Accelerating Logical Analysis of Data Using an Ensemble-Based Technique

Osama Elfar, Soumaya Yacout, and Hany Osman

Abstract— **Logical Analysis of Data (LAD) is a well-known classification technique that generates interpretable patterns with competitive accuracy. The challenge encountered in applying LAD comes from its long computational time, which makes it unsuitable for handling a large volume of data. In this paper, we propose a novel mechanism for developing an ensemble system for LAD (LAD-ENS) to improve its computational efficiency, while preserving its interpretability and promising accuracy. This new mechanism aims to maintain the explanatory power of classical LAD by combining the individual classifiers at the level of patterns. The developed ensemble system enables LAD to be run in parallel computing environments. Using datasets obtained from the UCI Machine Learning Repository, computational experiments are conducted to demonstrate the performance of LAD-ENS in terms of computational time, classification accuracy, and interpretability. Furthermore, we introduce the concept of the comprehensibility index in order to study the change in the explanatory power of LAD. In addition to achieving a statistically significant reduction in computational time, the developed LAD-ENS achieves competitive classification accuracies compared to two classical LAD approaches and five common machine learning algorithms.**

Index Terms— **Computational efficiency, ensemble system, parallel computation, Logical Analysis of Data**

I. Introduction

As an Artificial Intelligence (AI) application, a machine learning (ML) algorithm enhances the decision-making capabilities of various manufacturing and business systems. Therefore, achieving good performance of a ML algorithm is essential in ensuring efficient operation throughout these systems. Nevertheless, this goal becomes difficult to attain when dealing with a large volume of data, especially with the emergence of the Internet of Things (IoT) and Industry 4.0. IoT is the network of devices, buildings, machines, products, and other objects that are connected with sensors, software, and network connectivity, allowing these things to gather and interchange data [1]. Industry 4.0 aims to utilize advanced technologies in connectivity to gather all of the important data in manufacturing processes and products to develop analytical models through the means of ML techniques. One of the most important objectives of these models is predicting manufacturing performance and providing feedback so that corrective decisions can be made during the manufacturing processes. Therefore, the ML techniques that will be used in this context should have good explanatory power and interpretable results in order to provide root-cause analysis of certain phenomena to facilitate the decision-making process. However, well-known ML techniques, such as ensemble decision trees, support vector machines and neural network [2]-[4], show high accuracy in the literature, but do not have enough explanatory power and interpretable results. On the other hand, Logical Analysis of Data (LAD) is a classification approach that generates patterns containing structural knowledge that explain the hidden phenomena under study. As such, the patterns generated are the most useful results that indicate the candidate root causes behind the observed physical phenomena, and consequently, the best way to respond to them [5]-[7]. Additionally, LAD is used to develop regression models [8]. Because of these abilities, LAD is used in various fields such as medical, services, business, and manufacturing [7], [9]-[17]. However, a relatively long computational time makes LAD unsuitable for manipulating data with large volumes.

Some techniques have been presented in the literature to implement LAD with the aim of enhancing computational efficiency. A polynomial algorithm was used in [18] to enumerate all LAD patterns with a selected degree to limit the number of features in the generated patterns. The degree of a pattern is the number of features it is constructed with. However, this technique still generates a high number of patterns, which is computationally expensive to handle when there are datasets with large amounts of features. In [19], instead of generating all possible patterns with specific characteristics, the column generation technique is used. In this framework, the master problem has the objective of building a LAD model with a maximum separation margin between the classes by generating patterns to enlarge this margin in subproblems. This column generation framework generates only one pattern in each iteration, which affects its computational efficiency. Therefore, a multi-pattern generation framework is developed to improve the efficiency of LAD column generation models by generating more than one pattern in each iteration [20]. However, none of these techniques is aimed at achieving a scalable LAD that handles a large volume of data.

With a continuous increase in the volume of data, the need to develop scalable ML algorithms that can handle large datasets has attracted researchers [21]. A taxonomy was proposed in [22] for ML techniques that can handle a large volume of data by improving the computational efficiency

Manuscript received April 26, 2021; revised September 10, 2021.
O. Elfar is a PhD candidate in the Mathematical and Industrial Engineering Department, École Polytechnique de Montréal, Montréal, H3C 3A7 Canada (e-mail: osama.elfar@ polymtl.ca), and a Research and Teaching Assistant in the Mechanical Design and Production Engineering Department, Faculty of Engineering, Cairo University, Egypt.
S. Yacout is a Professor in the Mathematical and Industrial Engineering Department, École Polytechnique de Montréal, Montréal, H3C 3A7 Canada (e-mail: soumaya.yacout@polymtl.ca).
H. Osman is an Assistant Professor in the Systems Engineering Department, King Fahd University of Petroleum & Minerals, Dhahran, 31261 Kingdom of Saudi Arabia (e-mail: hanyosman@kfupm.edu.sa).

through either designing more efficient algorithms or relying on parallelism. The parallelism-based category reflects most of the state of the art in scalable ML algorithms [23]. Specifically, parallelized methods that make ML algorithms more scalable are classified into two sub-categories: (i) parallelized model/parameters: developing parallelized versions of learning algorithms by first dividing the learning model/parameters and then performing computations on each division concurrently, and (ii) parallelized data: partitioning input data vertically, horizontally, or even arbitrarily into manageable pieces, and then computing all data subsets simultaneously [22], [24]. The parallelized-data techniques are mainly ensemble based, such as random forests [25] and XGBoost [26]. In this paper, we introduce a novel technique that belongs to this sub-category: an ensemble LAD (LAD-ENS).

While building an ensemble LAD may seem like an intuitive way to improve both accuracy and computational time, the challenge is actually to build an ensemble system that preserves the explanatory power of LAD. Such explanatory power can guide the decision maker on keeping a current process under control, or directing that process to reach its maximum yield. For example, specifying the cutting conditions at which a machining process provides the desired surface roughness and the required output. The objective of the proposed technique is establishing a mechanism to combine the knowledge of individual LAD classifiers while preserving such explanatory power. In order to achieve this, LAD-ENS deals with the individual classifiers at the level of patterns, and the explanatory power is based on these patterns. Different ensemble systems in other ML techniques use voting mechanisms that significantly decrease the interpretability of the results, such as Random Forest and XGboost. Since the explanatory power of LAD is affected by the total number of generated patterns and their average degree [18], an index is introduced in this paper as a measure of interpretability.

The remainder of this paper is organized as follows: Section II provides essential background on the LAD classification technique. Section III introduces the LAD ensemble system (LAD-ENS). Computational experiments and a comparison with classical LAD and other machine learning algorithms are conducted in Section IV. Section V concludes the work and discusses the future directions.

II. Logical Analysis of Data

LAD, originated in [27], is a classification technique that is characterized by extracting interpretable patterns from two–class or multi-class datasets [28]. The generated patterns are utilized as decision rules, used to classify unlabeled data into distinct classes [29]. LAD can also be used to develop regression models [8]. However, in this paper we only address the classification models. The LAD classification technique consists of three main steps: data binarization, pattern generation, and theory formation, as shown in Fig. 1.

