Parameter-free classification in multi-class imbalanced data sets

Loic Cerf, Dominique Gay, Nazha Selmaoui-Folcher, Bruno Crémilleux, Jean-François Boulicaut

Parameter-free classification in multi-class imbalanced data sets

Loïc Cerf [a,*], Dominique Gay [b], Nazha Selmaoui-Folcher [c], Bruno Crémilleux [d], Jean-François Boulicaut [e]

[a] Department of Computer Science, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil
[b] Orange Labs, 2 avenue Pierre Marzin, 22307 Lannion, France
[c] PPME EA3325, University of New-Caledonia, BP R4 98851, Nouméa, New Caledonia
[d] GREYC CNRS UMR6072, University of Caen, Caen, France
[e] Université de Lyon, CNRS, INRIA, INSA-Lyon, LIRIS, UMR5205, F-69621 Villeurbanne, France

A B S T R A C T

Many applications deal with classification in multi-class imbalanced contexts. In such difficult situations, classical CBA-like approaches (Classification Based on Association rules) show their limits. Most CBA-like methods actually are One-Vs-All approaches (OVA), i.e., the selected classification rules are relevant for one class and irrelevant for the *union* of the other classes. In this paper, we point out recurrent problems encountered by OVA approaches applied to multi-class imbalanced data sets (e.g., improper bias towards majority classes, conflicting rules). That is why we propose a new One-Versus-Each (OVE) framework. In this framework, a rule has to be relevant for one class and irrelevant for *every* other class taken separately. Our approach, called fitcare, is empirically validated on various benchmark data sets and our theoretical findings are confirmed.

Keywords:
Classification
Association rules
Multi-class context
Imbalanced data set
One-Versus-Each framework

1. Introduction

Among the various paradigms for supervised classification, associative classification has generated much interest in the data mining community. Mining association rules [1] concluding on a class and that furthermore satisfy some relevance constraints can provide good classification rules. In particular, their left-hand sides, i.e., conjunctions of Boolean attributes, can advantageously replace the original attributes for classification purpose. The publication of the CBA algorithm [2] has opened up this area of research. To learn a classifier from a set of training objects described with Boolean attributes, CBA first extracts every classification rule having both a frequency and a confidence greater than two user-defined thresholds. Then, it selects a subset of these rules based on coverage and/or redundancy considerations. This subset is used for prediction, i.e., to classify new objects. Since this pioneering work, many CBA-like methods (e.g., [3–6]) have proposed improvements to the selection of the classification rules and/or to their combination in a classifier. We believe their wide use is mainly explained by the ability to "understand" the classification model, i.e., the analyst can present the rules that trigger the classification of an object which are easily interpretable.

Nevertheless, these methods are often inaccurate for the minority classes. By definition, a minority class contains significantly less objects than the other class(es). According to [7], "*the class imbalance problem is pervasive and ubiquitous, causing trouble to a large segment of the data mining community*". A frequency-confidence approach, such as CBA, ignores the class distribution during the initial selection of the classification rules. These rules, with high confidences, may also be negatively correlated, i.e., the conjunction of attributes they involve may better represent another (minority) class than the one they conclude on. In fact, it has been shown that a frequency-confidence approach (i) is not suitable for statistically significant rule mining [8] and (ii) is biased

* Corresponding author.
E-mail address: lcerf@dcc.ufmg.br (L. Cerf).

towards the majority class in imbalanced data sets [9]. This bias is also the reason for poor accuracy results, such as true positive rates, in the minority class(es). To overcome this weakness, the authors of [10,9] suggest a framework based on a correlation measure. Although the correlation measure takes into account the class distribution, their approach is not perfect either. Indeed, in a context with strictly more than two classes, two classification rules involving the same conjunction of attributes but concluding on different classes may both be positively correlated. In other terms, conflicting rules can be selected.

The common fundamental issue affecting these proposals is that they are OVA (One-Versus-All) methods. That is to say, the classification problem, into p classes, is split into p two-class classification problems (every class versus the other classes altogether). In this way, the OVA approaches look for rules that are relevant for one class and irrelevant for the union of the other classes. Notice also that the numerous classifiers based on emerging patterns [11] (see, e.g., [12] for a survey) follow as well the OVA principle, thus suffer from the same problems. The OVO (One-Versus-One) framework, which selects classification rules after successive restrictions of the training data set to every pair of classes, is not a solution either: one ignored class may perfectly satisfies a rule selected for one of the classes in the pair. The next section highlights the characteristics of these frameworks and more formally details their inherent problems with imbalanced classes and, therefore, motivates our proposal for a new classification framework, namely the OVE (One-Versus-Each) framework. A key idea of the OVE framework is to take into account the distribution of the objects supporting the classification rules *in each class*, then providing rules relevant for one class and irrelevant for *every* other class taken separately. We make the following contributions.

- We propose a new classification framework called OVE (One-Versus-Each). In this framework, the repartition, in the different classes, of the "errors" committed by a classification rule is considered. In this way, classification rules are selected based on their relevance in the predicted class and their irrelevance in *every* other class.
- We implement this framework with an algorithm selecting the classification rules based on their frequencies in every class. The bodies of these rules better represent the objects in the class they conclude on than the objects in any other class. The definition of the rules allows their efficient extraction (by search space pruning). Two simple constraints on the technique guarantee that no rule conflict occurs. The algorithm is parametrized with a matrix whose number of values is quadratic in the number of classes. We design and implement a constrained hill-climbing technique to automatically learn the parameter matrix. A performance study and an in-depth comparison with several classification methods on many data sets are conducted to evaluate our approach.

The rest of the paper is organized as follows. The next section formally sets the context of our work, highlights current issues and thus motivates our proposal for a new classification framework. Section 3 formally specifies the classifiers we propose to learn and how simple constraints on its parameters guarantee that it is conflict-free. Section 4 details the algorithm `fitcare` that computes such classifiers and learns the parameters of the model. We report the experimental validation of `fitcare` in Section 5. We discuss further related work in Section 6. Finally, Section 7 briefly concludes.

2. Context & motivations

This section explains the need for a new framework for classification with imbalanced classes. We start by giving preliminary definitions. Let $\mathcal{T}$ be a set of transactions (or objects or examples) and $\mathcal{I}$ a set of Boolean items (or attributes). All along this article, $\mathcal{R} \subseteq \mathcal{T} \times \mathcal{I}$ is a given binary database where some items $\mathcal{C} \subseteq \mathcal{I}$, called classes, partition the transactions, i.e., $\forall t \in \mathcal{T}, \exists! c \in \mathcal{C} | (t, c) \in \mathcal{R}$. A subset of items $I \subseteq \mathcal{I}$ is called itemset. Its support and frequency, in $\mathcal{R}$, are defined as follows:

Definition 1. Support of an itemset

The support of an itemset $I \subseteq \mathcal{I}$ in a binary database $\mathcal{R}$ is:

$$s(I, \mathcal{R}) = \{t \in \mathcal{T} | \{t\} \times I \subseteq \mathcal{R}\}.$$

Definition 2. Frequency of an itemset

The frequency of an itemset $I \subseteq \mathcal{I}$ in a binary database $\mathcal{R}$ is:

$$f(I, \mathcal{R}) = \frac{|s(I, \mathcal{R})|}{|\mathcal{T}|}.$$

In these definitions, $\mathcal{R}$ is a variable so that it is possible to talk about the support or the frequency of an itemset in the binary database $\mathcal{R}$ restricted to a subset $T \subseteq \mathcal{T}$ of transactions. Such a database is denoted $\mathcal{R}_T$ and formally defined as $\{(t, i) \in \mathcal{R} | t \in T\}$.

To avoid lengthy notations, the support of a class $c \in \mathcal{C}$, i.e., $s(\{c\}, \mathcal{R})$, is simply denoted $\mathcal{T}_c$. In this way, the distribution of the transactions in the 3-class binary database depicted in Fig. 1, is given by $(|\mathcal{T}_{c1}||\mathcal{T}_{c2}||\mathcal{T}_{c3}|)$. Because this toy database has 15 times more transactions in the class c_2 than in the class c_3, the classes can be said imbalanced and the limits of the OVA framework can (and will) be illustrated.

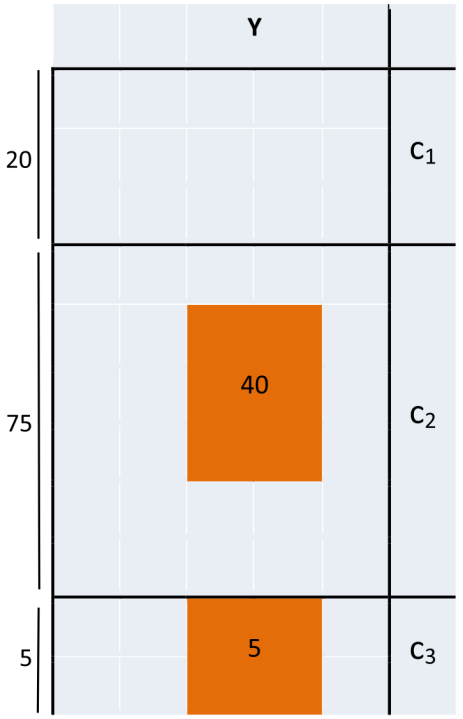

Fig. 1. A 3-class imbalanced data set.

The associative classification is based on *classification rules*, i.e., association rules of the form $I \rightarrow c$ where $I \subseteq \mathcal{I} \setminus \mathcal{C}$ is called the *body* of the rules and $c \in \mathcal{C}$ is the class on which the rule concludes. We now recall popular measures involved in the selection of the classification rules constituting the classification models that are learned:

Definition 3. Frequency-confidence

The frequency of a classification rule $I \rightarrow c$ in a binary database $\mathcal{R}$ is:

$$f(I \rightarrow c, \mathcal{R}) = f(I \cup \{c\}, \mathcal{R});$$

Its confidence is:

$$conf(I \rightarrow c, \mathcal{R}) = \frac{f(I \rightarrow c, \mathcal{R})}{f(I, \mathcal{R})}.$$

A frequency-confidence approach relies on a minimal frequency threshold and a minimal confidence threshold, i.e., the selected classification rules are frequent enough and confident enough.

Classification processes are linked to the search of contrasts between classes of transactions [13]. The growth rate is a well-used contrast measure [11].

Definition 4. Growth rate

The growth rate of a classification rule $I \rightarrow c$ in a binary database $\mathcal{R}$ is:

$$GR(I \rightarrow c, \mathcal{R}) = \begin{cases} 0 \text{ if } f\left(I, \mathcal{R}_{\mathcal{T}_c}\right) = 0 \\ \infty \text{ if } f\left(I, \mathcal{R}_{\mathcal{T}_c}\right) > 0 \wedge f\left(I, \mathcal{R}_{\mathcal{T} \setminus \mathcal{T}_c}\right) = 0 \\ \dfrac{f\left(I, \mathcal{R}_{\mathcal{T}_c}\right)}{f\left(I, \mathcal{R}_{\mathcal{T} \setminus \mathcal{T}_c}\right)} \text{ otherwise} \end{cases}.$$

An emerging pattern approach [11] relies on a minimal growth rate threshold, i.e., the selected classification rules have a growth rate that is large enough. Emerging patterns belong to the family of contrast patterns [14].

Other measures such as the lift [15] enable us to quantify interests of rules:

Definition 5. Lift

The lift of a classification rule $I \rightarrow c$, in a binary database $\mathcal{R}$ is:

$$lift(I \rightarrow c, \mathcal{R}) = \frac{f\left(I, \mathcal{R}_{\mathcal{T}_c}\right)}{f(I, \mathcal{R})}.$$

A positive correlation approach [16] relies on a minimal lift threshold, i.e., the selected classification rules must indicate a positive correlation, between I and c, that is large enough.

In the binary database represented by Fig. 1, considering the itemset $Y \subseteq \mathcal{I} \setminus \mathcal{C}$ such that $\left|s\left(Y, \mathcal{R}_{\mathcal{T}_{c_1}}\right)\right| = 0$, $\left|s\left(Y, \mathcal{R}_{\mathcal{T}_{c_2}}\right)\right| = 40$ and $\left|s\left(Y, \mathcal{R}_{\mathcal{T}_{c_3}}\right)\right| = 5$ as the body of classification rules is enough to understand relevance problems met by the aforementioned approaches in an OVA framework.

