Recent Advances in the Theory and Practice of Logical Analysis of Data

Miguel Lejeune* Vadim Lozin† Irina Lozina‡ Ahmed Ragab§
Soumaya Yacout¶

Abstract

Logical Analysis of Data (LAD) is a data analysis methodology introduced by Peter L. Hammer in 1986. LAD distinguishes itself from other classification and machine learning methods by the fact that it analyzes a significant subset of combinations of variables to describe the positive or negative nature of an observation and uses combinatorial techniques to extract models defined in terms of patterns. In recent years, the methodology has tremendously advanced through numerous theoretical developments and practical applications. In the present paper, we review the methodology and its recent advances, describe novel applications in engineering, finance, health care, and algorithmic techniques for some stochastic optimization problems, and provide a comparative description of LAD with well-known classification methods.
Keywords: Logical Analysis of Data, Boolean Mathematics, Pattern, Data Mining, Combinatorial Optimization.

1 Introduction

In 1986, Peter L. Hammer gave a lecture at the International Conference on Multi-attribute Decision Making via OR-based Expert Systems [47], where he outlined basic ideas of a new approach to data analysis, known nowadays as Logical Analysis of Data (LAD). Later this approach was expanded and developed in [33]. That first publication was followed by a stream of research studies developing the theory and methodology of LAD, see e.g. [10, 27, 30, 54, 88, 89]. One of the main advantages of LAD is its explanatory power, i.e. it offers a classification together with an explanation, which can be easily understood by experts. This has led to numerous practical applications of this methodology varying from medicine to credit risk ratings. A software implementation of the LAD methodology is publicly available on the LADWEKA web site [64] with a tutorial [19].

In recent years, LAD found many more applications and witnessed a remarkable progress in theoretical and methodological development. The purpose of the present paper is to review

*Corresponding author: Department of Decision Sciences, GWSB, George Washington University, Washington, DC, 22202, USA, mlejeune@gwu.edu

†Mathematics Institute, University of Warwick, Coventry, CV4 7AL, UK, V.Lozin@warwick.ac.uk

‡Mathematics Institute, University of Warwick, Coventry, CV4 7AL, UK, I.Lozina@warwick.ac.uk

§Department of Mathematics and Industrial Engineering, Ecole Polytechnique de Montréal, Montréal, Québec, Canada H3C 3A7 & Department of Industrial Electronics and Control Engineering, Faculty of Electronic Engineering, Menoufia University, Menouf, 32952, Egypt, ahmed.ragab@polymtl.ca

¶Department of Mathematics and Industrial Engineering, Ecole Polytechnique de Montréal, Montréal, Québec, Canada H3C 3A7, soumaya.yacout@polymtl.ca

Day	Food item								Headache
	1	2	3	4	5	6	7	8	
1		x		x		x	x		Yes
2	x		x		x		x		No
3				x	x	x			No
4	x	x		x		x		x	No
5	x	x		x	x			x	Yes
6			x		x		x		No
7		x	x		x			x	Yes

Table 1: Introductory example – diet record

and record the achievements of LAD obtained in recent years. Earlier overviews of the LAD methodology can be found in [2, 20, 30]. We start with a short tutorial introducing the reader to the fundamental concepts of Logical Analysis of Data in Section 2. Then in Section 3 we turn to theoretical and methodological developments obtained in the recent years. Section 4 illustrates the power of LAD by a variety of practical applications. Section 5 presents some open research areas and a comparative analysis of LAD's accuracy performance.

2 A Short LAD Tutorial

We start with an introductory example proposed in [33]. A physician would like to find out the combination of food items which cause a headache to one of his patients, and requests his patient to keep a record of his diet. One week later, the patient returns to the doctor and brings in the record displayed in Table 1.

After a brief examination, the doctor concludes that on the days when the patient had no headache, he never consumed food #2 without food #1, but he did so on some of the occasions when he had a headache. Similarly, our clever doctor concludes that the patient has never consumed food #4 without food #6 on the days when he had no headache; but he did so once, and he had a headache. He finally concludes that the two "patterns" noticed above explain every headache, and he puts forward the "theory" that this patient's headaches can always be explained by using these two patterns.

This example captures the essence of LAD methodology: detect patterns and build a theory. In most practical applications, these two major steps are preceded by preparatory work needed to make the data amenable to the techniques of LAD.

Typically, the data comes as a collection of *observations* and this collection is frequently referred to as an *archive*. Each observation is an n-dimensional vector having as components the values of n *attributes*, also known as *features* or *variables*. To make the data amenable to the techniques of LAD, it must be represented in a form known as partially defined Boolean function.

2.1 Boolean Function Terminology

Let us denote $B = \{0, 1\}$. The set B^n consists of all binary words (i.e. ordered sequences of 0's and 1's) of length n, also known as 0-1 n-vectors, and is commonly referred to as the *Boolean hypercube* of dimension n.

A *Boolean function* of n variables $x_1, \ldots, x_n$ is a mapping $f : B^n \to B$. Given a Boolean function f, a binary vector $\alpha = (\alpha_1 \alpha_2 \ldots \alpha_n)$ is a called a *true point* of the function if $f(\alpha) = 1$ and a *false point* if $f(\alpha) = 0$. The sets of true and false points of a function f will be denoted by $T = T(f)$ and $F = F(f)$, respectively.

If x is a Boolean variable (i.e. variable taking values 0 and 1), then $\overline{x} = 1 - x$ is the *complement* (or *negation*) of x . Both the variables and their complements are called *literals*. A *term* is a product of literals. The *degree* of a term is the number of literals in it. A term t is said to *cover* a point $\alpha \in \{0,1\}^n$ if $t(\alpha) = 1$. The subset of B^n covered by a term t is known as a *subcube* of B^n.

Every partition of the set B^n of all 0-1 n-vectors into two disjoint sets T and F defines a Boolean function on B^n. Now assume that the sets T and F are disjoint but cover B^n not entirely, i.e. some points of B^n belong neither to T nor to F. Then we have a function which is defined only partially, a *partially defined Boolean function* (pdBf). This function is given by:

$$f(\alpha) = \begin{cases} 1 \text{ if } \alpha \in T \\ 0 \text{ if } \alpha \in F \end{cases}$$

A function f defined on a set of true points T and a set of false points F will be denoted $f = (T, F)$.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	$f(x_1, \ldots, x_8)$
1	0	1	0	1	0	1	1	0	1
2	1	0	1	0	1	0	1	0	0
3	0	0	0	1	1	1	0	0	0
4	1	1	0	1	0	1	0	1	0
5	1	1	0	1	1	0	0	1	1
6	0	0	1	0	1	0	1	0	0
7	0	1	1	0	1	0	0	1	1

Table 2: A partially defined Boolean function

Table 2 gives an example of a partially defined Boolean function of eight variables. This function is defined only on 7 points of the hypercube B^8 numbered 1 through 7. The points 1,5,7 are the true points of the function and 2,3,4,6 are its false points. An attentive reader can easily recognize in this function the diet record of Table 1. This table can be extended by adding to it new records. Similarly, every partially defined Boolean function can be extended by defining its values on new points of the hypercube. Every Boolean function agreeing with a pdBf $f = (T, F)$ on $T \cup F$ and taking arbitrary 0-1 values elsewhere is called an *extension* of f. The number of extensions can be very large. Among many possible extensions, LAD aims at distinguishing a "right" one. There is no definition for a "right extension". However, we assume that any real-life dataset is not just a collection of random facts and that any rational phenomenon (headache, etc.) has a rational explanation. The idea of LAD is to learn this explanation from the partial information at hand. As we mentioned earlier, LAD does this job in two major steps: detecting patterns and building a theory.

2.2 Patterns

Let $f = (T, F)$ be a partially defined Boolean function. A term t is a *positive pattern* of f if it covers at least one true point and no false point of f. Alternatively, a *positive pattern* is a

subcube of B^n that intersects T and is disjoint from F. *Negative patterns* are defined by analogy.

Patterns play a key role in LAD, since they admit a clear interpretation by human experts. Consider, for instance, the partially defined Boolean function of Table 2, which models the diet record of the introductory example. The term $\overline{x}_1x_2$ equals 1 if and only if $x_1 = 0$ and $x_2 = 1$, therefore it covers points 1 and 7 and does not cover any other point of the table. Since 1 and 7 are the true points of the function, we conclude that $\overline{x}_1x_2$ is its positive pattern. This pattern suggests a special role of food 2 ($x_2 = 1$) consumed without food 1 ($x_1 = 0$). Similarly, $x_4\overline{x}_6$ is a positive pattern. The only point which is covered by this term is point 5, and this is a true point. Below we shall see that the pdBf of Table 2 has many more patterns.

Typically, a partially defined Boolean function has exceedingly many patterns and the identification of all of them is computationally expensive. In addition, it has been observed in empirical studies and practical applications that some patterns are more "suitable" than others for use in data analysis. Unfortunately, the concept of suitability does not have a unique definition. Among the many reasonable criteria of suitability, paper [51] distinguishes three basic types of patterns: prime, strong and spanned. To define these notions, let us denote by $Lit(P)$ the set of literals in a pattern P and by $Cov(P)$ the coverage of P, i.e. the set of true points covered by P.

A pattern P is *prime* if there is no pattern P' such that $Lit(P') \subset Lit(P)$, i.e. if the removal of any literal from $Lit(P)$ results in a term which is not a pattern. A pattern P is *strong* if there is no pattern P' such that $Cov(P) \subset Cov(P')$. A pattern P is *spanned* if it is strong and there is no pattern P' such that $Cov(P) = Cov(P')$ and $Lit(P) \subset Lit(P')$.

To illustrate these notions, let us return to the example of the partially defined Boolean function of Table 2. It is not difficult to see that the term $x_5\overline{x}_6x_8$ is a positive pattern of this function and the set of true points covered by it consists of points 5 and 7. In order to see if it is prime, let us try to obtain a shorter term by deleting one of its literals. By deleting x_5 we obtain the term $\overline{x}_6x_8$, which covers the negative point 4 and hence is not a positive pattern anymore. Similarly, by deleting x_8 we obtain a term which is not a pattern of the function. However, the term x_5x_8 obtained by deleting $\overline{x}_6$ is a positive pattern covering points 5 and 7. Therefore, $x_5\overline{x}_6x_8$ is not prime. On the other hand, x_5x_8 is prime, since the deletion of any literal from it results in a term which is not a pattern. The pattern x_5x_8 is also strong, simply because there are no patterns covering more than two true points, which can be easily verified. It is, however, not spanned. Indeed, the $x_5\overline{x}_6x_8$ covers the same set of true points as x_5x_8, but $Lit(x_5x_8) \subset Lit(x_5\overline{x}_6x_8)$. With a bit of work the reader can find out that $x_2x_5\overline{x}_6\overline{x}_7x_8$ is a spanned pattern, i.e. we cannot add more literals to the pattern without decreasing the coverage. Finally, we observe that the term $\overline{x}_1x_2$, which is a positive pattern, is prime, strong and spanned.

One more type of patterns was introduced in [21] under the name maximum patterns. For a binary vector α, an α-pattern is a pattern covering α. A *maximum* α-pattern is an α-pattern P with maximum coverage, i.e. with maximum number of positive points covered by P (if α is positive) or with maximum number of negative points covered by P (if α is negative). Remember that by definition P cannot cover both a positive and a negative point.

Patterns are also distinguished by three major parameters: degree, prevalence and homogeneity. We repeat that the degree of a pattern is the number of literals in it. In practice, patterns of small degree are always preferable because of their higher explanatory power. In other words, patterns of small degree are easier to interpret. Below we define the other two parameters.

The *prevalence* of a positive pattern is the ratio (sometimes expressed as the percentage) of the number of positive points covered by the pattern to the number of all positive points in the data set. The prevalence of a negative pattern is defined analogously. Obviously, patterns of high prevalence are more valuable.

In order to define the notion of homogeneity, we need to slightly relax the definition of a positive (negative) pattern. According to the original definition, a positive pattern is a subcube covering at least one positive and *no* negative point. In practice, finding such subcubes may result in patterns of very small prevalence. However, if we allow a subcube to cover a "few" negative points, the search may result in patterns with substantially higher prevalence. This observation justifies the following definition. The *homogeneity* of a positive pattern is the ratio (percentage) of the number of positive points covered by the pattern to the number of all points covered by it. The homogeneity of a negative pattern is defined analogously. Patterns of 100% homogeneity sometimes are called *pure* patterns.