The data binarization step converts numerical and nominal data to binary data. Pattern generation is an essential step that extracts structural information in the form of patterns that characterize each different class in the binarized dataset. Many approaches are used for pattern generation, mostly based on enumeration, heuristics, or mixed integer linear programming (MILP) algorithms. In this research, we use cbmLAD software [30], in which patterns are generated by using the ant colony optimization technique. Theory formation is the final step that uses the generated patterns to create a discriminant function that is used as a classifier for new data [6].

In the case of a two-class dataset, the training set is $\Omega = \Omega^+ + \Omega^-$, which is formed from positive and negative subsets with n features. After the binarization and pattern generation steps, LAD forms the pattern set $\Pi = \{P_1, \dots, P_r\}$, where r is the number of generated patterns from the training set Ω. The pattern set has positive and negative patterns. Each P_i is a conjunction of d features, where $d \le n$ is the pattern degree. Each P_i covers at least one observation from one of the subsets Ω^+ (or Ω^-) and no observations from the other set Ω^- (or Ω^+). Each P_i has characteristics that have been formed by the observations it covers. These characteristics are illustrated as follows:

1) C_i is the class (the sign positive or negative) which is assigned to pattern P_i,
2) Q_i is the number of observations covered by pattern P_i,
3) δ_i is the degree which is the number of features constructing the pattern P_i,
4) π_i is the prevalence, i.e., $\pi_i = Q_i/|\Omega^+|$ in case of positive C_i, or $\pi_i = Q_i/|\Omega^-|$ in case of negative C_i,
5) ω_i is the weight of the pattern P_i, i.e., $\omega_i = Q_i \,/\, \sum_{\substack{j=1: \\ C_j = C_i}}^{r} Q_j$.

In the theory formation step, LAD uses the weights of patterns to formulate the discriminant function as follows:

$$f(x) = \sum_{i=1}^{r} \omega_i y_i(x) \tag{1}$$

, where

$$y_i(x) = \begin{cases} 1 & \text{if } C_i \text{ is} +ve \text{ AND } P_i \text{ covers } x \\ -1 & \text{if } C_i \text{ is} -ve \text{ AND } P_i \text{ covers } x \\ 0 & \text{if } P_i \text{ doesn't cover } x \end{cases} \tag{2}$$

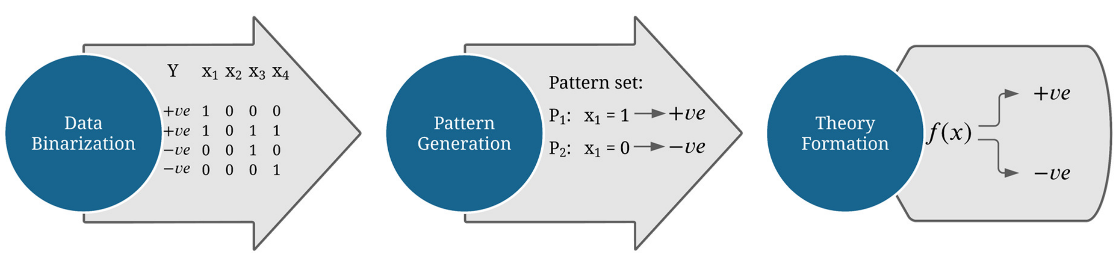

Fig. 1. Framework of LAD.

It is obvious that $f(x) = [-1,1]$. The new observation x is classified as a positive observation if $f(x) > 0$, negative observation if $f(x) < 0$, and unclassified if $f(x) = 0$.

In order to use LAD in multi-class classification problems, a decomposition approach is used to divide the multi-class problem into many two-class problems. This approach is applied in two different methods: One-Versus-All (OvA) and One-Versus-One (OvO) [31]. The OvA method divides the multi-class problem into k different two–class classification problems, where k is the number of classes. Each problem considers one class $i \in [1:k]$ as the positive class and all the remaining $(k-1)$ classes as the negative class. The OvO method divides the multi-class problem into $\binom{k}{2}$ two-class problems by considering each possible class pair as an individual two-class classification problem. Each problem considers class $i \in [1:k]$ as a positive class and $j \in [1:k]$ as a negative class, where $i \neq j$. In this research, we use the multi-class OvO method for handling multi-class datasets in our computational experiments.

III. Ensemble Logical Analysis of Data

Building an ensemble system is based on three pillars: (i) sampling the training data for individual basic classifiers, (ii) the training procedures of individual classifiers, (iii) the combination mechanism for merging individual classifiers and obtaining an ensemble model [32].

In this work, we use stratified sampling without a replacement to generate m different data subsets for training individual LAD classifiers, as shown in Fig. 2. In the case of a two-class dataset, the training set $\Omega = \Omega^+ + \Omega^-$, with n features, is partitioned into m subsets, indexed by i. Each subset $\Omega_i = \Omega_i^+ + \Omega_i^-$, has the same n features of Ω. LAD is applied on each data subset in a parallel manner to generate m individual LAD classifiers. Each classifier i has an independent pattern set, $\Pi_i = \{P_{i1}, \ldots, P_{ir_i}\}$, where r_i is the number of patterns generated from applying LAD on Ω_i. Each P_{ij}, where $j = 1, \ldots, r_i$, is a positive (or negative) pattern that covers at least one observation from the set Ω_i^+ (or Ω_i^-) and does not cover any observation from the other set Ω_i^- (or Ω_i^+).

The characteristics of any pattern P_{lk}, $l \in [1:m]$ and $k \in [1:r_l]$, are based on the data subset Ω_l, and are as follows:

1) C_{lk} is the class positive (or negative) of pattern k generated from data subset l.
2) Q_{lk} is the number of observations from Ω_l covered by pattern P_{lk},
3) ω_{lk} is the weight of the pattern P_{lk}, i.e., $\omega_{lk} = Q_{lk} / \sum_{\substack{j=1: \\ C_{lj}=C_{lk}}}^{r_l} Q_{lj}$.

In order to build a combination mechanism that merges the knowledge of individual classifiers and preserves the explanatory power of LAD, we introduced a mechanism at the level of patterns before formulating the discriminant function of LAD. This mechanism updates the weight of each pattern to take into consideration the other patterns from other individual classifiers. The weight ω_{lk} of pattern P_{lk} is adjusted to a combined weight ω_{lk}^c, which is the ratio of the coverage of P_{lk} to the total coverage of all classifiers' patterns with the same class as P_{lk}.

$$\omega_{lk}^c = \frac{Q_{lk}}{\sum_{i=1}^{m} \sum_{\substack{j=1: \\ C_{ij}=C_{lk}}}^{r_i} Q_{ij}} \tag{3}$$

After updating the weights of the patterns, the mechanism formulates the LAD-ENS discriminant function using all patterns from all individual classifiers in a single function as follows:

$$f_{ensemble}(x) = \sum_{i=1}^{m} \sum_{j=1}^{r_i} \omega_{ij}^c y_{ij}(x) \tag{4}$$

, where

$$y_{ij}(x) = \begin{cases} 1 & if\ C_{ij}\ is + ve\ AND\ P_{ij}\ covers\ x \\ -1 & if\ C_{ij}\ is - ve\ AND\ P_{ij}\ covers\ x \\ 0 & if\ P_{ij}\ doesn't\ covers\ x \end{cases} \tag{5}$$

Even though the voting mechanism provides good accuracy and results in the literature with other ML techniques, such as Random Forest and XGboost, this proposed mechanism preserves the explanatory power of LAD by using the patterns to directly predict the class instead of getting a vote from each individual classifier, which