2.1. Frequency-confidence approaches

The confidence of $Y \rightarrow c_2 (conf(Y \rightarrow c_2, \mathcal{R}) = 40/45)$ is much higher than that of $Y \rightarrow c_3 (conf(Y \rightarrow c_3, \mathcal{R}) = 5/45)$. However, Y better "represents" the transactions classified in c_3 (all of them are supersets of Y) than those in c_2 (slightly more than half of them are supersets of Y). More generally, a frequency-confidence approach favors rules concluding on majority classes. This problem even exists with two classes. It is, therefore, present as well in an OVO (One-Versus-One) framework where the binary database is successively restricted to every pair of classes.

2.2. Emerging pattern approaches

The growth rate of $Y \rightarrow c_2 (GR(Y \rightarrow c_2, \mathcal{R}) = (40/75)/(5/25) = 2.667)$ is higher than that of $Y \rightarrow c_3 (GR(Y \rightarrow c_2, \mathcal{R}) = (5/5)/(40/95) = 2.375)$. However, as explained above, Y better "represents" the transactions classified in c_3 than those in c_2. This issue only arises when there are strictly more than two classes. In this case, and as illustrated with Y, an emerging pattern approach favors rules concluding on majority classes.

2.3. Positive correlation approaches

It is obvious from Definition 5 that, given an itemset I as the body of classification rules, the one maximizing the lift concludes on the class c providing the highest $f(I, \mathcal{R}_{\mathcal{T}_c})$. In other terms, the rules favored in a positive correlation approach have their bodies that better represent the transactions in the class they conclude on than those in any other class. The problem raised by the positive correlation approach is computational. To the best of our knowledge, no algorithm has been developed to efficiently extract classification rules under a minimal lift constraint. The fundamental issue stems from the fact that the constraint "having a lift above a given threshold" is neither monotone [17] nor anti-monotone [18]. Notice also that it is neither succinct [17] nor convertible [19]. As a consequence, the classical itemset mining methods are unable to use this constraint to prune the search space, which exponentially grows with the number of items. The rare positive correlation approaches extract a pool of frequent bodies for the classification rules (the frequency constraint is anti-monotone) and, in a post-processing step, compute their lifts to select those constituting the classification model. Unfortunately, the time to extract the frequent itemsets quickly grows when the minimal frequency threshold is lowered. As a consequence, the best rules that concludes on minority classes usually are inaccessible. For instance, in the binary database depicted in Fig. 1, a classification rule that perfectly concludes on c_3 (i.e., the support of its body is $\mathcal{T}_{c_3}$) cannot be found unless the minimal frequency threshold is set to 5% or less.

2.4. Other issues

Another issue, common to all OVA approaches, is the lack of flexibility to extract suitable sets of candidate classification rules. Indeed, a same minimal frequency/confidence/growth rate/lift threshold is usually used to select all classification rules. This can lead to rule conflicts, i.e., to the selection of two rules $X \rightarrow c$ and $X' \rightarrow c'$ with $X \subseteq X'$ and $c \neq c'$. For instance, in the binary database depicted in Fig. 1, the lifts of $Y \rightarrow c_2 (lift(Y \rightarrow c_2, \mathcal{R}) = (40/75)/(45/100))$ is 1.19 and that of $Y \rightarrow c_3 (lift(Y \rightarrow c_3, \mathcal{R}) = (5/5)/(45/100))$ is 2.22. It indicates positive correlations (i.e., the values are greater than 1) between Y and the respective classes. As a consequence, both may be part of the classification model even though they are in conflict. A post-processing step could refine the parameter so that no conflict occurs. However this would be computationally costly (quadratic in the number of extracted rules). Furthermore, with one single parameter, such a post-processing step would discard many rules that do not raise any conflict. By allowing different parameters for the different classes on which the rules conclude, less rules are removed. However, because

conflicting rules can conclude on any pair of classes, it would be even better to have parameters relating to every pair of classes. That would mean taking into account the repartition, in the different classes, of the "errors" (misclassified transactions) committed by a rule and it is in contradiction with the mere definition of the OVA framework.

Although the OVO (One-Versus-One) algorithms can be parametrized according to any pair of classes (and leading to a number of values that is quadratic in the number of classes), they cannot avoid rule conflicts either. In this framework, the rules are selected according to measures (such as those defined above) evaluated on binary databases restricted to pairs of classes. For instance, in the binary database depicted in Fig. 1, the model to classify in c_1 or c_2 is learned independently from the model to classify in c_1 or c_3. The classification rule $Y \rightarrow c_2$ may be selected during the former learning step; $Y \rightarrow c_3$ during the latter one. Like in the OVA framework, post-processing the selected rules altogether could allow to refine the parameters but this would be both costly and in contradiction with the definition of the framework. Table 1 summarizes the key principles of the classification processes of these frameworks to get a better understanding of their characteristics and differences.

3. One-Versus-Each framework

Let us consider, without loss of generality, a context $\{\mathcal{T}, \mathcal{I}, \mathcal{R}\}$ with p classes $\{c_1, c_2, \ldots, c_p\} \subseteq \mathcal{I}$. In order to take into account the various class sizes, we use per-class minimal frequency thresholds to select the bodies of the classification rules.

If an itemset has a frequency below the threshold associated with the class c_i, it is considered insufficiently representative of this class to be at the body of a rule concluding on it. Moreover, to control the distribution of the errors made by such a rule, an infrequency threshold (i.e., a maximal frequency threshold) needs to be set for every class $c_j \neq c_i$. A matrix, denoted Γ, concisely represents all parameters:

$$\Gamma = \begin{pmatrix} \gamma_{1,1} & \gamma_{1,2} & \cdots & \gamma_{1,p} \\ \gamma_{2,1} & \gamma_{2,2} & \cdots & \gamma_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{p,1} & \gamma_{p,2} & \cdots & \gamma_{p,p} \end{pmatrix}$$

The ith line of Γ parameterizes the frequency and infrequency constraints that must be satisfied by every itemset X at the body of a classification rule concluding on c_i. More precisely, X must be $\gamma_{i,i}$-frequent in $\mathcal{R}_{\mathcal{T}_{c_i}}$ (i.e., $f\left(X, \mathcal{R}_{\mathcal{T}_{c_i}}\right) \geq \gamma_{i,i}$) and, for all $j \neq i$, it must be $\gamma_{i,j}$-infrequent in $\mathcal{R}_{\mathcal{T}_{c_j}}$ (i.e., $f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right) < \gamma_{i,j}$). An additional minimal body constraint complements the definition of the itemsets selected at the body of the classification rules.

Definition 6. OVE-classification rule

Given a parameter matrix Γ, a classification rule $X \rightarrow c_i$ is an OVE-classification rule, abbreviated OVE-CR, if and only if the three following assertions are true:

1. X is frequent in $\mathcal{R}_{\mathcal{T}_{c_i}}$, i.e., $f\left(X, \mathcal{R}_{\mathcal{T}_{c_i}}\right) \geq \gamma_{i,i}$;
2. X is infrequent in every other class, i.e., $\forall j \neq i, f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right) < \gamma_{i,j}$;
3. X is a minimal body, i.e., $\forall Y \subset X, \exists j \neq i \mid f\left(Y, \mathcal{R}_{\mathcal{T}_{c_j}}\right) \geq \gamma_{i,j}$.

Statements 1 and 2 force the extracted itemsets to respect the frequency and infrequency thresholds imposed by Γ. Statement 3 is the minimal body constraint. It is valuable for classification purpose because it avoids some redundancy in the set of OVE-CRs: when rules concluding on a same class having bodies that are included into each other, the more general ones (i.e., the simplest w.r.t. the body) are preferred as long as this generality does not imply more errors in the classes that are not predicted.

It is interesting to notice that the bodies of the OVE-CRs can be seen as frequent emerging patterns with per-class frequency constraints and per-pair-of-classes growth rates. In particular, with a same value $\alpha \in [0,1]$ on the diagonal of Γ (i.e., for the

Table 1
Characterization of the OVA, the OVO and the OVE frameworks for a p-class problem. The database $\mathcal{R}$ is partitioned in $\left\{\mathcal{R}_{\mathcal{T}_{c_i}}\right\} i = 1..p$. Following the terminology of "contrast data mining" [14], Contrast($\mathcal{R}_{\mathcal{T}_{c_i}}, \mathcal{R}_{\mathcal{T}_{c_j}}$) stands for a classification sub-task on a database restricted to classes c_i and c_j.

Strategies	#tasks	Unit sub-tasks (for $p = 3$)
OVA	p	Contrast($\mathcal{R}_{\mathcal{T}_{c_1}}, \mathcal{R}_{\mathcal{T}} \setminus \mathcal{R}_{\mathcal{T}_{c_1}}$); Contrast($\mathcal{R}_{\mathcal{T}_{c_2}}, \mathcal{R}_{\mathcal{T}} \setminus \mathcal{R}_{\mathcal{T}_{c_2}}$); Contrast($\mathcal{R}_{\mathcal{T}_{c_3}}, \mathcal{R}_{\mathcal{T}} \setminus \mathcal{R}_{\mathcal{T}_{c_3}}$);
OVO	$p(p-1)$	Contrast($\mathcal{R}_{\mathcal{T}_{c_1}}, \mathcal{R}_{\mathcal{T}_{c_2}}$); Contrast($\mathcal{R}_{\mathcal{T}_{c_2}}, \mathcal{R}_{\mathcal{T}_{c_1}}$); Contrast($\mathcal{R}_{\mathcal{T}_{c_1}}, \mathcal{R}_{\mathcal{T}_{c_3}}$); Contrast($\mathcal{R}_{\mathcal{T}_{c_3}}, \mathcal{R}_{\mathcal{T}_{c_1}}$); Contrast($\mathcal{R}_{\mathcal{T}_{c_2}}, \mathcal{R}_{\mathcal{T}_{c_3}}$); Contrast($\mathcal{R}_{\mathcal{T}_{c_3}}, \mathcal{R}_{\mathcal{T}_{c_2}}$);
OVE	p	Contrast($\mathcal{R}_{\mathcal{T}_{c_1}}, \mathcal{R}_{\mathcal{T}_{c_2}}$) $\wedge$ Contrast($\mathcal{R}_{\mathcal{T}_{c_1}}, \mathcal{R}_{\mathcal{T}_{c_3}}$); Contrast($\mathcal{R}_{\mathcal{T}_{c_2}}, \mathcal{R}_{\mathcal{T}_{c_1}}$) $\wedge$ Contrast($\mathcal{R}_{\mathcal{T}_{c_2}}, \mathcal{R}_{\mathcal{T}_{c_3}}$); Contrast($\mathcal{R}_{\mathcal{T}_{c_3}}, \mathcal{R}_{\mathcal{T}_{c_1}}$) $\wedge$ Contrast($\mathcal{R}_{\mathcal{T}_{c_3}}, \mathcal{R}_{\mathcal{T}_{c_2}}$);

frequency thresholds) and a same value $\beta \in [0,1]$ in the other cells of Γ (i.e., for the infrequency thresholds), these bodies are α-frequent $\frac{\alpha}{\beta}$-EPs (see Definition 4). If $\beta = 0$, they are called, in the literature, jumping EPs. Being able to consider different frequency thresholds, the OVE-framework allows us to address the class imbalance problem. Thanks to the per-class infrequency thresholds (and even if they have a same value β), it takes into consideration the multi-class aspect of the task.

