Logical Analysis of Multiclass Data with Relaxed Patterns

Travaughn Coren Bain*
Department of Mathematical Sciences
Florida Institute of Technology
Melbourne, FL, USA

Juan Felix Avila-Herrera†
Escuela de Informtica, Universidad
Nacional Escuela de Matemtica
Universidad de Costa Rica

Ersoy Subasi‡
Department of Engineering Systems
Florida Institute of Technology
Melbourne, FL, USA

Munevver Mine Subasi§¶
Department of Mathematical Sciences
Florida Institute of Technology
Melbourne, FL, USA

Abstract

This paper proposes a relaxed algorithm based on mixed integer linear programming (MILP) to extend the LAD methodology to solve multi-class classification problems, where One-vs-Rest (OvR) learning models are constructed to classify observations in predefined classes. The suggested algorithm has two control parameters, homogeneity and prevalence, for improving the classification accuracy of the generated patterns. The utility of the proposed approach is demonstrated through experiments on multi-class benchmark datasets.

1 Introduction

In various fields, research has been shifted from hypothesis driven to data driven where the classication problem has become ubiquitous in many real-world applications. Supervised learning algorithms are trained on a given set of observations with known outcome and multiple features and produce a classification model or classifier function to predict the outcome of a new/unseen observation.

For solving binary classication problems, a learning model is constructed to separate observations into two predefined classes. Well known classification methods such as support vector machines (Burges 1998; Schölkopf and Smola 2001), neural networks (Fausett 1994; Bishop 2007), decision trees (Bishop 2007; Duda, Hart, and Stork 2001), and a pattern based technique, called Logical Analysis of Data (Alexe et al. 2007), are designed to solve binary classification problems. However, many real-world applications require the identification of more than two subgroups of observations and the features and patterns associated with each subgroup (Subasi and Avila-Herrera 2016), Since the problem is of practical importance, there have been several attempts to extend binary classification algorithms to multi-class problems in literature.

The most common approaches to multi-class classication are the natural extension of binary classication problem known as One-vs-One (OvO) (Hastie and Tibshirani 1998), and One-vs-Rest (OvR). Given a $K-$class dataset, OvO scheme assumes that there exists a separator between any two classes and builds $(K-1)/2$ classiers, denoted by f_{ij}, to distinguish each pair of classes $\mathcal{C}_i, \mathcal{C}_j \in \mathcal{C}$, where $\mathcal{C} = \{\mathcal{C}_1, \cdots, \mathcal{C}_K\}$.The class of a new/unseen observation is then assigned by the use of the discriminant function:

$$f(\phi) = \arg\max_i \sum_j f_{ij}(\phi). \tag{1.1}$$

A less expensive approach OvR assumes the existence of a single separator between a class $\mathcal{C}_i$ (for some i) and all other classes, and builds K different binary classifiers. Let f_i be the ith classifier separating observations in class $\mathcal{C}_i$ (considered to be positive) and observations not in $\mathcal{C}_i$ (form the set of negative observations). In this case a new/unseen observation ϕ is classified by

$$f(\phi) = \arg\max_i f_i(\phi). \tag{1.2}$$

Since both approaches are easy to adopt, diverse groups of researchers invented them independently and the choice between the use of OvO and OvR in multi-class problems is largely computational.

In this paper, we integrate the mixed integer linear programming LAD pattern generation approach of (Ryoo and Jang 2009), with the multiclass LAD method of (Subasi and Avila-Herrera 2016), to develop a parametric multiclass LAD algorithm, where two control parameters, homogeneity and prevalence, are incorporated to generate relaxed patterns with high classification accuracy. LAD is a pattern-based two-class learning method which integrates principles of combinatorics, optimization, and the theory of Boolean functions. The research area of LAD was introduced and developed by Hammer (1986) and Crama, Hammer, and Ibaraki (1988). The LAD methodology has been expanded from theory to successful data applications in numerous biomedical, industrial, and economics case studies, see, e.g., (Boros et al. 2000; Reddy 2009; Hammer, Kogan, and Lejeune 2011; Subasi et al. 2017) and the references therein. The implementation of LAD algorithm was described in (Boros et al. 1997), and several further developments of the original algorithm were presented in (Alexe and Hammer 2006; Bonates, Hammer, and Kogan 2008; Hammer et al. 2004; Guo and Ryoo 2012; Ryoo and Jang 2009). An overview of

*Email: tbain2013@fit.edu
†Email: delagarita@gmail.com
‡Email: esubasi@fit.edu
§Email: msubasi@fit.edu
¶Corresponding Author.

standard LAD algorithm can be found in (Alexe et al. 2007; Bonates, Hammer, and Kogan 2008). Various recent applications of LAD are presented in (Dupuis, Gamache, and Pagé 2010; Esmaeili 2012; Kwok 2001; Lejeune and Margot 2011; Mortada, Yacout, and Lakis 2011). LAD method has been extended to survival analysis (Reddy 2009) and regression analysis (Bonates and Hammer 2007; Lemaire 2011) as well.

The key ingredient of two-class LAD method is the identification of patterns distinguishing between positive and negative observations in a dataset $\Omega = \Omega^+ \cup \Omega^-$, where Ω^+ is the set of positive observations and Ω^- is the set of negative observations containing n-dimensional real vectors and $\Omega^+ \cap \Omega^- = \emptyset$. LAD usually produces several hundreds (sometimes thousands) of patterns. Once all patterns are generated, a subset of patterns is selected by solving a set covering problem or by greedy-type heuristics to form an LAD classification model such that each positive (negative) observation is covered by at least one positive (negative) pattern (and ideally, is not covered by any negative (positive) pattern) in the model. The patterns selected into the LAD model are then used to define a discriminant function that allows the classification of new or unseen observations.