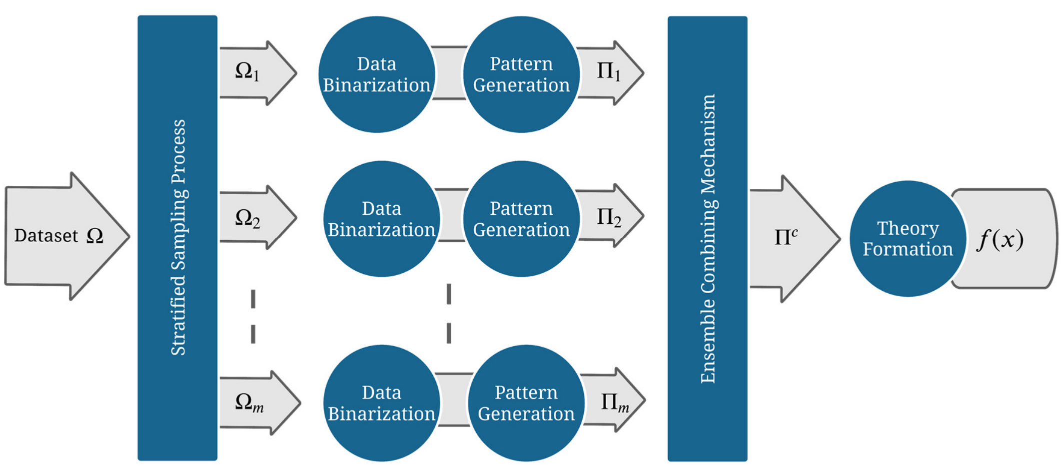

Fig. 2. Framework of LAD-ENS.

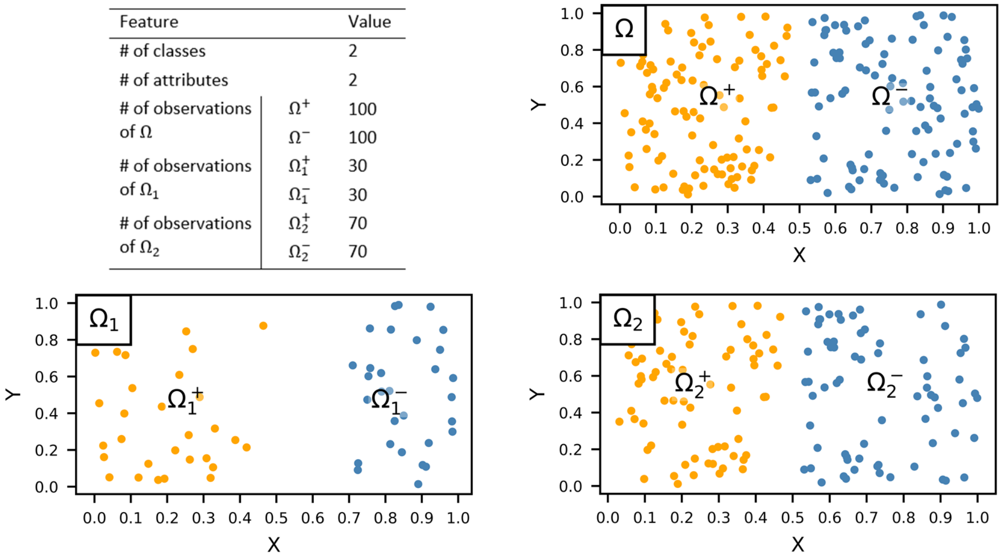

Fig. 3. The generated dataset Ω, and two data subsets Ω_1 and Ω_2.

significantly reduces the interpretability.

To illustrate how a LAD-ENS model is developed compared to classical LAD, we consider, for example, a two–class dataset $\Omega = \Omega^+ + \Omega^-$ with two features (X and Y). The data is partitioned by using a stratified sampling approach without replacement into two data subsets $\Omega_1 = \Omega_1^+ + \Omega_1^-$ and $\Omega_2 = \Omega_2^+ + \Omega_2^-$. The dataset Ω, and the two data subsets Ω_1 and Ω_2 are illustrated in Fig. 3. Running classical LAD on Ω generates a pattern set Π, as shown in Fig. 4.c. Each data subset Ω_1 and Ω_2 was processed by LAD separately, providing two basic pattern sets $\Pi_1 = \{P_{11}, P_{12}\}$, and $\Pi_2 = \{P_{21}, P_{22}\}$, from Ω_1 and Ω_2, respectively, as shown in Fig. 4.a and Fig. 4.b. The characteristics of the patterns are determined and illustrated in Table 1.

In order to prepare these two sets of patterns Π_1 and Π_2 for a LAD-ENS discriminant function, the weights of the patterns are adjusted according to (3). For example, ω_{11}=1 is adjusted to ω_{11}^c=0.3. Other patterns' weights are illustrated in Table 1. In this table, the patterns of Π_2 have relatively higher combined weights than Π_1 because they are generated using a relatively larger sized data subset.

$$\omega_{11}^{c} = \frac{Q_{11}}{\sum_{i=1}^{2} \sum_{\substack{j=1: \\ C_{ij}=C_{11}}}^{r_i} Q_{ij}} = \frac{Q_{11}}{Q_{11}+Q_{21}} = \frac{30}{30+70} = 0.3$$

Afterwards, (4) and (5) will be used to formulate the ensemble discriminant functions $f_{ensemble}$ to classify new unlabeled observations.

IV. COMPUTATIONAL EXPERIMENTS

In this section, the computational performance and the classification accuracy of the developed LAD-ENS are demonstrated using twenty datasets obtained from the UCI machine learning repository. Table 2 shows the descriptions of these datasets. The k–fold cross validation approach is used with $k = 5$ to average the results obtained from five different training data subsets and five corresponding testing data subsets. As our main objective is to reduce the computational time of LAD and solve classification problems with huge datasets, we compare between our proposed LAD-ENS and the classical LAD models in terms of computational time and classification accuracy. Additionally, the qualities of the patterns are compared in terms of the total number of patterns and their degree and prevalence. Moreover, a comprehensibility index (CI) is introduced in order to study the effect of LAD-ENS on the explanatory power of the generated patterns. Additionally, to evaluate the competitivity of LAD-ENS compared to other classifiers, LAD-ENS is compared to five ML techniques in terms of the classification accuracy.

The mechanism of the developed LAD-ENS allows the pattern generation step to be performed in a parallel manner. In these computational experiments, we run LAD-ENS in parallel on Cedar cloud computing clusters provided by Compute Canada[1] using 20 cores. Each core has a 2.1 GHz CPU. Utilizing cloud-computing systems allows LAD to handle large volumes of data. The number of subsets in LAD-ENS is different for each dataset, as shown in the last column of Table 2. This number is chosen empirically based on the size and the separability of the classes of the datasets to generate patterns from each subset in a reasonable computational time. We increase the number of individual classifiers if the data size is big, or if the separability is low.

Table 3 provides a comparison between LAD-ENS and two classical LAD models: cbmLAD [30] and the multi-pattern generation framework of LAD (MPG-LAD) [20]. The results of MPG-LAD are gathered from [20]. The reduction in processing time by using LAD-ENS is more than 75% in most of the datasets compared to the cbmLAD model,

TABLE 1
CHARACTERISTICS OF THE PATTERNS IN Π_1 AND Π_2 SETS.

