Available online at www.sciencedirect.com

SCIENCE DIRECT®

Discrete Applied Mathematics 144 (2004) 79–102

DISCRETE APPLIED MATHEMATICS

www.elsevier.com/locate/dam

Pareto-optimal patterns in logical analysis of data☆

Peter L. Hammer[a], Alexander Kogan[b], Bruno Simeone[c], Sándor Szedmák[a]

[a]*RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway, New Jersey 08854-8003, USA*
[b]*Accounting and Information Systems, Rutgers Business School, Rutgers University, 180 University Ave., Newark, New Jersey 07102, USA*
[c]*Dipartimento di Statistica, Probabilità e Statistiche Applicate,University of Rome I "La Sapienza",Piazzale Aldo Moro 5, 00185 - Roma, Italy*

Received 5 August 2002; received in revised form 10 March 2003; accepted 20 August 2003

Abstract

Patterns are the key building blocks in the logical analysis of data (LAD). It has been observed in empirical studies and practical applications that some patterns are more "suitable" than others for use in LAD. In this paper, we model various such suitability criteria as partial preorders defined on the set of patterns. We introduce three such preferences, and describe patterns which are Pareto-optimal with respect to any one of them, or to certain combinations of them. We develop polynomial time algorithms for recognizing Pareto-optimal patterns, as well as for transforming an arbitrary pattern to a better Pareto-optimal one with respect to any one of the considered criteria, or their combinations. We obtain analytical representations characterizing some of the sets of Pareto-optimal patterns, and investigate the computational complexity of generating all Pareto-optimal patterns. The empirical evaluation of the relative merits of various types of Pareto-optimality is carried out by comparing the classification accuracy of Pareto-optimal theories on several real life data sets. This evaluation indicates the advantages of "strong patterns", i.e. those patterns which are Pareto-optimal with respect to the "evidential preference" introduced in this paper.

Keywords: Extremal patterns, Data mining; Machine learning; Classification accuracy; Boolean functions

1. Introduction

This paper is devoted to a frequently encountered problem of data analysis, in which a set of "observations" is given, with each of the observations being represented as a vector of binary attribute values. The observations in the data set are of two types, and the type of each observation (e.g., positive or negative) is known. Typical data analysis problems related to such data sets include classification (i.e., identification of the type of a new observation not included in the data set), determination of characteristic properties of observations of the same type, analysis of the role of various attributes, etc.

The logical analysis of data (LAD) [22,13,4–8,21,19] is a methodology addressing the above kinds of problems. The mathematical foundation of LAD is in discrete mathematics, with a special emphasis on the theory of Boolean functions.

Patterns are the key building blocks in LAD [22,13], as well as in many other rule induction algorithms (such as C4.5rules [24], CN2 [10,9], AQ17-HCI [27], RISE [14], RIPPER [11] and SLIPPER [12]). Since a typical data set has an exceedingly large number of patterns, all these algorithms are limited to the consideration of small subsets of patterns. In most algorithms,

E-mail address: hammer@rutcor.rutgers.edu (P.L. Hammer), kogan@rutcor.rutgers.edu (A. Kogan), marsalis@rosd.sta.uniroma1.it (B. Simeone), szedmak@rutcor.rutgers.edu (S. Szedmák).

☆ The partial support of ONR (Grant N00014-92-J-1375) and NSF (Grant NSF-DMS-9806389) is gratefully acknowledged. The authors are grateful to the anonymous referees for their useful comments.

0166-218X/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.dam.2003.08.013

the choice of such a subset of patterns is not explicitly analyzed, in spite of the fact that it has been observed in empirical studies and practical applications that some patterns are more "suitable" than others for use in data analysis. The goal of this paper is to model various such suitability criteria as partial preorders defined on the set of patterns.

After providing some basic definitions and terminology in Section 2, we introduce in the following section the three basic preferences of "simplicity", "selectivity", and "evidence", as well as their combinations obtained by using intersections or refinements. We describe then patterns which are Pareto-optimal with respect to the introduced preferences. In Section 4, we develop polynomial time algorithms which transform an arbitrary pattern to a "better" Pareto-optimal one with respect to any one of the considered criteria. Section 5 is devoted to the investigation of the computational complexity of generating all Pareto-optimal patterns. In Section 6, we obtain analytical characterizations of various types of Pareto-optimal patterns as the solutions of certain associated Boolean equations.

The empirical evaluation of the relative merits of various types of Pareto-optimality is carried out in Section 7 by comparing the classification accuracy of Pareto-optimal theories on several real life data sets. This evaluation indicates the advantages of "strong" patterns (i.e., those patterns which are Pareto-optimal with respect to the evidential preference introduced in this paper), and ways of reducing the number of errors or unclassified observations through the use of those strong patterns which are also "spanned" or "prime", respectively (i.e., those patterns which are Pareto-optimal with respect to the refinement of the evidential preference with respect to either simplicity or selectivity).

2. Notation and terminology

A *Boolean function* $f(x_1, x_2, \ldots, x_n)$ is a mapping $\{0,1\}^n \to \{0,1\}$. While the values of a Boolean function are defined in every 0-1 n-vector, in LAD these values are usually known only in very few of the 0-1 n-vectors. A *partially defined Boolean function* (pdBf) is given by two disjoint sets of n-dimensional 0-1 vectors, and is hereafter denoted by (T, F), where $T \subseteq \{0,1\}^n$ is the set of "true" (or "positive") vectors, and $F \subseteq \{0,1\}^n$ is the set of "false" (or "negative") vectors. In line with the generally accepted terminology in many practical studies, the vectors in (T, F) will also be called *observations*. A Boolean function $f(x_1, x_2, \ldots, x_n)$ will be called an *extension* of a pdBf (T, F) iff

$$f(X) = \begin{cases} 1 & \text{if } X \in T, \\ 0 & \text{if } X \in F. \end{cases}$$

A *literal* is either a binary variable x_i or its negation $\bar{x}_i$, notation x_i^α refers to x_i if $\alpha = 1$, and to $\bar{x}_i$ if $\alpha = 0$. A *term* is a conjunction of distinct literals which does not contain both a variable and its negation. A term containing n literals will be called a *minterm*. Note that minterms are in one-to-one correspondence with Boolean vectors. We shall say that a term C *covers* an observation X iff $C(X) = 1$. The set of all Boolean vectors X, not necessarily in $T \cup F$, such that $C(X) = 1$, will be denoted by $S(C)$. Clearly, $S(C)$ is a subcube of $\{0,1\}^n$.

LAD is built around two central concepts: (positive or negative) patterns and (positive or negative) theories. Following the terminology of [22,13], a term C is called a *positive (negative) pattern* of a pdBf (T, F) if

1. $C(X) = 0$ for every $X \in F$ $(X \in T)$, and
2. $C(X) = 1$ for at least one vector $X \in T$ $(X \in F)$.

Notice that in the special case of Boolean functions, since $T \cup F = \{0,1\}^n$, condition 1 implies condition 2. In that case, a term which satisfies condition 1 is called a *positive (negative) implicant* of that Boolean function. Clearly, in the case of Boolean functions the concept of patterns reduces to that of implicants. Recall that a *prime implicant* of a Boolean function is defined as an implicant having the property that the removal of any of its literals results in a term which is not an implicant.

Example 2.1. Let us consider the pdBf given in Table 1. In this table, *a*, *b*, *c*, *d*, and *e* are positive observations, and *p*, *q*, *r*, *s*, and *t* are negative observations. It can be checked, for example, that $x_1x_2x_3$ is a positive pattern and $x_1\bar{x}_2\bar{x}_3$ is a negative pattern. It can also be seen that $x_1\bar{x}_2$ is neither a positive nor a negative pattern, since it covers both positive and negative observations. On the other hand, $x_1x_2\bar{x}_3$ is neither a positive nor a negative pattern, since it does not cover any of the given observations. □

Terms can be geometrically interpreted as subcubes of the n-dimensional cube $\{0,1\}^n$. Then positive (negative) patterns correspond to those subcubes that intersect the set T (respectively, F) but do not intersect the set F (respectively, T). Consider again Example 2.1. The term $C = \bar{x}_1\bar{x}_3x_4$ is a positive pattern. The set of those points where C takes the value 1, or equivalently, where $x_1 = 0$, $x_3 = 0$, $x_4 = 1$, is the subcube $Q = \{(01011), (00011), (01010), (00010)\}$. Notice that $Q \cap F = \emptyset$. Whenever it does not cause a confusion, we may refer to terms and corresponding subcubes interchangeably.

Table 1
A pdBf (T, F)

		x_1	x_2	x_3	x_4	x_5
	a	1	0	1	1	1
	b	0	0	0	1	1
T	c	1	1	1	1	1
	d	1	1	1	0	1
	e	1	1	1	0	0
	p	1	0	0	1	0
	q	0	0	1	0	1
F	r	1	0	1	0	0
	s	1	0	0	0	0
	t	0	0	1	0	0

Since the properties of positive and negative patterns are completely symmetric, without loss of generality we will focus in this paper on positive patterns. We will frequently refer to positive patterns simply as patterns.

In LAD, a (positive) theory (or simply theory) $\mathscr{T}$ of a pdBf (T, F) is a collection of patterns $P_1, \ldots, P_k$ with the property that for every $X \in T$ there exists a $P_i \in \mathscr{T}$ such that $P_i(X) = 1$. A theory $\mathscr{T}$ is associated with the Boolean function represented by the disjunction of terms (DNF) $\bigvee_{i=1}^{k} P_i$. This DNF can be used to predict whether a Boolean vector not in $T \cup F$ is true or false. Obviously, every theory is an extension of (T, F). Note that although every Boolean function can be represented by a DNF, not every extension is a theory in the sense that there may exist an extension such that any DNF representing it includes terms which are not patterns (i.e., do not cover any vectors in T).

In Example 2.1, $\mathscr{T} = \{\bar{x}_1\bar{x}_3x_4, x_2\bar{x}_4, x_4x_5\}$ is a theory, since its patterns cover all the vectors in T: $\bar{x}_1\bar{x}_3x_4$ covers the point b, $x_2\bar{x}_4$ covers d and e, while x_4x_5 covers a, b, and c. This theory defines the extension $\bar{x}_1\bar{x}_3x_4 \vee x_2\bar{x}_4 \vee x_4x_5$ of the pdBf given in Table 1.

3. Preferences and Pareto-optimal patterns

In order to model various pattern suitability criteria as partial preorders, we shall need some definitions. First, we recall that a binary relation ρ defined on a finite set S is called a *partial preorder* if it is

(i) *reflexive*, i.e., $x\rho x$ holds for any $x \in S$, and
(ii) *transitive*, i.e., for any $x, y, z \in S$, if both $x\rho y$ and $y\rho z$ hold, then $x\rho z$ also holds.

A partial preorder is called a *partial order* if it is also

(iii) *antisymmetric*, i.e., for any $x, y \in S$, if $x\rho y$ holds, then $y\rho x$ cannot hold.

A partial preorder ρ' is called a *refinement* of a partial preorder ρ if for any $x, y \in S$ the relation $x\rho y$ implies the relation $x\rho' y$.

Definition 3.1. Given a partial preorder $\succcurlyeq$ on the set of patterns, a pattern P will be called *Pareto-optimal* with respect to $\succcurlyeq$, if there is no distinct pattern P' such that $P' \succcurlyeq P$.

Unfortunately, the concept of suitability does not have a unique definition. Among the many reasonable criteria of suitability, we shall briefly discuss below the three (simplicity, selectivity, evidence) that look most important to us.

Since the earliest studies on LAD [22,13], it was realized that *simplicity* was an important preference criterion in comparing patterns. The criterion of simplicity, also known as the *Principle of Occam's Razor*, is a widely accepted tenet in the sciences. To formally define a simplicity-based partial preorder of patterns, let us denote by $\mathrm{Lit}(C)$ the set of literals in a term C.

Definition 3.2 (*Simplicity preference*). A pattern P_1 is *simplicity-wise preferred* to a pattern P_2 (denoted by $P_1 \succcurlyeq_\sigma P_2$) if and only if $\mathrm{Lit}(P_1) \subseteq \mathrm{Lit}(P_2)$.

In [22,13], a pattern P was called *prime* if the removal of any literal appearing in Lit(P) results in a term C which is not a pattern. It is clear that a pattern is Pareto-optimal with respect to simplicity if and only if it is prime.

Example 3.3. It can be seen that the pdBf given in Table 1 has among its prime patterns the following terms:

$$x_2, \bar{x}_1\bar{x}_3, \bar{x}_1x_4, x_1x_5, x_3x_4, \bar{x}_3x_5, x_4x_5.$$

On the other hand, the pattern x_1x_2 provides an example of a non-prime pattern.