Extensions of LAD algorithm to multi-class problems are studied by Moreira (2000) and Mortada (2010). Moreira (2000) proposed two methods to break down a multiclass classification problem into two-class problems using an OvO approach. The first method uses the typical OvO type approach which does not require the alteration of the structure of the standard LAD algorithm as described in (Boros et al. 2000). The second OvO-type method modifies the architecture of the pattern generation and theory formation steps in standard LAD method, where an LAD pattern P_{ij} is generated for each pair of classes $\mathcal{C}_i, \mathcal{C}_j \in \mathcal{C}$, $i \neq j$.

The paper by Mortada (2010) proposed a multi-class LAD method algorithm which integrates ideas from the second approach presented by Moreira (2000) which is based on OvO approach and an implementation of LAD based on mixed integer linear programming (MILP) presented by Ryoo and Jang (2009). The methodology of Mortada (2010) was applied to five multi-class benchmark datasets. The authors of this paper observed that the MILP based LAD approach of Ryoo and Jang (2009) combined with the second approach of Moreira (2000) provides classification models with higher accuracy than those models obtained by Moreira (2000) approach applied to standard LAD algorithm.

Recent papers by Subasi and Avila-Herrera (2013; 2016) and Kim and Choi (2015) have also considered the multiclass extension of LAD. Subasi and Avila-Herrera (2016) explored and rectified the limitations of the MILP LAD approach by Ryoo and Jang (2009), relating to its poor differentiating power in two-class classification. Ryoo and Jang's MILP approach produces a set of LAD patterns associated with a positive (negative) class that loops as many times as necessary until all observations in positive (negative) class are covered by at least one pattern, which is inconvenient because a single pattern is sufficient to cover every observation in positive (negative) class which results in a classifier with small number of patterns. Moreover, Ryoo and Jang's algorithm removes the observations covered by a pattern, whilst looping through execution. This is counterproductive because every time the algorithm loops through again, it uses less information (smaller training set) to compute new patterns. Subasi and Avila-Herrera's extension of Ryoo and Jang's MILP to multiclass LAD approach avoids the removal of observations from the training dataset when generating new patterns that form a multiclass LAD model, these modifications are discussed in further detail later.

In this paper we propose a parametrized/relaxed algorithmic approach that builds on the MILP pattern generation approach of Ryoo and Jang (2009) and multiclass LAD approach of Subasi and Avila-Herrera (2016) that constructs an OvR-type LAD classier to identify patterns in a multiclass dataset. This modification introduces two control parameters, homogeneity and prevalence, to improve the classification accuracy of the generated patterns. The organization of the paper is as follows. Section 2 describes the basic principles of the standard LAD algorithm. Section 3 presents the proposed MILP based parametrized/relaxed algorithmic multiclass LAD approach to obtain OvR-type multi-class LAD classier. In Section 4 we present experiments on multiclass benchmark datasets to demonstrate the utility of our proposed methodology.

2 Preliminaries: Logical Analysis of Data

Logical Analysis of Data (LAD) is a two-class learning algorithm based on combinatorics, optimization, and the theory of Boolean functions. The input dataset, Ω, consists of two disjoint classes Ω^+ (set of positive observations) and Ω^- (set of negative observations), that is, $\Omega = \Omega^+ \cup \Omega^-$ and $\Omega^+ \cap \Omega^- = \emptyset$. The main task of LAD algorithm is to identify patterns separating the positive and negative observations based on features measured (Boros et al. 2000). Below we briefly outline the basic components of the LAD algorithm. A more detailed overview can be found in (Alexe and Hammer 2006; Hammer and Bonates 2006).

2.1 Discretization/Binarization and Support Set Selection

This step is the transformation of numeric features (attributes/variables) into several binary features without losing predictive power. The procedure consists of finding cut-points for each numeric feature. The set of cut-points can be interpreted as a sequence of threshold values collectively used to build a global classification model over all features (Boros et al. 2000). Discretization is a very useful step in data mining, especially for the analysis of medical data (which is very noisy and includes measurement errors) – it reduces noise and produces robust results. The problem of discretization is well studied and many powerful methods are presented in literature, see, e.g., the survey papers (Kotsiantis and Kanellopoulus 2006; Liu et al. 2004)). Discretization step may produce several binary features some of which may be redundant. Support set is defined as the smallest (irredundant) subset of binary variables which can distinguish every pair of positive and negative observations in the dataset. Support sets can be identified by solving a

minimum set covering problem (Boros et al. 2000).

2.2 Pattern Generation

Patterns are the key ingredients of LAD algorithm. This step uses the features in combination to produce rules (combinatorial patterns) that can define homogenous subgroups of interest within the data. The simultaneous use of two or more features allows the identification of more complex rules that can be used for the precise classification of an observation.

Given a binary (or binarized) dataset $\Omega = \Omega^+ \cup \Omega^-$, where $\Omega^+ \cap \Omega^- = \emptyset$, a *pattern* P is simply defined as a subcube of $\{0,1\}^n$, where n is the number of features in the dataset. A pattern can be also described as a Boolean term, that is, a conjunction of literals (binary variables or its negation) which does not contain both a variable and its negation:

$$P = \bigwedge_{j \in N_P} x_j$$

where $N_P \subseteq \{1, \cdots, n\}$ and x_j is a literal. The number of literals (associated with features) involved in the definition of a pattern is called the *degree* of the pattern.