2.3 Theory Formation

The pattern generation step produces a set of patterns called the *pandect*. The next step is to build a *theory*, i.e. an extension of the partially defined Boolean function representing the data.

The number of patterns in the pandect may be too large to allow the effective utilization of all of them. This leads to the problem of selecting a representative subset of patterns capable of providing classifications for the same set of points in the archive which can be classified by the pandect. The set of the selected patterns is called a *model*. The model should, on the one hand, be of reasonable size, but, on the other hand, it should allow us to distinguish between the positive and the negative observations. In [23], the problem of selecting patterns for the model was formulated as a set covering problem. A variation of this problem was also studied [53].

LAD classifies observations on the basis of model's evaluation of them as follows. An observation satisfying the conditions of some of the positive (resp. negative) patterns in the model, and not satisfying the conditions of any of the negative (resp. positive) patterns in the model, is classified as positive (resp. negative). To classify an observation that satisfy both positive and negative patterns in the model, LAD constructs a *discriminant* (or *discriminating function*) that assigns relative weights to the patterns in the model.

2.3.1 Discriminant

The idea of the notion of discriminant is to emphasize the relative importance of patterns by assigning to them weights. To a positive pattern P_k we assign a positive weight w_k^+, and to a negative pattern N_l we assign a negative weight w_l^-. Then the discriminant is the following weighted sum:

$$\Delta(\alpha) = \sum_k w_k^+ P_k(\alpha) + \sum_l w_l^- N_l(\alpha), \tag{1}$$

where $P_k(\alpha)$ ($N_l(\alpha)$) is the value of P_k (N_l) at a point α (i.e. 1 or 0 depending on whether the point is covered or not by the pattern) . The weights of the patterns are chosen in such a way that a large positive (negative) value of the discriminant at a new observation point will be indicative of the positive (negative) character of that point.

If all weights have the same absolute value, then all patterns are equally important. On the other hand, the number q_k of observation points in the archive covered by a pattern P_k can be viewed as an indication of its relative importance, justifying the choice $|w_k| = q_k$. The relative

importance of patterns can be emphasized even stronger by choosing $|w_k| = q_k^2$ or $|w_k| = q_k^3$ or $|w_k| = 2^{q_k}$. This approach can be generalized by choosing weights on the basis of appropriately defined distances from a pattern to the sets of positive and negative observations in the archive. Another reasonable point of view emphasizing the role of simple (i.e. short) patterns defines $|w_k| = 1/d_k$, where d_k is the degree of the pattern P_k.

In view of the possible disparity between the number of positive and of negative patterns, the weights may have to be normalized by a constant factor, assuring that $\sum_k w_k^+ = -\sum_l w_l^- = 1$. In the simplest case of equal weights, the normalized discriminant is calculated as $\Delta(\alpha) = \alpha_p/p - \alpha_n/n$, where α_p and α_n are, respectively, the number of positive and the number of negative patterns covering α, while p and n are, respectively, the number of all positive and the number of all negative patterns in the model.

If $\Delta(\alpha)$ is positive, the observation α is classified as positive, and if $\Delta(\alpha)$ is negative, then α is classified as negative. LAD leaves unclassified any observation α for which $\Delta(\alpha) = 0$, since in this case either the model does not provide sufficient evidence, or the evidence it provides is contradictory.

2.4 Preprocessing

Let us repeat that, speaking theoretically, the main objective of LAD is to find an extension of a partially defined Boolean function by means of revealing logical patterns hidden in the data. In practice, this general goal is frequently accompanied by a number of auxiliary problems and intermediate steps that have to be implemented to achieve the goal. The main two of them are binarization and attribute selection.

2.4.1 Binarization

In the introductory example presented in the beginning of Section 2 the input data is given in the form of a partially defined Boolean function. In most real-life situations the input data is not necessarily binary and not necessarily numerical. To make such problems amenable to the techniques of LAD, the problems have to be transformed into a binary format. A procedure for implementing this transformation was proposed in [24] and was called *binarization*.

The simplest non-binary attributes are the so-called "nominal" (or descriptive) ones. A typical nominal attribute is "shape", whose values can be "round", "triangular", "rectangular", etc. The binarization of a nominal attribute x can be done as follows. Let $\{v_1, \ldots, v_k\}$ be the set of all possible values of x that appear in the dataset. Obviously, this set is finite, since the number of observations in the dataset is finite. With each value v_i of x we associate a Boolean variable $\alpha(x, v_i)$ such that $\alpha(x, v_i) = 1$ if $x = v_i$ and $\alpha(x, v_i) = 0$ otherwise.

The binarization of numerical attributes is based on the notion of *cutpoints*. Given a set of cutpoints for a numerical attribute x, the binarization of x consists in associating with each cutpoint t a Boolean variable x_t such that $x_t = 1$ if $x \geq t$ and $x_t = 0$ if $x < t$.

In some cases, the choice of cutpoints is suggested by the nature of the attributes (e.g. critical body temperature or blood pressure). In those cases where "critical" values of the attribute are unknown, a typical procedure for assigning cutpoints is as follows.

Let x be a numerical attribute. Since the number of observations in the dataset is finite, x can take only finitely many different values in the set. Let $v_1 < v_2 < \ldots < v_k$ be these values. Clearly, it is sufficient to use at most one cutpoint between any two consecutive values of x. Also, cutpoints below v_1 or above v_k are of no help. Therefore, one can be restricted to cutpoints

of the form $\frac{1}{2}(v_{i-1} + v_i)$. A cutpoint $\frac{1}{2}(v_{i-1} + v_i)$ is called *essential* if there exist both a positive and a negative observation such that in one of them the value of x is v_{i-1}, while in the other $x = v_i$. Obviously, it suffices to use only essential cutpoints in the binarization procedure.

2.4.2 Attribute Selection

Many real-life data sets contain exceedingly many attributes. The binarization procedure can only increase this number. In order to prevent unsurmountable computational difficulties at the pattern generation stage, various techniques reducing the number of attributes have been developed in the literature.

The standard LAD technique of selecting attributes is based on the notion of support sets. A set S of variables is called a *support set* for a partially defined Boolean function f if f has an extension depending only on the variables from S. Clearly the set of all variables is a support set. However, a partially defined Boolean function may have support sets containing not all variables. The task of finding a support set of minimum size admits a formulation as the basic set covering problem, see [23]. A modification of this approach has been proposed in [28]. Some other approaches to attribute selection can be found in [3, 4].

3 Recent Developments in the Theory and Methodology of LAD

In this section, we first review various techniques to generate patterns with the emphasis given to the latest developments (Section 3.1). Then in Section 3.2, we discuss the notion of bi-theory, which was recently introduced to increase the quality of LAD models. Finally, in Section 3.3, we discuss various approaches to apply LAD to non-binary classification problems.

3.1 Pattern Generation

The generation of patterns has always been the central issue in data analysis via LAD. In general, the number of patterns can be very large. Therefore, in practice, the generation is always restricted to patterns satisfying certain criteria. The choice of the criteria is problem-dependent and varies from data to data. These criteria may include specification of the type of the patterns to be generated (e.g. prime or spanned) or specification of some pattern parameters (e.g. small degree).

One of the typical approaches to pattern generation is based on enumeration. The first algorithm of this type was proposed in [23]. It systematically generates all prime patterns of bounded degree, i.e. of a predefined degree D. An accelerated algorithm for the generation of all prime patterns was proposed in [8]. An algorithm for the generation of spanned patterns was developed in [7].

One more approach to pattern generation is based on mathematical modeling. For instance, in [21], the authors propose an integer program for the problem of constructing a maximum α-pattern, i.e. a pattern of maximum coverage which covers a given point α. They also describe two heuristics for an approximate solution of this problem. In [100], the authors propose a Mixed 0-1 Integer and Linear Programming (MILP) approach to identifying LAD patterns that are optimal with respect to various preferences.

Recently, other approaches to pattern generation have been explored in the literature. In [66, 67], several mixed-integer linear programming formulations are proposed to derive prime

p-patterns that define sufficient conditions for a chance constraint in a stochastic programming problem to hold. In [58], the authors describe a genetic algorithm for generating patterns. A probabilistic approach to constructing a maximum pattern covering a given point $\alpha = (\alpha_1, \ldots, \alpha_n)$ was proposed in [29]. It includes a variable x_i in the constructed pattern with probability a/b, where a is the number of positive observations whose i-th attribute equals α_i and b is the total number of positive observations in the dataset. If the variable x_i was chosen for the inclusion in the pattern, it appears in the patter positively (as x_i) if $\alpha_i = 1$, and negatively (as $\overline{x}_i$), otherwise. The authors use this approach to develop a metaheuristic scheme generating a population of near-maximal α-patterns. Some other heuristics for constructing patterns can be found in [9].

A novel approach covering simultaneously two steps of the traditional LAD – pattern generation and model construction – was recently proposed in [18] and then further developed in [31]. It is based on the notion of large margin classifiers and we discuss this approach in the next section.

3.1.1 Large Margin Classifiers

The *separation margin* of a discriminant Δ is the difference between the smallest value that it takes over the positive points that are correctly classified and the largest value taken over the negative points that are correctly classified. More formally, the separation margin is defined as

$$\min\{\Delta(\alpha) \ : \ \Delta(\alpha) > 0 \text{ and } \alpha \text{ is a positive point}\} -$$

$$\max\{\Delta(\alpha) \ : \ \Delta(\alpha) < 0 \text{ and } \alpha \text{ is a negative point}\}.$$

By maximizing the separation margin, one can expect a robust classification of unseen observations. The problem of finding an optimal discriminant was formulated in [18] as a linear program as follows:

$$\begin{array}{lll}
\max & p + n - C \sum\limits_{\alpha} v_\alpha & \\
s.t. & \sum\limits_{k} w_k^+ P_k(\alpha) - \sum\limits_{l} w_l^- N_l(\alpha) + v_\alpha \geq p & \text{for each positive observation } \alpha \\
& \sum\limits_{k} w_k^+ P_k(\alpha) - \sum\limits_{l} w_l^- N_l(\alpha) - v_\alpha \leq -n & \text{for each negative observation } \alpha \\
& \sum\limits_{k} w_k^+ = 1 & \\
& \sum\limits_{l} w_l^- = 1 & \\
& p \geq 0, n \geq 0 & \\
& w_k^+ \geq 0 & \forall k \\
& w_l^- \geq 0 & \forall l \\
& v_\alpha \geq 0 & \text{for each observation } \alpha,
\end{array} \tag{2}$$

where the sum in the objective function is taken over all observations in the data set, and

- P_k and N_l stand for positive and negative patterns, respectively,
- w_k^+ and w_l^- are the weights of the positive and negative patterns, respectively,
- p and n represent the positive and the negative part of the separation margin, respectively,
- v_α is the violation of the separating constraint corresponding to the observation α,

- C is a nonnegative penalization parameter that controls how much importance is given to the violations v_α.

Following [31], we refer to the above problem as master problem (MP). When applied to the set of patterns included in the model, a solution to this problem provides an optimal discriminant function for this set, i.e. finds the weights of the patterns that maximize the separation margin. However, that discriminant function may not be optimal with respect to the entire set of patterns in the generated pandect, or more generally, with respect to the set of all possible patterns. In order to verify global optimality, we need to make sure that there is no pattern that once added to the current set of patterns, allows for an improvement in the value of the objective function.

An approach to finding a global optimum was proposed in [18]. It starts with the initial set of patterns each of which covers exactly one point and solves MP to determine an optimal discriminant for this set. Then with the solution produced by MP the algorithm refers to a pricing subproblem (SP), which provides either a certificate of global optimality of the current discriminant function or a new positive or negative (or both) candidate pattern to be added to the current set of patterns aiming at the improvement of the global solution. The pricing subproblem was further developed in [31] as follows.

Subproblem: pattern generation. We describe the generation of positive patterns, as the generation of negative ones is similar.