It is easy to make an argument in favor of the simplicity preference. The simplest argument for simplicity simply states that this criterion corresponds to the way in which human cognition works. It is natural to expect that the fewer variables a pattern involves, the easier it is to comprehend its meaning. While this is a popular point of view, it is not universally accepted (for a discussion, see [15,16]). Another argument used by some authors in favor of simplicity states that simplicity leads to higher accuracy (see e.g., the computational learning theory model of Occam's razor, which is proposed in [3]). This point of view is again not universally accepted. Moreover, various theoretical and empirical arguments were made, stating that simplicity in itself may even lead to lower accuracy (see e.g., [15,16,25]). In particular, it was shown in [23] that a decrease in the simplicity of patterns can result in higher accuracy.

In the special case of LAD, the simplicity preference favors short patterns. While an unknown point covered by a short pattern is, of course, not necessarily a positive one, a point which is *not* covered by any of the short patterns is quite likely a negative one. Therefore, in our view, the use of the simplicity preference in LAD tends to reduce the number of "false negatives", but does not offer *in itself* safeguards against "false positives".

A natural way of reducing the number of false positives is to favor more selective patterns. This can be achieved by reducing the size of the subcube $S(P)$ of a pattern P.

Definition 3.4 (*Selectivity preference*). A pattern P_1 is *selectivity-wise preferred* to a pattern P_2 (denoted by $P_1 \succcurlyeq_\Sigma P_2$) if and only if $S(P_1) \subseteq S(P_2)$.

Two immediate remarks are in order. First, it is obvious that $P_1 \succcurlyeq_\Sigma P_2$ if and only if $P_2 \succcurlyeq_\sigma P_1$. Second, it is equally obvious that the patterns which are Pareto-optimal with respect to selectivity are exactly the minterms corresponding to the positive observations. Therefore, the exclusive use of patterns which are Pareto-optimal with respect to selectivity would allow LAD to classify as positive exclusively the originally given set of positive points. Clearly, a theory based on such patterns would achieve the goal of avoiding false positives at the expense of declaring every unknown point negative. It will be seen below that in spite of this seemingly unproductive role of the selectivity preference, it can become extremely useful in combination with other criteria.

While previous theoretical LAD studies employed exclusively the simplicity preference, the implementation of LAD [6] took into account another very natural suitability criterion. This criterion is related to the so-called *coverage* Cov(P) of a pattern P, i.e. the set of vectors $X \in T$ for which $P(X) = 1$. Note that the second condition in the definition of a pattern guarantees that for every pattern P the coverage $\mathrm{Cov}(P) \neq \emptyset$. The effect of $|\mathrm{Cov}(P)|$ on the accuracy of rule induction algorithms was investigated within the framework of the so-called "problem of small disjuncts" (e.g., [23,26]).

While the relation $|\mathrm{Cov}(P_1)| > |\mathrm{Cov}(P_2)|$ could be interpreted as signifying that the pattern P_1 is more representative than the pattern P_2, in fact, it only takes into account the *number* of elements in the two sets $\mathrm{Cov}(P_1)$ and $\mathrm{Cov}(P_2)$. However, replacing the above cardinality relation between these two sets with the stronger set inclusion relation allows to take into account the individual observations covered by the two patterns. The observations in Cov(P) can be viewed as the "body of evidence" supporting the pattern P. This point of view leads to the following definition.

Definition 3.5 (*Evidential preference*). A pattern P_1 is *evidentially preferred* to a pattern P_2 (denoted by $P_1 \succcurlyeq_\varepsilon P_2$) if and only if $\mathrm{Cov}(P_1) \supseteq \mathrm{Cov}(P_2)$.

Evidentially Pareto-optimal patterns will be called *strong*. Clearly, a pattern P is strong if and only if there is no pattern P' such that $Cov(P') \supset Cov(P)$.

Example 3.6. It can be seen that for the pdBf given in Table 1, the pattern x_3x_4 is not strong; indeed, $Cov(x_4x_5) = \{a, b, c\} \supset \{a, c\} = Cov(x_3x_4)$. It will be seen later that the following terms are among the strong patterns of this pdBf:

$$x_2, x_1x_2, x_2x_3, x_1x_2x_3, x_1x_5, x_1x_3x_5, x_4x_5.$$

For any preference $\succcurlyeq$, the simultaneous satisfaction of the relations $P_1 \succcurlyeq P_2$ and $P_2 \succcurlyeq P_1$ will be denoted by $P_1 \approx P_2$. Also, the simultaneous satisfaction of the relations $P_1 \succcurlyeq P_2$ and $P_2 \not\succcurlyeq P_1$ will be denoted by $P_1 \succ P_2$.

It is easy to notice that the three preferences defined above are not independent of each other. First of all, as has been remarked, the simplicity and the selectivity preferences are exactly the opposite of each other. More interestingly, the following two implications can be easily seen to hold:

$$P_1 \succcurlyeq_\sigma P_2 \Longrightarrow P_1 \succcurlyeq_\varepsilon P_2, \tag{1}$$

$$P_1 \succcurlyeq_\Sigma P_2 \Longrightarrow P_2 \succcurlyeq_\varepsilon P_1. \tag{2}$$

In order to define the most suitable types of patterns, we shall consider below combinations of the preferences introduced above. The two most natural ways of combining a preference π with a preference ρ is to consider their intersection $\pi \wedge \rho$, or their lexicographic refinement $\pi \mid \rho$, as defined below.

Definition 3.7. Given preferences π and ρ on the set of patterns, a pattern P_1 is preferred to a pattern P_2 with respect to the intersection $\pi \wedge \rho$ (denoted by $P_1 \succcurlyeq_{\pi\wedge\rho} P_2$) if and only if $P_1 \succcurlyeq_\pi P_2$ and $P_1 \succcurlyeq_\rho P_2$.

Definition 3.8. Given preferences π and ρ on the set of patterns, a pattern P_1 is preferred to a pattern P_2 with respect to the lexicographic refinement $\pi \mid \rho$ (denoted by $P_1 \succcurlyeq_{\pi\mid\rho} P_2$) if and only if $P_1 \succ_\pi P_2$ or ($P_1 \approx_\pi P_2$ and $P_1 \succcurlyeq_\rho P_2$).

It is easy to notice that while the preferences $\pi \wedge \rho$ and $\rho \wedge \pi$ are identical, the preferences $\pi \mid \rho$ and $\rho \mid \pi$ are usually different.

Since each of the introduced preferences expresses a different aspect of the suitability of patterns, it seems reasonable to analyze various combinations of them. In spite of the apparent abundance of combinations of the discussed preferences that can be formally specified by using intersections and lexicographic refinements, it will be seen below that in fact only three of these combinations are meaningful.

First of all, because of their opposite nature, any combination of the simplicity and selectivity preferences using either intersection or lexicographic refinement makes no sense.

Second, since the evidential preference ε is a refinement of the simplicity preference σ (as stated by the condition (1)), the intersection $\sigma \wedge \varepsilon$ is identical to σ.

Third, for any preference π which is a partial order, and for any preference ρ, the lexicographic refinement $\pi \mid \rho$ is identical to π. Note that each of the relations $P_1 \approx_\sigma P_2$ and $P_1 \approx_\Sigma P_2$ implies that $P_1 = P_2$, meaning that the preorders of simplicity and selectivity are actually partial orders. Therefore, the lexicographic refinements $\sigma \mid \varepsilon$ and $\Sigma \mid \varepsilon$ coincide with σ and Σ, respectively. On the other hand, the evidential preference is not a partial order (since there can exist *distinct* patterns P_1 and P_2 such that $P_1 \approx_\varepsilon P_2$), and therefore the lexicographic refinements $\varepsilon \mid \sigma$ and $\varepsilon \mid \Sigma$ introduce new preferences.

In conclusion, the only new preferences that can be obtained by applying intersection and lexicographic refinement are $\Sigma \wedge \varepsilon$, $\varepsilon \mid \sigma$, and $\varepsilon \mid \Sigma$. We shall briefly examine these three preferences below. As far as terminology is concerned, we shall call the patterns which are Pareto-optimal with respect to $\Sigma \wedge \varepsilon$ *spanned*. It should be remarked that, as can be deduced from Theorem 4.5 below, the concept of spanned patterns is essentially equivalent to that of "maximally specific disjuncts" introduced in [23] (the only difference being the absence of any restrictions on the coverage of spanned patterns). It will be seen below that the patterns which are Pareto-optimal with respect to $\varepsilon \mid \sigma$ or $\varepsilon \mid \Sigma$ do not require new names, since they can be described using already introduced terms.

Example 3.9. It can be seen that for the pdBf given in Table 1, the pattern x_3x_4 is not spanned; indeed, $\mathrm{Cov}(x_1x_3x_4x_5) = \{a, c\} = \mathrm{Cov}(x_3x_4)$, and clearly, $\mathrm{Lit}(x_1x_3x_4x_5) \supset \mathrm{Lit}(x_3x_4)$. It will be seen later that $x_1x_3x_4x_5$ is spanned. Some of the other spanned patterns of this pdBf are

$$x_1x_2x_3,\ x_1x_3x_5,\ x_4x_5,\ x_1x_2x_3\bar{x}_4,\ \bar{x}_1\bar{x}_2\bar{x}_3x_4x_5.$$

Note that for any two preferences π and ρ, the Pareto-optimal patterns with respect to the lexicographic refinement $\pi \mid \rho$ are also Pareto-optimal with respect to π. Therefore, the Pareto-optimal patterns with respect to $\varepsilon \mid \sigma$ are strong. Similarly, the Pareto-optimal patterns with respect to $\varepsilon \mid \Sigma$ are also strong.

Theorem 3.10. *A pattern is Pareto-optimal with respect to $\varepsilon \mid \sigma$ if and only if it is both strong and prime.*

Proof. Let P be a Pareto-optimal pattern with respect to $\varepsilon \mid \sigma$. Then P is strong by the above remark. If P is not prime, then there must exist a pattern P' such that $\mathrm{Lit}(P') \subset \mathrm{Lit}(P)$. It follows then that $\mathrm{Cov}(P') \supseteq \mathrm{Cov}(P)$, and therefore $P' \succ_{\varepsilon\mid\sigma} P$, contradicting the Pareto-optimality of P.

Table 2
Types of Pareto-optimality

Preference	Pareto-optimal pattern
σ	Prime
Σ	Minterm
ε	Strong
$\Sigma \wedge \varepsilon$	Spanned
$\varepsilon \mid \Sigma$	Strong spanned
$\varepsilon \mid \sigma$	Strong prime

Table 3
Patterns of the pdBf given in Table 1

Pattern properties			Pattern examples
Prime	Strong	Spanned	
No	No	No	$\bar{x}_1\bar{x}_2\bar{x}_3$, $x_2x_3\bar{x}_4$, $x_3x_4x_5$
Yes	No	No	$\bar{x}_1\bar{x}_3$, $\bar{x}_1x_4$, x_3x_4, $\bar{x}_3x_5$
No	Yes	No	x_1x_2, x_2x_3
No	No	Yes	$x_1x_2x_3\bar{x}_4$, $\bar{x}_1\bar{x}_2\bar{x}_3x_4x_5$
Yes	Yes	No	x_2, x_1x_5
No	Yes	Yes	$x_1x_2x_3$, $x_1x_3x_5$
Yes	Yes	Yes	x_4x_5

Conversely, let P be a strong and prime pattern. If it is not Pareto-optimal with respect to $\varepsilon \mid \sigma$, then there must exist a pattern P' such that $P' \succ_{\varepsilon \mid \sigma} P$. Then either $P' \succ_{\varepsilon} P$ (contradicting the fact that P is strong), or $P' \approx_{\varepsilon} P$ and $P' \succ_{\sigma} P$ (contradicting the fact that P is prime). □

Example 3.11. It follows from Examples 3.3 and 3.6 that for the pdBf given in Table 1, the following patterns are both strong and prime:

$$x_2,\ x_1x_5,\ x_4x_5.$$

Theorem 3.12. *A pattern is Pareto-optimal with respect to $\varepsilon \mid \Sigma$ if and only if it is both strong and spanned.*

Proof. Let P be a Pareto-optimal pattern with respect to $\varepsilon \mid \Sigma$. Then P is strong. If P is not spanned, then there must exist a pattern P' such that $P' \succ_{\Sigma \wedge \varepsilon} P$. Therefore, $P' \succcurlyeq_{\Sigma} P$ and $P' \succcurlyeq_{\varepsilon} P$. Since P is strong, it follows that $P' \approx_{\varepsilon} P$ and $P' \succ_{\Sigma} P$. This simply states that $P' \succ_{\varepsilon \mid \Sigma} P$, contradicting the Pareto-optimality of P.

