased or not (labeled by experts)

This data set contains totally 882 instances; out of which 403(45.7%) instances are of class 0(non-diseased) and 479 (54.3%) instances are of class 1(diseased).

Outcome is a class label to split the samples into two classes (liver patient or not). 90.3% of the samples are used to train 50 weak classifiers and rest of the samples is used for testing the model.

Below figure shows the importance of the various features in the data set. It is observed that the feature SGPT has been given highest importance score among all the features and the feature gender has been given least importance.

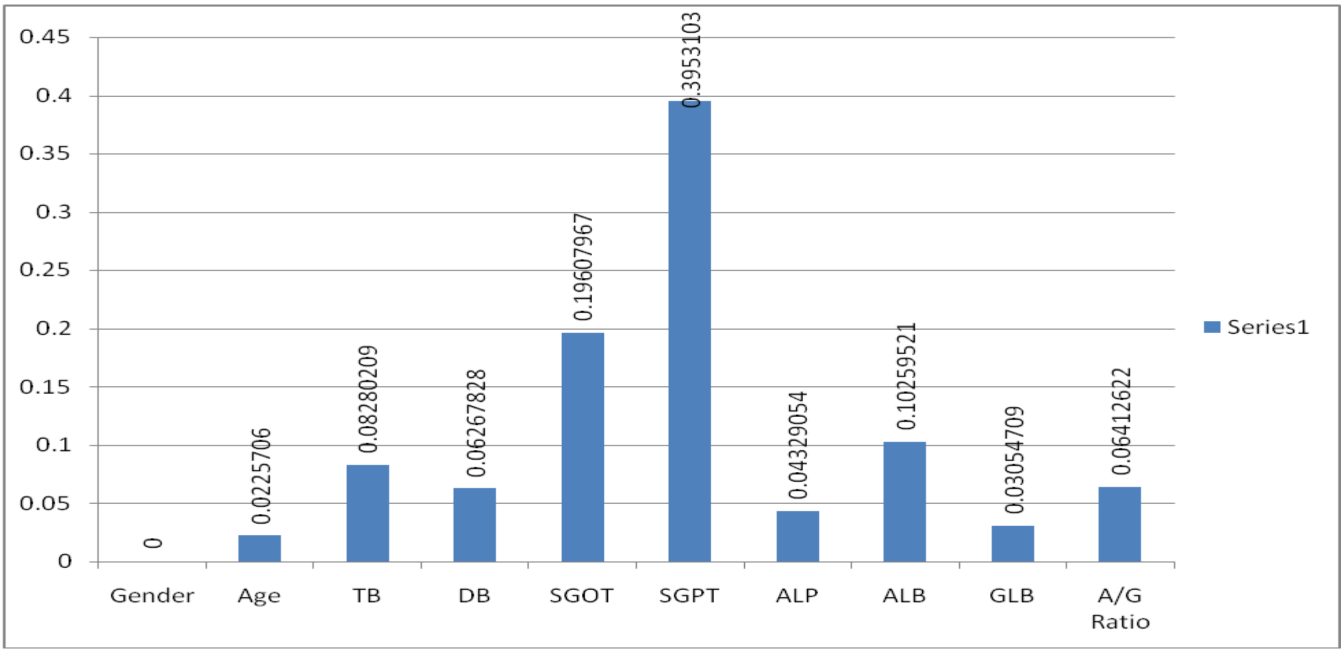

Fig.2:Feature importance graph

The Below figure shows the first weak tree that is generated by the XGBoost algorithm

