Early prediction of hypothyroidism based on feature selection and explainable artificial intelligence

Caio M. V. Cavalcante[1], **Rosana C. B. Rego**[2]

[1] Tecnologia da Informação
Computacional Intelligence Laboratory
Universidade Federal Rural do Semi-Árido (UFERSA) – Pau dos Ferros, RN – Brazil

[2]Departamento de Engenharia e Tecnologia
Universidade Federal Rural do Semi-Árido (UFERSA) – Pau dos Ferros, RN – Brazil

caio.cavalcante@alunos.ufersa.edu.br, rosana.rego@ufersa.edu.br,

Abstract. *Early and accurate diagnosis is required for adequate treatment of hypothyroidism. However, the presence of subjectivity in the interpretation of test results presents a significant challenge. In this study, we explored and evaluated the potential of machine learning (ML) algorithms for addressing this issue. These algorithms include decision trees, random forest, XGBoost, LightGBM, extra trees, gradient boosting, and a stacking ensemble model. The purpose is to predict hypothyroidism, which is a medical condition that affects the thyroid gland, using attributes derived from blood test results. These attributes include thyroxine, thyroid stimulating hormone, free thyroxine index, total thyroxine, and triiodothyronine. The results demonstrate the effectiveness of utilizing these algorithms for accurately classifying hypothyroidism and offering diagnostic assistance with 99.16% of accuracy.*

Index Terms - Classification, machine learning, hypothyroidism, thyroid.

1. Introduction

Hypothyroidism is a clinical condition that affects the thyroid gland in the human body. If left untreated, it can contribute to various health issues, such as high blood pressure, abnormal lipid levels, fertility problems, cognitive decline, and problems with the nerves and muscles [Gaitonde et al. 2012].

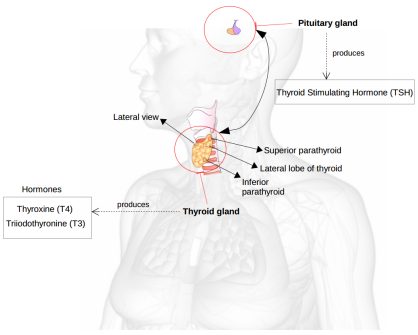


Figure 1. Thyroid gland and its connection with the pituitary gland situated in the brain.

Hypothyroidism happens when the thyroid gland does not produce enough thyroid hormones [Hueston 2001]. As depicted in Fig. 1, the thyroid gland is located in the front of the neck, and it is responsible for producing two hormones thyroxine (T4) and triiodothyronine (T3). The T4 and T3 by the thyroid gland are primarily regulated by the thyroid-stimulating hormone (TSH) produced by the pituitary gland. The pituitary gland produces TSH in response to thyrotropin-releasing hormone (TRH) from the hypothalamus. TSH plays a pivotal role in the regulation of thyroid function [Kostoglou-Athanassiou and Ntalles 2010]. The feedback loop between the thyroid and pituitary gland guarantees that the thyroid gland produces a proper quantity of thyroid hormones to satisfy the body's metabolic demands. When thyroid hormone levels are too low, the system is stimulated to produce more hormones, and when they are too high, the system is inhibited to prevent excessive hormone production [Bensenor et al. 2012]. Diagnosing hypothyroidism involves a combination of clinical evaluation, laboratory tests, and assessment of various symptoms and medical history [Vaidya and Pearce 2008]. Generally, the most common blood tests used to diagnose include TSH measure, Free Thyroxine (FT4), T3, and Thyroid Peroxidase Antibody (TPOAb) [Gaitonde et al. 2012]. However, are these features sufficient for diagnosis? Perhaps using a ML algorithm can help in determining which features are crucial for making a diagnosis.

