Diagnostic Support System for Euthyroid Sick Syndrome based on Machine Learning Algorithms Approaches

Rosana C. B. Rego*‡, Vinicius A. Almeida†¶ Caio M. V. Cavalcante†‖ and Nathalee C. A. Lima* § *Departament of Engineering and Technology
Federal University Rural of the Semi-Arid, BR-226, Pau dos Ferros RN, Brazil
Email: ‡ rosana.rego@ufersa.edu.br; §nathalee.almeida@ufersa.edu.br †Information Technology
Federal University Rural of the Semi-Arid, BR-226, Pau dos Ferros RN, Brazil
Email:¶vinicius.almeida37366@alunos.ufersa.edu.br; ‖caio.cavalcante@alunos.ufersa.edu.br

***Abstract*—Euthyroid sick syndrome is a medical condition that affects the thyroid gland. Early and accurate diagnosis is crucial for treatment. However, the interpretation of test results can be subjective. Machine learning algorithms could help doctors in their diagnostics. In this work, we applied machine learning algorithms, such as decision trees, random forest, logistic regression, naive Bayes, and a proposed model to classify the problem based on attributes obtained via blood tests, such as T4, TSH, and T3. Moreover, we presented a diagnostic support system. The final results indicate that the use of algorithms can be helpful in classifying the syndrome and diagnostic support.**

***Index Terms*—Data science, Machine learning, Classification, Healthcare.**

I. INTRODUCTION

Euthyroid Sick Syndrome (ESS) is a condition that presents a problem in the hormonal regulation of the thyroid glands in patients who have some other illness or infection [1]. ESS is assumed to be caused by rising circulating levels of cytokines and other inflammation mediators [2, 3]. These mediators can inhibit the thyroid axis at multiple levels, including the pituitary, i.e., decreased thyroid stimulating hormone (TSH) secretion, decreased thyroxine (T4) and triiodothyronine (T3), decreased thyroid hormone binding, and decreased conversion of T4 to T3 [2].

Although the thyroid is functioning normally, the production of its hormones is affected by inflammation, infection, or another condition [3, 4]. This can result in abnormal levels of thyroid hormones in the blood and symptoms similar to those of thyroiditis or hypothyroidism [5]. The TSH serum levels are low in patients who have a nonthyroidal systemic illness but who are actually euthyroid [2]. In this context, treatment of ESS usually includes treatment of the underlying disease, and assessment of thyroid function may be necessary to monitor the evolution of the condition [3].

McDermott, in [2], considered ESS as an adaptive reaction to decrease tissue metabolism and conserve energy during systemic illnesses. The author pointed out the treatment with thyroid hormone is not generally advised but may be helpful in patients with chronic heart failure. However, the management of ESS is also controversial.

Differentiating the ESS from true hypothyroidism usually can be achieved by the determination of serum T4, T3, and TSH levels and resin T3, which is commonly ordered as free thyroxine index [6].

Earlier and proper diagnosis is crucial for the effective treatment of the ESS. But, the interpretation of serum T4, T3, and TSH levels results can be hard and subjective, leading to misdiagnosis. In this context, machine learning (ML) algorithms may be able to support doctors in diagnosing the patient condition [7, 8].

ML models are able to learn and adapt to data characteristics, which can increase prognosis accuracy, as shown by [7]. By using ML techniques to classify Euthyroid sick syndrome, it is possible to identify patients who may be developing the condition early, which can lead to more effective treatment [9]. Furthermore, the use of algorithms to classify the ESS may save time and resources compared to conventional data analysis methods.

ML algorithms application is a promising approach for the early detection of ESS, as it allows a quick and accurate analysis of data, as well as the identification of patterns and trends in clinical data that can be used to improve the diagnosis and treatment of ESS [8, 10].

