Towards a Logic-Based View of Some Approaches to Classification Tasks

Didier Dubois, Henri Prade

Towards a logic-based view of some approaches to classification tasks

Didier Dubois and Henri Prade

IRIT, CNRS & Univ. Paul Sabatier,
118 route de Narbonne, 31062 Toulouse Cedex 9, France
dubois, prade@irit.fr

Abstract. This paper is a plea for revisiting various existing approaches to the handling of data, for classification purposes, based on a set-theoretic view, such as version space learning, formal concept analysis, or analogical proportion-based inference, which rely on different paradigms and motivations and have been developed separately. The paper also exploits the notion of conditional object as a proper tool for modeling if-then rules. It also advocates possibility theory for handling uncertainty in such settings. It is a first, and preliminary, step towards a unified view of what these approaches contribute to machine learning.

Keywords: data, classification, version space, conditional object, if-then rule, analogical proportion, formal concept analysis, possibility theory, possibilistic logic, bipolarity, uncertainty

1 Introduction

It is an understatement to say that the current dominant paradigms in machine learning rely on neural nets and statistics; see, e.g., [1, 7]. Yet, there have been quite a number of set theoretic- or logic-based views that have considered data sets from different perspectives: we can thus (at least) mention concept learning [21, 22], formal concept analysis [17], rough sets [25], logical analysis of data [3], test theory [6], and GUHA method [19]. Still some other works, mentioned later, may be also relevant. These various paradigms can be related to logic, but have been developed independently. Strangely enough, little has been done to move towards a unified view of them.

This research note aims to be a first step in this direction. However, the result will remain modest, since we shall only outline connections between some settings, while other ones will be left aside for the moment. Moreover we shall mainly focus on Boolean data, even if some of what is said could be extended to nominal, or even numerical data. Still, we believe that it is of scientific interest to better understand the relationships between these different theoretical settings developed with various motivations and distinct paradigms, while all are starting from the same object: a set of data. In the long range, such a better understanding may contribute to some cooperation between these set theory-based views and currently popular ones, such as neural nets or statistical approaches, perhaps providing tools for explanation capabilities; see, e.g., [5] for references and a tentative survey.

The paper is organized as follows. Section 2 states and discusses the problem of assessing a class to an item, given examples and counter-examples. Section 3 presents a simple propositional logic reading of the problem. Section 4 puts the discussion in a more appropriate setting using the notion of conditional object [10], which captures the idea of a rule, better than material implication. Moreover, a rule-based reading of analogical proportion-based classification [23] is also discussed in Section 5. Section 6 briefly recalls the version space characterization of the set of possible descriptions of a class, emphasizing its bipolar nature. Section 7 advocates the interest of possibilistic logic [14] for handling uncertainty and coping with noisy data, which is a known drawback of set-theoretic approaches to data handling. Section 8 briefly surveys formal concept analysis and suggests its connection and potential relevance to classification. Section 9 mentions some other related matters and issues, pointing out lines for further research.

2 Classification problem - A general view

Let us consider m pieces of data that describe items in terms of n attributes A_j. Namely an item is represented by a vector $\boldsymbol{a^i} = (a^i_1, a^i_2, \cdots, a^i_n)$, with $i = 1, m$, together with their class $cl(\boldsymbol{a^i})$, where a^i_j denotes the value of the j-th attribute A_j for item $\boldsymbol{a^i}$, namely $A_j(\boldsymbol{a^i}) = a^i_j \in dom(A_j)$ ($dom(A_j)$ denotes the domain of attribute A_j). Each domain $dom(A_j)$ can be described using a set of propositional variables $\mathcal{V}_j$ specific to A_j, by means of logical formulas. If $|dom(A_j)| = 2$, we can let $\mathcal{V}_j = \{v_j, \neg v_j\} = dom(A_j)$.

Let $\mathcal{C} = \{cl(\boldsymbol{a^i}) | i = 1, m\}$ be a set of classes, where each object is supposed to belong to one and only one class. The classification problem amounts to predicting the class $cl(\boldsymbol{a^*}) \in \mathcal{C}$ of a new item $\boldsymbol{a^*}$ described in terms of the same attributes, on the basis of the m examples $(\boldsymbol{a^i}, cl(\boldsymbol{a^i}))$ consisting of classified objects.

There are other problems that are akin to the classification problem, with different terminology. Let us at least mention case-based decision, and diagnosis. In the first situation, we face a multiple criteria decision problem where one wants to predict the value of a new item on the basis of a collection of valued items (assuming that possible values belong to a finite scale), while in the second situation attribute values play the role of symptoms (present or not) and classes are replaced by diseases. In both situations, the m examples constitute a repertory of reference cases already experienced. This is also true in case-based reasoning, where a *solution* is to be found for a new encountered *problem* on the basis of a collection of previously solved problems for which the solution is known; however, case-based reasoning usually includes an adaptation step of the past solution selected, for a better adequacy with the new problem. Thus, ideas and methods developed in these different fields may be also of interest in a classification perspective.

Two further comments are in order here. First, for each class C, one may partition the whole set of m data in two parts : the set $\mathcal{E}$ of examples associated with this class, and the set $\mathcal{E}'$ of examples of other classes, which can be viewed as *counter-examples* for this class. The situation is pictured in Table 1 below. It highlights the fact that the whole set of items in class C is bracketed between $\mathcal{E}$ and $\overline{\mathcal{E}'}$ (where the overbar means complementation). If the table is contradiction-free, there is no item that is both in $\mathcal{E}$ and

in $\mathcal{E}'$. Second, the classification problem can be envisaged in two different manners: i) as an *induction* problem, where one wants to build a plausible description of each class; in terms of if-then rules associating sets of attribute values with a class, and then using these rules for prediction purposes ; ii) as a *transduction* problem, where the prediction is made without the help of such descriptions, but by means of direct *comparisons* of the new item with the set of the m examples.