The use of ML models is demonstrating promise in detect hypothyroidism early [Duan et al. 2022, Hu et al. 2022]. This allows quickly and accurately check medical information for patterns and important details [Cavalcante et al. 2023]. These findings can improve how doctors diagnose and treat hypothyroidism [Shahid et al. 2019]. Moreover, ML algorithms can be helpful for feature selection in the context of hypothyroidism diagnosis by choosing the most relevant variables that are most likely to contribute to the diagnosis [Chaganti et al. 2022]. Many different ML methods have been suggested in scientific literature to help identify thyroid disorders at an early stage [Shahid et al. 2019, Guleria et al. 2022]. However, previous studies have not adequately designed models in a way that allows humans to understand and interpret their decisions and reasoning processes by creating feature importance scores, generating textual or visual explanations, and ensuring the model's internal workings are accessible for review.

Motivated by the aforementioned discussion, this research conducted a comparative analysis of Random Forest (RF), Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM), Extra Trees (ET), Gradient Boosting (GB), Decision Tree (DT), and a proposed approach by explain the model output with SHapley Additive exPlanations (SHAP) [Lundberg and Lee 2017] and Local Interpretable Model-Agnostic Explanations (LIME) [Ribeiro et al. 2016] to provide insight into how a ML model arrives at its predictions.

The manuscript is organized as follows: In Section 2, we discussed some related work and its models metrics. In Section 3, we explained the dataset used, and preprocessing steps undertaken to ensure data quality and consistency, from data cleaning, and data balancing. In Section 4, we discussed some feature selection tecniques. In Section 5, we presented some ML models, providing a comprehensive overview of the methodologies employed. In Section 6, we presented the core of the paper, combining discussions and results. Finally, in Section 7 the conclusions is presented.

2. Related work

In [Shahid et al. 2019], the authors proposed the application of Support Vector Machine (SVM), Random Forest (RF), and K-nearest neighbors (K-NN) algorithms for hypothyroidism diagnosis. The best results are achieved with the RF model. The models metrics are presented in Table 1. Differently, in [Guleria et al. 2022], the authors trained Naive Bayes (NB), Decision Tree (DT), RF, and Multiclass Classifier (MC). The most satisfactory result is reached by decision tree algorithm. Like [Guleria et al. 2022], in [Almahshi et al. 2022], the authors demonstrated that the decision tree algorithm outperforms the others models. Stroek et al. [Stroek et al. 2023] implemented six ML algorithms, the multilayer perceptron (MLP) achieved better accuracy, as depicted in Table 1. Differently, in [Sankar et al. 2022], the authors worked with Logistic Regression (LR), DT, KNN, and XGBoost (XGB) algorithms. The XGB model outperforms other models with higher accuracy.

Table 1. Models used in different works

Paper	Model	Accuracy	Precision	Recall	F1-score
[Shahid et al. 2019]	SVM	97.27	76.80	75.90	76.36
	RF	98.74	86.51	92.77	89.53
	KNN	63.15	69.56	57.83	96.08
[Guleria et al. 2022]	RF	99.30	-	99.30	-
	NB	95.30	94.60	95.30	94.50
	DT	99.60	-	99.60	-
	MC	95.40	95.10	95.40	94.30
[Almahshi et al. 2022]	SVM	75.10	-	-	-
	NB	96.70	-	-	-
	DT	97.60	-	-	-
	Ensemble	97.30	-	-	-
[Stroek et al. 2023]	SVM	92.53	-	-	-
	RF	91.20	-	-	-
	NB	90.67	-	-	-
	DT	90.13	-	-	-
	LR	91.73	-	-	-
	MLP	96.40	-	-	-
[Sankar et al. 2022]	KNN	96.87	-	-	-
	DT	87.50	-	-	-
	LR	81.25	-	-	-
	XGB	98.59	-	-	-

All the works mentioned, with the exception of [Stroek et al. 2023], used the UCI (University of California Irvine) repository in their research. From previous works, the ML algorithms showed potential in assisting with the diagnosis of patients with hypothyroidism. However, we need to analyze the features to understand which parameter values have the most significant influence on the model's predictions. Moreover, we need to explain the predictions made by the models. In this way, we applied the SHAP and LIME techniques to analyze the contribution of each selected feature in the model's output.