Many machine-learning methods have been presented in the literature to enable the early detection of thyroid conditions [7, 8, 9, 10]. In [9], the authors presented some ML models, the best one is a feedforward neural network, which achieves an accuracy of 95.70%. In [10], Mashonga et al. showed that the XG Boost model is a good choice with 98% of accuracy. In [8], a stacked ensemble model based on the decision tree, eXtreme Gradient Boosting (XGBoost), and a feedforward neural network has achieved an accuracy of 99.46%. However, prior research has failed in verifying if the models have suffered from overfitting. The model can achieve high accuracy but can present overfitting. A way to ascertain if the model is suffering overfitting is the analysis of error curve.

The models implemented in this work were: random forest (RF), logistic regression (RL), XGBoost, light gradient-boosting (LightGBM) and a proposed stacking ensemble to produce an optimal model. These are some of the most popular and widely used algorithms in many machine learning applications. Random forest is an extension of the decision tree algorithm, which builds multiple decision trees and combines their predictions to improve model accuracy [11]. Logistic regression is an algorithm used to solve binary classification problems [12]. The XGBoost is a machine learning method based on gradient boosting, which uses decision trees as base learners [13]. As the XGBoost algorithm, the LightGBM is

based on gradient boosting, the main distinction is the high memory efficiency of LightGBM [14].

Motivated by the aforementioned discussion, in this work, we implemented the RF, RL, XGBoost, LightGBM, and a stacking ensemble established on RF-XGBoost ML algorithms to classify the ESS based on attributes obtained via blood tests, such as T4, TSH, and free T4, and T3. Moreover, we provided an intelligent diagnostic support system that improves the precision of the diagnostic.

Thus, the work is divided as follows: In Section II, we described the dataset, presenting characteristics of the data, and the data cleaning process. In Section III, machine learning models fundamentals and characteristics of the models used are presented. In Section IV, the results and discussions are presented. Finally, in Section V, the final considerations are presented.

II. Dataset

We used the Euthyroid Sick Syndrome dataset provided by the University of California Irvin's (UCI) in the public machine learning repository. The dataset contains 25 attributes for each patient, including information on age, sex, medication, pregnancy, surgery, and thyroid function test results. The data, such as age, sex, sickness, TSH, T3, T4, T4U, and free thyroxine index (FTI) were used. We selected these parameters based on coefficients of correlation. The parameters provide important information about the patient's thyroid function and health status. For instance, age and gender can affect the risk of developing thyroid disorders, while TSH, T3, T4, total T4 (T4U), and FTI are indicators of important aspects of thyroid function and can help diagnose thyroid disorders. Furthermore, these parameters can be obtained through a blood test.

A. Data Cleaning

In data science, not all data we encounter is clean. Therefore, it is often necessary to prepare them in a process called data cleaning. Data cleaning aims to remove anomalies and noise from data to improve its quality. This process is part of the machine learning workflow, which is applied to identify and correct errors in datasets that may impact the final machine learning model. Also, to extract meaningful insights from the dataset, the cleaning step is crucial. First, we formatted the dataset, adding column headers. After, we analyzed and removed invalid characters and outline values. Thereafter, we treated the missing data in some columns of the dataset. For the treatment of missing data, we replaced the missing data with the mean of the existing data in the column. In other cases, the mode was applied, for example, in the gender parameter. Fig. 1 (a), is depicted the data with miss values, and Fig. 1 (b), the data after the data cleaning process.

B. Data Balancing

The data needs to be balanced to be introduced into the machine learning algorithm. Hence, the balancing process was implemented using the Synthetic Minority Over-sampling Technique (SMOTE) [15]. This technique generates extra data

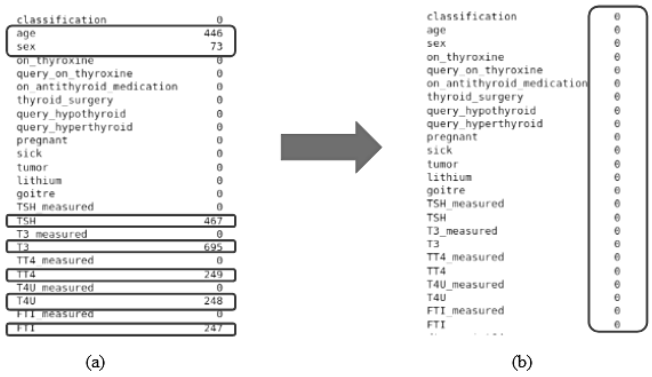

Fig. 1. (a) Data with miss values and (b) Data after cleaning process.