	A_1	A_2	$\cdots$	A_n	cl	
$\mathbf{e^1}$	a_1^1	a_2^1	$\cdots$	a_n^1	C	
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	C	$\mathcal{E}$
$\mathbf{e^r}$	a_1^r	a_2^r	$\cdots$	a_n^{1r}	C	
$\mathbf{e'^1}$	$a_1'^1$	$a_2'^1$	$\cdots$	$a_n'^1$	$\overline{C}$	
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\overline{C}$	$\mathcal{E}'$
$\mathbf{e'^s}$	$a_1'^s$	$a_2'^s$	$\cdots$	$a_n'^s$	$\overline{C}$	
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	?	
$\mathbf{e^\star}$	$a_1^\star$	$a_2^\star$	$\cdots$	$a_n^\star$	?	
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	?	

Tableau 1. Contradiction-free data table

3 A simple logical reading

An elementary idea for characterizing a class C is to look for an attribute such that the subset of values taken for this attribute by the available examples of class C is *disjoint* from the subset of values taken by the examples of the other classes. If there exists at least one such attribute A_{j^*}, then one may inductively assume that belonging or not to class C, for any new item, can be predicted on the basis of its value for A_{j^*}. More generally, if a particular combination of attribute values can be encountered only for items of a class C, then a new item with this particular combination should also be put plausibly in class C. Let us now have a more systematic logical analysis of the data.

Let us consider a particular class $C \in \mathcal{C}$. Then the m items $\boldsymbol{a^i}$ can be partitioned into two subsets, the items $\boldsymbol{a^i}$ such that $cl(\boldsymbol{a^i}) = C$, and those such that $cl(\boldsymbol{a^i}) \neq C$ (we assume that $|\mathcal{C}| \geq 2$). Thus we have a set $\mathcal{E}$ of examples for C, namely $\boldsymbol{e^i} = (a_1^i, a_2^i, \cdots, a_n^i, 1) = (\boldsymbol{a^i}, 1)$, where '1' means that $cl(\boldsymbol{a^i}) = C$, and a set $\mathcal{E}'$ of counter-examples $\boldsymbol{e'^j} = (a_1'^j, a_2'^j, \cdots, a_n'^j, 0)$ where '0' means that $cl(\boldsymbol{a'^j}) \neq C$.

Let us assume that the domains $dom(A_j)$ for $j = 1, n$ are finite and denote by v_C the propositional variable associated to class C (v_C has truth-value 1 for elements of C and 0 otherwise). Using the attribute values as propositional logic symbols, an example $\boldsymbol{e^i}$ expresses the truth of the logical statement

$$a_1^i \wedge a_2^i \wedge \cdots \wedge a_n^i \rightarrow v_C$$

meaning that if it is an example, then it belongs to the class, while counter-examples $\boldsymbol{e'^j}$ are encoded by stating that the formula $a_1'^j \wedge a_2'^j \wedge \cdots \wedge a_n'^j \rightarrow \neg v_C$ is true, or equivalently

$$\models v_C \rightarrow \neg a_1'^j \vee \neg a_2'^j \vee \cdots \vee \neg a_n'^j.$$

Then any class (or concept) C that agrees with the m pieces of data is such that

$$\bigvee_{i:\boldsymbol{e^i}\in\mathcal{E}} (a_1^i \wedge a_2^i \wedge \cdots \wedge a_n^i) \models v_C \models \bigwedge_{j:\boldsymbol{e'^j}\in\mathcal{E}'} (\neg a_1'^j \vee \neg a_2'^j \vee \cdots \vee \neg a_n'^j). \qquad (1)$$

Letting $\mathcal{E}$ be the set of models of $\bigvee_i \boldsymbol{a^i}$ (the examples) and $\mathcal{E}'$ be the set of models of $\bigvee_j \boldsymbol{a'_j}$ (the counter-examples), (1) simply reads $\mathcal{E} \subseteq C \subseteq \overline{\mathcal{E}'}$ where the overbar denotes complementation. Note that the larger the number of counter-examples, the more specific the upper bound of C; the larger the number of examples, the more general the lower bound of C.

This logical expression states that if an item is identical to an example on all attributes then it is in the class, and that if an item is in the class then it should be different from all counter-examples on at least one attribute.

Let us assume Boolean attributes for simplicity, and let us suppose that $a_1^i = v_1$ is true for all the examples of class C and false for all the examples of other classes. Then it can be seen that (1) can be put under the form $v_1 \wedge L \models v_C \models v_1 \vee L'$ where L and L' are logical expressions that do not involve any propositional variable pertaining to attribute A_1. This provides a reasonable support for inducing that an item belongs to C as soon as v_1 is true for it. Such a remark can be generalized to a combination of attribute values and to nominal attributes.

Let us consider a small toy example, still sufficient for an illustration of (1) and starting the discussion.

Example 1. It is an example with two Boolean attributes, two classes (C and $\overline{C}$), two examples and a counter-example. Namely, we have $\boldsymbol{e^1} = (a_1^1, a_2^1, 1) = (1, 0, 1) = (v_1, \neg v_2, v_C)$; $\boldsymbol{e^2} = (a_1^2, a_2^2, 1) = (0, 1, 1) = (\neg v_1, v_2, v_C)$; $\boldsymbol{e'^1} = (a_1'^1, a_2'^1, 0) = (0, 0, 0) = (\neg v_1, \neg v_2, \neg v_C)$.