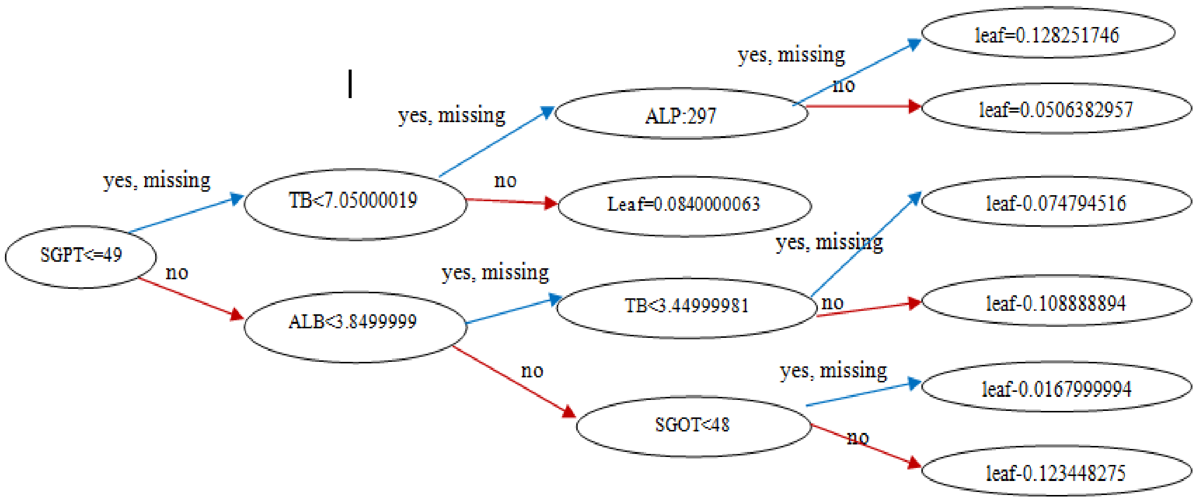

Fig.3. First weak classifier generated by the XGBoost algorithm

In the first weak tree, the features that participated in the splits are SGPT, TB,, ALB,ALP,SGOT. Hence we can say that these attributes played key role in predicting liver diseases initially.

The feature importance changes as more and more trees are built.

The Below figure shows the final classifier that is 50th tree generated by the XGBoost algorithm

