Logical Analysis of Data: Classification with Justification

Endre BOROS* Yves CRAMA† Peter L. HAMMER‡
Toshihide IBARAKI§ Alexander KOGAN¶
Kazuhisa MAKINO‖

January 2011

Abstract

Learning from examples is a frequently arising challenge, with a large number of algorithms proposed in the classification, data mining and machine learning literature. The evaluation of the quality of such algorithms is frequently carried out *ex post*, on an experimental basis: their performance is measured either by cross validation on benchmark data sets, or by clinical trials. Few of these approaches evaluate the learning process *ex ante*, on its own merits. In this paper, we discuss a property of rule-based classifiers which we call "justifiability", and which focuses on the type of information extracted from the given training set in order to classify new observations. We investigate some interesting mathematical properties of justifiable classifiers. In particular, we establish the existence of justifiable classifiers, and we show that several well-known learning approaches, such as decision trees or nearest neighbor based methods, automatically provide justifiable classifiers. We also identify maximal subsets of observations which must be classified in the same way by every justifiable classifiers. Finally, we illustrate by a numerical example that using classifiers based on "most justifiable" rules does not seem to lead to overfitting, even though it involves an element of optimization.

*RUTCOR, Rutgers Center for Operations Research, Piscataway, NJ 08854-8003, USA, boros@rutcor.rutgers.edu

†HEC Management School, University of Liège, Boulevard du Rectorat 7 (B31), B-4000 Liège, Belgium, Yves.Crama@ulg.ac.be

‡Our colleague and friend Peter L. Hammer passed away in a tragic car accident in December 2006, at a time when we were working on this research project.

§Department of Informatics, School of Science and Technology, Kwansei Gakuin University, 2-1 Gakuen, Sanda, Japan 669-1337, ibaraki@kwansei.ac.jp

¶Department of Accounting, Business Ethics and Information Systems, Rutgers Business School, Rutgers University, Newark, NJ 07102, and RUTCOR, Rutgers Center for Operations Research, Piscataway, NJ 08854-8003, USA, kogan@rutgers.edu

‖Graduate School of Information Science and Technology, University of Tokyo, Tokyo, 113-8656, Japan, makino@mist.i.u-tokyo.ac.jp

1 Introduction

An increasing number of machine learning tools assist daily decisions – including fully or partly automated systems used by banks (e.g., evaluation of loan worthiness, detection of credit card fraud), by communications companies (detection of illegal cellular phone use), by law enforcement authorities (criminal or terrorist profiling), or in medicine (pre-screening of patients). Most of these situations are governed by the conditions and rules of highly complex environments where, unlike in physics or chemistry, fundamental laws are rarely available to help the decision-maker in the process of reaching his conclusions. Instead, most of these systems derive their intelligence from databases of historical cases, described in terms of their most salient attributes. Sophisticated data analysis techniques and learning algorithms are used to derive diagnosis rules, or profile descriptions which are then implemented in practice.

The more these systems affect our everyday life, the more controversies and conflicts may arise: in certain cases, the consequences of potential mistakes may indeed be very expensive, or drastic in some other way (think for instance of a serious disease being diagnosed belatedly, due to a faulty screening decision). In such cases, the organization applying such automated tools might be forced to *justify* itself, and to demonstrate that it had solid, objective arguments to formulate its diagnosis. But in fact, it is usually not entirely clear what could amount to an acceptable *justification* of a classification rule, and how a classifier could be *certified* to provide justifiable classifications for each of its future applications.

In this paper, we want to argue that some minimal requirements for "justifiability" are satisfied by the classification rules introduced by Crama, Hammer and Ibaraki [15], and subsequently developed into a rich classification framework under the name of *Logical Analysis of Data*, or LAD (see for instance [7, 9, 10, 11, 12, 13, 22], etc.). We also aim at collecting some fundamental properties of LAD-type classification rules which have not appeared elsewhere, yet. Finally, we want to clarify the relation between these rules and certain popular classification rules used in the machine learning literature, such as the rules computed by nearest neighbor classification algorithms or decision trees.

The paper is organized as follows. In the remainder of this section, we rely on a small example to explain in some detail, but informally, what we mean by a "justifiable" classification rule. Section 2 recalls useful facts about partially defined Boolean functions and their extensions, and introduces the main concepts and definitions used in LAD. In particular, it introduces an interesting family of Boolean classifiers called *bi-theories*, which can be built on elementary rules called *patterns* and *co-patterns*. Our main results are presented in Section 3, together with relevant examples and interpretations, but without the proofs which are collected together in Appendix A, so as to facilitate the reading. In these sections, we establish some of the main

structural properties of patterns, co-patterns and bi-theories, and we examine their computational complexity. We also show that decision trees and nearest neighbor procedures fall under this generic LAD setting. In spite of their simplicity, we provide empirical evidence in Section 4 that the LAD rules perform and generalize well in a variety of applied situations. Section 5 mentions a number of challenging open questions. The proofs are collected in a technical appendix at the end of the paper.

1.1 An example

Let us first illustrate the basic issues and ideas on a small example. (Although this example is very small and artificial, we note that similar issues arise in many real situations where a simple scoring method is used to derive classifications.)

We assume that seven suspected cases of a rare disease have been documented in the medical literature. Three of the cases (patients A, B, and C) were "positive cases" who eventually developed the disease; the other four suspicious cases (patients T, U, V and W) turned out to be "negative", healthy cases. The following table displays the available data; each case is described by binary values indicating the presence or absence of four different symptoms.

	Symptoms			
Patients	x_1	x_2	x_3	x_4
A	1	1	0	1
B	0	1	1	1
C	1	1	1	0
T	0	0	1	1
U	1	0	0	1
V	1	0	1	0
W	0	1	1	0

Both Dr Perfect (or Dr P in short) and Dr Rush (or Dr R in short) have access to this table, and both develop their own diagnosis rules by analyzing the data set. Dr R notices that the positive cases, and only those, exhibit 3 out of the 4 symptoms; so, he derives this as a diagnosis rule, i.e., he decides to consider a patient described by the symptom vector $\mathbf{x} = (x_1, x_2, x_3, x_4)$ as a "positive case" if $x_1 + x_2 + x_3 + x_4 \geq 3$. Dr P performs a different analysis: he regards symptom x_3 as irrelevant, and he values symptom x_2 as twice more important than the other ones. Consequently, he diagnoses a patient as "positive" if $x_1 + 2x_2 + x_4 \geq 3$.

It is easy to check that both doctors have derived a "perfect" diagnosis rule in the sense that all cases in the small data base are correctly diagnosed by these rules. Hence, both doctors could feel that their classification rules are well-grounded, given the current state of knowledge.

Still, the above two diagnosis rules are not identical, and therefore they may certainly provide contradictory conclusions in some possible future cases. If we assume that no random effect and no exogenous information (e.g., additional knowledge about the properties of the classification rule, or about the interdependence of symptoms, or about other relevant attributes) are available to resolve such potential disagreements, then it is reasonable to distinguish among the rules on the basis of their endogenous justifiability only. To explain this point, imagine that a new patient, say Mrs Z, shows up with the symptom vector $\mathbf{x}_Z = (1, 0, 1, 1)$. Dr R will diagnose her as a "positive" case, thus leading Mrs Z to undergo a series of expensive, time consuming and painful tests, before she learns that she is in fact healthy. If Mrs Z later finds out that Dr P would have diagnosed correctly her condition, without going through the extra tests and difficulties, then she may want to ask Dr R to explain what lead him to his initial diagnosis. In particular, Mrs Z might insist on understanding which particular combination of her symptoms triggered Dr R's diagnosis.

Indeed, every diagnosis rule can equivalently be expressed in terms of a set of "conjunctive logical" rules, each of the form: "if certain symptoms occur and some others do not, then the patient is a positive case". In particular, Dr R's diagnosis can equivalently be modeled by the disjunction of four simple rules, namely:

$$R(\mathbf{x}) = x_1x_2x_3 \vee x_1x_2x_4 \vee x_1x_3x_4 \vee x_2x_3x_4,$$

where a rule $x_1x_2x_3$, for example, expresses that the patient is positive if $x_1x_2x_3 = 1$, i.e., if the symptoms 1, 2, and 3 are simultaneously present. Similarly, Dr P's diagnosis can be described by the disjunction of two simple rules:

$$P(\mathbf{x}) = x_1x_2 \vee x_2x_4.$$

A basic underlying assumption of this paper is that the classifiers to be considered are expressed as disjunctions of simple conjunctive rules, as illustrated by the example. (Note that every Boolean classifier can be expressed in this way.)

So, how can Dr R justify his diagnosis? The only reason why he declared Mrs Z positive is to be found in his third rule, that is, the co-occurrence of symptoms 1, 3 and 4. But in fact, there is *no supporting evidence* in the initial data to justify this rule, since the combination of symptoms 1, 3 and 4 was never observed in the data set! Note that Dr R could have foreseen this difficulty from the very beginning, even before Mrs Z showed up, and he should probably never have adopted his third rule!

A similar situation arises when an observation is classified as a "negative case" by either doctor: a set of rules hides behind every such conclusion, and these rules can be explicitly identified by "negating" the appropriate classifier R or P. To illustrate this, imagine that Mr Y shows up in Dr R's office and that he displays the symptom vector $\mathbf{x}_Y = (0, 1, 0, 1)$. Mr Y will

	Symptoms				Classification by	
Patients	x_1	x_2	x_3	x_4	Dr R	Dr P
A	1	1	0	1	1	1
B	0	1	1	1	1	1
C	1	1	1	0	1	1
T	0	0	1	1	0	0
U	1	0	0	1	0	0
V	1	0	1	0	0	0
W	0	1	1	0	0	0
Z	**1**	**0**	**1**	**1**	**1**	**0**
Y	**0**	**1**	**0**	**1**	**0**	**1**

Table 1: Classification results of Dr R's and Dr P's classifiers for the given data, as well as for two future cases, Mrs Z and Mr Y

be diagnosed by Dr R as a negative case (i.e., a healthy patient) and sent home accordingly. Later when he finds himself in an emergency room, he may learn from Dr P that he is in fact seriously ill. What did Dr R miss? We can see that Dr R based his negative diagnosis on the lack of symptoms 1 and 3. Indeed, the negation of Dr R's classifier is

$$\overline{R}(\mathbf{x}) = \overline{x}_1\overline{x}_3 \vee \overline{x}_1\overline{x}_4 \vee \overline{x}_2\overline{x}_3 \vee \overline{x}_2\overline{x}_4$$

and rule $\overline{x}_1\overline{x}_3$ is the only active rule that applies to Mr Y's case. On the other hand, the negation of Dr P's classifier is

$$\overline{P}(\mathbf{x}) = \overline{x}_1\overline{x}_4 \vee \overline{x}_2$$

and none of the corresponding two rules would have indicated Mr Y as a healthy patient. We can notice again that the rule $\overline{x}_1\overline{x}_3$ in Dr R's classifier (for negative cases) does not have any support in the given data (in the sense that none of the patients in the data set satisfies $\overline{x}_1\overline{x}_3 = 1$), while both rules of Dr P are well supported by the data.

Moreover, we can also see that for each rule selected by Dr R when declaring that a patient is either positive or negative, there is another rule selected by Dr P which is *better supported* by the initial data set, but which does not always lead to the same conclusion. For instance, Dr R's rule $x_1x_2x_3$ is only supported by the observation of patient C, while Dr P's rule x_1x_2 is supported by the cases A and C. Thus, we could even wonder whether *it is professionally justifiable* for Dr R to consider the rules he used, since he had the opportunity to realize (just like Dr P did) that all positive cases can be explained by some other rules, each of which is better supported by the given data.

The above questions are of course highly debatable in a real-world context, but in the learning framework that we investigate here, where an automated learning procedure has access only to the given data, and to no

exogenous information, Dr R's algorithmic choices do not appear to be well-justified. Let us add that our dissatisfaction with Dr R's classifier is based solely on the data set initially presented to us, and not on the subsequent cases of Mr Y and Mrs Z. We used the latter cases as illustrative examples, but in fact, our main point is elsewhere: the learning approach of Dr R is not satisfactory because his approach ended up accepting rules which are not supported at all by the data.