Conversely, let P be a strong and spanned pattern. If it is not Pareto-optimal with respect to $\varepsilon \mid \Sigma$, then there must exist a pattern P' such that $P' \succ_{\varepsilon \mid \Sigma} P$. Since P is strong, $P' \approx_{\varepsilon} P$, and therefore $P' \succ_{\Sigma} P$, contradicting the fact that P is spanned. □

Example 3.13. It follows from Examples 3.6 and 3.9 that for the pdBf given in Table 1, the following patterns are both strong and spanned:

$$x_1x_2x_3,\ x_1x_3x_5,\ x_4x_5.$$

Moreover, note that the pattern x_4x_5 is not only strong and spanned, but also prime.

In conclusion, we summarize in Table 2, the properties of patterns which are Pareto-optimal with respect to the preferences and combinations of preferences discussed above.

This paper will focus on the study of spanned, strong spanned, and strong prime patterns of pdBfs. In Table 3 we summarize the examples presented above in order to highlight the existence of patterns having various combinations of properties discussed in this section. The only combination of pattern properties, which is missing from Table 3, is "Prime, Not Strong, Spanned". It will be seen later (Theorem 4.10) that this combination is not possible, since a pattern, which is both prime and spanned, must also be strong.

It can be seen that in the special case of Boolean functions (i.e., $T \cup F = \{0,1\}^n$), only two of the above combinations of pattern properties are possible: "Not Prime, Not Strong, Spanned" and "Prime, Strong, Spanned". This observation follows from the fact that in this case every pattern is spanned, and a pattern is prime if and only if it is strong. Additionally, in this case both the concepts of prime pattern and of strong pattern are simply reduced to that of prime implicant.

4. Pareto-optimization of patterns and theories

After having introduced several important types of preferences and the corresponding Pareto-optimal patterns in the previous section, we shall now focus our attention on developing efficient computational procedures for transforming an arbitrary pattern into a comparable Pareto-optimal one.

For each preference listed in Table 2 (with the only exception of the trivial case of selectivity, Σ), simple, polynomial time transformations will be developed in this section to associate to an arbitrary pattern a "better" Pareto-optimal one. Obviously, every Pareto-optimal pattern resulting from the transformations presented below will cover a superset of the observations covered by the original pattern.

4.1. Prime patterns

Given a term C and a pdBf (T, F), it is extremely easy to check whether C is a pattern of (T, F) by simply verifying on the one hand whether there exists an observation in T covered by C, and on the other hand whether no observation in F is covered by C. Similarly, it is also very easy to check whether a pattern P of a pdBf (T, F) is prime. Indeed, all one has to do is to examine one by one the terms obtained from P by eliminating one of its literals, and to verify that none of these "shortened" terms is a pattern, i.e. that no such term covers an observation in F.

Example 4.1. For the pdBf in Table 1, the term $x_1x_3x_5$ is a pattern which is not prime, because the elimination of the literal x_3 results in the term x_1x_5 which is a pattern. On the other hand, the pattern x_1x_5 is prime because neither the "shortened" term x_1 nor the other "shortened" term x_5 is a pattern.

The procedure described above, which recognizes the primality of a pattern P of a pdBf (T, F), leads naturally to the following procedure, which transforms an arbitrary pattern P to a prime pattern $P' \succcurlyeq_\sigma P$, implying the following *coverage condition*:

$$\mathrm{Cov}(P') \supseteq \mathrm{Cov}(P). \tag{3}$$

Algorithm 1. Transformation of patterns to prime patterns

Step 1: Given a pattern $P = \prod_{k=1}^{t} x_{j_k}^{\alpha_k}$, where $j_1 \leqslant \cdots \leqslant j_t$, let $P' := P$ and $i := 0$.

Step 2: If $i = t$, then go to Step 5, otherwise let $i := i + 1$ and let $C := P' \backslash \{x_{j_i}^{\alpha_i}\}$.

Step 3: If C is a pattern, then let $P' := C$.

Step 4: Go to Step 2.

Step 5: Output P'.

It is easy to check that the running time of this algorithm is $O(n \cdot |\mathrm{Lit}(P)| \cdot |F|)$, where n is the number of Boolean variables and F is the set of false points.

Clearly, the above procedure is correct, i.e. starting from an arbitrary pattern P, it will yield a prime pattern P' satisfying

$$\mathrm{Lit}(P') \subseteq \mathrm{Lit}(P), \tag{4}$$

i.e., such that $P' \succcurlyeq_\sigma P$.

Note that variants of Algorithm 1 can be developed so as to attemp to enlarge $|\mathrm{Cov}(P')|$ or to decrease $|\mathrm{Lit}(P')|$.

4.2. Spanned patterns

We recall that spanned patterns were defined as those patterns which are Pareto-optimal with respect to $\Sigma \wedge \varepsilon$. We shall need below the following interesting property of the preference $\Sigma \wedge \varepsilon$.

Lemma 4.2. *If P_1 and P_2 are patterns of a pdBf (T, F) which satisfy the relation $P_1 \succcurlyeq_{\Sigma \wedge \varepsilon} P_2$, then* $\mathrm{Cov}(P_1) = \mathrm{Cov}(P_2)$.

Proof. The relation $P_1 \succcurlyeq_{\Sigma \wedge \varepsilon} P_2$ implies on the one hand that

$$\mathrm{Cov}(P_1) \supseteq \mathrm{Cov}(P_2), \tag{5}$$

and on the other hand that

$$\mathrm{Lit}(P_1) \supseteq \mathrm{Lit}(P_2). \tag{6}$$

It follows from (6) that

$$\mathrm{Cov}(P_1) \subseteq \mathrm{Cov}(P_2). \tag{7}$$

The statement of the lemma immediately follows from (5) and (7). □

In order to characterize spanned patterns, let us define the *convex hull* $[S]$ of a non-empty subset $S \subseteq \{0,1\}^n$ as the smallest subcube containing S. This concept is well-defined, since the intersection of any two subcubes containing S is a subcube containing S.

Lemma 4.3. *Consider a non-empty subset $S \subseteq \{0,1\}^n$ and let I be the set of all those indices i for which the ith components of all the vectors $X \in S$ have a common value, say $\alpha_i \in \{0,1\}$. Then*

$$[S] = \prod_{i \in I} x_i^{\alpha_i}.$$

Proof. Since for every $X = (X_1, \ldots, X_n) \in S$ we have $X_i = \alpha_i$, i.e., $X_i^{\alpha_i} = 1$, it follows that every $X \in S$ is covered by $\prod_{i \in I} x_i^{\alpha_i}$.

Let us assume that there exists a term C such that every $X \in S$ is covered by C, but $\bigcup_{i \in I} x_i^{\alpha_i} \not\supseteq \mathrm{Lit}(C)$. This implies that C contains a literal $x_t^{\alpha_t} \notin \bigcup_{i \in I} x_i^{\alpha_i}$. Since every $X \in S$ is covered by C, it follows that $X_t = \alpha_t$ for every $X \in S$. It follows that $t \in I$, contradicting the conclusion above. □

Lemma 4.4. *If P is a pattern of a pdBf (T, F), then*

$$[\mathrm{Cov}(P)] \succcurlyeq_{\Sigma \wedge \varepsilon} P, \tag{8}$$

and therefore

$$\mathrm{Cov}([\mathrm{Cov}(P)]) = \mathrm{Cov}(P). \tag{9}$$

Proof. It follows from the definition of the convex hull that

$$\mathrm{Cov}(P) \subseteq \mathrm{Cov}([\mathrm{Cov}(P)]). \tag{10}$$

On the other hand, it follows from Lemma 4.3 that

$$\mathrm{Lit}(P) \subseteq \mathrm{Lit}([\mathrm{Cov}(P)]), \tag{11}$$

because for every $x_i^{\alpha_i} \in \mathrm{Lit}(P)$ and every $X \in \mathrm{Cov}(P)$ we have $X_i = \alpha_i$.

Inclusions (10) and (11) imply (8), which in its turn implies (9) by Lemma 4.2. □

Theorem 4.5. *A pattern P of a pdBf (T, F) is spanned if and only if*

$$P = [\mathrm{Cov}(P)], \tag{12}$$

or, equivalently, P is a fixed point of the mapping

$$C \Longrightarrow [\mathrm{Cov}(C)].$$

Proof. Let us first prove the "only if" part of the statement. Lemma 4.4 states that (8) holds. Since P is assumed to be spanned, i.e., Pareto-optimal with respect to $\Sigma \wedge \varepsilon$, it must hence follow that (12) holds.

Let us now prove the "if" part of the statement. Let a pattern P be such that (12) holds, and let us assume by contradiction that P is not spanned. Then there must exist a distinct pattern P' (i.e., $\mathrm{Lit}(P') \neq \mathrm{Lit}(P)$) such that $\mathrm{Cov}(P') \supseteq \mathrm{Cov}(P)$ and $\mathrm{Lit}(P') \supset \mathrm{Lit}(P)$, in contradiction with the assumption that P is the smallest subcube containing $\mathrm{Cov}(P)$. □

Corollary 4.6. *If P is a pattern of a pdBf (T, F), then*

$$\widehat{P} = [\mathrm{Cov}(P)] \tag{13}$$

is a spanned pattern of (T, F) which has the same coverage as P.

Proof. In view of Lemma 4.4 and Theorem 4.5, the statement follows from the following results:

$$[\mathrm{Cov}(\widehat{P})] = [\mathrm{Cov}([\mathrm{Cov}(P)])] = [\mathrm{Cov}(P)] = \widehat{P}. \quad \square$$

Corollary 4.6 can be restated as the following computationally inexpensive procedure, which transforms an arbitrary pattern P to the spanned pattern $\widehat{P} \succcurlyeq_{\Sigma \wedge \varepsilon} P$ (which, by Lemma 4.2, will satisfy the coverage condition (3)).

Algorithm 2. Transformation of patterns to spanned patterns

Step 1: Given a pattern P, let $S := \mathrm{Cov}(P)$.

Step 2: Output $\widehat{P} := [S]$.

It is easy to check that the running time of this algorithm is $O(n \cdot |T|)$, where n is the number of Boolean variables and T is the set of true points.

If P and P' are two arbitrary patterns of a pdBf (T, F), then the condition

$$\mathrm{Lit}(P) \supseteq \mathrm{Lit}(P')$$

obviously implies the condition

$$\mathrm{Cov}(P) \subseteq \mathrm{Cov}(P').$$

The converse implication does not hold in general. However, as stated below, it does hold in the special case when P is spanned.

Corollary 4.7. *If P is a spanned pattern of a pdBf (T, F) and P' is another pattern of (T, F) such that* $\mathrm{Cov}(P') \supseteq \mathrm{Cov}(P)$, *then* $\mathrm{Lit}(P') \subset \mathrm{Lit}(P)$.

4.3. Strong patterns

By definition, a pattern P is strong if and only if there is no pattern P' such that

$$\mathrm{Cov}(P') \supset \mathrm{Cov}(P).$$

A direct combinatorial characterization of strong patterns is provided by the following theorem.

Theorem 4.8. *A pattern P of a pdBf (T, F) is strong if and only if the term* $[\mathrm{Cov}(P) \cup X]$ *is not a pattern for any point* $X \in T \backslash \mathrm{Cov}(P)$.

Proof. Note that the "only if" part of the statement is obvious, since if there exists a point $X \in T \backslash \mathrm{Cov}(P)$ such that the term $[\mathrm{Cov}(P) \cup X]$ is a pattern, then P is not strong, because $\mathrm{Cov}(P) \subset \mathrm{Cov}([\mathrm{Cov}(P) \cup X])$.

Let us now prove the "if" part of the statement by contradiction. Let us assume that P is not strong, and that the term $[\mathrm{Cov}(P) \cup X]$ is not a pattern for any point $X \in T \backslash \mathrm{Cov}(P)$. Then there must exist a pattern P' such that

$$\mathrm{Cov}(P') \supset \mathrm{Cov}(P). \tag{14}$$

It follows from (14) that there exists a point $X' \in \mathrm{Cov}(P') \backslash \mathrm{Cov}(P)$, or equivalently,

$$\mathrm{Cov}(P) \cup X' \subseteq \mathrm{Cov}(P').$$

This implies

$$\mathrm{Lit}([\mathrm{Cov}(P) \cup X']) \supseteq \mathrm{Lit}([\mathrm{Cov}(P')]) \supseteq \mathrm{Lit}(P').$$

It then follows that $[\mathrm{Cov}(P) \cup X']$ is a pattern, contradicting the assumption above. $\square$

Theorem 4.8 leads naturally to the following procedure, which transforms an arbitrary pattern P to a strong pattern $P' \succcurlyeq_{\varepsilon} P$.

