Pattern Recognition 53 (2016) 36–45

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Fuzzy rough classifiers for class imbalanced multi-instance data

Sarah Vluymans [a,b,*], Dánel Sánchez Tarragó [c], Yvan Saeys [b,e], Chris Cornelis [a,d], Francisco Herrera [d,f]

[a] Department of Applied Mathematics, Computer Science and Statistics, Ghent University, Belgium
[b] VIB Inflammation Research Center, Ghent, Belgium
[c] Department of Computer Science, Central University of Las Villas, Cuba
[d] Department of Computer Science and Artificial Intelligence, University of Granada, Spain
[e] Department of Respiratory Medicine, Ghent University, Belgium
[f] Faculty of Computing and Information Technology – North Jeddah, King Abdulaziz University, Saudi Arabia

ARTICLE INFO

Article history:
Received 29 April 2015
Received in revised form
22 October 2015
Accepted 3 December 2015
Available online 12 December 2015

Keywords:
Multi-instance learning
Fuzzy rough set theory
Imbalanced data

ABSTRACT

In multi-instance learning, each learning object consists of many descriptive instances. In the corresponding classification problems, each training object is labeled, but its constituent instances are not. The classification objective is to predict the class label of unseen objects. As in traditional single-instance classification, when the class sizes of multi-instance data are imbalanced, classification is degraded. Many multi-instance classifiers have been proposed, but few take into account the possibility of class imbalance, which causes them to fail in this situation. In this paper, we propose a new type of classifier that embodies a solution to the multi-instance class imbalance problem. Our proposal relies on the use of fuzzy rough set theory. We present two families of classifiers respectively based on information extracted at bag-level and at instance-level. We experimentally show that our algorithms outperform state-of-the-art solutions to multi-instance imbalanced data classification, evaluated by the popular metrics AUC and geometric mean.

1. Introduction

In machine learning, multi-instance learning (MIL, [1]) is a generalization of the traditional single-instance attribute-value approach. While in the single-instance setting each learning object has a single descriptive vector, in MIL each learning object is composed of many vectors, although all vectors relate to the same set of descriptive attributes. In MIL jargon, a learning object is called a *bag* and every descriptive vector is an *instance*. As in traditional learning, classification is one of the most important tasks of MIL. Only the bags, and not their instances, have class labels. The objective of multi-instance classification is to predict the class label of unseen bags using a model built from a training set.

Many multi-instance classification algorithms have been proposed. However, most have been designed and evaluated considering data with balanced classes. The class imbalance problem affects both single and multi-instance learning. The problem occurs when at least one of the classes has a disproportionately small size compared to the other classes. In these cases, classifiers tend to make more errors on small classes and may even ignore them completely, although small classes are usually more of interest. This problem has received much attention in single-instance learning (e.g. [2–4]), but has barely been studied in MIL. To our knowledge, existing solutions in the MIL scenario are limited to the contributions of [5–7], that consider both preprocessing techniques to modify the class imbalance as well as a set of cost-sensitive boosting algorithms.

The *K*-nearest neighbor classifier (KNN, [8]) is one of the most popular learning algorithms [9]. It assigns an unseen object to the decision class most frequent among the *K* closest training objects to the unseen object. Over the years, several improvements and adaptations of KNN have been proposed. One successful modification is the fuzzy rough nearest neighbor classifier (FRNN, [10]), which introduces fuzzy rough set theory into KNN. Fuzzy rough set theory [11] is a framework to model vague (fuzzy) and incomplete (rough) information, by introducing fuzzy set theory [12] into rough set theory [13]. Rough sets approximate a concept by means of a lower and upper approximation. The former contains elements which *certainly* belong to the concept, while the latter consists of elements *possibly* belonging to it. The integration of fuzzy set theory in rough sets allows for a more flexible instance similarity measure and graded membership degrees of elements to the approximations. Concretely, similarity between instances is

* Corresponding author at: Department of Applied Mathematics, Computer Science and Statistics, Ghent University, Belgium. Tel.: +32 9 264 47 57; fax: +32 9 264 49 95.
E-mail address: Sarah.Vluymans@UGent.be (S. Vluymans).

http://dx.doi.org/10.1016/j.patcog.2015.12.002
0031-3203/

measured by a fuzzy relation and the constructed concept approximations are fuzzy sets. Fuzzy rough set theory has been used successfully in many single-instance machine learning applications, including classification (e.g. [14–16]). In the classification framework, it allows to model a degree of membership of elements to approximations of the decision classes. The FRNN method uses the unseen object's K nearest neighbors to construct the lower and upper approximation of each decision class and then computes the membership of the unseen object to this approximations. The hybridization between fuzzy rough sets and KNN results in a classifier more robust to vague and incomplete information [10]. Recently, a further improvement on FRNN was introduced in [17] by using ordered weighted average operators (OWA, [18]) and class dependent weight vectors. This method, called IFROWANN, was specifically designed to handle class imbalance and proved to be very effective in single-instance class imbalanced classification.

In this paper, we introduce a new type of multi-instance classifiers, based on the IFROWANN method, which inherently contain a solution to the class imbalance problem in multi-instance classification. The more complex nature of multi-instance data prompts us to propose two families of classifiers: (1) bag-based fuzzy rough classifiers which rely on relationships between bags, considering the bag as a whole and (2) instance-based fuzzy rough classifiers based on affinities that instances themselves have with classes. In our experimental study, we show that the proposed fuzzy rough nearest neighbors classifiers outperform state-of-the-art solutions to class imbalanced multi-instance classification.

The remainder of this paper is structured as follows. We set out in Section 2 with a specification of multi-instance classification and the class imbalance problem and review previous proposals. In Section 3, we recall the IFROWANN method from [17], which forms the inspiration for our proposal. Section 4 considers multi-instance classification and introduces our proposed method dealing with class imbalance in this situation. The experimental evaluation of our proposal is conducted in Section 5. We conclude the paper and lay out future research paths in Section 6.

2. The class imbalance problem in multi-instance classification

In this preliminary section, we specify the formal definition of multi-instance classifiers. We recall the class imbalance problem and how it has been dealt with in single-instance classification. Finally, we review the efforts made to handle class imbalance in multi-instance classification problems.

2.1. Multi-instance classification

MIL was introduced in [1] in a study of drug activity prediction based on multiple molecular conformations. Since then, it has attracted a considerable amount of attention due to its ability to model data ambiguity and the link it forms between classical attribute-value learning and relational learning [19]. MIL has mainly been used in applications related to image recognition (e.g. [20–22]). Other important application domains include bioinformatics (e.g. [23–25]), text classification and web mining (e.g. [26–29]) and computer-aided medical diagnosis and medical imaging (e.g. [30–32]).

Given the instance space $\mathcal{X}$ and the label set $\mathcal{Y}$, a bag X_i is a multiset of instances $\{x_{i1}, x_{i2}, \ldots, x_{in_i}\}$ with $x_{ij} \in \mathcal{X}$. The number n_i denotes the cardinality of X_i. Note that we use lowercase letters to denote instances and uppercase letters for bags. Each bag is paired with a label $y_i \in \mathcal{Y}$. Considering training data $T = \{(X_1, y_1), \ldots, (X_m, y_m)\}$, we formally define a multi-instance classifier $h(X)$ as an approximate model to the real function $f : \mathbb{N}^{\mathcal{X}} \rightarrow \mathcal{Y}$, where $\mathbb{N}^{\mathcal{X}}$ is the set of all multisets consisting of elements from $\mathcal{X}$, that is, the set of all possible bags.

Multi-instance datasets traditionally consist of two classes, one positive and one negative. Several hypotheses exist to decide when a bag of instances can be considered as positive [33]. The standard multi-instance hypothesis assumes that a bag is positive when at least one of its instances is positive. If not, the bag is negative. An alternative is the threshold based assumption, which states that the number of positive instances in a bag should exceed a given threshold before the bag can be considered positive. Overall, it implies that it is a too naive approach to assume that all instances in a positive bag can be labeled as positive and all instances in a negative bag as negative. This was for instance shown in the review of [34]. We will take this into account in the development of our classifiers.

2.2. The class imbalance problem

In single-instance classification, class imbalance occurs when the elements in the dataset are unequally distributed among the classes. The main focus has been on binary imbalanced problems, where elements of the *majority* class outnumber those of the *minority* class. The elements of the majority class are traditionally denoted as *negative* and those of the minority class as *positive*. This coincides with the fact that the minority or positive class is usually the class of interest (e.g. [35]).