Pattern	Class	Q	ω	ω^c
P_{11}	Positive	30	1	0.3
P_{12}	Negative	30	1	0.3
P_{21}	Positive	70	1	0.7
P_{22}	Negative	70	1	0.7

[1] https://docs.computecanada.ca/wiki/Cedar

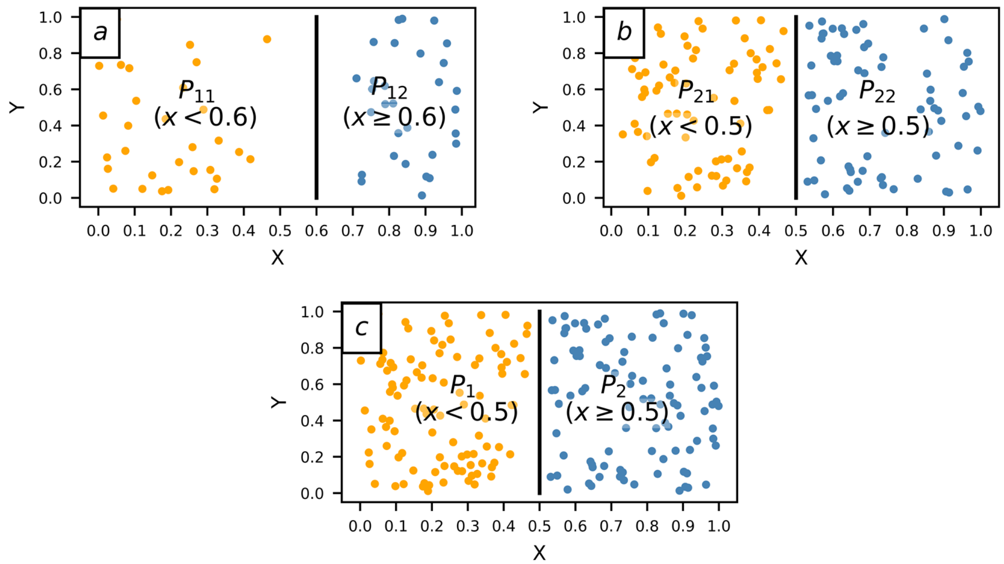

Fig. 4. a: pattern set Π_1, b: pattern set Π_2, and c: the patterns generated by classical LAD.

and more than 99% in all datasets compared to the MPG-LAD model. The n/a means that the classical LAD models were not able to solve the problems within 24 hours of running time. However, LAD-ENS was able to solve these problems in fewer than 10 minutes. Sampling datasets that have complex boundaries and low separability between the classes into different subsets enables LAD-ENS to handle them with fewer binary attributes and in less computational time. In terms of classification accuracy, LAD-ENS performs competitively for almost all datasets except wpbc, hrt-h, hrt-s, hrt-lb, and prks.

In order to statistically evaluate the computational time and accuracy of LAD-ENS, the Friedman test is used to compare the results of the three LAD models. The Friedman test is a non-parametric statistical test used to detect differences between the values of various populations' means [33]. The test is applied to the computational time and the accuracy for the three LAD models, cbmLAD, MPG-LAD and LAD-ENS, in two phases. We do not include the last 7 datasets in the test, since cbmLAD and MPG-LAD failed to handle them. The null hypothesis of phase 1 states that all of those models have the same means of computational time (or accuracy), as follows:

$$H_0: \ \mu_{cbmLAD} = \mu_{MPG-LAD} = \mu_{LAD-ENS}$$

The alternative hypothesis is formulated as follows:

$$H_a: \textit{Not all means are equal}$$

To reject or accept H_0, the test statistic F_r and the calculated significance level p are calculated as given in [33] and compared to the significant values; $F_{critical} = 6.0$ and $p_{critical} = 0.05$. Tables 4 and 5 show phase 1 of the Friedman tests on computational time and accuracy, respectively. For computational time, H_0 is rejected with a high value of $F_r = 26.0$ which means that there are very significant differences in terms of the computational time between the three LAD models. Meanwhile, H_0 is accepted in the test for accuracy, which indicates that the three LAD models are similar in

TABLE 2
DATASETS DESCRIPTION

ID	Name	# of Classes	# of observations	# of positive observations	# of negative observations	# of features	# of subsets
wbc	Wisconsin Breast Cancer	2	699	458	241	9	10
wpbc	Wisconsin Prognostic Breast Cancer	2	198	47	151	33	2
wdbc	Wisconsin Diagnostic Breast Cancer	2	569	212	357	30	4
hrt-c	Heart disease diagnosis - Cleveland	2	303	139	164	13	4
hrt-h	Heart disease diagnosis - Hungarian	2	294	106	188	10	4
hrt-s	Heart disease diagnosis – Switzerland	2	122	114	8	10	2
hrt-lb	Heart disease diagnosis - Long Beach VA	2	200	149	51	8	2
hpts	Hepatitis Domain	2	142	28	114	18	2
bld	BUPA liver disorders	2	325	200	125	6	2
pid	Pima Indians Diabetes	2	768	500	268	8	10
SPECTF	SPECTF Heart Data	2	267	212	55	44	2
SPECT	SPECT Heart Data	2	267	212	55	22	2
prks	Parkinsons Disease	2	195	147	48	22	2
SB	Spambase	2	4,601	2788	1813	56	10
WFR	Wall-Following Robot Navigation	4	5,456	--	--	24	10
LR	Letter Recognition	26	20,000	--	--	16	10
MG	MAGIC Gamma Telescope	2	19,020	12332	6688	10	65
MBP	MiniBooNE particle identification	2	130,065	93565	36499	50	500
SS	Skin Segmentation	2	245,057	194198	50859	3	90
CT[a]	Covertype	7	581,012	--	--	7	50

All datasets are available on the UCI machine learning repository: https://archive.ics.uci.edu/ml/index.php
[a] A sample of only 50,000 observations from Covertype dataset was used in the experiments to fit the available memory in compute Canada clusters.

TABLE 3
COMPARING THE PERFORMANCE OF LAD-ENS WITH CLASSICAL LAD MODELS

Dataset	cbmLAD		MPG-LAD		LAD-ENS			
	Accur (%)	Time (sec.)	Accur (%)	Time (sec.)	Accur (%)	Time (sec.)	Time reduction % wrt:	
							cbmLAD	MPG-LAD
wbc	**95.13**	6.07	95	2283	94.44	**0.98**	**83.86**	**99.96**
wpbc	68.86	21.42	**98**	534	60.56	**4.36**	**79.65**	**99.18**
wdbc	95.81	41.90	**99**	1798	94.96	**1.97**	**95.30**	**99.89**
hrt-c	82.11	8.28	81	1438	**83.11**	**1.54**	**81.40**	**99.89**
hrt-h	68.61	5.08	**78**	2816	61.56	**1.14**	**77.56**	**99.96**
hrt-s	47.88	0.70	**78**	106	50.0	**0.60**	**14.29**	**99.43**
hrt-lb	56.68	1.71	**68**	3505	55.85	**1.41**	**17.54**	**99.96**
hpts	77.80	0.81	71	794	**79.86**	**0.74**	**8.64**	**99.91**
bld	**70.52**	11.24	63	3843	66.90	**2.08**	**81.49**	**99.95**
pid	**75.88**	140.21	70	16354	72.87	**4.74**	**96.62**	**99.97**
SPECTF	72.81	29.55	**73**	1595	64.81	**2.12**	**92.83**	**99.87**
SPECT	67.92	1.21	70	1809	**74.81**	**0.93**	**23.14**	**99.95**
prks	70.45	4.61	**100**	205	68.07	**1.38**	**70.07**	**~100**
SB	91.99	16191	n/a	n/a	**93.73**	**61.06**	**99.62**	**~100**
WFR	**99.74**	1582	n/a	n/a	98.83	**16.01**	**98.99**	**~100**
LR	n/a	n/a	n/a	n/a	**83.17**	**181.49**	**~100**	**~100**
MG	n/a	n/a	n/a	n/a	**84.42**	**35.37**	**~100**	**~100**
MBP	n/a	n/a	n/a	n/a	**88.88**	**276.83**	**~100**	**~100**
SS	n/a	n/a	n/a	n/a	**84.52**	**145.33**	**~100**	**~100**
CT	n/a	n/a	n/a	n/a	**71.52**	**612.64**	**~100**	**~100**