Patterns define homogeneous subgroups of observations with distinctive characteristics. An observation $\phi \in \Omega$ satisfying the conditions of a pattern P, i.e., $P(\phi) = 1$, is said to be *covered* by that pattern. A pure positive (negative) pattern is defined as a combination of features which covers a proportion of positive (negative) observations, but none of the negative (positive) ones: $P(\phi) = 1$ for at least one $o \in \Omega^+$ (or, $\phi \in \Omega^-$), and $P(\phi) = 0$ for every $\phi \in \Omega^-$ (or, $o \in \Omega^+$). *Coverage* of a positive (negative) pattern P, denoted by $Cov(P)$, is the set of observations $\phi \in \Omega^+ (or, \ \phi \in \Omega^-)$ for which $P(\phi) = 1$. A pattern P is called a *strong pattern* if there is no pattern P' such that $Cov(P) \subset Cov(P')$. Pattern P is called a *prime pattern* if the deletion of any literal from P results in a term that is no longer a pattern.

The most straightforward approach to pattern generation is based on the use of combinatorial enumeration techniques, for example, a *bottom-up/top-down* approach as described by Boros et al. (2000). The bottom-up approach follows a lexicographic order in generating the patterns in order to reduce the amount of computations necessary. The approach starts with terms of degree one that cover some positive observations. If such a term does not cover any negative observation, it is a positive pattern. Otherwise, literals are added to the term one by one until generating a pattern of prefixed degree. The top-down pattern generation approach starts by considering all uncovered observations as patterns of degree n and for each of those patterns, literals are removed one by one, until a prime pattern is reached. The enumeration type pattern generation approach is a costly process. Given a two-class binary dataset with n features, the total number of candidate patterns to be searched is $\sum_{i=1}^{n} 2^i \binom{n}{i}$ and the number of degree d patterns can be $2^d \binom{n}{d}$.

Since patterns play a central role in LAD methodology, various types of patterns (e.g., prime, spanned, maximum) have been studied and several pattern generation algorithms have been developed for their enumeration (Alexe et al. 2007; Bonates, Hammer, and Kogan 2008; Hammer et al. 2004; Guo and Ryoo 2012; Ryoo and Jang 2009). Our OvR-type multi-class LAD algorithm is motivated by the MILP approach of Ryoo and Jang (2009) that generates strong LAD patterns in a two-class dataset. This approach is outlined below:

Consider a two-class dataset Ω consisting of m binary observations and n features. Let $I^+ = \{i : \phi_i \in \Omega^+\}$ and $I^- = \{i : \phi_i \in \Omega^-\}$, where $\Omega = \Omega^+ \cup \Omega^-$ and $\Omega^+ \cap \Omega^- = \emptyset$. For each observation $\phi_i \in \Omega$, let ϕ_{ij} denote the binary value of the j-th feature in that observation. Let $a_j, j = 1, \cdots, n$, denote the features in Ω and introduce n new features $a_{n+j} = 1 - a_j$, $j = 1, \cdots, n$ (negation of a_j). Ryoo and Jang (2009) formulated the following MILP to generate strong patterns:

$$\begin{aligned}
&\text{Minimize } \ z = c\,d + \sum_{i \in I^+} w_i \\
&\text{subject to} \\
&\sum_{j=1}^{2n} \phi_{ij} y_j + n w_i \geq d, \quad i \in I^+ \\
&\sum_{j=1}^{2n} \phi_{ij} y_j \leq d - 1, \qquad i \in I^- \\
&y_j + y_{n+j} \leq 1, \quad j = 1, \cdots, n \\
&\sum_{j=1}^{2n} y_j = d; \qquad 1 \leq d \leq n \\
&w_i, y_j \in \{0,1\}, \ i = 1, \cdots, m; j = 1, \cdots, 2n
\end{aligned} \tag{2.3}$$

where $c \in \mathbb{R}$ is a constant and variables y_j and y_{n+j} are associated with features a_j and $a_{n+j}, j = 1, \cdots, n$, respectively. Binary variables w_i's are associated with the coverage of a positive pattern P and are defined by

$$w_i = \begin{cases} 1 & \text{if } P(\phi_i) = 0, \ i \in I^+ \\ 0 & \text{if } P(\phi_i) = 1, \ i \in I^+ \end{cases}$$

Ryoo and Jang (2009) proved that when $c > 0$, an optimal solution $(\mathbf{w}, \mathbf{y}, d)$ of problem (2.3) is a positive strong prime pattern of the form:

$$P = \bigwedge_{\{j \,:\, y_j = 1, j = 1, \cdots, n\}} a_j \qquad \bigwedge_{\{j \,:\, y_{n+j} = 1, j = 1, \cdots, n\}} \bar{a}_j \ .$$

Note that if we change the roles of index sets I^+ and I^- in problem (2.3), an optimal solution of the problem provides us with a pure negative strong prime pattern when $c > 0$.

2.3 LAD Model

An LAD model is a collection of positive and negative patterns which provides the same separation of the positive and negative observations as the entire collection of patterns, called *pandect* and denoted by $\mathcal{P} = \mathcal{P}^+ \cup \mathcal{P}^-$, where $\mathcal{P}^+$ and $\mathcal{P}^-$ are disjoint sets of all positive and negative patterns generated in *pattern generation step*, respectively. In many cases, when constructing an LAD model, every observation in the training dataset is required to be covered at least k times ($k \in \mathbb{Z}^+$) by the patterns in the model, $\mathcal{M} = \mathcal{M}^+ \cup \mathcal{M}^-$, where $\mathcal{M}^+ \subseteq \mathcal{P}^+$ and $\mathcal{M}^- \subseteq \mathcal{P}^-$.