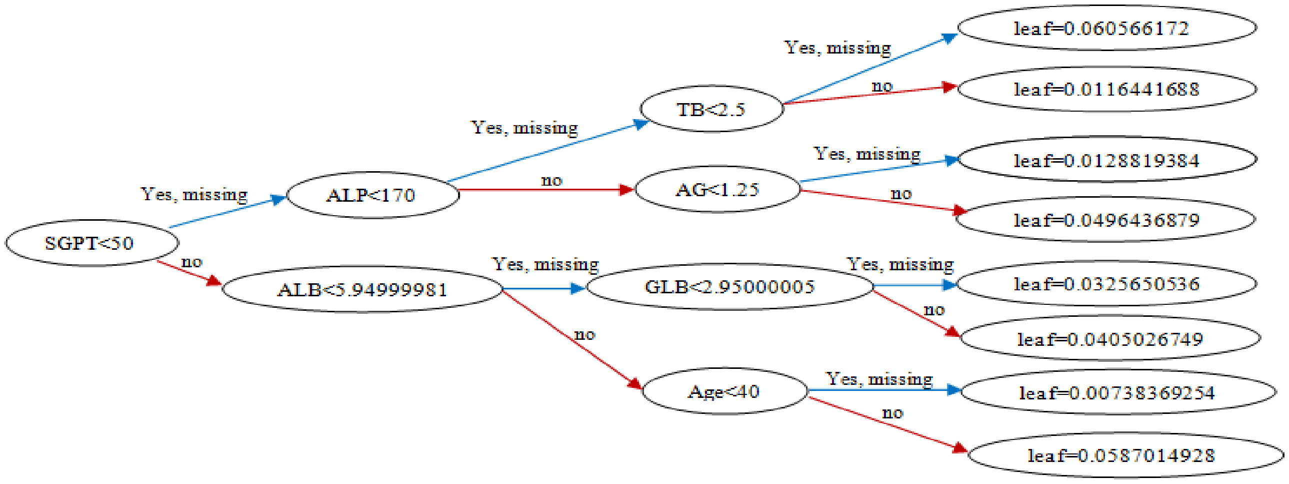

Fig.4. 50th model generated by the XGBoost algorithm

From the above tree it is observed that DB,SGOT,AGE did not play any role in predicting the liver disease. The tree above is interpreted as, if SGPT<50 and ALP< 170 and TB<2.5, it returns 0.060566172 else returns 0.0116441688.if SGPT<50 and ALB NOT < 170 and AG<1.25, it returns 0.0128819384 else returns 0.0496436879.if SGPT not<50 and ALB<5.94999981 and GLB<2.95000005 then return 0.0325650536 else return 0.0405026749.if SGPT not<50 and ALB not <5.94999981 and AGE <40 then return 0.00738369254 else return 0.0587014928.The leaf nodes contain the raw score for a class. It can be converted to a probability score using the formula below

1/ (1+np.exp (-1*leaf value)) (3)

The probability score tells the probability of that object being class yes or class no, if a data point ends up being distributed to this leaf. In the final tree, the features that participated in the splits are SGPT, ALP, ALB, TB, AG, GLB, and AGE.

Table III: Accuracy Comparison of Various classification Algorithms

Algorithm	**Naïve Bayes**	**C4.5**	**AD Tree**	**SVM**	**RBF**	**MLFFDNN**	**XGBoost**
Accuracy	71%	97%	92%	75%	83%	**98%**	99%

Accuracy comparison of Different classification methods is shown in table 3. From table 3, it is observed that XGBoost technique exhibits 99% accuracy score compared to other classification methods. In this paper, mainly, we considered accuracy score as the metric because our dataset is having balanced target classes.

V. CONCLUSION AND FUTURE WORK

We tuned an efficient XGBoost model by using important hyper parameters such as L2 regularization, logistic loss function, learning rate and number of estimators. It is observed that the model tuned by us is giving 99% accuracy on liver data sets. Fig. 4 shows the 50th tree that was generated by the XGBoost algorithm. It is observed that our model gave 99 % accuracy which more than our previously implemented model called Multi layer Feed Forward Deep Neural Network (MLFFDNN) with 98% accuracy and also some widely used existing classification techniques like Naïve Bayes (NB), C4.5, AD tree, RBF Network, Support Vector Machines (SVM).In future we will try to further improve the accuracy by using other boosting techniques.

REFERENCES

1. Carmona, P., Climent, F., & Momparler, A. (2019). Predicting failure in the US banking sector: An extreme gradient boosting approach. International Review of Economics & *Finance*, *61*, pp: 304-323.
2. Georganos, S., Grippa, T., Vanhuysse, S., Lennert, M., Shimoni, M., & Wolff, E. (2018). Very high resolution object-based land use–land cover urban classification using extreme gradient boosting. *IEEE Geoscience and Remote Sensing Letters*, *15*(4), 607-611.
3. Huynh, X. P., Park, S. M., & Kim, Y. G. (2016, November). Detection of driver drowsiness using 3D deep neural network and semi-supervised gradient boosting machine. In *Asian Conference on Computer Vision* (pp. 134-145). Springer, Cham.
4. Murauer, B., & Specht, G. (2018, April). Detecting Music Genre Using Extreme Gradient Boosting. In *Companion of the The Web Conference 2018 on The Web Conference 2018* (pp. 1923- 1927). International World Wide Web Conferences Steering Committee
5. Wang, H., Liu, C., & Deng, L. (2018). Enhanced prediction of hot spots at protein-protein interfaces using extreme gradient boosting. *Scientific reports*, *8*(1), 14285.
6. Ayumi, V. (2016, December). Pose-based human action recognition with Extreme Gradient Boosting. In *2016 IEEE Student Conference on Research and Development (SCOReD)* (pp. 1-5).