We now identify two constraints, on the parameter matrix Γ, that must be satisfied for the set of OVE-CRs to be relevant. The body of such a rule $X \rightarrow c_i$ intuitively needs to better represent the class c_i than any other class. This gives us the first constraint on Γ: the frequency threshold $\gamma_{i,i}$ must be greater than the infrequency thresholds $\gamma_{i,j}$ on the same line of Γ. More formally, Γ must satisfy the following constraint, namely $\mathbb{C}_{row}$:

$$\mathbb{C}_{row} \equiv \forall i \in \{1, \ldots, n\}, \forall j \neq i, \gamma_{i,j} \leq \gamma_{i,i}.$$

Although $\mathbb{C}_{row}$ has just been introduced by intuition, it can be further justified with some wanted properties it gives to the OVE-CRs. $\mathbb{C}_{row}$ actually makes the OVE-CRs have both a growth-rate and a lift strictly greater than 1 (proof in the Appendix):

Proposition 1. *If Γ satisfies $\mathbb{C}_{row}$, then any OVE-CR $X \rightarrow c_i$ extracted in the binary database $\mathcal{R}$ under the constraints expressed by Γ is such that $GR(X \rightarrow c_i, \mathcal{R}) > 1$.*

Proposition 2. *If Γ satisfies $\mathbb{C}_{row}$, then any OVE-CR $X \rightarrow c_i$ extracted in the binary database $\mathcal{R}$ under the constraints expressed by Γ is such that lift $(X \rightarrow c_i, \mathcal{R}) > 1$.*

As illustrated earlier on the toy example in Fig. 1, two classification rules $Y \rightarrow c_2$ and $Y \rightarrow c_3$ sharing a same body can both have growth rates and positive correlations greater than 1. In other terms, these properties are not sufficient to avoid rule conflicts. This leads us to the second constraint on Γ, namely $\mathbb{C}_{column}$:

$$\mathbb{C}_{column} \equiv \forall i \in \{1, \ldots, n\}, \forall j \neq i, \gamma_{i,j} \leq \gamma_{j,j}.$$

$\mathbb{C}_{column}$ forces the number of errors $X \rightarrow c_i$ makes in a class $c_j \neq c_i$ to be, in proportion of $\left|\mathcal{T}_{c_j}\right|$, lower than $\gamma_{j,j}$. Without it, X would also represent the class c_j, i.e., $X \rightarrow c_j$ could be extracted thus leading to rule conflicts. More formally, the following proposition holds (proof in the Appendix):

Proposition 3. *Let S_Γ be the complete set of OVE-CRs satisfying the frequency and infrequency constraints parametrized by Γ. If Γ satisfies $\mathbb{C}_{column}$, then S_Γ is conflict-free, i.e., it does not contain a pair of OVE-CRs $(X \rightarrow c_i, Y \rightarrow c_j)$ such that $X \subseteq Y$ and $i \neq j$.*

To conclude this section with a broader perspective, notice that any method following the OVA or the OVO framework somewhat works on several two-class problems ("every class against the union of the other classes" or "every class against every different class"), whereas the OVE framework directly takes into consideration the multi-class aspect of the problem. Furthermore, because every parameter of the OVE framework is relative to one single class, the imbalance aspect is, as well, directly taken into consideration.

4. Introducing the `fitcare` algorithm

Our approach, `fitcare`, is an instance of the OVE framework. This section describes it in three steps. First, given a parameter matrix Γ, an efficient algorithm is proposed in detail to compute exactly the OVE-CRs. Subsequently, the combination of the OVE-CRs into a classifier is explained. Finally, an optimization method to discover Γ is described.

4.1. Extraction

Given a context $\{\mathcal{T}, \mathcal{I}, \mathcal{R}\}$ with p classes $\{c_1, c_2, \ldots, c_p\} \subseteq \mathcal{I}$ and a parameter matrix Γ, the extraction of the complete set of OVE-CRs is divided into p independent sub-problems. They correspond to the p rows of the parameter matrix Γ, i.e., the OVE-CRs concluding on a class c_i (respecting the frequency constraint parametrized by $\gamma_{i,i}$ and the infrequency constraints parametrized by $(\gamma_{i,j})_{j \neq i}$ – let us call them $S_{\Gamma,i}$ – and are mined independently from those concluding on another class $c_j \neq c_i$. Algorithm 1, namely EXTRACT, computes such a set $S_{\Gamma,i}$. The complete set of OVE-CRs, S_Γ, is $\cup_{i=1}^{p} S_{\Gamma,i}$.

EXTRACT takes advantage of the anti-monotonicity of both the minimal frequency and the minimal body constraints to prune the search-space. Since the conclusion of the rule is fixed, this search-space is the potential bodies of the OVE-CRs. It is traversed in a breadth-first way, i.e., EXTRACT starts with a level containing the only itemset having zero item, $\emptyset$ (Line 4), and iteratively computes the level (called `futureParents`) of itemsets with $k+1$ items from the level of itemsets with k items (called `parents`). More precisely,

an itemset, called `child`, with $k+1$ items is constructed by union of an itemset, called `parent`, with k items and an additional item (class items excluded) that is greater than any item already in `parent` (Line 7). Any order on the items ensures that every OVE-CR is discovered only once. However, ordering the items by increasing frequency is a good heuristic (the famous *fail-first principle*) to decrease the number of itemsets EXTRACT enumerates, hence its running time.

For `child` to actually be inserted among the `futureParents` (Line 15), it must be frequent in $\mathcal{R}_{\mathcal{T}_{c_i}}$ (Line 10) and in at least another class (Line 11). Indeed, by anti-monotonicity of the frequency, if the former constraint is violated then every `child`'s superset is infrequent in $\mathcal{R}_{\mathcal{T}_{c_i}}$ and the search-space is safely pruned. If the former constraint is satisfied but the latter is not, then `child` is the body of an OVE-CR (output at Line 12) and, by definition, none of its strict supersets can be a minimal body. Again, the search-space is safely pruned. However, this time, not inserting `child` in `futureParents` is not enough to force the itemsets, enumerated later on, not to be supersets of `child`. That is why `child` is stored in a prefix tree, called `forbiddenPrefixes` (Line 13). When constructing new itemsets from a `parent`,

Algorithm 1: EXTRACT

Input : $\{\mathcal{T}, \mathcal{I}, \mathcal{R}\}$ a context with p classes $\{c_1, c_2, \ldots, c_p\} \subseteq \mathcal{I}$
c_i a class
$\gamma_{i,i} \in [0,1]$ a minimal frequency threshold in $\mathcal{R}_{\mathcal{T}_{c_i}}$
$(\gamma_{i,j})_{j\neq i} \in [0,1]^{p-1}$ maximal frequency thresholds in the $(\mathcal{R}_{\mathcal{T}_{c_j}})_{j\neq i}$
Output: $S_{\Gamma,i}$ the set of OVE-CRs concluding on c_i

```
1  forbiddenPrefixes ← ∅;
2  parents ← {∅};
3  while parents ≠ ∅ do
4      futureParents ← ∅;
5      for all parent ∈ parents do
6          forbiddenItems ← FORBIDDENITEMS(forbiddenPrefixes, parent);
7          for all item > LASTITEM(parent) do
8              if item ∉ forbiddenItems then
9                  child ← parent ∪ {item};
10                 if f(child, R_{T_{c_i}}) ≥ γ_{i,i} then
11                     if ∀j ≠ i, f(child, R_{T_{c_i}}) > γ_{i,j} then
12                         output (child→ c_i, (f(child, R_{T_{c_j}}))_{j∈{1,...,p}});
13                         INSERT(child, forbiddenPrefixes);
14                     else
15                         futureParents ← futureParents ∪ {child};
16                 else
17                     INSERT(child, forbiddenPrefixes);
18     parents ← futureParents;
```

EXTRACT efficiently computes the set of additional items (the `forbiddenItems`) that would make these itemsets be supersets of a branch of `forbiddenPrefixes` (Line 6). By excluding those items as valid extensions of `parent` (Line 8), only minimal bodies are enumerated. Taking advantage of this necessary structure, bodies that are infrequent in $\mathcal{R}_{\mathcal{T}_{c_i}}$ are stored as well in `forbiddenPrefixes` (Line 17). In this way, no strict superset of an infrequent body is ever enumerated.

From a technical point of view, the support of an itemset X is stored in p bitsets representing $\left(s\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right)\right)_{j\in\{1,\ldots p\}}$. The supports of the individual items (but the classes) are stored in the same way. Therefore, the support of a `child` is computed, just before Line 10, with a simple "bitwise and" operation between the support of the `parent` and that of the appended `item`. The choice of a prefix tree to store and access the `forbiddenPrefixes` is based on performance considerations too.

4.2. Classification

Algorithm EXTRACT not only outputs every OVE-CR but also its frequencies in every class (Line 12 in Algorithm 1). All these frequencies are used when it comes to classifying a new transaction t ($\subseteq\mathcal{I}$, i.e., seen as its set of items). To do so, a likeliness score is computed for each class. It simply is the sum of the frequencies in this class of the bodies of the OVE-CRs "matching" t, i.e., that are subsets of t:

$$l(t, c_i) = \sum_{j=1}^{p} \sum_{\{X\to c_j \in S_\Gamma | X\subseteq t\}} f\left(X, \mathcal{R}_{\mathcal{R}_{\mathcal{T}_{c_i}}}\right).$$

Notice that the class in which an OVE-CR concludes does not "hide" the exceptions it may have in other classes. The class predicted for t is the one providing the highest likelihood score, i.e., $\arg\max_{ci} l(t, c_i)$.

4.3. Automatic discovery of a locally optimal Γ

Given a p-class problem, the matrix Γ contains p^2 parameters to be set. That is why an exhaustive search of the parameters leading to the best classifier is practically impossible. Since manually tuning Γ would be tedious, an automatic optimization procedure was designed.

4.3.1. A constrained hill-climbing technique

The hill-climbing is a technique that aims at discovering a local optimum of a function f with a discrete domain. This domain — the search space — can be represented as a graph. Its vertices are the possible inputs for f and an edge connects two close inputs. The hill-climbing technique traverses this graph using f to decide what vertex to visit next. If rollbacks are allowed (when the newly visited vertex provides a worse solution than the previous one), this technique stops at a local optimum of f.

Our method is a constrained hill-climbing technique. The search space are the possible parameter matrices Γ. It is discrete since there is a finite number of transactions in the context, hence a finite number of frequency and infrequency thresholds that lead to different extractions with Algorithm 1. For a better efficiency (evaluating the function to optimize requires computing a set of OVE-CRs with Algorithm 1) and a better effectiveness, `fitcare` is not a generic hill-climbing. It takes advantage of the constraints $\mathbb{C}_{row}$ and $\mathbb{C}_{column}$ on Γ (see Section 3) and uses as well coverage considerations (i.e., the maximal possible number of transactions in $\mathcal{T}_{c_i}$ must be matched by the OVE-CRs concluding on c_i). The formal definition of the coverage rate δ_i of $\mathcal{T}_{c_i}$ by a set $S_{(\Gamma,i)}$ of OVE-CRs concluding on c_i is:

$$\delta_i = \frac{\left|\cup_{X\to c_i \in S_{(\Gamma,i)}} \{t \in \mathcal{T}_{c_i} | X \subseteq t\}\right|}{\left|\mathcal{T}_{c_i}\right|}.$$

We now provide details about `fitcare`'s key points.

4.3.1.1. Initialization. During an initialization step, `fitcare` learns the maximal coverage rates $(\delta_i)_{i \in \{1,\ldots,p\}}$ that can be achieved. It is first assumed that $(1, \ldots, 1)$ is reachable. Starting with every frequency and infrequency threshold at 1, these parameters in Γ are lowered until the full-coverage is reached or until a zero frequency (to avoid lengthy extractions when the full coverage is impossible, lower bounds of the per-class frequencies can be fixed). More precisely, when attempting to cover $\mathcal{T}_{c_i}$, $\gamma_{i,i}$ is lowered by units of $1/\left|\mathcal{T}_{c_i}\right|$ and, to satisfy $\mathbb{C}_{row}$, any infrequency threshold $\gamma_{i,j}$ is set to the highest multiple of $1/\left|\mathcal{T}_{c_j}\right|$ lesser than $\gamma_{i,i}$. If the full-coverage of $\mathcal{T}_{c_i}$ is not reached, the ith row of Γ is set to the values providing the highest δ_i that has been met.

The parameter matrix computed in this way violates $\mathbb{C}_{column}$ (unless a same frequency threshold suits all classes). To satisfy $\mathbb{C}_{column}$, every infrequency threshold $\gamma_{i,j}$ strictly greater than $\gamma_{j,j}$ is set to the highest multiple of $1/\left|\mathcal{T}_{c_j}\right|$ lesser than $\gamma_{j,j}$. EXTRACT is then called for each row that has been modified. If the best coverage rate of $\mathcal{T}_{c_i}$ has been lost, $\gamma_{i,i}$ is, again, lowered by units of $1/|\mathcal{T}_{c_i}|$ and, if necessary to satisfy $\mathbb{C}_{row}$, so are the infrequency thresholds $\gamma_{i,j}$. This procedure stops when the δ_i is met again. If this does not occur, this rate is set to the highest one that has been met and the ith row of Γ is rolled back accordingly.