Algorithm 3. Transformation of patterns to strong patterns

Step 1: Given a pattern P of a pdBf (T, F), let $P' := [\mathrm{Cov}(P)]$ and $K := T \backslash \mathrm{Cov}(P)$.
Step 2: If $K = \emptyset$, then go to Step 6, otherwise let $X \in K$.
Step 3: Let $C := [\mathrm{Cov}(P') \cup X]$, and let $K := K \backslash X$.
Step 4: If C is a pattern, then let $P' := C$ and $K := K \backslash \mathrm{Cov}(P')$.
Step 5: Go to Step 2.
Step 6: Output P'.
It is easy to check that the running time of this algorithm is $O(n \cdot |T| \cdot |F|)$, where n is the number of Boolean variables.

Note that variants of this transformation procedure can be developed so as to attempt to enlarge $|\mathrm{Cov}(P')|$ or to decrease $|\mathrm{Lit}(P')|$.

Remark 4.9. The output P' of Algorithm 3 is not only strong, but is also spanned, i.e., $P' \succcurlyeq_{\varepsilon \mid \Sigma} P$. This pattern P' may or may not be prime. On the other hand, by applying to this pattern Algorithm 1 (described in Subsection 4.1), we obtain a pattern P'' which is not only prime but also strong, i.e., $P'' \succcurlyeq_{\varepsilon \mid \sigma} P$.

The above remark speaks about those patterns which are simultaneously strong and spanned, or simultaneously strong and prime. The only two combinations of these pattern properties which have not yet been discussed are prime-spanned and strong-prime-spanned. As a matter of fact, these two classes coincide:

Theorem 4.10. *If a pattern P of a pdBf (T, F) is both prime and spanned, then it is also strong.*

Proof. Let us prove this theorem by contradiction, and assume that P is not strong. Then there must exist a pattern P' such that $\mathrm{Cov}(P') \supset \mathrm{Cov}(P)$. Since P is spanned, Corollary 4.7 implies that $\mathrm{Lit}(P') \subset \mathrm{Lit}(P)$. This implies that P is not prime, contradicting the assumption. □

4.4. Pareto-optimization of theories

Let us recall that a theory $\mathscr{T}$ of a pdBf (T, F) is a collection of patterns $P_1, \ldots, P_k$ with the property that for every $X \in T$ there exists a $P_i \in \mathscr{T}$ such that $X \in \mathrm{Cov}(P_i)$. All the transformations described in this section result in patterns which satisfy the coverage condition (3). It follows that applying any of these transformations to all the patterns of a theory will result in a collection of patterns which again form a theory. This fact makes it possible to produce theories consisting exclusively of the type of Pareto-optimal patterns we are interested in, e.g., strong and prime, or strong and spanned. We shall call such theories strong prime theories and strong spanned theories respectively.

In order to contsruct a Pareto-optimal theory by the above approach, we have to start with some initial theory. Perhaps the most straightforward way of producing an initial theory is to form the "minterm theory", i.e., the collection of all the minterms associated to the individual points in T. While this seems to be the simplest way to produce a Pareto-optimal theory, it is far from sure that it will produce a "best" one. In spite of that, such Pareto-optimal theories can be useful for empirically comparing the relative merits of theories based on various types of Pareto-optimal patterns. In Section 7, we shall present the results of several such comparisons using some real life data sets.

5. Complexity of generation of strong spanned patterns

In the previous section, we have presented efficient polynomial algorithms forr transforming a pattern into a better Pareto-optimal one with respect to various preferences. In this section, we shall analyze the computational complexity of generating *all* the Pareto-optimal patterns.

Among the types of Pareto-optimal patterns considered in this paper, the strong spanned patterns possess the most appealing combination of properties. An additional advantage of the strong spanned patterns is that their number does not exceed that of any other type of Pareto-optimal patterns studied here, with the exception of the (not very interesting) case of minterms. (This statement follows from the easily observed fact that for every strong spanned pattern there exists at least one strong prime pattern which covers exactly the same set of true points.)

In order to analyze the complexity of generating all strong spanned patterns, let us introduce Boolean variables $z_1, \ldots, z_{|T|}$ associated to the true points $X_1, \ldots, X_{|T|}$ of the pdBf (T, F); these variables are defined by:

$$z_k = \begin{cases} 1 & \text{if } X_k \in \mathrm{Cov}(P), \\ 0 & \text{otherwise.} \end{cases}$$

In order to easily distinguish between positive and negative observations, we shall denote the points of T by $\beta_k = (\beta_{k1}, \ldots, \beta_{kn})$, for $k = 1, \ldots, |T|$, and the points of F by $\gamma_l = (\gamma_{l1}, \ldots, \gamma_{ln})$, for $l = 1, \ldots, |F|$.

Lemma 5.1. *A Boolean vector $(Z_1, \ldots, Z_{|T|})$ is the characteristic vector of a subset of the coverage set of a pattern if and only if for every false point $(\gamma_{l1}, \ldots, \gamma_{ln})$, $l = 1, \ldots, |F|$, there exists a coordinate $j \in \{1, \ldots, n\}$ such that the implication "if $Z_k = 1$ then $\beta_{kj} \neq \gamma_{lj}$" holds for every true point $(\beta_{k1}, \ldots, \beta_{kn})$, $k = 1, \ldots, |T|$.*

Proof. Let us first prove the "only if" part. Let $(Z_1, \ldots, Z_{|T|})$ be the characteristic vector of the coverage set of a pattern P, and $(\gamma_{l1}, \ldots, \gamma_{ln})$ be a false point. Since P is a pattern, $P(\gamma_{l1}, \ldots, \gamma_{ln}) = 0$, and hence there must exist a coordinate $j \in \{1, \ldots, n\}$ such that $P = x_j^{\overline{\gamma}_{lj}} P'$. This shows that for every $(\beta_{k1}, \ldots, \beta_{kn}) \in \mathrm{Cov}(P)$ we have $\beta_{kj} = \overline{\gamma}_{lj}$.

Let us now prove the "if" part. Let

$$S = \{\beta_k \in T \mid Z_k = 1\},$$

and let $P = [S]$. We have to prove that P is a pattern. Let γ_l be a false point. Then, by our assumption, there exists a coordinate $j \in \{1, \ldots, n\}$ such that for every $\beta_k \in S$ we have $\beta_{kj} = \overline{\gamma}_{lj}$. Therefore, $P = x_j^{\overline{\gamma}_{lj}} P'$, and hence $P(\gamma_l) = 0$. □

Let us associate to a pdBf (T, F) the Boolean function

$$\tau(z_1, \ldots, z_{|T|}) = \bigvee_{l=1}^{|F|} \prod_{j=1}^{n} \bigvee_{k=1}^{|T|} z_k (\beta_{kj} \vee \overline{\gamma}_{lj})(\overline{\beta}_{kj} \vee \gamma_{lj}), \tag{15}$$

which, in view of the corollary below, will be called the *coverage function* of (T, F).

Corollary 5.2. *A Boolean vector $(Z_1, \ldots, Z_{|T|})$ is the characteristic vector of a subset of the coverage set of a pattern of a pdBf (T, F) if and only if it is a solution of*

$$\tau(z_1, \ldots, z_{|T|}) = 0.$$

Note that the coverage function τ is monotone non-decreasing.

Corollary 5.3. *The strong spanned patterns of a pdBf (T, F) are in one-to-one correspondence with the maximal false points of its coverage function τ.*

Lemma 5.4. *The coverage function τ of a pdBf (T, F) does not have any linear implicants.*

Proof. Let us assume by contradiction that τ has a linear implicant. In this case, as can be seen from (15), there exists a false point γ_l and a true point β_k such that for every $j \in \{1, \ldots, n\}$ the following condition holds:

$$(\beta_{kj} \vee \overline{\gamma}_{lj})(\overline{\beta}_{kj} \vee \gamma_{lj}) = 1.$$

However, this condition simply means that $\beta_{kj} = \gamma_{lj}$ for every $j \in \{1, \ldots, n\}$, i.e., $\beta_k = \gamma_l$, contradicting the disjointness of T and F. □

Interestingly, the necessary condition of Lemma 5.4 actually characterizes the set of coverage functions.

Theorem 5.5. *Every monotone non-decreasing Boolean function without linear implicants is the coverage function of a pdBf.*

Proof. Let us consider an arbitrary monotone non-decreasing Boolean function without linear implicants:

$$g(z_1,\ldots,z_n)=\bigvee_{l=1}^{m}\prod_{j\in S_l} z_{lj},$$

where $|S_l|\geqslant 2$, for $l=1,\ldots,m$. Let us construct now the pdBF (T,F) with $|T|=n$ and $|F|=m$ such that:

$$\beta_{kj}=\begin{cases}1 & \text{if } k=j,\\ 0 & \text{if } k\neq j,\end{cases}$$

and

$$\gamma_{lj}=\begin{cases}1 & \text{if } j\in S_l,\\ 0 & \text{if } j\notin S_l.\end{cases}$$

It is easy to notice that (T,F) is indeed a pdBf, i.e., $T\cap F=\emptyset$, since $|S_l|\geqslant 2$, for every $l=1,\ldots,m$.

In order to calculate the coverage function τ given by (15), let us consider an arbitrary $l\in\{1,\ldots,m\}$, and let us evaluate

$$\tau_l=\prod_{j=1}^{n}\bigvee_{k=1}^{n} z_k(\beta_{kj}\vee\overline{\gamma}_{lj})(\overline{\beta}_{kj}\vee\gamma_{lj}).$$

Let us notice first that for any $j\in S_l$ we have:

$$\bigvee_{k=1}^{n} z_k(\beta_{kj}\vee\overline{\gamma}_{lj})(\overline{\beta}_{kj}\vee\gamma_{lj})=z_j,$$

since $(\beta_{kj}\vee\overline{\gamma}_{lj})(\overline{\beta}_{kj}\vee\gamma_{lj})=0$ if and only if $k\neq j$ for every $j\in S_l$.

Let us remark now that for every $j\notin S_l$ we have:

$$\bigvee_{k=1}^{n} z_k(\beta_{kj}\vee\overline{\gamma}_{lj})(\overline{\beta}_{kj}\vee\gamma_{lj})=\bigvee_{k=1}^{j-1} z_k\vee\bigvee_{k=j+1}^{n} z_k,$$

since $(\beta_{kj}\vee\overline{\gamma}_{lj})(\overline{\beta}_{kj}\vee\gamma_{lj})=0$ if and only if $k=j$ for every $j\notin S_l$.

Consequently, since $|S_l|\geqslant 2$, for every $l=1,\ldots,m$, we have:

$$\begin{aligned}\tau_l &= \left(\prod_{j\in S_l}\bigvee_{k=1}^{n} z_k(\beta_{kj}\vee\overline{\gamma}_{lj})(\overline{\beta}_{kj}\vee\gamma_{lj})\right)\left(\prod_{j\notin S_l}\bigvee_{k=1}^{n} z_k(\beta_{kj}\vee\overline{\gamma}_{lj})(\overline{\beta}_{kj}\vee\gamma_{lj})\right)\\ &= \left(\prod_{j\in S_l} z_j\right)\left(\prod_{j\notin S_l}\left(\bigvee_{k=1}^{j-1} z_k\vee\bigvee_{k=j+1}^{n} z_k\right)\right)=\prod_{j\in S_l} z_j.\end{aligned}$$

It follows that the coverage function of (T,F) is $g(z_1,\ldots,z_n)$. □

Corollary 5.3 together with the construction used in the proof of Theorem 5.5 imply the following result, which shows that the complexity of generating all strong spanned patterns of a pdBf is not easier than the dualization of a monotone non-decreasing Boolean function. Various results about the complexity of this dualization problem can be found in [1,17,18,20].

Corollary 5.6. *The dualization of a monotone non-decreasing Boolean function can be reduced in quadratic time to the generation of all strong spanned patterns of a pdBf.*

Corollary 5.7. *For every integer $n\geqslant 2$, there exists a pdBF (T,F) depending on $2n$ variables, such that $|T|=2n$, $|F|=n$, and the number of strong spanned patterns of (T,F) is 2^n.*

Proof. Let us consider the following Boolean function

$$g(z_1,\ldots,z_{2n})=\bigvee_{i=1}^{n} z_i z_{n+i},$$

and construct the pdBf (T, F) as in the proof of Theorem 5.5. It is well know (see, e.g., [17]) that $g(z_1, \ldots, z_{2n})$ has 2^n maximal false points, and therefore, by Corollary 5.3, (T, F) has 2^n strong spanned patterns. □

It is clear from the above corollary that the generation of all strong spanned patterns can require time which is exponential in the size of the pdBf. If the complexity of the generation algorithm is evaluated in terms of both the input and output lengths, then by Corollary 5.6, the existence of a polynomial total time algorithm for this problem would imply the possibility to dualize a positive DNF in time which is polynomial in the length of the DNF and its dual CNF. Note that the best-known dualization algorithm is quasi-polynomial [20].