Such an LAD model can be obtained from the pandect $\mathcal{P}$ by solving a set covering problem. However, in general, the size of the pandect is very large. In this case the standard LAD algorithm (where patterns are generated by, for example, top-down/bottom-up approach) uses greedy heuristics to solve the set-covering problem to generate an LAD model.

In case of MILP approaches to generate LAD patterns, Ryoo and Jang (2009) presented the following *pattern generation* algorithm based on their MILP approach to produce an LAD model (a set of positive and negative patterns):

Algorithm 1: Pattern Generation

Data: Training data, Support Features, MILP model (2.3) for pattern generation
Result: Set of $+$ and $-$ patterns ($\mathcal{M}^+$ and $\mathcal{M}^-$, resp.)

```
for * ∈ {+, −} do
    set M* = ∅ ;
    while I* ≠ ∅ do
        formulate and solve an instance of the MILP problem (2.3);
        form a pattern P from the solution obtained;
        M* ← M* ∪ {P};
        I* ← I* \ {i ∈ I* : φ_i is covered by P};
return M*;
```

Algorithm 1 generates the minimum number of patterns required to cover the training data set. Note that after a pattern is generated, observations covered by that pattern is deleted from the training data to prevent the algorithm from finding the same pattern found in the previous solutions of problem (2.3). The resulting set of positive and negative patterns form an LAD model $\mathcal{M}$.

2.4 Classification and Accuracy

In the final step for the LAD framework, generated patterns are employed to form a classication model, known as theory. The theory plays the role of a classier as the weighted linear combination of positive and negative patterns. Therefore, given an LAD model $\mathcal{M} = \mathcal{M}^+ \cup \mathcal{M}^-$, the classification of a new/unseen observation $\phi \notin \Omega$ is determined by the sign of a discriminant function $\Delta : \{0,1\}^n \to I\!R$ associated with the model $\mathcal{M}$, where $\Delta(\phi)$ is defined as the difference between the proportion of positive patterns and negative patterns covering o, that is,

$$\Delta(\phi) = \sum_{P_k^+ \in \mathcal{M}^+} \omega_k^+ P_k^+(\phi) \; - \sum_{P_k^- \in \mathcal{M}^-} \omega_k^- P_k^-(\phi),$$

where $\omega_k^+ \geq 0$ and $\omega_k^- \geq 0$ are the weights assigned to positive patterns $P_k^+ \in \mathcal{M}^+$ and negative patterns $P_k^- \in \mathcal{M}^-$, respectively. The weights ω_k^+ and ω_k^- can be calculated in several ways. One possibility is to use the proportion of positive (negative) observations covered by a positive pattern $P_k^+ \in \mathcal{M}^+$ (a negative pattern $P_k^- \in \mathcal{M}^-$) to the total number of positive (negative) observations (called the prevalence of a pattern):

$$\omega_k^+ = \frac{1}{|\Omega^+|} \sum_{i \in I^+} P_k^+(\phi_i) \quad \text{and} \quad \omega_k^- = \frac{1}{|\Omega^-|} \sum_{i \in I^-} P_k^-(\phi_i)$$

where $I^+ = \{i : \phi_i \in \Omega^+\}$, and $I^- = \{i : \phi_i \in \Omega^-\}$.

The accuracy of the model is estimated by classical cross-validation procedure (Dietterich 1998; Efron and Tibshirani 1986; Hastie et al. 2005; Kohavi 1995). If an external dataset (test/validation set) is available, the performance of model $\mathcal{M}$ is evaluated on that set.

3 Relaxed Multi-class LAD Algorithm

In this section we present an OvR-type extension of LAD algorithm to multi-class classification problems. As in conventional LAD algorithm our multi-class LAD approach has four steps: (i) binarization and support set selection, (ii) pattern generation, (iii) theory formation, and (iv) prediction. We refer to (Subasi and Avila-Herrera 2016) for the discussion of step (i) and focus on the pattern generation step for the relaxed OvR multiclass LAD approach.

3.1 Pattern Generation: MILP Based Approach

Let $\Omega = \Omega_1 \cup \cdots \cup \Omega_K$ be a K-class binary dataset with n features and m observations. Let $\mathcal{C} = \{\mathcal{C}_1, \cdots, \mathcal{C}_K\}$ denote the corresponding family of classes in Ω, that is, any observation in Ω_k has class $\mathcal{C}_k$ ($k = 1, \cdots, K$).

In order to formulate an MILP to generate a pattern $P_{\mathcal{C}_p}$ covering some of the observations in class $\mathcal{C}_p$ and none of the observations in $\mathcal{C}_k$, $k \neq p$, we proceed as follows:

(1) Associate a vector $\mathbf{y} = (y_1, \cdots, y_{2n}) \in \{0,1\}^{2n}$ to pattern $P_{\mathcal{C}_p}$, where the components $y_1, \cdots, y_{2n}$ of vector $\mathbf{y}$ are relative to the features such that if we have $y_j = 1$ for some $j = 1, \cdots, n$, then the literal x_j (associated with the j-th feature in Ω) is included in pattern $P_{\mathcal{C}_p}$ and if $y_{n+j} = 1$, then the literal $\bar{x}_j$ (complement of x_j) is included in pattern $P_{\mathcal{C}_p}$. Since a pattern cannot include both x_j and $\bar{x}_j$, we impose the condition

$$y_j + y_{n+j} \leq 1, \;\; j = 1, \cdots, n. \tag{3.4}$$

(2) Define a binary vector $\mathbf{w} = (w_1, w_2, \cdots, w_m)$ that is associated with the coverage of the pattern $P_{\mathcal{C}_p}$ and will be used to score penalization as follows: For $1 \leq i \leq m$

$$w_i = \begin{cases} 1 & \text{if } \phi_i \in \mathcal{C}_p \text{ is not covered by pattern } P_{\mathcal{C}_p} \\ 0 & \text{otherwise.} \end{cases}$$