The initialization goes on in this way, oscillating between the enforcement of $\mathbb{C}_{row}$ and that of $\mathbb{C}_{column}$. It ends when a parameter matrix satisfying both $\mathbb{C}_{row}$ and $\mathbb{C}_{column}$ leads to the extraction of OVE-CRs covering the $(\mathcal{T}_{c_i})_{i\in\{1,\ldots,p\}}$ with the highest possible rates $(\delta_i)_{i \in \{1,\ldots,p\}}$ discovered earlier. In practice, these rates often are $(1, \ldots, 1)$, i.e., EXTRACT is never called with very low minimal frequency thresholds and is fast thanks to frequency-based pruning.

The constrained hill-climbing technique starts from the parameter matrix discovered at the end of the initialization. All along the remaining traversal of the parameter space, the coverage rates $(\delta_i)_{i \in \{1,\ldots,p\}}$, discovered at the end of the initialization, are preserved. In other terms, any set of OVE-CRs that does provide these rates directly is considered worse than the best set so far. Now that we know where the hill-climbing starts its exploration, we present the objective function which locally optimized and how this function indicates the next move to do in the parameter space.

4.3.1.2. Function optimized by hill-climbing. The choice of the function to optimize obviously is critical. Given a parameter matrix at input, it must reflect the quality of the classification by the related OVE-CRs. Any classical measure for the assessment of a classifier could be used. However, many of these measures have a bias towards the approaches favoring the majority classes. For instance, the accuracy (i.e., the proportion of transactions, in a testing data set, that are correctly classified) of the classifier that always returns the majority class is high if a high proportion of transactions indeed is in this class. Since the OVE framework aims

at avoiding this bias, fitcare optimizes a new function. It is based on a measure of the tendency to classify in c_j the transactions in $\mathcal{T}'_{c_i}$, i.e., labeled with c_i in a testing data set[1]:

$$g\left(c_i, c_j\right) = \frac{\sum_{t \in \mathcal{T}'_{c_i}} l(t, c_i)}{\sum_{t \in \mathcal{T}'_{c_i}} l\left(t, c_j\right)}.$$

We baptize this measure *global growth rate*. It is a ratio of a sum of frequencies in $\mathcal{T}'_{c_i}$ and another one in $\mathcal{T}'_{c_j}$. The number of terms in both sums is the same. It corresponds to the sum of the numbers of OVE-CRs matching each transaction in $\mathcal{T}'_{c_i}$. The higher $g(c_i, c_j)$, the less confusion with c_j when classifying the transactions in $\mathcal{T}'_{c_i}$. fitcare maximizes the worse global growth rate, $\min_{i \neq j} g(c_i,c_j)$. The next paragraph explains how a deeper analysis of the terms at its denominator allows to find a reason for it to be low and therefore gives the direction, in the parameter space, in which moving would probably improve the classifier.

4.3.1.3. Driving the hill-climbing. Since the function to optimize is the minimal global growth rate over all pairs of different classes, improving the function is improving this minimum. Assume that it is obtained for the pair (c_i,c_j) of different classes. To order the causes for this low $g(c_i, c_j)$, the terms at its denominator (see the definition of l in Section 4.2) are grouped w.r.t. the classes at the conclusions of the matching OVE-CRs:

$$\begin{pmatrix} \sum_{t \in \mathcal{T}_{c_i}} \sum_{\{X \to c_1 \in S_\Gamma | X \subseteq t\}} f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right) \\ \sum_{t \in \mathcal{T}_{c_i}} \sum_{\{X \to c_2 \in S_\Gamma | X \subseteq t\}} f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right) \\ \vdots \\ \sum_{t \in \mathcal{T}_{c_i}} \sum_{\{X \to c_p \in S_\Gamma | X \subseteq t\}} f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right) \end{pmatrix}.$$

Sorting these values is also sorting the causes for the low $g(c_i, c_j)$. The two greatest values (i.e., the primary causes for a low $g(c_i, c_j)$) usually are the ith and the jth ones. The ith value indicates to what extent the OVE-CRs concluding on c_i make errors in c_j. The jth value indicates to what extent the OVE-CRs concluding on c_j apply to the transactions in c_j. The remaining values correspond to the errors made in c_j by the rules concluding on the p-2 other classes.

Each of these values directly relates to an infrequency threshold that fitcare lowers: a large jth value is associated with lowering $\gamma_{j,i}$ (by $1/|\mathcal{T}_{c_i}|$), while a large kth value ($k \neq j$) is an invitation to a smaller $\gamma_{k,j}$ (lowered by $1/\left|\mathcal{T}_{c_j}\right|$). After such an alteration of an infrequency threshold in the parameter matrix, the coverage rate of the transactions related to the modified row may decrease. A procedure similar to the one used for the initialization attempts to have, again, the best possible coverage rates of the transactions in every class while satisfying $\mathbb{C}_{row}$ and $\mathbb{C}_{column}$. Nevertheless, this time, the impossibility to reach the optimal coverage rates means an abortion: the parameters in the matrix Γ are rolled back to those that the optimized function has scored best so far. This occurs as well when the optimal coverage rates are reached but the score, provided by optimized function, actually is smaller. After a rollback of the parameter matrix, the second cause for the lowest global growth rate gives the new move in the parameter space, and so on until lowering any infrequency threshold leads to either suboptimal coverage rates or a worse classifier. When this happens, fitcare returns the best classifier (i.e., the best set of OVE-CRs) it has discovered and terminates.

4.3.2. fitcare*: algorithm*

Algorithm 2 more formally presents fitcare. The initialization of Γ (Line 1), thoroughly described earlier, is not detailed though. However, the way fitcare moves in the parameter space during the initialization (respecting both $\mathbb{C}_{row}$ and $\mathbb{C}_{column}$) is similar to the way it does it during the hill-climbing. Two variables coordinate the execution: isParameterModified and classId. classId is the index of Γ's row that fitcare currently considers or equals $p + 1$ (Line 16) when it has just been computed a set of OVE-CRs allowing maximal cover rates. When classId is at most p, the Boolean variable isParameterModified successively indicates whether the classId-$^{\text{th}}$ row of Γ may violate $\mathbb{C}_{column}$ (Line 5) and whether the OVE-CRs, related to this row of parameters, have not been extracted (Line 7). Just after the initialization, isParameterModified is set to false (Line 2) and classId to p (Line 3) but it immediately becomes $p + 1$ (Line 15). That is why, below, the validation or invalidation of a new set of OVE-CRs is presented first. Then, the way fitcare moves in the parameter space, respecting $\mathbb{C}_{row}$ and $\mathbb{C}_{column}$, is detailed.

[1] This data set is set apart from the learning data set before the initialization. In practice, we have chosen a stratified selection of 10% of the transactions. In the experimental section, fitcare is evaluated on yet other transactions that are set apart even earlier.

4.3.2.1. (In)validation of a new set of OVE*-CRs allowing maximal cover rates.* Given a set of OVE-CRs, WGGR evaluates the function to be optimized, i.e., computes the worst global growth rate (hence the acronym) between any pair of different classes. After comparing the value obtained with the current set of OVE-CRs with that of the best solution so far (Line 17), this solution is eitherreplaced by an even better one (Lines 18 and 19) or rolled back (Line 21). Then, and according to the ordering specified in Section 4.3.1, an infrequency threshold is lowered (Line 22). If a better set of OVE-CRs has just been found, this parameter relates to the primary cause for the worst global growth rate, otherwise the rank of the cause depends on how many moves from Γ_{best} have been tried so far. If the answer is "all of them", LOWERMOSTPROBLEMATICPARAMETER returns a value strictly above p, $S_{\Gamma_{best}}$ is output (Line 24) and `fitcare` terminates.

4.3.2.2. Traversing the parameter space respecting $\mathbb{C}_{row}$ *and* $\mathbb{C}_{column}$. If the `classId`-th row of Γ has not been modified since the last extraction of the related OVE-CRs, RATIONALIZEROW checks whether the enforcement of $\mathbb{C}_{column}$ justifies such a modification, which is made if necessary (Line 6). If it is not necessary, the next row of Γ is considered (Line 15) unless `classId` already indicates the last one, in which case a validation or invalidation of the set of OVE-CRs is done. If the `classId`-th row of Γ has been modified (Line 7), either to enforce $\mathbb{C}_{column}$ (Line 6) or earlier (Line 23), the related OVE-CRs must be extracted. The function LEARNRULESCONCLUDINGON does so. It calls EXTRACT with the parameter row as arguments and decreases the frequency threshold until the maximal cover rate is reached. Doing so, the infrequency thresholds may be decreased as well so that $\mathbb{C}_{row}$ is always satisfied (as described earlier for the initialization step). LEARNRULESCONCLUDINGON returns true if the maximal cover rate is reached, false otherwise. In the former case, `classId` is set back to 1 (Line 10) because, even though the previous parameter rows have not been modified (Line 9), they may now violate $\mathbb{C}_{column}$. In the latter case, the parameter matrix is rolled back to the best solution so far (Line 12) and the next best move in the parameter space is made (Line 13) unless no such move remained, in which case $S_{\Gamma_{best}}$ is output (Line 24) and `fitcare` terminates.

Algorithm 2: `fitcare`

Input : $\{\mathcal{T}, \mathcal{I}, \mathcal{R}\}$ a context with p classes $\{c_1, c_2, \ldots, c_p\} \subseteq \mathcal{I}$
Output: $S_{\Gamma_{best}}$ the set of OVE-CRs locally optimizing the worst global growth rate

```
1  Γ ← INIT(r);
2  isParametersModified ← false;
3  classId ← p;
4  while classId ≤ p do
5      if ¬ isParametersModified then
6          isParametersModified ← RATIONALIZEROW(classId);
7      if isParametersModified then
8          if LEARNRULESCONCLUDINGON(classId) then
9              isParametersModified ← false;
10             classId ← 1;
11         else
12             Γ ← Γ_best;
13             classId ← LOWERMOSTPROBLEMATICPARAMETER();
14     else
15         classId ← classId + 1;
16         if classId > p then
17             if WGGR(S_Γ) > WGGR(S_Γbest) then
18                 Γ_best ← Γ;
19                 S_Γbest ← S_Γ;
20             else
21                 Γ ← Γ_best;
22             classId ← LOWERMOSTPROBLEMATICPARAMETER();
23             isParametersModified ← true;
24 output S_Γbest;
```

5. Experimental validation

`fitcare`'s C++ implementation is distributed under the terms of the GNU GPLv3.[2] In this section, it is compared with recent and competitive state-of-the-art rule-based classifiers: CPAR [20] (based on inductive rules), HARMONY [21] (based on association

[2] http://dcc.ufmg.br/lcerf/en/prototypes.htmlfitcare.

Table 2
Global accuracies.

Data sets	CPAR	fitcare	HARMONY	DeEPs
breast-cancer	**70.63** ± 8.91	66.08 ± 11.79	69.93 ± 11.19	58.33 ± 7.59
breast-w	94.14 ± 4.16	**96.70** ± 2.03	95.70 ± 2.64	94.42 ± 3.04
colic	81.25 ± 3.89	81.79 ± 6.27	**82.88** ± 4.43	79.92 ± 5.75
credit-a	85.51 ± 4.25	81.01 ± 3.80	**85.65** ± 4.32	78.69 ± 4.20
diabetes	**73.31** ± 5.36	64.45 ± 5.95	73.04 ± 4.99	62.87 ± 11.34
heart-c	78.82 ± 6.97	**80.52** ± 6.83	78.87 ± 5.73	74.26 ± 6.37
heart-h	78.3 ± 9.77	78.57 ± 8.33	**82.31** ± 6.34	76.23 ± 5.26
heart-s	81.48 ± 6.20	**83.33** ± 4.76	81.48 ± 6.41	74.44 ± 6.30
hepatitis	78.54 ± 11.10	79.35 ± 8.63	**85.16** ± 6.47	67.04 ± 15.42
labor	68.67 ± 17.71	**80.7** ± 18.93	**80.7** ± 22.91	76.00 ± 20.15
meningite	87.51 ± 5.09	**93.61** ± 3.43	92.7 ± 3.36	68.38 ± 4.38
sonar	75.48 ± 7.24	76.44 ± 5.36	**81.73** ± 10.76	68.76 ± 10.53
ticTacToe	70.98 ± 2.05	89.87 ± 11.55	**97.18** ± 1.63	93.01 ± 2.68
balance-scale	70.08 ± 4.95	**75.04** ± 4.08	73.12 ± 3.56	55.84 ± 8.86
car	78.42 ± 3.59	83.85 ± 1.81	**89.35** ± 2.72	87.79 ± 1.73
iris	**94.67** ± 4.99	**94.67** ± 4.00	**94.67** ± 5.81	85.33 ± 10.24
waveform	74.28 ± 1.70	79.74 ± 1.03	80.3 ± 0.93	**82.04** ± 2.01
wine	93.86 ± 4.61	91.57 ± 5.16	**96.06** ± 3.59	81.41 ± 9.79
zoo	93.18 ± 6.05	**95.04** ± 8.49	92.07 ± 8.48	65.28 ± 15.11
Average	80.48	82.75	84.89	75.27
Average rank	2.8158	2.0263	1.6316	3.5263

rules) and DeEPs [22] (based on emerging patterns). These algorithms are chosen because they rely on different kinds of rules and have been shown to achieve better performances than older proposals (such as RIPPER [23], CBA [2], CMAR [3] and other classifiers based on emerging patterns). HARMONY was kindly provided by their authors [21], the Java implementation of CPAR is that of [24] and the KEEL platform [25] includes the DeEPs algorithm that is used for these experiments.