First, the algorithm selects a reference observation α that maximizes the total Hamming distance[1] between α and the observations in the opposite class, i.e. it finds a positive observation α which is the most distant from the set of negative observations (ties are broken arbitrarily). Then the algorithm states a Mixed 0-1 Integer and Linear Program (MILP) associated with α. This program assigns weights to observations and by solving it the algorithm finds an α-pattern that maximizes the total weighted sum of observations covered by it. This MILP program is stated as follows:

$$\begin{array}{lll} \max & \sum\limits_{i\in I^+} \frac{1}{\theta^{n_i}} x_i - C \sum\limits_{i\in I^-} z_i & \\ s.t. & (1-b_{ik})y_k \leq 1 - x_i & \forall i \in I^+,\ k \in K \\ & \sum\limits_{k\in K} (1-b_{ik})y_k \geq 1 - z_i & \forall i \in I^- \\ & \sum\limits_{k\in K} y_k \geq 1 & \\ & x_i,\ y_k,\ z_i \ \in \ \{0,1\} & \end{array} \tag{3}$$

where

- I^+ and I^- are the sets of positive and negative observations, respectively, and K is the set of variables (attributes),
- $x_i \in \{0,1\}$ indicates whether or not observation $i \in I^+$ is covered by the constructed pattern,
- $y_k \in \{0,1\}$ indicates whether or not the k-th variable is included in the constructed pattern,
- $z_i \in \{0,1\}$ indicates whether or not observation $i \in I^-$ is misclassified or misplaced,

[1]The Hamming distance between two binary points is the number of components (attributes) where these points have different values.

- $b_{ik} = 1$ if observation i coincides with observation α at the k-th attribute, and $b_{ik} = 0$ otherwise,
- $\frac{1}{\theta^{n_i}}$ is the weight of observation $i \in I^+$, where n_i is the number of times observation i is covered by previously generated patterns and $\theta \geq 1$ is a control parameter (constant). With any real value $\theta > 1$, an observation covered by previously generated patterns have a lower weight (or chance) to be covered by the constructed pattern.
- C is a penalty (constant).

The first constraint guarantees that if $x_i = 1$ then observation $i \in I^+$ is covered by the constructed pattern. The second constraint ensures that the constructed pattern does not cover correctly classified negative patterns. If a negative observation is misclassified, a penalty is introduced in the objective function.

Problem (3) is solved several times (each time with a new not yet covered reference pattern) until all observations are covered. Then for each generated pattern P, the reduced cost is calculated by

$$c(P) = \lambda^+ + \sum_{i \in I^+} \mu_i^+ P(i) - \sum_{i \in I^-} \mu_i^- P(i) \quad \text{if } P \text{ is positive}$$

or by

$$c(P) = \lambda^- - \sum_{i \in I^+} \mu_i^+ P(i) + \sum_{i \in I^-} \mu_i^- P(i) \quad \text{if } P \text{ is negative,}$$

where $\mu^+, \mu^-, \lambda^+, \lambda^-$ are dual variables corresponding to the first four constraints of Problem (2). Only patterns with positive reduced costs are eligible to be added to MP.

Then MP (i.e. Problem (2)) is solved again with the added patterns and the procedure iterates until a stopping criterion is met (for instance, no new candidate patterns are found or the best-saved solution is not improved after a number of iterations). To conclude this section, we observe in [31] the initial set of patterns is generated by means of Problem (3).

3.2 Bi-theories

Let $f = (T, F)$ be a partially defined Boolean function. To build a theory (i.e. to find an extension of f), LAD identifies a number of positive and negative patterns for f. Let us call the disjunction of positive patterns a *positive theory* and the disjunction of negative patterns a *negative theory*. An extension ϕ of f such that ϕ is a positive theory and $\overline{\phi}$ is a negative theory was called in [22] a *bi-theory*.

Example. Consider the pdBf of three variables defined by

$$T = \{(100), (111)\} \text{ and } F = \{(000), (001), (011)\}.$$

It is not difficult to verify by complete enumeration that the set of positive patterns consists of

$$x_1,\ x_1x_2,\ x_1\overline{x}_2,\ x_1x_3, x_1\overline{x}_3,\ x_1x_2x_3,\ x_1\overline{x}_2\overline{x}_3$$

and the set of negative patterns consists of

$$\overline{x}_1,\ \overline{x}_1x_2,\ \overline{x}_1\overline{x}_2,\ \overline{x}_1x_3,\ \overline{x}_1\overline{x}_3,\ \overline{x}_1x_2x_3,\ \overline{x}_1\overline{x}_2x_3,\ \overline{x}_2x_3,\ \overline{x}_1\overline{x}_2\overline{x}_3.$$

Therefore, $\phi = x_1$ is a bi-theory for (T, F), since x_1 is a positive theory by itself and $\overline{x}_1$ is a negative theory by itself. Also, $\phi = x_1x_2 \vee x_1\overline{x}_3$ is a bi-theory, since ϕ is a positive theory consisting of two positive patterns and $\overline{\phi} = \overline{x}_1 \vee \overline{x}_2x_3$ is a negative theory consisting of two negative patterns. It can be shown that there are no other bi-theories for this pdBf.

The notion of bi-theories was introduced in [22] with the objective to provide convincing justifications for classification of each individual point, rather than obtaining a high rate of correct classifications. In other words, the objective is the *a priori justification* of the rules rather than their *a posteriori performance*.

In [22], it was shown that every pdBf has bi-theory extensions. The simplest way of showing this is through the notion of decision trees.

A *decision tree* is a rooted directed graph in which the root has zero in-degree (i.e. there is no arc coming to the root), every non-leaf vertex has exactly two outgoing arcs (left and right) and every leaf has zero out-degree (i.e. there is no arc leaving a leaf). Each non-leaf vertex v is labeled by an index $j(v) \in \{1, 2, \ldots, n\}$ and the leaf vertices are labeled by either 0 or 1.

With each decision tree D one can associate a Boolean function $\phi_D : \{0,1\}^n \to \{0,1\}$ as follows. Let $x = (x_1, \ldots, x_n)$ be a binary vector. Starting from the root, we move from vertex to vertex, always following the left arc out of v if $x_{j(v)} = 0$, and the right arc otherwise, and stop when we arrive at a leaf, in which case we say that x is classified into this leaf. The label of the leaf defines the value of $\phi_D(x)$.

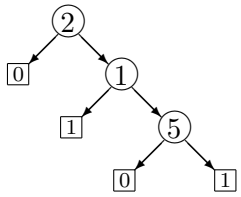

Figure 1: An example of a decision tree

Given a pdBf $f = (T, F)$, we say that a decision tree defines an extension of f if ϕ_D is an extension of f. Also, we say that a decision tree is *reasonable* for f if

- D defines an extension for f,
- for every leaf of D, at least one vector in $T \cup F$ is classified into the leaf.
- for every non-leaf vertex v, at least one vector from T is classified into a descendant of v, and at least one vector from F is classified into another descendant of v.

The importance of reasonable decision trees for partially defined Boolean functions is due to the following theorem proved in [22].

Theorem 1. *Let $f = (T, F)$ be a pdBf and D a reasonable decision tree for f. Then ϕ_D is a bi-theory of f.*

Many of the classical decision tree building methods yield reasonable trees. In particular, the following generic algorithm does this job. In the description of the algorithm, we use the following notation: if $j \in \{1, 2, \ldots, n\}$ and $i \in \{0, 1\}$, then T_j^i is the set of true points where the j-th variable equals i and F_j^i is the set of false points where the j-th variable equals i.

Algorithm $\mathcal{A}$
Input: a pdBf (T, F)
Output: a decision tree

1. Create root node v of the tree.
2. If $T = \emptyset$, then mark v as 0 and return v.
3. If $F = \emptyset$, then mark v as 1 and return v.
4. If $T \neq \emptyset$ and $F \neq \emptyset$, then choose a variable j which is not a constant, mark v by j and return v together with $\mathcal{A}((T_j^0, F_j^0))$ as a left subtree and $\mathcal{A}((T_j^1, F_j^1))$ as a right subtree.

Every leaf in a reasonable decision tree corresponds to a pattern in the bi-theory defined by this tree: a leaf labeled by 1 corresponds to a positive pattern and a leaf labeled by 0 corresponds to a negative pattern. This pattern can be constructed by reading the variables assigned to the non-leaf vertices on the unique path connecting the root to the leaf. If a non-leaf vertex is left through the right arc, the respective variable appears in the pattern positively, otherwise it appears negatively (negated). For instance, the decision tree in Figure 1 gives rise to two positive patterns $x_2\overline{x}_1$ and $x_2x_1x_5$ and two negative patterns $\overline{x}_2$ and $x_2x_1\overline{x}_5$.

It is important to note that for every partially defined Boolean function the number of bi-theories is typically larger than the number of reasonable decision trees. To give an example, consider the pdBf consisting of two positive observations (1100) and (0011) and three negative observations (1010), (0101) and (0000). It is not difficult to check the function $f = x_1x_2 \vee x_3x_4$ is a bi-theory, for which $\overline{f} = \overline{x}_1\overline{x}_3 \vee \overline{x}_1\overline{x}_4 \vee \overline{x}_2\overline{x}_3 \vee \overline{x}_2\overline{x}_4$. Moreover, the set of positive points of f cannot be covered with fewer then 2 terms and the set of negative points of f cannot be covered with fewer then 4 terms. Therefore, any reasonable decision tree representing f should contain at least 6 leaves. On the other hand, for every leaf in a reasonable decision tree there must exist an observation classified into this leaf, and hence the total number of leaves in any reasonable decision tree for this pdBf does not exceed 5. Therefore, f is a bi-theory that cannot be represented by a reasonable decision tree.

3.3 Multi-class LAD

Let us repeat that originally LAD has been developed to solve binary classification problems, i.e. problems where the data consists of two classes (positive and negative observations). However, in real life the data frequently comprises more classes, for instance, different types of a certain disease.

There are two basic approaches to transform a multi-class classification problem into a collection of binary classification problems. One of them is known as One-vs-One (OvO) and the other as One-vs-All (OvA).

Given a data with K classes, the OvO approach solves a binary classification problem for each pair of classes, of which there are $K(K-1)/2$. For each pair ij it builds a classifier f_{ij} (numerical discriminant) and then it classifies a new observation x according to the following function

$$f(x) = \arg\max_i \sum_{j \neq i} f_{ij}(x),$$

where $f_{ji} = -f_{ij}$.

The idea of the OvA approach is to separate each class of observations from the remaining $K-1$ classes. Thus, for each class C_i it builds a binary classifier f_i (numerical discriminant) separating the observations in the class C_i (positive observations) from the rest of the data (negative observations). Then a new observation x is classified by

$$f(x) = \arg\max_i f_i(x).$$

Each of these two approaches has advantages and disadvantages, but both of them diminish the explanatory power of LAD. To overcome this difficulty the authors of [59] propose the following hierarchical approach.

In each level of the hierarchy, the approach separates one class of observations from the remaining classes and then proceeds with the remaining classes inductively. Figure 2 illustrates this idea with a data consisting of four classes C_1, C_2, C_3, C_4. First, the algorithm separates class C_1 reducing the problem to the data consisting of three classes C_2, C_3, C_4. Then it separates class C_2, which reduces the analysis to a binary classification problem for classes C_3 and C_4.

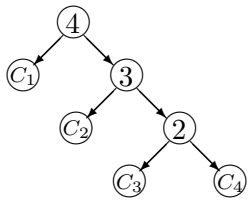

Figure 2: A hierarchical approach

In order to distinguish a class of observations to be separated, the algorithm generates OvA-type patterns for each class under consideration (i.e. patterns covering observations only of the given class) and then separates the class minimizing the following expression:

$$w_1|P^C| - w_2 AR(P^C) + w_2 AG(P^C),$$

where $|P^C|$, $AR(P^C)$ and $AG(P^C)$ are, respectively, the number of patterns for class C, their average coverage and average degree. Also, w_1, w_2, w_3 are the weights of the corresponding parameters (can be determined using a sensitivity experiment). This form of distinguishing one pattern from the rest of the data was derived by the authors of [59] empirically. It can be explained by the fact that a small number of simple (low degree) patterns of high coverage describing a class of observations suggests a specific role of this class within the data and justify its separation from the remaining observation.

4 Applications of Logical Analysis of Data

The LAD methodology has found numerous applications across multiple fields. In the present section, we highlight some recent ones.