terms of accuracy.

Phase 2 of the Friedman test is aimed to distinguish the best model that will lead to a significantly lower computational time or significantly higher accuracy. Accordingly, pairwise comparisons were performed between LAD-ENS and each model j of the two other LAD models. The null and the alternative hypothesis are formulated as follows:

$$H_0:\ \mu_{LAD-ENS} = \mu_j$$
$$H_a:\ \mu_{LAD-ENS} \neq \mu_j$$

The absolute difference (AD) between the rank sums, $|R_{LAD-ENS} - R_j|$, is computed, where R_j is the rank summation for the model j. If the AD exceeds the post-hoc value $d_{\alpha_f}\sqrt{Nk(k+1)/6}$, the null hypothesis is rejected. d_{α_f} is the $100(1-\alpha_f)^{th}$ of the standard normal distribution, α_f is the family-wise significant level, N is the number of datasets, and k is the number of models [33]. As shown in Table 4, the results of phase 2 declared that the LAD-ENS model significantly outperforms both cbmLAD and MPG-LAD models in terms of computational time. However, phase 2 declared that the accuracies are not significantly different between LAD-ENS and the other models, as shown in Table 5. Moreover, LAD-ENS was able to handle the last seven datasets, while cbmLAD only handled two and MPG-LAD was not able to handle any.

In order to study how LAD-ENS affects the interpretability power of LAD, we introduce the complexity measure shown in equation (4). This measure is computed for LAD-ENS and cbmLAD models for each dataset to give a quantitative measure of the model's complexity. Furthermore, a comprehensibility index (CI) is introduced in equation (5). Values of CI near 0 indicate a low explanatory power, whereas values near 1 indicate a high explanatory power.

$$Complexity = \frac{No.of\,Patterns \times Avg.degree\ of\ Patterns}{No.of\,Classes} \tag{4}$$

$$CI = \frac{1}{Complexity} \tag{5}$$

Table 6 shows a comparison between LAD-ENS and the cbmLAD model in terms of the number of patterns, average degree of the patterns, and the CI over the datasets that cbmLAD was able to solve.

TABLE 5
FREIDMAN TEST ON ACCURACY PERFORMANCE

Model	Phase 1		Phase 2		
	Sum of ranks (R)	Mean of ranks	AD	$AD\ >\ post-hoc$?	Significant?
cbmLAD	25	1.92	5.0	*No*	*No*
MPG-LAD	23	1.77	7.0	*No*	*No*
LAD-ENS	30	2.30			
($F_r = 2.0\ <\ 6.0,\ p = 0.368\ >\ 0.05$) → fail to reject H_0			$\alpha_f = 0.025,\ \ d_{\alpha_f} = 1.96,\ \ post\text{-}hoc\ value = 9.994$		

TABLE 4
FREIDMAN TEST ON COMPUTATIONAL TIME PERFORMANCE

Model	Phase 1		Phase 2		
	Sum of ranks (R)	Mean of ranks	AD	$AD\ >\ post\text{-}hoc$?	Significant?
cbmLAD	26	2	13	*yes*	*yes*
MPG-LAD	39	3	26	*yes*	*yes*
LAD-ENS	13	1			
($F_r = 26.0\ >\ 6.0,\ p = 0.0 < 0.05$) → Rejecting H_0			$\alpha_f = 0.025,\ \ d_{\alpha_f} = 1.96,\ \ post\text{-}hoc\ value = 9.994$		

TABLE 6
COMPARING THE PATTERNS GENERATED BY LAD-ENS AND CBMLAD

Dataset	LAD-ENS			cbmLAD		
	# of patterns	Avg. Degree	CI (E-3)	# of patterns	Avg. Degree	CI (E-3)
wbc	35	**2.24**	**25.52**	**28**	4.95	14.44
wpbc	24	**7.50**	**11.12**	**23**	10.41	8.36
wdbc	25	**5.16**	**15.5**	**22**	9.25	9.82
hrt-c	**46**	**4.13**	**10.52**	54	5.36	6.9
hrt-h	**40**	**3.83**	**13.06**	49	4.72	8.64
hrt-s	**10**	**3.96**	**50.5**	11	4.75	38.28
hrt-lb	44	**3.29**	**13.82**	44	3.41	13.32
hpts	**10**	**3.96**	**50.5**	16	4.99	25.06
bld	**70**	4.46	**6.4**	75	**4.36**	6.12
pid	**127**	**4.38**	**3.6**	158	5.71	2.22
SPECTF	**19**	**12.1**	**8.7**	28	15.86	4.5
SPECT	35	**4.80**	**11.9**	**33**	5.42	11.18
prks	**13**	**3.16**	**48.68**	17	6.48	18.16
SB	307	**12.63**	**0.52**	**283**	15.26	0.46
WFR	129	4.89	6.36	**34**	**3.26**	**36.08**
WFR[a]	36	**2.72**	**40.84**	**34**	3.26	36.08

[a] Patterns with homogeneity less than the 75^{th} percentile are eliminated from the LAD-ENS model.

LAD-ENS generated a number of patterns, up to 25% more in some datasets, compared to the number of patterns in cbmLAD. In other datasets, LAD-ENS produced a number of patterns that was less than that of cbmLAD by 6.6% to 37%. On the other hand, the average degree of patterns was reduced in most datasets, which led to an increase in the CI of LAD-ENS over classical LAD. The sampling process that is used to extract the data subsets with better separability between the classes explains this decrease in the average degree of the patterns. Therefore, a small number of patterns are generated from each data subset with low degrees. This resulted in developing a LAD-ENS model with a reasonable number of patterns and with low degrees compared to the classical LAD model. For a WFR dataset, this reduction in the average degree of patterns did not prevent an increase in the complexity of the LAD-ENS model. This is due the number of patterns that increases significantly, 3.8 times, compared to the number of patterns generated by the classical cbmLAD model. Nevertheless, LAD-ENS has an efficient computational time of 16 seconds compared to 1582 seconds of cbmLAD. However, by eliminating LAD-ENS patterns that have homogeneity less than the 75^{th} percentile, the number of patterns is considered only 5% more than the number of patterns in cbmLAD. Moreover, the average degree is reduced to 2.72, resulting in a CI of 40.84×10^{-3} which is better than the CI of cbmLAD. Removing these low homogeneity patterns does not affect the accuracy of LAD-ENS on a WFR dataset, as will be discussed further in the paper.