1.2 Justifiable rules

Let us try to generalize the previous discussion. In this paper, we want to consider classifiers which, when expressed as *disjunctions of simple conjunctive rules*, can be *justified with respect to the given data set* $\mathbb{D}$ in the sense that they satisfy the following axiom:

(A1) Each rule is supported by (at least one) observation, and is not contradicted by any observation in $\mathbb{D}$. This requirement should hold both for positive and for negative classifications.

Let us remark here that we consider Boolean classifiers ϕ which are "complete" in the sense that they classify all possible input cases (either as positive or as negative). In other words, the positive rules we use are the prime implicants of ϕ, while the negative rules are the prime implicants of $\overline{\phi}$. Thus, the "justifiability" of the positive rules according to (A1) depends only on the known positive cases, while the negative rules can be derived from the positive ones by a unique algebraic procedure (the negation of ϕ). Consequently, the known negative examples do not seem, to play any role in deriving our negative rules, even though we require in (A1) that those negative rules are also supported (by the known negative cases). It is hence not at all obvious that classifiers satisfying axiom (A1) exist. Our intention is to show that classification rules satisfying axiom (A1) do exist, display many interesting mathematical properties, are sufficiently general to encompass many well-known families of classifiers, and can be successfully applied to real-world situations.

Let us stress, if necessary, that the above objective shifts the focus for the development of learning approaches: the main objective is no longer on obtaining a high rate of correct classifications, but on being able to provide convincing justifications for each individual classification! In other words, we are interested in the *a priori justification* of the rules rather than in their *a posteriori performance*. Note that other machine learning frameworks (in particular, probabilistic models such as those discussed by Angluin [3] or Valiant [34]) also provide *a priori* measures of performance for learning algorithms; but the nature of the quality criteria in [3, 34] is radically different from those introduced in Axiom (A1), as they still concentrate on the rate of correct classifications achieved by the rules, and not on the justification derived from past observations.

Of course, when keeping posterior performance in sight, rules with a large support still appear to be quite appealing. One might expect such rules to lead to overfitting, but we shall actually provide computational evidence that overfitting does not generally seem to take place when we generate classifiers (i.e., collections of rules) satisfying the axiom:

(A2) No rule can be substituted by another justified rule which has a larger support within $\mathbb{D}$.

In order to proceed with this discussion, we need to specify more precisely all the relevant notions that we have so far informally described. This is the topic of the next sections, where we present a framework for the construction of Boolean classifiers expressed as disjunctions of elementary conjunctions, and where we give an overview of our main results.

2 Notations, definitions, and main results

In this section we introduce the necessary terminology about Boolean and partially defined Boolean functions, recall some of their basic properties, and close the section by stating our main results. In the subsequent sections we present detailed proofs and results of computational experiments.

2.1 Partially defined Boolean functions

We start with a few definitions and notations relative to Boolean functions (see e.g. Crama and Hammer [14] or Muroga [29]).

Let n be a positive integer and let $\mathbb{V} = \{1, 2, \ldots, n\}$. A *Boolean function of n variables* is a mapping $\mathbb{B}^n \mapsto \mathbb{B}$, where $\mathbb{B}$ is the set $\{0, 1\}$ and $\mathbb{B}^n$ denotes the n-fold cartesian product of $\mathbb{B}$ with itself. If S is any set with cardinality n, we also write $\mathbb{B}^S$ for $\mathbb{B}^n$. A vector $x^* \in \mathbb{B}^n$ is a *true vector* (resp. *false vector*) of the Boolean function f if $f(x^*) = 1$ (resp. $f(x^*) = 0$). We denote by $T(f)$ (resp. $F(f)$) the set of *true vectors* (resp. *false vectors*) of f. Clearly, any partition $T \cap F = \emptyset$, $T \cup F = \mathbb{B}^n$ uniquely defines a Boolean function $f_{T,F}$ such that $T = T(f_{T,F})$ and $F = F(f_{T,F})$, and thus there are 2^{2^n} Boolean functions of n variables. The *negation* (or *complement*) of a function f is the function $\overline{f}$ defined by $T(\overline{f}) = F(f)$ (and $F(\overline{f}) = T(f)$).

A *partially defined Boolean function* (abbreviated as "pdBf") on $\mathbb{B}^n$ is defined as a pair of sets (T, F) such that $T, F \subseteq \mathbb{B}^n$ and $T \cap F = \emptyset$. We refer to the set T as the set of true vectors (sometimes called *positive examples*) and to F as the set of false vectors (or *negative examples*) of the pdBf (T, F). As illustrated by the examples in subsequent sections, (binary) data sets arising in classification problems can be viewed as pdBfs. (Let us remark that the condition $T \cap F = \emptyset$ may not be satisfied in certain real-world data sets for classification problems, and that many of our claims and algorithms can be modified to accommodate such practical cases.)

In this framework, classifiers correspond to *extensions* of pdBfs, where a Boolean function f is called an *extension* of the pdBf (T, F) if

$$T(f) \supseteq T \quad \text{and} \quad F(f) \supseteq F. \tag{1}$$

When the Boolean function f is an extension of (T, F), we shall also say that f *correctly classifies* all the vectors $a \in T$ and $b \in F$.

We denote by $\mathcal{E}(T, F)$ the family consisting of all extensions of (T, F). Since $T \cap F = \emptyset$, it is clear that $\mathcal{E}(T, F) \neq \emptyset$; more precisely,

$$|\mathcal{E}(T, F)| = 2^{2^n - |T| - |F|} > 0.$$

Given a pdBf (T, F), we associate with it two special Boolean functions, respectively called its minimum and its maximum extension, and denoted by $f_{\min}$ and $f_{\max}$, which we define as follows:

$$\begin{aligned} T(f_{\min}) &= T, & F(f_{\min}) &= \mathbb{B}^n - T \\ T(f_{\max}) &= \mathbb{B}^n - F, & F(f_{\max}) &= F. \end{aligned} \tag{2}$$

For two Boolean functions g and h, let us say that the relation $g \leq h$ holds if and only if $T(g) \subseteq T(h)$, or equivalently, if and only if $F(g) \supseteq F(h)$. The following properties are obvious.

Claim 2.1. *For every pdBf (T, F), we have*

$$\mathcal{E}(T, F) = \{f \mid f_{\min} \leq f \leq f_{\max}\}.$$

Moreover, given any two Boolean functions g, h on $\mathbb{B}^n$ satisfying $g \leq h$, there exists a unique pdBf (T, F) such that

$$\mathcal{E}(T, F) = \{f \mid g \leq f \leq h\}.$$

The set of extensions of the pdBf (F, T) is

$$\mathcal{E}(F, T) = \{\overline{f} \mid f \in \mathcal{E}(T, F)\}.$$

Furthermore, the minimum and maximum extensions of (F, T) are $\overline{f}_{max}$ and $\overline{f}_{min}$. □

We call the pdBf (F, T) the *negation of* (T, F), and we say that the functions $f \in \mathcal{E}(F, T)$ are the *co-extensions* of (T, F). Clearly, the extension $f \in \mathcal{E}(T, F)$ and the co-extension $\overline{f} \in \mathcal{E}(F, T)$ provide the same information about the pdBf (T, F). However, different algebraic representations of f and $\overline{f}$ may have very different sizes, and hence obtaining one or the other of these representations may not be computationally equivalent. For this reason, we shall aim in the sequel at finding both concise extensions and co-extensions for a given pdBf.

2.2 Terms, patterns, DNF representations and decision trees

A *term* is a Boolean function t whose true set $T(t)$ is of the form

$$T(t) = \{x \in \mathbb{B}^n \mid x_i = 1 \text{ for all } i \in A \text{ and } x_j = 0 \text{ for all } j \in B\} \tag{3}$$

for some sets $A, B \subseteq \{1, 2, \ldots, n\}$. It can be represented by an *elementary conjunction*, that is, by a Boolean expression of the form

$$t(x) = \left(\bigwedge_{i \in A} x_i\right) \wedge \left(\bigwedge_{j \in B} \overline{x}_j\right). \tag{4}$$

Geometrically, the true set (3) of a term t is a *subcube*, or a *face* of the Boolean hypercube. It can equivalently be viewed as an *interval* of the form

$$[a, b] = \{x \in \mathbb{B}^n \mid x_j \in \{a_j, b_j\} \text{ for } j = 1, 2, \ldots, n\},$$

where $a, b \in \mathbb{B}^n$, $a_j = 1$ if and only if $j \in A$, and $b_j = 0$ if and only if $j \in B$. Let us add that we can view binary vectors also as *points* in the hypercube, and we will use the terms "vector" and "point" interchangeably.

For a term t and a point $a \in \mathbb{B}^n$, we say that t (or $T(t)$) *covers* a if $t(a) = 1$, i.e., if $a \in T(t)$. We denote by t_a the (unique) term which covers $a \in \mathbb{B}^n$ and no other point; i.e. for which $T(t_a) = \{a\}$. It is easy to see that

$$t_a(x) = \left(\bigwedge_{i:a_i=1} x_i\right) \wedge \left(\bigwedge_{i:a_i=0} \overline{x}_i\right). \tag{5}$$

We call t_a the *minterm* of a.

Every Boolean function can be represented by a *disjunctive normal form* (DNF), i.e., by a disjunction of terms (elementary conjunctions).

Let us observe that for every pdBf (T, F), a DNF of the minimal extension f_{min} can be determined efficiently. Namely, the DNF

$$\varphi(x) = \bigvee_{a \in T} t_a(x) \tag{6}$$

is clearly a DNF representation of f_{min}, where $t_a(x)$ denotes the minterm of a as in (5).

It is somewhat less trivial to find a short DNF representation for f_{max}. But it can be shown that f_{max} has a DNF representation involving no more than $\frac{1}{2}n|F|$ terms, and such a representation can be found in polynomial time (see e.g., [25, 26, 27]).

From their very definition (3), it is clear that Boolean terms correspond to certain combination of attribute values. When analyzing a pdBf (T, F), we can often view such combinations as "rules" which are more specifically associated with one of the classes T or F. Thus, a term (or rule) t classifies a point $a \in \mathbb{B}^n$ as a positive observation if $t(a) = 1$. Intuitively, we can

consider the term (or rule) t to be "justified" by the data set (T, F), if $t(a) = 1$ holds for *some* vectors of T (the more the better), and $t(b) = 0$ for *all* vectors $b \in F$.

In order to turn this idea into a mathematically useful notion, we follow here the presentation of Crama, Hammer and Ibaraki [15] and we call a term t a *pattern* of the pdBf (T, F) if

$$|T \cap T(t)| > 0 \quad \text{and} \quad |F \cap T(t)| = 0. \tag{7}$$

Thus, geometrically speaking, a pattern is a subcube which covers at least one point of T and no point of F.

Patterns can be considered as simple rules providing evidence that a vector is a positive observation. For a pattern t of (T, F), we can also say that the set of vectors $T \cap T(t)$ *justifies* t, in the sense that this set of vectors provides a justification for any possible future conclusions we might draw from t. Of course, the larger the number of vectors in (T, F) justifying a pattern t, the higher our confidence may be in the classification based on t.

An *implicant* of a Boolean function f is a term t such that $t \leq f$. Note that those terms such that $t(b) = 0$ holds for all $b \in F$ are the implicants of the unique largest extension $f_{max} \in \mathcal{E}(T, F)$ defined by (2). Thus, the patterns of (T, F) are those implicants of f_{max} which cover some vectors of T.