There are three main types of solutions used in traditional classification to deal with class imbalance. Firstly, solutions at the data level (resampling methods) perform undersampling of the majority class, oversampling of the minority class or a combination of both in order to balance the number of examples in the two classes. Secondly, there exist solutions at the algorithmic level, in which heuristics are incorporated into classic algorithms to handle class imbalance, for example, by adjusting probabilities and weights to favor the positive class. Of particular interest in this type of solutions are cost-sensitive methods [36], which assign higher costs to the misclassification of positive examples, while aiming to minimize the overall classification cost. The third group consists of ensemble solutions, that introduce one of the above solutions (e.g. resampling or cost-sensitivity) in an ensemble algorithm to create a layer of abstraction effectively separating the method used to counteract the class imbalance from the base classifier used in the ensemble.

Although many solutions to class imbalance have been proposed in traditional classification, they are not directly applicable to multi-instance scenarios due to the structural differences in the datasets. In particular, multi-instance data consists of two levels: instances and bags. The grouping of instances in bags is essential additional information that should be taken into account. Furthermore, the actual labeled data samples (bags) in multi-instance data are far more complex than those in single-instance data (instances) and the single-instance solutions simply cannot process them. Class imbalance appears in multi-instance problems like text, web, and image applications [5–7], but it has been little addressed in the literature so far. In multi-instance classification, class imbalance presents itself as an unequal distribution of the bags among the classes, that is, we encounter a larger number of negative bags compared to positive ones. The imbalance ratio (IR) expresses the degree of class imbalance and is defined, for a two-class dataset, as the ratio of the number of negative over the number of positive bags, i.e., $\mathrm{IR} = |N|/|P|$, where P and N are the positive and negative classes respectively. While multi-instance classification is not limited to two-class problems, this setting has been the main focus of researchers in the field [34]. Moreover, class imbalance has also been mainly studied for binary problems in single-instance classification. All previous proposals dealing

with class imbalance in multi-instance classification are developed for two-class datasets as well, as discussed in Section 2.3. For these reasons, we also focus on two-class problems.

2.3. Related work

In [5,6], two techniques to handle class imbalance in multi-instance data are proposed. Firstly, they develop oversampling methods based on the single-instance SMOTE method [37]. In their BagSMOTE method, they increase the size of the positive class by artificially generating new positive bags. In their second algorithm InstanceSMOTE, they create new descriptive instances and add them to existing positive bags in order to obtain a better representation of this class. The IR of the dataset remains the same in this case. Both methods are preprocessing algorithms and need to be combined with a multi-instance classifier to complete the classification process. In their experimental study, the authors showed that BagSMOTE yields better results compared to InstanceSMOTE.

The second part of the contribution of [5,6] is their development of cost-sensitive classification procedures, based on the AdaBoost.M1 boosting scheme [38]. The AdaBoost.M1 method trains a base classifier in each iteration and reweighs instances based on their classification outcome, to ensure that misclassified instances are more focused on in the next iteration. The traditional weight update formula is

$$D_{t+1}(i) = \frac{D_t(i)K_t(X_i, y_i)}{Z_t} \tag{1}$$

with

$$K_t(X_i, y_i) = \exp(-\alpha_t y_i h_t(X_i)). \tag{2}$$

Here, t is the iteration number, Z_t is a normalization factor chosen such that D_{t+1} is a probability distribution and $\alpha_t \in \mathbb{R}$ is the coefficient associated with the classifier h_t, representing its weight in the final classification aggregation of the ensemble. The methods of [5,6] introduce class-dependent costs in the weight updates. The cost ratios are set in favor of the positive class, implying that relatively more effort is taken to correctly classify positive bags. The authors note that the real cost ratios are generally unavailable and advise the heuristic use of the imbalance ratio as cost ratio. Four cost sensitive boosting are proposed, similar to the single-instance cost-sensitive boosting algorithms from [39]:

Ab1: $K_t(X_i, y_i) = \exp(-C_i \alpha_t y_i h_t(X_i))$
Ab2: $K_t(X_i, y_i) = C_i \exp(-\alpha_t y_i h_t(X_i))$
Ab3: $K_t(X_i, y_i) = C_i \exp(-C_i \alpha_t y_i h_t(X_i))$
Ab4: $K_t(X_i, y_i) = C_i^2 \exp(-C_i^2 \alpha_t y_i h_t(X_i))$

The values C_i are the cost items associated with the bags, where bags of the same class are associated with the same costs. The experimental work of [5,6] showed that the Ab3 version performed best among the four alternatives. We briefly note two shortcomings of the cost-sensitive approaches. Firstly, as stated above, the cost ratio usually has to be heuristically set to the imbalance ratio, since the actual differences in misclassification costs of the classes are rarely available. Secondly, within a class, all bags are assigned the same cost, while the misclassification of a noisy sample should probably be attributed less importance compared to a typical sample of that class.

The authors of [7] proposed a preprocessing method to improve the classification of imbalanced multi-instance data as well. They construct a function which estimates the degree to which a descriptive instance, not a bag, can be considered as negative. This measure is used to locate likely and unlikely positive elements within positive bags and use them in resampling procedures. The method consists of three main steps: (1) oversampling instances within positive bags, (2) undersampling within positive bags and (3) undersampling within negative bags. All steps occur at the instance-level. The oversampling step is based on SMOTE. In the undersampling procedure, the decision criterion to remove or retain instances is based on the amount of opposite-class instances among their nearest neighbors.

3. Fuzzy rough ordered weighted average approach to imbalanced classification

In this section, we recall the proposal of [17] of a fuzzy rough classifier dealing with class imbalance in single-instance problems. The strength of this method relies on its use of class-dependent weighting schemes, which are described in detail in Section 3.2.

3.1. The IFROWANN algorithm

IFROWANN [17], Imbalanced Fuzzy Rough Ordered Weighted Average Nearest Neighbor Classification, is an extension of the FRNN classifier of [10] addressing the class imbalance problem in single-instance classification. FRNN is a nearest neighbor classifier based on fuzzy rough set theory.

To classify an unseen instance x, FRNN makes use of the fuzzy rough lower and upper approximations of the decision classes. In general, the membership degrees of x to the lower and upper approximations of class C are respectively defined as

$$\underline{C}(x) = \min_{y \in T}[\mathcal{I}(R(x,y), C(y))] \tag{3}$$

and

$$\overline{C}(x) = \max_{y \in T}[\mathcal{T}(R(x,y), C(y))]. \tag{4}$$

Here, T is the training set and $R(\cdot,\cdot)$ is a given fuzzy relation expressing similarity between the instances, taking on values in the range $[0,1]$. The function $C(\cdot)$ corresponds to the characteristic function of class C, that is, it takes on value 0 when an instance does not belong to C and 1 when it does. The operator $\mathcal{I}$ is an implicator, a generalization of traditional Boolean implication. An implicator $\mathcal{I}: [0,1]^2 \to [0,1]$ is decreasing in its first argument, increasing in the second and satisfies the boundary conditions $\mathcal{I}(1,0) = 0$, $\mathcal{I}(1,1) = 1$, $\mathcal{I}(0,0) = 1$ and $\mathcal{I}(0,1) = 1$. Similarly, the operator $\mathcal{T}$ is a triangular norm (t-norm). Fuzzy set theory defines a t-norm as an associative and commutative operator from the unit square to the unit interval, that satisfies $\mathcal{T}(x,1) = x$, $\forall x \in [0,1]$.

In the spirit of a nearest neighbor algorithm, FRNN first locates the K nearest neighbors of x in the stored training set T. Based on this set NN of nearest neighbors, the membership degrees of x to the lower and upper approximations of all decision classes are calculated, that is,

$$\underline{C}(x) = \min_{y \in NN}[\mathcal{I}(R(x,y), C(y))] \tag{5}$$

and

$$\overline{C}(x) = \max_{y \in NN}[\mathcal{T}(R(x,y), C(y))]. \tag{6}$$

To finally classify x, FRNN predicts its membership degree $C(x)$ by taking its average membership to the lower and upper approximations of C:

$$C(x) = \frac{\underline{C}(x) + \overline{C}(x)}{2}. \tag{7}$$

It assigns x to the class for which this value is largest.