Formula (15) provides an expression of the coverage function τ, which is of polynomial length in the size of (T, F). However, the Boolean expression in (15) is neither a CNF nor a DNF. Corollary 5.7 implies that every CNF representation of τ may have exponential length in the size of (T, F). The following result shows that the same phenomenon may also occur with the DNF representations of τ.

Theorem 5.8. *For every integer $n \geqslant 2$, there exists a pdBf (T, F) depending on n variables, such that $|T| = 2n - 2$, $|F| = 1$, and its coverage function τ has $2^{n-1} - 1$ prime implicants.*

Proof. Let us construct the pdBf (T, F) depending on n variables, having only one false point $(1, 1, \ldots, 1)$, and having $2n - 2$ true points defined by:

$$\beta_{kj} = \begin{cases} 1 & \text{if } k < n,\ j \in \{k, n\}, \\ 1 & \text{if } k \geqslant n,\ j = k - n + 1, \\ 0 & \text{otherwise,} \end{cases}$$

where $k = 1, 2, \ldots, 2n - 2$ and $j = 1, \ldots, n$.

It follows from (15) that the coverage function τ of (T, F) is

$$g(z_1, \ldots, z_{2n-2}) = \prod_{j=1}^{n-1} (z_j \vee z_{j+n-1}) \wedge \left(\bigvee_{k=1}^{n-1} z_k \right).$$

It is easy to see that every prime implicant of this function has the form:

$$\bigwedge_{j=1}^{n-1} (p_j z_j \vee \overline{p}_j z_{j+n-1}),$$

where $(p_1, \ldots, p_{n-1})$ is a (0, 1) vector of parameters. Clearly, every such (0, 1) vector different from $(0, \ldots, 0)$ defines a prime implicant of τ, and no two vectors define the same prime implicant. Hence, it follows that the number of prime implicants of τ is $2^{n-1} - 1$. □

It is interesting to compare Corollary 5.6 with Theorem 5.8. The corollary shows that the generation of all strong spanned patterns is not easier than dualization. Are the two problems equivalent? They would certainly be, if a DNF representation of τ were easy to construct. Theorem 5.8 could be interpreted as indicating that the two problems are not equivalent, since any DNF representation of τ may have exponential length.

6. Analytical description of Pareto-optimal patterns

We have described in Section 4 computational procedures for transforming a pattern to a better Pareto-optimal one for different types of Pareto-optimality. In Section 5, the analysis of computational complexity of generating all Pareto-optimal patterns has been based on analytical models using decision variables associated to the observations in the data set. In view of the fact that the number of attributes is typically much smaller than that of observations, a more practical approach to generating all Pareto-optimal patterns should utilize analytical models based on decision variables associated not to the observations but to the attributes. Such analytical models are developed in this section by characterizing the Pareto-optimal patterns studied in this paper as the solutions of certain Boolean equations.

We shall present below characterizations of those patterns P which are Pareto-optimal with respect to a certain criterion. For this purpose, we shall use Boolean variables $y_1, \ldots, y_n$ associated to the original variables $x_1, \ldots, x_n$ of the pdBf (T, F), with

the following meaning:

$$y_j = \begin{cases} 1 & \text{if } x_j \text{ or } \overline{x}_j \text{ is present in } P, \\ 0 & \text{otherwise.} \end{cases}$$

We shall sometimes denote the pattern described by the values of the variables $y_1, \ldots, y_n$ as $P_{y_1,\ldots,y_n}$.

6.1. Description of patterns and prime patterns

Without loss of generality, we shall characterize those Pareto-optimal patterns P which cover a particular true point, say $(\alpha_1, \ldots, \alpha_n) \in T$. As in the previous section, we shall denote the points of $T \setminus (\alpha_1, \ldots, \alpha_n)$ by $(\beta_{k1}, \ldots, \beta_{kn})$, for $k = 1, \ldots, |T| - 1$, while the points of F will be denoted by $(\gamma_{l1}, \ldots, \gamma_{ln})$, for $l = 1, \ldots, |F|$. Using these notations, let us characterize those positive patterns P which cover the observation $(\alpha_1, \ldots, \alpha_n)$. This coverage requirement means that

$$P_{y_1,\ldots,y_n}(x_1, \ldots, x_n) = \prod_{j=1}^{n} (x_j^{\alpha_j} \vee \overline{y}_j). \tag{16}$$

Since this pattern P does not cover any negative points, the decision variables $y_1, \ldots, y_n$ must satisfy the following condition:

$$\bigvee_{l=1}^{|F|} \prod_{j=1}^{n} (\gamma_{lj}^{\alpha_j} \vee \overline{y}_j) = 0. \tag{17}$$

Since $\gamma_{lj}^{\alpha_j} = 1$ iff $\gamma_{lj} = \alpha_j$, condition (17) implies that all the positive patterns covering $(\alpha_1, \ldots, \alpha_n)$ are described by the solutions of the following Boolean equation:

$$\pi = \bigvee_{l=1}^{|F|} \prod_{j \mid \alpha_j \neq \gamma_{lj}} \overline{y}_j = 0. \tag{18}$$

Example 6.1. For the pdBf given in Table 1, all the positive patterns covering the point (10111) are described by the solutions of the Boolean equation

$$\pi = \overline{y}_3\overline{y}_4\overline{y}_5 \vee \overline{y}_1\overline{y}_4\overline{y}_5 \vee \overline{y}_3\overline{y}_5 \vee \overline{y}_1\overline{y}_4 \vee \overline{y}_4\overline{y}_5 = 0,$$

or simply

$$\pi = \overline{y}_1\overline{y}_4 \vee \overline{y}_3\overline{y}_5 \vee \overline{y}_4\overline{y}_5 = 0. \tag{19}$$

While there is a one-to-one correspondence between the solutions of (18) and the positive patterns covering $(\alpha_1, \ldots, \alpha_n)$, different implicants of the complement $\overline{\pi}$ of π may describe the same positive pattern. However, since (18) implies that $\overline{\pi}$ is a monotone non-decreasing Boolean function, each of its *prime implicants* is of the form $\prod_{j \in S} y_j$, and describes a distinct positive prime pattern $P = \prod_{j \in S} x_j^{\alpha_j}$ covering $(\alpha_1, \ldots, \alpha_n)$.

Example 6.2. Continuing Example 6.1, we find:

$$\overline{\pi} = y_1y_5 \vee y_3y_4 \vee y_4y_5, \tag{20}$$

showing that the prime patterns covering the point (10111) are x_1x_5, x_3x_4, and x_4x_5.

6.2. Description of spanned patterns

We shall turn now our attention to characterizing spanned patterns. It follows from Theorem 4.5 that a pattern $P_{y_1,\ldots,y_n}$ covering $(\alpha_1, \ldots, \alpha_n)$ is spanned iff $y_p = 0$ exactly when there exists $(\beta_{k1}, \ldots, \beta_{kn}) \in \mathrm{Cov}(P_{y_1,\ldots,y_n})$ having $\beta_{kp} \neq \alpha_p$. Notice that if $(\beta_{k1}, \ldots, \beta_{kn}) \in \mathrm{Cov}(P_{y_1,\ldots,y_n})$ and $\beta_{kp} \neq \alpha_p$, then the assumption that $(\alpha_1, \ldots, \alpha_n) \in \mathrm{Cov}(P_{y_1,\ldots,y_n})$ implies $y_p = 0$. Therefore, a pattern $P_{y_1,\ldots,y_n}$ covering $(\alpha_1, \ldots, \alpha_n)$ is spanned iff $y_p = 0$ implies that there exists $(\beta_{k1}, \ldots, \beta_{kn}) \in \mathrm{Cov}(P_{y_1,\ldots,y_n})$ having $\beta_{kp} \neq \alpha_p$.

Since $(\beta_{k1}, \ldots, \beta_{kn}) \in \mathrm{Cov}(P_{y_1,\ldots,y_n})$, it follows from (16) that

$$\prod_{j=1}^{n} (\beta_{kj}^{\alpha_j} \vee \overline{y}_j) = 1,$$

or equivalently,

$$\bigvee_{j=1}^{n} y_j(\alpha_j \overline{\beta}_{kj} \vee \overline{\alpha}_j \beta_{kj}) = 0.$$

The condition $\beta_{kp} \neq \alpha_p$ can be expressed as

$$\alpha_p \beta_{kp} \vee \overline{\alpha}_p \overline{\beta}_{kp} = 0.$$

Therefore, $y_p = 0$ implies

$$\prod_{k=1}^{|T|-1} \left(\alpha_p \beta_{kp} \vee \overline{\alpha}_p \overline{\beta}_{kp} \vee \bigvee_{j=1}^{n} y_j(\alpha_j \overline{\beta}_{kj} \vee \overline{\alpha}_j \beta_{kj}) \right) = 0. \tag{21}$$

Since (21) must hold for every p, a pattern is spanned if and only if its characteristic vector $(y_1, \ldots, y_n)$ satisfies the following condition

$$\bigvee_{p=1}^{n} \overline{y}_p \prod_{k=1}^{|T|-1} \left(\alpha_p \beta_{kp} \vee \overline{\alpha}_p \overline{\beta}_{kp} \vee \bigvee_{j=1}^{n} y_j(\alpha_j \overline{\beta}_{kj} \vee \overline{\alpha}_j \beta_{kj}) \right) = 0. \tag{22}$$

In conclusion, all the positive spanned patterns covering $(\alpha_1, \ldots, \alpha_n)$ are described by the solutions of the system of Boolean equations (18) and 22, or simply by the solutions of the following equation:

$$\bigvee_{l=1}^{|F|} \prod_{j=1}^{n} \overline{y}_j(\alpha_j \overline{\gamma}_{lj} \vee \overline{\alpha}_j \gamma_{lj}) \vee \bigvee_{p=1}^{n} \overline{y}_p \prod_{k=1}^{|T|-1} \left(\alpha_p \beta_{kp} \vee \overline{\alpha}_p \overline{\beta}_{kp} \vee \bigvee_{j=1}^{n} y_j(\alpha_j \overline{\beta}_{kj} \vee \overline{\alpha}_j \beta_{kj}) \right) = 0. \tag{23}$$

Example 6.3. All the positive spanned patterns covering the point (10111) in the pdBf of Table 1 are described by the solutions of the Boolean equation

$$\overline{y}_1\overline{y}_4 \vee \overline{y}_3\overline{y}_5 \vee \overline{y}_4\overline{y}_5 \vee \overline{y}_1 y_3 \vee \overline{y}_3 y_1 \vee \overline{y}_4 y_2(\overline{y}_4 y_2 \vee \overline{y}_4 y_5) \vee (\overline{y}_5 y_2 \vee \overline{y}_5 y_4) = 0,$$

or, after simplification,

$$y_1\overline{y}_3 \vee \overline{y}_1 y_3 \vee \overline{y}_1\overline{y}_4 \vee y_2\overline{y}_4 \vee \overline{y}_5 = 0. \tag{24}$$

The solutions of this equation are given in Table 4.

Therefore, the positive spanned patterns covering the point (10111) are x_4x_5, $\overline{x}_2x_4x_5$, $x_1x_3x_4x_5$, $x_1\overline{x}_2x_3x_4x_5$, and $x_1x_3x_5$.