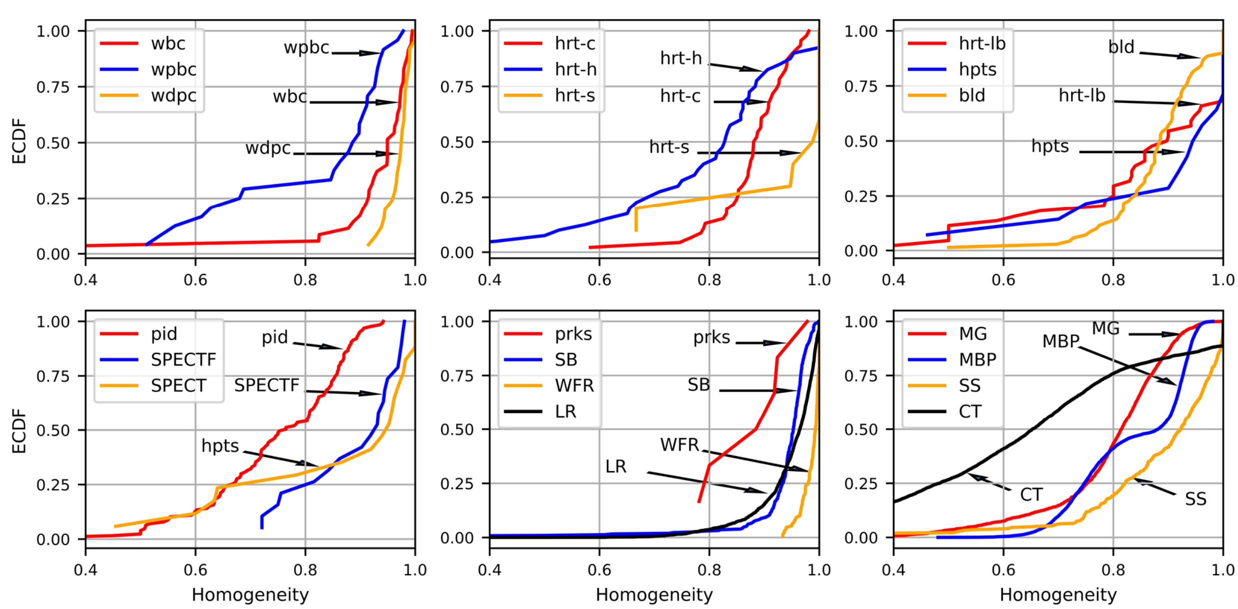

Fig. 5. Empirical cumulative distribution functions for the homogeneity of LAD-ENS patterns

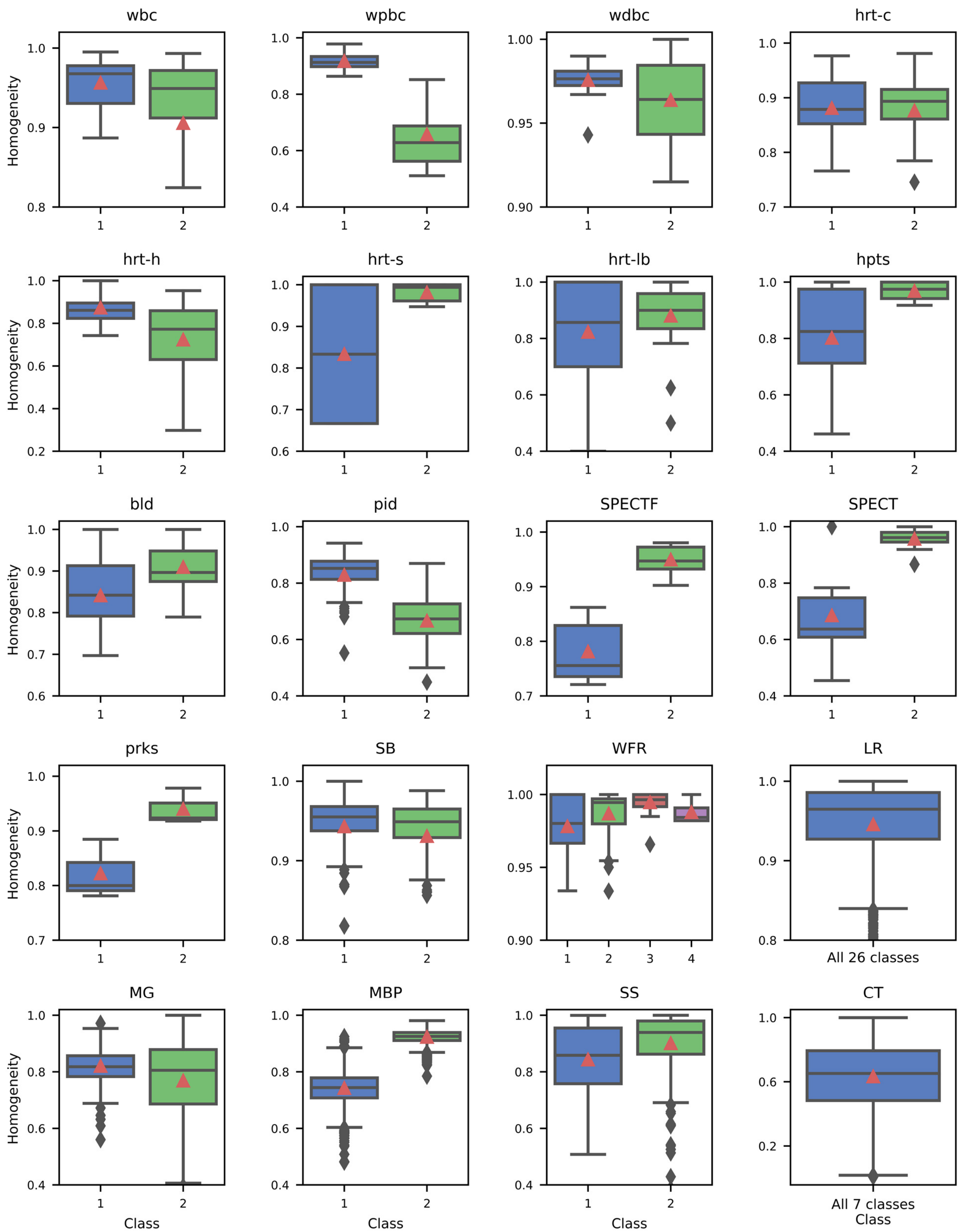

Fig. 6. Box plots of the homogeneity of LAD-ENS patterns

The main limitation of LAD-ENS is that the homogeneity of some patterns can change when scanning the entire dataset. As each pattern was generated on one chunk of data, a pattern might cover observations that belong to opposite classes from other chunks. In the case of a positive (negative) pattern, homogeneity is the proportion of the covered positive (negative) observations over all of the covered observations from positive and negative classes. The homogeneity of a pattern is an important characteristic, since it refers to the confidence of a positive/negative pattern belonging to a positive/negative class.