from the minority class to overcome data imbalance. The generation of new data is based on the K Nearest Neighbors (KNN) algorithm. After the formatting, cleaning, and balancing process, the data is ready to be introduced into the machine learning model.

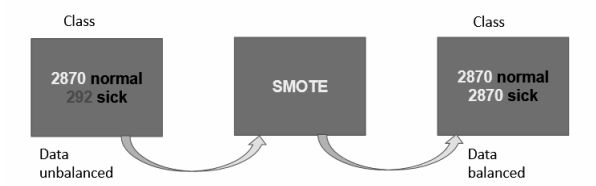

Fig. 2. Data before and after the application of SMOTE technique.

III. Machine learning models

In this section, different machine learning models is presented, such as random forest, logistic regression, XGBoost, LightGBM, and a stacking ensemble with RF and XGBoost to perform the classification.

A. Random Forest

The random forest model is a machine learning algorithm that uses supervised learning for classification or regression [11]. The algorithm uses the simplicity provided by decision trees with randomness. It can build multiple decision trees, thus creating a forest, and then combine the outputs to improve model accuracy. However, if forest growth is not controlled, the model may overfit or underfit [16].

To implement the RF model, we used 10 trees in the forest, with a maximum depth of 10. In addition, the logarithmic function was used to evaluate the quality of the data divisions between the trees and model evaluation. Also, the minimum number of samples needed to be in the leaf node was established as 5, and the minimum number of samples to divide an internal node was established as 2. Finally, as in the decision tree model, weights were inversely assigned proportionally to the frequencies of the classes in the input data.

B. Extreme gradient boosting

XGBoost, proposed by [17], is an algorithm based on gradient boosting. The technique used the gradient to train the decision trees in the ensemble, which implies that it uses the gradient of the loss function to update the tree parameters. The algorithm is faster and more scalable.

XGBoost could deal with unbalanced data, with large datasets, and presented high computational efficiency [14, 17]. We implemented the XGBoost using the official XGBoost library. To adjust the model parameters, initially, we applied the grid search method. However, the method return parameters that fit too much to the training data, provoking overfitting. Hence, we had to manually adujst the parameters carefully, not to cause overffiting.

C. Light gradient boosting

LightGBM, as XGBoost, is a gradient boosting method based on decision trees. The method can be used for both classification and regression. LightGBM created decision trees that rise per leaf, which means that, given a condition, only a single leaf is split depending on the gain. Leaf-based trees can sometimes overfit, especially with smaller datasets. But limiting the depth of the tree can help to avoid overfitting [14].

To implement the LightGBM model, we applied a boosting learning rate of 0.3, and we set the maximum tree depth as 15 with a number of boosted trees as 5. Moreover, we defined the maximum number of tree leaves for base learners as 15.

D. Logistic Regression

Logistic regression is a classification algorithm that employs the multivariate analysis technique applied to verify the probability of a given event arising from the identification of specific characteristics within each category determined by the division of the belonging area [18]. This technique has direct static tests and the incorporation of non-linear effects [19].

To implement the algorithm, we used the scikit-learn library with the one-vs-rest (OvR) training scheme [20]. We set weights inversely proportional to the class frequencies in the input data. In addition, the random state mode was also used to control the random number generator.

E. Stacking ensemble

Stacking is an ensemble machine learning approach where multiple models are trained to produce predictions on the same dataset, and a meta-model is trained to produce predictions based on the outputs of the individual models. The main idea is to combine the strengths of models, while minimizing their weaknesses [21, 22].

