RUTCOR
RESEARCH
REPORT

A New Approach to Select Significant Patterns in Logical Analysis of Data

Juan Felix Avila Herrera [a] Munevver Mine Subasi [b]

RRR 31-2012, November 2012

RUTCOR
Rutgers Center for
Operations Research
Rutgers University
640 Bartholomew Road
Piscataway, New Jersey
08854-8003
Telephone: 732-445-3804
Telefax: 732-445-5472
Email: rrr@rutcor.rutgers.edu
http://rutcor.rutgers.edu/∼rrr

[a]Department of Mathematical Sciences, Florida Institute of Technology 150 W. University Blvd., Melbourne, FL 32901; Email: javilaherrer2009@my.fit.edu

[b]Department of Mathematical Sciences, Florida Institute of Technology 150 W. University Blvd., Melbourne, FL 32901; Email: msubasi@fit.edu

Rutcor Research Report
RRR 31-2012, November 2012

A New Approach to Select Significant Patterns in Logical Analysis of Data

Juan Felix Avila Herrera Munevver Mine Subasi

Abstract. Logical Analysis of Data (LAD) is a supervised learning algorithm which integrates principles of combinatorics, optimization and the theory of Boolean functions. Current implementations of LAD use greedy-type heuristics to select patterns to form an LAD model. In this paper we present a new approach based on integer programming and network flows to identify significant patterns to generate an LAD model. Our approach allows the user-specified significance requirements such as statistical significance, Hamming distance, homogeneity, coverage, and/or prevalence of patterns. We present experiments on benchmark datasets to demonstrate the utility of our integer programming and network flow based pattern selection method.

Acknowledgements: The authors thank Gabriela Alexe and Anupama Reddy for valuable discussions.

1 Introduction

With the advent of new technologies, the analysis of large-scale data has become ubiquitous in life sciences (biomarker detection in the fields of genomics and proteomics) and in virtually every industry ranging from manufacturing (test analysis), finance (fraud detection), marketing (customer relationship management), transportation (traffic analysis), telecommunication (churn analysis), and many other areas of human activity. The extraction of knowledge from large-scale data represents one of the fundamental challenges confronting researchers in these fields. In response to the need for data analysis in numerous disciplines, the efforts of researchers with diverse and analytic background have been channeled to this area. In order to explore, analyze, and interpret the information effectively and efficiently, the traditional statistical methods have been complemented by sophisticated data mining and machine learning methods including, support vector machines [12, 52], decision trees [46], neural networks [39, 41, 48, 49, 50], visualization techniques, and emerging technologies such as grid computing and web services.

Logical Analysis of Data (LAD) is a pattern-based two-class learning algorithm which integrates principles of combinatorics, optimization and the theory of Boolean functions. The research area of LAD was introduced and developed by Peter L. Hammer [25] whose vision expanded the LAD methodology from theory to successful data applications in numerous biomedical, industrial, and economics case studies (see, e.g., [27, 47] and the references therein). The implementation of LAD algorithm was described in [10], and several further developments of the original algorithm were presented in [3, 4, 6, 9, 18, 26, 23, 51]. An overview of LAD algorithm can be found in [5, 27]. Various applications of LAD are presented in [16, 20, 30, 40, 42]. LAD algorithm has been recently extended to survival analysis [32].

In many data analysis problems a "dataset" $\mathbb{D}$ consists of two disjoint sets $\mathbb{D}^+$ and $\mathbb{D}^-$ of n-dimensional real vectors. Typically each of the vectors in the dataset corresponds to observations (or samples), where the vectors in $\mathbb{D}^+$ corresponding to observations having a specific condition (e.g., patients with specific disease) are called positive observations, and the vectors in $\mathbb{D}^-$ corresponding to those observations that do not have the condition (e.g., patients not having the disease) are called negative observations. The components of the vectors, called "features" (or alternatively, attributes/variables), can represent the results of certain measurements, for example, medical tests, the expression levels of genes or proteins, etc. in medical datasets. Given a new or unseen observation, i.e., a vector which is neither in $\mathbb{D}^+$ nor in $\mathbb{D}^-$, one usually has to determine whether this vector should be classified as positive or negative. The main task in classification problems is to extract useful information from the dataset to be able to recognize the positive or negative nature of new observations.

The key ingredient of the LAD algorithm is the identification of patterns, i.e., complex rules distinguishing between positive and negative observations in the dataset. Given a dataset, $\mathbb{D}$, LAD algorithm usually produces several hundreds (sometimes thousands) of patterns. Once all patterns are generated, greedy-type heuristics are used to select patterns 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) to generate an LAD

classification 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. In this paper we propose a new approach based on integer programming and network flows to select significant patterns to generate an LAD model. Our approach allows the user-specified significance requirements such as statistical significance, Hamming distances to ideal patterns, homogeneity, coverage, and/or prevalence of patterns.

The organization of this paper is as follows. Section 2 describes the basic principles of LAD algorithm. In Section 3 we present our integer programming and network flow based pattern selection method. In Section 4 we evaluate, through several experiments on artificial and benchmark datasets, the accuracy of LAD classification models built using our proposed approach, as compared to the accuracy of greedy heuristic based LAD models.

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, $\mathbb{D}$, consists of two disjoint classes $\mathbb{D}^+$ (set of positive observations) and $\mathbb{D}^-$ (set of negative observations), that is, $\mathbb{D} = \mathbb{D}^+ \cup \mathbb{D}^-$ and $\mathbb{D}^+ \cap \mathbb{D}^- = \emptyset$. The main task of LAD algorithm is to identify complex rules separating the positive and negative observations based on features measured [10, 11]. Below we briefly outline the basic components of the LAD algorithm. A more detailed overview can be found in [5, 27].