IFROWANN extends FRNN in two ways. First, it reduces its sensitivity to noise. It was shown in [40] that the classification of

FRNN depends on only one element, namely the closest sample to the test instance, such that small changes in the data can result in considerably different results. To deal with this issue, IFROWANN steers away from the neighborhood sets N and uses ordered weighted average operators (OWA, [18]) in the definition of the approximation operators (3) and (4), as proposed in [41]. The OWA aggregation of a sequence V of m scalar values is computed by first ordering V in decreasing order, then weighting their values according to their ordered position by a weight vector $W=\langle w_1,\ldots,w_m\rangle$, such that $\sum_{i=1}^{m} w_i=1$ and $w_i\in[0,1]$ for all $i\in\{1,\ldots,m\}$, and finally taking their weighted average. In particular, if c_i represents the ith largest value in V, we find

$$\mathrm{OWA}_W(V)=\sum_{i=1}^{m} w_i c_i.$$

The flexibility of OWA allows for a wide range of aggregation strategies. For example, the standard minimum and maximum operators can be modeled by the weight vectors $\langle 0,0,\ldots,0,1\rangle$ and $\langle 1,0,\ldots,0,0\rangle$ respectively. Softened versions of the maximum and minimum operators can be obtained by assigning non-zero weights to other values in the ordered sequence as well. In this way, not only one value, the extreme, is taken into account, but several values are considered, resulting in more robust operators. In definitions (3) and (4), IFROWANN replaces the minimum and maximum by OWA aggregations with weight vectors that soften these operators.

The second IFROWANN improvement over FRNN is directed to counteract the class imbalance problem. It consists of using different weight aggregations in the fuzzy rough approximation for the two classes. It was shown in [17] that an increased classification performance is obtained by making the choice of the weight vectors class-dependent in the OWA-weighted approximations. In binary classification problems with classes P and N, the lower approximations are given respectively by

$$\underline{P}_{W_P}(x)=\mathrm{OWA}_{W_P,\,y\in T}[\mathcal{I}(R(x,y),P(y))] \tag{8}$$

and

$$\underline{N}_{W_N}(x)=\mathrm{OWA}_{W_N,\,y\in T}[\mathcal{I}(R(x,y),N(y))], \tag{9}$$

where W_P and W_N are the class-dependent weight vectors for the lower approximations. IFROWANN's classification rule only takes into account the lower approximations, because in a two-class problem the upper approximation of a class equals the lower approximation of the other class. Accordingly, IFROWANN assigns a test object x to the positive class when $\underline{P}_{W_P}(x)\geq\underline{N}_{W_N}(x)$ and to the negative class otherwise. We describe the weight vectors used by IFROWANN in a separate section below for clarity, as we later refer to them on multiple occasions. We will demonstrate that the vectors introduced in [17] in the context of single-instance classification are also very effective in multi-instance classification.

3.2. Class dependent OWA weight vectors

The authors of [17] consider two types of weight vectors for the negative class:

$$W_N^1=\left\langle \underbrace{0,\ldots,0}_{n},\frac{2}{p(p+1)},\frac{4}{p(p+1)},\ldots,\frac{2(p-1)}{p(p+1)},\frac{2}{p+1}\right\rangle,$$

$$W_N^2=\left\langle \underbrace{0,\ldots,0}_{n},\frac{1}{2^p-1},\frac{2}{2^p-1},\ldots,\frac{2^{p-2}}{2^p-1},\frac{2^{p-1}}{2^p-1}\right\rangle,$$

where p and n are the number of positive and negative training examples in the dataset respectively. On the other hand, they construct three types of weight vectors for the positive class:

$$W_P^1=\left\langle \underbrace{0,\ldots,0}_{p},\frac{2}{n(n+1)},\frac{4}{n(n+1)},\ldots,\frac{2(n-1)}{n(n+1)},\frac{2}{n+1}\right\rangle,$$

$$W_P^2=\left\langle \underbrace{0,\ldots,0}_{p},\frac{1}{2^n-1},\frac{2}{2^n-1},\ldots,\frac{2^{n-2}}{2^n-1},\frac{2^{n-1}}{2^n-1}\right\rangle,$$

$$W_P^{1,\gamma}=\left\langle \underbrace{0,\ldots,0}_{p+n-r},\frac{2}{r(r+1)},\frac{4}{r(r+1)},\ldots,\frac{2(r-1)}{r(r+1)},\frac{2}{r+1}\right\rangle.$$

The value r is determined as $r=\lceil p+\gamma(n-p)\rceil$ and is used to limit the number of non-zero weights in the aggregation. In [17], the default value $\gamma=0.1$ was proposed.

Six combinations of weight vectors were put forward and evaluated in [17]:

1. $\mathcal{W}_1=\langle W_P^1,W_N^1\rangle$
2. $\mathcal{W}_2=\langle W_P^1,W_N^2\rangle$
3. $\mathcal{W}_3=\langle W_P^2,W_N^1\rangle$
4. $\mathcal{W}_4=\langle W_P^2,W_N^2\rangle$
5. $\mathcal{W}_5=\langle W_P^{1,\gamma},W_N^1\rangle$, with $\gamma=0.1$.
6. $\mathcal{W}_6=\langle W_P^{1,\gamma},W_N^2\rangle$, with $\gamma=0.1$.

In the experimental study of [17], weighting schemes $\mathcal{W}_4$ and $\mathcal{W}_6$ obtained the best classification results. As part of our experimental comparison conducted in Section 5, we verify whether this conclusion also holds in our proposal.

4. Fuzzy rough multi-instance classifiers for imbalanced classification

Based on the IFROWANN algorithm for single-instance classification, we present a framework for fuzzy rough multi-instance classification algorithms resistant to class imbalance. In general, we define a *fuzzy rough multi-instance classifier* as $\mathcal{F}_{FRM}:\mathbb{N}^{\mathcal{X}}\to\mathcal{Y}$ such that

$$\mathcal{F}_{FRM}(X)=\arg\max_{C\in\mathcal{Y}}\Phi(\underline{C}(X),\overline{C}(X)), \tag{10}$$

where $\underline{C}(X)$ (resp. $\overline{C}(X)$) is the membership degree of bag X to the lower (resp. upper) approximation of class C and Φ is an aggregating function of both $\underline{C}(X)$ and $\overline{C}(X)$. In this paper, we make the choice to set $\Phi(\underline{C}(X),\overline{C}(X))=\underline{C}(X)$. Although the information contained in the upper approximation is discarded, this was an effective choice for the original IFROWANN method, such that we provide a faithful extension of the latter here. As done by the original method, we assign a bag X to the positive class in case of a draw in the computed values $\underline{C}(X)$.

There are different ways to obtain the membership degree $\underline{C}(X)$ of a bag X to the lower approximation $\underline{C}$ of a class C. We propose two families of multi-instance fuzzy rough classifiers, which differ in that one is based on relationships between the bags, considering the bag as a whole, while the other is based on information derived from the instances. Classifiers in the first family are *bag-based* (Section 4.1), while those in the second family are *instance-based* (Section 4.2). A visual overview of the flow of the two families, that is, how the corresponding methods derive the values $\underline{C}(X)$, is presented in Fig. 1.

4.1. Fuzzy rough bag-based multi-instance classifiers

The calculation of $\underline{C}(X)$ by bag-based classifiers is executed entirely at the bag-level and is an extension of (8) and (9). A visual overview is presented in Fig. 2. The general formulation of our fuzzy rough bag-based classifiers to calculate $\underline{C}(X)$ is therefore

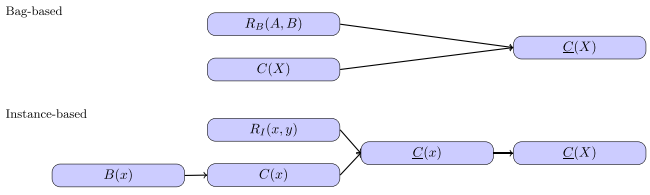

Fig. 1. Overview of the proposed framework. The figure presents the flow of both the bag-based and instance-based methods.