4.1 Industrial Applications of LAD

The LAD methodology has been widely applied in engineering to understand, detect, and predict physical phenomena, such as the occurrence of faults and the equipment aging process.

These phenomena usually have drastic consequences on user's safety, environment's protection, energy and natural resources' consumption, operational costs and efficiency. Besides detecting and predicting, it is essential to be bale to explain the physics behind these phenomena. As such, the LAD methodology, and specifically the explanatory power of its patterns, makes it a unique approach to engineering data analysis since the patterns can usually be linked to physical phenomena that are hidden in the data and that need to be explained based on scientific evidence and knowledge. In the this section, we review some engineering applications, in which LAD was applied and we pay special attention to present how LAD's patterns were used in order to exploit the experts' knowledge about the physical phenomena at hand.

4.1.1 Fault Detection and Diagnosis with LAD

Rogue components are known in the airline industry as repairable components which repeatedly exhibit failure modes that cannot be detected because they are outside the scope of the standard repair and overhaul procedures. As such, their presence causes havoc and has a negative impact on asset management programs, since they keep circulating in the system, without any possible way to detect them. In [78], LAD was used to detect and isolate rogue components in airplanes. By monitoring certain performance indicators and expert system's knowledge, patterns unique to rogue components were discovered. Data were extracted from the maintenance records of 61 airplanes during the period stretching from March, 1999 to June, 2009. An observation consisted of the reason-for-removal codes and the time-to-removal codes for the last 3 years, as well the manufacturer's identifiers. Experts were asked to tag each observation as rogue or non-rogue. LAD was used in order to find patterns that distinguish these two classes of turbo compressors The results show a high quality of classification ranging from a minimum of 82.26% to a maximum of 99.65%. A key result in this application was the conservation of the human expert knowledge in an automated way.

Shaban *et al.* proposed a process control technique applicable to the routing process for carbon fiber reinforced polymer (CFRP), which is a composite material used in the aerospace industry [103]. LAD was used to evaluate and to control the quality of the machined parts by monitoring some machining features and parameters. Unlike most pattern recognition techniques, LAD generates, not only positive patterns, but also negative patterns. The positive ones were used to detect the tool wear status up to a certain threshold, and the negative ones were used in an adaptive control loop in order to move away from the conditions that lead to defective products to the conditions that will return the process back to normal conditions and to the production of conforming products. This mechanism was evaluated for online control of a simulated routing process of CFRP developed using the patterns found off-line and applied to the high speed routing of woven carbon fiber reinforced epoxy.

Shaaban *et al.* proposed a tool wear monitoring and alarm system based on LAD [104]. It is a non-intrusive online system that measures the cutting forces and relates them to tool wear through a set of patterns. The main objective is to avoid producing defective products due to tool wear by raising an alarm at the right time. The system deals with external and internal factors that affect the machining process. The proposed system was validated with data obtained under different machining conditions of turning titanium metal matrix composites. The alarm limit obtained by using LAD's patterns was compared to the one obtained from a common statistical method: the proportional hazards model. The results showed that the proposed LAD alarm system detects the worn patterns and gives an additional 40% accuracy in the detection of worn patterns that initiate an alarm signal in order to replace the cutting tool at an age that

is relatively closer to the actual failure time.

Mortada *et al.* proposed an approach for automatic diagnosis of faults in rolling bearings by using a modified LAD pattern generation method [79]. The vibration signals were acquired by accelerometers which collect reading every few seconds and were used for the detection of bearing faults at an earlier stage of the crack propagation. Since the vibration signals are not labeled as faulty and no-fault, a visualization procedure was used in order to observe the point in time at which the cracks seem to begin. Up to this stage, the data point was labeled as non-faulty, and after it the data point was labeled as faulty. Different experiments were done in order to analyze the effect of leaving an unlabeled interval between the faulty and non-faulty data in order to obtain better classification accuracy. LAD was compared to SVM and neural networks and was shown to outperformed those in terms of accuracy.

Mortada *et al.* developed a multi-class LAD classifier to diagnose faults in power transformers [80]. The objective was to design a tool for the detection and identification of faults in the power transformers by using dissolved gas analysis data. In that work, an extension of two-class LAD to multi-class applications was proposed using a One-vs-One technique. This technique has the advantage to generate a less complex decision model which has a better execution time. As a result of that research, the software cbmLAD [36] was developed to deal with multiple faults diagnosis.

Shaban *et al.* presented a new unsupervised multi-class detection method to deal with machining applications [105]. Using experimental data and experts' opinions, it was shown that the tool wear degradation increases according to five stages determined by the Douglas-Peucker algorithm [37]. After this step, LAD was used to generate patterns that characterize each class of wear. The generated patterns fulfill the double objective of detecting the present tool wear class based on the recent sensors' readings of the time-dependent machining variables and deriving novel information about the inter-correlation between the tool wear and the machining variables. The results showed that the proposed method detects the tool wear class correctly and with high accuracy.

Jocelyn *et al.* [57] applied LAD in the occupational health and safety filed, and, in particular, to characterize different types of machinery-related accidents and to relate them to the root causes of faults. The data comprises classes of observations representing the types (maintenance-related and production-related) of accidents. Each observation is a vector of indicators values recorded at the time the accident occurs. These indicators describe the accident's conditions such as the categorical variables "Presence of safeguarding" (yes or no) at the time of the accident and "Worker's time in current position" (0-4 or 5-10). The information provided by LAD was used in a logical way to prioritize risk factors, which help safety practitioners make decisions regarding machines' safety measures.

Ragab *et al.* have recently applied LAD to diagnose faults in complex industrial chemical processes [92]. Two case studies exemplify the importance of the interpretability of the LAD patterns. The first case study is the Tenessee Eastman process, which is a well-known benchmark problem in chemical engineering and the second one is a black liquor recovery boiler in the pulp and paper industry. In both cases, LAD was capable of dealing with the highly correlated and nonlinear effects of the variables, which is typical in most industrial chemical processes datasets. The extracted patterns were found useful for the boilers operator wherein LAD can detect a certain fault and relate it to the causes of its occurrence.

4.1.2 Fault Prognosis with LAD

LAD was used to predict the estimated time to failure and the remaining useful life (RUL) of equipment working under different operating conditions and subjected to either single or multiple failure modes [42, 93, 94].

Ragab *et al.* developed a reliability-based prognostic methodology to predict the health states of an equipment, based on the lifetime and condition monitoring data [93]. Their method uses the condition-based maintenance (CBM) data collected just before the occurrence of complete failure in the equipment. LAD was used as an event-driven diagnostic technique and merged with Kaplan-Meier (KM) estimation. The key idea of merging LAD to KM was to reflect the effect of the operating conditions on the probability of survival of the monitored equipment. Knowledge is extracted from the lifetime and the condition monitoring data, in the form of non-parametric survival curves. LAD extracts the knowledge in the form of patterns, while KM estimates the baseline survival curve that reflects the effect of aging, based on the observed historical lifetime data. A survival curve was estimated for each pattern based on the failure time of the equipments covered by this pattern. Given a new observation collected from the equipment, the baseline survival function estimated using KM is updated with the diagnostic information obtained from the LAD decision model. The updated function is then used to estimate the failure time and the RUL of the monitored equipment. The performance of the estimated RUL was measured in terms of the difference between the predicted and the actual RUL of the monitored equipment. The methodology was validated and compared with the Cox proportional hazard model on the turbofan degradation dataset available at NASA prognostic data repository [102].

Ragab *et al.* proposed a methodology for multiple failure modes prognostics in rotating machinery in [94]. The methodology merges multi-class LAD with a set of non-parametric cumulative incidence functions. It is based on condition monitoring data collected from a system that experiences several competing failure modes over its life span. The objective is to predict the RUL while considering the possible interaction between the failure modes that are resulting from the failures of different components in the overall system. The explanatory power of LAD's generated patterns was used, not only to classify new collected information, but also to identify the interactions between different failure modes through the appearance of patterns of different classes of failure mode simultaneously. The prognostic methodology was validated using vibration data collected from bearing test rigs in the industry. To train the multi-class LAD classifier, five time-domain features and ten wavelet-based features were extracted from each collected vibration signal. The comparison showed that the method is capable of estimating accurately the RUL of an individual system in the presence of multiple failure modes.

A prognostic methodology that exploits all condition monitoring data collected from a group of systems during their life spans, both normal and failure observations, was proposed in [94]. This methodology captures the effect of the instantaneous conditions on the health state of the monitored system. It uses pattern selection procedure to select a set of significant patterns. A survival curve is estimated for each subset of observations covered by each selected pattern. A weight that reflects the coverage (i.e., its importance) of each pattern is assigned to its survival curve. Individualized survival curves are formed for each new system over time. This curve captures the most recent conditions of each system, and replace the generalized KM curve that represents the failure time of many similar systems by a survival curve based on the exploitation of LAD's generated patterns.

4.1.3 Other Engineering Applications

LAD was applied in the airline industry in [38] to estimate overbooking by predicting the show rates of passengers. The objective was to detect sets of patterns that differentiate passengers with high and low show rate. Each observation represents a passenger characterized by a set of attributes, such as gender, day of the week, itinerary origin, number of passengers, etc. The LAD model classifies passengers as show and no-show. The proposed LAD method was compared to Air Canadas current tool for overbooking forecasts, which is based on historical statistics. The results showed that the LAD prediction model is very competitive.

A multi-class LAD version was used in for face recognition purposes [96]. LAD was linked to image preprocessing techniques based on the Eigenfaces and Fisherfaces. An extension of this study to deal with multiple changes in facial expressions was proposed in [97]. The results showed that LAD improves the classification of Eigenfaces and Fisherfaces with minimum error rate, and outperforms other face recognition techniques.

The explanatory power of LADs patterns was exploited in supply chains management [76]. The aim of that work was to optimize the inventory management process through a multi-criteria ABC inventory classification method and to set policies and rules to avoid financial losses and customers dissatisfaction. The main concept was to correct the familiarity bias in experts opinions, which is defined as a bias that exists naturally in every humans judgment due to previous experience. LAD was used to identify and to correct the bias in the ABC items classification done by inventory experts. Databases were analyzed with a LAD-based classification technique to study the impact of such bias on inventory management performance. LAD was shown capable of correcting inconsistencies and biases through the interpretability of its generated patterns. If an item is wrongly classified, the patterns covering this item are analyzed. Logical reasons for its misclassification, possibly due to the familiarity bias in experts' opinions, were seeked. It was shown that LADs patterns were capable of correcting inconsistencies and biases, thus resulting in more accurate inventory classification performance that increases from an average of 63% to an average of 93%.

4.2 Medical Applications of LAD

LAD was used successfully in the medical field in order to diagnose patients's condition and to predict the propagation of some diseases. In particular, it was applied to breast cancer diagnosis [62] and breast cancer prognosis [1], ovarian cancer detection [5, 87], coronary risk prediction [6, 65], early diagnosis of acute ischemic stroke in [98], identifying survival patterns for clear cell renal cell carcinoma [25].

In [6], the risk of death was estimated for the two groups of patients who died or who survived during a 9-year follow-up period. LAD provided a function called prognostic index. The value of the prognostic index was shown to be closely correlated with the patients' risk of death. The prognostic index is also shown to outperform the widely used Cox Score, which is the indicator used by most cardiologists, even though no statistical distribution assumptions where made.

The methodology called Logical Analysis of Survival Data (LASD) was proposed in [63] to identify survival patterns and to build a survival function. Each survival pattern can cover only a proportion of observations in the dataset. A group of patterns can be combined in an infinite number of ways to construct the survival model for the entire dataset. The performance of LASD was compared with survival decision trees and KM estimator. The empirical study showed that LASD is an accurate prognostic tool since the confidence intervals of the predictions are very small. These confidence intervals also indicate the robustness of the resulting LASD model.

Yacout *et al.* apply LAD in medical diagnosis to model the nonlinear and dynamic causal relationships between three clinical procedures (i.e., blood transfusion, surgery and organ transplant) and Alzheimer's disease (AD) in [110]. The goal was to develop a better understanding of the effect and causality in order to prevent and treat this disease. LAD was used to find the effects (if any) of the clinical procedures on the risk of AD, or conversely, the effect of AD on clinical procedures. A group of twenty-five risk factors, including the clinical procedures (e.g., transplants cell, or organ or tissue, age, gender) done before and after diagnosis, are considered. The results showed that there is no evidence of relation between blood transfusion, surgery or organ transplant on the onset or the development of Alzheimer's disease.