7. Babajide Mustapha, I., & Saeed, F. (2016). Bioactive molecule prediction using extreme gradient boosting. *Molecules*, *21*(8), 983.[8] Zhang, F., Du, B., & Zhang, L. (2016). Scene classification via a gradient boosting random convolutional network framework. *IEEE Transactions on Geoscience and Remote Sensing*, *54*(3), 1793-1802.
8. Urraca, R., Antonanzas, J., Antonanzas-Torres, F., & Martinez-de-Pison, F. J. (2016, October). Estimation of daily global horizontal irradiation using extreme gradient boosting machines. In *International Joint Conference SOCO'16-CISIS'16-ICEUTE'16* (pp. 105-113). Springer.
9. Song, R., Chen, S., Deng, B., & Li, L. (2016, June). eXtreme gradient boosting for identifying individual users across different digital devices. In *International Conference on Web-Age information Management* (pp. 43-54). Springer, Cham.
10. Sheridan, R. P., Wang, W. M., Liaw, A., Ma, J., & Gifford, E. M. (2016). Extreme gradient boosting as a method for quantitative structure–activity relationships. *Journal of chemical Information and modeling*, *56*(12), pp: 2353-2360.
11. Johnson, N. E., Ianiuk, O., Cazap, D., Liu, L., Starobin, D., Dobler, G., & Ghandehari, M. (2017). Patterns of waste generation: A gradient boosting model for short-term waste prediction in New York City. *Waste management*, *62*, 3-11.
12. Aler, R., Galván, I. M., Ruiz-Arias, J. A., & Gueymard, C. A. (2017). Improving the separation of direct and diffuse solar radiation components using machine learning by gradient boosting. *Solar Energy*, *150*, 558-569.
13. Manning, B. (2017, February). Extreme gradient boosting and behavioral biometrics. In Thirty-First AAAI Conference on Artificial Intelligence.
14. Ajit, P. (2016).” Prediction of employee turnover in organizations using machine learning “,international Journal of Advanced Research in Artificial Intelligence, Vol. 5, No.9, pp:22-26
15. Dhaliwal, S., Nahid, A. A., & Abbas, R. (2018). Effective intrusion detection system using XGBoost. Information, 9(7), 149.
16. Mesut Gumus , M., &Kiran, M. S. (2017, October). Crude oil price forecasting using XGBoost. In 2017 International Conference on Computer Science and Engineering (UBMK) (pp. 1100-1103). IEEE
17. Chen, T., &Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acmsigkdd international conference on knowledge discovery and data mining (pp. 785-794). ACM.
18. Zheng, H., Yuan, J., & Chen, L. (2017). Short-term load forecasting using EMD-LSTM neural networks with aXgboost algorithm for feature importance evaluation. Energies, 10(8), 1168, doi:10.3390/en10081168

AUTHORS PROFILE

S Vishnu Murty, is a research scholar in JNT University ,Kakinada (A.P) ,India . and currently working as Associate Professor in the Department of Computer Science and Engineering at Aditya Institute of Technology and Management, Tekkali, Srikakulam, Andhra Pradesh, India. He received his M.Tech. Degree in Computer Science and Engineering in 2011 from Jawaharlal Nehru Technological University (JNTU), Kakinada, Andhra Pradesh, India. He is pursuing Ph.D. in the Department of Computer Science from JNT University, Kakinada, Andhra Pradesh, India. He published around 7 papers in International Journals. His current Research interests include Data Mining and Machine Learning.

R Kiran Kumar, is currently working as Assistant Professor in the department of computer Science and Engineering and Principal In charge of University college of Engineering and Technology ,Krishna University ,Machilipatnam, Andhra Pradesh, India. He did his MCA from Andhra University, Visakhapatnam, Andhra Pradesh ,India and he received his M.Tech. Degree in Computer Science and Engineering from Jawaharlal Technological University (JNTU), Kakinada, Andhra Pradesh, India. He did his Ph.D. in the Department of Computer Science and Engineering from Nagarjuna University ,Guntur, Andhra Pradesh, India in 2009. He published more than 40 papers in International and National Conferences and Journals. His research interests include Data Warehousing and Mining, Neural Networks, Image Processing, and Pattern Recognition