3. Dataset

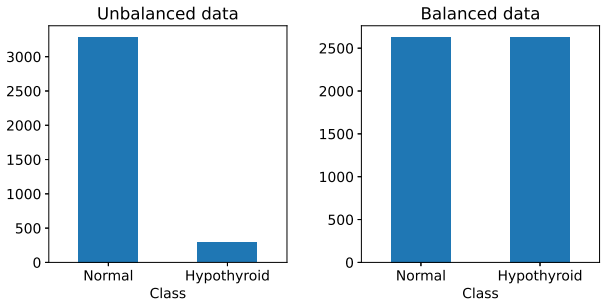
We utilized the Thyroid disease dataset, which was obtained from the publicly available ML repository of the UCI [Quinlan 1987]. We worked with the hypothyroid data. This dataset encompasses comprehensive information for each patient, including 29 attributes, such as age, sex, medication history, pregnancy status, surgical history, and results from thyroid function tests. After downloading the dataset, we needed to guarantee the data consistency then we proceeded with the data cleaning process.

3.1. Data cleaning

In the data science field, not all data is straightforward to use. Therefore, we need to prepare it through the data cleaning process, which is designed to get rid of any unnecessary or messy information from the data. This step is an important part of ML workflows because it helps fix errors in the dataset that could affect the accuracy of the final ML model [Van Der Aalst and van der Aalst 2016]. In data cleaning process, we first formatted the data, adding column headers. After formatting, we discarded invalid characters or values. Thereafter, we treated the missing data with imputation techniques commonly used in data science, such as replacing the missing data with the average of the existing data in the column. In other cases, we applied the mode, for example, in the sex parameter.

3.2. Data balancing

Generally, when we work with classification problems, the data must be balanced to be fed into the ML algorithm. We implemented the Synthetic Minority Over-sampling Technique (SMOTE) to balance the dataset. This technique generates extra data from the minority class to overcome data imbalance [Chawla et al. 2002]. The new data generation is based on the K nearest neighbors algorithm (K Nearest Neighbors - KNN). Fig. 2 (a) shows the unbalanced data with the minority class as hypothyroidism, and the majority as data from healthy people. After the synthetic data generation with the SMOTE technique, we achieve the balanced data, as shown in Fig. 2 (b).

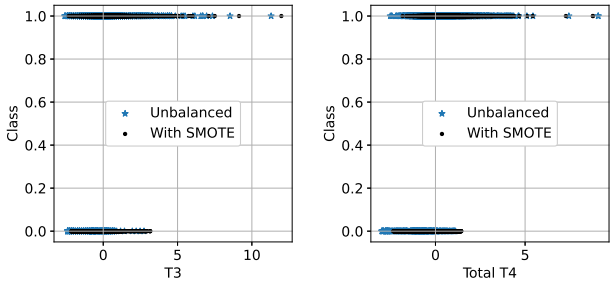


(a) (b)

Figure 2. Database: (a) Before SMOTE and (b) after SMOTE.

From the original dataset, only 11% of the data corresponds to data of patients with hypothyroidism, therefore the oversampling can increase the overlapping of classes and can introduce additional noise. However, we analyzed the scatter plot of the class distribution before and after SMOTE, as shown in Fig. 3 (a) and (b), the synthetic data

balanced the class distributions and induced the classifier to create more considerable decision regions helping the classifier generalize better. After the formatting, cleaning, and balancing process, the data is ready to be used.



(a) (b)

Figure 3. Synthetic data: (a) T3 values before SMOTE and (b) with SMOTE and Total T4 values.

4. Features selection

In this section, we discussed the methods used to select meaningful features, including the Recursive Feature Elimination (RFE) method, Pearson Correlation, and also the selection based on K-means clustering.

4.1. Pearson correlation coefficient

Pearson Correlation estimates the statistical relationship between two variables and is calculated using the correlation coefficient. This correlation must fall within the range of -1, which means a strongly negative correlation, to 1, which means a strongly positive correlation [Cohen et al. 2009, Saidi et al. 2019]. We used Pearson correlation to identify which features have a strong linear relationship with the target variable and, therefore, infer which one may have a significant impact on the ML model. The observed features are depicted in Table 2.