2.1 Discretization/Binarization

This step is the transformation of numeric features 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 [11, 27]. 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, for example, the survey papers [36, 31]).

2.2 Support Set

Discretization step may produce several binary features some of which may be redundant. Support set is defined as a 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 [11, 27].

2.3 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. A pattern P can be described as a conjunction of binary features associated with numeric features. Patterns define homogeneous subgroups of observations with distinctive characteristics. An observation satisfying the conditions of a pattern is said to be *covered* by that pattern. A positive (negative) pattern is defined as a combination of features which covers a large proportion of positive (negative) observations, but only a few of negative (positive) ones. A pure positive (negative) pattern is one which covers only positive (negative) observations.

In order to have the flexibility of generating patterns which are not necessarily pure, LAD algorithm associates important characteristics to patterns: (*i*) *Degree* of a pattern is the number of features (or conditions) involved in the definition of the pattern. (*ii*) *Prevalence* of a positive (negative) pattern with respect to a given set of observations is the percentage of positive (negative) observations that are covered by that pattern. (*iii*) *Homogeneity* of a positive (negative) pattern is the percentage of positive (negative) observations among the set of observations covered by it. A large collection of positive and negative patterns with given degree, prevalence, and homogeneity, may be generated by a combinatorial enumeration process (see, e.g., [11]). The parameters (degree, prevalence, and homogeneity) are calibrated using cross-validation experiments.

2.4 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 $\mathcal{P}^+$ (respectively, $\mathcal{P}^-$) is the set of all positive (respectively, negative) patterns and $\mathcal{P}^+ \cap \mathcal{P}^- = \emptyset$). In many cases, when constructing an LAD model, every observation in the training dataset is required to be covered at least $k, (k \in \mathbb{Z}^+)$, times 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 LAD algorithm uses greedy heuristics to solve the set-covering problem to generate an LAD model.

2.5 Classification and Accuracy

Given an LAD model $\mathcal{M} = \mathcal{M}^+ \cup \mathcal{M}^-$, the classification of a new (or unseen) observation $o \notin \mathbb{D}$ is determined by the sign a discriminant function $\Delta : \{0,1\}^n \to \mathbb{R}$ associated to the model $\mathcal{M}$, where $\Delta(o)$ is defined as the difference between the proportion of positive patterns

and negative patterns covering o, that is,

$$\Delta(o) = \frac{u}{|\mathcal{M}^+|} - \frac{v}{|\mathcal{M}^-|},$$

where u and v denote the number of positive patterns and negative patterns covering o, respectively.

The accuracy of the model is estimated by classical cross-validation procedure [17, 19, 28, 29]. If an external dataset (test set) is available, the performance of model $\mathcal{M}$ is evaluated on that set.

2.6 Software Implementations

There exist several implementations of the Logical Analysis of Data algorithm: Datascope [7], Ladoscope [34], Cap-LAD [9], and LFW [8]. In our experiments we mainly use Ladoscope and its companion LFW.

In this paper we propose a new approach to select significant patterns from the pandect $\mathcal{P}$ to generate an LAD model $\mathcal{M}$. Our approach allows the user-specified significance requirements including statistical significance of patterns as well as other pattern characteristics (homogeneity, coverage, prevalence, etc.). In what follows we describe the details of our proposed methodology.

3 A New Approach to Generate LAD Models

As discussed in Section 2, the pandect (collection of all patterns with given degree, prevalence, and homogeneity) used in the implementation of LAD is generated by a combinatorial enumeration process [11]. Even with the best choice of the values of control parameters the size of the pattern collection produced is very large and requires in most cases the application of a filtering procedure, which selects small subsets of patterns to form highly accurate predictive models.

In this section we propose a discrete optimization and network flow based algorithm to select the significant patterns from the pandect to generate an LAD model and show that the accuracy of LAD models based on these patterns is highly competitive with (if not better than) that of the original LAD models generated by greedy heuristics.

Given a dataset $\mathbb{D}$ with m observations, assume that the pandect $\mathcal{P}$, containing all positive and negative patterns with given characteristics (degree, homogeneity, and prevalence), is generated. Let $o_1, \cdots, o_m \in \mathbb{R}^n$ designate the observations in $\mathbb{D}$ and assume that $|\mathcal{P}| = p$. Consider the pattern-observation incidence matrix defined by $B = (b_{ij})_{p\times m}$ with entries $b_{ij} = 1, i = 1, \cdots p, j = 1, \cdots m$ if pattern $P_i \in \mathcal{P}$ covers observation $o_j \in \mathbb{D}$, and $b_{ij} = 0$ otherwise.

Define the decision variables

$$x_{ij} = \begin{cases} 1 & \text{if } P_i \text{ covers } o_j \\ 0 & \text{otherwise} \end{cases}, \quad i = 1, \cdots, p, \quad j = 1, 2, \cdots m. \tag{1}$$

Let us associate a variable $y_i, i = 1, \cdots, p$, to each pattern $P_i \in \mathcal{P}$ as the number of observations covered by the pattern P_i. Similarly, let $z_j, j = 1, \cdots, m$, denote the number of patterns covering observation $o_j \in \mathbb{D}$.