Example 2.2. Let us consider the pdBf (T, F) given in Table 2. For this

Table 2: An example of pdBf (T, F).

		x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
	$a^{(1)} =$	0	1	0	1	0	1	1	0
T	$a^{(2)} =$	1	1	0	1	1	0	0	1
	$a^{(3)} =$	0	1	1	0	1	0	0	1
	$b^{(1)} =$	1	0	1	0	1	0	1	0
F	$b^{(2)} =$	0	0	0	1	1	1	0	0
	$b^{(3)} =$	1	1	0	1	0	1	0	1
	$b^{(4)} =$	0	0	1	0	1	0	1	0

pdBf, the corresponding extension f_{max} has several implicants, including $\overline{x}_1 x_2$, $\overline{x}_6 \overline{x}_7 x_8$, $x_7 x_8$, etc. It is easy to see that

- $\overline{x}_1 x_2$ is a pattern that covers $a^{(1)}$ and $a^{(2)}$, and
- $\overline{x}_6 \overline{x}_7$ is a pattern that covers $a^{(2)}$ and $a^{(3)}$, while
- $x_7 x_8$ is not a pattern, since it does not cover any vector of T, and
- $\overline{x}_6 \overline{x}_7 x_8$ is a pattern that covers $a^{(2)}$ and $a^{(3)}$, but it is "dominated" by the shorter term $\overline{x}_6 \overline{x}_7$ which is also a pattern, as we observed above.

□

Interchanging the roles of T and F in a given pdBf (T, F), we can analogously derive simple rules for indicating if a given vector is a negative observation. Let us observe that for the pdBf (F, T), the minimum and maximum extensions defined by (2) are $\overline{f}_{min}$ and $\overline{f}_{max}$, respectively. Accordingly, let us call *co-pattern* of (T, F) any implicant t of $\overline{f}_{min}$ which covers at least one negative example $b \in F$. In other words, a term t is a co-pattern of (T, F) if

$$|T \cap T(t)| = 0 \quad \text{and} \quad |F \cap T(t)| > 0. \tag{8}$$

Example 2.3. For the pdBf (T, F) of Table 2, the term $x_5\overline{x}_8$ is a co-pattern that covers $b^{(1)}, b^{(2)}$ and $b^{(4)}$, and $\overline{x}_5 x_8$ is a co-pattern that covers $b^{(3)}$. □

Note that the notions of "interesting rules" and "patterns" generated from a given data set (T, F) are very closely related to concepts which have been (re)discovered and applied by other researchers in various frameworks, e.g., the concepts of *association rules* (see [2]) and of *jumping emerging patterns* (see [16]) which have been more recently introduced in the data mining literature (see also [19, 20, 33]) for related ideas). In particular, jumping emerging patterns are exactly identical to patterns and co-patterns.

For a pdBf (T, F) let us denote by $P(T, F)$ the set of its patterns, and by $coP(T, F)$ the set of its co-patterns. The following properties are obviously implied by the definitions.

Claim 2.4. *For arbitrary subsets $T, F \subseteq \mathbb{B}^n$ we have*

(i) $P(T, F) = \emptyset$ *if and only if* $T \subseteq F$,
(ii) $coP(T, F) = P(F, T)$,
(iii) $P(T, F') \subseteq P(T, F)$ *whenever* $F \subseteq F'$,
(iv) $P(T, F) \subseteq P(T', F)$ *whenever* $T \subseteq T'$,
(v) $P(T, F) = P(T \setminus F, F)$. □

Observe that when we use a DNF φ representing an extension of the pdBf (T, F) to classify an as-yet unclassified vector $x \in \mathbb{B}^n$, and when we classify x as a positive example, then we actually derive our conclusion from the existence of a term t of the DNF φ such that $t(x) = 1$. Since patterns are special terms, which are supported by "evidence" collected from the given training data (T, F), it is a natural idea to consider special extensions of a pdBf (T, F) which can be built from patterns of (T, F).

So, following Crama, Hammer and Ibaraki [15], let us call an extension $f \in \mathcal{E}(T, F)$ a *theory* of the pdBf (T, F) if it can be represented by a disjunction of patterns of (T, F). Thus, a theory is a disjunction of patterns

which together cover all the examples in T. (When no confusion arises, we sometimes call "theory" the DNF itself, rather than the function it represents.) We note that special classes of theories, e.g., "prime theories" and "irredundant theories", have also been introduced in [15], but we will not explicitly refer to them in this paper.

Example 2.5. Consider again the pdBf in Table 2. It is easy to see that

$$a^{(1)} \text{ is covered by the pattern } \overline{x}_1 x_2,$$
$$a^{(2)} \text{ is covered by the pattern } x_2 x_5,$$
$$a^{(3)} \text{ is covered by the pattern } x_3 x_8,$$

and thus the DNF

$$\varphi = \overline{x}_1 x_2 \vee x_2 x_5 \vee x_3 x_8$$

defines a theory of (T, F). Another theory of (T, F) is obtained by removing $x_3 x_8$ from φ, since the resulting DNF φ' still covers all vectors in T:

$$\varphi' = \overline{x}_1 x_2 \vee x_2 x_5.$$

□

Let us denote by $\mathcal{E}_{\mathcal{T}}(T, F)$ the set of theories for a given pdBf (T, F). Clearly, we have $\mathcal{E}_{\mathcal{T}}(T, F) \subseteq \mathcal{E}(T, F)$, in general. Furthermore, in most cases only a very small number of all extensions are theories.

Example 2.6. Consider the pdBf on $\mathbb{B}^n$ defined by $T = \{(11\ldots1)\}$ and $F = \{(00\ldots0)\}$. It has $h(n) = 2^{2^n-2}$ extensions. Its patterns are all the terms built on uncomplemented variables only. Thus, the theories of (T, F) are exactly the non-constant monotone Boolean functions on $\mathbb{B}^n$, and the number of theories of (T, F) is equal to $d(n)-2$, where $d(n)$ is the number of monotone Boolean functions on n variables. It is well known that $\log_2 d(n)$ is asymptotic to the middle binomial coefficient $\binom{n}{\lfloor n/2 \rfloor}$ [24], which implies that $d(n)$ is much smaller than $h(n)$. □

By interchanging the roles of T and F, we can analogously define *co-theories*. Thus, if φ is a co-theory of (T, F) and $\varphi(x) = 1$, then φ recommends to classify x as a "negative observation".

The terms "theory" and "co-theory" are sometimes used slightly differently in the literature. In learning theory and data-mining, "theory" is often used as a synonym of "extension". We prefer to distinguish these two notions, and to reserve the name "theory" only for the special type of extensions defined above. Also, a "co-theory" is sometimes referred to as a "negative theory" (in which case, a "theory" is called a "positive theory") to emphasize its role with respect to the set of negative examples F.

Thus, when we use a theory φ for classification purposes, a vector x such that $\varphi(x) = 1$ is classified as a positive observation based on the evidence provided by the patterns that appear in φ. On the other hand, if $\varphi(x) = 0$, then the only rationale for classifying x as a negative observation would in fact be based on the "lack" of evidence supporting the opposite conclusion. From this point of view, it would be much more convincing to use a theory f *only if* its negation $\overline{f}$ simultaneously happens to be a co-theory. In this case the terms of $\overline{f}$ being co-patterns would provide justifying evidence for negative classifications, as well. In fact, adhering to our axiom (A1) requires us to restrict our attention to such special theories for classification purposes.

In view of this, let us say that a function f is a *bi-theory* of (T, F) if f is a theory and $\overline{f}$ is a co-theory of (T, F). We denote by $\mathcal{E}_B(T, F)$ the set of bi-theories of (T, F):

$$\mathcal{E}_B(T, F) = \{f \in \mathcal{E}_{\mathcal{T}}(T, F) \mid \overline{f} \in \mathcal{E}_{\mathcal{T}}(F, T)\}. \tag{9}$$

Example 2.7. Consider the pdBf (T, F) in three variables defined by

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

It can be checked easily by complete enumeration that

$$P(T, F) = \{x_1,\ x_1x_2,\ x_1\overline{x}_2,\ x_1x_3,\ x_1\overline{x}_3,\ x_1x_2x_3,\ x_1\overline{x}_2\overline{x}_3\}, \quad \text{and}$$
$$coP(T, F) = \{\overline{x}_1,\ \overline{x}_1x_2,\ \overline{x}_1\overline{x}_2,\ \overline{x}_1x_3,\ \overline{x}_1\overline{x}_3,\ \overline{x}_1x_2x_3,\ \overline{x}_1\overline{x}_2x_3,\ \overline{x}_2x_3,\ \overline{x}_1\overline{x}_2\overline{x}_3\}.$$

Thus, we see that $f = x_1$ is a bi-theory for (T, F), since its complement $\overline{f} = \overline{x}_1$ is a co-theory. There is another bi-theory $g = x_1x_2 \vee x_1\overline{x}_3$, since $\overline{g} = \overline{x}_1 \vee \overline{x}_2x_3$. It can be shown that there are no other bi-theories for this pdBf. □

Let us add that instead of elementary conjunctions (terms) and DNF representations we could as well consider elementary disjunctions (clauses) and CNF representations of Boolean functions. All the concepts and properties introduced so far have their natural counterparts for clauses and CNF representations.

In subsequent sections, we will also consider an additional type of representations of Boolean and partially defined Boolean functions, namely representations based on decision trees.

A *decision tree* is a rooted directed tree D on the vertex set $N \cup L$, where the *leaf vertices* in L have out-degree zero, the vertices in N have exactly two outgoing arcs (left and right), and the root $r \in N$ has in-degree zero while all other vertices have exactly one incoming arc. Each vertex $v \in N$ is labelled by an index $j(v) \in \{1, ..., n\}$ and the leaf vertices $v \in L$ are labelled by either 0 or 1. We denote by L_0 and L_1 the sets of leaves labelled respectively by 0 and 1.

Given a binary vector $x \in \mathbb{B}^n$, we can use the decision tree D to classify x into one of its leaves: Starting from the root $v = r$ of D, we move from vertex to vertex, always following the left arc out of v if $x_{j(v)} = 0$, and the right arc otherwise. We stop when we arrive at a leaf $u \in L$. Denoting by $B_u \subseteq \mathbb{B}^n$ the set of binary vectors classified by D into leaf u, we have $B_u \cap B_v = \emptyset$ if $u \neq v$, and $\bigcup_{u \in L} B_u = \mathbb{B}^n$. Defining $T = \bigcup_{u \in L_1} B_u$ and $F = \bigcup_{u \in L_0} B_u$, we get a partition of $\mathbb{B}^n$ defining a unique Boolean function f_D with $T(f_D) = T$. Conversely, it is well known that every Boolean function can be represented by some decision tree in this way (typically there are many decision trees representing the same function).

Given a pdBf (T, F), we say that a decision tree D defines an extension of (T, F) (or simply that D is a decision tree for (T, F)) if f_D is an extension of (T, F), that is, if $T(f_D) \supseteq T$ and $T(f_D) \cap F = \emptyset$. Finally, we say that a decision tree D is *reasonable* for (T, F) if

(i) D defines an extension of (T, F),

(ii) for every leaf $u \in L$, $B_u \cap (T \cup F) \neq \emptyset$ (for every leaf u of D, at least one example of (T, F) is classified into u), and

(iii) for every nonterminal vertex $v \in N$, at least one vector $a \in T$ is classified into a descendant of v, and at least one vector $b \in F$ is classified into another descendant of v.

Finally, we denote by $\mathcal{D}(T, F)$ the collection of all reasonable decision trees for (T, F), and we denote by $\mathcal{E}_D(T, F)$ all those extensions of (T, F) which can be represented by a tree in $\mathcal{D}(T, F)$. There are numerous learning algorithms which construct reasonable binary decision trees for a given pdBf (T, F), see e.g., [1, 5, 28, 30, 32] and Section A.3.

3 Main results

Bi-theories can be viewed as those extensions of pdBfs which satisfy our axiom (A1). They are, therefore, our main object of study. The purpose of this section is to describe the main properties of bi-theories and of their building blocks, that is, patterns and co-patterns. Proofs and more detailed statements will be provided in Sections A.1–4.