4.2. Recursive Feature Elimination

RFE is a technique that determines a subset of features within a dataset for feature selection. The technique gradually eliminates the features through a ML model, until the minimum number of attributes to be included is reached [Darst et al. 2018]. In RFE implementation, we selected the random forest model to select the most important features from a dataset. Random is selected because it is capable of raking the importance of features. We set the number of features to eliminate as 2, 3, 5, 8, 10, and 12. Finally, we selected 10 as the number of features because it demonstrates better model performers. Table 2 depicts the 10 features selected with the RFE method.

4.3. K-means clustering

K-means clustering is an unsupervised learning algorithm that groups data into clusters. Each cluster is composed of similar instances, i.e., it seeks to find patterns and structures in the data without the need for pre-defined labels or categories

[Arora et al. 2022]. We varied the cluster's number between 3, 5, 8, 10, and 12, with the number of features varied between 2, 3, 5, 8, 10 and 12. After clustering, we calculated the cluster centroids for each cluster. We used the proximity of each feature to the cluster centroids as a measure of feature importance.

Table 2. Features with the highest frequency in each method.

Method	Features selected									
RFE	TSH	FTI	T3	TT4	Age	T4U	Sex	OT	RS	QH
Correlation	TSHm	TSH	T3m	T3	TT4m	TT4	T4Um	T4U	FTIm	FTI
Clustering	TT4m	T4Um	TT4	T3m	FTI	T3	Pregnant	I131	Psych	Sick
Proposed	TT4	TT4m	T4Um	T3m	FTI	T3	TSH	T4U	Pregnant	I131

The suffix *m* in each feature means measured, and RS is Referral source, OT means on thyroxine, and QH query hypothyroid.

4.4. Feature selected results

We implemented an algorithm to select the features with the highest frequency among the feature selection method's (RFE, Pearson, K-means) outputs. The output of our algorithm is shown in the Table 2 as proposed. The feature T4U refers to total T4, TT4 is the level of thyroxine in the patient's body, and I131 indicates if the patient uses Iodo-131 or not.

5. Machine learning models for classification

In this section, we introduced various ML models, including random forest, XGBoost, LightGBM, extra trees, gradient boosting, decision tree, and a proposed approach based on stacking ensemble model.

5.1. Random Forest

Random Forest is an ML method used for classification or regression [Fawagreh et al. 2014]. The algorithm consists of an arrangement of tree classifiers, generating a forest, where each of the classifiers is generated using a random input vector, and each generated tree casts a vote for the most popular class to classify the input vector [Pal 2005]. To implement the classifier, we set 12 decision trees, each with a maximum depth limited to 3 levels. Furthermore, we defined the maximum number of variables considered in each tree node as log2. We used sampling criteria with 2 minimum samples for the division of nodes and 1 minimum sample for the formation of leaf nodes.

5.2. Extreme Gradient Boosting

XGBoost, introduced by [Chen and Guestrin 2016], is an model based on the gradient boosting algorithm. This method utilizes gradients to train decision trees within the ensemble, meaning it leverages the gradient of the loss function to adjust tree parameters. As a result, the algorithm offers improved speed and scalability.

We implemented the XGBoost with a learning rate of 0.04, which regulates the step size during training, and a value of 0.5 for the penalty parameter, controlling additional divisions in the trees. Furthermore, we established 200 trees, with a maximum depth of 9 levels and a minimum weight of 5 samples per node. These parameters made it possible to adapt the XGBoost model to the specific needs of the problem, resulting in an effective and accurate model.

5.3. Light Gradient Boosting Machine

LightGBM is a grid learning model based on decision trees algorithm. The method uses algorithms based on histograms to speed up the training process, which results in less memory usage and a better growth strategy on each sheet with depth restrictions [Fan et al. 2019]. We implemented the algorithm using 100 trees, with a maximum depth limited to 5 levels and 15 leaf nodes in each tree.