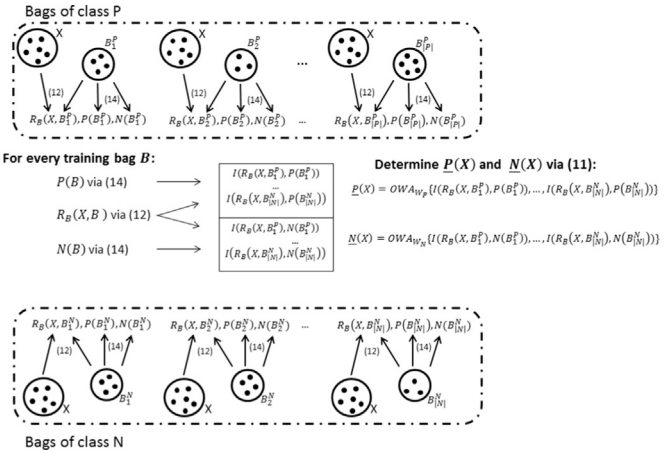

Fig. 2. Overview of the calculations of our bag-based methods when classifying a bag X. In this example, the positive class P consists of bags $B^P_1,\ldots,B^P_{|P|}$ and the negative class N contains bags $B^N_1,\ldots,B^N_{|N|}$.

given by

$$\underline{C}(X) = \underset{Y \in T}{\mathrm{OWA}_{W_C}}[\mathcal{I}(R_B(X,Y), C(Y))] \tag{11}$$

where W_C is the class-dependent weight vector for the lower approximation of the class C, which can be set to the weights from each combination listed in Section 3.2. Note that the aggregation is taken over all the bags from the training set T. The implicator $\mathcal{I}$ used in this paper is the Łukasiewicz implicator, defined as

$$(\forall a,b \in [0,1])(\mathcal{I}(a,b) = \min(1-a+b,1)).$$

Other fuzzy implicators can be used as well. The relation $R_B(\cdot,\cdot)$ represents the bag-wise similarity. We define this measure as the complement to 1 of the average Hausdorff distance [42], a popular distance measure in MIL, that is,

$$R_B(A,B) = 1 - \frac{\sum_{a\in A}\min_{b\in B}[\delta(a,b)] + \sum_{b\in B}\min_{a\in A}[\delta(a,b)]}{|A|+|B|}, \tag{12}$$

where $\delta(\cdot,\cdot) \in [0,1]$ is a normalized distance function between instances. In this paper, we use the cosine distance, which is defined as

$$\delta(x,y) = 1 - \frac{x\cdot y}{\|x\|\,\|y\|}, \tag{13}$$

where $\|x\|$ represents the vectorial norm of x. The cosine distance is the complement of the cosine similarity, which has a low computational cost and is widely used in machine learning, especially in textual applications.

The final missing piece in the calculation of (11) is the definition of the class membership degrees $C(Y)$ of bags to classes. This term is an estimate of the membership degree of the training bag Y to class C. Here, we take a step away from the single-instance IFROWANN method, which uses crisp membership degrees of instances to classes. The simplest approach is indeed to assign $C(Y)=1$ if Y is labeled with class C and $C(Y)=0$ otherwise. However, it was already shown in [43] that more elaborate estimates can provide better results. Several methods can be applied to compute the membership degree of bags to classes. In this paper, we limit ourselves to two ways to compute the membership degree $C(Y)$ of a bag Y to a class C, but we stress that alternatives to do so can be easily plugged in. We consider the use of an OWA aggregation of the similarities between the given bag Y to training bags belonging to class C, that is,

$$C(Y) = \underset{B\in T_C}{\mathrm{OWA}_W}[R_B(Y,B)], \tag{14}$$

where T_C is the set of training bags labeled with class C. We note that these values are calculated for training bags Y only, for which the class label is actually known. Assume that we have two classes C_1 and C_2 in the datasets and that Y is labeled with C_1. As stated above, we do not simply set $C_1(Y)=1$ and $C_2(Y)=0$. Instead, to determine $C_1(Y)$ we select all training bags of class C_1, meaning that Y itself is among them, and compute their similarity to Y. The value $R_B(Y,Y)$ that is included in this calculation will always be 1, which is the maximum value that the similarity measure can attain. Nevertheless, by also assigning some weight to the similarity of Y with other bags of this class, the final value $C_1(Y)$ can be lower than 1. To determine $C_2(Y)$, the analogous steps as for C_1 are performed, although $R_B(Y,Y)$ will clearly not partake in the aggregation here. This procedure counteracts the effects of noise, as an atypical bag Y labeled with class C_1 will receive a low membership degree $C_1(Y)$. This motivates our use of the softened rather than strict maximum operator by means of the OWA aggregations. Furthermore, since bags within the same class can still be very different, the softened maximum can also be preferred over the average, which assigns equal weights to all values $R_B(Y,B)$. Indeed, let us consider the discussion in Section 2.1 on the standard multi-instance hypothesis. If we have two positive bags, each consisting of ten instances, the first one can consist of one positive and nine negative instances, while the second bag contains ten positive instances. Even though they belong to the same class, we can expect their similarity $R_B(\cdot,\cdot)$ to be low. If a bag Y is very different from all other bags in its own class, we can expect it to be noisy, but if Y still has a large similarity with some of them, it probably remains a proper example of the class. This situation is modeled by the OWA aggregation, by letting the aggregation weights depend on the similarity with Y, but it cannot be handled by using the average operator.

As OWA weight vectors W in (14), we evaluate two versions of a softened maximum operator. Firstly, we consider a vector with linearly decreasing weights

$$W_{\mathrm{L}} = \left\langle \frac{2}{m+1}, \frac{2(m-1)}{m(m+1)}, \ldots, \frac{4}{m(m+1)}, \frac{2}{m(m+1)} \right\rangle, \tag{15}$$

where m is the number of values to be aggregated, which is the number of training bags of class C in this case. The vector W_{L} is a normalized version of $\langle m, m-1, \ldots, 2, 1\rangle$ and corresponds to the Borda count or law of Borda–Kendall in decision making [44].

The second version contains inverse additive weights and is given as

$$W_{\mathrm{IA}} = \left\langle \frac{1}{\sum_{i=1}^{m}\frac{1}{i}}, \frac{1}{2\sum_{i=1}^{m}\frac{1}{i}}, \ldots, \frac{1}{(m-1)\sum_{i=1}^{m}\frac{1}{i}}, \frac{1}{m\sum_{i=1}^{m}\frac{1}{i}} \right\rangle. \tag{16}$$

The name of this vector refers to the occurrence of $\sum_{i=1}^{m} 1/i$ in the denominator. It was shown to be robust against noise in [45], which is why we include it here. Other alternatives would include the use of a weight vector with exponential weights similar to $W_N{}^2$ and $W_P{}^2$ defined in Section 3.2 or using another aggregation operator, like the maximum or average. In this study, when using (15), we refer to (14) as OWAL aggregation. In the other case, we denote it as OWAIA aggregation.

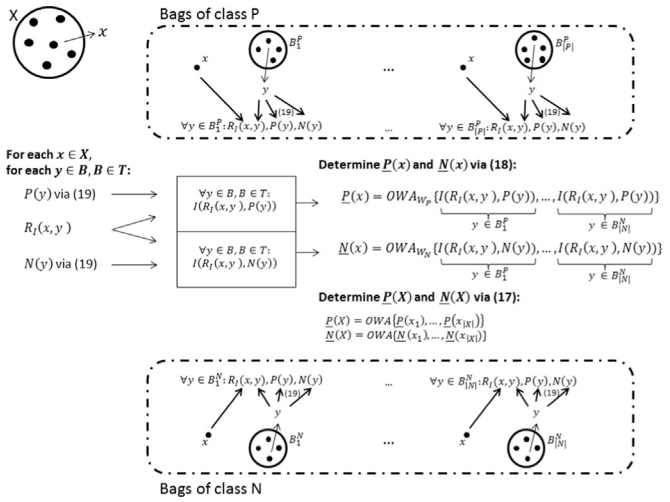

Fig. 3. Overview of the calculations of our instance-based methods when classifying a bag X. In this example, the positive class P consists of bags $B_1^P, \ldots, B_{|P|}^P$ and the negative class N contains bags $B_1^N, \ldots, B_{|N|}^N$. The instances in the unlabeled bag X are denoted as $x_1, \ldots, x_{|X|}$.