The Logical Analysis of Data was also used in [26] to analyze computed tomography data in order to distinguish between three types of idiopathic interstitial pneumonias (IIPs):

- Idiopathic Pulmonary Fibrosis (IPF),
- Non Specific Interstitial Pneumonia (NSIP),
- Desquamative Interstitial Pneumonia (DIP).

The dataset consisted of 56 patients (observations), including 34 IPFs, 15 NSIPs, and 7 DIPs cases, and involved 13 variables (attributes), 10 of which were binary.

In order to distinguish between the three types of IIPs, three LAD models are developped: IPF vs non-IPF, NSIP vs non-NSIP and DIP vs non-DIP. In particular, the first model consists of 20 positive and 20 negative observations and allows the accurate classification of IPF/non-IPF patients. The NSIP/non-NSIP model was built on the support set of 8 attributes, and includes 16 positive and 4 negative patterns. The DIP/non-DIP model is built on the support set of 6 attributes, and includes 7 positive and 15 negative patterns.

The three LAD models correctly classify 54 of the 56 patients. In view of the suspicions related to the remaining two observations, the medical records of these two patients have been re-examined. It was found that one of them, which appears in the data as a DIP patient, was exposed to asbestos, and therefore its classification as DIP is uncertain. Asbestosis may be responsible for a pathologic aspect similar to that of IFF, but very different from DIF. It is also possible that the pathologic result on the biopsy of a very small area of the lung was wrong. Similarly, it was found that the data of the second patient unclassified by LAD are highly atypical in all the features (age, clinical data and lung pathology). Based on the clinical, radiographic and pathologic data, this patient does not seem to belong to any of the three classes in the initial classification, and it was suggested that in view of these reasons, the patient should be considered non-classable and removed from the dataset.

The LAD has been also applied to reveal, for the first time, a correlation between the chemical structures of poly(β-amino esters) and their efficiency in transfecting DNA [45], predicting secondary structure of proteins [16, 17], and for establishing morphologic code [77].

4.3 Applications of Logical Analysis of Data in Finance

In this section, we first discuss how LAD can be employed in the credit risk industry [48, 49, 50, 60]. The LAD method was utilized to reverse-engineer and construct credit risk ratings that reflect the creditworthiness of countries and financial institutions. Second, we describe how LAD was used in the international finance area, and in particular in identifying supply chain factors that can play a key role in attracting foreign direct investments [12].

4.3.1 Credit Risk Rating and Ranking Systems

LAD was applied in the area of credit risk ratings to two types of obligors: financial institutions and countries.

Country Risk Ratings. Country risk is defined as the "risk of national governments defaulting on their obligations" [39] and reflects a government's ability and willingness to repay its public debt on a timely fashion. Country risk ratings play a fundamental role on the interest rates at which countries can obtain credit and impact the credit risk ratings of domestic banks and companies. In developing economies, a firm is indeed unlikely to receive a rating higher than the rating of the country where it operates, which is known as the "sovereign ceiling effect".

The LAD method has been successfully applied to induce a credit risk system from a set of country risk rating evaluations [48, 49, 60]. Two LAD methods, each using nine economic and three political explanatory variables, have been developed to construct countries' credit rating systems. Both methods involve three common steps:

- The construction of pseudo-observations for every pair of countries, which allow for a comparative creditworthiness characterization of the two countries.
- The construction of an LAD model, which takes the form of a weighted sum of combinatorial patterns, from the set of pseudo-observations involving pair of countries.
- The derivation of relative preferences, which provides an assessment of how "superior" the creditworthiness of one country in the pair is over that of the other country.

The two methods differ in the way they use and extend the relative preferences to form the rating system. We detail below the challenges raised by the rating of countries on the basis of their creditworthiness and how the LAD implementation presented in [48] overcomes them.

From pairwise country comparisons to pseudo-observations

Due to the limited number of countries and thus of usable data points, which severely restricts the application of standard econometric methods, Hammer *et al.* [48, 49] have examined the *relative riskiness* of one country compared to another one, rather than modeling the riskiness of each individual country, and have "transformed"' the original observations describing the 69 countries into a set of 2346 pseudo-observations or pairs-of-countries observations. The original dataset describes the creditworthiness of each country $i \in I = \{1, \ldots, 69\}$ with a 13-dimensional vector C_i, whose first component is the country risk rating given by Standard and Poor's, while the remaining 12 components specify the values of the nine economic/financial and three political variables. For every pair of countries $i, j \in I$, a pseudo-observation P_{ij} is constructed, providing a comparative description of the two countries. The pseudo-observations are also 13-dimensional vectors. The first component is an indicator which takes the value 1 if the country i in the pseudo-observation P_{ij} has a higher rating (i.e., lower risk) than the country j, takes the value 1 if j has a higher rating than i, and takes the value 0 if the two countries have the same rating. The other components $k, k = 2, \ldots, 13$ of the pseudo-observation $P_{ij}[k]$ are obtained simply by taking the differences of the corresponding components of C_i and C_j:

$$P_{ij}[k] = C_i[k] - C_j[k] \ . \tag{4}$$

We note the similarity between the pseudo-observation concept in LAD and the pairwise comparison table concept in dominance-based rough set theory [43, 44].

The fundamental idea behind the relative riskiness approach is that a rating system can be essentially reconstructed from the knowledge of the relative standings of all pairs of countries, which is in line with the argument that "credit ratings express risk in relative rank order, which is to say they are ordinal measures of credit" [40]. An additional advantage of transformation (4) is that it allows us to avoid the problems related to the small size ($|I|$) of the original dataset. While the larger set of pseudo-observations is obviously not independent, since $P_{hi} + P_{ij} = P_{hj}$, the non-independence of pseudo-observations does not create any problems for LAD, which contrasts with traditional econometric methods.

From pseudo-observations to relative preferences

The LAD method is used as a large margin classifier to the set of all pseudo-observations P_{ij}, which correspond to pairs of countries i and j with different ratings. Each pseudo-observation P_{ij} is classified as positive or negative, according to the value of the indicator variable, i.e., depending on whether i is rated higher than j. The application of LAD to the set of pseudo-observations provides an LAD model constructed as a weighed sum of 320 patterns. A discriminant $\Delta(P_{ij})$ (1) is the computed for each pseudo-observation P_{ij}. The values of the discriminant are called the relative preferences and form the relative preference matrix.

From relative preferences to a partial order on the set of countries

A naive approach to deriving country ratings from relative preferences would rely on the direct interpretation of their signs as indicators of rating superiority. Hammer *et al.* [48] have relaxed the overly constrained search for country ratings whose pairwise orderings are in precise agreement with the signs of relative preferences, to the more flexible search for a partial order on the set of countries, which approximates well the set of relative preferences. They have defined a strengthened version of it, called *dominance relationship*, which, besides the sign of the relative preference $\Delta(P_{ij})$, also accounts for the values of the relative preferences of each of these two countries i and j over every other country $k \in I$.

Let $S_{ij}[k] = \Delta(P_{ik}) - \Delta(P_{jk})$ define the external preference of country i over j with respect to k, $S_{ij} = \frac{\sum_{k \in I} S_{ij}[k]}{|I|}$ define the average external preference of i over j, and $\sigma_{ij} = \sqrt{\frac{\sum_{k \in I} (\Delta(P_{ik}) - \Delta(P_{jk}) - S_{ij})^2}{|I|}}$ be the the standard deviation of the external preference of i over j. The dominance relationship of i over j is defined with two conditions. The first one stipulates that the relative preference of i over j must be positive. The second one requires that the external preference of i over j must be positive at a certain confidence level. The level of confidence is parameterized by the multiplier $\nu > 0$ of the standard deviation σ_{ij}. More formally, a country i is said to dominate another country j if:

$$\Delta(P_{ij}) > 0 \quad \text{and} \quad S_{ij} - \nu\sigma_{ij} > 0 \; . \tag{5}$$

If the sign of the above two relationships is reversed, then i is said to be dominated by j. In all other cases countries i and j are said to be not comparable due to lack or conflicting evidence about the dominance of i over j. Hammer *et al.* [48] have devised a procedure that assigns to ν the lowest possible value, thereby maximizing the number of comparable country pairs, such that the dominance relationship is transitive.

From partially ordered sets to extreme linear preorders

The dominance relationship represents faithfully the extracted information about country preferences. However, the large amount of data needed to describe a partial order makes its

use impractical. As country ratings provide a compact way of expressing country preferences since they constitute a special type of partial orders called linear preorders, Hammer *et al.* [48] have designed an approach based on the Condorcet method to transform the logical dominance relationship into linear preorders which preserve all the order relations between countries (i.e., which constitute extensions of the partial order), and are as close as possible to it. Two extreme linear preorders, called the optimistic and pessimistic extensions, are derived. The former (resp., later) is such that it assigns to each country the highest (resp., lowest) level it can expect. The construction of these two preorders is based on the concepts of weak Condorcet winners and losers.

The second LAD method for country risk ratings [49] also uses the relative preferences, but in a very different way, applying multiple linear regression to generate ratings, called logical rating scores, which are numerical values whose pairwise differences approximate optimally the relative preferences over countries as expressed in their risk ratings.

Validation tests show that the two types of LAD rating systems: (i) avoid overfitting issues, (ii) correlate highly with those of the main rating agencies (Standard & Poor's, Moody's, The Institutional Investor), and (iii) are stable, having an excellent classification accuracy when applied to the following years' data. Additionally, the rating systems distinguish themselves from the rating models in the literature by their self-contained nature, i.e., by their non-reliance on any information derived from lagged ratings. This feature makes possible to use them to assess the creditworthiness of not-yet-rated countries and shows that the high correlation between predicted and actual ratings cannot be attributed to the reliance on lagged ratings. The two studies also provide new insights on the importance of variables by supporting the inclusion, in addition to economic variables, of political variables (i.e., political stability), and by identifying the variable "financial depth and efficiency" as a new critical factor in assessing country risk.

Credit Risk Ratings of Financial Institutions. Central banks are afraid of widespread bank failures since they could amplify cyclical recessions and result in severe financial crises. The credit risk rating of a bank can be viewed as the "bank's intrinsic safety and soundness" [81]. The evaluation of the creditworthiness of banks is challenging given the opaqueness of financial institutions. Part of the difficulty is due to the volatility of the credit risk ratings of banks which is significantly higher than it is for corporations and countries.

Using an absolute creditworthiness perspective, Hammer *et al.* [50] employ LAD to reverse-engineer the Fitch bank credit ratings using a set of fourteen financial variables (loans, other earning assets, total earning assets, non-earning assets, net interest revenue, customer and short-term funding, overheads, equity, net income, total liability and equity, operating income), nine representative financial ratios (ratio of equity to total assets, net interest margin; ratio of interest income to average assets; ratio of other operating income to average assets; ratio of non-interest expenses to average assets; return on average assets; return on average equity cost to income ratio; ratio of net loans to total assets), and the S&P risk rating of the country where the bank is located. The core of the LAD model is composed of patterns allowing for the separation between banks with high credit risk ratings and those with low ones. The model is very parsimonious and comprises only twenty-two patterns, defined with respect to at most three of the explanatory variables. The LAD model and its patterns are then used to compute discriminant values from which an accurate bank rating system is extracted. A convex optimization problem is used to map the numerical values of the LAD discriminant to the nine bank rating categories $(A, A/B, \ldots, E)$ that are used by Fitch Ratings. The solution of the optimization problem partitions the interval of the discriminant values into nine sub-intervals that are associated

to the nine rating categories. The numerical evaluation of the model shows that the LAD ratings are in very close agreement with the Fitch ratings and cross-validate very well. The combinatorial nature of the LAD method permits to discover the interactions between small groups of explanatory variables on the bank ratings. This analysis suggests the importance and predictive power of the country risk rating, return on average assets, and return on average equity variables.