5.4. Extra Trees Classifier

Extra Trees model is a variation of Random Forest, which introduces more randomness during the construction of the decision tree. When creating decision trees, the classifier randomly selects subsets of smaller features for each of the node divisions, making the model less sensitive to the occurrence of overfitting and enabling improved performance in more robust data sets [Geurts et al. 2006]. We used 50 trees with a maximum depth of 30 levels, and a minimum number of 4 samples to form a leaf node in the algorithm.

5.5. Gradient Boosting Classifier

Gradient Boosting algorithm has the ability to build high-precision classification models. It works by building decision trees sequentially, where each new tree aims to correct the errors made by the previous one [Friedman 2001]. We used 20 trees in the algorithm, with a maximum depth of 5 levels. Also, we set a learning rate of 0.02 to control the step size during model training, and the number of features considered in each split was set as the square root of the total available features.

5.6. Decision Tree

Decision Tree is an algorithm based on hierarchical tree structures in which each node represents a decision or test on a feature, each branch represents the result of one of the tests, and each leaf of the tree represents one of the classes or an output value [Quinlan 1986]. We implemented the algorithm using the entropy function to measure the impurity of the node divisions to improve the quality of the divisions. We used the square root function to split the features at each node, and a minimum number of 24 samples to form a leaf node.

5.7. The proposed approach

We proposed a stacking model based on GB, ET, DT, and RF algorithms to improve the predictive capacity. Fig. 4 shows the hierarchical approach. We chose these models because they presented the best metrics. The GB, ET, and DT models perform the primary classifications based on the available input data. Then, the RF model receives the predictions from base models and performs the final classification. The final prediction is introduced into the SHAP and LIME methods for explanation of the results.

6. Discussions and Results

6.1. Models evaluate

We evaluated the models using metrics like the confusion matrix, accuracy, recall, precision, and F1-score. Table 3 shows the model's metrics. To improve the models, we

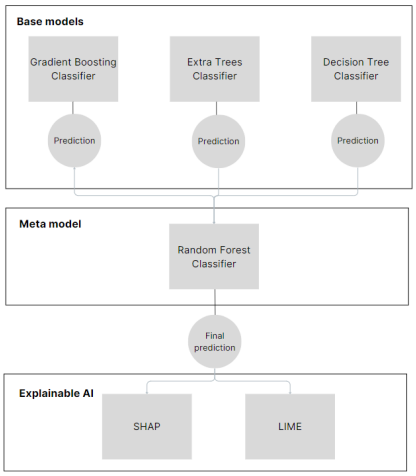


Figure 4. Proposed scheme with the stacking model and explainable AI.

used a grid search method along with cross-validation with 5 K-fold to make sure that we reduce overfitting. According to Table 3, the proposed model (stacking) achieved an accuracy of 0.9916 and an AUC of 0.9973. It showed high precision and F1-score values, demonstrating its effectiveness in classification.

Table 3. Model evaluation metrics.

Model	Accuracy	AUC	Recall	Precision	F1
RF	0.9908	0.9951	0.9848	**0.9969**	0.9908
XGBoost	0.9840	0.9960	0.9817	0.9862	0.9840
LightGBM	0.9460	0.9459	**0.9878**	0.9116	0.9482
ET	0.9763	0.9960	0.9605	0.9918	0.9759
GB	0.9901	0.9963	0.9832	0.9969	0.9900
DT	0.9809	0.9861	0.9772	0.9846	0.9809
Stacking Model	**0.9916**	**0.9973**	0.9863	**0.9969**	**0.9915**

Fig. 5 (a), (b), (c), (d), (e), (f), and (g) depicted the confusion matrices for each model. As shown in Fig. 5 (g), the stacking model classified 655 patients as healthy and 649 as sick. However, it made an error by classifying 2 healthy patients as sick and 9 sick patients as healthy. But, the proposed model demonstrated the best overall performance among the classifiers.