We implemented a stacking based on voting, where the output of each base model is treated as vote, and the meta-classifier makes a final classification based on the votes. We combine random forest model with XGBoost classify, as depicted in Fig. 3. For implementation, we used Stacking Classifier model from scikit learn library.

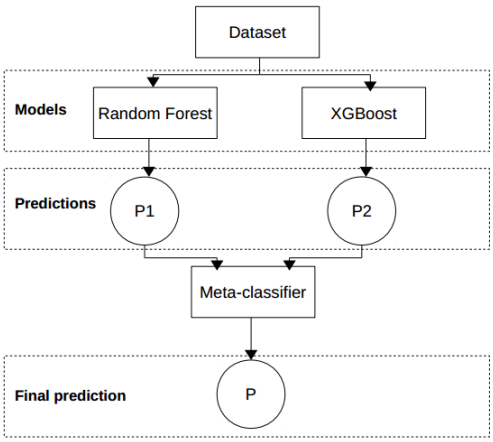

Fig. 3. Stacking ensemble Random-XGBoost.

IV. DISCUSSIONS AND RESULTS

A. Models evaluation

To evaluate the number of patients correctly classified with random forest, XGBoost, LightGBM, logistic regression, and stacking ensemble random-XGBoost model, we used metrics such as confusion matrix, classification error curve, accuracy, recall, precision, and F1-score.

Fig. 4 (a), (b), (c), (d), and (e) depicted the confusion matrices for each model. In Fig. 4 (a), we observed that random forest correctly classified 579 patients as normal and 550 patients as sick, with misclassification of 9 normal patients as sick and 10 sick patients as normal. The logistic regression, in Fig. 4 (b), classified 534 patients as normal and 522 patients as sick, with misclassification of 54 normal patients as sick and 38 sick patients as normal.

XGBoost classified 580 patients as normal and 552 patients as sick, with misclassifications of 8 normal patients as sick and 8 sick patients as normal. On the other hand, LightGBM classified 576 patients as normal and 545 patients as sick, with misclassification of 12 normal patients as sick and 15 sick patients as normal. Therefore, each model had different results.

As depicted in Fig. 4 (e), random-XGBoost model classified 581 patients as normal and 553 as sick, with the misclassification of 7 normal patients as sick and 7 sick patients as normal. Hence, random-XGBoost presented the best performance between classifies.

Fig. 5 (a), (b), (c), (d) and (e) displayed the models' misclassification error curves. Analysis of the curves allows acknowledgment of whether the model is suffering from overfitting or underfitting. The random forest, XGBoost, random-XGBoost, and LightGBM models had a good fit, initially with a elevated error during training and testing, which gradually declined when adding more data and gradually dropped off. Differently, logistic regression model, benefits from adding more data, both during training and during testing, however, the model starts suffering overfitting.

According to Table I, the XGBoost and random forest models perform well with accuracies of 98.60% and 98.34%, re-