4.3.2 Supply Chain Determinants of Foreign Direct Investments

The LAD method was used to ascertain whether supply chain variables – and if yes which of those – play a role to attract foreign direct investments (FDI) in a country [12]. The three supply chain variables used as explanatory variables to build the LAD model and construct combinatorial patterns are:

- supply infrastructure,
- absorptive capacity,
- supply environment, which is itself decomposed into four dimensions: (i) buyer sophistication, (ii) local supplier quantity, (iii) local supplier quality, and (iv) local availability of components and parts.

Since supply infrastructure, supply environment, and absorptive capacity are closely intertwined and their joint effect is unlikely to be linear, the use of the combinatorial-based LAD method is particularly suitable to capture the possible individual as well as the combined impact of these variables on the FDI potential of a country. In particular, the LAD methodology employs these variables to learn the UNCTAD classification of countries in terms of FDI potential. An LAD model is derived, allowing for the construction of a preorder of countries in terms of FDI potential and the development of a rating system that evaluates the countries' attractiveness with respect to FDI. The rating model can be developed with various granularity levels. Multinational corporations can use the model to decide where to invest, while developing countries can use it to determine their supply chain and logistics development needs. The patterns in the LAD model provide a compact representation of the supply chain conditions that affect the potential of a nation to attract FDI. They highlight the criticality of developing a strong supply infrastructure and the interactions between these variables. For example, it is shown that a lower level of supply infrastructure can be offset by proposing strong supply environment and absorptive capacity.

4.4 LAD for Stochastic Optimization

LAD was also applied to solve several classes of stochastic optimization problems, i.e., chance-constrained (probabilistic) programming and simulation-based optimization, that, at first tight, do not seem to have much commonality with LAD.

Within this area, LAD was first applied to improve the solution of simulation-optimization problems [70]. The primary objective was to evaluate the performance of a system running under unknown values taken by its stochastic parameters. A number of simulations with very few replications were carried out and the mean value of the directly measurable quantities, called observables, were recorded. These observables were then used as inputs in an LAD-classification model that produces a prediction of the performance of the system. An application to a specific assemble-to-order production line was presented in details in [70]. A crucial challenge

in simulation-optimization concerns the allocation of the computational resources between the search for a better solution and the evaluation of the current candidate solution. In this study, a highly accurate LAD-classification model was derived to enhance the local heuristic search for better candidate solutions. The model permits to shrink the search space for the heuristic by rejecting quickly all settings not classified as good. As the time for computing the classification of a setting is a fraction of the time to get the estimated performance of the setting, precious computing time can be saved during optimization.

The main type of stochastic problems in which LAD was used is the class of probabilistically constrained stochastic programming problems that require a system of stochastic inequalities to be jointly satisfied with a prescribed probability level p [85, 86]. LAD, and its Boolean programming and combinatorial pattern foundations, have been instrumental to develop (see [66, 67]) a new reformulation and solution method for probabilistically constrained optimization problems. In this setting, the patterns provide a compact representation of sets of conditions that are sufficient for the satisfaction of a probabilistic constraint. The method involves the binarization of the probability distribution using a set of appropriately chosen, consistent cut points, which in turn permits the construction of a partially defined Boolean function (pdBf) representing the satisfiability of the chance constraint [66]. The pdBf is then extended as a disjunctive normal form (DNF), which is a collection of combinatorial p-patterns, each defining sufficient conditions for a probabilistic constraint to hold. Using the properties of threshold Boolean functions and the concept of (tight) minorant [61], mixed-integer programming models are derived allowing for the concurrent generation of p-patterns and the solution of the deterministic reformulation of the stochastic problem. The mathematical models obtained with this Boolean/LAD approach can be exact reformulations or inner approximations of the chance-constrained problem, and can handle chance constraints

- with random right-hand side vector and system of linear stochastic inequalities [66, 67].
- with random technology matrix (i.e., the matrix of coefficients multiplying the decision variables is stochastic) and system of linear stochastic inequalities [61].
- with random right-hand side vector and technology matrix and system of nonlinear stochastic inequalities [71].

Extensions to multi-objective probabilistically constrained programming problems have also been proposed in [73]. Chance-constrained optimization models with both endogenous and exogenous sources of uncertainty were most recently proposed in [72]. Applications of the LAD-inspired reformulation method for probabilistically constrained stochastic programming problems have been used in disaster management [55], forestry management [68, 69], finance [56, 73], and evacuation of severe casualties [72].

5 Evaluation and Extensions

In this section, we first discuss LAD with respect to two measures of performance - accuracy and computational time - are generally used in order to evaluate the performance of machine learning techniques. We then provide a comparative analysis of LAD with respect to some of the most popular data analysis methods (i.e., support vector machine, rough set theory, decision tree, artificial neural network).

5.1 Performance Measures

Accuracy. Some computational tests have showed that the accuracy of the LAD approach compares favorably to many machine learning techniques. It was shown [23] that LAD is competitive with the well-established classification methods, such as decision tree, machine-Learning regression models, and artificial neural networks. In [53], the authors use well-known datasets to compare the performance of their LAD-based algorithm and report that it improves on the best results obtained with some well-established supervised learning approaches. In [59], the performance of multi-class LAD was compared to other machine learning approaches, namely, directed acyclic graph of support vector machine, unbalanced decision tree-based support vector machine, sequential minimal optimization, neural network, decision tree, and a naïve Bayesian classifier, available through the Weka software package. The results showed that the LAD accuracy is equal to or higher than the one obtained with each of these other approaches. In [29], the authors apply their LAD-based algorithm to ten datasets available on the University of California Irvines (UCI) machine learning repository. They compared the accuracy of their algorithm with the best known result obtained with one of five commonly used machine learning algorithms, namely, support vector machine, C4.5 decision tree, random forest, multilayer perceptron, and logistic regression, which are all available within Weka. They report that the LAD approach is competitive in term of classification accuracy with the best results obtained.

Computational Efficiency. Few studies report on the computational efficiency of LAD-based techniques. One reason may reside in the fact that for off-line training, the computational efficiency is not as critical as the accuracy. However, due to the increasing importance of using machine learning algorithms capable of dealing with Big Data, some recent LAD-based algorithms are evaluated based on accuracy and computational time as well. For example, ten LAD-based algorithms are compared in [31] using fourteen datasets from the UCI machine learning repository. It appears that the reported computational times depend heavily on the properties of the datasets. For example, classes separability, number of observations, and homogeneity between observations in a same class impact strongly the computational times which vary from 300 to 44000 seconds. In our opinion, improving the computational time of LAD is an important area that needs further research. To our knowledge, parallel implementation of LAD techniques has only been reported very recently in [31, 111]. The existence of open-source platforms for parallel computing (e.g., MapReduce and Spark) paves a relatively easy way to the parallelization of LAD algorithms. As above-mentioned, the LAD approach is based on the four key steps, i.e., binarization, features selection, pattern generation, and theory formation. In a parallel computing paradigm, each step is an area calling for further research. The pattern generation step is the building block of LAD approach and is the most challenging and promising one for parallel computing. LAD is based on Boolean logic and combinatorial optimization and many LAD-based pattern-generation algorithms require the solution of some set covering problem, which is known to be an NP-hard problem. Other pattern generation algorithms are based on the solution of other types of mixed-integer linear programming problems, which are computationally challenging too.

Other directions that deserve attention is the exploration and exploitation of LAD algorithms based on systems of homogeneous boxes in n dimensions [3, 8, 11, 52]. Since these algorithms are not based on any mathematical programming formulation, the problem of solving a set covering problem, which is NP hard, does not exist.

5.2 Comparison with Other Methods

There exists a plethora of methods and algorithms originating from distinct disciplines, such as statistical learning, artificial intelligence and operations research, to classify and analyze data. Some are powerful predictive tools, but can sometimes have a black-box structure, which can make them somewhat difficult to interpret and to use for knowledge presentation. A distinctive advantage of LAD, besides its competitive accuracy, resides in its interpretable patterns. As mentioned earlier, this property leads the way to root-cause analysis and cause-effect interpretation, which are both important in many practical applications. Moreover, the patterns generated by LAD have different properties that depend on the pattern-generation algorithm, and the user controls these properties. For example, he/she can impose the generation of only strong patterns, which have the highest coverage, or non-pure patterns that allow the coverage of a restricted percentage of observation from the opposite class. These features of the LAD approach add some valuable advantages to any domain-expert user who desires maximum flexibility from the machine learning algorithm. Other pattern classification methods have some similarities with LAD. These are in general rule-induction methods based on the extraction of descriptive rules allowing for the understanding of the most interesting relationships in data. Among those are the rough set theory (RST) [83] and decision tree (DT) [91] methods, which are interpretable and extensively used for inductive inference. Other examples of rule induction methods are the ENDER [35], SLIPPER [32], MLRules [34], and RuleFit [41]. In what follows, we provide a succinct comparison of LAD with the widely used support vector machine (SVM), RST, DT, and artificial neural network data analysis methods. We refer the reader to Chikalov *et al.* [30] for a comprehensive discussion on the similarities and differences between LAD, RST, and DT.

Rough Set Theory. Rough set theory [83, 84] is a data mining and knowledge discovery methodology used to extract meaningful and humanreadable decision rules from data with incomplete values and allowing for the classification of imprecise data [90]. The fundamental concept of the rough set theory is indiscernibility, which is used to define the equivalence classes for the observations [84]. Considering a specific combination of variables, observations are indiscernible (similar) if they have the same values for this subset of variables [107]. Rough set structures allow the implementation of the notions of lower and upper approximations of each data class. The objective is to search for the optimal rough approximation with the minimal boundary region separating between classes of observations [106]. The RST approximations are determined for the training observations only. Therefore, as in the problem of searching for an extension theory in the LAD approach, a challenge in the RST inductive learning approach is to construct optimal approximations for the extension of the rough membership function. Researches have been conducted to modify and to improve the classical RST approach to construct rough classifiers [74]. As for LAD, RST can rely on efficient algorithms to find hidden patterns. In addition to its interpretability, one of the advantages of RST is that it can handle uncertain problems, and it can process the knowledge by obtaining the minimal representation of any type of information.

The RST is not only useful for rule extraction and classification; it can also be employed as a feature selection method [106]. In rough set theory, the feature selection problem is defined in terms of reducts, i.e., subsets of the most informative variables in a given dataset. The main issue of the application of the RST approach to real-world problems is that the number of all extracted rules can be exponential with respect to the size of the data. An approach to deal

with such hard problems is to use heuristic techniques to search for an approximate instead of an exact or optimal solution. Nguyen presents in [82] a rough set and approximate Boolean reasoning (RSABR) methodology to solve RST problems involving the search for reducts and for decision rules. For the reducts search problem, the objective is to create a Boolean function such that a set of attributes is a reduct if and only if it corresponds to a prime implicant of the function. The methodology starts by calculating the discernibility matrix of the decision table and then derives the corresponding discernibility function. The discernibility function is then transforming into a DNF, and finally, the reducts are obtained after searching all the prime implicants. Similarly to the reduct problem, the rule search problem is solved by calculating the discernibility function. The rules are the derived to form the rough classifier.

As LAD, the Dominance-based Rough Set Approach method (see, e.g., [15, 43, 44]), which is an extension of rough set theory and applies to data describing ordinal classification problems with monotonicity constraints, provides an interpretable classification system. DRSA represents upward and downward unions of decision classes with dominance cones and induces the description of objects in terms of five main types of decision rules. These are formulated in terms of "if ..., ... , and if ..., then ..." statements that can be viewed as conjunctive normal forms and have some resemblance with the LAD patterns.