The performances of these algorithms are compared with those of fitcare on 18 UCI data sets [26] and on a real-world data set, "meningitis", describing children hospitalized for acute bacterial or viral meningitis (the two classes). The continuous attributes are discretized according to the entropy (as described in [27]) and the sole training set is used to do so. The same discretization is then applied to the test set when it comes to assess the learned classifiers. The listed results are all obtained from a 10-fold stratified cross-validation. The global accuracy, the balanced error rate and the per-class accuracy are chosen as quality measures. In all cases, the statistical significance of the measured differences of performance is tested. The AUC (area under the ROC curve) is not used because the chosen implementations of fitcare's competitors only predict the class of a transaction instead of scoring each class for this transaction.

The remainder of this section first shows that there is no critical difference between the global performance of fitcare and those of the best contender. It then stresses that fitcare outperforms them when it comes to predicting minority classes of an imbalanced context. Finally, further experiments on synthetic data sets assert that fitcare is not biased towards the majority class.

5.1. Global accuracy

The global accuracy simply is the proportion of transactions that are correctly classified. The results are listed in Table 2. The average accuracy of each classifier is reported as well as its average rank.

The Friedman test [28] is applied to the ranking results so that their statistical significance is assessed. At a confidence level of 0.1, the null hypothesis is rejected, i.e., the classifiers show global performances that are significantly different. Proceeding to the

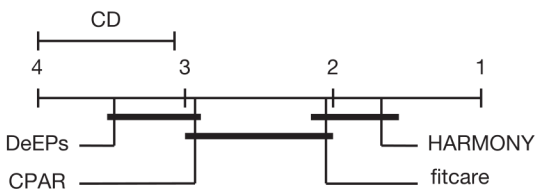

Fig. 2. The critical difference diagram for the global accuracy.

Table 3
Balanced error rates.

Data sets	CPAR	fitcare	HARMONY	DeEPs
breast-cancer	**29.95** ± 12.82	37.04 ± 12.96	35.66 ± 12.16	44.53 ± 8.27
breast-w	5.74 ± 4.26	**3.30** ± 2.26	5.05 ± 3.07	5.35 ± 2.23
colic	**18.17** ± 4.66	19.31 ± 5.26	19.51 ± 6.1	23.75 ± 5.70
credit-a	**13.98** ± 3.78	19.14 ± 3.62	14.81 ± 4.55	20.80 ± 3.74
diabetes	**29.18** ± 6.42	33.02 ± 5.09	30.05 ± 6.40	41.63 ± 11.53
heart-c	19.81 ± 7.56	**19.43** ± 7.30	21.36 ± 6.09	25.92 ± 6.26
heart-h	21.2 ± 11.38	24.17 ± 9.34	**20.83** ± 6.81	25.00 ± 6.57
heart-s	17.28 ± 6.13	**16.84** ± 4.58	18.92 ± 7.44	25.75 ± 6.55
hepatitis	25.24 ± 13.51	**18.80** ± 10.67	27.85 ± 16.00	25.77 ± 18.69
labor	30.00 ± 19.97	**18.32** ± 17.85	20.61 ± 26.25	22.92 ± 19.74
meningite	16.48 ± 6.59	**8.20** ± 6.35	13.90 ± 6.88	21.23 ± 2.92
sonar	21.27 ± 6.91	23.58 ± 5.03	**18.16** ± 10.77	30.75 ± 10.45
ticTacToe	31.45 ± 2.99	9.38 ± 11.94	**3.72** ± 2.20	5.35 ± 2.07
balance-scale	49.13 ± 6.22	**44.62** ± 4.55	47.11 ± 2.63	59.63 ± 6.55
car	44.48 ± 10.29	26.35 ± 6.04	31.66 ± 5.66	**25.43** ± 8.19
iris	**4.89** ± 4.90	5.33 ± 4.00	5.33 ± 5.81	14.67 ± 10.24
waveform	25.64 ± 1.71	20.17 ± 1.04	19.69 ± 0.92	**17.88** ± 2.01
wine	5.29 ± 4.07	7.46 ± 4.79	**3.93** ± 0.92	19.75 ± 10.54
zoo	31.48 ± 11.51	**11.06** ± 9.37	16.81 ± 11.34	49.57 ± 14.16
Average	23.19	19.24	19.73	26.61
Average rank	2.4211	2.0526	2.1579	3.4211

Nemenyi post-hoc test, the chart in Fig. 2 is obtained. It represents the critical difference of global performance between the four algorithms (with $CD \simeq 0.96$). HARMONY is better ranked than fitcare but this advantage is not statistically significant. As a consequence, it can be written that, despite fitcare's emphasis on correctly predicting minority classes, it also competes with the state-of-the-art associative classifiers in terms of global performance.

5.2. Balanced error rate

Let us recall that, given a p-class context, the balanced error rate (BER) of a classifier is the average of the error rates in each class:

$$BER = \frac{ER_{c_1} + ER_{c_2} + \cdots + ER_{cp}}{p},$$

where ER_{c_i} is the error rate on the transactions that should be classified in c_i.

BER results are listed in Table 3. Again, the average BER accuracy of each classifier is reported as well as its average rank.

Like for the global accuracies, the Friedman test [28] is applied to the ranking results. At a confidence level of 0.1, the null hypothesis is rejected, i.e., the classifiers show performances that are significantly different. Proceeding to the Nemenyi post-hoc test, the chart in Fig. 3 is obtained (still with $CD \simeq 0.96$). Although fitcare is ranked first, its advantage over HARMONY and CPAR is not statistically significant. In fact, the only statistically significant statement that can be made relates to the inferiority of DeEPs, which is also ranked last according to the global accuracy.

5.3. Performance in minority classes

In imbalanced contexts, the minority classes often are those of interest. In this section, a class is considered a minority class if its number of transactions is, at most, 60% that of the largest class. With this definition, 19 of the 49 classes, in the considered data sets, qualify as minority classes. They are spread across 12 different data sets, which are marked with an asterisk in Tables 4 and 5.

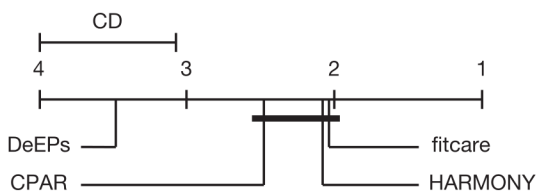

Fig. 3. The critical difference diagram for the balanced error rate.

Table 4
Per-class accuracies over 19 minor classes involved in 12 data sets.

Data sets	CPAR	fitcare	HARMONY	DeEPs
breast-cancer	78.15/**61.95**	70.64/55.29	78.1/50.58	62.19/48.75
201/**85**	±6.08/24.47	±11.58/18.97	±11.87/18.52	±9.52/14.05
breast-w	94.48/94.04	96.72/**96.68**	97.37/92.53	93.87/95.42
458/**241**	±4.72/5.92	±2.81/4.45	±2.91/5.48	±5.24/2.92
colic	84.35/**79.31**	84.91/76.47	89.65/71.32	90.09/62.42
232/**136**	±4.63/10.91	±10.56/6.92	±4.87/14.21	±6.60/6.60
credit-a	79.03/93.01	79.47/82.24	81.1/89.29	83.71/74.68
307/383	±5.75/2.96	±3.27/5.84	±7.39/3.96	±2.92/8.32
diabetes	77.23/64.42	58.6/**75.37**	80.2/59.7	73.20/43.55
500/**268**	±4.74/9.01	±9.59/7.69	±4.24/12.33	±13.27/16.03
heart-c	80.79/79.6	80/81.15	81.21/76.08	75.04/73.13
165/138	±8.02/12.61	±5.39/13.04	±7.96/12.22	±12.20/11.65
heart-h	81.64/**75.96**	85.63/66.03	90.42/67.92	79.74/70.27
188/**106**	±7.02/19.01	±7.96/15.62	±6.1/10.02	±10.11/17.40
heart-s	83.52/81.92	84.66/81.66	84.66/77.5	76.00/72.50
150/120	±7.79/10.24	±10.35/9.72	±6/17.89	±9.52/12.94
hepatitis	56.59/92.93	84.37/78.04	50/94.3	**86.67**/61.79
32/123	±25.21/6.42	±21.88/11.47	±33.33/3.99	±26.67/14.15
labor	54.83/85.17	**85**/78.37	75/83.78	80.00/74.17
20/37	±26.23/29.07	±22.91/24.49	±40.31/20.57	±24.49/25.67
meningite	71.46/95.59	88.09/95.51	72.61/99.59	**100.00**/57.55
84/245	±11.96/2.71	±14.13/3.21	±14.14/1.2	±0.00/5.83
sonar	70.1/87.37	76.28/76.57	83.5/80.18	79.11/59.39
97/111	±7.84/12.13	±14.7/15.32	±12.37/14.03	±14.31/17.47
ticTacToe	77.08/60.03	88.17/93.07	99.2/93.37	89.29/**100.00**
626/**332**	±1.76/6.87	±11.1/14.04	±1.07/4.25	±4.14/0.00
balance-scale	66.52/3/83.1	79.86/**4**/82.29	80.55/0/78.12	62.83/0/58.28
288/**49**/288	±6.58/6.4/12.7	±8.98/8/7.35	±12.08/0/12.91	±18.25/0.00/13.99
car	91.52/54.16/39.34/37.05	92.39/61.71/**66.67**/**73.84**	95.37/**85.67**/24.63/67.69	95.45/71.38/60.24/71.19
1210/**384**/**69**/**65**	±2.59/6.64/16.55/38.36	±0.86/7.57/19.52/12.55	±2.67/8.64/15.71/16.67	±2.00/7.21/24.14/16.75
iris	100/92.67/92.67	100/90/94	100/92/92	100/66/90
50/50/50	±0/9.04/9.04	±0/10/9.17	±0/9.8/13.27	±0/23.75/10
waveform	71.28/75.99/75.8	66.84/87.77/84.89	77.71/81.24/81.99	71.45/87.96/86.95
1657/1647/1696	±1.32/3.42/3.24	±4.62/6.22/4.3	±2.49/1.46/1.63	±4.48/2.91/2.16
wine	92.14/93.67/98.33	96.61/83.09/97.91	96.61/95.77/95.83	81.67/88.57/70.50
59/71/48	±7.9/8.11/5	±10/10.69/6	±6.77/6.55/9.07	±20.34/14.00/23.07
zoo	94.67/**100**/20/**100**/30/55/80	97.56/**100**/**60**/**100**/**75**/**100**/**90**	97.56/**100**/40/92.3/**75**/87.5/**90**	73/65/30/75/20/40/50
41/**20**/**5**/**13**/**4**/**8**/**10**	±11.07/0/40/0/45.83/47.17/33.17	±7.5/0/45.83/0/45.83/40/30	±7.5/0/40/15/45.83/45.83/30	±17.64/39.05/45.83/ 40.31/40/48.99/50
Avg rank (minor classes)	2.7632	1.7105	2.7105	2.8158

5.3.1. Per-class accuracy results

The per-class accuracies are reported in Table 4. The distribution of the transactions in the classes is indicated in the column entitled *data sets*. The numbers in bold face relate to the 19 minority classes. The last row gives the average rank of the four classifiers over these 19 classes.