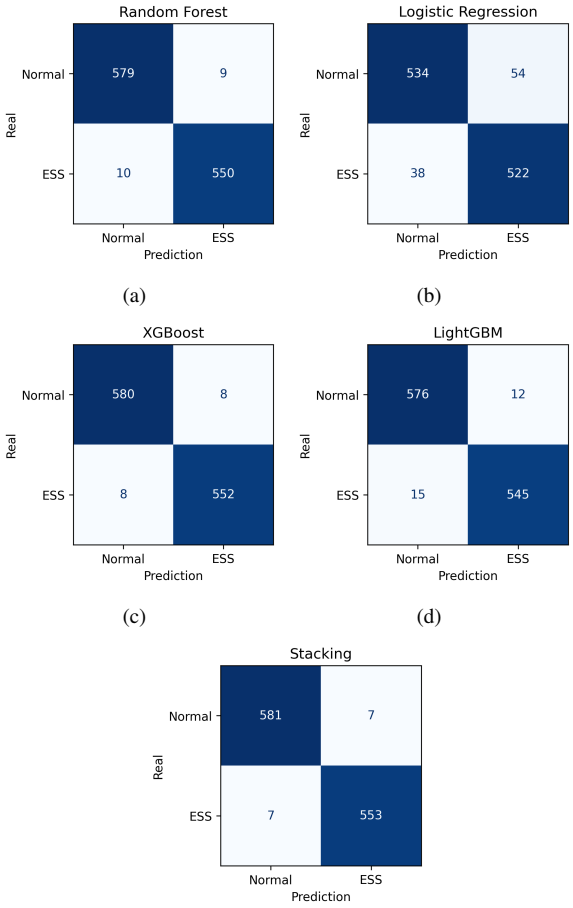

Fig. 4. Confusion matrix: (a) RF model, (b) LR model, (c) XGBoost model, (d) LightGBM, and (e) Random-XGBoost.

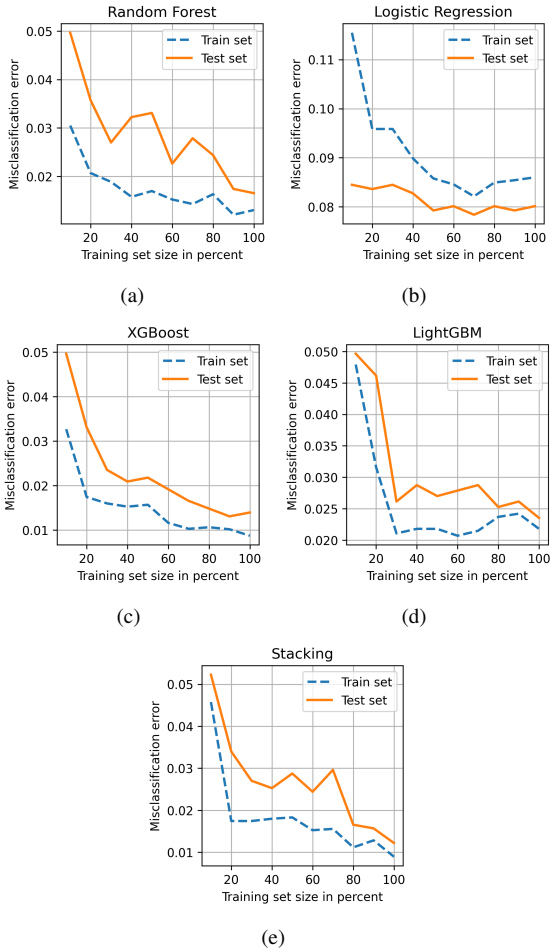

Fig. 5. Misclassification error: (a) RF model, (b) LR model, (c) XG Boost model, (d) LightGBM, and (e) Random-XGBoost.

spectively. The LightGBM model, on the other hand, presents a good performance, but slightly below the random forest and XGBoost, with an accuracy of 97.86%, a recall of 97.32% and a precision of 97.64%. The logistic regression model achived an accuracy of 91.98%, a recall of 93.21% and a precision of 90.62%, indicating a performance below the other models. The F1-score of 91.90% indicates that it is failing to properly balance accuracy and recall. The best metrics results are achieved with the random-XGBoost model with an accuracy of 98.78%, a recall of 98.75%, a precision of 98.75%, and an F1-score of 98.75%.

TABLE I
MODEL METRICS.

Model	Accuracy	Recall	Precision	F1-score
Random forest	98.34%	98.21%	98.38%	98.30%
XGBoost	98.60%	98.77%	98.57%	98.57%
LightGBM	97.64%	97.32%	97.64%	97.58%
Logistic regression	91.98%	93.21%	90.62%	91.90%
Random-XGBoost	98.78%	98.75%	98.75%	98.75%

B. Diagnostic Support System

We implemented a diagnostic support system using the model that best performed, i.e., the random-XGBoost model.