3.1 Patterns and co-patterns

Let us first note that if $T \neq T'$, then $P(T, F) \neq P(T', F)$, since e.g., the minterm t_a (as introduced in Equation (5)) corresponding to a vector $a \in T \setminus T'$ is a pattern of (T, F), while it is not a pattern of (T', F). However, $P(T, F)$ may not change when we replace the set of negative examples F by another set F'. In fact the following precise claim can be made:

Theorem 3.1. *For every pdBf (T, F), there are unique sets $F^-, F^+ \subseteq \mathbb{B}^n$ such that*

$$P(T, F') = P(T, F) \quad \text{if and only if} \quad F^- \subseteq F' \subseteq F^+,$$

and there are unique sets $T^-, T^+ \subseteq \mathbb{B}^n$ such that

$$coP(T', F) = coP(T, F) \quad \text{if and only if} \quad T^- \subseteq T' \subseteq T^+.$$

In the sequel, we often look at $(\cdot)^+$ and $(\cdot)^-$ as operators acting on sets. These notations are somewhat ambiguous, since the definitions of S^+ or S^- depend on whether the set S is viewed as a set of positive or negative examples, as well as on the second member or the pdBf. But this should not create any confusion in the sequel.

The intrinsic meaning of the sets F^+ and T^+ is clarified by the following result.

Theorem 3.2. *For a pdBf (T, F), F^+ is exactly the set of vectors of $\mathbb{B}^n \setminus T$ which are not covered by any pattern of (T, F), and T^+ is the set of vectors of $\mathbb{B}^n \setminus F$ which are not covered by any co-pattern of (T, F).*

In view of the previous statement, a vector belonging to F^+ should always be classified as a negative observation by *every* classification rule based on the patterns of (T, F): indeed, Theorem 3.2 implies that *no evidence* can be derived from (T, F) to support the conclusion that a vector $x \in F^+$ is a positive observation. Similarly, a vector in T^+ should always be considered to be a positive observation by *every* classification rule based on the co-patterns of (T, F). More formally, we can state:

Corollary 3.3. *Let (T, F) be a pdBf.*
(a) *If f is a theory of (T, F), then $F^+ \subseteq F(f)$, i.e., $f(u) = 0$ for all $u \in F^+$.*
(b) *If g is a co-theory of (T, F), then $T^+ \subseteq F(g)$, i.e., $g(v) = 0$ for all $v \in T^+$.*
(c) *If f is a bi-theory of (T, F), then $F^+ \subseteq F(f)$ and $T^+ \subseteq T(f)$, i.e., $f(u) = 0$ for all $u \in F^+$ and $f(v) = 1$ for all $v \in T^+$.*

We shall return to the interpretation of the sets F^+, F^-, T^+ and T^- in Section 3.2. For the time being, we want to provide a more constructive characterization for these sets, which will allow us to draw some algorithmic consequences as well.

Theorem 3.4. *For a pdBf* (T, F)*,*

$$F^+ = \{x \in \mathbb{B}^n \mid [x, a] \cap F \neq \emptyset \text{ for all } a \in T\}, \tag{10}$$

$$F^- = \{b \in F \mid \exists\ a \in T \text{ such that } [a, b] \cap (F \setminus \{b\}) = \emptyset\}, \tag{11}$$

$$T^+ = \{x \in \mathbb{B}^n \mid [x, b] \cap T \neq \emptyset \ \ \textit{for all} \ \ b \in F\}, \tag{12}$$

$$T^- = \{a \in T \mid \exists\ b \in F \ \ \textit{such that}\ [a, b] \cap (T \setminus \{a\}) = \emptyset\}. \tag{13}$$

Let us call a pair of vectors $a \in T$ and $b \in F$ *closest*, if $[a, b] \cap (T \cup F) = \{a, b\}$, i.e., if their spanned cube does not include any other vectors from T and F. The above result then implies that F^- and T^- are exactly the vectors from F and T, respectively, which participate in such closest pairs. We can view them as the frontiers defining the difference between T and F. It is an easy consequence of the above characterizations (in fact, from Theorem 3.1) that starting from the pdBf (T^-, F^-) we can recover the same extremal sets F^+ and T^+. So in a sense T^- and F^- are the minimal sets from which we can get the same conclusions. More formally,

Corollary 3.5. *For every pdBf* (T, F)*, we have*

$$(F^+)^+ = (F^-)^+ = F^+ \quad \textit{and} \quad (F^+)^- = (F^-)^- = F^-$$

where the operators $(\cdot)^+$*,* $(\cdot)^-$ *are defined with respect to the set of positive examples* T*, and*

$$(T^+)^+ = (T^-)^+ = T^+ \quad \textit{and} \quad (T^+)^- = (T^-)^- = T^-,$$

where the operators $(\cdot)^+$*,* $(\cdot)^-$ *are defined with respect to the set of negative examples* F*.*

The sets T^+ and F^+ can both be exponentially large (even simultaneously) in terms of the input sizes n, $|T|$ and $|F|$, as shown by the following Example 3.6.

Example 3.6. Consider any pdBf (T, F) defined by $T = \{(00...0)\}$ and $F \subseteq \{x \in \mathbb{B}^n \mid x_1 = 1\}$, with $(10...0) \in F$. Then we have

$$F^- = \{(10...0)\} \quad \text{and} \quad F^+ = \{x \in \mathbb{B}^n \mid x_1 = 1\},$$

that is $|F^-| = 1$ and $|F^+| = 2^{n-1}$, independently of the size of F. □

In spite of this, membership in both sets T^+ and F^+ can be tested in polynomial time, simply by checking the conditions of the definitions in (10) and (12).

Corollary 3.7. *Given a pdBf (T, F) and a vector $x \in \mathbb{B}^n$, the membership queries $x \in T^+$ and $x \in F^+$ can both be tested in $O(n|T||F|)$ time.*

Let us also add that while the sets T^- and F^- can easily be generated from (T, F) in view of their characterizations (11) and (13), the complexity of generating T^+ and F^+ is much less obvious. Since these sets are potentially very large, we need to understand the complexity of their sequential generation. In particular in light of Corollary 3.5, it would be interesting to determine the complexity of deciding whether $T^+ \setminus T$ is empty or not. As far as we know this problem is open.

3.2 Maximal theories

Given a pdBf (T, F), let us associate with it a special theory and a special co-theory, namely the disjunctions of all its patterns and co-patterns:

$$A_{(T,F)} = \bigvee_{t \in P(T,F)} t \qquad \text{and} \qquad B_{(T,F)} = \bigvee_{t \in coP(T,F)} t. \tag{14}$$

The DNF $A_{(T,F)}$ (resp., $B_{(T,F)}$) is the largest theory (resp., co-theory) of the pdBf (T, F). Let us also note that by (ii) of Claim 2.4, we have

$$A_{(T,F)} = B_{(F,T)} \quad \text{and} \quad B_{(T,F)} = A_{(F,T)}.$$

As an important property of $A_{(T,F)}$ and $B_{(T,F)}$, we can show that every point in $\mathbb{B}^n$ is classified by at least one of these two theories. Moreover, the sets of false points of $A_{(T,F)}$ and $B_{(T,F)}$ coincide respectively with the sets F^+ and T^+, as defined in Theorem 3.1 (compare also with the statement of Theorem 3.3).

Theorem 3.8. *For every pdBf (T, F), we have*

$$T(A_{(T,F)}) \cup T(B_{(T,F)}) = \mathbb{B}^n, \tag{15}$$

meaning that any vector in $\mathbb{B}^n$ is a true vector of either $A_{(T,F)}$ or $B_{(T,F)}$, and

$$F(A_{(T,F)}) = F^+, \quad F(B_{(T,F)}) = T^+, \quad T^+ \cap F^+ = \emptyset. \tag{16}$$

As a consequence of Theorem 3.8 and of Corollary 3.7, the value of $A_{(T,F)}$ and of $B_{(T,F)}$ can be computed in polynomial time for every vector $x \in \mathbb{B}^n$. Of course, it may happen that $T(A_{(T,F)}) \cap T(B_{(T,F)}) \neq \emptyset$, as illustrated by Example 3.9 below. In this case, the classifications derived from $A_{(T,F)}$ and $B_{(T,F)}$, respectively, may not always be compatible.

Example 3.9. Let us return to the small pdBf in Example 2.7. From the list of its patterns and co-patterns we can see that

$$A_{(T,F)} = x_1, \quad \text{and} \quad B_{(T,F)} = \overline{x}_1 \vee \overline{x}_2 x_3.$$

It is easy to check that we indeed have

$$F(A_{(T,F)}) = F^+ = \{(000), (001), (011), (010)\}, \text{ and}$$
$$F(B_{(T,F)}) = T^+ = \{(100), (111), (110)\},$$

as shown in Figure 1. Note that the remaining vector (101) belongs to both $T(A_{(T,F)})$ and $T(B_{(T,F)})$. Hence, it is classified as positive example by $A_{(T,F)}$ and as negative example by $B_{(T,F)}$. □

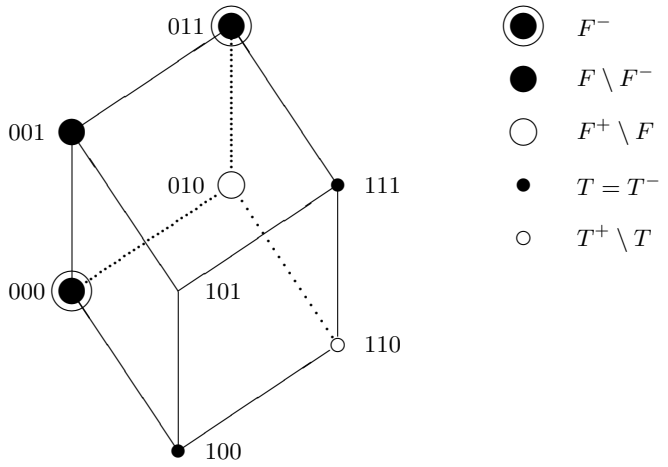

Figure 1: The 3-dimensional pdBf of Example 2.7.

Let us call the pdBF (T^+, F^+) the *closure* of (T, F). A pdBf and its closure are mathematically closely related, as evidenced by the next statement.

Theorem 3.10. *For a pdBf (T, F) and its closure (T^+, F^+), the following claims hold:*

(i) *Every pattern (resp., co-pattern) of (T, F) is a pattern (resp., co-pattern) of (T^+, F^+).*

(ii) *Every pattern (resp., co-pattern) of (T^+, F^+) is an implicant of $A_{(T,F)}$ (resp., $B_{(T,F)}$).*

(iii) *$A_{(T,F)} = A_{(T^+,F^+)}$ and $B_{(T,F)} = B_{(T^+,F^+)}$.*

Let us stress that the equalities in (iii) hold for the Boolean functions defined by the patterns and co-patterns. The sets of patterns and co-patterns themselves are not identical, e.g., we may have $P(T^+, F^+) \supsetneqq P(T, F)$. What the above claim in fact implies is that every pattern in $P(T^+, F^+) \setminus P(T, F)$ is a logical consequence of some patterns in $P(T, F)$.

We can show an even stronger relation between these pdBfs. For a subset $S \subseteq \{1, ..., n\}$ and a vector $a \in \mathbb{B}^n$ let us denote by $a[S] = (a_i \mid i \in S)$ the projection of a on $\mathbb{B}^S$, and for a set $X \subseteq \mathbb{B}^n$ let $X[S] = \{x[S] \mid x \in X\}$.

Simplicity is one of the guiding principles of learning approaches. In this spirit many learning algorithms start with the elimination of unnecessary variables. Following [15], let us call a subset $S \subseteq \{1, ..., n\}$ a *support set* of a given pdBf (T, F), if $T[S] \cap F[S] = \emptyset$, and S is minimal with respect to this property. (Thus, the values of the variables in a support set S are minimally sufficient to distinguish positive examples from negative examples in (T, F).) Then we can extend Theorem 3.10 by the following property:

Theorem 3.11.

(iv) *The pdBfs* (T^-, F^-), (T, F) *and* (T^+, F^+) *have the same support sets.*

We will show in the next two subsections that every pdBf has bi-theory extensions; as a matter of fact, we will show that some of the best-known classical learning methods automatically produce bi-theories. Before turning to those classical methods, let us note that whenever $F^+ = F$, the maximal theory $A_{(T,F)}$ is automatically a bi-theory, and whenever $T^+ = T$, then $\overline{B}_{(T,F)}$ is a bi-theory. One may think that these theories are always bi-theories. However this is not the case, as shown by the next example.