Decision Tree. Decision tree methods [91] are widely used and practical for inductive inference because of their simple representation and easy readability. It is a flow-chart-like structure where each node represents a test on a variable, and each branch represents an outcome of the test and the values of the class variable are placed at the leaves of the tree [109]. Each branch represents an outcome of the test, and leaf nodes represent decision classes. Some measures are used to estimate the quality of tests such as the entropy, Ginis index, sum-minority, max-minority and sum-impurity [109, 75]. The maximal-discernibility (MD) algorithm, presented in [82], uses discernibility measure to evaluate the quality of tests. The advantage of the decision trees is that they are represented as sets of if-then rules that are readable (interpretable) to the human [30]. A limitation is that large decision trees with many branches are difficult and time-consuming to interpret. The search for the shortest decision tree has been shown to be NP-hard, and one of the main challenges related to this approach is how to construct an optimal tree. Therefore, heuristic algorithms are used to find the tree that is very close to the optimal one. Well-known algorithms for constructing decision trees are classification and regression trees (e.g., C4.5, C5.0 and the iterative dichotomiser 3) and are mainly based on the top-down recursive strategy [99]. The training of decision trees requires optimization to determine the best split of each node and to select optimal combining weights to prune the decision tree. To avoid overfitting, most decision tree algorithms use a post-pruning strategy that involves constructing the tree from the data until all possible leaf nodes have been reached [75]. Nguyen proposed the RSABR approach for the construction of decision tree based on managing the discernible objects. The method uses a discernibility measure in the induction of maximal-discernibility decision trees [82] and solves a problem called MD partition to determine the optimal binary partition with respect to discernibility. The Boolean reasoning approach to discretization was evaluated by their discernibility properties based on the MD algorithm. The method can also guarantee maximal discernibility of observations in different decision classes and hence can deal with datasets with missing values. In that study, a fuzzy decision tree approach, which has similarities with non-pure LAD patterns, was proposed. The proposed approach can overcome the overfitting problem without pruning and can construct soft decision trees from large datasets.

Support Vector Machine The SVM is a classification technique developed originally by Vapnik and his co-workers and is based on statistical learning theory [108]. The classification patterns of the SVM are obtained by finding the optimal hyperplanes separating the classes of data observations by lifting them from their original input space to a higher dimensional feature space by using different kernel functions [112]. The separating hyperplanes are obtained by solving constrained optimization problems that aim at maximizing the margins between the hyperplanes and the training data of different classes. One of the advantages of the SVM is that the SVM can provide a unique solution and is a strongly regularized method that seeks a globally optimized solution and hadges against poor generalization. However, one of the limitations of the SVM approach is that the accuracy of the model depends on the choice of the defined kernel and its parameters. These parameters play a crucial role and should be optimized to yield better generalization performance. A similarity between SVM and LAD is that both of them, in general, need to preprocess the data before solving the optimization problem aimed at classifying the data. In LAD, the data are binarized, while for SVM the data are sometimes lifted to a higher-dimensional (feature) space via kernel functions. A difference between SVM and LAD is that SVM is a predictive method and not really descriptive as LAD. The SVM model is defined in terms of weights and bias, which are not always straightforward to interpret.

Artificial Neural Network. The ANN classifier was inspired by the biological neurons and proven to be a powerful tool for learning by constructing a nonlinear mapping between a given set of input and output data [14]. It is a well-known classification method, due to its inherent pattern recognition capabilities and its ability to handle noisy data [101]. The classification patterns are extracted from the data in the form of connection weights between network layers. The weights are updated during the learning process based on the backpropagation error signal representing the difference between the desired output and the actual outputs of the network. One of the limitations of the ANN classifier is that the user cannot extract the knowledge from the weights that are distributed throughout the whole network, making the network resemble a black box. Another limitation is the tedious parameter tuning of the network structure. Moreover, the ANN algorithm is based on the principle of empirical risk minimization, which can lead to local minima.

References

[1] G. Alexe, S. Alexe, D.E. Axelrod, T.O. Bonates, I.I. Lozina, M. Reiss, P.L. Hammer, Breast cancer prognosis by combinatorial analysis of gene expression data, *Breast Cancer Research*, 8, 2006, R41.

[2] G. Alexe, S. Alexe, T.O. Bonates, A. Kogan, Logical Analysis of Data - the vision of Peter L. Hammer, *Annals of Mathematics and Artificial Intelligence*, 49: 265-312, 2007.

[3] G. Alexe, S. Alexe, P.L. Hammer, Pattern-based clustering and attribute analysis, *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 10(5): 442-452, 2006.

[4] G. Alexe, S. Alexe, P.L. Hammer, B. Vizvari, Pattern-based feature selection in genomics and proteomics, *Annals of Operations Research*, 148: 189-201, 2006.

[5] G. Alexe, S. Alexe, L. Liotta, E. Petricoin, M. Reiss, P.L. Hammer, Ovarian cancer detection by logical analysis of proteomic data, *Proteomics*, 4(3): 766-783, 2004.

[6] S. Alexe, E. Blackstone, P.L. Hammer, H. Ishwaran, M.S. Lauer, C.E. Pothier Snader, Coronary risk prediction by Logical Analysis of Data, *Annals of Operations Research*, 119:15-42, 2003.

[7] G. Alexe, P.L. Hammer, Spanned patterns in Logical Analysis of Data, *Discrete Applied Mathematics*, 154: 1039-1049, 2006.

[8] S. Alexe, P.L. Hammer, Accelerated algorithm for pattern detection in Logical Analysis of Data, *Discrete Applied Mathematics*, 154: 1050-1063, 2006.

[9] A.N. Antamoshkin, I.S. Masich, R.I. Kuzmich, Heuristics and criteria for constructing logical patterns in data, *IOP Conf. Series: Materials Science and Engineering*, 94, article id. 012003, 2015.

[10] M. Antony and J. Ratsaby, Robust cutpoints in the logical analysis of numerical data, *Discrete Applied Mathematics*, 160(4):355-364, 2012.

[11] M. Anthony, J. Ratsaby, A Hybrid Classifier based on Boxes and Nearest Neighbors, *Discrete Applied Mathematics*, 172: 1-11, 2014.

[12] P. Bagchi, M.A. Lejeune, A. Alam, How supply chain competency affects FDI decisions: Some insights, *International Journal of Production Economics*, 147: 239-251, 2014.

[13] A. Bennane, S. Yacout, LAD-CBM; new data processing tool for diagnosis and prognosis in condition-based maintenance, *Journal of Intelligent Manufacturing*, 23: 265-275, 2012.

[14] C.M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[15] J. Blaszczyński, S. Greco, R. Slowiński, Multi-criteria classification - A new scheme for application of dominance-based decision rules, *European Journal of Operational Research*, 181(3): 1030-1044, 2007.

[16] J. Blazewicz, P.L. Hammer, P. Lukasiak, Prediction of protein secondary structure using Logical Analysis of Data algorithm, *Computational Methods in Science and Technology*, 7(1): 7-25, 2001.

[17] J. Blazewicz, P.L. Hammer, P. Lukasiak, Predicting secondary structures of proteins, *IEEE Engineering in Medicine and Biology*, 24(3): 88-94, 2005.

[18] T.O. Bonates, Large margin ruled-based classifiers, in *Wiley Encyclopedia of Operations Research and Management Science*, 1-12, 2010.

[19] T.O. Bonates, V.S.D. Gomes, *LAD-WEKA Tutorial*, Version 1.0, 2014.

[20] T.O. Bonates, P.L. Hammer, Logical Analysis of Data - an overview: From combinatorial optimization to medical applications, *Annals of Operations Research*, 148(1): 203-225, 2006.

[21] T.O. Bonates, P.L. Hammer, A. Kogan, Maximum patterns in datasets, *Discrete Applied Mathematics*, 156(6): 846-861, 2008.

[22] E. Boros, Y. Crama, P.L. Hammer, T. Ibaraki, A. Kogan, K. Makino, Logical Analysis of Data: Classification with justification, *Annals of Operations Research*, 188: 33-61, 2011.

[23] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz, I. Muchnik, An implementation of Logical Analysis of Data, *IEEE Transactions on Knowledge and Data Engineering*, 12: 292-306, 2000.

[24] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, Logical analysis of numerical data, *Mathematical Programming*, 79: 163-190, 1997.

[25] A.R. Brannon, A. Reddy, M. Seiler, A. Arreola, D.T. Moore, R.S. Pruthi, E.M. Wallen, M.E. Nielsen, H.-Q. Liu, K.L. Nathanson, B. Ljunberg, H.-J. Zhao, J.D. Brooks, S. Ganesan, G. Bhanot, W.K. Rathmell, Molecular stratification of clear cell renal cell carcinoma by consensus clustering reveals distinct subtypes and survival patterns, *Gene & Cancer*, 1(2): 152-163, 2010.

[26] M.W. Brauner, N. Brauner, P.L. Hammer, I. Lozina, D. Valeyre, Logical analysis of computed tomography data to differentiate entities of idiopathic interstitial pneumonias, *Data Mining in Biomedicine*, Springer Optimization and its Applications Series Vol. 7, Part 1, 193-208, 2007.

[27] N. Brauner, S. Gravier, L.-P. Kronek, F. Meunier, LAD models, trees, and an analog of the fundamental theorem of arithmetic. *Discrete Applied Mathematics*, 161(7):909–920, 2013.

[28] R. Bruni, Reformulation of the support set selection problem in the Logical Analysis of Data, *Annals of Operations Research*, 150: 79-92, 2007.

[29] M. Caserta, T. Reiners, A pool-based pattern generation algorithm for Logical Analysis of Data with automatic fine-tuning, *European Journal of Operational Research*, 248: 593-606, 2016.

[30] I. Chikalov, V. Lozin, I. Lozina, M. Moshkov, H.S. Nguyen, A. Skowron, B. Zielosko, *Three Approaches to Data Analysis Test Theory, Rough Sets and Logical Analysis of Data.* Springer Berlin, Heidelberg, 2013.

[31] C.-A. Chou, T.O. Bonates, C. Lee, W.A. Chaovalitwongse, Multi-pattern generation framework for Logical Analysis of Data, *Annals of Operations Research*, 249(1): 329-349, 2017.

[32] W.W. Cohen, Y. Singer, A simple, fast, and effective rule learner, *AAAI-99 Proceedings*, 99: 335-342, 1999.

[33] Y. Crama, P.L. Hammer, T. Ibaraki, Cause-effect relationships and partially defined Boolean functions, *Annals of Operations Research*, 16: 299-326, 1988.

[34] K. Dembczyński, W. Kotlowski, R. Slowiński, Maximum likelihood rule ensembles, *Proceedings of the 25th International Conference on Machine Learning*, 224-231. 2008.

[35] K. Dembczyński, W. Kotlowski, R. Slowinski, ENDER: A Statistical Framework for Boosting Decision Rules, *Data Mining and Knowledge Discovery*, 21(1): 52-90, 2010.

[36] DexinCanada. cbmLAD. Available: `https://www.dexincanada.com/`. Last accessed: February 20, 2018.

[37] D.H. Douglas and T.K. Peucker, Algorithms for the reduction of the number of points required to represent a digitized line or its caricature, *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10: 112-122, 1973.

[38] C. Dupuis, M. Gamache, J.-F. Pagé, Logical Analysis of Data for estimating passenger show rates at Air Canada, *Journal of Air Transport Management*, 18: 78-81, 2012.

[39] A. Eliasson Sovereign credit ratings. *Working Papers 02-1*, 2002, Deutsche Bank.

[40] Fitch Ratings. The role of support and joint probability analysis in bank ratings, 2006, *Fitch Special Report*.

[41] J.H. Friedman, B.E. Popescu, Predictive learning via rule ensembles, *The Annals of Applied Statistics*, 2: 916-954, 2008.

[42] A. Ghasemi, S. Esmaeili, S. Yacout, Equipment mean residual life estimation using Logical Analysis of Data, *International Journal of Decision Sciences, Risk and Management*, 6: 16-33, 2015.

[43] S. Greco, B. Matarazzo, R. Slowiński, Rough approximation of a preference relation by dominance relations, *European Journal of Operational Research*, 117: 63-83, 1999.

[44] S. Greco, B. Matarazzo, R. Slowiński, Rough sets theory for multicriteria decision analysis, *European Journal of Operational Research* 129(1) (2001), 1-47.

[45] A.V. Gubskaya, T.O. Bonates, V. Kholodovych, P.L. Hammer, W.J. Welsh, R. Langer, J. Kohn. Logical Analysis of Data in structure-activity investigation of polymeric gene delivery. *Macromolecular Theory and Simulations*, 20: 275-285, 2011.

[46] A.B. Hammer, P.L. Hammer, I. Muchnik, Logical analysis of Chinese labor productivity patterns, *Annals of Operations Research*, 87: 165-176, 1999.

[47] Peter L. Hammer, Partially defined boolean functions and cause-effect relationships. In: *International Conference on Multi-attribute Decision Making via OR-based Expert Systems*. University of Passau, Passau, Germany, 1986.

[48] Peter L. Hammer, A. Kogan, M.A. Lejeune, Modeling country risk ratings using partial orders, *European Journal of Operational Research*, 175(2): 836-859, 2006.