We formulate the following integer programming problem which is the starting point of our investigation:

$$
\begin{aligned}
& \text{maximize} \sum_{i=1}^{p} y_i \\
\text{subject to} \quad & \\
& y_i = \sum_{j=1}^{m} x_{ij}, \quad i = 1, \cdots, p \\
& z_j = \sum_{i=1}^{p} x_{ij}, \quad j = 1, \cdots, m \\
& \sum_{j=1}^{m} z_j = \sum_{i=1}^{p} y_i \\
& 0 \leq x_{ij} \leq b_{ij}, \quad i = 1, \cdots, p\ , \quad j = 1, \cdots, m. \\
& 0 \leq y_i \leq m, \quad i = 1, \cdots, p \\
& 0 \leq z_j \leq k, \quad j = 1, \cdots, m \\
& x_{ij} \in \{0, 1\}, \quad y_i, z_j \in \mathbb{Z}, \quad i = 1, \cdots, p\ , \quad j = 1, \cdots, m
\end{aligned}
\tag{2}
$$

where m is the number of observations in $\mathbb{D}$, p is the number of patterns in pandect $\mathcal{P}$, and $k \in \mathbb{Z}^+ (1 \leq k \leq p)$ is a constant.

Note that, given a dataset $\mathbb{D}$ with m observations and the corresponding pandect $\mathcal{P}$, $(|\mathcal{P}| = p)$ generated by the implementation of LAD algorithm, problem (2) produces an LAD model consisting of patterns with maximum coverage from $\mathcal{P}$, where each observation is covered at least once. However, it does not necessarily select patterns based on their significance set by data analyst.

In order to allow the selection of significant patterns into an LAD model based on user-specific significance requirements, we set an order relation $\prec$ on the pandect $\mathcal{P}$. Let $\mathcal{P}_o$ designate the set of all patterns ordered according to $\prec$. We shall present an algorithm that produces a minimal subset $\mathcal{M}^* \subseteq \mathcal{P}_o$ forming an LAD model where each observation is covered by at least one pattern. If P_i, P_j are two distinct patterns in $\mathcal{P}_o$ such that $P_i \prec P_j$, our algorithm chooses P_j before P_i unless there is a conflict regarding the coverage of all observations. In order to achieve this goal we integrate ideas and principles from network flow theory as described below.

3.1 Obtaining an LAD model by using network flows

In this section we present an algorithm to choose a minimal subset of patterns $\mathcal{M}^*$ from the ordered collection of patterns $\mathcal{P}_o$ such that every observation in $\mathbb{D}$ is covered at least once. Some of the possible significance requirements on patterns to be selected from $\mathcal{P}_o$ to form

an LAD model will be outlined later on in this paper.

Let $G = (V, E)$ denote a directed bipartite graph where $V = V_p \cup V_m$ with $V_p \cap V_m = \emptyset$ and the nodes in partitions V_p and V_m represent the patterns in $\mathcal{P}_o$ and the observations in $\mathbb{D}$, respectively. There exists an arc $(i, j) \in E$ with $i \in V_p$ and $j \in V_m$ if the ith pattern P_i in $\mathcal{P}_o$ covers observation o_j in $\mathbb{D}$. Hence, we have $|V_p| = p$ and $|V_m| = m$. Let us expand G into a weighted network G' introducing a super-source s and a super-sink t and adding arcs $(s, i),\ i = 1, ..., p$, with capacities equal to m and arcs $(j, t),\ j = 1, ..., m$, with capacities equal to k $(1 \leq k \leq p)$. Assume also that for $i = 1, ..., p, j = 1, ..., m$ capacities of arcs $(i, j) \in E$ are set to 1. Network G' is shown in Figure 1.

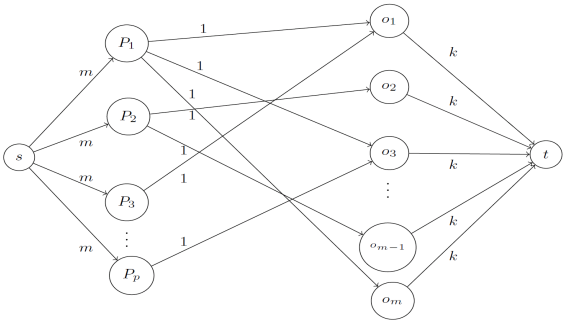

Figure 1: Network G' corresponding to problem (2)

With this definition of G' problem (2) is transformed into a network flow problem where we determine the maximum flow sent from source s to sink t in network G'. Note that each arc from source s to node $i \in V_p, i = 1, ..., p$, representing a pattern P_i, has capacity equal to m because a pattern in pandect $\mathcal{P}_o$ could cover at most m observations. Similarly, the capacities of the arcs of type $(j, t) \in E, j = 1, ..., m$, ensure that each observation $o_j \in \mathbb{D}$ is covered at least once. We remark that it is easy to construct the network G' using the pattern-observation incidence matrix obtained from pandect $\mathcal{P}_o$ by the implementation of LAD algorithm. In fact, the incidence matrix is readily available as an output of various LAD software packages including Datascope [7], Ladoscope [34], and LFW [8].