6.2. Model explanation

6.2.1. SHAP

Fig. 6 shows the distribution of SHAP values. The TSH values have a stronger positive impact on the prediction. Also, the features FTI, TT4, T3, T3 measured, and T4U are crucial for the model predicting if the person has hypothyroidism. The distribution of points provides insight into how the chosen feature affects model predictions across the dataset.

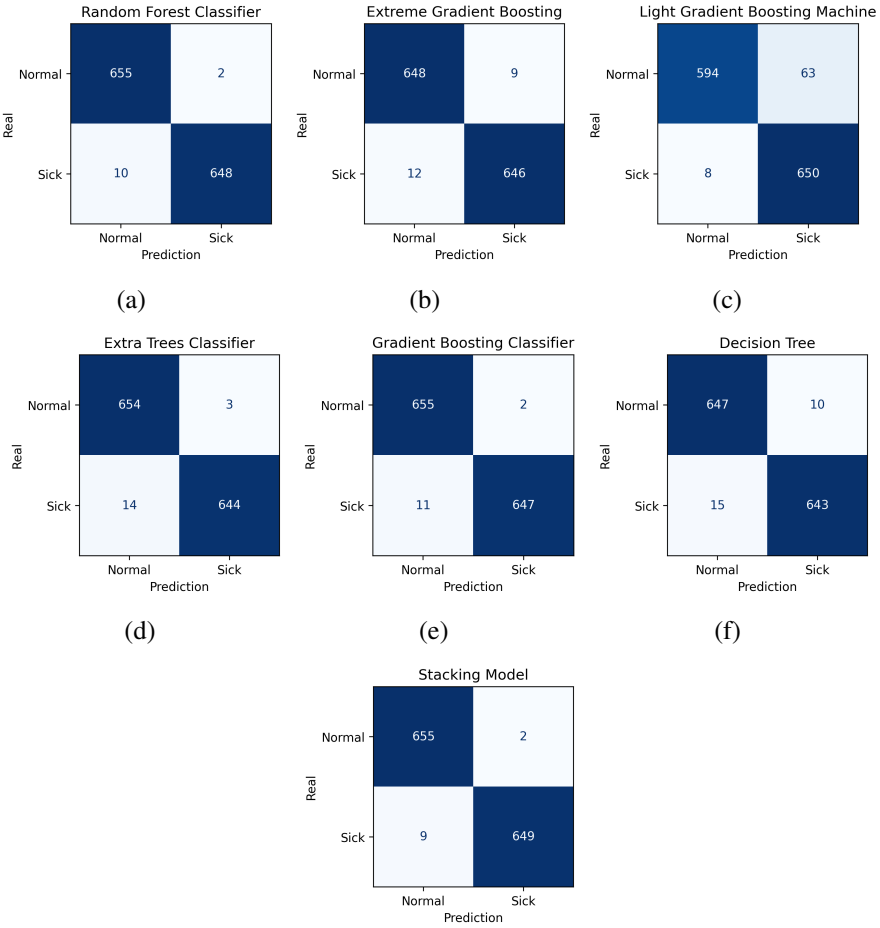


Figure 5. Confusion matrix: (a) RF, (b) XGBoost, (c) LightGBM, (d) ET, (e) GBC, (f) DT, and (g) Stacking model.

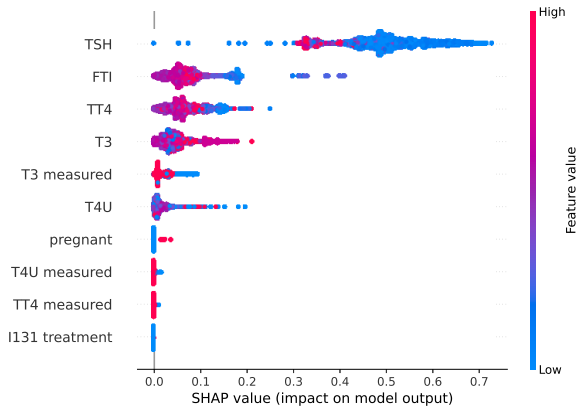
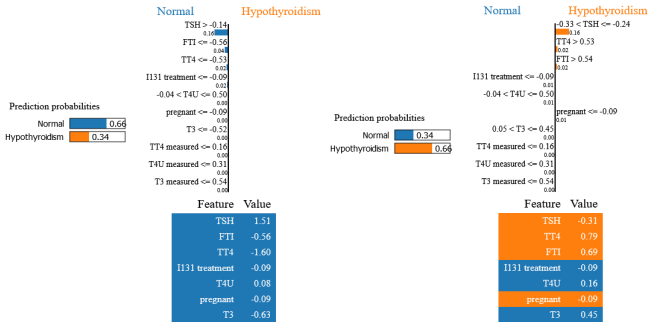