[49] Peter L. Hammer, A. Kogan, M.A. Lejeune, Reverse engineering country risk ratings: A combinatorial non-recursive model, *Annals of Operations Research*, 188: 185-213, 2011.

[50] Peter L. Hammer, A. Kogan, M.A. Lejeune, A logical analysis of bank's financial strength ratings, *Expert Systems with Applications*, 39(9): 7808-7821, 2012.

[51] Peter L. Hammer, A. Kogan, B. Simeone, S. Szedmák, Pareto-optimal patterns in Logical Analysis of Data, *Discrete Applied Mathematics*, 144(1-2): 79-102, 2004.

[52] Peter L. Hammer, Y. Liu, B. Simeone, S. Szedmák, Saturated systems of homogeneous boxes and the logical analysis of numerical data, *Discrete Applied Mathematics*, 144(1-2): 103-109, 2004.

[53] J. Han, N. Kim, B.-J. Yum, M.-K. Jeong, Pattern selection approaches for the Logical Analysis of Data considering the outliers and the coverage of a pattern, *Expert Systems with Applications*, 38: 13857-13862, 2011.

[54] P. Hansen and C. Meyer, A new column generation algorithm for Logical Analysis of Data, *Annals of Operations Research*, 188: 215-249, 2011.

[55] X. Hong, M.A. Lejeune, N. Noyan, Stochastic network design for disaster preparedness, *IIE Transactions*, 47: 329-357, 2015.

[56] R. Ji, M.A. Lejeune, Risk-budgeting multi-portfolio optimization with portfolio and marginal risk constraints, *Annals of Operations Research*, 2017, Accepted: `https://doi.org/10.1007/s10479-015-2044-9`.

[57] S. Jocelyn, Y. Chinniah, M.-S. Ouali, S. Yacout, Application of Logical Analysis of Data to machinery-related accident prevention based on scarce data, *Reliability Engineering & System Safety*, 159: 223-236, 2017.

[58] H.H. Kim, J.Y. Choi, A LAD-based evolutionary solution procedure for binary classification problem, *International Journal of Industrial Engineering : Theory Applications and Practice*, 21: 360-375, 2014.

[59] H.H. Kim, J.Y. Choi, Hierarchical multi-class LAD based on OvA binary tree using genetic algorithm, *Expert Systems with Applications*, 42: 8134-8145, 2015.

[60] A. Kogan, M.A. Lejeune, Combinatorial methods for constructing credit risk ratings, *Handbook of Financial Econometrics and Statistics*, C.-F. Lee, J. Lee (eds.) Springer, 439-483, 2014.

[61] A. Kogan, M.A. Lejeune, Threshold Boolean form for joint probabilistic constraints with random technology matrix, *Mathematical Programming*, 147: 391-427, 2014.

[62] R. Kohli, R. Krishnamurtib, K. Jedidi, Subset-conjunctive rules for breast cancer diagnosis, *Discrete Applied Mathematics*, 154: 1100-1112, 2006.

[63] L.-P. Kronek, A. Reddy, Logical analysis of survival data: Prognostic survival models by detecting high-degree interactions in right-censored data, *Bioinformatics*, 24: i248-i253, 2008.

[64] *LADWEKA*: `http://www.lia.ufc.br/ tiberius/lad/`. Last accessed on December 27, 2017.

[65] M.S. Lauer, S. Alexe, C.E. Pothier Snader, E.H. Blackstone, H. Ishwaran, P.L. Hammer, Use of the Logical Analysis of Data method for assessing long-term mortality risk after exercise electrocardiography, *Circulation*, 106: 685-690, 2002.

[66] M.A. Lejeune, Pattern-based modeling and solution of probabilistically constrained optimization problems. *Operations Research*, 60: 1356-1372, 2012.

[67] M.A. Lejeune, Pattern definition of the p-efficiency concept, *Annals of Operations Research*, 200: 23-36, 2012.

[68] M.A. Lejeune, J. Kettunen, Managing reliability and stability risks in forest harvesting, *Manufacturing & Service Operations Management*, 19(4): 620-638, 2018.

[69] M.A. Lejeune, J. Kettunen, Fractional stochastic integer programming problem for reliability-to-stability ratio in forest harvesting. *Computational Management Science*. Accepted, 2018.

[70] M.A. Lejeune, F. Margot, Optimization for simulation: LAD accelerator, *Annals of Operations Research*, 188: 285-305, 2011.

[71] M.A. Lejeune, F. Margot, Solving chance-constrained optimization problems with stochastic quadratic inequalities, *Operations Research*, 64: 939-957, 2016.

[72] M.A. Lejeune, F. Margot, A.D. de Oliveira. Chance-constrained programming models with endogenous and exogenous uncertainty. *Working Paper*, 2018.

[73] M.A. Lejeune, S. Shen, Multi-objective probabilistically constrained programs with variable risk: Models for multi-portfolio financial optimization, *European Journal of Operational Research*, 252: 522-539, 2016.

[74] T.Y. Lin, N. Cercone, *Rough Sets and Data Mining: Analysis of Imprecise Data*, Springer Science & Business Media, 2012.

[75] R. Lior, *Data Mining with Decision Trees: Theory and Applications*, World Scientific, 2014.

[76] D. López-Soto, S. Yacout, F. Angel-Bello, Root cause analysis of familiarity biases in classification of inventory items based on logical patterns recognition, *Computers & Industrial Engineering*, 93: 121-130, 2016.

[77] L. Lupsa, I. Chiorean, L. Neamtiu, Use of LAD in establishing morphologic code, *Proceedings of the 2010 IEEE International Conference on Automation Quality and Testing Robotics (AQTR)*, 1-6, 2010.

[78] M.-A. Mortada, T. Carroll, S. Yacout, A. Lakis, Rogue components: their effect and control using Logical Analysis of Data, *Journal of Intelligent Manufacturing*, 23: 289-302, 2012.

[79] M.-A. Mortada, S. Yacout, A. Lakis, Diagnosis of rotor bearings using Logical Analysis of Data, *Journal of Quality in Maintenance Engineering*, 17: 371-397, 2011.

[80] M.-A. Mortada, S. Yacout, A. Lakis, Fault diagnosis in power transformers using multi-class Logical Analysis of Data, *Journal of Intelligent Manufacturing*, 25: 1429-1439, 2014.

[81] Moody's, Bank financial strength ratings: Revised methodology, *Moody's Global Credit Research Report*, 2006.

[82] H.S. Nguyen, Approximate Boolean reasoning: Foundations and applications in data mining, in: *Transactions on Rough Sets V*, eds: J.F. Peters, A. Skowron, Springer, 334-506, 2006.

[83] Z. Pawlak, Rough sets, *International Journal of Computer & Information Sciences*, 11: 341-356, 1982.

[84] Z. Pawlak, Rough set theory and its applications to data analysis, *Cybernetics & Systems*, 29: 661-688, 1998.

[85] A. Prékopa, On probabilistic constrained programming, *Proceedings of the Princeton Symposium on Mathematical Programming*, Princeton University Press, 113-138, 1970.

[86] A. Prékopa A, *Stochastic Programming*, Kluwer, Dordrecht-Boston, 1995.

[87] K. Puszyński, Parallel implementation of Logical Analysis of Data (LAD) for discriminatory analysis of protein mass spectrometry data, *Lecture Notes in Computer Science*, 3911: 1114-1121, 2006.

[88] H. Ono, K. Makino and T. Ibaraki, Logical Analysis of Data with decomposable structures, *Theoretical Computer Science*, 289 (2): 977-995, 2002.

[89] H. Ono, M. Yagiura and T. Ibaraki, A decomposability index in Logical Analysis of Data, *Discrete Applied Mathematics*, 142: 165-180, 2004.

[90] J.-T. Peng, C. Chien, T. Tseng, Rough Set Theory for Data Mining for Fault Diagnosis on Distribution Feeder, *IEE Proceedings-Generation, Transmission and Distribution*, 151: 689-697, 2004.

[91] J. R. Quinlan, Induction of decision trees, *Machine Learning*, 1: 81-106, 1986.

[92] A. Ragab, M. El-Koujok, B. Poulin, M. Amazouz, S. Yacout, Fault diagnosis in industrial chemical processes using interpretable patterns based on Logical Analysis of Data, *Expert Systems With Applications*, 95: 368-383, 2018.

[93] A. Ragab, M.-S. Ouali, S. Yacout, H. Osman, Remaining useful life prediction using prognostic methodology based on Logical Analysis of Data and Kaplan-Meier estimation, *Journal of Intelligent Manufacturing*, 27: 943-958, 2016.

[94] A. Ragab, S. Yacout, M.-S. Ouali, H. Osman, Prognostics of multiple failure modes in rotating machinery using a pattern-based classifier and cumulative incidence functions, *Journal of Intelligent Manufacturing*, 1-20, Accepted, 2016.

[95] A. Ragab, S. Yacout, M.-S. Ouali, H. Osman, Pattern-based prognostic methodology for condition-based maintenance using selected and weighted survival curves, *Quality and Reliability Engineering International*, Accepted, 2017.

[96] A. Ragab, S. Yacout, M.-S. Ouali, Intelligent data mining For automatic face recognition, *The Online Journal of Science and Technology*, 3(2): 97-101, 2013.

[97] A. Ragab, S. Yacout, M.-S. Ouali, Face recognition using Logical Analysis of Data, *Pattern Recognition and Image Analysis*, Accepted, 2017.

[98] A. Reddy, H. Wang, H. Yu, T.O. Bonates, V. Gulabani, J. Azok, G. Hoehn, P.L. Hammer, A.E Baird, K.C. Li, Logical Analysis of Data (LAD) model for the early diagnosis of acute ischemic stroke, *BMC Medical Informatics and Decision Making*, 8-30, 2008.

[99] L. Rokach, O. Maimon, Top-down induction of decision trees classifiers - A survey, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35: 476-487, 2005.

[100] H.S. Ryoo, I.-Y. Jang, MILP approach to pattern generation in Logical Analysis of Data, *Discrete Applied Mathematics*, 15(4): 749-761, 2009.

[101] S. Samarasinghe, *Neural Networks for Applied Sciences and Engineering: From Fundamentals to Complex Pattern Recognition*, CRC Press, 2016.

[102] A. Saxena, K. Goebel, D. Simon, and N. Eklund, Damage propagation modeling for aircraft engine run-to-failure simulation, in *Prognostics and Health Management*, 2008. PHM 2008. International Conference, 1-9, 2008.

[103] Y. Shaban, M. Meshreki, S. Yacout, M. Balazinski, H. Attia, Process control based on pattern recognition for routing carbon fiber reinforced polymer, *Journal of Intelligent Manufacturing*, 28: 165-179, 2017.

[104] Y. Shaban, S. Yacout, M. Balazinski, Tool wear monitoring and alarm system based on pattern recognition with Logical Analysis of Data, *Journal of Manufacturing Science and Engineering*, 137: 041004, 2015.

[105] Y. Shaban, S. Yacout, M. Balazinski, Krzysztof Jemielniak, Cutting tool wear detection using multi-class Logical Analysis of Data, *Journal of Machining Science and Technology*, Accepted, 2017.

[106] R.W. Swiniarski, A. Skowron, Rough set methods in feature selection and recognition. *Pattern Recognition Letters*, 24: 833-849, 2003.

[107] F.E. Tay, L. Shen, Fault diagnosis based on rough set theory, *Engineering Applications of Artificial Intelligence* 16: 39-43, 2003.

[108] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1995.

[109] I.H. Witten, E. Frank, M.A. Hall, C.J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, 2016.

[110] S. Yacout, A. Danish, S. ElSaadany, J.-P. Kapongo, S. Mani, J. Gomes, Knowledge discovery from observational data of causal relationship between clinical procedures and Alzheimer's disease, *Journal of Public Health*, 2: 1-10, 2013.

[111] S. Yacout, M. Mahmoud, H. Danish, Parallel computing of Logical Analysis of Data: A discrete optimization approach for pattern generation, *IFORS/CORS Conference*, Quebec City, Canada, 2017.

[112] B.-S. Yang, A. Widodo, Support vector machine for machine fault diagnosis and prognosis, *Journal of System Design and Dynamics*, 2: 12-23, 2008.