For the sake of simplicity, let us relabel the nodes of network G' as shown in Figure 2, where $q = p + m$. The node-arc adjacency matrix, M, of network G' is given by

$$M = \begin{array}{c|c||c|c|c|c|c|c|c} & 1 & 2 & \cdots & p+1 & p+2 & \cdots & q+1 & q+2 \\ \hline\hline 1 & 0 & m & \cdots & m & 0 & \cdots & 0 & 0 \\ \hline 2 & 0 & 0 & \cdots & 0 & b_{11} & & b_{1m} & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ p+1 & 0 & 0 & \cdots & 0 & b_{p1} & & b_{pm} & 0 \\ \hline p+2 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & k \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ q+1 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & k \\ \hline q+2 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 \end{array}$$

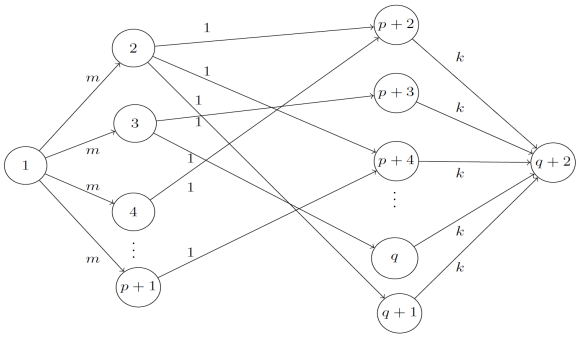

Figure 2: Relabeled Network G'

where the top row and left most column represent the labels of the nodes in G' and entries $b_{ij}, i = 1, ..., p, j = 1, ..., m$, are defined as before.

In Algorithm 1 we present a systematic way of generating an LAD model $\mathcal{M}^*$ that is the minimal subset of the collection $\mathcal{P}_0$ of all patterns ordered according to a specific significance requirement set by the data analyst. This is equivalent to the problem of finding the maximum flow sent from source s to sink t in network G' [1, 14]. The proposed algorithm systematically searches for augmenting paths from s to t of the form $s \to P_i \to o_j \to \cdots \to t$ until there is none available. The algorithm is specifically designed to choose a pattern P_j before another pattern P_i, whenever P_j is more significant than P_i, that is, $P_i \prec P_j$. Although the size p of pandect $\mathcal{P}_o$, and hence, the dimension $(q + 2) \times (q + 2)$ of the adjacency matrix M, is large in general, the computational effort of our proposed algorithm will be reduced because M is a sparse matrix.

3.2 User-specific significance requirements on patterns

Algorithm 1 uses the collection $\mathcal{P}_o$ of patterns where patterns are assumed to be ordered according to a specific significance requirement. In this section we shall describe a few possible significance requirements set on patterns so that patterns are selected depending on their significance from the pandect to form an LAD model. The possibilities are not limited to the ones described below and could be chosen by the data analyst based on the problem under study.

Later on we shall evaluate, through several experiments on artificial and benchmark datasets, the accuracy of LAD classification models built using our proposed approach, as compared to the accuracy of greedy heuristic based LAD models.

3.2.1 Sorting patterns according to their statistical significance

Statistical significance is used to determine whether the outcome of an experiment is the result of a relationship between specific factors or merely the result of chance. The concept is commonly used in the fields of bioinformatics, psychology, biology, and other experimental

Algorithm 1: Generating an LAD model $\mathcal{M}^*$

Data: $G' = (V', E')$: Network in Figure 2 ;
m: No. of observations;
p: No. of patterns;
$\mathcal{P}_o$: Ordered set of all patterns;
B: Pattern-observation incidence matrix;
Result: LAD model $\mathcal{M}^*$

```
Initialize the flow function f(u, v) = 0 at each edge (u, v) ∈ E;
M* = {};
for i = 1 to p do
    for j = 1 to m do
        if b_ij = 1 then
            for all path [ρ = (s → P_i → o_j → ··· → t)] do
                x = maximum amount of flow that can be sent through path ρ;
                if x > 0 then
                    Increase flow f on G' by x using the path ρ;
for i = 2 to p + 1 do
    if f(1, i) > 0 then
        M* = M* ∪ {P_i}
return M*;
```

sciences. Statistical significance is measured by p-value that is referred to as *significance level*. p-value is simply the likelihood that the observed relationship between two variables occurred by chance. Depending on the nature of the problem under study, p-values corresponding to a certain experiment can be calculated by various statistical tests including, for example, Fisher's Exact Test [21, 22], t-Test [37, 44], χ^2-Test [13, 35, 45], Wilcoxon Test [54, 33, 38], etc.

We shall adopt this idea to assign a "significance level" to each pattern generated by LAD algorithm as follows:

Step 1. Assume that, given a dataset $\mathbb{D}$, an implementation of LAD algorithm has produced the set of all possible patterns with given degree, homogeneity, and prevalence. Let $\mathcal{P}$ denote the resulting collection of patterns.

Step 2. Obtain the pattern-observation matrix B from pandect $\mathcal{P}$.

Step 3. Let $\mathcal{C}$ designate the class vector corresponding to the dataset $\mathbb{D}$. For each $P_i \in \mathcal{P}_o$ $(i = 1, ..., p)$ run a significance test (say, Fisher's Exact Test) to compare the vector P_i (ith row of matrix B) and the vector $\mathcal{C}$.

Step 4. Assign each pattern P_i $(i = 1, ..., p)$ a significance score which is the p-value provided by the significance test.

Assume that pandect $\mathcal{P}_o$ is obtained from the collection $\mathcal{P}$ of all patterns by ordering them according to their p-values in increasing order (i.e., the first pattern in $\mathcal{P}_o$ is "statistically" the most significant one among all patterns in $\mathcal{P}$). We refer to this approach as *sorting patterns according to their p-values* (SPAPV). Algorithm 1 with input $\mathcal{P}_o$ and the corresponding network G' generates an LAD model where the patterns are selected from the collection of all patterns based on their statistical significance.