In order to analyze the quality of patterns in terms of homogeneity and prevalence, all observations are scanned by each pattern for every dataset. Homogeneity and prevalence are calculated for each pattern. Empirical cumulative distribution functions and box plots of the homogeneity are shown in Fig. 5 and Fig. 6, respectively. The means of homogeneity are illustrated with red triangles in Fig. 5. These

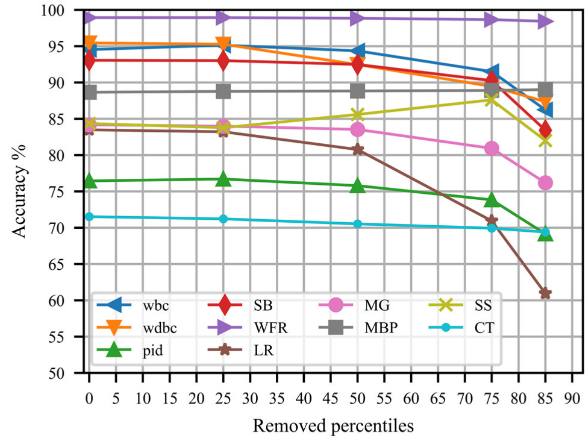

Fig. 7. Testing accuracy of LAD-ENS after discarding patterns with homogeneity lower than different percentiles.

figures illustrate that 25% or fewer patterns have homogeneity that is lower than 70% over most datasets. The figures also show that most of the generated patterns have a very high homogeneity/confidence of 0.8 or more.

The results obtained with the proposed LAD-ENS mechanism are encouraging, since they demonstrate that the patterns generated by LAD-ENS could be used to guide a decision maker when monitoring a business or industrial process of interest. For example, controlling parameters of a manufacturing process such as the operating conditions and the measurements that keep the process under control, and predicting future events such as failure, alarm, and fraudulent cases. In addition, patterns with a high confidence are also useful in monitoring the status of a patient regarding a specific disease, and in detecting network intruders and spam emails.

In order to investigate whether patterns with low homogeneity negatively affect classification accuracy, the patterns with homogeneity lower than the 25^{th}, 50^{th}, 75^{th} and 85^{th} percentiles were eliminated when forming the discriminant function. After each elimination, the testing accuracy is computed using a new set of patterns. The accuracy of the results is shown in Fig. 7. The results demonstrate that accuracy is not negatively affected by keeping low homogeneity patterns. In most of the datasets, testing accuracy decreases by discarding patterns that have a homogeneity lower than the 75^{th} percentiles. Therefore, low homogeneity patterns could be eliminated to enhance the CI of the model, as we did earlier for the WFR dataset.

TABLE 7
THE PREVALENCE OF PATTERNS GENERATED BY LAD-ENS AND CBMLAD

Dataset	Average prevalence	
	LAD-ENS	cbmLAD
wbc	**0.79**	0.42
wpbc	**0.44**	0.33
wdbc	**0.82**	0.75
hrt-c	**0.39**	0.22
hrt-h	**0.29**	0.15
hrt-s	**0.49**	0.45
hrt-lb	**0.12**	0.09
hpts	**0.58**	0.477
bld	**0.13**	0.08
pid	**0.29**	0.07
SPECTF	**0.56**	0.51
SPECT	**0.25**	0.11
prks	**0.6**	0.5
SB	**0.37**	0.15
WFR	**0.75**	0.68

Additionally, the prevalence of patterns is illustrated in Table 7 and compared with cbmLAD for the datasets that cbmLAD was able to solve. The patterns of LAD-ENS show a high average prevalence compared with the patterns of cbmLAD. Overall, while the purity of the patterns might be lost in LAD-ENS, it provides better prevalence and explanatory power.

Finally, Table 8 provides a comparison between LAD-ENS with accuracy results of other well-known machine learning techniques, as summarized in [20], such as support vector machines (SVM) [3], decision tree (J48) [34], random forest (RF) [25], multilayer perceptron (NN) [4] and logistic regression (LR). It can be observed that LAD-ENS provides competitive classification performance. This is obvious for the wbc, wdbc, hrt-c, bld, and pid datasets. Although SVM outperforms LAD-ENS in some datasets, SVM lacks the capability of providing interpretable results. As explained before, LAD-ENS provides the interpretability power that guides a decision maker when monitoring a business or industrial process of interest.

V. CONCLUSION AND FUTURE WORK

In this paper, we have developed an ensemble LAD system called LAD-ENS to enhance computation time and to train, test and classify large volumes of data. This ensemble system was built based on stratified sampling without a replacement technique, in addition to a proposed combining mechanism that integrates all patterns that are provided by the individual LAD classifiers. This mechanism combines the knowledge at the level of patterns and then formulates a new ensemble discriminant function. This mechanism preserves the explanatory power of LAD patterns and does not reduce their

TABLE 8
COMPARING THE ACCURACY OF LAD-ENS WITH OTHER MACHINE LEARNING TECHNIQUES

Dataset	LAD-ENS	SVM	J48	RF	NN	LR
wbc	**94±4**	**97±2**	94±2	97±1	96±2	96±2
wpbc	60±7	77±2	75±5	80±4	77±5	**80±5**
wdbc	**95±3**	97±1	93±2	96±2	97±1	**97±2**
hrt-c	**83±4**	**84±5**	78±5	83±5	79±5	83±5
hrt-h	61±2	81±4	79±4	80±4	78±5	**83±5**
hrt-s	50±4	**94±2**	93±2	93±3	89±6	92±4
hrt-lb	55±5	75±1	72±5	**75±4**	69±7	74±4
hpts	79±4	**87±5**	82±6	**87±5**	81±6	85±6
bld	**66±5**	58±0	62±5	**73±6**	68±6	69±5
pid	**72±6**	**77±3**	74±3	76±3	75±3	77±3
SPECTF	74±5	79±0	78±5	**81±3**	77±5	79±4
SPECT	64±4	**83±4**	80±3	82±4	80±4	82±5
prks	68±8	87±4	83±7	91±5	**92±5**	85±6

interpretability like the voting mechanism does. By means of this system, we successfully ran LAD-ENS on cloud computing clusters. The LAD-ENS system was evaluated in terms of computational time, accuracy, pattern quality and comprehensibility. The statistical Friedman tests revealed that LAD-ENS significantly outperforms classical LAD models in terms of computational performance. Moreover, Friedman tests revealed that very competitive accuracy is obtained. Although the patterns may lose their purity, the patterns of LAD-ENS have lower degrees than those of classical LAD, which enhanced the comprehensibility index. The computational experiments show that if the sampling process is not able to relax complex boundaries of a dataset enough, the number of generated patterns in LAD-ENS will increase significantly and reduce the explanatory power of the patterns. However, by eliminating low homogeneity patterns, the explanatory power will improve significantly, and accuracy performance will not be affected. In general, the proposed LAD-ENS demonstrates better performance in terms of computational time and quality of patterns over classical LAD models.

The novel research on ensemble LAD systems introduced in this paper could be extended into many different directions. One of these directions would be to use various sampling methods to enhance the quality of the data subsets provided to the pattern generation processes. This feature level could be another direction, focusing mainly on the features of original data in order to select appropriate subsets, or sample the features in subsets and provide them to the pattern generation processes with an aim to enhance accuracy. The combining mechanism level is another direction, which would focus mainly on enhancing the combining process and selecting the most appropriate patterns. This research direction could enhance the explanatory power by reducing the number of patterns.