(3) Consider the augmented matrix $B = [\Omega|\overline{\Omega}]$, where $\overline{\Omega}$ is the binary data obtained from Ω by replacing 0 entries by 1 and 1 entries by 0. Define the vector $\mathbf{v} = B\mathbf{y}$. In order to produce a pure pattern $P_{\mathcal{C}_p}$ with degree d we prescribe the following constraints:

$$v_i + n w_i \geq d, \quad i \in I_p \,, \tag{3.5}$$

$$v_i \leq d - 1, \quad i \in I_k \,,\; k = 1, \cdots, K \;\text{ and }\; k \neq p \tag{3.6}$$

and

$$\sum_{j=1}^{2n} y_j = d \,, \tag{3.7}$$

where $1 \leq d \leq n$, $I_p = \{i : \phi_i \text{ is in class } \mathcal{C}_p\}$ and $I_k = \{i : \phi_i \text{ is in class } \mathcal{C}_k\}$ for all $k \neq p$.

The conditions in (3.4)-(3.7) can be used to write an MILP whose optimal solution produces a pure pattern $P_{\mathcal{C}_p}$ associated with class $\mathcal{C}_p$ for some $1 \leq p \leq K$ as shown below:

$$\begin{aligned}
&\text{Minimize } z = d + \sum_{i \in I_p} w_i \\
&\text{subject to} \\
&v_i + n w_i \geq d, \quad i \in I_p \\
&v_i \leq d - 1, \quad i \in I_k\ ,\ k = 1, \cdots, K,\ \ k \neq p \\
&y_j + y_{n+j} \leq 1, \quad j = 1, 2, \cdots, n \\
&\sum_{j=1}^{2n} y_j = d; \qquad 1 \leq d \leq n \\
&w_i, y_j \in \{0, 1\},\ i = 1, \cdots, m;\ \ j = 1, \cdots, 2n
\end{aligned} \tag{3.8}$$

Notice that problem (3.14) is a modified version of the MILP problem (2.3) of Ryoo and Jang (2009) that is designed to generate patterns in a two-class dataset. An optimal solution of problem (3.14) can be used to form a pure strong prime pattern $P_{\mathcal{C}_p}$ associated with class $\mathcal{C}_p$, $1 \leq p \leq K$. The objective function of (3.14) ensures that the coverage of pattern $P_{\mathcal{C}_p}$ is maximized and the degree of $P_{\mathcal{C}_p}$ (i.e., the number of literals used in $P_{\mathcal{C}_p}$) is as small as possible.

3.2 Relaxed Multiclass LAD: *OvR Type Relaxed Multi-class LAD Method*

In this section we present an algorithm that produces an OvR-type multi-class LAD model based on the multi-class MILP approach given in Section 3.1. Note that in case of two-class MILP approach, Algorithm 1 of Ryoo and Jang (2009) (shown in Section 2.3) produces a set of patterns associated with a positive (negative) class that loops as many times as necessary until all observations in positive (negative) class are covered by at least one pattern. The setup proposed by Ryoo and Jang (2009) is inconvenient because a single pattern is sufficient to cover every observation in positive (negative) class which results in a classifier with small number of patterns and hence, poor differentiating power between the two classes of a dataset. In such cases the prediction of a new or unseen observation would depend on a single or a few patterns. Note also that once a positive (negative) pattern P is found as an optimal solution of problem (2.3), in order to produce a new positive (negative) pattern P', i.e., another optimal solution of problem (2.3), Algorithm 1 removes the observations covered by pattern P while execution. This is counterproductive because every time the algorithm uses less information (smaller training set) to compute new patterns. Mortada (2010) has adopted a similar approach to develop an OvO-type multi-class LAD algorithm, where observations covered by a pattern are removed from the training dataset while executing the proposed algorithm. The difference between Ryoo-Jang's algorithm (2009) and Mortada's algorithm (2010) is that in Mortada's algorithm the looping stops when each observation is covered by l patterns.

In order to avoid the removal of observations from the training dataset when generating new patterns that form a multi-class LAD model, (Avila-Herrera 2013) modified constraint (3.5) as follows:

Define κ as an m-vector that keeps track of the number of patterns covering an observation $\phi_i \in \Omega$ for all $i = 1, \cdots, m$. Initially, for each class $\mathcal{C}_k, 1 \leq k \leq K$ we set $\kappa = \mathbf{0}$. This vector shall be updated as new solutions of the MILP problem (3.14) are found. With the help of new vector κ, condition (3.5) can be replaced by

$$v_i + n\,(w_i + \kappa_i) \geq d, \quad i \in I_p\ . \tag{3.9}$$

where $\kappa_i \geq 0, i = 1, \cdots, m$.