Example 3.12. Let us consider 4 vectors in the 10-dimensional Boolean space, namely

$$a_1 = (0011110000), \quad a_2 = (0000011100),$$
$$b_1 = (0111110000), \quad b_2 = (0000111110),$$

and let us consider the pdBf (T, F) defined by $T = \{a_1, a_2\}$ and $F = \{b_1, b_2\}$. Let us first observe that $x = (1111111111) \in F^+$ since $b_1 \in [x, a_1]$ and $b_2 \in [x, a_2]$. Let us next consider the vectors $y_1 = (1111111000)$ and $y_2 = (0001111111)$. Since $[y_1, a_2] \cap F = [y_2, a_1] \cap F = \emptyset$, we see that $y_1 \notin F^+$ and $y_2 \notin F^+$. Moreover, $y_1 \in [x, b_1]$ and $y_2 \in [x, b_2]$; hence, any co-pattern of (T, F) covering x must contain either y_1 or y_2, and such a co-pattern cannot be a subset of F^+.

Now, by Theorem 3.8, $T(\overline{A}_{(T,F)}) = F(A_{(T,F)}) = F^+$. But since no co-pattern of (T, F) is a subset of F^+, it follows that $T(\overline{A}_{(T,F)})$ cannot be covered (exactly) by co-patterns. Thus, $\overline{A}_{(T,F)}$ is not a co-theory, and hence $A_{(T,F)}$ is not a bi-theory. □

3.3 Decision trees and bi-theories

Decision trees provide the simplest proof that every pdBf has bi-theory extensions. Indeed, we can prove:

Theorem 3.13. *The function f_D associated with a reasonable decision tree $D \in \mathcal{D}(T,F)$ is a bi-theory of (T,F), i.e.,*

$$\mathcal{E}_D(T,F) \subseteq \mathcal{E}_B(T,F).$$

(This result was already observed by Ehrenfeucht and Haussler [18].) We note that $\mathcal{E}_D(T,F) \neq \emptyset$, since many of the classical decision tree building methods yield reasonable trees, see e.g., Quinlan [32].

Let us also remark that despite the strong relations existing between bi-theories and decision trees, $\mathcal{E}_B(T,F)$ typically properly contains $\mathcal{E}_D(T,F)$, and does not coincide with it.

Example 3.14. Let us consider the pdfBf (T,F) given by

$$T = \{(1100),(0011)\},$$
$$F = \{(1010),(0101),(0000)\},$$

and consider the function

$$f = x_1x_2 \vee x_3x_4.$$

It is easy to see that the terms of this DNF are patterns of (T,F), and in fact f is a bi-theory, for which

$$\overline{f} = \overline{x}_1\overline{x}_3 \vee \overline{x}_1\overline{x}_4 \vee \overline{x}_2\overline{x}_3 \vee \overline{x}_2\overline{x}_4$$

is a DNF representation consisting of co-patterns of (T,F). It is also easy to verify that both of these DNF-s are shortest, that is every DNF representation of f and $\overline{f}$ must contain together at least 6 terms.

This implies that if a decision tree represents f (and $\overline{f}$) it must contain at least 6 leaves. But since $|T \cup F| = 5$, all decision trees in $\mathcal{D}(T,F)$ contain at most 5 leaves, from which $f \in \mathcal{E}_B(T,F) \setminus \mathcal{E}_D(T,F)$ follows. □

The strong relation between bi-theories and reasonable decision trees is further demonstrated by the following characterization of closure sets:

Theorem 3.15. *Let (T,F) be a pdBf and let $u,v \in \mathbb{B}^n$. The following statements are equivalent:*
(a) *$u \in F^+$ and $v \in T^+$;*
(b) *$f(u) = 0$ and $f(v) = 1$ for all reasonable decision tree extensions $f \in \mathcal{E}_D(T,F)$;*
(c) *$f(u) = 0$ and $f(v) = 1$ for all bi-theories $f \in \mathcal{E}_B(T,F)$.*

In words, T^+ (resp., F^+) contains exactly those points which are classified as positive (resp., negative) observations by every reasonable decision tree and by every bi-theory. Note that Theorem 3.15 completes and strengthens Corollary 3.3 (c).

3.4 Nearest neighbor methods and bi-theories

Let us consider finally nearest neighbor type classifications.

We say that $\rho : \mathbb{B}^n \times \mathbb{B}^n \longrightarrow \mathbb{R}_+$ is a *subcube monotone similarity measure* if the following properties hold for all vectors $a, b, v \in \mathbb{B}^n$:

$$\rho(a,b) = \rho(b,a), \tag{17}$$

$$\rho(a,b) = 0 \iff a = b, \tag{18}$$

$$\rho(a,v) \leq \rho(b,v) \implies \rho(a,u) \leq \rho(b,u) \text{ for all } u \in [a,v]. \tag{19}$$

Conditions (17) and (18) are classical, and the interpretation of (19) is rather simple: if v is "closer" to a than to b according to the similarity measure ρ, then the same must hold for all vectors u in the interval between a and v. For instance, most metrics, including weighted Hamming distances satisfy these conditions,

For a subset $X \subseteq \mathbb{B}^n$ and a vector $u \in \mathbb{B}^n$ let us define

$$\rho(u, X) = \min_{v \in X} \rho(u, v). \tag{20}$$

A *nearest neighbor classification rule* f_ρ can be naturally associated with every similarity measure ρ by declaring that an arbitrary vector v is "positive" if and only if v at least as close to T as to F. We will prove in Section A.4 that, when ρ is subcube monotone (which is the case for most usual similarity measures), then the classifier produced by this typical rule is always a bi-theory:

Theorem 3.16. *If (T, F) is a pdBf and if ρ is a subcube monotone similarity measure, then the Boolean function f_ρ defined by*

$$f_\rho(v) = \begin{cases} 1 & \text{if } \rho(v,T) \leq \rho(v,F), \\ 0 & \text{otherwise} \end{cases} \tag{21}$$

is a bi-theory of (T, F).

4 Empirical evidence

The focus of the present paper is primarily theoretical, as it aims at developing the notion of "justifiability", and at analyzing the mathematical structure of the resulting concepts, patterns and bi-theories, in particular. Our line of arguing, however, naturally leads to favoring rules (in our case, patterns or co-patterns) which "fit well" the given data, as expressed by Axiom (A2) in Section 1.2 of the paper. As is well known, such "maximalist" requirement may possibly lead to overfitting, as many researchers observed in similar situations when using different classification approaches.

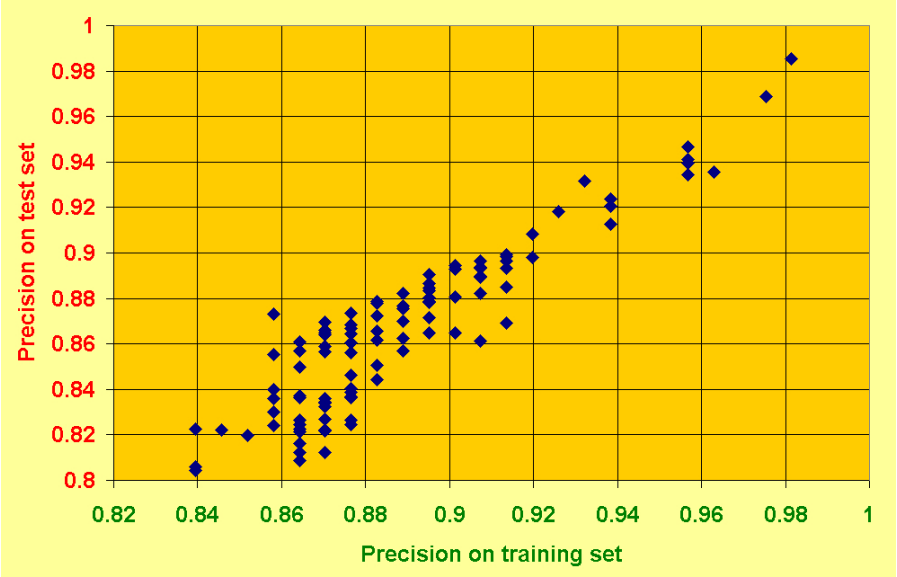

Figure 2: Results with the mushroom data set. Each point represents a pattern, the coordinates of which are the percentages of correctly classified training and test cases, when we use this pattern as a classifier. The horizontal scale is the percentage of correctly classified training cases, while the vertical axis represents the percentage of correctly classified test cases.

For this reason, it is important to stress that such overfitting behavior does not necessarily occur. This claim is based on extensive empirical evidence which has been reported elsewhere, and which we briefly summarize hereunder.

First of all, there were several recent attempts in the literature to build patterns which have the highest coverage in the given training set and to use such patterns for classification, see e.g., [8, 9, 17, 21]. All of these papers reported good results, derived highly robust classifiers, and none experienced overfitting.

We also designed various other computational experiments to test this behavior. In order to illustrate our point, we include here a small representative example. We considered some examples from the UC Irvine machine learning repository [4], chose randomly a small part of the data set as training set (typically 5-10%), left the rest as test data, and generated exhaustively all patterns from the training set. Then, each pattern t was used as a classifier, both on the training set and on the test set (in both cases, an example a is classified as a positive example if and only if it is covered by the pattern, that is, if and only if $t(a) = 1$). For each pattern we computed the percentage of the training cases classified correctly by this pattern, as well as the percentage of the test cases classified correctly. Thus each pattern is characterized by two percentages. We include here as illustration the results obtained for the so called "mushroom" data set. Figure 2

indicates the quality of the classification achieved by the 218 patterns which performed best on the training set; each of these patterns classified correctly at least 84% of the training cases.

What is even more surprising, however, is that the graph indicates a clear trend: Namely, those patterns performing better on the training set also perform better on the test set. Furthermore, even the variance of the test performance seems to decrease when the training performance increases. In fact we detected the same (sharp) tendency on all the data sets we examined. We view this as strong evidence supporting Dr P's approach: Choosing the best performing patterns based on training set performance does not seem to lead to overfitting, and serves well our requirement to base classifications on the best available justification.

Of course, we cannot claim that using patterns and/or bi-theories will never result in overfitting: since we have shown that several classical families of classifiers (e.g., decision trees) are bi-theories, the worst-case behavior of bi-theories, for instance, cannot be better than the worst-case behavior of decision trees in this respect (and decision trees are known to overfit!). Conversely, however, one may hope that appropriately chosen bi-theories display little overfitting; and this is what seems indeed to emerge from the experiments reported above.

5 Future research

We can summarize our main contributions as follows: We introduced the notion of justifiability of a classifier, and concluded that all justifiable classifiers must be bi-theories. We also established that bi-theories are closely related to decision trees and nearest neighbor methods, but still form a larger class than these two classes produced by classical methods. We also analyzed the structure of the pattern space in relation with bi-theories, and revealed the existence of various remarkable subsets of vectors (T^-, T^+, F^-, F^+) associated with an arbitrary pdBf (T, F).

Many open questions emerge from our study: How much larger is the family of bi-theories than the family of decision trees? What is the proportion of bi-theories within the family of theories? Which Boolean functions can appear as maximal theories for a pdBf? How difficult is to test whether $T^+ = T$ (or $F^+ = F$)? How difficult is to generate T^+ and F^+? We leave these questions for future research.

References

[1] H. Alhammady and K. Ramamohanarao, Using Emerging Patterns and Decision Trees in Rare-Class Classification, In: *Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM'04)*, 315-318, 2004.

[2] R. Agrawal, T. Imielinski and A. Swami, Mining association rules between sets of items in large databases, In: *International Conference on Management of Data (SIGMOD 93)*, (1993), 207-216.

[3] D. Angluin, Queries and concept learning, *Machine Learning* 2 (1988) 319–342.

[4] A. Asuncion and D.J. Newman, UCI Machine Learning Repository [http://www.ics.uci.edu/ mlearn/MLRepository.html]. Irvine, CA: University of California, School of Information and Computer Science. (2007)

[5] L. Breiman, J.H. Friedman, R.A. Olshen and C.J. Stone, *Classification and Regression Trees*, Wadsworth International Group, 1984.

[6] T.O. Bonates and P.L. Hammer, Logical Analysis of Data: From combinatorial optimization to medical applications, *Annals of Operations Research* 148 (2006) 203–225.