Figure 6. Features impact on the prediction.

6.2.2. LIME

Fig. 7 shows how the model provides the predictions by application of the LIME method. Fig. 7 (a) depicts which features are relevant for the model to classify the patient

as normal with a probability of 66%. Fig. 7 (b) shows how the model performs the prediction for hypothyroidism with a probability of 66%. The features determinant for deciding if this patient has hypothyroidism are TSH, TT4, FTI, and knowing if the patient is pregnant. Differently, for the model to decide if the patient has no hypothyroidism, the features most influential are TSH, FTI, TT4, I131, T4U, T3, and know if the patient is pregnant.



(a) (b)

Figure 7. Features that impacted in the predicted value and their respective meanings: (a) Normal and (b) Hypothyroidism.

These two methods offer different approaches to explaining ML model predictions, highlighting the importance of model interpretability in diagnostic contexts. SHAP shows the distribution of feature values, revealing which ones have a more significant impact on the model. On the other hand, the LIME method enables a local interpretation of the model predictions.

7. Conclusions

ML algorithms hold significant promise in the identification of hypothyroidism, offering the benefits of the discovery of patterns and trends within clinical data. These insights are instrumental in improving the diagnosis and treatment of hypothyroidism, eventually improving patient care and outcomes. From all models evaluated, the proposed stacking ensemble model demonstrated the highest classification accuracy, reaching 99.16%. This research emphasizes the potential of combining medical data with ML algorithms to improve diagnostic processes and potentially enhance the overall well-being of patients. Moreover, for a more transparent, interpretable, and understandable model to doctors, we applied two explainable AI methods: SHAP and LIME. SHAP method provide a global insights into the model's overall behavior, and LIME provides a local explanations for specific predictions. Fundamentally, this research demonstrates how the association between medical data and ML can pave the way for advancement in healthcare.

Acknowledgment

To UFERSA for financial support in granting a Scientific Initiation scholarship and UFERSA/PROPPG 65/2022 (PAPC) support for research groups.

References

Almahshi, H. M., Almasri, E. A., Alquran, H., Mustafa, W. A., and Alkhayyat, A. (2022). Hypothyroidism prediction and detection using machine learning. In *2022 5th International Conference on Engineering Technology and its Applications (IICETA)*, pages 159–163. IEEE.

Arora, N., Singh, A., Al-Dabagh, M. Z. N., and Maitra, S. K. (2022). A novel architecture for diabetes patients' prediction using k-means clustering and svm. *Mathematical Problems in Engineering*, 2022.

Bensenor, I. M., Olmos, R. D., and Lotufo, P. A. (2012). Hypothyroidism in the elderly: diagnosis and management. *Clinical Interventions in Aging*, pages 97–111.

Cavalcante, C. M., Almeida, V. A., Barros, M., Lima, N., and Rego, R. C. (2023). Thyroid syndrome detection using machine learning algorithms: A comparative analysis. In *XVI Brazilian Conference on Computational Intelligence (CBIC 2023)*.

Chaganti, R., Rustam, F., De La Torre Díez, I., Mazón, J. L. V., Rodríguez, C. L., and Ashraf, I. (2022). Thyroid disease prediction using selective features and machine learning techniques. *Cancers*, 14(16):3914.