Again, the Friedman test [28] is applied to the ranking results. At a confidence level of 0.1, the null hypothesis is rejected, i.e., the classifiers show performances that are significantly different. Proceeding to the Nemenyi post-hoc test, the chart in Fig. 4 is obtained (still with $CD \simeq 0.96$). This time, `fitcare` is found to provide statistically better results than any of the three other competitors. We therefore conclude on the superiority of `fitcare` when it comes to correctly classify transactions in minority classes. That makes `fitcare` an appealing choice to learn *alarms*, i.e., classifiers identifying malfunctioning states of a system (those states being, hopefully, exceptional hence minority classes). As shown with previous experiments, `fitcare`'s performances the minority classes are not achieved at the detriment of the global accuracy or the balanced error rate.

Back to the detailed results in Table 4, it can be observed that the superiority of `fitcare` is particularly obvious in contexts with strictly more than two classes. Indeed, `fitcare` provides the best per-class accuracy in nine out of the ten minority classes. On the contrary, over all imbalanced contexts (i.e., those having at least one minority class), there are 13 classes that are not minority classes and `fitcare` only provides the best per-class accuracy for one of them.

5.3.2. F-score results

Because per-class precisions do not take into account false positive rates, the statistically better results obtained by `fitcare` could hide a bias towards the minority classes, i.e., those classes would be over-predicted. The results in Sections 5.1 and 5.2 have

Table 5
F-score results.

Data sets	CPAR	fitcare	HARMONY	DeEPs
breast-cancer	**0.5792**	0.4921	0.5000	0.4080
breast-w	0.9209	**0.9530**	0.9370	0.9218
colic	**0.7714**	0.7564	0.7549	0.6967
credit-a	0.8423	0.7884	0.8342	0.7776
diabetes	**0.6223**	0.5968	0.6072	0.4500
heart-c	0.7857	0.7915	0.7664	0.7214
heart-h	0.7297	0.6897	**0.7347**	0.6820
heart-s	0.8066	0.8133	0.7881	0.7160
hepatitis	0.6102	**0.6279**	0.5818	0.5234
labor	0.5946	**0.7556**	0.7317	0.7111
meningite	0.7742	**0.8757**	0.8356	0.6176
sonar	0.7598	0.7513	0.8100	0.7032
titactoe	0.5896	0.8643	**0.9583**	0.9083
balance-scale	0.0102	**0.0250**	0.0000	0.0000
car	0.5347	0.6295	**0.7815**	0.7220
	0.1298	0.2480	0.1560	**0.2847**
	0.1170	0.2560	**0.3235**	0.3136
iris	0.9259	0.9259	0.9259	0.8197
waveform	0.6615	0.7419	0.7313	0.7646
wine	0.9038	0.8624	0.9293	0.6733
zoo	0.7273	**0.8889**	0.8333	0.4127
	0.1176	**0.5454**	0.3333	0.0976
	0.6341	**0.8387**	0.7500	0.3509
	0.1176	**0.5454**	0.4286	0.0513
	0.3478	**0.7619**	0.6364	0.1395
	0.5161	**0.7826**	0.6923	0.2128
Avg rank (minor classes)	2.8421	1.7368	2.0263	3.3947

already shown that, globally, such a bias is not perceptible. Nevertheless, this could be due to the fact that the tendency to over-predict the minority classes affects a quantity of transactions which is small w.r.t. the total number of transactions (but not w.r.t. the sizes of the minority classes).

To more convincingly assert the absence of a bias towards the minority classes, the precisions in the minority classes, used in the previous set of experiments, are dropped in favor of F-scores. The F-score (also F-measure) is the harmonic mean of precision and recall. In this way, if all transactions in a minority class are correctly classified, the precision in this class is 1; but if this same class is predicted for as many transactions in other classes, then the F-score only is $2 \times \frac{1 \times 0.5}{1 \times 0.5} = 0.75$. Contrary to global measures, the total number of transactions does not intervene and a bias towards minority classes would therefore significantly affect the F-score.

Table 5 lists the F-scores obtained by `CPAR`, `fitcare`, `HARMONY` and `DeEPs` in all 19 minority classes (whose sizes are reported in bold face in Table 4). `fitcare` gets the best F-score for 11 of the 19 classes. The last row of the table gives the average rank of the four classifiers. At a confidence level of 0.1, the Friedman test allows to reject the null hypothesis, i.e., the classifiers show performances that are significantly different. Proceeding to the Nemenyi post-hoc test, the chart in Fig. 5 is obtained (with $CD \simeq 0.96$). `fitcare` still is ranked first, hence an absence of a bias towards minority classes. The F-scores it gets in the minority classes are significantly better than those of `CPAR` and `DeEPs`. `HARMONY`, ranked second, provides, in the minority classes, F-scores that are not significantly different from `fitcare`'s.

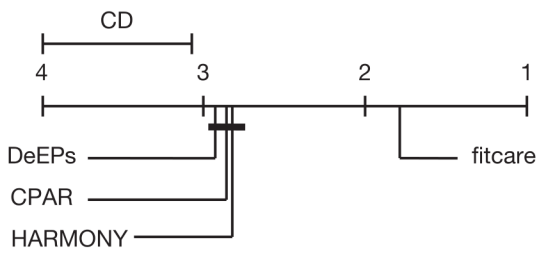

Fig. 4. The critical difference diagram for the per-class accuracy in minor classes.

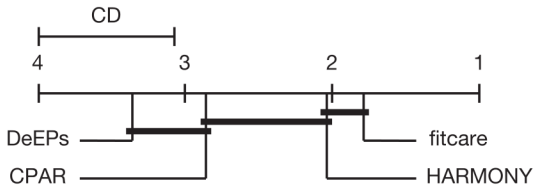

Fig. 5. The critical difference diagram for the F-score on minor classes.

5.4. Evolution of performance in imbalanced contexts

The `waveform` data set has three classes that are balanced since they respectively contain 1657, 1647 and 1696 transactions. By removing transactions from one or two classes, various artificially imbalanced data sets are built. More precisely, a class c_i is turned into a minority class by randomly partitioning T_{c_i} into subsets of the same size ($+1$ transaction) that amounts to x% of the original size ($x \in \{50, 33, 25, 20, 16, 10\}$). One single partition is kept and classifiers are learned with `CPAR`, `fitcare`, `HARMONY` and `DeEPs`. Like earlier, they are assessed by 10-CV. Figs. 6, 7, 8 and 9 give, in function of x, the achieved per-class accuracies (averaged over all partitions of the class(es) that is/are artificially smaller).

Fig. 6 reports the results obtained when only c_1 is reduced. The three other figures correspond to artificial reductions of two classes. In all cases, a same conclusion is drawn: `CPAR` and `HARMONY` are biased towards the larger class(es), whereas `fitcare` and `DeEPs` are not. Indeed, when x increases, the per-class accuracies of `CPAR` and `HARMONY` significantly increase in the majority class(es) and significantly decrease in the minority class(es). Notice that this fall is rather impressive when only c_1 is reduced. When only 10% of its transactions are kept, `CPAR` correctly classifies one tenth of them; `HARMONY` one quarter. On the contrary, the per-class accuracies of `fitcare` and `DeEPs` are stable in the reduced class(es). Surprisingly, and contrary to `fitcare`, `DeEPs` looks biased towards the minority class(es). Indeed, `DeEPs` seems to maintain stable the accuracy in the minority class(es) at the cost of a lower accuracy in the majority class(es). This is particularly noticeable in Figs. 7 and 8. Furthermore, we recall that, for all three quality measures used in the previous experiments, `DeEPs` always ranks last.

In conclusion, the experiments in the previous sections have shown that, without any significant loss at the global scale, `fitcare` provides better results in the minority classes, yet it is not biased towards such classes. By gradually turning a balanced data set into more and more imbalanced ones, the effect of class imbalance on the four tested classifiers becomes clearer: `fitcare`'s and `DeEPs`'s performance are not affected by this parameter, whereas `CPAR` and `HARMONY` are biased towards the majority classes. Because `CPAR` and `HARMONY` instantiate the `OVA` framework, we believe it is an empirical observation of the bias towards majority classes that is theoretically explained in Section 2.

5.5. Run time

The EXTRACT algorithm takes advantage of the anti-monotonicity of both the minimal frequency and the minimal body constraints to prune the classification rule search space. In this regard, it can be said "efficient". The constrained hill-climbing, we have designed, initializes the frequency thresholds at 1 and, ignoring the rollbacks, it only decreases them. Therefore the fastest extractions are tested first. Nevertheless, and despite $\mathbb{C}_{row}$ and $\mathbb{C}_{column}$, which discard many parameter matrices, thousands of them are sometimes tried before the discovery of a parameter matrix that locally maximizes the worst global growth rate. That is why `CPAR`, `HARMONY` and `DeEPs` usually run faster than `fitcare`. Anyway, over all 19 tested data sets, `fitcare` usually requires less than 10 s to learn the classifier. Table 6 reports the four exceptions to this rule, i.e., the run times greater than 10 s, on the most demanding data sets.

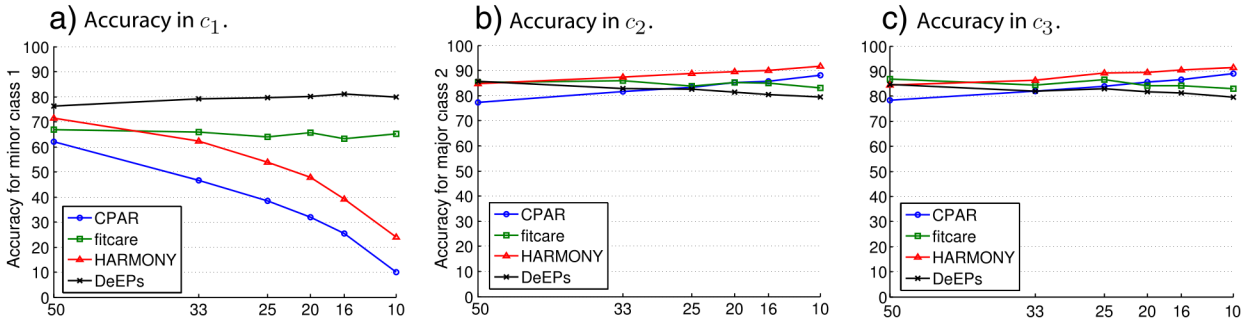

Fig. 6. Per-class accuracy results when the class c_1 is reduced to 50, 33, 25, 20, 16 and 10% of its original size.

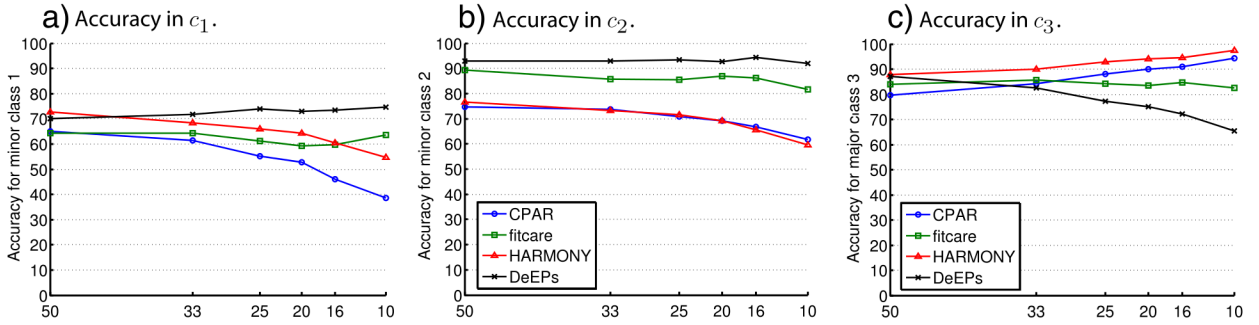

Fig. 7. Per-class accuracy results when the classes c_1 and c_2 are reduced to 50, 33, 25, 20, 16 and 10% of their original sizes.

6. Related work

The related work is organized in four categories corresponding to the key domains of the paper.