3.2.2 Sorting patterns according to their distance to ideal patterns

In this section we introduce another significance requirement based on Hamming distance. Assume that, given a dataset $\mathbb{D}$, an implementation of LAD algorithm has produced the pandect $\mathcal{P}$ and hence, the pattern-observation incidence matrix B as discussed earlier. An *ideal positive (negative) pattern* can be defined as a pattern with 100% positive (negative) homogeneity and 100% positive (negative) prevalence. If LAD algorithm produces an ideal positive pattern, then the row of the incidence matrix B corresponding to that pattern is same as the class vector $\mathcal{C} \in \{0,1\}^m$ in dataset $\mathbb{D}$. Similarly, the vector $\tilde{\mathcal{C}}$ whose components are defined by

$$\tilde{\mathcal{C}}_j = \begin{cases} 1 & \text{if} \quad \mathcal{C}_j = 0 \\ 0 & \text{if} \quad \mathcal{C}_j = 1 \end{cases}, \quad j = 1, ..., m$$

would be the row of B corresponding to an ideal negative pattern.

When analyzing real-world datasets it is very unlikely to obtain "ideal patterns" because such datasets are usually noisy and/or may contain measurement errors. However, we may determine how close a pattern to an ideal one by using Hamming distance as described below:

Step 1. Assume that, given a dataset $\mathbb{D}$, an implementation of LAD algorithm has produced the pandect $\mathcal{P} = \mathcal{P}^+ \cup \mathcal{P}^-$, where $\mathcal{P}^+$ is the collection of all positive patterns, $\mathcal{P}^-$ is the collection of all negative patterns, and $\mathcal{P}^+ \cap \mathcal{P}^- = \emptyset$.

Step 2. Obtain the pattern-observation matrix B corresponding to $\mathcal{P}$.

Step 3. Let $\mathcal{C}$ and $\tilde{\mathcal{C}}$ designate the binary vectors corresponding to ideal positive and negative patterns as described above. For each row of the pattern-incidence matrix B find the Hamming distance:

$$d(\mathcal{C}, B_i) \quad \text{for all } i \text{ such that } P_i \in \mathcal{P}^+$$

$$d(\tilde{\mathcal{C}}, B_i) \quad \text{for all } i \text{ such that } P_i \in \mathcal{P}^-,$$

where $B_i \in \{0,1\}^m$ is the ith row of B and $d(X, Y)$ is the Hamming distance between $X, Y \in \{0,1\}^m$ that is equal to the number of entries on which they differ.

Step 4. Assign each pattern P_i $(i = 1, ..., p)$ a significance score which is the corresponding Hamming distance found in Step 3.

Assume that pandect $\mathcal{P}_o$ is obtained from the collection $\mathcal{P}$ of all patterns by ordering them according to their Hamming distances in increasing order. This approach is referred to as *sorting patterns according to their distance to ideal patterns* (SPADIP). In this case Algorithm 1 with input $\mathcal{P}_o$ and the corresponding network G' generates an LAD model where the patterns are selected from the collection of all patterns based on their Hamming distances to the ideal pattern.

3.2.3 Sorting patterns according to homogeneity

As discussed earlier an implementation of LAD algorithm produces patterns with given characteristics (homogeneity, prevalence, and degree). In LAD algorithm parameters "homogeneity and prevalence" are implemented as lower bounds on the homogeneity and prevalence of the patterns to be produced whereas "degree" is an upper bound on the number of features to be used in the patterns. For example, if positive homogeneity of a pattern is set to 80%, then the algorithm would produce various positive patterns whose homogeneities would be ranging from 80%-100%.

Algorithm 1 can be used to systematically select high quality patterns to form an LAD model. In this case the patterns in pandect $\mathcal{P} = \mathcal{P}^+ \cup \mathcal{P}^-$ corresponding to a dataset $\mathbb{D}$ are ordered according to their homogeneity in decreasing order. We use the prevalence of the patterns as a tie-breaking rule. We call this approach *sorting our patterns according to their homogeneity* (SPAH). Let $\mathcal{P}_o$ denote the resulting pandect. With input $\mathcal{P}_o$ and the corresponding network G' Algorithm 1 generates an LAD model where the patterns are selected from the collection of all patterns based on their pattern characteristics. When assigning a significance score to patterns in pandect $\mathcal{P}$ another pattern characteristic could be chosen as hazard ratio of the patterns.

3.2.4 Sorting patterns by combining various significance requirements

Our last approach takes advantage of the synergistic effects of different sorting approaches described above. The sorting procedures SPAPV, SPADIP, and SPAH assign different scores to a pattern $P_i \in \mathcal{P}$. We normalize the scores associated with a pattern so that each score lies between 0 and 1. We then sort patterns according to their maximum score obtained by SPAPV, SPADIP, and SPAH to obtain the pandect $\mathcal{P}_o$ and run Algorithm 1 with input $\mathcal{P}_o$ and the corresponding network G' to form an LAD model where the patterns are selected from the collection of all patterns based on their maximum scores. The approach of *sorting patterns according to their maximum scores* is abbreviated as SPAM.

4 Experiments

4.1 An Overview of Experiments

In order to test our proposed method we conduct experiments on artificial and benchmark datasets, and compare the accuracy of LAD classification models built using our proposed

Dataset	# Pos. Observations	# Neg. Observations	# Features
WBC	458	241	9
BLD	200	145	6
AGD	50	50	20

Table 1: Characteristics of Datasets

approach with the accuracy of greedy heuristic based LAD models. For each dataset $\mathbb{D}$ and each of the sorting approach described in Section 3, we randomly selected 90% of the dataset (90% of positive observations and 90% of negative observations are selected from $\mathbb{D}$) as the training set and the remaining 10% as test set. We then apply Algorithm 1 combined with sorting procedures SPAPV, SPADIP, SPAH, and SPAM to generate an LAD model on the training data. The accuracy of the resulting network flow based LAD model is validated on the test set. For each dataset we also build a greedy heuristic based LAD model on the training set and validate it on the test set. These steps are outlined below:

(*i*) Divide $\mathbb{D} = \mathbb{D}_{TR} \cup \mathbb{D}_{TS}$, where $\mathbb{D}_{TR}$ is the training set, $\mathbb{D}_{TS}$ is the test set, and $\mathbb{D}_{TR} \cap \mathbb{D}_{TS} = \emptyset$.