4.2. Fuzzy rough instance-based multi-instance classifiers

Our instance-based classifiers determine the value $\underline{C}(X)$ by aggregating the corresponding values $\underline{C}(x)$ for all instances $x \in X$. Accordingly, the general formulation of fuzzy rough instance-based multi-instance classifiers to calculate $\underline{C}(X)$ is given as

$$\underline{C}(X) = \underset{x \in X}{Agg}[\underline{C}(x)] \tag{17}$$

with

$$\underline{C}(x) = \underset{y \in B, B \in T}{\text{OWA}_{W_C}}[\mathcal{I}(R_I(x,y), C(y))]. \tag{18}$$

where W_C is the class-dependent weight vector for the lower approximation of class C. Again, W_C can take on weights from each combination listed in Section 3.2. An overview of the classification procedure of our instance-based methods is presented in Fig. 3. The aggregation in (18) is taken over all instances from all training bags. The term $R_I(\cdot,\cdot)$ represents the similarity relation between instances, where we again use the cosine similarity. *Agg* represents an aggregation method over all the instances $x \in X$. We consider OWA aggregation for *Agg*, using the softened maximum weight vectors (15) and (16). We denote the former as OWAL aggregation and the latter as OWAIA aggregation. As before, our use of this type of aggregation is motivated by the nature of multi-instance data and the multi-instance hypotheses referenced in Section 2.1. Not all instances x in a bag X should contribute equally to the prediction $\underline{C}(X)$, as we cannot expect all of them to be affiliated with this class. Using a strict maximum operator can result in discarding some crucial information, while the application of an average operator may cause opposite-class instances to cancel out the membership degrees of the instances in the bag actually belonging to a class C.

The term $C(y)$ in (18) is an estimate of the membership degree to class C of the instance y, belonging to some training bag. Referring back to Section 2, simply imposing the class label of the bag on all its instances is not suitable for multi-instance data. Using that procedure would make the estimation of $\underline{C}(x)$ coincide with the single-instance IFROWANN method, but the structural differences in single-instance and multi-instance data warrant us to take a clear step away from the original proposal. Several heuristics can be used to determine $C(y)$. In our experiments, we use the inverse additive OWA maximum membership $B(y)$ of y to training bags B from class C,

$$C(y) = \underset{B \in T_C}{\text{OWA}_{W_{IA}}}[B(y)]. \tag{19}$$

Here, we compute $B(y)$, given a similarity relation $R_I(\cdot,\cdot)$, as the maximum similarity of y with any instance in the bag B,

$$B(y) = \max_{z \in B}[R_I(y,z)]. \tag{20}$$

As should be clear from (19), we do not simply assign the label of its parent bag to y, but opt to use all the information in the training set. We evaluate the affinity $B(y)$ of the instance to all training bags of a class, regardless of whether these bags contain the instance or not. By aggregating them, we find the final class membership degree. Our motivation to use an OWA aggregation in this step is similar as the one given with expression (14).

We stress that different heuristics can be used to compute the membership degree $\underline{C}(X)$ of a bag X to the lower approximation of class C, the membership degree $C(x)$ of an instance x to a class C and the membership degree $B(x)$ of an instance x to a training bag B and be plugged in to evaluate (17). Likewise, several alternatives may be used to calculate the membership degree $C(X)$ of a bag X to a class C in Eq. (11). The selection made here is motivated by a preliminary empirical study.

4.3. Discussion: weight assignment and differences with cost-sensitive methods

We want to stress that our methods are inherently different from the cost-sensitive approaches discussed in Section 2.3. In the Ab1, Ab2, Ab3 and Ab4 algorithms, training bags are assigned a cost, representing how severe we consider their misclassification. For each bag, this value can be interpreted as its weight in the construction of the classification model. As noted in Section 2.3, all bags from the same class are assigned the same cost. These costs are fixed at the start of the algorithm.

The methods from our proposed framework model class membership of unseen bags by means of expressions (11) (bag-based) or (17) and (18) (instance-based). These values are used in the final prediction process (10). The OWA aggregations assign weights to the training bags in (11) and to training instances in (18). The weight of a particular training bag Y or instance y can be different in every prediction procedure and depends on the unseen bag X or instance x at hand. More specifically, consider expression (11). To classify the unseen bag, the contribution of the training bags Y is ranked and weighted based on the fuzzy implication of $R_B(X,Y)$ and $C(Y)$. The former is dependent on X itself and this can result in a different weight being assigned to the same bag Y for all new bags X. Furthermore, all training bags are assigned a different weight in the OWA-step, even when they belong to the same class. Analogous remarks hold for (18). Summarizing, the weights assigned in our methods can differ among all bags and instances, not just among classes, and they are adaptive. This is clearly different from the setup of the cost-sensitive methods.

5. Experiments

In this section, we present the experimental evaluation of our proposal. In Section 5.1, we discuss the setup of the experiments. The evaluation itself is divided into two main parts. Firstly, in Sections 5.2 and 5.3 we compare the different weighting schemes for the proposed bag-based and instance-based classifiers respectively. The second part, presented in Section 5.4, provides a comparison of our proposal to previously introduced multi-instance classifiers handling class imbalance.

5.1. Experimental setup

Table 1 lists all datasets included in our experiments. Among these datasets, 26 (those above the line: WIRSel and Corel) are used in Sections 5.2 and 5.3 to evaluate the different weighting schemes within the two families of classifiers. In order to make the comparison with state-of-the-art methods in Section 5.4, we use all the datasets listed in Table 1. In total, there are 34 datasets from different application domains, namely textual, image and pharmaceutical applications. Their IR ranges from 2.98 to 19.0.

As evaluation measures, we use the Area Under the ROC-Curve (AUC, [46]) and the geometric mean of the class-wise accuracies (GMean). Both are commonly used to evaluate classifier performance in the context of class imbalance (e.g. [47]). All reported results are obtained by five-fold cross validation. We use the Wilcoxon test [48] to check for statistical significance in the observed differences in performance of two classifiers. This is a non-parametric test, which ranks the differences in performances of two classifiers for each dataset, ignoring the signs, and compares the ranks for the positive and the negative differences. The p-value calculated by the Wilcoxon test represents the probability of obtaining a result at least as extreme as that obtained in the experiment, assuming that the two classifiers have similar performance (null hypothesis). A p-value smaller than a given significance level α suggests that the null hypothesis is false, i.e., there are statistically significant differences between the compared methods. In this paper, we use $\alpha = 0.05$. Non-parametric tests are preferred over parametric alternatives [49,50].

5.2. Bag-based classifiers

Figs. 4 and 5 present the results of OWAL and OWAIA bag-to-class aggregations respectively. Taking the two metrics into account, both figures show the superiority of weighting schemes $\mathcal{W}_4$ and $\mathcal{W}_5$. The former was also among the best performing ones in the original IFROWANN proposal [17]. In their experimental work, scheme $\mathcal{W}_6$ also attained good results, but this does not hold in our case. The difference between schemes $\mathcal{W}_5$ and $\mathcal{W}_6$ lies in their aggregation weights for the negative class. In multi-instance classification, the exponential vector used by $\mathcal{W}_6$ seems to be a less optimal combination with the linearly increasing weight vector for the positive class. We note that $\mathcal{W}_4$ also makes use of the exponential vector for the negative class, but uses an analogous vector for the positive class as well. This results in a balanced approximation of the two classes, which is reflected in the classification results.

Table 1
Datasets used in the experimental study.

Name	# Att.	# Bag	IR	Name	# Att.	# Bag	IR
WIRSel-1	304	113	4.38	Corel8	9	2000	19.00
WIRSel-2	298	113	4.38	Corel9	9	2000	19.00
WIRSel-3	303	113	4.38	Corel10	9	2000	19.00
WIRSel-4	303	113	3.71	Corel11	9	2000	19.00
WIRSel-5	302	113	3.71	Corel12	9	2000	19.00
WIRSel-6	304	113	3.71	Corel13	9	2000	19.00
Corel1	9	2000	19.00	Corel14	9	2000	19.00
Corel2	9	2000	19.00	Corel15	9	2000	19.00
Corel3	9	2000	19.00	Corel16	9	2000	19.00
Corel4	9	2000	19.00	Corel17	9	2000	19.00
Corel5	9	2000	19.00	Corel18	9	2000	19.00
Corel6	9	2000	19.00	Corel19	9	2000	19.00
Corel7	9	3000	19.00	Corel20	9	2000	19.00
Thioredoxin	8	193	6.72	Function	200	5242	10.83
Elephant	230	125	4.00	Atoms	10	167	2.98
Fox	230	121	4.76	Bonds	16	160	3.57
Tiger	230	126	3.85	Chains	24	152	4.63

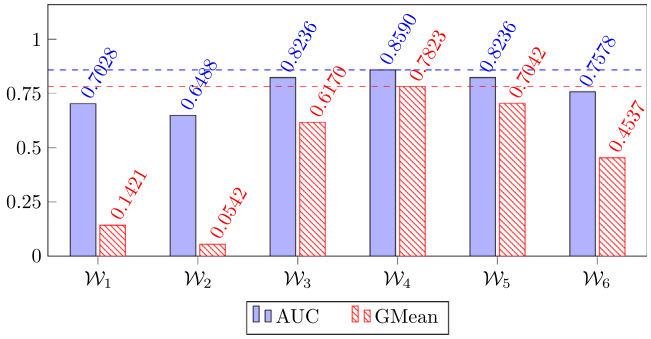

Fig. 4. Ranking of bag-based classifiers using OWAL aggregation based on their AUC and GMean. The horizontal lines correspond to the highest values attained for the AUC (blue) and GMean (red). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