We can easily see that $(v_1 \wedge \neg v_2) \vee (\neg v_1 \wedge v_2) \models v_C \models v_1 \vee v_2$, i.e., we have $v_1 \dot{\vee} v_2 \models v_C \models v_1 \vee v_2$, where $\dot{\vee}$ stands for exclusive *or*. Indeed depending on whether $(1, 1)$ is an example or a counter-example, the class C will be described by $v_1 \vee v_2$, or by $v_1 \dot{\vee} v_2$ respectively.

Note that in the absence of any further information or principle, the two options for assessing a class to $(1, 1)$ on the basis of $\boldsymbol{e^1}$, $\boldsymbol{e^2}$ and $\boldsymbol{e'^1}$, are equally possible here. □

Observe that if the bracketing of C in (1) is consistent, the conjunction of the lower bound expression and the upper bound expression yields the lower bound. But in case of an item which would appear both as an example and as a counter-example for C (noisy data), this conjunction would not be a contradiction, as we might expect, in general, as shown by the example below.

Example 2. Assume we have $\boldsymbol{e^1} = (1, 0, 1)$; $\boldsymbol{e^2} = (1, 1, 1)$; $\boldsymbol{e'^1} = (1, 1, 0)$. The classes $\mathcal{E}$ and $\mathcal{E}'$ overlap since $\boldsymbol{e^2}$ and $\boldsymbol{e'^1}$ are the same item, classified differently. As a consequence we do not have that $\mathcal{E} \subseteq \overline{\mathcal{E}'}$. So equation (1) is not valid : we do not have that

$(v_1 \wedge \neg v_2) \vee (v_1 \wedge v_2) \models C \models \neg v_1 \vee \neg v_2$, i.e., $v_1 \models C \models \neg v_1 \vee \neg v_2$ is wrong even if $v_1 \wedge (\neg v_1 \vee \neg v_2) = v_1 \wedge \neg v_2 \neq \bot$. □

A more appropriate treatment of inconsistency will be proposed in the next section.

The two expressions bracketing C in (1) have a graded counterpart, proposed in [15], for assessing how satisfactory an item is, given a set of examples and a set of counter-examples supposed to describe what we are looking for. Then an item is all the better ranked as it is similar to at least one example on all important attributes, and that it is dissimilar to all counter-examples on at least one important attribute (where similarity, dissimilarity, and importance are matters of degrees). However, this ranking problem is somewhat different from the classification problem where each item should be assigned to a class. Here if an item is both close to an example and to a counter-example, it has a poor evaluation, just as it would be if it is close to a counter-example only.

Note that if one considers examples only, the graded counterpart amounts to searching for items that are similar to examples. In terms of classification, it means to look for the pieces of data that are as much similar (on all attributes) as possible to the item for which one wants to predict the class, and to assess the class shared by the majority of these similar data. This is the k-nearest neighbor method. This is also very close to fuzzy case-based reasoning and instance-based learning [20, 9].

4 Conditional objects and rules

A conditional object $b|a$, where a, b are propositions, is a three-valued entity, which is *true* if $a \wedge b$ is true; *false* if $a \wedge \neg b$ is true; *inapplicable* if a is false; see, e.g., [10]. It can be thought as the rule 'if a then b'. Indeed, the rule can be fired only if a is true; the examples of this rule are such that $a \wedge b$ is true, while its counter-examples are such that $a \wedge \neg b$ is true. This view of conditionals dates back to De Finetti 's works in the 1930's.

An (associative) quasi-conjunction & can be defined for conditional objects:

$$b|a \;\&\; d|c = (a \to b) \wedge (c \to d)|(a \vee c)$$

where $\to$ denotes the material implication. It fits with the intuition that a set of rules can be fired as soon as at least one rule can be fired, and when a rule is fired, the rule behaves like material implication. Moreover, entailment between conditional objects is defined by $b|a \vDash d|c$ iff $a \wedge b \vDash c \wedge d$ and $c \wedge \neg d \vDash a \wedge \neg b$, which expresses that the examples of rule 'if a then b' are examples of rule 'if c then d', and the counter-examples of rule 'if c then d' are counter-examples of rule 'if a then b'. It can be checked that $b|a = (a \wedge b)|a = (a \to b)|a$ since these three conditional objects have the same examples and the same counter-examples. It can be also shown that $a \wedge b|\top \vDash b|a \vDash a \to b|\top$ (where $\top$ denotes tautology), thus highlighting the fact that $b|a$ is bracketed by the conjunction $a \wedge b$ and the material implication $a \to b$.

Let us revisit expression (1) in this setting. For an example $\boldsymbol{e} = (\boldsymbol{a}, 1)$, and a counter-example $\boldsymbol{e'} = (\boldsymbol{a'}, 0)$ with respect to a class C, it leads to consider the conditional objects $v_C|\boldsymbol{a}$ and $\neg v_C|\boldsymbol{a'}$ respectively (if it is an example we are in the class, otherwise not).