3.3 Relaxed MILP LAD Modifications

The MILP model in the previous section whose optimal solution produces pure patterns can be modified to incorporate relaxed homogeneity and/or minimum prevalence preferences and generate robust patterns. Adopting the ideas from (Ryoo and Jang 2009) relaxed MILP approach and integrating it with (Subasi and Avila-Herrera 2016) multiclass MILP approach, problem (3.14) can be modified to generate a strong p -pattern with freedom given to ignore up to $(\alpha \times |\Omega_p|)\%$ of $\overline{p}$ observations outside class, where $|\Omega_p|$ is the number of observation in class p and $\alpha \in (0, 1)$, usually, $\alpha \leqslant 0.1$, by replacing constraint (3.5) with constraints $(3.10) - (3.12)$ below:

$$v_i - z_i \leq d - 1, \quad i \in I_k\ ,\ k = 1, \cdots, K,\ \ k \neq p \tag{3.10}$$

$$\sum_{i \in I_k} z_i \leq \alpha |\Omega_p|, \quad i \in I_k\ ,\ k = 1, \cdots, K,\ \ k \neq p \tag{3.11}$$

$$0 \leqslant z \leqslant 1, \tag{3.12}$$

Observe that these conditions together relax the homogeneity requirement of the pattern generated. Additionally, the problem can be further modified to incorporate a minimum prevalence requirement by the introduction of constraint (3.13)

$$\sum_{i \in I_p} w_i \leq (1 - \beta)|\Omega_p|, \quad i \in I_p, \tag{3.13}$$

Condition (3.13) imposes a prevalence requirement on the new MILP model that generates a pattern with the prevalence of β or better, where $\beta \in (0, 1)$, usually, $\beta \in [0.05, 0.2]$.

$$\begin{aligned}
&\text{Minimize } z = cd + \sum_{i \in I_p} w_i \\
&\text{subject to} \\
&v_i + n(w_i + k_i) \geq d, \quad i \in I_p \\
&\textstyle\sum_{i \in I_p} w_i \leq (1 - \beta)|\Omega_p|, \quad i \in I_p, \\
&v_i - z_i \leq d - 1, \quad i \in I_k\ ,\ k = 1, \cdots, K,\ \ k \neq p \\
&\textstyle\sum_{i \in I_k} z_i \leq \alpha |\Omega_p|, \quad i \in I_k\ ,\ k = 1, \cdots, K,\ \ k \neq p \\
&y_j + y_{n+j} \leq 1, \quad j = 1, 2, \cdots, n \\
&\sum_{j=1}^{2n} y_j = d; \qquad 1 \leq d \leq n \\
&w_i, z_i, y_j \in \{0, 1\},\ i = 1, \cdots, m;\ \ j = 1, \cdots, 2n
\end{aligned} \tag{3.14}$$

Note that in Algorithm 2 we do not require the removal of observations from the training dataset at any iteration by

Algorithm 2: Multi-class LAD Algorithm

Input: p: index of current class
Global data: Ω*: binary dataset, b: class vector*
Result: $\mathrm{MyPats}[p]$: patterns for class $\mathcal{C}_p$

$B = [\Omega|\overline{\Omega}]$;
$\mathbf{v} = B\,\mathbf{y}$; (* $\mathbf{y}$ unknown variable *)
$\mathrm{MyPats}[p] = \{\}$;
$\kappa = 0$;
$NewConstraint = \{\}$;
$TotCov = 0$;
while $TotCov < |I_p|$ **do**
  $\mathcal{R} = \{constraints from : (3.14)\} \cup NewConstraint$;
  $pat = Minimize \left[cd + \sum_{i \in I_p} w_i : \mathcal{R} \text{ and } \mathbf{v}, \mathbf{y}, \mathbf{w}, d \in \mathbb{Z} \right]$
  $\mathbf{y}^*$ part of pat corresponding to variables $\mathbf{y}$;
  for $i = 1$ **to** m **do**
    if $v_i = d$ **then**
      $\kappa_i = \kappa_i + 1$;
  $TotCov = 0$;
  for $i = 1$ **to** m **do**
    if $(i \in I_p) \wedge (\kappa_i \neq 0)$ **then**
      $TotCov = TotCov + 1$;
  $NotFound = True$;
  for $i = 1$ **to** m **do**
    if $(i \in I_p) \wedge (\kappa_i = 0) \wedge (v_i < d) \wedge (NotFound)$ **then**
      $NewConstraint = \{v_i = d\}$; (* d and Y as unknown variables *)
      $NotFound = False$;
  $\mathrm{MyPats}[p] = \mathrm{MyPats}[p] \cup \{\mathbf{y}^*\}$;
return $\mathrm{MyPats}[p]$;

adding NewConstraint to the relaxed MILP problem each time a new pattern is generated to prevent the algorithm from finding the same pattern found at the previous iterations. This is achieved by introducing that keeps track of the number of patterns covering observations $\phi_i \in \Omega$ and TotCov that counts the number of observations covered so far.

3.4 OvR Theory Formation

Given a $K-$class dataset $\Omega = \Omega_1 \cup \cdots \cup \Omega_K$ and a corresponding multi-class LAD model $\mathcal{M} = \mathcal{M}_1 \cup \cdots \cup \mathcal{M}_K$, ($\mathcal{M}_i \cap \mathcal{M}_j = \emptyset$, $i \neq j$), the classification of a new (or unseen) observation $o \notin \Omega$ is determined by the value of the discriminant function

$$\Delta(\phi) = \arg\max_k \Delta_k(\phi) \tag{3.15}$$

where $\Delta_k(\phi) = \sum_{P_{\mathcal{C}_k} \in \mathcal{M}_k} \omega_k P_{\mathcal{C}_k}(\phi), \quad k = 1, \cdots, K$ and $\omega_k \geq 0$ are the weights assigned to patterns $P_{\mathcal{C}_k} \in \mathcal{M}_k$. The weights $\omega_k, k = 1, \cdots, K)$ can be calculated in several ways. One possibility is to use the prevalence of patterns that is defined by $\omega_k = \frac{1}{|\Omega_k|} \sum_{i \in I_{\mathcal{C}_k}} P_{\mathcal{C}_k}(\phi_i)$, where $\Omega_k \subset \Omega$ is the set of observations in class $\mathcal{C}_k$ and $I_{\mathcal{C}_k} = \{i : \phi_i \in \Omega_k\}$ for some $1 \leq k \leq K$. If $\Delta(\phi) = \Delta_p(\phi) = \Delta_q(\phi)$ for some $p \neq q$, then the observation o is *unclassified*.