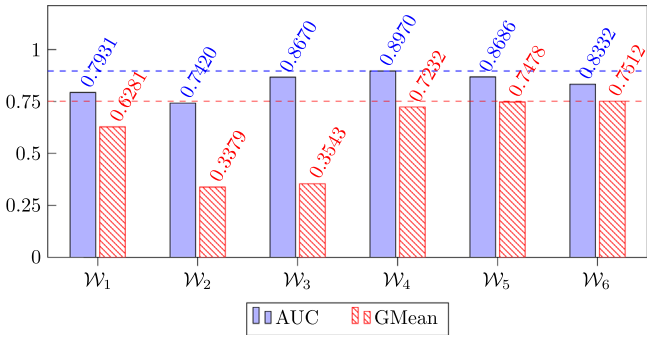

Fig. 5. Ranking of bag-based classifiers using OWAIA aggregation based on their AUC and GMean. The horizontal lines correspond to the highest values attained for the AUC (blue) and GMean (red). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

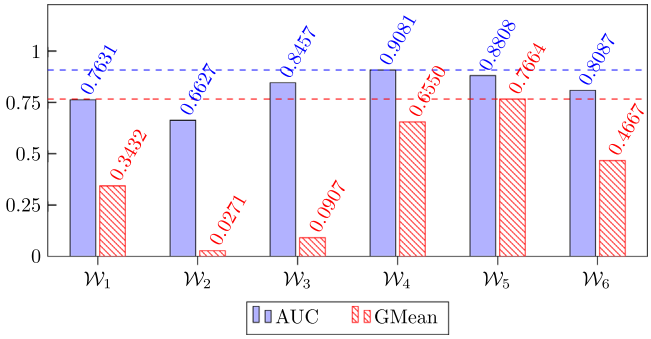

Fig. 6. Ranking of instance-based classifiers using OWAIA aggregation based on their AUC and GMean. The horizontal lines correspond to the highest values attained for the AUC (blue) and GMean (red). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

5.3. Instance-based classifiers

The results of the OWAIA bag-to-class-approximation aggregation can be found in Fig. 6. The same conclusions as in Section 5.2 can be drawn: weighting schemes $\mathcal{W}_4$ and $\mathcal{W}_5$ yield the best results.

For the OWAL aggregation, we took the comparison a little bit further, varying the γ parameter in schemes $\mathcal{W}_5$ and $\mathcal{W}_6$ instead of using its default value $\gamma = 0.1$ recommended in [17]. We evaluated $\gamma = 0.05, 0.1, 0.2, 0.3, 0.4, 0.5$. The results are presented in Fig. 7. We again find $\mathcal{W}_4$ and $\mathcal{W}_5$ on top and observe that $\mathcal{W}_5$ attains good results for various values of γ. The performance of $\mathcal{W}_6$ improves by lowering the value of γ to 0.05, but this does not bring it to the same level as $\mathcal{W}_4$ and $\mathcal{W}_5$. An important observation is that $\mathcal{W}_4$ has

a slight dominance over $\mathcal{W}_5$ with respect to their AUC, but considering the GMean, a large gap is observed between the results of $\mathcal{W}_5$ and that of $\mathcal{W}_4$. This difference can be explained as follows. The higher AUC value of $\mathcal{W}_4$ suggests that it could outperform $\mathcal{W}_5$, if the classification decision procedure were to be changed. Indeed, AUC provides a global picture of the classification strength of a method among different thresholds on the predicted class probability above which an instance is classified as positive, that is, among different decision procedures. The GMean considers the classification outcome associated with one particular, chosen procedure. The results show that scheme $\mathcal{W}_5$ seems to combine better with our selected class assignment method compared to $\mathcal{W}_4$. Taking both evaluation measures into account, $\mathcal{W}_5$ might therefore be favored over $\mathcal{W}_4$ for the instance-based classifiers.

In order to evaluate whether the class-dependency of the weight vectors indeed leads to an improvement in classification performance, we also consider versions of the instance-based classifiers in which the weight vectors do not differ for the two classes. We fix the aggregation to OWAL aggregation, but replace the use of OWA_{W_c} in (18) by one of three class-independent weighting schemes. In the first one, we use the traditional minimum operator. We refer to this version as STDmin. Secondly, we consider OWA aggregation by means of a weight vector softening the minimum. We use the two alternatives of linear or inverse additive weights, but take care to reverse the order of the weights given in (15) and (16), as we now require a softening of the minimum rather than of the maximum. These two schemes are referred to as OWALmin and OWAIAmin respectively. In Fig. 8, we compare the fuzzy rough instance-based classifiers using class-dependent weight vectors with these three alternatives. Classifiers using $\mathcal{W}_4$ and $\mathcal{W}_5$ obtained the best results with respect to AUC,

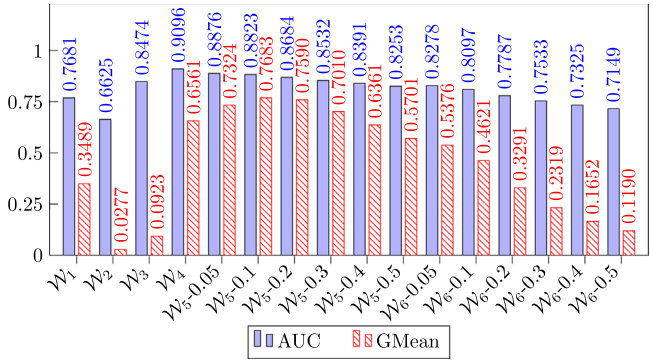

Fig. 7. Ranking of instance-based classifiers using OWAL aggregation based on their AUC and GMean.

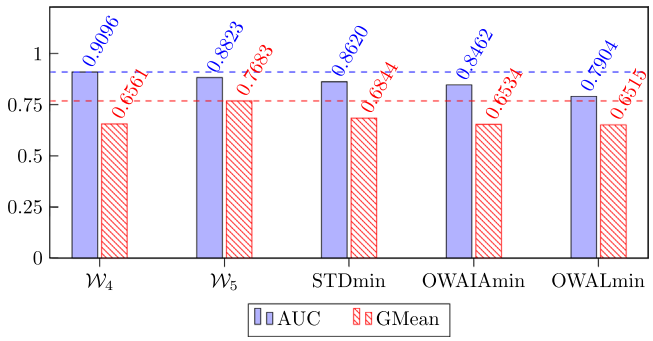

Fig. 8. Ranking of FRI-OWAL classifiers using class-dependent and class-independent weights. The horizontal lines correspond to the highest values attained for the AUC (blue) and GMean (red). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

while regarding GMean the one using $\mathcal{W}_5$ performs best. The difference in the two evaluation measures has been discussed above. The figure shows that our current proposals yield the best results, but we must recognize that even those classifiers using the same weight vector for both classes show a fairly good performance, which demonstrates that our fuzzy rough classifiers are inherently robust to class imbalance. Among these classifiers, the best performance is achieved with the standard minimum operator. It is followed by the OWA operator with inverse additive weights, whose weight distribution is closer to the standard minimum than that of the linearly increasing weights model.

5.4. Comparison with the state-of-the-art

In this section, we compare our proposal to state-of-the-art multi-instance classification methods for class imbalanced data. Note that we use the full set of 34 datasets in Table 1 in this comparison. Among our fuzzy rough classifiers, there are several that stand out for their good performance, like those bag-based using $\mathcal{W}_4$ and $\mathcal{W}_5$ in Figs. 4 and 5 and those instance-based using $\mathcal{W}_4$ and $\mathcal{W}_5$ in Figs. 6 and 7. We select one representative of each family: the bag-based classifier using weighting scheme $\mathcal{W}_4$ and the instance-based classifier with scheme $\mathcal{W}_5$ ($\gamma = 0.1$). Both use OWAL aggregation. In the remainder, we denote them as FRB and FRI for short.

We compare these methods to state-of-the-art methods described in Section 2.3. We select BagSMOTE as a representative preprocessing method, as well as the cost-sensitive boosting algorithms Ab1, Ab2, Ab3 and Ab4. In the experiments, we use the decision tree learning algorithm MITI [51] as multi-instance base classifier for both BagSMOTE and the boosting methods, as recommended by [6].