[7] T.O. Bonates and P.L. Hammer, Large margin LAD classifiers. Technical Report RRR 22-2007, RUTCOR - Rutgers Center for Operations Research, Rutgers University, 2007.

[8] T.O. Bonates and P.L. Hammer, A branch-and-bound algorithm for a family of pseudo-Boolean optimization problems. Technical Report RRR 21-2007, RUTCOR - Rutgers Center for Operations Research, Rutgers University, 2007.

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

[10] E. Boros, V. Gurvich, P.L. Hammer, T. Ibaraki and A. Kogan, Decomposability of partially defined Boolean functions, *Discrete Applied Mathematics* 62 (1995) pp. 51-75.

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

[12] E. Boros, P.L. Hammer, T. Ibaraki and A. Kogan, Logical analysis of numerical data, *Mathematical Programming* 79 (1997) pp. 163-190.

[13] E. Boros, T. Ibaraki and K. Makino, Error-free and best-fit extensions of partially defined Boolean functions, *Information and Computation* 140 (2) (1998) pp. 254-283.

[14] Y. Crama and P.L. Hammer, *Boolean Functions – Theory, Algorithms, and Applications*, Cambridge University Press, to appear (`http://www.rogp.hec.ulg.ac.be/crama/`).

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

[16] G. Dong and J. Li, Efficient mining of emerging patterns: discovering trends and differences, In: *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, (1999) 43-52.

[17] J. Eckstein, P.L. Hammer, Y. Liu, M. Nediak and B. Simeone, The maximum box problem and its application to data analysis, *Computational Optimization and Applications*, 23(3):285298, 2002.

[18] A. Ehrenfeucht and D. Haussler, Learning decision trees from random examples, *Information and Computation* 82 (1989) 231–246.

[19] C. Flament, L'analyse booléenne de questionnaires, *Mathématiques et Sciences Humaines* 12 (1966) 3-10.

[20] B. Ganter and R. Wille, *Formal Concept Analysis - Mathematical Foundations*, Springer-Verlag, Berlin, 1999.

[21] N. Goldberg and Ch. Shan, Boosting optimal logical patterns, In: *Proceedings of the Seventh SIAM International Conference on Data Mining Edited by Chid Apte, Bing Liu, Srinivasan Parthasarathy, and David Skillicorn*, 2007.

[22] P.L. Hammer, A. Kogan, B. Simeone and S. Szedmak. Pareto-optimal patterns in Logical Analysis of Data. *Discrete Applied Mathematics*, 144(1-2) (2004) pp. 79102.

[23] L. Hyafil and R.L. Rivest, Constructing optimal binary decision trees is NP-complete, *Information Processing Letters*, 5 (1976) 15-17.

[24] D. Kleitman, On Dedekind's problem: The number of monotone Boolean functions, *Proceedings of the American Mathematical Society* 21 (1969) 677-682.

[25] A. Kogan and Y. I. Zhuravlev, Realization of Boolean functions with a small number of zeros by disjunctive normal forms and related problems, *Soviet Mathematics - Doklady (American Mathematical Society)*, 32 (3) (1985) 771-775.

[26] A. Kogan, Disjunctive normal forms of Boolean functions with a small number of zeros, *USSR Computational Mathematics and Mathematical Physics (Pergamon Press)*, 27 (3) (1987) 185-190.

[27] A. Kogan, Lower bounds for the complexity of disjunctive normal forms of Boolean functions with a small number of zeros, *USSR Computational Mathematics and Mathematical Physics (Pergamon Press)*, 27 (6) (1987) 175-181.

[28] K. Makino, T. Suda, T. Ono and T. Ibaraki, Data analysis by positive decision trees, *IEICE Transactions on Information and Systems*, Vol. E82-D, No.1, pp. 76-88, January 1999.

[29] S. Muroga, *Threshold Logic and its Applications*, Wiley-Interscience, New York, 1971.

[30] S.K. Murthy, S. Kasif and S. Salzberg, A system for induction of oblique decision trees, *JAIR*, 2 (1994) 1-32.

[31] R. Potharst, J.C. Bioch and T. Petter, Monotone decision trees, Technical Report EUR-FEW-CS-97-07, Erasmus University, Rotterdam, 1997.

[32] J.R. Quinlan, Induction of decision trees, *Machine Learning* 1 (1986) 81-106.

[33] C.C. Ragin, *The Comparative Method*, University of California Press, Berkeley Los Angeles London, 1987.

[34] L. G. Valiant, A theory of the learnable, *Communications of the ACM* 27 (1984) 1134–1142

A Proofs of main results

In this section we provide the proofs and necessary background of the results stated in Section 3.

A.1 Patterns and co-patterns

Let us first state an easy property of subcubes.

Lemma A.1. *If $c \in [a, b]$ and $c \neq b$, then $b \notin [a, c]$.*

Proof. By definition of the subcube $[a, b]$, $c \in [a, b]$ means that $c_j = a_j = b_j$ for all indices $j \in \{1, ..., n\}$ for which $a_j = b_j$. Thus, $c \neq b$ implies the existence of an index i such that $c_i = a_i \neq b_i$, which then by the definition of subcubes implies that $b \notin [a, c]$. □

Now we are ready to prove a main result about patterns and closures.

Proof of Theorem 3.1.

The second half of the claim follows from the first one by (ii) of Claim 2.4(ii). So, we concentrate on the first statement only.

Let us denote by $\mathcal{F}_T$ the family of all such subsets $F' \subseteq \mathbb{B}^n$ for which $P(T, F') = P(T, F)$, and observe that by (iii) of Claim 2.4 we have

$$F' \subseteq F'' \subseteq F''' \quad \text{and} \quad F', F''' \in \mathcal{F}_T \quad \text{imply} \quad F'' \in \mathcal{F}_T.$$

Thus, to complete the proof of the theorem it is enough to show that $\mathcal{F}_T$ has unique minimal and maximal elements.

Let us next show that by (iii) of Claim 2.4 and by the definition of a pattern we have

$$F', F'' \in \mathcal{F}_T \quad \text{implies} \quad F' \cup F'' \in \mathcal{F}_T. \tag{22}$$

This is because if t is a pattern of both (T, F') and (T, F''), then we must have $t(b) = 0$ for all $b \in F' \cup F''$, and $t(a) = 1$ for some $a \in T$. Thus, t is also a pattern of $(T, F' \cup F'')$, implying $P(T, F') = P(T, F'') \subseteq P(T, F' \cup F'')$, which together with (iii) of Claim 2.4 implies that $P(T, F) = P(T, F' \cup F'')$.

Thus, (22) implies that $\mathcal{F}_T$ has a unique maximal element, which we can denote by F^+.

To establish the existence of a unique minimal element F^- in $\mathcal{F}_T$, it is enough to show that

$$F', F'' \in \mathcal{F}_T \quad \text{implies} \quad F' \cap F'' \in \mathcal{F}_T. \tag{23}$$

So, let us assume by contradiction that $F', F'' \in \mathcal{F}_T$ but $F' \cap F'' \notin \mathcal{F}_T$. Since (iii) of Claim 2.4 implies $P(T, F') = P(T, F'') \subseteq P(T, F' \cap F'')$, the assumption means that there is a term $t \in P(T, F' \cap F'')$ which is not a member of $P(T, F') = P(T, F'')$, or in other words, for which there exist vectors $a \in T$, $b' \in F' \setminus F''$ and $b'' \in F'' \setminus F'$ such that $t(a) = t(b') = t(b'') = 1$.

This means that $T(t)$ is a subcube of $\mathbb{B}^n$ which intersects T, F' and F'', but does not intersect $F' \cap F''$. Let us then choose a minimal subcube included in $T(t)$ and which intersects both T and $F' \cup F''$. Clearly, by Lemma A.1, such a minimal subcube contains exactly one vector from T and one vector from $(F' \cup F'') \setminus (F' \cap F'')$. Let us assume, without any loss of generality that $a \in T$ and $b \in F' \setminus F''$ are such vectors, and the minimal subcube is $[a, b]$. Then, the term t^* defined by $T(t^*) = [a, b]$ would be a pattern of (T, F'') but not a pattern of (T, F'), contradicting our assumption that $P(T, F') = P(T, F'')$. This contradiction proves our claim, and hence completes the proof of the theorem. □

Proof of Theorem 3.2

Assume that $x \in \mathbb{B}^n \setminus T$ is covered by some pattern t of (T, F). Then, by definition, t is not a pattern of $(T, F \cup \{x\})$, and Theorem 3.1 implies that $F \cup \{x\} \not\subseteq F^+$, that is that $x \notin F^+$.

Conversely, let $x \notin F^+$. Then, by Theorem 3.1, $P(T, F) \neq P(T, F \cup \{x\})$. But this implies by (iii) of Claim 2.4 that there is a pattern of (T, F) which covers x.

The statement about T^+ follows from the above by (ii) of Claim 2.4. □

It is easy to check that Corollary 3.3 immediately follows Theorem 3.2.

Proof of Theorem 3.4

We want to prove that $F^+ = F^*$, where

$$F^* = \{x \in \mathbb{B}^n \mid [x, a] \cap F \neq \emptyset \text{ for all } a \in T\}. \tag{24}$$

To see that $F^+ \subseteq F^*$, let us consider a vector $x \notin F^*$: thus, there exists a vector $a \in T$ such that $[x, a] \cap F = \emptyset$. For the term t defined by $T(t) = [x, a]$, we have $t(a) = 1$ and $t(b) = 0$ for all $b \in F$; hence, t is a pattern of (T, F). Since this pattern covers x, Theorem 3.2 implies that $x \notin F^+$, and we conclude that $F^+ \subseteq F^*$ as required.

To see the equality, it is enough to show by (iii) of Claim 2.4 that $P(T, F) \subseteq P(T, F^*)$, since $F \subseteq F^*$ by definition. Let us assume indirectly that $t \in P(T, F) \setminus P(T, F^*)$, or in other words that t is a term for which $t(b) = 0$ for all $b \in F$, $t(a) = 1$ for some $a \in T$ and $t(x) = 1$ for some $x \in F^*$. Then we must have $x \in F^* \setminus F$, and $[a, x] \subseteq T(t)$. However, by (24) we must have $[a, x] \cap F \neq \emptyset$, proving that $T(t) \cap F \neq \emptyset$, and hence contradicting the assumption that t is a pattern of (T, F). This contradiction proves the equality $F^+ = F^*$, as claimed.

To establish (11), let us denote by $\hat{F}$ the right hand side of this equality, i.e.,

$$\hat{F} = \{b \in F \mid \exists\ a \in T \text{ such that } [a, b] \cap (F \setminus \{b\}) = \emptyset\}. \tag{25}$$

We must prove that $\hat{F} = F^-$ as defined in Theorem 3.1.

We claim first that $\hat{F} \subseteq F^-$. Indeed, if $b \in \hat{F} \setminus F^-$, then there exists $a \in T$ for which $[a,b] \cap (F \setminus \{b\}) = \emptyset$ by (25). The term t defined by $T(t) = [a,b]$ is not a pattern of (T,F) since we have $t(a) = t(b) = 1$ for this particular $a \in T$ and $b \in F$. Hence, by $P(T,F) = P(T,F^-)$, the term t cannot be a pattern of (T,F^-), either. Since $t(a) = 1$ and $a \in T$, this implies that there exists a vector $b' \in F^-$ for which $t(b') = 1$, implying $b' \in [a,b]$. Then, we have $[a,b'] \subseteq [a,b]$, and by Lemma A.1 $b \notin [a,b']$ is also implied. Let us then consider the term t' for which $T(t') = [a,b']$. Since $[a,b'] \cap F = \emptyset$ by the above construction, t' is a pattern of (T,F). However $t'(b') = 1$ implies that t' is not a pattern of (T,F^-), contradicting the definition of F^-.

Next, we claim that $P(T,\hat{F}) = P(T,F)$, which together with the previous claim and with the definition of F^- will prove that $\hat{F} = F^-$. By (iii) of Claim 2.4, it in fact enough to show that $P(T,\hat{F}) \subseteq P(T,F)$.