REFERENCES

[1] I. Odun-Ayo, C. Okereke, and H. Orovwode, "Cloud computing and internet of things: Issues and developments," Lecture Notes in Engineering and Computer Science: Proceedings of *The World Congress on Engineering* 2018, London, U.K., 4-6 July, 2018, pp182-187.

[2] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE transactions on systems, man, and cybernetics*, vol. 21, no. 3, pp660-674, 1991.

[3] B. Scholkopf and A. J. Smola, Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press, 2001.

[4] M. T. Hagan, H. B. Demuth, M. H. Beale, and O. De Jesús, Neural network design. Pws Pub. Boston, 1996.

[5] E. Boros, P. L. Hammer, T. Ibaraki, and A. Kogan, "Logical analysis of numerical data," *Mathematical Programming*, vol. 79, no. 1-3, pp163-190, 1997.

[6] E. Boros, P. L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz, and I. Muchnik, "An implementation of logical analysis of data," (in English), *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 2, pp292-306, Mar-Apr 2000, doi: 10.1109/69.842268.

[7] P. L. Hammer and T. O. Bonates, "Logical analysis of data—An overview: From combinatorial optimization to medical applications," *Annals of Operations Research*, vol. 148, no. 1, pp203-225, 2006.

[8] R. M. Khalifa, S. Yacout, and S. Bassetto, "Developing machine-learning regression model with Logical Analysis of Data (LAD)," *Computers & Industrial Engineering*, vol. 151, 2021, doi: 10.1016/j.cie.2020.106947.

[9] M.-A. Mortada, S. Yacout, and A. Lakis, "Diagnosis of rotor bearings using logical analysis of data," *Journal of Quality in Maintenance Engineering*, vol. 17, no. 4, pp371-397, 2011.

[10] A. Bennane and S. Yacout, "LAD-CBM; new data processing tool for diagnosis and prognosis in condition-based maintenance," (in English), *Journal of Intelligent Manufacturing*, vol. 23, no. 2, pp265-275, Apr 2012, doi: 10.1007/s10845-009-0349-8.

[11] M.-A. Mortada, S. Yacout, and A. Lakis, "Fault diagnosis in power transformers using multi-class logical analysis of data," *Journal of Intelligent Manufacturing*, vol. 25, no. 6, pp1429-1439, 2013, doi: 10.1007/s10845-013-0750-1.

[12] A. Ragab, S. Yacout, M.-S. Ouali, and H. Osman, "Multiple failure modes prognostics using logical analysis of data," in 2015 *Annual Reliability and Maintainability Symposium (RAMS)*, 2015: IEEE, pp1-7.

[13] A. Ragab, M.-S. Ouali, S. Yacout, and H. Osman, "Remaining useful life prediction using prognostic methodology based on logical analysis of data and Kaplan–Meier estimation," *Journal of Intelligent Manufacturing*, vol. 27, no. 5, pp943-958, 2016.

[14] Y. Shaban, S. Yacout, M. Balazinski, and K. Jemielniak, "Cutting tool wear detection using multiclass logical analysis of data," *Machining Science and Technology*, vol. 21, no. 4, pp526-541, 2017.

[15] S. Jocelyn, Y. Chinniah, M.-S. Ouali, and S. Yacout, "Application of logical analysis of data to machinery-related accident prevention based on scarce data," *Reliability Engineering & System Safety*, vol. 159, pp223-236, 2017.

[16] A. Ragab, M. El-Koujok, B. Poulin, M. Amazouz, and S. Yacout, "Fault diagnosis in industrial chemical processes using interpretable patterns based on Logical Analysis of Data," *Expert Systems with Applications*, vol. 95, pp368-383, 2018.

[17] A. Ghasemi, S. Esmaeili, and S. Yacout, "Development of Equipment Failure Prognostics Model Based on Logical Analysis of Data (LAD)," *Engineering Letters*, vol. 21, no. 4, pp256-263, 2013.

[18] S. Alexe and P. L. Hammer, "Accelerated algorithm for pattern detection in logical analysis of data," (in English), *Discrete Applied Mathematics*, vol. 154, no. 7, pp1050-1063, May 1 2006, doi: 10.1016/j.dam.2005.03.032.

[19] P. Hansen and C. Meyer, "A new column generation algorithm for Logical Analysis of Data," *Annals of Operations Research*, vol. 188, no. 1, pp215-249, 2011, doi: 10.1007/s10479-011-0850-2.

[20] C.-A. Chou, T. O. Bonates, C. Lee, and W. A. Chaovalitwongse, "Multi-pattern generation framework for logical analysis of data," *Annals of Operations Research*, vol. 249, no. 1-2, pp329-349, 2015, doi: 10.1007/s10479-015-1867-8.

[21] V. S. Moertini, G. W. Suarjana, L. Venica, and G. Karya, "Big Data Reduction Technique using Parallel Hierarchical Agglomerative Clustering," *IAENG International Journal of Computer Science*, vol. 45, no. 1, pp188-205, 2018.

[22] L. Zhou, S. Pan, J. Wang, and A. V. Vasilakos, "Machine learning on big data: Opportunities and challenges," *Neurocomputing*, vol. 237, pp350-361, 2017.

[23] Y. Zhang, S. Ren, Y. Liu, and S. Si, "A big data analytics architecture for cleaner manufacturing and maintenance processes of complex products," (in English), *Journal of Cleaner Production*, vol. 142, pp626-641, Jan 20, 2017, doi: 10.1016/j.jclepro.2016.07.123.

[24] H. Li, R. Liu, J. Wang, and Q. Wu, "An enhanced and efficient clustering algorithm for large data using MapReduce," *IAENG International Journal of Computer Science*, vol. 46, no. 1, pp61-67, 2019.

[25] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp5-32, 2001/10/01 2001, doi: 10.1023/A:1010933404324.

[26] T. Q. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Kdd'16: Proceedings of the 22nd *Acm Sigkdd International Conference on Knowledge Discovery and Data Mining*, 2016, pp785-794, doi: 10.1145/2939672.2939785.

[27] Y. Crama, P. L. Hammer, and T. Ibaraki, "Cause-effect relationships and partially defined Boolean functions," *Annals of Operations Research*, vol. 16, no. 1, pp299-325, 1988, doi: 10.1007/bf02283750.

[28] M. Lejeune, V. Lozin, I. Lozina, A. Ragab, and S. Yacout, "Recent advances in the theory and practice of Logical Analysis of Data," *European Journal of Operational Research*, vol. 275, no. 1, pp1-15, 2019, doi: 10.1016/j.ejor.2018.06.011.

[29] L. M. Moreira, "The use of Boolean concepts in general classification contexts," EPFL, 2000.

[30] S. Yacout, D. Salamanca, and M.-A. Mortada, "Tools and Methods for fault Detection of devices by Condition based Maintenance.," USA Patent US9824060B2, 2017.

[31] D. M. Tax and R. P. Duin, "Using two-class classifiers for multiclass classification," in *Object recognition supported by user interaction for service robots,* vol. 2: IEEE, pp124-127, 2002.

[32] C. Zhang and Y. Ma, Ensemble Machine Learning. 2012.

[33] D. J. Henderson and C. F. Parmeter, "Applied nonparametric econometrics," Cambridge University Press, 2015

[34] J. R. Quinlan, C4.5: programs for machine learning. Elsevier, 2014.