For a collection of examples we have

$$(v_C|\boldsymbol{a^1}) \,\&\, \cdots \,\&\, (v_C|\boldsymbol{a^r}) = ((\boldsymbol{a^1} \vee \cdots \vee \boldsymbol{a^r}) \rightarrow v_C)|(\boldsymbol{a^1} \vee \cdots \vee \boldsymbol{a^r}) \\ = v_C|(\boldsymbol{a^1} \vee \cdots \vee \boldsymbol{a^r})$$

Similarly, we have

$$(\neg v_C|\boldsymbol{a'^1}) \,\&\, \cdots \,\&\, (\neg v_C|\boldsymbol{a'^s}) = ((\boldsymbol{a'^1} \vee \cdots \vee \boldsymbol{a'^s}) \rightarrow \neg v_C)|(\boldsymbol{a'^1} \vee \cdots \vee \boldsymbol{a'^s}) \\ = \neg v_C|(\boldsymbol{a'^1} \vee \cdots \vee \boldsymbol{a'^s})$$

Letting $\phi_E = \bigvee_{i=1}^{r} \boldsymbol{a^i}$ and $\phi_{E'} = \bigvee_{j=1}^{s} \boldsymbol{a'^j}$, we can join the two conditional expressions:

$$(v_C|\phi_E) \,\&\, (\neg v_C|\phi_{E'}) = (\phi_E \rightarrow v_C) \wedge (\phi_{E'} \rightarrow \neg v_C)|(\phi_E \vee \phi_{E'})$$

where

$$(\phi_E \wedge v_C) \vee (\phi_{E'} \wedge \neg v_C)|\top \vDash (v_C|\phi_E) \,\&\, (\neg v_C|\phi_{E'}) \vDash (\phi_E \rightarrow v_C) \wedge (\phi_{E'} \rightarrow \neg v_C)|\top$$

A set of conditional objects K is said to be consistent if and only if for no subset $S \subseteq K$ does the quasi-conjunction $Q(S)$ of the conditional objects in S entail a conditional contradiction [10]. Contrary to material implication, the use of three-valued conditionals reveals the presence of contradictions in the data.

Example 3. (Example 2 continued) The data are $\boldsymbol{e^1} = (1, 0, 1)$; $\boldsymbol{e^2} = (1, 1, 1)$; $\boldsymbol{e'^1} = (1, 1, 0)$. In terms of conditional objects, considering the subset $\{\boldsymbol{e^2}, \boldsymbol{e'^1}\}$, we have

$$v_C|(v_1 \wedge v_2) \,\&\, \neg v_C|(v_1 \wedge v_2) = (v_1 \wedge v_2) \rightarrow (v_C \wedge \neg v_C)|(v_1 \wedge v_2) \\ = (v_C \wedge \neg v_C)|(v_1 \wedge v_2) = \bot|v_1 \wedge v_2,$$

which is a conditional contradiction. □

5 Analogical proportion-based transduction

Apart from the k-nearest neighbor method, there is another transduction approach to the classification problem which applies to Boolean, nominal and numerical attribute values [4]. For simplicity here, we only consider Boolean attributes. It relies on the notion of analogical proportion [23]. Analogical proportions are statements of the form "a is to b as c is to d", often denoted by $a : b :: c : d$, which express that "a differs from b as c differs from d and b differs from a as d differs from c". This statement can be encoded into a Boolean logical expression which is true only for the 6 following assignments $(0, 0, 0, 0)$, $(1, 1, 1, 1)$, $(1, 0, 1, 0)$, $(0, 1, 0, 1)$, $(1, 1, 0, 0)$, and $(0, 0, 1, 1)$ for (a, b, c, d). Boolean Analogical proportions straightforwardly extend to vectors of attributes values such as $\boldsymbol{a} = (a_1, ..., a_n)$, by stating $\boldsymbol{a} : \boldsymbol{b} :: \boldsymbol{c} : \boldsymbol{d}$ iff $\forall i \in [1, n],\ a_i : b_i :: c_i : d_i$. The basic analogical inference pattern, is then

$$\frac{\forall i \in \{1, ..., p\},\ \ a_i : b_i :: c_i : d_i \text{ holds}}{\forall j \in \{p+1, ..., n\},\ \ a_j : b_j :: c_j : d_j \text{ holds}}$$

Thus analogical reasoning amounts to finding completely informed triples $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})$ appropriate for inferring the missing value(s) in $\boldsymbol{d}$. When there exist several suitable triples, possibly leading to distinct conclusions, one may use a majority vote for concluding. This inference method is an extrapolation, which applies to classification (then the class $cl(\boldsymbol{x})$ is the unique solution, when it exists, such as $cl(\boldsymbol{a}) : cl(\boldsymbol{b}) :: cl(\boldsymbol{c}) : cl(\boldsymbol{x})$ holds).

Let us examine more carefully how it works. The inference in fact takes items pair by pair, and then puts two pairs in parallel. Let us first consider the case where three items belong to the same class ; the fourth item is the one, the class of which one wants to predict (denoted by 1 in the following). Considering a pair of items $\mathbf{a^i}$ and $\mathbf{a^j}$. There are attributes for which the two items are equal and attributes for which they differ. For simplicity, we assume that they differ only on the first attribute (the method easily extend to more attributes). So we have $\mathbf{e^i} = (a_1^i, a_2^i, \cdots, a_n^i, 1)$ and $(\mathbf{e^j} = a_1^j, a_2^j, \cdots, a_n^j, 1)$ with $a_1^j = \neg a_1^i$ and $a_t^j = a_t^i = v_t$ for $t = 2, n$. This means that the change from a_1^i to a_1^j in context $(v_2, \cdots, v_n)$ does not change the class. Assume we have now another pair $\mathbf{e^k} = (v_1, a_2^k, \cdots, a_n^k, 1)$ and $\mathbf{e}^\star = (\neg v_1, a_2^\star, \cdots, a_n^\star, ?)$ involving the item for we which we have to predict the class and exhibiting the same change on attribute A_1 and being equal elsewhere, i.e., we have $a_t^k = a_t^\star = v_t^\sharp$ for $t = 2, n$). Putting the two pairs in parallel, we obtain the following pattern