Table 4
Solutions of (24)

y_1	y_2	y_3	y_4	y_5
0	0	0	1	1
0	1	0	1	1
1	0	1	1	1
1	1	1	1	1
1	0	1	0	1

6.3. Description of strong spanned patterns

In view of Theorem 4.8, a spanned pattern $P_{y_1,\ldots,y_n}$ covering $(\alpha_1,\ldots,\alpha_n)$ is strong if and only if for any $(\beta_{k1},\ldots,\beta_{kn}) \in T\backslash\mathrm{Cov}(P_{y_1,\ldots,y_n})$ there exists $(\gamma_{l1},\ldots,\gamma_{ln}) \in F$ such that for every index j, if $y_j = 1$ and $\beta_{kj} = \alpha_j$, then $\gamma_{lj} = \alpha_j$. This last condition can be expressed as

$$y_j(\alpha_j\beta_{kj} \vee \overline{\alpha}_j\overline{\beta}_{kj})(\alpha_j\overline{\gamma}_{lj} \vee \overline{\alpha}_j\gamma_{lj}) = 0,$$

or simply,

$$y_j(\alpha_j\beta_{kj}\overline{\gamma}_{lj} \vee \overline{\alpha}_j\overline{\beta}_{kj}\gamma_{lj}) = 0.$$

Since there must exist an index l such that the last condition holds for every j, we have:

$$\prod_{l=1}^{|F|}\left[\bigvee_{j=1}^{n} y_j(\alpha_j\beta_{kj}\overline{\gamma}_{lj} \vee \overline{\alpha}_j\overline{\beta}_{kj}\gamma_{lj})\right] = 0. \tag{25}$$

It follows from (16) that the condition $(\beta_{k1},\ldots,\beta_{kn}) \in T\backslash\mathrm{Cov}(P_{y_1,\ldots,y_n})$ can be expressed as

$$\bigvee_{j=1}^{n} y_j(\alpha_j\overline{\beta}_{kj} \vee \overline{\alpha}_j\beta_{kj}) = 1. \tag{26}$$

Since a spanned pattern $P_{y_1,\ldots,y_n}$ covering $(\alpha_1,\ldots,\alpha_n)$ is strong iff (26) implies (25), all the strong spanned patterns covering $(\alpha_1,\ldots,\alpha_n)$ are described by the solutions of (23) and

$$\bigvee_{k=1}^{|T|-1}\left\{\left[\bigvee_{j=1}^{n} y_j(\alpha_j\overline{\beta}_{kj} \vee \overline{\alpha}_j\beta_{kj})\right]\prod_{l=1}^{|F|}\left[\bigvee_{j=1}^{n} y_j(\alpha_j\beta_{kj}\overline{\gamma}_{lj} \vee \overline{\alpha}_j\overline{\beta}_{kj}\gamma_{lj})\right]\right\} = 0. \tag{27}$$

In other words, all the strong spanned patterns covering $(\alpha_1,\ldots,\alpha_n)$ are described by the solutions of the following equation.

$$\bigvee_{l=1}^{|F|}\prod_{j=1}^{n}\overline{y}_j(\alpha_j\overline{\gamma}_{lj} \vee \overline{\alpha}_j\gamma_{lj}) \vee \bigvee_{p=1}^{n}\overline{y}_p\prod_{k=1}^{|T|-1}\left(\alpha_p\beta_{kp} \vee \overline{\alpha}_p\overline{\beta}_{kp} \vee \bigvee_{j=1}^{n} y_j(\alpha_j\overline{\beta}_{kj} \vee \overline{\alpha}_j\beta_{kj})\right) \vee$$
$$\bigvee_{k=1}^{|T|-1}\left\{\left[\bigvee_{j=1}^{n} y_j(\alpha_j\overline{\beta}_{kj} \vee \overline{\alpha}_j\beta_{kj})\right]\prod_{l=1}^{|F|}\left[\bigvee_{j=1}^{n} y_j(\alpha_j\beta_{kj}\overline{\gamma}_{lj} \vee \overline{\alpha}_j\overline{\beta}_{kj}\gamma_{lj})\right]\right\} = 0. \tag{28}$$

Example 6.4. Let us now continue Example 6.3 and describe all the strong spanned patterns covering the point (10111). Assuming that $k = 2,\ldots,5$ correspond to the points $b,\ldots,e$ of T, we present in Table 5 the corresponding expressions of (25) and (26) after simplifications.

Using the results in Table 5, we have:

$$\begin{aligned}(27) &= (y_1 \vee y_3)y_4y_5 \vee y_2(y_1y_5 \vee y_3y_4 \vee y_4y_5) \vee (y_2 \vee y_4)y_1y_5\\ &= y_1y_2y_5 \vee y_1y_4y_5 \vee y_2y_3y_4 \vee y_2y_4y_5 \vee y_3y_4y_5.\end{aligned}$$

Since the expression of (23) was calculated in Example 6.3 (see (24)), we see that Eq. (28) for our example becomes

$$(28) = y_1\overline{y}_3 \vee \overline{y}_1y_3 \vee \overline{y}_1\overline{y}_4 \vee y_2\overline{y}_4 \vee \overline{y}_5 \vee y_1y_2y_5 \vee y_1y_4y_5 \vee y_2y_3y_4 \vee y_2y_4y_5 \vee y_3y_4y_5 = 0,$$

or, after simplification,

$$y_1\overline{y}_3 \vee \overline{y}_1y_3 \vee y_1y_4 \vee \overline{y}_1\overline{y}_4 \vee y_2 \vee \overline{y}_5 = 0. \tag{29}$$

The solutions of this equation are given in Table 6.

Therefore, the positive strong spanned patterns covering the point (10111) are x_4x_5 and $x_1x_3x_5$.

Table 5
Evaluation of (25) and (26)

k	(26)	(25)
2	$y_1 \vee y_3$	$(y_4 \vee y_5)y_5y_4 = y_4y_5$
3	y_2	$(y_3 \vee y_4 \vee y_5)(y_1 \vee y_4 \vee y_5)(y_3 \vee y_5)(y_1 \vee y_4)(y_4 \vee y_5) = y_1y_5 \vee y_3y_4 \vee y_4y_5$
4	$y_2 \vee y_4$	$(y_3 \vee y_5)(y_1 \vee y_5)y_1y_5 = y_1y_5$
5	$y_2 \vee y_4 \vee y_5$	$y_3y_1 \cdot 0 = 0$

Table 6
Solutions of (29)

y_1	y_2	y_3	y_4	y_5
0	0	0	1	1
1	0	1	0	1

Table 7
Datasets used in empirical evaluation

Name of dataset	Number of observations			Attributes		Observation with missing values	
	Positive	Negative	Total	Binary or categorical	Numerical	Number	Handling
Australian credit	307	383	690	9	6	37	Removed
Breast cancer Wisconsin	241	458	699	0	9	16	Removed
Boston housing	253	253	506	1	13	0	
Pima Indians diabetes	268	500	768	0	8	0	
Heart disease(Cleveland)	139	164	303	8	6	6	Removed
Oil	702	930	1632	0	7	0	
Congressional voting	267	168	435	16	0	203	Substituted

7. Empirical evaluation

The aim of this section is to empirically evaluate the relative merits of various theories, each of which consists of a specific type of Pareto-optimal patterns. The evaluation is based on comparing the classification performances of three such Pareto-optimal theories (prime, strong spanned, and strong prime), made on several real life data sets which are widely used in the machine learning literature. As a benchmark, we also evaluate the performance of the "standard implementation" of LAD (as described in [6]) on the same datasets.

Note that—although the uses of patterns in LAD are not confined to classification (see [6])—classification performance is used here in the empirical evaluation of the relative merits of various types of Pareto-optimal patterns because of the public availability of standard classification datasets.

Seven datasets are used in our experiments, six taken from the Irvine repository (see [2]), and one ("Oil") provided by the Chevron Corporation (see description in [6]). The key parameters concerning these datasets are presented in Table 7. This table also indicates the number of observations which have some missing attribute values and which have therefore been removed from the datasets in our experiments. In the special case of "Congressional voting", the number of observations with missing attribute values is so large that, instead of removing them from consideration, we have simply substituted the missing Boolean values with the majority value for the corresponding class.

In order to be able to apply the Boolean methods of this paper to numerical data, we "binarize" the datasets, i.e. replace the numerical attributes by one or more binary attributes, which indicate whether the value of the corresponding numerical attribute is above or below certain thresholds; the binarization procedure, including the determination of the thresholds, is described in [6].

For the sake of comparability, the Pareto-optimal theories used in the comparisons are all produced starting from the same initial theory, by applying to it the three transformations described in Section 4. As mentioned in Section 4.4, a most straightforward way of obtaining the initial theory is to rewrite the given pdBf (T, F) as the theory consisting of all its minterms. By applying the three transformations described in Section 4 to this minterm-generated theory, we obtain Pareto-optimal theories consisting either of prime patterns, or of strong spanned patterns, or of strong prime patterns. The reason for choosing the minterm theory as the starting point is its ready availability and the fact that it is not biased towards any of the types of Pareto-optimality considered here. Note that the Pareto-optimal theories obtained in this way are not necessarily the best performing ones within their categories. However, this is not a hindrance, since our objective is only to evaluate the *relative* merits of the above-mentioned three main concepts of Pareto-optimality. Our experimental evaluation is not aimed at producing an LAD algorithm with the best possible performance, but is designed so as to make the results truly comparable by guaranteeing that the only difference between the three theories is the type of the Pareto-optimal patterns used.

Conforming to the implementation of LAD [6], the classification process is symmetric with respect to both positive and negative observations, and consists of the following two stages. First, for each type of Pareto-optimality we follow the procedure described above to construct a positive and a negative Pareto-optimal theory. Second, the positive and negative patterns obtained in this way are combined into a "discriminant" (a pseudo-Boolean polynomial) using as positive and negative weights their respective coverages (separately normalized for the positive and the negative patterns; see [6]). Those new observations are classified as positive or negative according to the sign of this discriminant. New observations for which the value of the discriminant is 0 are not classified.

The data sets are randomly partitioned into two parts of equal sizes. A theory is derived using one of these parts as training set, and its performance is then evaluated on the other part (testing set); afterwards the roles of the training and testing sets are reversed, and the experiment is repeated, thus completing a 2-fold cross-validation. For each data set, the average performance (along with the standard deviation) on 25 such random partitions is reported in Table 8. The classification performance of the theories on the testing sets is evaluated using the following parameters:

- the percentages of those positive and negative observations which are correctly classified;
- the percentage of those positive (negative) observations which are misclassified as negative (positive);
- the percentages of those positive and negative observations which are not classified (these appear in the tables in the columns marked "?").

The aggregate measure of classification inaccuracy can take into account either the number of errors alone, or both the number of errors and the number of unclassified observations, or some combination of these points of view. Let m be the percentage of misclassified observations, and u be the percentage of unclassified observations, and let us define the aggregate "cost" of classification inaccuracy as

$$c = m + \lambda u,$$

where the parameter $\lambda \in [0, 1]$ represents the relative weight of unclassified observations. The values of c in our experiments for three values of λ (0, 0.5, 1) are presented in Table 9[1] and Figs. 1,2,3. While we report the results of the experiments for all the three values of λ, the most realistic estimate of the cost of classification inaccuracy is given by $\lambda = 0.5$. Indeed, the estimates given by $\lambda = 0$ impose no penalty for a theory which would never give an answer, while the estimates given by $\lambda = 1$ overpenalize the decision of not providing an answer due to high uncertainty. In other words, the estimate for $\lambda = 0$ assumes that "no answer" is *always* the correct answer, while the estimate for $\lambda = 1$ assumes that "no answer" is *never* the correct answer. On the other hand, the estimate for $\lambda = 0.5$ coincides with the expected error rate of a procedure which accepts the answer of a theory whenever it is given, and in case of "no answer", selects the answer by flipping a fair coin.

Finally, in Table 10 we present the results of pairwise comparisons of classification inaccuracies of the four types of theories discussed in this section. The average differences between the classification inaccuracies of the compared theories are reported in the columns Δc for each of the three values of λ. In the columns *Signif.* we report the critical probability values given by the t-test, and indicate by * those differences which are significant at the level of 99.9%.

The results presented above show that the classification performance of Pareto-optimal theories is roughly comparable to that of LAD, which is known to be a competitive classification method (see [6]). It is not entirely surprising that in most cases LAD does statistically outperform the other theories (although by a slim margin), since the LAD theory results from an extensive pattern enumeration procedure, while the Pareto-optimal theories used here are constructed in a simplistic greedy way. Since these Pareto-optimal theories do exhibit reasonable performance, their pairwise comparison is justified, and can be

[1] The slight difference in the LAD performance reported here and in [6] is due to the fact that in this paper (1) observations with missing attribute values were removed from the datasets, and (2) only 100%-homogeneous patterns were used.