(ii) Run the LAD algorithm on the set $\mathbb{D}_{TR}$ to obtain pandect $\mathcal{P}$ containing all patterns with given homogeneity, prevalence, and degree.

(*iii*) Use a greedy approach to select patterns from $\mathcal{P}$ to form an LAD model on $\mathbb{D}_{TR}$ and compute the accuracy of the resulting greedy-heuristic based LAD model on $\mathbb{D}_{TS}$.

(*iv*) Use Algorithm 1 and sorting methods SPAPV, SPADIP, SPAH, and SPAM (one at a time) to select patterns from $\mathcal{P}$ and form an LAD model on $\mathbb{D}_{TR}$. Compute the accuracy of the resulting LAD model on $\mathbb{D}_{TS}$.

(*v*) Compare the accuracies of the models obtained in (iii) and (iv).

For each dataset $\mathbb{D}$ we repeat steps (i)-(v) ten times, each time randomly partitioning the dataset into subsets $\mathbb{D}_{TR}$ and $\mathbb{D}_{TS}$.

4.2 Datasets

In our experiments we consider two benchmark datasets "Wisconsin Breast Cancer (WBC)" and "Bupa Liver Disease (BLD)" from UCI Machine Learning Repository [53]. It is known from the literature that WBC is a clean dataset on which most data analysis methods provide highly accurate classification models. On the other hand, BLD is known to be very noisy and it is very hard to find accurate models for this dataset. We also generate a two-class artificial data (AGD) following Gaussian distribution as shown in Figure 3. Table 1 describes the characteristics of these datasets.

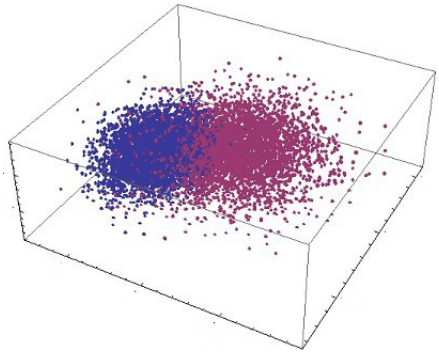

Figure 3: Two-Class Artificial Data following Gaussian Distribution

4.3 Experimental Results

In this section we present experimental results obtained by the use of techniques described in Section 4.1. For each dataset (and for each experiment) the pandect is obtained by using software Ladoscope [34] and its companion LFW [8] with given parameters (homogeneity, prevalence, and degree). For all data-sets Tables 2–4 show the accuracies of the LAD models obtained by greedy approach as well as by Algorithm 1 based on significance requirements SPAPV, SPADIP, SPAH, SPAM. In those tables we do not claim that the accuracies are better than the previously reported accuracies for WBC and BLD datasets in literature, but rather compare the LAD models obtained by Algorithm 1 based on the user-specific significance requirements on patterns and the model obtained by classical greedy type heuristics (as implemented in various LAD software [7, 34, 8]).

Experiment	GREEDY	np	SPAPV	np	SPADIP	np	SPAH	np	SPAM	np
1	94.83%	6	94.83%	10	94.83%	12	91.38%	19	93.10%	21
2	94.83%	4	94.83%	8	93.97%	9	88.79%	22	93.10%	17
3	93.97%	6	94.83%	7	93.10%	11	93.10%	21	93.10%	19
4	89.66%	5	94.83%	10	93.10%	10	93.10%	23	93.10%	19
5	92.24%	8	93.97%	14	93.97%	13	93.10%	25	91.38%	21
6	88.79%	6	94.83%	12	94.83%	11	93.10%	27	93.10%	22
7	93.10%	5	94.83%	11	94.83%	11	93.10%	24	94.83%	18
8	89.66%	6	94.83%	13	94.83%	11	93.10%	26	93.10%	21
9	94.83%	5	94.83%	12	93.10%	12	93.10%	27	93.10%	18
10	94.83%	5	93.10%	11	93.10%	12	93.10%	24	93.10%	18
Average	92.67%	5.6	94.57%	10.8	93.97%	11.2	92.50%	23.8	93.10%	19.4

Table 2: Accuracies of different LAD models on WBC dataset. (np: number of patterns)

It appears from the results of the experiments that our proposed method achieves comparable accuracies to (if not better than) those obtained by greedy approach. For the bencmark datasets WBC and PLD the accuracies of the models obtained by Algorithm 1 where patterns are chosen based on their statistical significance, are, on average, better than the accuracies of the other models. For the artificial data, SPAPV approach gives, on average, the worst

Experiment	GREEDY	np	SPAPV	np	SPADIP	np	SPAH	np	SPAM	np
1	53.13%	13	53.13%	16	53.13%	11	53.13%	15	53.13%	16
2	64.06%	6	60.94%	7	59.38%	5	60.94%	7	62.50%	6
3	54.69%	7	57.81%	11	57.81%	6	56.25%	10	57.81%	11
4	62.50%	5	62.50%	6	62.50%	6	62.50%	4	62.50%	7
5	54.69%	8	64.06%	9	62.50%	7	60.94%	8	62.50%	11
6	53.13%	12	62.50%	10	60.94%	9	57.81%	10	60.94%	14
7	60.94%	8	64.06%	9	64.06%	7	62.50%	11	65.63%	9
8	59.38%	8	57.81%	10	59.38%	6	59.38%	8	64.06%	10
9	60.94%	7	65.63%	6	65.63%	5	65.63%	6	65.63%	8
10	53.13%	8	53.13%	11	53.13%	8	54.69%	9	54.69%	10
Average	57.66%	8.2	60.16%	9.5	59.84%	7	59.38%	8.8	60.94%	10.2