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

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

Cohen, I., Huang, Y., Chen, J., Benesty, J., Benesty, J., Chen, J., Huang, Y., and Cohen, I. (2009). Pearson correlation coefficient. *Noise reduction in speech processing*, pages 1–4.

Darst, B. F., Malecki, K. C., and Engelman, C. D. (2018). Using recursive feature elimination in random forest to account for correlated variables in high dimensional data. *BMC genetics*, 19(1):1–6.

Duan, L., Zhang, H.-Y., Lv, M., Zhang, H., Chen, Y., Wang, T., Li, Y., Wu, Y., Li, J., and Li, K. (2022). Machine learning identifies baseline clinical features that predict early hypothyroidism in patients with graves' disease after radioiodine therapy. *Endocrine Connections*, 11(5).

Fan, J., Ma, X., Wu, L., Zhang, F., Yu, X., and Zeng, W. (2019). Light gradient boosting machine: An efficient soft computing model for estimating daily reference evapotranspiration with local and external meteorological data. *Agricultural water management*, 225:105758.

Fawagreh, K., Gaber, M. M., and Elyan, E. (2014). Random forests: from early developments to recent advancements. *Systems Science & Control Engineering: An Open Access Journal*, 2(1):602–609.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Gaitonde, D. Y., Rowley, K. D., and Sweeney, L. B. (2012). Hypothyroidism: an update. *South African Family Practice*, 54(5):384–390.

Geurts, P., Ernst, D., and Wehenkel, L. (2006). Extremely randomized trees. *Machine learning*, 63(1):3–42.

Guleria, K., Sharma, S., Kumar, S., and Tiwari, S. (2022). Early prediction of hypothyroidism and multiclass classification using predictive machine learning and deep learning. *Measurement: Sensors*, 24:100482.

Hu, M., Asami, C., Iwakura, H., Nakajima, Y., Sema, R., Kikuchi, T., Miyata, T., Sakamaki, K., Kudo, T., Yamada, M., et al. (2022). Development and preliminary validation of a machine learning system for thyroid dysfunction diagnosis based on routine laboratory tests. *Communications Medicine*, 2(1):9.

Hueston, W. J. (2001). Treatment of hypothyroidism. *American family physician*, 64(10):1717–1725.

Kostoglou-Athanassiou, I. and Ntalles, K. (2010). Hypothyroidism-new aspects of an old disease. *Hippokratia*, 14(2):82.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Pal, M. (2005). Random forest classifier for remote sensing classification. *International journal of remote sensing*, 26(1):217–222.

Quinlan, J. R. (1986). Induction of decision trees. In *Machine learning*, volume 1, pages 81–106. Kluwer Academic Publishers.

Quinlan, R. (1987). Thyroid Disease. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5D010.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Saidi, R., Bouaguel, W., and Essoussi, N. (2019). Hybrid feature selection method based on the genetic algorithm and pearson correlation coefficient. *Machine learning paradigms: theory and application*, pages 3–24.

Sankar, S., Potti, A., Chandrika, G. N., and Ramasubbareddy, S. (2022). Thyroid disease prediction using xgboost algorithms. *J. Mob. Multimed*, 18:1–18.

Shahid, A. H., Singh, M. P., Raj, R. K., Suman, R., Jawaid, D., and Alam, M. (2019). A study on label tsh, t3, t4u, tt4, fti in hyperthyroidism and hypothyroidism using machine learning techniques. In *2019 International Conference on Communication and Electronics Systems (ICCES)*, pages 930–933. IEEE.

Stroek, K., Visser, A., van der Ploeg, C. P., Zwaveling-Soonawala, N., Heijboer, A. C., Bosch, A. M., van Trotsenburg, A. P., Boelen, A., Hoogendoorn, M., and de Jonge, R. (2023). Machine learning to improve false-positive results in the dutch newborn screening for congenital hypothyroidism. *Clinical Biochemistry*, 116:7–10.

Vaidya, B. and Pearce, S. H. (2008). Management of hypothyroidism in adults. *Bmj*, 337.

Van Der Aalst, W. and van der Aalst, W. (2016). *Data science in action*. Springer.