Table 2 presents the AUC and GMean results. Our proposals appear in the top position for both evaluation measures: FRI performs best for AUC, FRB for GMean. For the other measure, they each appear in third place. We also observe that the results exhibit a relatively low variance over the datasets, meaning that our methods perform consistently well. Depending on whether the application warrants the optimization of AUC or GMean, FRI or FRB can be selected. The difference between the two measures has been discussed above: AUC represents the behavior of the method among a range of decision procedures, while GMean considers the classification outcome at hand. A multi-instance classification procedure consists of assigning labels to unseen bags. The better GMean of the bag-based classifier FRB could intuitively be explained by its natural extraction of this assignment from information solely derived from the training bags, not their instances. However, the higher AUC value of the instance-based classifier FRI shows that aggregating instance-level predictions might even improve upon this, provided an adjusted decision procedure is used.

As in [5,6], we can conclude that Ab3 performs best among the alternative cost-sensitive boosting algorithms. It attains a good

Table 2
AUC and GMean results for the classifiers. We include the standard deviation of these values taken over all datasets.

Method	AUC	Method	GMean
FRI	0.8568 ± 0.0875	FRB	0.7364 ± 0.1391
Ab3	0.8309 ± 0.1038	BagSMOTE	0.7227 ± 0.1505
FRB	0.8190 ± 0.1155	FRI	0.7167 ± 0.1752
Ab4	0.7798 ± 0.0965	Ab3	0.5383 ± 0.2005
Ab1	0.7757 ± 0.1076	Ab1	0.4921 ± 0.2367
BagSMOTE	0.7536 ± 0.1066	Ab2	0.3654 ± 0.2493
Ab2	0.7230 ± 0.1450	Ab4	0.0455 ± 0.1184

Table 3
Results of the Wilcoxon test. For significance level $\alpha = 0.05$, p-values implying significant differences are printed in bold.

Measure	Comparison	R^+	R^-	p-value
AUC	FRI vs Ab3	405.0	190.0	0.064831
AUC	FRI vs BagSMOTE	591.0	4.0	$\leq$ **0.000001**
GMean	FRB vs Ab3	529.0	66.0	**0.000073**
GMean	FRB vs BagSMOTE	308.0	287.0	0.850828

Table 4
AUC and GMean results for the classifiers, taken as averages over the 8 datasets below the horizontal line in Table 1. We include the standard deviation of these values taken over all datasets.

Method	AUC	Method	GMean
FRI	0.7742 ± 0.1166	FRB	0.5870 ± 0.1902
Ab3	0.7495 ± 0.1200	BagSMOTE	0.5718 ± 0.1846
Ab4	0.7268 ± 0.1093	FRI	0.5492 ± 0.2547
Ab1	0.7081 ± 0.1243	Ab3	0.5300 ± 0.2528
FRB	0.6892 ± 0.1436	Ab1	0.4238 ± 0.3074
BagSMOTE	0.6617 ± 0.1088	Ab2	0.2004 ± 0.2329
Ab2	0.6179 ± 0.1395	Ab4	0.1427 ± 0.1881

AUC result, but, although outperforming its relatives, looses many points when considering the GMean. The BagSMOTE preprocessing algorithm obtains good GMean values, but its performance with regard to AUC is poor. Since our proposals obtain good results for both measures, they can clearly be preferred.

For each evaluation measure, we used the Wilcoxon test to determine whether significant differences are present between the best performing method (either FRI or FRB) and its competitors Ab3 and BagSMOTE. The results of this analysis are presented in Table 3. We can conclude that our proposal, represented by FRI, is particularly strong for the AUC measure. For GMean, FRB is shown to significantly outperform Ab3. FRB obtains equivalent GMean results as BagSMOTE, but since the performance of the latter with regard to AUC is very poor, we can still conclude that FRB can be preferred over BagSMOTE.

Lastly, we consider the comparison of this group of methods with respect to the eight datasets below the horizontal line in Table 1. These datasets were not used in Sections 5.2 and 5.3 and our selected classifiers FRB and FRI are therefore not optimized for them. Nevertheless, Table 4 shows that they also perform well in this situation. As before, FRI attains the highest AUC value and FRB performs best for GMean. Roughly the same ranking of methods as in Table 2 can be observed, apart from a slight drop in performance of FRB with respect to the AUC, with Ab1 and Ab4 now also outperforming our method for this measure. However, its vast dominance over these methods considering the GMean shows that FRB is still preferred over them.

6. Concluding remarks

Class imbalance is encountered in several multi-instance applications, but has been little studied in the literature so far. In this paper, we developed an extension of the successful single-instance classification method IFROWANN to the multi-instance setting. We proposed two classifier families, one at bag-level and one at instance-level. The classifiers are based on fuzzy rough set theory and their decision criterion relies on the predicted membership degree of an unseen bag to the lower approximation of the classes.

The defining characteristic of the proposal is its use of class-dependent weight vectors in OWA aggregations. We experimentally compared several weighting schemes and were able to put forward the best performing ones. Furthermore, our experiments showed that our fuzzy rough classifiers outperform the existing proposals of multi-instance classifiers dealing with class imbalance.

We have limited ourselves to two-class problems, as is common practice in MIL and imbalanced classification. Nevertheless, multi-class imbalance also presents itself. An important future research challenge is therefore the extension of the current proposal to handle more than two classes. The key step will be the development of appropriate weighting schemes in this setting.

Conflict of interest

There is no conflict of interest.

Acknowledgments

The research of Sarah Vluymans is funded by the Special Research Fund (BOF) of Ghent University (Grant number BOF.DOC.2014.0074). This work was partially supported by the Spanish Ministry of Economy and Competitiveness under the project TIN2014-57251-P and the Andalusian Research Plans P11-TIC-7765 and P10-TIC-6858, and by project PYR-2014-8 of the Genil Program of CEI BioTic GRANADA.

References

[1] T.G. Dietterich, R.H. Lathrop, T. Lozano-Perez, Solving the multiple instance problem with axis-parallel rectangles, Artif. Intell. 89 (1–2) (1997) 31–71.
[2] H. He, E. Garcia, Learning from imbalanced data, IEEE Trans. Knowl. Data Eng. 21 (9) (2009) 1263–1284.
[3] V. López, A. Fernández, S. García, V. Palade, F. Herrera, An insight into classification with imbalanced data: empirical results and current trends on using data intrinsic characteristics, Inf. Sci. 250 (2013) 113–141.
[4] Y. Sun, A. Wong, M. Kamel, Classification of imbalanced data: a review, Int. J. Pattern Recognit. Artif. Intell. 23 (4) (2009) 687–719.
[5] X. Wang, X. Liu, N. Japkowicz, S. Matwin, Resampling and cost-sensitive methods for imbalanced multi-instance learning, in: Proceedings of the 2013 IEEE 13th International Conference on Data Mining Workshops (ICDMW), 2013, pp. 808–816.
[6] X. Wang, S. Matwin, N. Japkowicz, X. Liu, Cost-sensitive boosting algorithms for imbalanced multi-instance datasets, in: O. Zaïane, S. Zilles (Eds.), Advances in Artificial Intelligence, Springer, Regina, Canada, 2013, pp. 174–186.
[7] C. Mera, M. Orozco-Alzate, J. Branch, Improving representation of the positive class in imbalanced multiple-instance learning, in: A. Campilho, M. Kamel (Eds.), Image Analysis and Recognition, Springer, Vilamoura, Portugal, 2014, pp. 266–273.
[8] R. Duda, P. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973.
[9] X. Wu, V. Kumar, J.R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G.J. McLachlan, A. Ng, B. Liu, S.Y. Philip, et al., Top 10 algorithms in data mining, Knowl. Inf. Syst. 14 (1) (2008) 1–37.
[10] R. Jensen, C. Cornelis, Fuzzy-rough nearest neighbour classification, in: J. Peters, A. Skowron, C. Chan, J. Grzymala-Busse, W. Ziarko (Eds.), Transactions on Rough Sets XIII, Springer, Berlin, Heidelberg, 2011, pp. 56–72.
[11] D. Dubois, H. Prade, Rough fuzzy sets and fuzzy rough sets, Int. J. Gen. Syst. 17 (2–3) (1990) 191–209.
[12] L. Zadeh, Fuzzy sets, Inf. Control 8 (3) (1965) 338–353.
[13] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (5) (1982) 341–356.
[14] R. Bhatt, M. Gopal, FRCT: fuzzy-rough classification trees, Pattern Anal. Appl. 11 (1) (2008) 73–88.
[15] D. Chen, Q. He, X. Wang, FRSVMs: fuzzy rough set based support vector machines, Fuzzy Sets Syst. 161 (4) (2010) 596–607.
[16] R. Jensen, C. Cornelis, Fuzzy-rough nearest neighbour classification and prediction, Theor. Comput. Sci. 412 (42) (2011) 5871–5884.
[17] E. Ramentol, S. Vluymans, N. Verbiest, Y. Caballero, R. Bello Perez, C. Cornelis, F. Herrera, IFROWANN: imbalanced fuzzy-rough ordered weighted average nearest neighbor classification, IEEE Trans. Fuzzy Syst. 23 (5) (2015) 1622–1637.
[18] R. Yager, On ordered weighted averaging aggregation operators in multicriteria decision making, IEEE Trans. Syst. Man Cybern. 18 (1) (1988) 183–190.