Fig. 6 and 7 shows the first and second page of the diagnostic support system, respectively. In first page is presented a explanation about ESS. In page 2, we have the button for perform prediction. When the user clicks the button, it takes to page 3 which is shown in the Fig. 8. On page 3, the patient data fields are displayed as age, sex, sickness, TSH, T3, T4, T4U, and free thyroxine index.

The system could support and help identify ESS. By using the diagnostic support system, doctors can improve the speed of the diagnostic. However, we should acknowledge that the system should not be used as an isolated basis of information or as an absolute decision. A healthcare professional should use their clinical knowledge and judgment to evaluate the results, ensuring that patients will receive the appropriate and safe treatment.

V. CONCLUSIONS

The results demonstrated that the use of machine learning algorithms for classifying Euthyroid Sick Syndrome is a helpful tool to assist in the efficient diagnosis of the condition. The work demonstrates that the combination of medical data and machine learning methods is a hopeful strategy to improve diagnostic processes and possibly increase patients' quality of life.

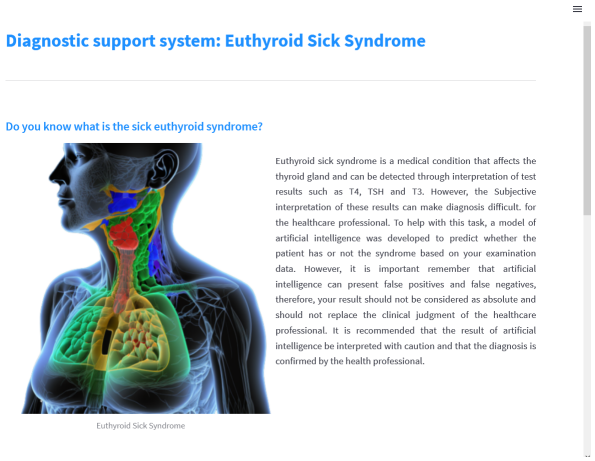

Fig. 6. Interface first page of the proposed system.

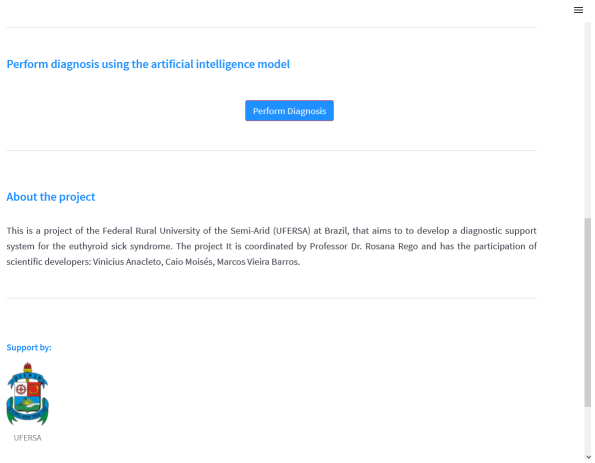

Fig. 7. Interface second page of the proposed system.

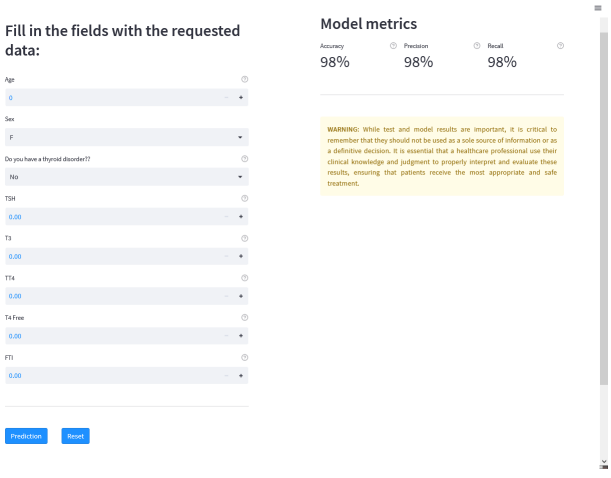

Fig. 8. Interface third page of the proposed system.