To verify this latter relation, let us assume indirectly that there exists a pattern $t \in P(T,\hat{F}) \setminus P(T,F)$. Consequently, we must have vectors $a \in T$ and $b \in F \setminus \hat{F}$ such that $t(a) = t(b) = 1$. Since $b \notin \hat{F}$, we must have $[a,b] \cap (F \setminus \{b\}) \neq \emptyset$ by (25). Since t is a pattern of $(T,\hat{F})$ and, since $T(t) \supseteq [a,b]$, all elements $b' \in [a,b] \cap (F \setminus \{b\})$ must belong to $F \setminus \hat{F}$, and clearly $[a,b'] \subset [a,b]$ holds for all such elements. Let us then choose a vector $b' \in [a,b] \cap (F \setminus \{b\})$ for which $[a,b']$ is a minimal sub-cube among all such sub-cubes. Since $b' \notin \hat{F}$, there must exist another vector $b'' \in [a,b'] \cap (F \setminus \{b'\})$, and by the selection of b' we must have $b'' \in \hat{F}$. However, $t(b'') = 1$ follows from $[a,b'] \subseteq T(t)$, contradicting the fact that t was chosen as a pattern of $(T,\hat{F})$. This contradiction proves that $P(T,\hat{F}) = P(T,F)$ as we claimed.

The statements (12) and (13) follow from the above by interchanging the roles of T and F. □

A.2 Maximal theories

Proof of Theorem 3.8

Let $x \in \mathbb{B}^n$ be an arbitrary binary vector and let $a \in T \cup F$ be a vector closest to x in the sense of the Hamming distance. We can assume without any loss of generality that $a \in T$. Let us then consider the term t defined by $T(t) = [x,a]$. By Lemma A.1 we can assume that $a \in T$ is the only vector from $T \cup F$ in $[x,a]$, and hence $t(b) = 0$ must hold for all $b \in F$, implying that t is a pattern of (T,F). Hence, $A_{(T,F)}(x) = 1$ must hold. Analogously, if $a \in F$, then we can derive that $B_{(T,F)}(x) = 1$. This completes the proof of the first claim.

The identities $F(A_{(T,F)}) = F^+$ and $F(B_{(T,F)}) = T^+$ are a mere restatement of Theorem 3.2, and the relation $T^+ \cap F^+ = \emptyset$ follows readily from this fact and from the first claim. □

Proof of Theorem 3.10

Claim (i) follows readily by Theorem 3.8 and by the definition of patterns and co-patterns.

To see (ii), let us consider an arbitrary pattern t of (T^+, F^+). By definition we have $t(b) = 0$ for all $b \in F^+$, and thus $T(t) \subseteq \mathbb{B}^n \setminus F^+ = T(A_{(T,F)})$ follows from Theorem 3.8, implying that t is an implicant of $A_{(T,F)}$.

Claim (iii) is implied by (i) and (ii). □

Proof of Theorem 3.11

To prove claim (iv), first note that any support set of (T^+, F^+) is also a support set of (T, F) since we have $T^+ \supseteq T$ and $F^+ \supseteq F$. Therefore, let us consider a support set S of (T, F) and show that it is also a support set of (T^+, F^+). If this were not true, there would exist two vectors $a \in T^+$ and $b \in F^+$ for which $a_i = b_i$ for all $i \in S$. Assume without loss of generality that $a_i = b_i = 1$ for all $i \in S$ and let $t = \bigwedge_{i \in S} x_i$, so that we have

$$t(a) = 1 \quad \text{and} \quad t(b) = 1. \tag{26}$$

Now, let us choose a vector $c \in T \cup F$ for which the cardinality of the set $I = \{i \in S : c_i = 1\}$ is as large as possible, and define the term $t' = \bigwedge_{i \in I} x_i$. Clearly, $t'(c) = 1$. Without any loss of generality, we can assume that $c \in T$ (the case $c \in F$ would be similar). Then, we claim that $t'(w) = 0$ for all $w \in F$. Indeed, if $t'(w) = 1$ for some $w \in F$, then it means that $w_i = 1$ for all $i \in I$. Since S is a support set of (T, F), there must be an index $j \in S \setminus I$ such that $w_j \neq c_j$. Now, $j \notin I$ implies that $c_j = 0$; hence this and $w_j = 1$ contradicts the choice of c (since the set $\{i \in S : w_i = 1\}$ is larger than I).

Thus, $t'(w) = 0$ for all $w \in F$, and $t'(c) = 1$, meaning that t' is a pattern of (T, F). Then $t'(b) = 0$ follows from Theorem 3.2 and from the assumption that $b \in F^+$. Since $t \leq t'$, we conclude that $t(b) = 0$, which contradicts the second equality in (26). □

A.3 Decision trees and bi-theories

Recall the definition of reasonable decision trees given at the end of Section 2.2. Such decision trees offer an algorithmic representation of Boolean functions and of pdBfs. They are widely used in machine learning, data mining, and other fields. Moreover, we shall establish their close relationship with bi-theories.

Example A.2. Figure 3 shows an example of a decision tree D. This decision tree classifies, for instance, all binary vectors for which $x_1 = x_2 = x_5 = 1$ into the rightmost leaf belonging to L_1. Thus, we have $f_D(1, 1, x_3, x_4, 1) = 1$ for all x_3, x_4. □

For each node $v \in N \cup L$ of a decision tree D, let us denote by $P_v = \{r = u_0, u_1, ..., u_k = v\}$ the unique path from the root r to v. We can associate an elementary conjunction t_v with v by defining

$$t_v = \Big(\bigwedge_{\substack{u_i \in P_v: \\ u_{i+1} \text{ is the right successor of } u_i}} x_{j(u_i)} \Big) \wedge \Big(\bigwedge_{\substack{u_i \in P_v: \\ u_{i+1} \text{ is the left successor of } u_i}} \overline{x}_{j(u_i)} \Big).$$

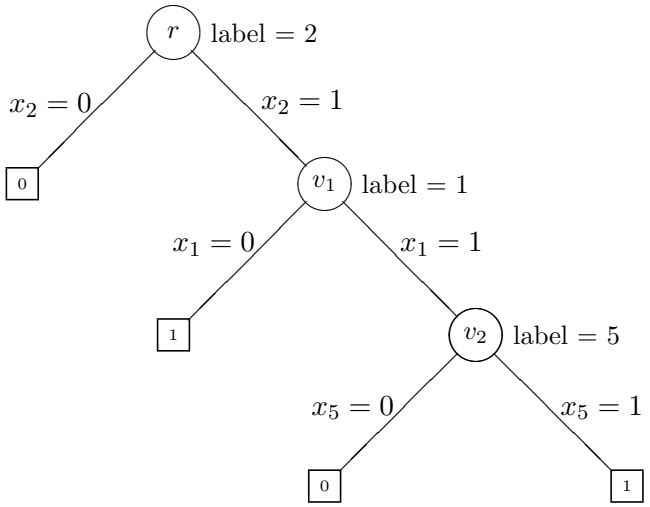

Figure 3: An example of a decision tree.

In words, we include the literal x_j in t_v if the path P_v follows the right successor from a node $u \in P_v$ with $j(u) = j$, and we include $\overline{x}_j$ if P_v follows the left successor. It is then immediate to see that DNF representations of f_D and $\overline{f}_D$ are given by

$$f_D = \bigvee_{v \in L_1(D)} t_v \quad \text{and} \quad \overline{f}_D = \bigvee_{v \in L_0(D)} t_v. \tag{27}$$

Example A.3. The decision tree D in Figure 3 has two leaves in $L_1(D)$, and thus we obtain the following DNF for the function represented by D:

$$f_D = \overline{x}_1 x_2 \vee x_1 x_2 x_5.$$

Considering the leaves in $L_0(D)$ we get

$$\overline{f}_D = \overline{x}_2 \vee x_1 x_2 \overline{x}_5.$$

It is well-known that every pdBf (T, F) can be represented by a reasonable decision tree. We provide next a generic procedure to build such a decision tree $D \in \mathcal{D}(T, F)$. Although we do not claim any originality for this procedure, its description will be useful in order to establish the results in Section A.3.1.

With every node v of D (to be built), let us associate the pdBf (T_v, F_v), consisting of those vectors of T and F, respectively, which are classified into node v. During the execution of the proposed algorithm we maintain a list Q of nodes v such that

$$T_v \cup F_v \neq \emptyset \quad \text{and} \quad T_v \cap F_v = \emptyset. \tag{28}$$

DT-BUILD(T, F)

Initialize: Let $N = Q = \{r\}$, $L_1 = L_0 = \emptyset$, $A(D) = \emptyset$, $T_r = T$ and $F_r = F$. (Upon completion, the vertex set $N \cup L_0 \cup L_1$ and the arc set $A(D)$ define the constructed decision tree.)

While $Q \neq \emptyset$ **do**

Choose a node $u \in Q$ and remove it from Q.

If $T_u = \emptyset$, then add u to L_0 and remove it from N.

If $F_u = \emptyset$, then add u to L_1 and remove it from N.

If $T_u \neq \emptyset$ **and** $F_u \neq \emptyset$, then choose an index j such that x_j is not constant for all $x \in T_u \cup F_u$, and set $j(u) = j$. Let v and w, respectively, be the left and right successors of u, and define the associated pdBfs by

$$T_v = \{a \in T_u \mid a_j = 0\} \quad \text{and} \quad F_v = \{b \in F_u \mid b_j = 0\},$$
$$T_w = \{a \in T_u \mid a_j = 1\} \quad \text{and} \quad F_w = \{b \in F_u \mid b_j = 1\}.$$

Finally, add both v and w to Q and N, and add the arcs (u, v) and (u, w) to the set of arcs $A(D)$.

Theorem A.4. *For every pdBf (T, F), the algorithm* DT-BUILD(T, F) *produces a reasonable decision tree $D \in \mathcal{D}(T, F)$. Moreover, all reasonable decision trees in $\mathcal{D}(T, F)$ arise in this way.*

Proof. Let us note first that $T_v \cup F_v \neq \emptyset$ and $T_w \cup F_w \neq \emptyset$ holds in the above algorithm for the successors v and w of a node u if and only if the index $j = j(u)$ is chosen so that x_j is not a constant in all vectors $x \in T_v \cup F_v$. Furthermore, if $T_u \cap F_u = \emptyset$, then we have both $T_v \cap F_v = \emptyset$ and $T_w \cap F_w = \emptyset$. Thus, conditions (28) are indeed maintained during the procedure, assuming that we had initially $T \cap F = \emptyset$.

Let us also remark that as long as $T_u \neq \emptyset$, $F_u \neq \emptyset$, and $T_u \cap F_u = \emptyset$, there must exist an index j for which x_j is not constant in all $x \in T_u \cup F_u$. After the data splitting at each vertex u, we have $x_{j(u)} = 1$ in all vectors $x \in T_z \cup F_z$ for all nodes z belonging to the subtree rooted at the right successor w of u. Similarly, we have $x_{j(u)} = 0$ for all vectors $x \in T_z \cup F_z$ for all nodes z belonging to the subtree rooted at the left successor v of u. This implies that the same index will not be selected twice along any of the paths going from the root to a leaf.

Let us finally observe that every time we add a node u to either L_0 or L_1, the corresponding pdBf (T_u, F_u) contains some vectors, due to condition (28).

Therefore, the decision tree D produced by DT-BUILD(T, F) indeed represents (T, F) and it is reasonable. We can also see that the total number

of leaves is not more than $|T \cup F|$, and hence the procedure terminates in $O(|T \cup F|)$ steps.

It is obvious that every reasonable decision tree D for (T, F) can be produced by DT-Build, since it suffices to choose the indices $j(u)$ as prescribed by D. □

Corollary A.5. *If (T, F) is a pdBf, then $\mathcal{D}(T, F) \neq \emptyset$.* □

Many variants of DT-Build are proposed in the literature, differing (only) in the way of choosing the splitting index $j = j(u)$ in each iteration. One of the best-known procedures is the algorithm ID3 due to Quinlan [32].

A.3.1 Properties of decision trees as bi-theories

Let us now analyze the connection between decision trees and patterns, co-patterns, theories, and co-theories.