[19] L. De Raedt, Attribute-value learning versus inductive logic programming: the missing links, in: D. Page (Ed.), Inductive Logic Programming, Lecture Notes in Computer Science, vol. 1446, Springer, Berlin/Heidelberg, 1998, pp. 1–8.
[20] S. Feng, W. Xiong, B. Li, C. Lang, X. Huang, Hierarchical sparse representation based multi-instance semi-supervised learning with application to image categorization, Signal Process. 94 (2014) 595–607.
[21] K. Zhang, H. Song, Real-time visual tracking via online weighted multiple instance learning, Pattern Recognit. 46 (1) (2013) 397–411.
[22] B. Babenko, M.H. Yang, S. Belongie, Robust object tracking with online multiple instance learning, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1619–1632.
[23] F.U. Minhas, A. Ben-Hur, Multiple instance learning of Calmodulin binding sites, Bioinformatics 28 (18) (2012) i416–i422.
[24] G. Fu, X. Nan, H. Liu, R.Y. Patel, P.R. Daga, Y. Chen, D.E. Wilkins, R.J. Doerksen, Implementation of multiple-instance learning in drug activity prediction, BMC Bioinform. 13 (15) (2012) 1–12.
[25] R. Teramoto, H. Kashima, Prediction of protein-ligand binding affinities using multiple instance learning, J. Mol. Graph. Model. 29 (3) (2010) 492–497.
[26] D.S. Tarragó, C. Cornelis, R. Bello, F. Herrera, A multi-instance learning wrapper based on the Rocchio classifier for web index recommendation, Knowl.-Based Syst. 59 (2014) 173–181.
[27] A. Zafra, C. Romero, S. Ventura, E. Herrera-Viedma, Multi-instance genetic programming for web index recommendation, Expert Syst. Appl. 36 (9) (2009) 11470–11479.
[28] Z. Zhou, K. Jiang, M. Li, Multi-Instance Learning Based Web Mining, Appl. Intell. 22 (2005) 135–147.
[29] S. Andrews, I. Tsochantaridis, T. Hofmann, Support vector machines for multiple-instance learning, in: Advances in Neural Information Processing Systems, vol. 15, 2003, pp. 561–568.
[30] L. Sun, Y. Lu, K. Yang, S. Li, ECG analysis using multiple instance learning for myocardial infarction detection, IEEE Trans. Biomed. Eng. 59 (12) (2012) 3348–3356.
[31] M. Popescu, A. Mahnot, Early illness recognition using in-home monitoring sensors and multiple instance learning, Methods Inf. Med. 51 (4) (2012) 359–367.
[32] S. Wang, M.T. McKenna, T.B. Nguyen, J.E. Burns, N. Petrick, B. Sahiner, R. M. Summers, Seeing is believing: video classification for computed tomographic colonography using multiple-instance learning, IEEE Trans. Med. Imaging 31 (5) (2012) 1141–1153.
[33] N. Weidmann, E. Frank, B. Pfahringer, A two-level learning method for generalized multi-instance problems, in: Machine Learning: ECML 2003, Springer, Cavtat-Dubrovnik, Croatia, 2003, pp. 468–479.
[34] J. Amores, Multiple instance classification: review, taxonomy and comparative study, Artif. Intell. 201 (2013) 81–105.
[35] G. Weiss, Mining with rare cases, in: O. Maimon, L. Rokach (Eds.), Data Mining and Knowledge Discovery Handbook, Springer, New York, USA, 2005, pp. 765–776.
[36] V. López, A. Fernández, J. Moreno-Torres, F. Herrera, Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics, Expert Syst. Appl. 39 (7) (2012) 6585–6608.
[37] N. Chawla, K. Bowyer, L. Hall, W. Kegelmeyer, SMOTE: synthetic minority oversampling technique, J. Artif. Intell. Res. 16 (2002) 321–357.
[38] Y. Freund, R. Schapire, Experiments with a new boosting algorithm, in: Proceedings of the 1996 International Conference on Machine Learning (ICML), vol. 96, 1996, pp. 148–156.
[39] Y. Sun, M. Kamel, A. Wong, Y. Wang, Cost-sensitive boosting for classification of imbalanced data, Pattern Recognit. 40 (12) (2007) 3358–3378.
[40] N. Verbiest, C. Cornelis, R. Jensen, Fuzzy rough positive region based nearest neighbour classification, in: Proceedings of the 2012 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2012, pp. 1–7.
[41] C. Cornelis, N. Verbiest, R. Jensen, Ordered weighted average based fuzzy rough sets, in: J. Yu, S. Greco, P. Lingras, G. Wang, A. Skowron (Eds.), Rough Set and Knowledge Technology, Springer, Beijing, China, 2010, pp. 78–85.
[42] M. Zhang, Z. Zhou, Multi-instance clustering with applications to multi-instance prediction, Appl. Intell. 31 (1) (2009) 47–68.
[43] J. Keller, M. Gray, J. Givens, A fuzzy k-nearest neighbor algorithm, IEEE Trans. Syst. Man Cybern. 15 (4) (1985) 580–585.
[44] M. Lamata, E. Pérez, Obtaining OWA operators starting from a linear order and preference quantifiers, Int. J. Intell. Syst. 27 (3) (2012) 242–258.
[45] N. Verbiest, Fuzzy rough and evolutionary approaches to instance selection, Ph.D. Thesis, Ghent University, 2014.
[46] T. Fawcett, An introduction to ROC analysis, Pattern Recognit. Lett. 27 (8) (2006) 861–874.
[47] S. García, F. Herrera, Evolutionary undersampling for classification with imbalanced datasets: proposals and taxonomy, Evolut. Comput. 17 (3) (2009) 275–306.
[48] F. Wilcoxon, Individual comparisons by ranking methods, Biom. Bull. 1 (6) (1945) 80–83.
[49] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.
[50] S. García, A. Fernández, J. Luengo, F. Herrera, Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power, Inf. Sci. 180 (10) (2010) 2044–2064.
[51] H. Blockeel, D. Page, A. Srinivasan, Multi-instance tree learning, in: Proceedings of the 22nd International Conference on Machine Learning (ICML), 2005, pp. 57–64.

Sarah Vluymans holds an M.Sc. degree (2014) in Mathematical Computer Science from Ghent University. Currently, she is a Ph.D student at Ghent University. Her research is focused on the integration of fuzzy rough set theory in machine learning techniques.

Dánel Sánchez Tarragó obtained his Ph.D. degree from the University of Granada in 2014. He is currently working as a postdoctoral researcher at the University of Las Villas, Cuba. His primary research focus is on multi-instance learning.

Yvan Saeys obtained his M.Sc. (2000) and Ph.D. (2004) in Computer Science at Ghent University. He is currently leading the DAMBI research group (Data Mining and Modeling for Biomedicine), where his research focuses on the development and application of data mining and machine learning techniques for biological and medical applications.

Chris Cornelis holds an M.Sc. (2000) and Ph.D. degree (2004) in Computer Science from Ghent University. Currently, he is a postdoctoral fellow at the University of Granada (Ramón y Cajal programme) and a guest professor at Ghent University. His research interests include fuzzy rough sets, instance selection and classification.

Francisco Herrera received his M.Sc. (1988) and Ph.D. (1991) in Mathematics from the University of Granada and is a professor at the Department of Computer Science and Artificial Intelligence. He is an Editor in Chief of “Information Fusion” and “Progress in Artificial Intelligence” and on the editorial board of several more.