$(v_1, v_2, \cdots, v_n, 1)$
$(\neg v_1, v_2, \cdots, v_n, 1)$
$(v_1, v_2^\sharp, \cdots, v_n^\sharp, 1)$
$(\neg v_1, v_2^\sharp, \cdots, v_n^\sharp, ?)$

It is not difficult to check that $\mathbf{a^i}$, $\mathbf{a^j}$, $\mathbf{a^k}$ and $\mathbf{a}^\star$ are in analogical proportion for each attribute. So $\mathbf{a^i} : \mathbf{a^j} :: \mathbf{a^k} : \mathbf{a}^\star$ holds. The solution of $1 : 1 :: 1 :?$ is obviously $? = 1$, so the prediction is $cl(\mathbf{a}^\star) = 1$. This conclusion is thus based on the idea that since the change from a_1^i to a_1^j in context $(v_2, \cdots, v_n)$ does not change the class, it is the same in the other context $(v_2^\sharp, \cdots, v_n^\sharp)$.

The case where $\mathbf{e^i}$ and $\mathbf{e^k}$ belong to class C while $\mathbf{e^j}$ is in $\neg C$ leads to another analogical pattern, where the change from a_1^i to a_1^j now changes the class in context $(v_2, \cdots, v_n)$. The pattern is

$(v_1, v_2, \cdots, v_n, 1)$
$(\neg v_1, v_2, \cdots, v_n, 0)$
$(v_1, v_2^\sharp, \cdots, v_n^\sharp, 1)$
$(\neg v_1, v_2^\sharp, \cdots, v_n^\sharp, ?)$

The conclusion is now $? = 0$, i.e., $\mathbf{a}^\star$ is not in C. This thus implements the idea that the change from a_1^i to a_1^j that changes the class in context $(v_2, \cdots, v_n)$, leads also to the same change in context $(v_2^\sharp, \cdots, v_n^\sharp)$.

It has been theoretically established that analogical classifiers *always* yield exact prediction for Boolean affine functions describing the class (which includes x-or functions), and only for them [8]. Still a majority vote among the predicting triples often yields the right prediction in other situations [4].

Let us see how it works on Example 1 and variants.

Example 4. In Example 1 we have: $\boldsymbol{e^1} = (1, 0, 1)$; $\boldsymbol{e^2} = (0, 1, 1)$; $\boldsymbol{e'^1} = (0, 0, 0)$. We can check that there is no analogical prediction in this case for $(1, 1, ?)$. Indeed, whatever the way we order the three vectors, either we get the 4-tuple $(1, 0, 0, 1)$ on one component, which is not a pattern making true an analogical proportion, or the equation $0 : 1 :: 1 :?$ which has no solution. So analogy remains neutral in this case.

However, in the situation where would have $\boldsymbol{e^1} = (1, 0, 1)$; $\boldsymbol{e^2} = (1, 1, 1)$; $\boldsymbol{e'^1} = (0, 1, 0)$. Taking the triple $(\boldsymbol{e^2}, \boldsymbol{e^1}, \boldsymbol{e'^1})$, we can check that $(1, 1) : (1, 0) :: (0, 1) : (0, 0)$ holds on each of the two vector components. The solution of the equation $1 : 1 :: 0 :?$ is $? = 0$, which is the analogical prediction for $(0, 0, ?)$.

Similarly, in the case $\boldsymbol{e^1} = (1, 0, 1)$, $\boldsymbol{e^2} = (1, 1, 1)$ and $\boldsymbol{e^3} = (0, 1, 1)$, we would obtain $? = 1$ for $(0, 0, ?)$ as expected, using triple $(\boldsymbol{e^2}, \boldsymbol{e^1}, \boldsymbol{e^3})$.

□

It is clear that the role of analogical reasoning here is to complete the data set with new examples or counter-examples obtained by transduction, assuming analogical inference patterns are valid in the case under study. It may be a first step prior to the induction of a classification model.

6 Concept learning, version space and logic

The version space setting, as proposed by Mitchell [21, 22], offers an elegant elimination procedure, exploiting examples and counter-examples of a class, then called "concept", for restricting the hypotheses space and providing an approach to rule learning.

Let us recall the approach using a simple example, with 3 attributes: $A_1 =$ *Sky* (with possible values `Sunny`, `Cloudy`, and `Rainy`), $A_2 =$ *Air Temp* (with values `Warm` and `Cold`), and $A_3 =$ *Humidity* (with values `Normal` and `High`). The problem is to learn the concept of $C =$ *Nice Day* on the basis of examples and counter-examples. This means finding all hypotheses h, such that the implication $h \rightarrow v_C$ is compatible with the examples and the counter-examples.

Each hypothesis is described by a conjunction of constraints on the attributes, here *Sky*, *Air Temp*, and *Humidity*. Constraints may be $?$ (any value is acceptable), $\emptyset$ (no value is acceptable), a specific value, or a disjunction thereof. The target concept C, here *Nice Day*, is supposed to be represented by a disjunction of hypotheses (there may exist different h and h' such that $h \rightarrow v_C$ and $h' \rightarrow v_C$). Descriptions of examples or counter-examples can be *ordered* according to their generality / specificity. Thus, the following descriptions are ordered according to decreasing generality: $<?, ?, ?>$, $<$`Sunny` $\vee$ `Cloudy`$, ?, ?>$, $<$`Sunny`$, ?, ?>$, $<$`Sunny`$, ?,$ `Normal`$>$, $<\emptyset, \emptyset, \emptyset>$.