Table 8
Classification accuracy

	Observ.	Classif.	Credit		Breast cancer		Housing		Diabetes		Heart		Oil		Voting	
			Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.
LAD	+	Correct	**87.4**	3.4	**95.2**	2.6	**87.1**	3.4	**64.3**	5.5	**80.3**	6.7	**92.7**	1.9	**96.2**	2.6
	+	Wrong	**11.3**	2.9	**3.9**	2.1	**12.0**	3.2	**21.3**	5.2	**17.5**	5.7	**3.9**	1.1	**3.2**	2.1
	+	?	**1.4**	1.5	**0.9**	1.2	**0.9**	1.1	**14.4**	5.7	**2.2**	2.5	**3.4**	1.6	**0.5**	1.2
	–	Correct	**86.6**	2.9	**97.3**	0.9	**84.3**	4.5	**72.0**	4.9	**81.5**	5.8	**90.5**	2.2	**95.0**	2.4
	–	Wrong	**12.0**	2.7	**2.6**	0.8	**14.1**	4.0	**18.0**	2.9	**16.5**	5.6	**5.8**	1.5	**4.1**	1.5
	–	?	**1.3**	1.2	**0.1**	0.3	**1.6**	1.4	**10.0**	4.9	**2.0**	2.3	**3.7**	1.6	**0.9**	2.3
	All	Correct	**87.0**	2.1	**96.5**	1.0	**85.7**	2.4	**69.3**	3.7	**80.8**	4.2	**91.7**	1.2	**95.5**	2.1
	All	Wrong	**11.7**	1.7	**3.1**	0.8	**13.1**	2.3	**19.2**	2.5	**17.1**	3.5	**4.8**	0.8	**3.8**	1.0
	All	?	**1.3**	1.1	**0.4**	0.5	**1.3**	1.0	**11.5**	5.0	**2.1**	1.9	**3.5**	1.3	**0.8**	1.8
Strong spanned	+	Correct	**81.6**	4.3	**91.5**	3.0	**82.1**	4.4	**50.0**	5.6	**73.4**	8.1	**92.4**	1.9	**94.1**	2.8
	+	Wrong	**10.6**	3.0	**4.1**	2.2	**12.6**	3.7	**27.6**	4.9	**18.9**	6.1	**4.7**	1.2	**3.9**	2.4
	+	?	**7.8**	3.0	**4.3**	2.4	**5.3**	2.3	**22.4**	5.6	**7.6**	4.5	**2.9**	1.5	**2.0**	1.8
	–	Correct	**82.8**	3.3	**96.7**	1.1	**83.4**	4.1	**73.6**	3.9	**78.4**	5.4	**89.5**	2.2	**95.9**	1.7
	–	Wrong	**11.8**	2.5	**2.5**	1.0	**11.3**	3.7	**18.2**	3.1	**14.5**	5.5	**5.9**	1.6	**3.1**	1.3
	–	?	**5.4**	2.1	**0.8**	0.7	**5.4**	2.3	**8.2**	2.4	**7.1**	3.7	**4.6**	1.9	**1.0**	1.3
	All	Correct	**82.2**	2.3	**94.9**	1.1	**82.7**	1.9	**65.4**	3.0	**76.0**	3.3	**91.0**	1.2	**95.2**	1.2
	All	Wrong	**11.3**	1.5	**3.1**	0.9	**12.0**	2.0	**21.5**	2.0	**16.6**	3.0	**5.2**	0.8	**3.4**	1.0
	All	?	**6.5**	1.8	**2.1**	1.0	**5.3**	1.7	**13.2**	3.0	**7.4**	3.0	**3.7**	1.3	**1.4**	1.1
Strong prime	+	Correct	**83.6**	4.0	**91.5**	3.3	**83.3**	3.8	**56.4**	5.2	**75.5**	7.2	**92.8**	1.8	**95.1**	2.5
	+	Wrong	**11.3**	3.2	**5.8**	2.7	**14.2**	3.5	**30.1**	5.1	**19.4**	6.4	**4.7**	1.2	**3.7**	2.3
	+	?	**5.1**	2.2	**2.7**	1.8	**2.4**	1.3	**13.5**	4.3	**5.1**	3.4	**2.5**	1.4	**1.2**	1.4
	–	Correct	**84.0**	3.0	**97.4**	1.0	**85.2**	3.7	**75.2**	3.6	**79.7**	5.3	**89.7**	2.1	**95.9**	1.6
	–	Wrong	**12.3**	2.6	**2.3**	1.0	**11.7**	3.5	**20.2**	3.3	**16.3**	5.4	**6.2**	1.5	**3.5**	1.3
	–	?	**3.7**	1.4	**0.3**	0.4	**3.1**	1.9	**4.5**	1.9	**4.0**	2.4	**4.0**	1.8	**0.6**	0.8
	All	Correct	**83.8**	2.1	**95.3**	1.1	**84.2**	1.6	**68.6**	2.7	**77.7**	3.0	**91.4**	1.2	**95.6**	1.1
	All	Wrong	**11.8**	1.5	**3.5**	1.0	**13.0**	1.8	**23.7**	2.2	**17.8**	3.0	**5.4**	0.8	**3.6**	0.9
	All	?	**4.4**	1.4	**1.1**	0.7	**2.8**	1.3	**7.7**	2.3	**4.5**	2.3	**3.2**	1.2	**0.9**	0.8
Prime	+	Correct	**73.2**	5.8	**89.7**	3.7	**74.0**	4.9	**13.2**	6.5	**70.5**	6.4	**74.5**	5.8	**91.8**	4.5
	+	Wrong	**12.7**	3.4	**5.0**	2.7	**9.6**	3.5	**10.6**	4.7	**21.1**	5.7	**3.7**	1.0	**6.3**	3.7
	+	?	**14.1**	5.2	**5.3**	2.6	**16.4**	5.4	**76.2**	7.9	**8.4**	4.6	**21.8**	5.7	**1.9**	2.4
	–	Correct	**78.6**	4.5	**94.2**	1.6	**65.9**	7.1	**41.3**	5.7	**73.1**	8.4	**79.6**	4.1	**93.7**	2.4
	–	Wrong	**7.3**	2.4	**3.5**	1.4	**9.6**	3.9	**5.8**	3.1	**14.7**	5.9	**3.8**	1.6	**5.5**	2.1
	–	?	**14.1**	4.5	**2.3**	1.4	**24.5**	7.5	**52.9**	6.6	**12.1**	6.9	**16.6**	4.0	**0.9**	0.9
	All	Correct	**76.1**	3.4	**92.6**	1.5	**69.8**	4.4	**31.4**	4.3	**71.8**	3.9	**76.8**	3.5	**92.9**	2.0
	All	Wrong	**9.8**	1.7	**4.0**	1.3	**9.7**	2.4	**7.5**	2.6	**17.7**	3.3	**3.7**	1.0	**5.8**	1.8
	All	?	**14.1**	4.0	**3.4**	1.5	**20.5**	5.8	**61.1**	6.6	**10.5**	4.7	**19.4**	3.9	**1.3**	1.1

Table 9
Cost of classification inaccuracy

	λ	Credit		Breast cancer		Housing		Diabetes		Heart		Oil		Voting	
		Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.
LAD	0	**11.7**	1.7	**3.1**	0.8	**13.1**	2.3	**19.2**	2.5	**17.1**	3.5	**4.8**	0.8	**3.8**	1.0
	0.5	**12.4**	1.8	**3.3**	0.8	**13.7**	2.3	**24.9**	1.9	**18.1**	3.7	**6.5**	0.8	**4.1**	1.4
	1	**13.0**	2.1	**3.5**	1.0	**14.3**	2.4	**30.7**	3.7	**19.2**	4.2	**8.3**	1.2	**4.5**	2.1
Strong spanned	0	**11.3**	1.5	**3.1**	0.9	**12.0**	2.0	**21.5**	2.0	**16.6**	3.0	**5.2**	0.8	**3.4**	1.0
	0.5	**14.5**	1.7	**4.1**	0.8	**14.7**	1.8	**28.1**	2.1	**20.3**	2.8	**7.1**	0.8	**4.1**	0.9
	1	**17.8**	2.3	**5.1**	1.1	**17.3**	1.9	**34.6**	3.0	**24.0**	3.3	**9.0**	1.2	**4.8**	1.2
Strong prime	0	**11.8**	1.5	**3.5**	1.0	**13.0**	1.8	**23.7**	2.2	**17.8**	3.0	**5.4**	0.8	**3.6**	0.9
	0.5	**14.0**	1.7	**4.1**	1.0	**14.4**	1.6	**27.5**	2.2	**20.1**	2.8	**7.0**	0.8	**4.0**	0.9
	1	**16.2**	2.1	**4.7**	1.1	**15.8**	1.6	**31.4**	2.7	**22.3**	3.0	**8.6**	1.2	**4.4**	1.1
Prime	0	**9.8**	1.7	**4.0**	1.3	**9.7**	2.4	**7.5**	2.6	**17.7**	3.3	**3.7**	1.0	**5.8**	1.8
	0.5	**16.8**	1.8	**5.7**	1.2	**19.9**	2.0	**38.0**	1.4	**23.0**	2.7	**13.5**	1.7	**6.4**	1.8
	1	**23.9**	3.4	**7.4**	1.5	**30.2**	4.4	**68.5**	4.3	**28.2**	3.9	**23.2**	3.5	**7.1**	2.0

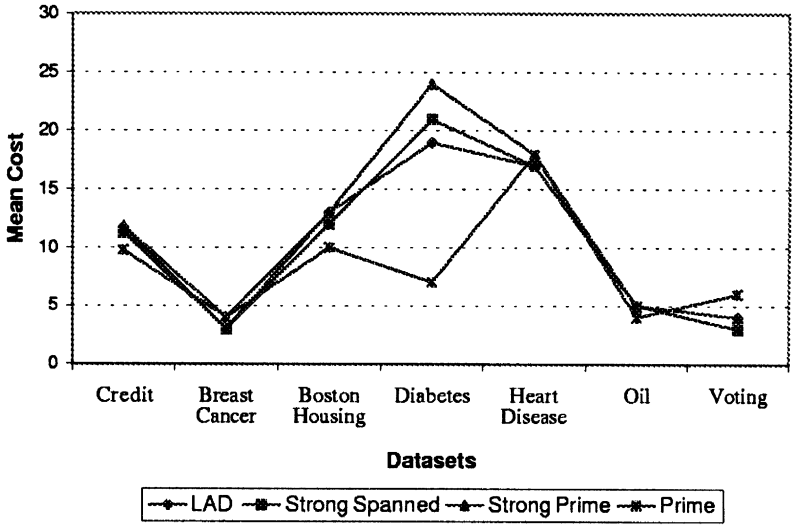

Fig. 1. Cost of classification inaccuracy for $\lambda = 0$.

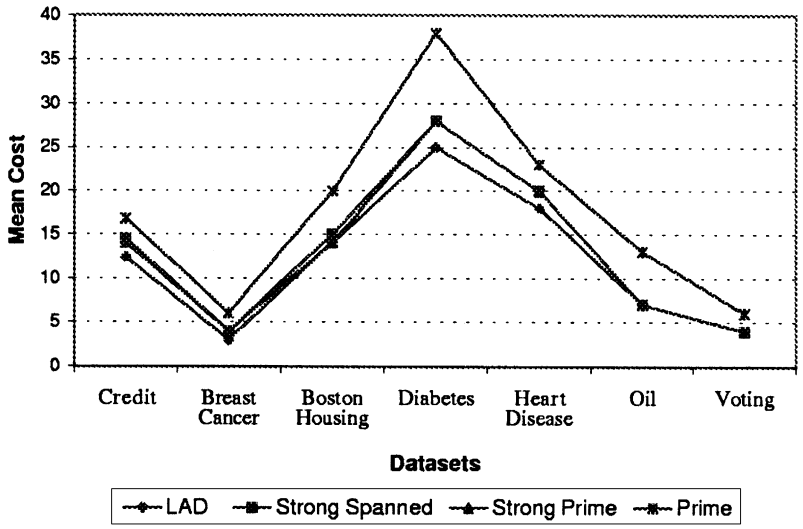

Fig. 2. Cost of classification inaccuracy for $\lambda = 0.5$.

a basis for meaningful conclusions about the relative merits of various types of Pareto-optimality. In view of the discussion above, we shall derive our conclusions based primarily on the results corresponding to $\lambda = 0.5$. The main conclusions are the following:

- *The simplicity preference does not seem to lead to a good performance.* Indeed, in the above results, the classification accuracy of the minterm-generated prime theories is statistically worse than that of the other three types of theories.
- *The evidential preference in itself seems to lead to a good performance.* Indeed, in all cases the best performing Pareto-optimal theory is one of the two strong theories. Moreover, although the preferences $\varepsilon \mid \Sigma$ and $\varepsilon \mid \sigma$ are obtained by lexicographically refining the evidential preference ε in two opposite ways, in most cases the corresponding (minterm-generated) strong spanned and strong prime theories have statistically insignificant differences in overall performance.

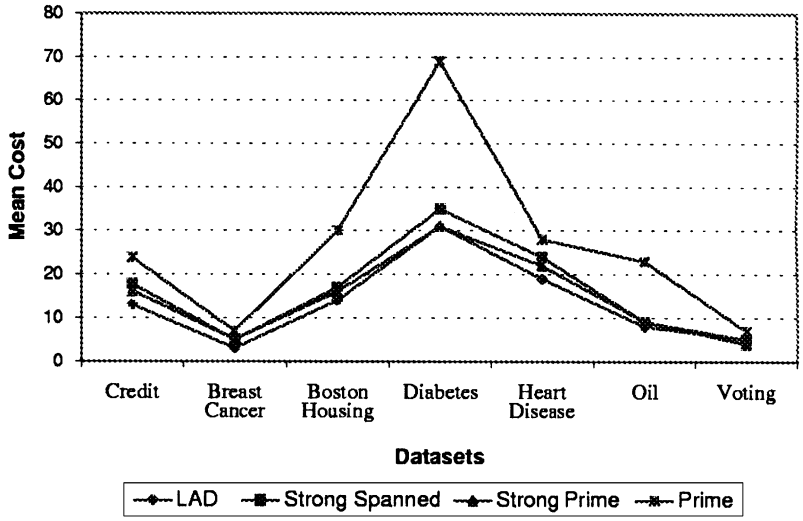

Fig. 3. Cost of classification inaccuracy for $\lambda = 1$.