ACKNOWLEDGMENT

To PICI/UFERSA for financial support in granting a Scientific Initiation scholarship and UFERSA/PROPPG 12/2020 support for research groups.

REFERENCES

[1] D. Sidebotham, *Cardiothoracic critical care*. Elsevier Health Sciences, 2007.

[2] M. T. McDermott, *Endocrine Secrets E-Book*. Elsevier Health Sciences, 2019.

[3] J. J. Zimmerman and B. P. Fuhrman, *Pediatric Critical Care E-Book*. Elsevier Health Sciences, 2011.

[4] M. Zacharin, *Practical pediatric endocrinology in a limited resource setting*. Academic Press, 2013.

[5] F. E. Wondisford and S. Radovick, *Clinical Management of Thyroid Disease E-Book*. Elsevier Health Sciences, 2009.

[6] D. B. Allen, S. A. Hagen, and A. L. Carrel, "Disorders of the endocrine system relevant to pediatric critical illness," in *Pediatric Critical Care*. Elsevier, 2006, pp. 1105–1124.

[7] Y.-T. Lu, H.-J. Chao, Y.-C. Chiang, and H.-Y. Chen, "Explainable machine learning techniques to predict amiodarone-induced thyroid dysfunction risk: Multicenter, retrospective study with external validation," *Journal of Medical Internet Research*, vol. 25, p. e43734, 2023.

[8] M. Karmeni, E. B. Abdallah, K. Boukadi, and M. Abed, "Towards an accurate stacked ensemble learning model for thyroid earlier detection," in *2022 IEEE/ACS 19th International Conference on Computer Systems and Applications (AICCSA)*. IEEE, 2022, pp. 1–8.

[9] S. S. Islam, M. S. Haque, M. S. U. Miah, T. B. Sarwar, and R. Nugraha, "Application of machine learning algorithms to predict the thyroid disease risk: an experimental comparative study," *PeerJ Computer Science*, vol. 8, p. e898, 2022.

[10] A. Mashonga, L. KudzaiNyandoro, and K. Zvarevashe, "A comparative analysis of the effectiveness of feature engineering techniques on thyroid disease prediction," in *2022 1st Zimbabwe Conference of Information and Communication Technologies (ZCICT)*. IEEE, 2022, pp. 1–6.

[11] K. Fawagreh, M. M. Gaber, and E. Elyan, "Random forests: from early developments to recent advancements," *Systems Science & Control Engineering: An Open Access Journal*, vol. 2, no. 1, pp. 602–609, 2014.

[12] S. Dreiseitl and L. Ohno-Machado, "Logistic regression and artificial neural network classification models: a methodology review," *Journal of biomedical informatics*, vol. 35, no. 5-6, pp. 352–359, 2002.

[13] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou *et al.*, "Xgboost: extreme gradient boosting," *R package version 0.4-2*, vol. 1, no. 4, pp. 1–4, 2015.

[14] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, 2017.

[15] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[16] A. Cutler, D. R. Cutler, and J. R. Stevens, "Random forests," *Ensemble machine learning: Methods and applications*, pp. 157–175, 2012.

[17] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[18] F. Thabtah, N. Abdelhamid, and D. Peebles, "A machine learning autism classification based on logistic regression analysis," *Health information science and systems*, vol. 7, pp. 1–11, 2019.

[19] D. G. Kleinbaum, K. Dietz, M. Gail, M. Klein, and M. Klein, *Logistic regression*. Springer, 2002.

[20] Y. Alhessi and R. Wicentowski, "Swatac: A sentiment analyzer using one-vs-rest logistic regression," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 2015, pp. 636–639.

[21] D. H. Wolpert, "Stacked generalization," *Neural networks*, vol. 5, no. 2, pp. 241–259, 1992.

[22] L. Breiman, "Stacked regressions," *Machine learning*, vol. 24, pp. 49–64, 1996.