The version space is represented by its most general and least general members. The so-called general boundary G is the set of maximally general members of the hypothesis space that are consistent with the data. The specific boundary S is the set of maximally specific members of the hypothesis space that are consistent with the data. G and S are initialized as $G =<?, ?, ?>$ and $S =<\emptyset, \emptyset, \emptyset>$ (for 3 attributes as in the example).

The procedure amounts to finding a maximally specific hypothesis which covers the positive examples. Suppose we have two examples of *Nice Day*:

Ex1. <`Sunny`, `Warm`, `Normal`> , *Ex2.* <`Sunny`, `Warm`, `High`>.

Then, taking into account *Ex1*, S is updated to $S_1 =<\texttt{Sunny}, \texttt{Warm}, \texttt{Normal}>$.

Adding *Ex2*, S is improved into $S_2 =<\texttt{Sunny}, \texttt{Warm}, ? >$, which corresponds to the disjunction of *Ex1* and *Ex2*. The positive training examples force the S boundary of the version space to become increasingly general (S_2 is more general than S_1).

Although the version space approach was not cast in a logical setting, it is perfectly compatible with the logical encoding (1). Indeed here we have two examples of the form (v_1, v_2, v_3) and $(v_1, v_2, \neg v_3)$ (with $v_1 = \texttt{Sunny}; v_2 = \texttt{Warm}; v_3 = \texttt{Normal}, \neg v_3 = \texttt{High}$). A tuple of values such that $< v, v', v'' >$ is to be understood as the conjunction $v \wedge v' \wedge v$". So we obtain $(v_1 \wedge v_2 \wedge v_3) \vee (v_1 \wedge v_2 \wedge \neg v_3) \rightarrow v_C$. It corresponds to the left part of Equation (1) for $n = 3$ and $|\mathcal{E}| = 2$, which yields $(v_1 \wedge v_2) \wedge (v_3 \vee \neg v_3) \rightarrow v_C$, i.e., $(v_1 \wedge v_2) \rightarrow v_C$. So the more positive examples we have, the more general the lower bound of C in (1) (the set of models of a disjunction is larger than the set of models of each of its components). This lower bound, here $v_1 \wedge v_2$, is a maximally specific hypothesis h.

Negative examples play a complementary role. They force the G boundary to become increasingly specific. Consider we have the following counter-example for *Nice Day*: *cEx3*. $<\texttt{Rainy}, \texttt{Cold}, \texttt{High}>$

The hypothesis in the G boundary must be specialized until it correctly classifies the new negative example. There are several alternative minimally more specific hypotheses. Indeed, the 3 attributes can be specialized for avoiding to cover *cEx3* by being $\neg\texttt{Rainy}$, or being $\neg\texttt{Cold}$, or being $\neg\texttt{High}$. This exactly corresponds to Equation (1), which here gives $v_C \rightarrow \neg\texttt{Rainy} \vee \neg\texttt{Cold} \vee \neg\texttt{High}$, i.e., $v_C \rightarrow \texttt{Sunny} \vee \texttt{Cloudy} \vee \texttt{Warm} \vee \texttt{Normal}$.

The elements of this disjunction correspond to maximally general potential hypotheses. But in fact we have only two new hypotheses in G: $<\texttt{Sunny}, ?, ? >$ and $<?, \texttt{Warm}, ? >$, as explained now. Indeed, the hypothesis $h = (?, ?, \texttt{Normal})$ is not included in G, although it is a minimal specialization of G that correctly labels *cEx3* as a negative example. This is because example *Ex2* whose attribute value for A_3 is $\texttt{High}$, disagrees with the implication $\texttt{Normal} \rightarrow v_C$. So, hypothesis $<?, ?, \texttt{Normal}>$ is excluded. Similarly, examples *Ex1* and *Ex2* (for which the attribute value for A_1 is $\texttt{Sunny}$) disagree with implication $\texttt{Cloudy} \rightarrow v_C$. This kind of elimination applies in Equation (1) as well. Indeed the expression $v \wedge L \models \neg v \vee L'$ can be simplified into $v \wedge L \models L'$.

We thus obtain upper and lower bounds from *Ex1*, *Ex2*, and *cEx3*

$$S_3: <\texttt{Sunny}, \texttt{Warm}, ? > \quad G_3: \{<\texttt{Sunny}, ?, ? >, <?, \texttt{Warm}, ? >\}.$$

where $\{< v_1, v_1', v_1'' >, < v_2, v_2', v_2'' >\}$ logically reads $(v_1 \wedge v_1' \wedge v_1'') \vee (v_2 \wedge v_2' \wedge v_2'')$ ($?$ stands for $\top$). The S boundary of the version space thus summarizes the previously encountered positive examples. Any hypothesis more general than S will, by definition, cover any example that S covers and thus will cover any past positive example. In a dual fashion, the G boundary summarizes the information from previously encountered *negative* examples. Any hypothesis more specific than G is assured to be consistent with past negative examples. The set of all the hypotheses between S and G has a lattice structure. This in full agreement with Equation (1). The approach provides an iterative procedure that takes advantage of the examples and counter-examples progressively.