Similar to the two-class classification problem the accuracy of a multi-class model $\mathcal{M}$ is estimated by classical cross-validation procedure (Dietterich 1998; Efron and Tibshirani 1986; Hastie et al. 2005; Kohavi 1995). If an external dataset (test/validation set) is available, the performance of the model is evaluated on that set.

4 Experiments

In this section we present experimental results to show how Algorithm 2 described in Section 3.2 can be used for multiclass classication of publicly available multiclass datasets. Regarding the stopping criterion, Algorithm 2 ends once all patterns for each class have been computed. In the worst case, an adhoc pattern can be built by the algorithm to cover a single observation. Table 1 shows the characteristics of the datasets. In Table 2 we give the average accuracy of our proposed Relaxed Multiclass LAD Method for each class of the datasets. Finally, Table 3 presents the overall classification accuracy of all five datasets using the relaxed multiclass LAD method and six other LAD based multiclass classification methods. Note that our method produces comparable or better results on these datasets.

4.1 Experimental Results

In order to test our proposed multi-class LAD methodology we conduct experiments on five multi-class datasets from UCI Machine Learning Repository [1]. Table 1 summarizes the characteristics of these datasets. For each dataset the average accuracy of ten experiments. The average sensitivities per class for each dataset are shown in Tables 2 and 3.

Dataset	No. of Observations / Class	No. of Features
Iris	50 / 1, 50 / 2, 50 / 3	4
Glass ID	69 / 1, 76 / 2, 17 / 3, 13 / 4, 9 / 5, 29 / 6	10
Wine	59 / 1, 71 / 2, 48 / 3	12
E. Coli	143 / 1, 77 / 2, 52 / 3, 35 /4, 20 /5, 5 /6	34
Dermatology	112 / 1, 61 / 2, 72 / 3, 49 /4, 52 /5, 20 /6	19

Table 1: Five multi-class datasets from UCI repository

Dataset	C1	C2	C3	C4	C5	C6
Iris	90%	100%	100%			
Glass ID	89%	80%	30%	86%	89%	86%
Wine	98%	94%	88%			
E. Coli	95%	84%	60%	70%	60%	85%
Dermatology	95%	98%	93%	100%	92%	95%

Table 2: Average sensitivity

	Iris	Wine	Glass	E.Coli	Dermatology
Relaxed MC-LAD	**97.03 ± 1.90**	94.67 ± 2.14	80.37 ± 4.87	**82.5 ± 5.79**	**96.06 ± 2.85**
(a) MC-LAD	n.a	92.70 ± 2.54	62.41 ± 5.88	78.34 ± 3.40	89.07 ± 2.84
(b) C4.5	n.a	89.90 ± 3.11	62.80 ± 4.43	80.59 ± 4.14	94.48 ± 2.69
(c) MC- LAD	n.a	93.10 ± 3.20	65.00 ± 5.40	79.20 ± 4.20	n.a
(d) MC-LAD	94.00 ± 2.20	91.33 ± 3.54	79.54 ± 5.35	n.a	n.a
(e) IGA-FCMP	94.80 ± 0.40	96.18 ± 1.76	**96.26 ± 1.06**	75.07 ± 0.60	92.18 ± 0.43
(f) IGA-FCMP	95.73 ± 0.53	**96.86 ± 1.48**	93.46 ± 1.92	75.6	91.74

(a): MC-LAD in Moreira (2000) -OvO Type
(b): Data mining method in Moreira (2000) OvR type
(c) MC- LAD in Mortada et al. (2013) - OvO type
(d) MC-LAD in Herrera et al. (2013) - OvR type
(e) IGA-FCMP in Kim and Choi (2014) -OvR type
(f) IGA-FCMP in Kim and Choi (2014) OvO type

Table 3: Classification Accuracy of data sets (%)

5 Conclusions

In this paper we have proposed a multiclass LAD classification algorithm. Our construct has adopted the vision of Ryoo and Jang (2009) by using an MILP approach to generate LAD patterns and the modifications by Subasi and Avila-Herrera (2016) who proposed an algorithm that works properly with multiclass datasets. We have extended Subasi and Avila-Herrera's work to a relaxed multiclass LAD approach using homogeneity and prevalence as parameters. Our experiments on five benchmark multiclass datasets show that by themselves and in comparison to previously successful multiclass LAD classification methods the proposed relaxed multiclass LAD algorithm produces highly comparable and accurate classification models. Our multiclass methodology integrates principles from integer programming and computer related advancements to efficiently generate relaxed LAD patterns. It is a very promising option to solve multiclass classification problems.

References

Alexe, G., and Hammer, P. 2006. Spanned patterns for the logical analysis of data. *Discrete Applied Mathematics* 154(7):1039–1049.

Alexe, G.; Alexe, S.; Bonates, T. O.; and Kogan, A. 2007. Logical analysis of data–the vision of Peter L. Hammer. *Annals of Mathematics and Artificial Intelligence* 49(1):265–312.

Avila-Herrera, J. F. 2013. *Mixed Integer Linear Programming Based Implementations of Logical Analysis of Data and Its Applications*. Florida Institute of Technology, PhD Thesis.