Table 3: Accuracies of different LAD models on BLD dataset. (np: number of patterns)

Experiment	GREEDY	np	SPAPV	np	SPADIP	np	SPAH	np	SPAM	np
1	95%	3	90%	6	95%	4	95%	7	90%	4
2	100%	3	90%	9	100%	4	95%	8	100%	4
3	100%	3	100%	8	95%	5	100%	10	100%	6
4	95%	3	90%	9	85%	5	95%	8	95%	6
5	90%	3	90%	8	85%	6	90%	8	90%	6
6	95%	3	95%	9	95%	6	95%	9	95%	6
7	95%	3	90%	8	95%	6	95%	10	95%	6
8	85%	3	85%	8	90%	5	85%	9	85%	5
9	85%	3	80%	9	85%	6	85%	11	85%	6
10	80%	3	70%	9	80%	5	80%	11	80%	5
Average	92%	3	88%	8.3	90.50%	5.2	91.50%	9.1	91.50%	5.4

Table 4: Accuracies of different LAD models on AGD dataset. (np: number of patterns)

accuracy as compared to the other approaches.

5 Conclusion

LAD is an exciting supervised learning algorithm that allows data analysts to identify complex rules (combinatorial patterns) separating two classes in a dataset. Various real-world applications have shown that the method produces robust and highly accurate classification models obtained by greedy-type heuristics as implemented in LAD software packages. In this paper we propose a systematic way of integrating principles from discrete optimization and network flows to generate LAD models where the patterns are selected based on user-specific significance requirements. The empirical results show that the performance of our approach is comparable with greedy type approaches. In addition to producing accurate LAD models, it gives data analysts the flexibility to set specific requirements such as statistical significance of patterns. One drawback of our method may be considered as computational time of Algorithm 1 which is implemented to work as Ford-Fulkerson's algorithm [1, 14] in a downward manner. However, the network flow problem we define is to find the maximum flow in a bipartite graph and Gusfield et al. [24] showed that the standard augmenting path algorithms are more efficient in unbalanced bipartite graphs, while Ahuja et al. [2] showed

small modifications to existing push-relabel algorithms yielded better time bounds (also, see [43]).

References

[1] R.K. Ahuja, T.L. Magnanti, J.B. Orlin, Network Flows, Prentice Hall, Upper Saddle River, New Jersey, 1993.

[2] R.K. Ahuja, J.B. Orlin, C. Stein, R.E. Tarjan, Improved algorithms for bipartite network flow problems, SIAM J. of Comp. 23 (1994), 906–933.

[3] G. Alexe, P.L. Hammer, Spanned patterns for the logical analysis of data, Discrete Appl. Math. 154 (2006), 1039–1049.

[4] G. Alexe, S. Alexe, P.L. Hammer, A. Kogan, Comprehensive vs. comprehensible classifiers in logical analysis of data, Discrete Applied Mathematics 156 (2008), 870–882.

[5] G. Alexe, S. Alexe, P.L. Hammer, A. Kogan, Logical analysis of data - the vision of Peter L. Hammer, Annals of Math Artif Intell 49 (2007), 265–312.

[6] S. Alexe, P.L. Hammer, Accelerated algorithm for pattern detection in logical analysis of data, Discrete Applied Mathematics, 154 (2006), 1050–1063.

[7] S. Alexe, Datascope: An Implementation of Logical Analysis of Data Methodology
`http://rutcor.rutgers.edu/~salexe/LAD_kit/SETUP-LAD-DS-SE20.zip`

[8] J.F. Avila-Herrera, Integer Programming Applied to Rule Based Systems, Procedia Computer Science 9 (2012), 1553–1562.

[9] T.O. Bonates, P.L. Hammer, A. Kogan, Maximum patterns in datasets, Discrete Applied Mathematics 156 (2008), 846–861.

[10] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, Logical analysis of numerical data, Mathematical Programming 79 (1997), 163–190.

[11] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz and I. Muchnik. An implementation of logical analysis of data, IEEE Trans. on Knowledge and Data Engineering 12 (2000), 292–306.

[12] C.J.C. Burges, A tutorial on support vector machines for pattern recognition, Data Mining and Knowledge Discovery 2 (1998), 121–167.

[13] H. Chernoff, E.L. Lehman, The use of maximum likelihood estimates in chi2 tests for goodness-of-fit, The Annals of Mathematical Statistics 25 (1954), 576–586.

[14] T.H. Cormen, C.E. Leiserson, R.L. Rivest, Intorduction to Algorithms, McGraw-Hill Book Company, 1990.

[15] Y. Crama, P.L. Hammer, and T. Ibaraki, Cause-effect relationships and partially defined Boolean functions, Annals of Operations Research 16 (1988), 299–325.

[16] C. Dupuis, M. Gamache, J.F. Páge. Logical analysis of data for estimating passenger show rates in the airline industry, Journal of Air Transport Management 18 (2012), 78–81.

[17] T.G. Dietterich, Approximate statistical tests for comparing supervised classification learning algorithms, Neural Computation 10 (1998), 1895–1924.

[18] J. Eckstein, P.L. Hammer, Y. Liu, M. Nediak, B. Simeone, The maximum box problem and its application to data analysis, Computational Optimization and Applications 23 (2002), 285–298.