Thus, the general procedure for obtaining the bounds of the version space are as follows. If $\mathbf{e}$ is a positive example, i) remove from G any hypothesis inconsistent with $\mathbf{e}$; ii) substitute in S any minimal generalization h consistent with $\mathbf{e}$. If $\mathbf{e}$ is a negative example, i) remove from S any hypothesis inconsistent with $\mathbf{e}$; ii) substitute in G any minimal specialization h consistent with $\mathbf{e}$.

7 Towards a possibilistic variant of the version space

The main drawback of the version space approach is its sensitivity to noise. Indeed each example and each counter-example influence the result. In [16], the authors use rough set approximations to cope with this problem.

Here we make another suggestion using possibility theory. The idea is to associate each example and each counter-example with a certainty level, as in possibilistic logic (see, e.g., [14]) in order to express to what extent we consider it is certain that the corresponding piece of information is true (rather than false). This certainty level expresses our confidence in the piece of data as being exact. It can reflect the confidence we have in the source that provided it, or be the result of an analysis or filtering of the data that disqualifies outliers. In that respect we should remember that one semantics of possibility theory is in terms of (dis)similarity [26].

In other words, we have a multi-tiered set of examples and a multi-tiered set of counter-examples. So, given some certainty level α, considering all examples and all counter-examples whose certainty is above or equal to α yields a regular version space with classical bounds. Thus, for each α, it gives birth to a bounded set of hypotheses to which α can be associated. We have thus a natural basis for rank-ordering hypotheses. The smaller α, the larger the numbers of examples and counter-examples taken into account, and the tighter the bounds.

This can be illustrated on the example of the previous section.

Example 5. Examples and counter-examples now come with certainty weights. Assume we have *Ex1*: (<Sunny, Warm, Normal>, 1); *cEx3*: (<Rainy, Cold, High>, α); *Ex2*: (<Sunny, Warm, High>, β), with $1 > \alpha > \beta$.

So, we obtain a layered version of the upper and lower bounds of the version space:

- at level 1, we have $G_1 =$<?, ?, ? > and $S_1 =$<Sunny, Warm, Normal>.
- at level α, we have $G_\alpha = \{$<Sunny, ?, ? >, <Cloudy, ?, ? >, <?, Warm, ? >$\}$ and $S_\alpha =$<Sunny, Warm, Normal>.
- at level β, we have $G_\beta = \{$<Sunny, ?,?>, <?, Warm,?>$\}$ and $S_\beta =$<Sunny, Warm, ? >.

□

The above syntactic view is simpler than the semantic one presented in [24] where the paper starts with a pair of possibility distributions over hypotheses, respectively induced by the examples and by the counter-examples.

8 Formal concept analysis

Formal concept analysis [17] is another setting where association rules between attributes can be extracted from a formal context $R \subseteq X \times Y$, which is nothing but a relation linking items in X with properties in Y. It provides a theoretical basis for data mining. Tableau 1 can be viewed as a context, restricting to rows $\mathcal{E} \cup \mathcal{E}'$ and considering the class of examples as just another attribute.

Let Rx and $R^{-1}y$ respectively denote the set of properties possessed by item x and the set of items having property y. Let $E \subseteq X$ and $A \subseteq Y$. The set of items having all properties in A is given by $A^{\downarrow} = \{x \mid A \subseteq Rx\}$ and the set of properties possessed by all items in E is given by $E^{\uparrow} = \{y \mid E \subseteq R^{-1}y\}$. A formal concept is then defined as a pair (E, A) such that $A^{\downarrow} = E$ and $E^{\uparrow} = A$ where E and A provides the extent and the intent of the formal concept respectively. Then, it can be shown that $E \times A \subseteq R$, and is maximal with respect to set inclusion, i.e., (E, A) defines a maximal rectangle in the formal context.

Let A and B be two subsets of Y. Then R satisfies the attribute implication $A \Rightarrow B$ if for every $x \in X$, such that $x \in A^{\downarrow}$, then $x \in B^{\downarrow}$. Formal concept analysis is not primarily oriented towards concept learning, but towards mining attribute implications (i.e., association rules). However, it might be interesting to consider formal contexts where Y also contains the names of classes, i.e., $\mathcal{C} \subseteq Y$. Then being able to find attribute implications of the form $A \Rightarrow C$ where $A \cap \mathcal{C} = \emptyset$ and $C \subseteq \mathcal{C}$, would be of a particular interest, especially if C is a singleton.

The rectangular nature of formal concepts expresses a form of convexity, which fits well with the ideas of Gärdenfors about conceptual spaces [18]. Moreover, using also operators other than $^{\downarrow}$ and $^{\uparrow}$ (see [12]) help characterizing independent sub-contexts and other noticeable structures. Formal concept analysis can be also related to the idea of clustering [13], where clusters are unions of overlapping concepts in independent sub-contexts. The idea of approximate concepts, i.e., rectangles with "holes", suggests a convexity-based completion principle, which might be useful in a classification perspective.

9 Concluding remarks

This paper is clearly a preliminary step toward a unified, logical, study of set theory-based approaches in data management. It is preliminary in at least two respects: several of these approaches have been only cited in the introduction, while the others have been only briefly discussed. All these theoretical settings start with a Boolean table in the simplest case, and many of them extend to nominal, and possibly to numerical data. Still they have been motivated by different concerns such as describing a concept, predicting a class, or mining rules. Due to their set theory-based nature, they can be considered from a logical point of view, and a number of issues are common, such that handling incomplete information, missing values, inconsistent information, or non applicable attributes.