Bishop, C. 2007. *Pattern Recognition and Machine Learning*. Springer.

Bonates, T., and Hammer, P. 2007. Pseudo-boolean regression. *RUTCOR Research Report* (3).

Bonates, T.; Hammer, P.; and Kogan, A. 2008. Maximum patterns in datasets. *Discrete Applied Mathematics* 156(6):846–861.

Boros, E.; Hammer, P.; Ibaraki, T.; and Kogan, A. 1997. Logical analysis of numerical data. *Mathematical Programming* 79(1):163–190.

Boros, E.; Hammer, P.; Ibaraki, T.; Kogan, A.; Mayoraz, E.; and Muchnik, I. 2000. An implementation of logical analysis of data. *Knowledge and Data Engineering, IEEE Transactions on* 12:2.

Burges, C. 1998. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery* 2(2):121–167.

Crama, Y.; Hammer, P. L.; and Ibaraki, T. 1988. Cause-effect relationships and partially defined boolean functions. *Annals of Operations Research* 16:299–326.

Dietterich, T. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation* 10(7):1895–1923.

Duda, R.; Hart, P.; and Stork, D. 2001. *Pattern Classification*. John Wiley & Sons, Inc.

Dupuis, C.; Gamache, M.; and Pagé, J. 2010. Logical analysis of data for estimating passenger show rates in the airline industry. Technical report, Working paper, École Polytechnique de Montréal. `www.agifors.org/award/submissions2010` on June 30.

Efron, B., and Tibshirani, R. 1986. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical science* 1(1):54–75.

Esmaeili, S. 2012. Development of equipment failure prognostic model based on logical analysis of data (lad).

Fausett, L. 1994. *Fundamentals of neural networks: architectures, algorithms, and applications*. Prentice-Hall Englewood Cliffs, NJ.

Guo, C., and Ryoo, H. 2012. Compact MILP models for optimal and pareto-optimal lad patterns. *Discrete Applied Mathematics* 160(16–17):2339–2348.

Hammer, P., and Bonates, T. 2006. *Logical analysis of data - An overview: From combinatorial optimization to medical applications*. Annals of Operations Research 148.

Hammer, P.; Kogan, A.; Simeone, B.; and Szedmák, S. 2004. Pareto-optimal patterns in logical analysis of data. *Discrete Applied Mathematics* 144(1):79–102.

Hammer, P.; Kogan, A.; and Lejeune, M. 2011. Reverse engineering country risk ratings: Statistical and combinatorial non-recursive models. *Annals of Operations Research* 188:185–213.

Hammer, P. 1986. Partially defined boolean functions and cause-effect relationships. *International Conference on*

Multi-attribute Decision Making Via OR-based Expert Systems.

Hastie, T., and Tibshirani, R. 1998. Classification by pairwise coupling. *The annals of statistics* 26(2):451–471.

Hastie, T.; Tibshirani, R.; Friedman, J.; and Franklin, J. 2005. The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer* 27(2):83–85.

Kim, H. H., and Choi, J. Y. 2015. Pattern generation for multi-class lad using iterative genetic algorithm with flexible chromosomes and multiple populations. *Expert Systems with Applications* 42(2):833–843.

Kohavi, R. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *International joint Conference on artificial intelligence*, volume 14, 1137–1145. Lawrence Erlbaum Associates Ltd.

Kotsiantis, S., and Kanellopoulus, D. 2006. Discretization techniques: A recent survey. *GESTS International Transactions on Computer Science and Engineering* 32:47–58.

Kwok, P. 2001. Methods for genotyping single nucleotide polymorphisms. *Annual Review of Genomics and Human Genetics* 2(1):235–258.

Lejeune, M., and Margot, F. 2011. Optimization for simulation: LAD accelerator. *Annals of Operations Research* 188(1):285–305.

Lemaire, P. 2011. Extensions of logical analysis of data for growth hormone deficiency diagnoses pseudo-boolean regression. *Annals of Operations Research* 186:199–211.

Liu, H.; Hussain, F.; Tan, C.; and Dash, M. 2004. Discretization: An enabling technique. *Data Mining and Knowledge Discovery* 393-423.

Moreira, L. 2000. *The use of Boolean concepts in general classification contexts*. Ph.D. Dissertation, Universidade do Minho, Portugal.

Mortada, M.; Yacout, S.; and Lakis, A. 2011. Diagnosis of rotor bearings using logical analysis of data. *Journal of Quality in Maintenance Engineering* 17(4):371–397.

Mortada, M. 2010. *Applicability and interpretability of logical analysis of data in condition based maintenance*. Ph.D. Dissertation, Polytechnique de Montréal.

Reddy, A. 2009. *Combinatorial pattern-based survival analysis with applications in Biology and Medicine*. Dissertation submitted to the Graduate School-New Brunswick Rutgers, The State University of New Jersey.

Ryoo, H., and Jang, I. 2009. MILP approach to pattern generation in logical analysis of data. *Discrete Applied Mathematics* 157(4):749–761.

Schölkopf, B., and Smola, A. 2001. *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press.

Subasi, M. M., and Avila-Herrera, J. F. 2016. Logical analysis of multiclass data. *2016 International Symposium on Artificial Intelligence and Mathematics* `http://isaim2016.cs.virginia.edu/papers.html`.

Subasi, E.; Subasi, M.; Hammer, P.; Roboz, J.; Anbalagan, V.; and Lipkowitz, M. 2017. A classification model to predict the rate of decline of kidney function. *Frontiers in Medicine* 4:97.