6.1. Rule-based algorithms

Two families of algorithms aim at discovering a set of rules for classification purposes: the induction-rule-based algorithms and the association-rule-based (i.e., `CBA-like`) algorithms. `FOIL` [29], `IREP` [30], `RIPPER` [23] and `CPAR` [20] belong to the former family. They follow a sequential database covering strategy (see, e.g., [16] chap. 5, p. 213) and greedily "grow" the body of a rule one item at a time. This item is chosen according to a heuristics such as the maximization of the information gain. Association-rule-based techniques like `CBA` [2], `APRIORI-C` [31], `CMAR` [3], `ARC-BC` [32] use association rule mining algorithms (see, e.g., [33,34]) to extract a complete set of frequent and confident association rules. From this set, classification rules are selected w.r.t. coverage, redundancy and/or relevance. They become the basis for a classifier. EPs-based classifiers like `CAEP` [35], `JEP-C` [36], `DeEPs` [22] or [37] follow the same strategy (see also [38]). Another rule-based algorithm, namely `HARMONY` [21], has recently proposed a new instance-centric strategy to directly mine classification rules. `HARMONY` uses per-class frequency thresholds (instead of a global one) for more accurate predictions in imbalanced multi-class contexts. Today, it is often cited as the best associative classifier with `CPAR` as a contender for two-class problems. All these rule-based algorithms follow the `OVA` framework.

6.2. Classifying in strictly more than two classes

The `OVO` (One Versus One, also known as pairwise [39]) framework aims at improving the effectiveness of the classification in strictly more than two classes. In this framework, a p-class problem is divided into $p(p-1)/2$ two-class sub-problems, i.e., that many classifiers are learned to address each sub-problem. Given a new object, these classifiers are combined so that it is assigned to a unique class. In this framework, the base learner (i.e., the algorithm learning to discriminate between two classes) is a variable. As such, the methods following this framework are meta-models. For example, [40] uses `RIPPER` [23] as a base learner and a simple voting strategy to combine the classifiers. Compared to the simple use of the base learner, the `OVO` framework indeed enables more accurate predictions in multi-class contexts. Nevertheless, some limits have been pointed out. In particular, it does not solve the imbalanced class issue: if the base learner is biased towards the majority class, the meta-classifier suffers from the same problem.

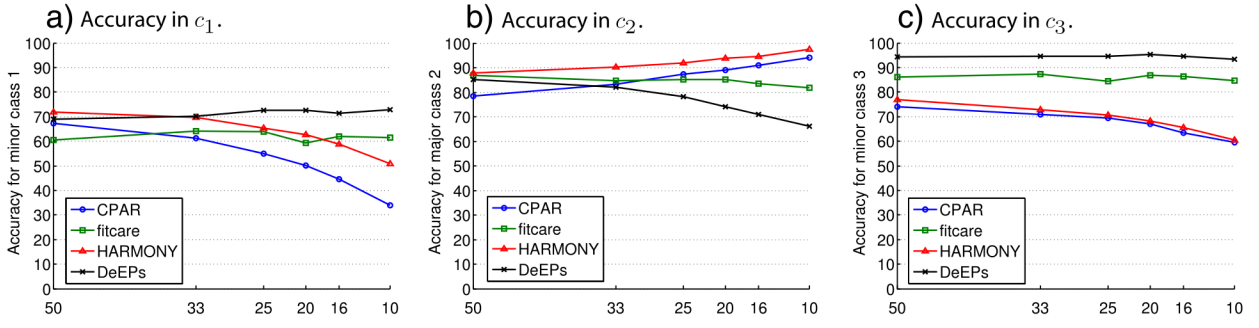

Fig. 8. Per-class accuracy results when the classes c_1 and c_3 are reduced to 50, 33, 25, 20 16, and 10% of their original sizes.

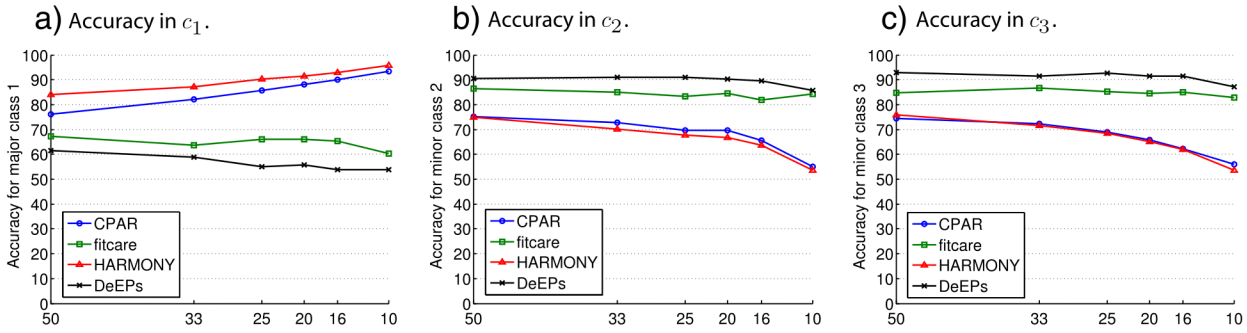

Fig. 9. Per-class accuracy results when the classes c_2 and c_3 are reduced to 50, 33, 25, 20, 16 and 10% of their original sizes.

6.3. Classifying in imbalanced classes

Barandela et al. [41] identify three main strategies to learn classifiers from imbalanced contexts:

- (i) *By re-sampling data.* This strategy – the most famous one – is a pre-processing step. It turns the imbalanced context into a balanced one. Either *under-sampling* or *over-sampling* is used to annul the difference between the sizes of the positive and the negative classes. Under-sampling is at the risk of ignoring useful information. Over-sampling entails a computational overhead and often leads to over-fitting the minority class(es). Anyway, in imbalanced contexts, both techniques increase the accuracy of the classifiers learned in a second step. Under-sampling usually gets the upper hand. `SMOTE` [42] combines under and over-sampling. Some experimentations are conducted in [43] combining different classifiers and *re-sampling* methods to study the effect on accuracy rate.
- (ii) *By declaring higher costs to misclassifying objects from the minority class(es).* In many imbalanced problems, errors in the prediction of the minority classes are very costly. It is about learning an alarm that detects (rare) anomalies. On the contrary, the misclassification of negative examples (i.e., false alarms) is not as costly. Cost-based classification is about learning a classifier from the data given a cost matrix, i.e., a matrix specifying the cost related to misclassifying in c_j an object from c_i (for every pair of classes (c_i, c_j)). The classifier is to minimize the total misclassification cost. By associating high costs with the misclassification of objects from the minority classes, the usual bias towards the majority class(es) is counterbalanced. Since the first workshop dedicated to this topic [44], many approaches have been studied. Some proposals weight the learning objects according to the costs [45,46]. Other proposals rely on meta-methods [47,48]. Qin et al. [49] wrote a quick, yet well-structured, survey for readers interested in those approaches and other ones not mentioned here.
- (iii) *By integrated algorithms.* The last strategy is the one `fitcare` embraces. It is about modifying the design of existing algorithms to cope with imbalanced contexts. Since msCBA [50], several associative classification methods have been adapted. To the best of our knowledge, `HARMONY` [21] is `fitcare`'s toughest contender. Other classification frameworks have been extended as well. For example, in the domain of classification trees, [51] uses *m*-estimates to smooth the probability estimates at the leaves and [52,53] propose new criteria to split a node while constructing the tree.

Today, class imbalance still is considered a difficult and open problem. Some works [54–56] and recent workshops [7,57] have been dedicated to it. Interested readers may refer to a recent survey by Sun et al. [58].

6.4. Parameter tuning

Automatically tuning the parameters (e.g., frequency thresholds) of an associative classifier is a difficult and open problem. A few recent methods tackle it. [59] computes polynomial approximations and fuzzy estimations to automatically learn an adequate frequency threshold for frequent pattern mining. [60] uses a hill-climbing technique to "navigate" in the parameter space and maximize the accuracy of `CBA`-like classifiers. Our algorithm, `fitcare`, uses as well a hill-climbing technique. However, it tunes several (in)frequency thresholds trying to minimize the worst confusion between any pair of classes.

Table 6
`fitcare`'s run times on the most demanding data sets (average over the learning sets of the 10-CV).

Data sets	Run time (s)
colic	254.1 ± 191.6
credit-a	660.6 ± 366.2
sonar	1454.5 ± 4594.6
waveform	652.2 ± 732.6

7. Conclusion and perspectives

This article deals with the pattern-based classification in multi-class imbalanced contexts. The contribution is three-fold: a new framework dedicated to this problem has been presented; an efficient algorithm, instantiating the framework and presenting good properties (such as the absence of rule conflict), has been designed; and a constrained hill-climbing technique has been proposed to automatically tune the parameters so that the confusion between any pair of classes is minimized. By taking into account the errors a classification rule makes into *each* class, the approach has been both theoretically and empirically shown to not over-classify in the majority class(es), a problem affecting the state-of-the-art associative classifiers such as CPAR and HARMONY. Moreover, experiments demonstrate that the superior effectiveness in minority classes does not harm the accuracy at a global scale. The proposal actually is statistically on par with the best contender, HARMONY. The anti-monotone constraints, that the selected rules satisfy, all allow to prune the search space. However, and despite the traversal of the parameter space from the easiest extractions to the hardest ones, tuning the parameters may require thousands of extractions. Turning the approach lazy may be a solution we plan to investigate. Another interesting perspective relates to the ability to affect costs to classification errors. Indeed, the cost-sensitive classification (see, e.g., [61]) is intimately linked with the classification in imbalanced contexts.

Acknowledgments

We wish to thank Jianyong Wang and George Karypis for providing the HARMONY prototype, and Frans Coenen for providing an efficient implementation of CPAR. We also wish to thank Marc Boullé for his relevant comments on this work. This work has been partly funded by the CNPq, FAPEMIG and InWeb, and by two French contracts, ANR-07-MDCO-014 BINGO2 and ANR-2010-COSI-012-01 FOSTER.

Appendix A

Proof of Propositions 1 and 2. Let us first prove Proposition 2 and then explain how this reasoning can be easily adapted to prove Proposition 1.

By Definition 1, the lift of $X \to c_i$ in $\mathcal{R}$ is:

$$lift(X \to c_i, \mathcal{R}) = \frac{f\left(X, \mathcal{R}_{\mathcal{T}_{c_i}}\right)}{f(X, \mathcal{R})}.$$

Rewriting $f(X, \mathcal{R})$ using Definition 2 gives:

$$lift(X \to c_i, \mathcal{R}) = \frac{|\mathcal{T}| f\left(X, \mathcal{R}_{\mathcal{T}_{c_i}}\right)}{s(X, \mathcal{R})}.$$

Because the p classes partition the transactions, we have:

$$lift(X \to c_i, \mathcal{R}) = \frac{|\mathcal{T}| f\left(X, \mathcal{R}_{\mathcal{T}_{c_i}}\right)}{\sum_{j=1}^{p} s\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right)}.$$

Using again Definition 2, these per-class supports are turned into per-class frequencies:

$$lift(X \to c_i, \mathcal{R}) = \frac{|\mathcal{T}| f\left(X, \mathcal{R}_{\mathcal{T}_{c_i}}\right)}{\sum_{j=1}^{p} \left|\mathcal{T}_{c_j}\right| f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right)}.$$

Γ forces $f\left(X, \mathcal{R}_{\mathcal{T}_{c_i}}\right) \geq \gamma_{i,i}$ and $\forall\, j \neq i, f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right) < \gamma_{i,j}$. Therefore:

$$lift(X \rightarrow c_i, \mathcal{R}) > \frac{|\mathcal{T}|\gamma_{i,i}}{\sum_{j=1}^{p}\left|\mathcal{T}_{c_j}\right|\gamma_{i,j}}.$$

Finally, because $\mathbb{C}_{row}$ imposes that $\forall\, j \neq i$, $\gamma_{i,j} \leq \gamma_{i,i}$, we conclude on the positive correlation between X and c_i:

$$lift(X \rightarrow c_i, \mathcal{R}) > \frac{|\mathcal{T}|\gamma_{i,i}}{\sum_{j=1}^{p}\left|\mathcal{T}_{c_j}\right|\gamma_{i,i}} = \frac{|\mathcal{T}|}{\sum_{j=1}^{p}\left|\mathcal{T}_{c_j}\right|} = \frac{|\mathcal{T}|}{|\mathcal{T}|} = 1.$$

The proof of Proposition 1 starts with the definition of the growth rate and follows the same steps as above. The only difference is that the denominators of the fractions exclude the class at the conclusion of the rule. The proof therefore ends with $GR(X \rightarrow c_i, \mathcal{R}) > \frac{|\mathcal{T}\setminus\mathcal{T}_{c_i}|\gamma_{i,i}}{\sum_{j\neq i}\left|\mathcal{T}_{c_j}\right|\gamma_{i,i}} = \frac{\left|\mathcal{T}\setminus\mathcal{T}_{c_i}\right|}{\left|\mathcal{T}\setminus\mathcal{T}_{c_i}\right|} = 1.$

Proof of Proposition 3. Assume, by contradiction, that S_Γ contains a pair of OVE-CRs $(X \rightarrow c_i, Y \rightarrow c_j)$ such that $X \subseteq Y$ and $i \neq j$. By anti-monotonicity of the frequency, we have:

$$f\left(Y, \mathcal{R}_{\mathcal{T}_{c_j}}\right) \leq f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right).$$

$X \rightarrow c_i$ and $Y \rightarrow c_j$ respect the frequency/infrequency constraints parametrized by Γ. In particular:

$$\begin{cases} f\left(Y, \mathcal{R}_{\mathcal{T}_{c_j}}\right) \geq \gamma_{j,j} \\ f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right) < \gamma_{i,j} \end{cases}.$$

Because $\mathbb{C}_{column}$ imposes that $\gamma_{i,j} \leq \gamma_{j,j}$, a contradiction ($\gamma_{j,j} < \gamma_{j,j}$) is reached:

$$\gamma_{j,j} \leq f\left(Y, \mathcal{R}_{\mathcal{T}_{c_j}}\right) \leq f\left(X, \mathcal{R}_{\mathcal{T}_{c_j}}\right) < \gamma_{i,j} \leq \gamma_{j,j}.$$

Therefore the initial assumption is false, i.e., S_Γ is conflict-free.