[19] B. Efron, R. Tibshirani, Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy, Statistical Science 1 (1986), 54–75.

[20] S. Esmaeili, Development of equipment failure prognostic model based on logical analysis of data, Master of Applied Science Thesis, Dalhousie University, Halifax, Nova Scotia, July 2012.

[21] R. A. Fisher, On the interpretation of χ^2 from contingency tables, and the calculation of P. Journal of the Royal Statistical Society 85 (1922), 87–94.

[22] R.A. Fisher, Statistical Methods for Research Workers, Oliver and Boyd, 1954.

[23] C. Guoa, H.S. Ryoo, Compact MILP models for optimal and Pareto-optimal LAD patterns, Discrete Applied Mathematics 160 (2012), 2339–2348.

[24] D. Gusfield, C. Martel, D. Fernandez-Baca, Fast algorithms for bipartite network flow, SIAM J. Comput. 16(2), 1987.

[25] P.L. Hammer, The Logic of Cause-effect Relationships, Lecture at the International Conference on Multi-Attribute Decision Making via Operations Research based Expert Systems, Passau, Germany, 1986.

[26] P.L. Hammer, A. Kogan, B. Simeone, S. Szedmák, Pareto-optimal patterns in logical analysis of data, Discrete Applied Mathematics 144 (2004), 79–102.

[27] P.L. Hammer, T.O. Bonates, Logical analysis of data: From combinatorial optimization to medical applications, Annals of Operations Research 148 (2006), 203–225.

[28] T. Hastie, R. Tibshirani, J.H. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, 2001.

[29] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence 14 (1995), 1137–1143.

[30] K. Kim, H.S. Ryoo, Selecting genotyping oligo probes via logical analysis of data, Advances in Artificial Intelligence Lecture Notes in Computer Science 4509 (2007) 86–97.

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

[32] L.P. Kronek, A. Reddy, Logical analysis of survival data: prognostic survival models by detecting high degree interactions in right-censored data, Bioinformatics 24 (2008), i248–i253.

[33] W.H. Kruskal, Historical notes on the Wilcoxon unpaired two-sample test, Journal of American Statistical Association 52 (1957), 356–360.

[34] P. Lemaire, Ladoscope: An Implementation of Logical Analysis of Data Methodology: `http://www.kamick.org/lemaire/LAD`

[35] H. Liu, R. Setiono, Chi2: Feature selection and discretization of numeric attributes, Proc. 7th IEEE International Conference Tools with Artificial Intelligence, 1995, pp. 88.

[36] H. Liu, F. Hussain, C.L. Tan, M. Dash, Discretization: An enabling technique, Data Mining and Knowledge Discovery (2004), 393–423.

[37] R. Mankiewicz, The Story of Mathematics, Princeton University Press, 2000.

[38] H.B. Mann, D.R. Whitney, On a test of whether one of two random variables is stochastically larger than the other, Annals of Mathematical Statistics 18 (1947), 50–60.

[39] W.S. McCulloch, W. Pitts, A logical calculus of ideas immanent in nervous activity, Bulletin of Mathematical Biophysics 5 (1943), 115–137.

[40] A.L. Miguel, F. Margot, Optimization for simulation: LAD accelerator, Annals of Operations Research 188 (2011), 285–305.

[41] M.L. Minsky, S. Papert, Perceptrons: An Introduction to Computational Geometry, MIT Press, Cambridge, MA, 1969.

[42] M.A. Mortada, S. Yacout, A. Lakis, Diagnosis of rotor bearings using logical analysis of data, Journal of Quality in Maintenance Engineering 17 (2011), 371–397.

[43] C.S. Negrucseri, M.B. Pacsosi, B. Stanley, C. Stein, C.G. Strat, Solving maximum flow problems on real-world bipartite graphs, ACM Journal of Experimental Algorithmics 16 (2011), 14–28.

[44] J.J. O'Connor, Robertson, Edmund F., Student's t-test, MacTutor History of Mathematics Archive, University of St Andrews. http://www-history.mcs.st-andrews.ac.uk/Biographies/Gosset.html

[45] K. Pearson (1900), On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling, Philosophical Magazine Series 5, 50 (1900), 157–175.

[46] J.R. Quinlan, C4. 5: Programs for Machine Learning, Morgan Kaufmann, 1993.

[47] A.R. Reddy, Combinatorial Pattern-Based Survival Analysis with Applications in Biology and Medicine, Ph.D. Dissertation, Rutgers University, 2009.

[48] F. Rosenblatt, The Perceptron:A Theory of Statistical Separability in Cognitive Systems (Project PARA), US Dept. of Commerce, Office of Technical Services, 1958.

[49] F. Rosenblatt, A comparison of several perceptron models, in: G.T. Jacobi M.C. Yovits and G.D. Goldstein, editors, Self-Organizing Systems, Spartan Books, Washington, 1962, pp. 463–484.

[50] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning Internal Representations by Error Propagation, MIT Press Cambridge, MA, USA, 1986.

[51] H.S. Ryoo, I.Y. Jang, MILP approach to pattern egneration in logical analysis of data, Discrete Applied Mathematics 157 (2009), 749–761.

[52] B. Schölkopf, A.J. Smola, Learning with Kernels, MIT Press Cambridge, Mass, 2002.

[53] University of California at Irvine Machine Learning Repository http://www.ics.uci.edu/~mlearn/MLRepository.html

[54] F. Wilcoxon, Individual comparisons by ranking methods, Biometrics Bulletin 1 (1945), 80–83.