Lemma A.6. *For a decision tree $D \in \mathcal{D}(T, F)$ and for a leaf node v of D, the corresponding term t_v is a pattern of (T, F) if $v \in L_1(D)$, while it is a co-pattern of (T, F) if $v \in L_0(D)$.*

Proof. This is almost immediate by the definitions. For instance, if $v \in L_1(D)$, then $T_v \neq \emptyset$ and $F_v = \emptyset$, implying that $t_v(a) = 1$ for all $a \in T_v$ and $t_v(b) = 0$ for all $b \in F$. □

Proof of Theorem 3.13

Let us consider an arbitrary reasonable decision tree of (T, F). By Corollary A.5 there are such decision trees. Then by Lemma A.6 every term of f_D (and $\overline{f}_D$), given by (27) is a pattern (co-pattern). Consequently f_D is a bi-theory. □

Before we turn to a proof of Theorem 3.15, we need an additional definition. Let us call a term t a prime pattern (prime co-pattern) of a pdBf (T, F) if it is a pattern (co-pattern) of (T, F) but every term obtained by dropping any one of its literals is not a pattern (co-pattern). In other words, when viewed as either Boolean functions or subcubes, prime patterns (prime co-patterns) are maximal patterns (co-patterns).

Lemma A.7. *Every prime pattern t of (T, F) appears as $t = t_v$ for some reasonable decision tree $D \in \mathcal{D}(T, F)$ and for some true leaf $v \in L_1(D)$. Similarly, every prime co-pattern t of (T, F) appears as $t = t_u$ for some reasonable decision tree $D \in \mathcal{D}(T, F)$ and for some false leaf $u \in L_0(D)$.*

Proof. Let us prove the statement for prime patterns. The case of prime co-patterns can be treated analogously.

Let $t = \bigwedge_{j_i \in P(t)} x_{j_i} \bigwedge_{j_i \in N(t)} \overline{x}_{j_i}$ be a prime pattern of (T, F), let $P(t) \cup N(t) = \{j_i \mid i = 1, ..., k\}$, and let us consider a small decision tree D^* consisting of one path u_1, u_2, ..., u_{k+1}. Here u_1 is the root of D^*, and u_{i+1} is the right successor of u_i if $j_i \in P(t)$, and it is the left successor if $j_i \in N(t)$. Let v_i denote the other successor of vertex u_i, $i = 1, ..., k$, and set $L_1(D^*) = \{u_{k+1}\}$ and $L_0(D^*) = \{v_1, v_2, ..., v_k\}$.

Let us now note that $F_{u_{k+1}} = \{b \in F \mid t(b) = 1\} = \emptyset$ and $T_{u_{k+1}} = \{a \in T \mid t(a) = 1\} \neq \emptyset$ since t is a pattern of (T, F). Since t is prime, it does not remain a pattern if any variable x_{j_i} is deleted from it. Thus there must exist a vector $b \in F$ for which $b_{j_i} \neq a_{j_i}$ for some i and $b_{j_\ell} = a_{j_\ell}$ for $\ell \neq i$, where $a \in T$ is an arbitrary vector for which $t(a) = 1$. Therefore, $F_{v_i} = \{b \in F \mid b_{j_i} \neq a_{j_i} \text{ and } b_{j_\ell} = a_{j_\ell},\ \ell < i\} \neq \emptyset$. Since this is true for all indices j_i, $i = 1, ..., k$, every node of D^* has some vectors of $T \cup F$ classified into it.

Then let us choose an arbitrary decision tree $D_i \in \mathcal{D}(T_{v_i}, F_{v_i})$ and identify the root of D_i with v_i, for $i = 1, ..., k$. In this way we obtain a decision tree D for which we have $D \in \mathcal{D}(T, F)$ and in which t appears as $t = t_{u_{k+1}}$. □

This lemma allows us to give one more characterization of the closure (T^+, F^+) of a pdBf (T, F).

Proof of Theorem 3.15

Let us first show the equivalence $(a) \iff (b)$. Note that no decision tree $D \in \mathcal{D}(T, F)$ can classify a vector $v \in T^+$ into a false leaf, or a vector $u \in F^+$ into a true leaf, since every decision tree $D \in \mathcal{D}(T, F)$ represents a bi-theory of (T, F) by Theorem 3.13, and since $T^+ = F(B_{(T,F)})$ and $F^+ = F(A_{(T,F)})$ by Theorem 3.8. Thus, to complete the proof of this equivalence we only need to show that no vectors $u \notin T^+$ or $u \notin F^+$ have the same property.

For this, let us consider a vector $u \notin T^+$. Then, by Definition 12, there exists a vector $b \in F$ such that $[u, b] \cap T = \emptyset$. Thus the term t defined by $T(t) = [u, b]$ is a co-pattern of (T, F). Let $t' \geq t$ be a prime co-pattern of (T, F). Then, by Lemma A.7 there exists a decision tree $D \in \mathcal{D}(T, F)$ for which $t' = t_v$ for some leaf $v \in L_0(D)$. This decision tree classifies u into the false leaf v, showing that not all decision trees in $\mathcal{D}(T, F)$ classify u into a true leaf, as claimed.

The case of a vector $u \notin F^+$ is similarly handled.

To see the equivalence $(a) \iff (c)$, let us note that $(c) \implies (b) \implies (a)$ by Theorem 3.13 and the above proof. Furthermore, Corollary 3.3 implies $(a) \implies (c)$, thus completing our proof. □

A.4 Nearest neighbor methods and bi-theories

The following useful property can be deduced from (18)–(19): It states that if u lies between a and v, then v is closer to u than to a.

Lemma A.8. *For all vectors $a, u, v \in \mathbb{B}^n$, if $u \in [a, v]$ and $a \neq u$, then $\rho(u, v) < \rho(a, v)$ when ρ is a subcube monotone similarity measure.*

Proof. Let $u \in [a, v]$. Condition (19) implies that, if we indirectly assume $\rho(a, v) \leq \rho(u, v)$, then $\rho(a, u) \leq \rho(u, u) = 0$. Hence $\rho(a, u) = 0$, and (18) implies that $a = u$, a contradiction. □

Proof of Theorem 3.16

Let us note first that since $T \cap F = \emptyset$, we have for all $a \in T$ and $b \in F$: $\rho(a, F) > 0 = \rho(a, a) = \rho(a, T)$ and $\rho(b, T) > 0 = \rho(b, b) = \rho(b, F)$, by (18) and (20). Thus $T \subseteq T(f_\rho)$ and $F \subseteq F(f_\rho)$, implying that $f_\rho \in \mathcal{E}(T, F)$.

To see that f_ρ is a theory, let us associate with every vector $v \in T(f_\rho)$ the term $t_{[v,a]}$ defined by $T(t_{[v,a]}) = [v, a]$, where $a \in T$ is any vector such that $\rho(v, a) = \rho(v, T)$, and let us define

$$\varphi = \bigvee_{v \in T(f_\rho)} t_{[v,a]}.$$

We want to show that φ is a theory and that $\varphi = f_\rho$.

Let us first show that every term $t_{[v,a]}$, $v \in T(f_\rho)$, is a pattern of (T, F). Clearly, the term $t_{[v,a]}$ covers a point of T (namely, a). Moreover, suppose that $t_{[v,a]}$ covers a point of F, say $u \in F \cap [v, a]$. Then, by Lemma A.8, $\rho(u, v) < \rho(a, v)$, which contradicts the assumption that $v \in T(f_\rho)$.

Thus, for all $v \in T(f_\rho)$, $t_{[v,a]}$ intersects T and does not intersect F, meaning that $t_{[v,a]}$ is a pattern of (T, F), and that φ is a theory. We claim next that this theory is f_ρ, that is,

$$T(\varphi) = T(f_\rho). \tag{29}$$

Clearly, for all $v \in T(f_\rho)$, the term $t_{[v,a]}$ covers v, and hence $\varphi(v) = 1$; this shows that $T(f_\rho) \subseteq T(\varphi)$.

For the converse inclusion, consider an arbitrary vector $u \in T(\varphi)$, and let $t_{[v,a]}$, $v \in T(f_\rho)$, be any term of φ which covers u: $u \in T(t_{[v,a]}) = [v, a]$. For every vector $b \in F$, we have by definition of f_ρ that $\rho(v, T) = \rho(a, v) \leq \rho(b, v)$; hence, (19) implies $\rho(a, u) \leq \rho(b, u)$. Since this holds for all $b \in F$, we conclude that $\rho(u, T) \leq \rho(u, F)$, hence $f_\rho(u) = 1$. This establishes the claim (29).

Finally, to see that f_ρ is a bi-theory, we have to show that its complement $\overline{f}_\rho$ is a co-theory of (T, F). To this end, let us associate with every vector $w \in F(f_\rho)$ the term $t_{[w,b]}$ defined by $T(t_{[w,b]}) = [w, b]$, where $b \in F$ is a vector for which $\rho(w, b) = \rho(w, F)$, and define

$$\psi = \bigvee_{w \in F(f_\rho)} t_{[w,b]}.$$

By similar arguments as above, it follows that all terms of ψ are co-patterns, and that $T(\psi) = F(f_\rho)$. Thus, $\overline{f}_\rho$ is a co-theory of (T, F), which completes the proof of the theorem. □

As mentioned earlier, many mappings $\rho : \mathbb{B}^n \times \mathbb{B}^n \to \mathbb{R}_+$ are subcube monotone similarity measures. The Hamming distance provides a simple example. We describe below a large family of subcube monotone mappings which generalize the Hamming distance (the Hamming distance is obtained when $\omega_j = 1$ for all $j = 1, 2, \ldots, n$).

Lemma A.9. *Let $\omega_j > 0$ be positive real numbers for $j = 1, 2, \ldots, n$, and let*

$$\rho_\omega(a, b) = \sum_{j : a_j \neq b_j} \omega_j \quad \text{for all } a, b \in \mathbb{B}^n. \tag{30}$$

Then ρ_ω is a subcube monotone similarity measure.

Proof. Condition (17) holds trivially, and condition (18) follows from the positivity of ω_j for $j = 1, 2, \ldots, n$. To see condition (19), let us consider arbitrary binary vectors a, b, u and v, such that $u \in [v, a]$, and let us define the following index sets

$$\begin{aligned}
A &= \{j \mid a_j = v_j = u_j = b_j\},\\
B &= \{j \mid a_j = v_j = u_j \neq b_j\},\\
C &= \{j \mid a_j = b_j = u_j \neq v_j\},\\
D &= \{j \mid b_j = u_j = v_j \neq a_j\},\\
E &= \{j \mid a_j = b_j \neq u_j = v_j\},\\
F &= \{j \mid a_j = u_j \neq b_j = v_j\}.
\end{aligned}$$

These sets are pairwise disjoint, and since $u \in [v, a]$, we have $\mathbb{V} = A \cup B \cup C \cup D \cup E \cup F$. To simplify notation, we write $\omega(S)$ instead of $\sum_{j \in S} \omega_j$. Then we have

$$\begin{aligned}
\rho_\omega(a, v) &= \omega(C \cup D \cup E \cup F)\\
\rho_\omega(b, v) &= \omega(B \cup C \cup E)\\
\rho_\omega(a, u) &= \omega(D \cup E),\\
\rho_\omega(b, u) &= \omega(B \cup E \cup F).
\end{aligned}$$

By elementary computations,

$$\begin{aligned}
\rho_\omega(b, v) - \rho_\omega(a, v) &= \omega(B \cup C \cup E) - \omega(C \cup D \cup E \cup F)\\
&= \omega(B) - \omega(D \cup F)\\
&\leq \omega(B) - \omega(D \cup F) + 2\omega(F)\\
&= \omega(B \cup F) - \omega(D)\\
&= \omega(B \cup E \cup F) - \omega(D \cup E)\\
&= \rho_\omega(b, u) - \rho_\omega(a, u)
\end{aligned}$$

using the facts that the sets B, C, D, E and F are pairwise disjoint, and that $\omega(F) \geq 0$ by the nonnegativity of ω. Condition (19) then follows immediately. □