- The (minterm-generated) strong spanned and strong prime theories achieve a different balance in performance: the strong spanned theories seem to reduce the number of errors, while the strong prime theories seem to reduce the number of unclassified observations. This conclusion is not entirely surprising in view of the conservative bias of selectivity and the aggressive bias of simplicity.

8. Conclusions

Patterns are the main building blocks in LAD, as well as in many other methods of data analysis. This paper introduces an axiomatic approach to comparing the value of various patterns in LAD, leading to the concept of Pareto-optimal patterns.

Prime and strong patterns are the Pareto-optimal patterns which correspond respectively to the simplicity and the evidential preferences. Because of the wide acceptance of the simplicity preference, LAD, as well as many other data analysis techniques, traditionally used prime patterns. While the importance of the coverage of patterns has been widely recognized in the machine learning literature, to the best of the authors knowledge, the concept of strong patterns has never been studied before. This paper focuses on strong patterns, with a special attention to the two extreme cases of strong prime and strong spanned patterns.

It is shown in Section 4 that an arbitrary pattern can be transformed in polynomial time to a better Pareto-optimal one according to any one of the preferences discussed in this paper. On the other hand, generating *all* Pareto-optimal patterns turns out to be intractable, and it is shown in Section 5 that even the set of strong spanned patterns, which are *distinct* (maximally specific) representatives of the classes of evidentially equivalent strong patterns, can have exponential cardinality. The links established in Section 5 between the generation of all strong spanned patterns and the dualization of a positive DNF indicate the intrinsic computational difficulty of the former problem (even if the computational complexity is evaluated in terms of both the input and the output lengths). In Section 6, the Pareto-optimal patterns studied in this paper are characterized as the solutions of certain Boolean equations using decision variables associated to the attributes of the data set.

The computational evaluation of the various types of Pareto-optimal patterns described in Section 7 was carried out on several real life data sets (mostly taken from the Irvine repository). The results seem to indicate that in itself the simplicity preference does not necessarily lead to a good performance, while the evidential preference does. Moreover, the refinement of the evidential preference with respect to *simplicity* will bias the Pareto-optimal theories in favor of reducing the number of *unclassified observations*. On the other hand, the refinement of the evidential preference with respect to *selectivity* will bias the Pareto-optimal theories in favor of reducing the number of *errors*.

To summarize, the use of strong patterns leads to a superior performance, and the use of strong spanned or strong prime patterns reduces the number of errors or unclassified observations, respectively.

Table 10
Pairwise cost comparison of theories

		$\lambda = 0$		$\lambda = 0.5$		$\lambda = 1$	
Dataset	Theories compared	Δ I	Sign.	Δ I	Sign.	Δ I	Sign.
Credit	LAD–Strong spanned	0.42	0.0247*	−2.16	0.0000*	−4.75	0.0000*
	LAD–Strong prime	−0.14	0.4814	−1.65	0.0000*	−3.16	0.0000*
	LAD–Prime	1.93	0.0000*	−4.46	0.0000*	−10.85	0.0000*
	Strong spanned–Strong prime	−0.57	0.0000*	0.51	0.0000*	1.59	0.0000*
	Strong spanned–Prime	1.51	0.0000*	−2.30	0.0000*	−6.10	0.0000*
	Strong prime–Prime	2.07	0.0000*	−2.81	0.0000*	−7.69	0.0000*
Breast cancer	LAD–Strong spanned	−0.01	0.8999	−0.85	0.0000*	−1.68	0.0000*
	LAD–Strong prime	−0.45	0.0005*	−0.83	0.0000*	−1.21	0.0000*
	LAD–Prime	−0.95	0.0000*	−2.44	0.0000*	−3.94	0.0000*
	Strong spanned–Strong prime	−0.43	0.0000*	0.02	0.7922	0.47	0.0000*
	Strong spanned–Prime	−0.93	0.0000*	−1.60	0.0000*	−2.26	0.0000*
	Strong prime–Prime	−0.50	0.0048*	−1.62	0.0000*	−2.73	0.0000*
Housing	LAD–Strong spanned	1.07	0.0008*	−0.96	0.0016*	−2.99	0.0000*
	LAD–Strong prime	0.07	0.8102	−0.69	0.0178*	−1.45	0.0000*
	LAD–Prime	3.41	0.0000*	−6.20	0.0000*	−15.82	0.0000*
	Strong spanned–Strong prime	−1.00	0.0000*	0.27	0.0880	1.54	0.0000*
	Strong spanned–Prime	2.34	0.0000*	−5.24	0.0000*	−12.83	0.0000*
	Strong prime–Prime	3.35	0.0000*	−5.51	0.0000*	−14.37	0.0000*
Diabetes	LAD–Strong spanned	−2.32	0.0000*	−3.13	0.0000*	−3.94	0.0000*
	LAD–Strong prime	−4.56	0.0000*	−2.62	0.0000*	−0.68	0.1742
	LAD–Prime	11.67	0.0000*	−13.09	0.0000*	−37.85	0.0000*
	Strongspanned–Strong prime	−2.24	0.0000*	0.51	0.0005*	3.26	0.0000*
	Strong spanned–Prime	13.99	0.0000*	−9.96	0.0000*	−33.91	0.0000*
	Strong prime–Prime	16.24	0.0000*	−10.47	0.0000*	−37.17	0.0000*
Heart	LAD–Strong spanned	0.49	0.2499	−2.16	0.0000*	−4.82	0.0000*
	LAD–Strong prime	−0.72	0.0706	−1.95	0.0000*	−3.19	0.0000*
	LAD–Prime	−0.61	0.2555	−4.85	0.0000*	−9.08	0.0000*
	Strong spanned–Strong prime	−1.21	0.0000*	0.21	0.2512	1.63	0.0000*
	Strong spanned–Prime	−1.10	0.0400*	−2.68	0.0000*	−4.27	0.0000*
	Strong prime–Prime	0.11	0.8243	−2.89	0.0000*	−5.90	0.0000*
Oil	LAD–Strong spanned	−0.45	0.0000*	−0.55	0.0000*	−0.66	0.0002*
	LAD–Strong prime	−0.63	0.0000*	−0.49	0.0000*	−0.35	0.0347
	LAD–Prime	1.06	0.0000*	−6.91	0.0000*	−14.88	0.0000*
	Strong spanned–Strong prime	−0.18	0.0000*	0.07	0.0732	0.31	0.0000*
	Strong spanned–Prime	1.51	0.0000*	−6.35	0.0000*	−14.22	0.0000*
	Strong prime–Prime	1.69	0.0000*	−6.42	0.0000*	−14.53	0.0000*
Voting	LAD–Strong spanned	0.34	0.0288*	0.03	0.8638	−0.27	0.3885
	LAD–Strong prime	0.17	0.2361	0.13	0.4808	0.09	0.7540
	LAD–Prime	−2.05	0.0000*	−2.30	0.0000*	−2.56	0.0000*
	Strong spanned–Strong prime	−0.17	0.0394*	0.10	0.2756	0.37	0.0049*
	Strong spanned–Prime	−2.39	0.0000*	−2.34	0.0000*	−2.29	0.0000*
	Strong prime–Prime	−2.21	0.0000*	−2.43	0.0000*	−2.66	0.0000*

References

[1] J.C. Bioch, T. Ibaraki, Complexity of identification and dualization of positive Boolean functions, Inform. and Comput. 123 (1995) 51–75.
[2] C.L. Blake, C.J. Merz, UCI Repository of Machine Learning Databases, Department of Information and Computer Science, University of California, Irvine, CA, 1998.http://www.ics.uci.edu/mlearn/MLRepository.html

[3] A. Blumer, A. Ehrenfeucht, D. Haussler, M.K. Warmuth, Occam's razor, Inform. Process. Lett. 24 (1987) 377–380.
[4] E. Boros, V. Gurvich, P.L. Hammer, T. Ibaraki, A. Kogan, Decomposability of partially defined Boolean functions, Discrete Appl. Math. 62 (1995) 51–76.
[5] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, Logical analysis of numerical data, Math. Programming 79 (1997) 163–190.
[6] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz, I. Muchnik, An implementation of logical analysis of data, IEEE Trans. Knowledge and Data Engineering 12 (2) (2000) 292–306.
[7] E. Boros, T. Ibaraki, K. Makino, Error-free and best-fit extensions of partially defined Boolean functions, Inform. and Comput. 140 (2) (1998) 254–283.
[8] E. Boros, T. Ibaraki, K. Makino, Logical analysis of binary data with missing bits, Artificial Intelligence 107 (2) (1999) 219–263.
[9] P. Clark, R. Boswell, Rule induction with CN2: some recent improvements, in: Y. Kodratoff (Ed.), Machine Learning—Proceedings of the Fifth European Conference (EWSL-91), Springer, Berlin, 1991, pp. 151–163. http://www.cs.utexas.edu/users/pclark/papers/newcn.ps
[10] P. Clark, T. Niblett, The CN2 induction algorithm, Mach. Learning 3 (1) (1989) 261–283.
[11] W.W. Cohen, Fast effective rule induction, in: Machine Learning: Proceedings of the Twelth International Conferencem (ML95),Morgan Kaufmann, Lake Tahoe CA, 1995, pp. 115–123.
[12] W.W. Cohen, Y. Singer, A simple, fast, and effective rule learner, in: Proceedings of the Sixteenth National Conference on Artificial Intelligence,AAAI Press, Menlo Park CA, 1999, pp. 335–342.
[13] Y. Crama, P.L. Hammer, T. Ibaraki, Cause-effect relationships and partially defined Boolean functions, Ann. Oper. Res. 16 (1988) 299–326.
[14] P. Domingos, Unifying instance-based and rule-based induction, Machine Learning, 24 (1996) 141–168. http://www.cs.washington.edu/homes/pedrod/mlj96.ps.gz
[15] P. Domingos, Occam's two razors: the sharp and the blunt, in: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining,AAAI Press, New York, NY, 1998, pp. 37–43. http://www.cs.washington.edu/homes/pedrod/kdd98.ps.gz
[16] P. Domingos, The role of Occam's razor in knowledge discovery, Data Mining and Knowledge Discovery, 3 (4) (1999) 409–425. http://www.cs.washington.edu/homes/pedrod/dmkd99.ps.gz
[17] T. Eiter, G. Gottlob, Identifying the minimal transversals of a hypergraph and related problems, SIAM Journal on Computing 24 (1995) 1278–1304.
[18] T. Eiter, G. Gottlob, K. Makino, New results on monotone dualization and generating hypergraph transversals, in: Proceedings of the 34th ACM Symposium on Theory of Computing, (STOC-02), Montreal, Quebec, Canada, 14–22, ACM Press, New York, NY, May 19–21 2002.
[19] O. Ekin, P.L. Hammer, A. Kogan, Convexity and logical analysis of data, Theoret. Comput. Sci. 244 (1–2) (2000) 95–116.
[20] M.L. Fredman, L.Khachiyan. On the complexity of dualization of monotone disjunctive normal forms, J. Algorithms 21 (3) (1996) 618–628.
[21] A.B. Hammer, P.L. Hammer, I.Muchnik. Logical analysis of Chinese productivity patterns, Ann. Oper. Res. 87 (1999) 165–176.
[22] P.L. Hammer, Partially Defined Boolean Functions and Cause-Effect Relationships. International Conference on Multi-Attrubute Decision Making Via OR-Based Expert Systems, University of Passau, Passau, Germany, April 1986.
[23] R.C. Holte, L. Acker, B. Porter, Concept learning and the problem of small disjuncts, in: Ptroceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89),813–818, Morgan Kaufmann, Detroit, MI, 1989. http://www.csi.uottawa.ca/holte/Publications/small_disjuncts.ps
[24] J.R. Quinlan, C45: Programs for Machine Learning, Morgan Kaufmann, San Mateo, CA, 1993.
[25] G. Webb, Further experimental evidence against the utility of Occam's razor, J. Artificial Intelligence Res. 4 (1996) 397–417. http://www3.cm.deakin.edu.au/webb/Papers/bias3.ps.Z
[26] G.M. Weiss, H.Hirsh. A quantitative study of small disjuncts, in: Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000),AAAI Press, Menlo Park CA, 2000, pp. 665–670. http://www.cs.rutgers.edu/gweiss/papers/aaai00.pdf
[27] J. Wnek, R.S. Michalski, Hypothesis-driven constructive induction in AQ17-HCI: a method and experiments., Mach. Learning 14 (1994) 139–168.