References

[1] R. Agrawal, T. Imielinski, A.N. Swami, Mining association rules between sets of items in large databases, Proceedings SIGMOD'93, ACM Press, 1993, pp. 207–216.
[2] B. Liu, W. Hsu, Y. Ma, Integrating classification and association rule mining, Proceedings KDD'98, AAAI Press, 1998, pp. 80–86.
[3] W. Li, J. Han, J. Pei, CMAR: accurate and efficient classification based on multiple class-association rules, Proceedings ICDM'01, IEEE Computer Society, 2001, pp. 369–376.
[4] B. Crémilleux, J.-F. Boulicaut, Simplest rules characterizing classes generated by delta-free sets, Proceedings ES'02, Springer, 2002, pp. 33–46.
[5] M.-L. Antonie, O.R. Zaïane, An associative classifier based on positive and negative rules, Proceedings DMKD'04, ACM Press, 2004, pp. 64–69.
[6] E. Baralis, S. Chiusano, Essential classification rule sets, ACM Transactions on Database Systems 29 (4) (2004) 635–674.
[7] N.V. Chawla, N. Japkowicz, A. Kotcz, Editorial: special issue on learning from imbalanced data sets, ACM SIGKDD Explorations 6 (1) (2004) 1–6.
[8] G.I. Webb, Discovering significant rules, Proceedings KDD'06, ACM Press, 2006, pp. 434–443.
[9] F. Verhein, S. Chawla, Using significant positively associated and relatively class correlated rules for associative classification of imbalanced datasets, Proceedings ICDM'07, IEEE Computer Society, 2007, pp. 679–684.
[10] B. Arunasalam, S. Chawla, CCCS: a top-down associative classifier for imbalanced class distribution, Proceedings KDD'06, ACM Press, 2006, pp. 517–522.
[11] G. Dong, J. Li, Efficient mining of emerging patterns: discovering trends and differences, Proceedings KDD'99, ACM Press, 1999, pp. 43–52.
[12] K. Ramamohanarao, H. Fan, Patterns based classifiers, World Wide Web 10 (1) (2007) 71–83.
[13] P.K. Novak, N. Lavrac, G.I. Webb, Supervised descriptive rule discovery: a unifying survey of contrast set, emerging pattern and subgroup mining, Journal of Machine Learning Research 10 (2009) 377–403.
[14] G. Dong, J. Bailey, Contrast Data Mining: Concepts, Algorithms, and Applications, Chapman & Hall/CRC, 2012.
[15] L. Geng, H.J. Hamilton, Interestingness measures for data mining: a survey, ACM Computing Surveys 38 (3) (2006).
[16] P.-N. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, Addison-Wesley, 2005.

[17] R.T. Ng, L.V.S. Lakshmanan, J. Han, A. Pang, Exploratory mining and pruning optimizations of constrained association rules, Proceedings SIGMOD'98, ACM Press, 1998, pp. 13–24.
[18] G. Grahne, L.V.S. Lakshmanan, X. Wang, Efficient mining of constrained correlated sets, Proceedings ICDE'00, IEEE Computer Society, 2000, pp. 512–521.
[19] J. Pei, J. Han, Can we push more constraints into frequent pattern mining? Proceedings KDD'00, ACM Press, 2000, pp. 350–354.
[20] X. Yin, J. Han, CPAR: classification based on predictive association rules, Proceedings SDM'03, SIAM, 2003, pp. 369–376.
[21] J. Wang, G. Karypis, On mining instance-centric classification rules, IEEE Transactions on Knowledge and Data Engineering 18 (11) (2006) 1497–1511.
[22] J. Li, G. Dong, K. Ramamohanarao, L. Wong, DeEPs: a new instance-based lazy discovery and classification system, Machine Learning 54 (2) (2004) 99–124.
[23] W.W. Cohen, Fast effective rule induction, Proceedings ICML'95, Morgan Kaufmann, 1995, pp. 115–123.
[24] F. Coenen, The LUCS-KDD software, library, http://www.csc.liv.ac.uk/#frans/KDD/Software/2004.
[25] J. Alcalá-Fdez, L. Sánchez, S. García, M. del Jesus, S. Ventura, J. Garrell, J. Otero, C. Romero, J. Bacardit, V. Rivas, J. Fernández, F. Herrera, KEEL: a software tool to assess evolutionary algorithms for data mining problems, Soft Computing — A Fusion of Foundations, Methodologies and Applications 13 (3) (2009) 307–318.
[26] A. Asuncion, D. Newman, UCI machine learning repository, http://archive.ics.uci.edu/ml/2007.
[27] U.M. Fayyad, K.B. Irani, Multi-interval discretization of continous-valued attributes for classification learning, Proceedings IJCAI'93, Morgan Kaufmann, 1993, pp. 1022–1027.
[28] J. Demsar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.
[29] J.R. Quinlan, R.M. Cameron-Jones, FOIL: a midterm report, Proceedings ECML'93, Springer, 1993, pp. 3–20.
[30] J. Fürnkranz, G. Widmer, Incremental reduced error pruning, Proceedings ICML'94, Morgan Kaufmann, 1994, pp. 70–77.
[31] V. Jovanoski, N. Lavrac, Classification rule learning with APRIORI-C, Proceedings EPIA'01, Springer, 2001, pp. 44–51.
[32] M.-L. Antonie, O.R. Zaïane, Text document categorization by term association, Proceedings ICDM'02, IEEE Computer Society, 2002, pp. 19–26.
[33] R. Agrawal, R. Srikant, Fast algorithms for mining association rules in large databases, Proceedings VLDB'94, Morgan Kaufmann, 1994, pp. 487–499.
[34] J. Han, J. Pei, Y. Yin, Mining frequent patterns without candidate generation, in: W. Chen, J.F. Naughton, P.A. Bernstein (Eds.), Proceedings SIGMOD'00, ACM Press, 2000, pp. 1–12.
[35] G. Dong, X. Zhang, L. Wong, J. Li, CAEP: classification by aggregating emerging patterns, Proceedings DS'99, Springer, 1999, pp. 30–42.
[36] J. Li, G. Dong, K. Ramamohanarao, Making use of the most expressive jumping emerging patterns for classification, Knowledge and Information Systems 3 (2) (2001) 131–145.
[37] J. Bailey, T. Manoukian, K. Ramamohanarao, Classification using constrained emerging patterns, Proceedings WAIM'03, Springer, 2003, pp. 226–237.
[38] H. Fan, K. Ramamohanarao, Fast discovery and the generalization of strong jumping emerging patterns for building compact and accurate classifiers, IEEE Transactions on Knowledge and Data Engineering 18 (6) (2006) 721–737.
[39] S.-H. Park, J. Fürnkranz, Efficient pairwise classification, Proceedings ECML'07, Springer, 2007, pp. 658–665.
[40] J. Fürnkranz, Round robin classification, Journal of Machine Learning Research 2 (2002) 721–747.
[41] R. Barandela, J.S. Sánchez, V. García, E. Rangel, Strategies for learning in class imbalance problems, Pattern Recognition 36 (3) (2003) 849–851.
[42] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, Journal of Artificial Intelligence Research 16 (2002) 321–357.
[43] J.V. Hulse, T. Khoshgoftaar, Knowledge discovery from imbalanced and noisy data, Data & Knowledge Engineering 68 (12) (2009) 1513–1542.
[44] In: T. Dietterich, D. Margineantu, F. Provost, P. Turney (Eds.), ICML 2000 Workshop on Cost-sensitive Learning, 2000.
[45] K.M. Ting, An instance-weighting method to induce cost-sensitive trees, IEEE Transactions on Knowledge and Data Engineering 14 (3) (2002) 659–665.
[46] B. Zadrozny, J. Langford, N. Abe, Cost-sensitive learning by cost-proportionate example weighting, ICDM'03, 2003, pp. 435–442.
[47] P. Domingos, Metacost: a general method for making classifiers cost-sensitive, KDD'99, 1999, pp. 155–164.
[48] Y. Sun, M.S. Kamel, A.K.C. Wong, Y. Wang, Cost-sensitive boosting for classification of imbalanced data, Pattern Recognition 40 (12) (2007) 3358–3378.
[49] Z. Qin, C. Zhang, T. Wang, S. Zhang, Cost sensitive classification in data mining, ADMA'10, 2010, pp. 1–11.
[50] B. Liu, Y. Ma, C.K. Wong, Improving an association rule based classifier, PKDD'00, 2000, pp. 504–509.
[51] B. Zadrozny, C. Elkan, Learning and making decisions when costs and probabilities are both unknown, KDD'01, 2001, pp. 204–213.
[52] P. Lenca, S. Lallich, T.-N. Do, N.-K. Pham, A comparison of different off-centered entropies to deal with class imbalance for decision trees, PAKDD'08, 2008, pp. 634–643.
[53] D.A. Cieslak, T.R. Hoens, N.V. Chawla, W.P. Kegelmeyer, Hellinger distance decision trees are robust and skew-insensitive, Data Mining and Knowledge Discovery 24 (1) (2012) 136–158.
[54] P. Jeatrakul, K.W. Wong, C.C. Fung, Classification of imbalanced data by combining the complementary neural network and smote algorithm, in: K.W. Wong, B.S.U. Mendis, A. Bouzerdoum (Eds.), ICONIP (2), Vol. 6444 of Lecture Notes in Computer Science, Springer, 2010, pp. 152–159.
[55] T. Liu, Y. Liang, W. Ni, A learning strategy for highly imbalanced classification, in: C. Xu, R. Steinmetz, A. El-Saddik, C.-W. Ngo, X. Wu (Eds.), ICIMCS, ACM International Conference Proceeding Series, ACM, 2011, pp. 116–119.
[56] S. Ertekin, C. Rudin, On equivalence relationships between classification and ranking algorithms, Journal of Machine Learning Research 12 (2011) 2905–2929.
[57] In: N.V. Chawla, N. Japkowicz, Z.-H. Zhou (Eds.), PAKDD'09 Workshop: Data Mining When Classes are Imbalanced and Errors Have Costs, 2009.
[58] Y. Sun, A.K.C. Wong, M.S. Kamel, Classification of imbalanced data: a review, International Journal of Pattern Recognition and Artificial Intelligence 23 (4) (2009) 687–719.
[59] S. Zhang, X. Wu, C. Zhang, J. Lu, Computing the minimum-support for mining frequent patterns, Knowledge and Information Systems 15 (2) (2008) 233–257.
[60] F. Coenen, P.H. Leng, The effect of threshold values on association rule based classification accuracy, Data & Knowledge Engineering 60 (2) (2007) 345–360.
[61] N.V. Chawla, D.A. Cieslak, L.O. Hall, A. Joshi, Automatically countering imbalance and its empirical relationship to cost, Data Mining and Knowledge Discovery 17 (2) (2008) 225–252.