In a logical setting, the handling of uncertainty can be conveniently handled using possibility theory and possibilistic logic [14]. We have suggested above how it can

be applied to concept learning and how it may take into account uncertain pieces of data. Possibilistic logic can also handle default rules that can be obtained from Boolean data by looking for suitable probability distributions [2]; such rules provide useful summaries of data. The possible uses of possibilistic logic in data management is a general topic for further investigation.

Acknowledgements

The authors acknowledge a partial support of ANR-11-LABX-0040-CIMI (Centre International de Mathématiques et d'Informatique) within the program ANR-11-IDEX-0002-02, project ISIPA ("Intégrales de Sugeno, Interpolation, Proportions Analogiques").

References

1. Y. S. Abu-Mostafa, M. Magdon-Ismail, H.-T. Lin. Learning from data. A short course. AML-book.com, 2012.
2. S. Benferhat, D. Dubois, S. Lagrue, H. Prade. A big-stepped probability approach for discovering default rules. Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems,11(Supplement-1), 1-14, 2003.
3. E. Boros, Y. Crama, P. L. Hammer, T. Ibaraki, A. Kogan, and K. Makino. Logical analysis of data: classification with justification. Annals OR, 188(1): 33-61, 2011.
4. M. Bounhas, H. Prade, G. Richard. Analogy-based classifiers for nominal or numerical data. Int. J. Approx. Reasoning, 91: 36-55, 2017.
5. Z. Bouraoui, A. Cornuéjols, T. Denoeux, S. Destercke, D. Dubois, R. Guillaume, J. Marques-Silva, J. Mengin, H. Prade, S. Schockaert, M. Serrurier, C. Vrain. From shallow to deep interactions between knowledge representation, reasoning and machine learning (Kay R. Amel group). CoRR abs/1912.06612, 2019.
6. I. Chikalov, V. V. Lozin, I. Lozina, M. Moshkov, H. S. Nguyen, A. Skowron, and B. Zielosko. Three Approaches to Data Analysis - Test Theory, Rough Sets and Logical Analysis of Data, volume 41 of Intelligent Systems Reference Library. Springer, 2013.
7. A. Cornuejols, F. Koriche, and R. Nock. Statistical computational learning. In P. Marquis, O. Papini, and H. Prade, editors, A Guided Tour of Artificial Intelligence Research. Vol. 1 Knowledge Representation, Reasoning and Learning, 341-388. Springer-Verlag, 2020.
8. M. Couceiro, N. Hug, H. Prade, G. Richard. Analogy-preserving functions: A way to extend Boolean samples. Proc. IJCAI'17, Stockholm, 1575-1581, 2017.
9. D. Dubois, E. Hüllermeier, H. Prade. Fuzzy methods for case-based recommendation and decision support. J. Intell. Inf. Syst. 27(2): 95-115 (2006)
10. D. Dubois, H. Prade. Conditional objects as nonmonotonic consequence relationships. IEEE Trans. on Systems, Man and Cybernetics, 24(12), 1724-1740, 1994.
11. D. Dubois, H. Prade. Fuzzy relation equations and causal reasoning. Fuzzy Sets and Systems 75(2): 119-134, 1995.
12. D. Dubois, H. Prade. Possibility theory and formal concept analysis: Characterizing independent sub-contexts. Fuzzy Sets and Systems 196: 4-16 (2012).
13. D. Dubois, H. Prade. Bridging gaps between several forms of granular computing. Granular Computing 1(2), 115-126, 2016.
14. D. Dubois, H. Prade. Possibilistic logic: From certainty-qualified statements to two-tiered logics - A prospective survey. JELIA 2019: 3-20.

15. D. Dubois, H. Prade, F. Sèdes. Fuzzy logic techniques in multimedia database queyring: a preliminary investigation of the potentials. IEEE Trans. on Knowledge and Data Engineering 13(3): 383-392, 2001.
16. V. Dubois, M. Quafafou. Concept learning with approximation: Rough version spaces. Proc. 3rd Int. Conf. on Rough Sets and Current Trends in Computing (RSCTC'02), (J. J. Alpigini, J. F. Peters, J. Skowronek, N. Zhong, eds.), Malvern, PA, USA, Oct. 14-16, Springer, LNCS 2475, 239-246, 2002.
17. B. Ganter, R. Wille. Formal Concept Analysis: Mathematical Foundations. Springer-Verlag, 1998.
18. P. Gärdenfors. Conceptual Spaces. The Geometry of Thought. MIT Press, 2000.
19. P. Hájek and P. Havránek. Mechanising Hypothesis Formation - Mathematical Foundations for a General Theory. Springer Verlag, 1978.
20. E. Hüllermeier, D. Dubois, H. Prade. Model adaptation in possibilistic instance-based reasoning. IEEE Trans. Fuzzy Systems 10(3): 333-339, 2002.
21. T. M. Mitchell. Version spaces: A candidate elimination approach to rule learning. IJCAI 1977: 305-310.
22. T. M. Mitchell. Version spaces: An approach to concept learning. PhD thesis, Stanford University, 1979.
23. H. Prade, G. Richard. Analogical proportions and analogical reasoning - An introduction. ICCBR 2017: 16-32.
24. H. Prade, M. Serrurier. Bipolar version space learning. Int. J. Intell. Syst. 23(10): 1135-1152 (2008).
25. Z. Pawlak. Rough Sets. Theoretical Aspects of. Reasoning about Data. Kluwer Acad. Publ., Dordrecht, 1991.
26. T. Sudkamp. Similarity and the measurement of possibility. Actes Rencontres Francophones sur la Logique Floue et ses Applications (Montpellier, France), Toulouse: Cépaduès Editions, 13-26, 2002.