

Reducing Dimensionality in Multiple Instance Learning with a Filter Method

Amelia Zafra¹, Mykola Pechenizkiy², and Sebastián Ventura¹

¹ Department of Computer Science and Numerical Analysis. University of Cordoba

² Department of Computer Science. Eindhoven University of Technology

Abstract. In this article, we describe a feature selection algorithm which can automatically find relevant features for multiple instance learning. Multiple instance learning is considered an extension of traditional supervised learning where each example is made up of several instances and there is no specific information about particular instance labels. In this scenario, traditional supervised learning can not be applied directly and it is necessary to design new techniques. Our approach is based on principles of the well-known Relief-F algorithm which is extended to select features in this new learning paradigm by modifying the distance, the difference function and computation of the weight of the features. Four different variants of this algorithm are proposed to evaluate their performance in this new learning framework. Experiment results using a representative number of different algorithms show that predictive accuracy improves significantly when a multiple instance learning classifier is learnt on the reduced data set.

1 Introduction

Theoretically, having more features to carry out machine learning tasks should give us more discriminating power. However, the real-world provides us with many reasons why this is not generally the case. Thus, if we reduce the set of features considered by the algorithm, we can considerably decrease the running time of the induction algorithms increasing the accuracy of the resulting model. In light of this, a considerable amount of research has addressed the issue of feature subset selection in machine learning. Most studies are focused on traditional supervised learning and very few studies have dealt with a multiple instance learning framework. Multiple Instance Learning (MIL) introduced by Dietterich et al. [1] consists of representing each example in a data set as a collection or a *bag* composed of single or multiple instances. In machine learning, MIL extends traditional supervised learning for problems maintaining incomplete knowledge about labels in training examples. In supervised learning, every training instance is assigned a label that is discrete or has a real-value. In comparison, in MIL, the labels are assigned only to bags of instances. In the binary case, a bag is labeled positive if at least one instance in that bag is positive, and the bag is labeled negative if all the instances in it are negative, but there are no labels for individual instances. The goal of MIL is to classify unseen bags by using the labeled bags as the training data. From the formulation of MIL, it is easy to see

that a positive bag may contain some negative instances in addition to one or more positive instances. Hence, the true labels for the instances in a positive bag may or may not be the same as the corresponding bag label and, consequently, the instance labels are inherently ambiguous. There have been very few feature selection studies done about MIL [2,3,4,5,6] and most of them deal with wrapper approaches. These approaches directly incorporate the bias of MIL algorithms and they proposed to achieve a better method for MIL that improves on previous algorithms which do not use dimensional reduction. However, there is no proposal for a filter method which can be used as a preprocessing step for any algorithm designed nor general empirical studies with a representative number of MIL algorithms to work with the same reduced data set to show the relevance of feature selection in the MIL framework.

In this paper, we address the problem of selecting a subset of important features for MIL from a filter perspective. In the filter model, feature selection is performed as a preprocessing step to induction y therefore it can be applied in any previous MIL algorithm. We propose an effective feature selection approach to MIL called ReliefF-MI which extends the ReliefF algorithm [7] to multiple instance learning. The proposed method assigns a real-valued weight to each feature to indicate its relevance to the problem. First, the features are ranked according to their importance and then a subset of important features is selected. One of the relevant characteristics of this method is that it can be applied to continuous and discrete problems in multiple instance classification, is aware of contextual information and can correctly estimate the quality of attributes in problems with strong dependencies between attributes. To check the effectiveness of the proposed model, different versions of ReliefF-MI are considered, modifying the similarity function to calculate the distance between patterns and the weight of the different features. Seventeen algorithms including different paradigms of machine learning, such as, decision rules, support vector machines, naive Bayes, decision trees, logistic regression, diverse density and based on distances are tested to establish the effectiveness of filter methods in MIL. Experimental results show, on one hand, that the new similarity function designed gets that ReliefF-MI method achieves the best results and, on the other hand, it is proven that algorithms always out-perform the results obtained when they apply the dimensionality reduction provided by ReliefF-MI as a preprocessing step. Thereby, it is demonstrated considering an important number of different methods that knowing the more relevant features in classification with MIL, this task is become more efficient.

The paper is structured as follows. Section 2 describes the proposals developed to reduce features in MIL. Section 3 evaluates and compares the methods proposed with respect to considering the whole feature set. Finally, section 4 ends with conclusions and raises several issues for future work.

2 A Filter Approach - ReliefF-MI

The procedure proposed is based on the principles of the ReliefF algorithm [7]. The original ReliefF algorithm estimates the quality of attributes by how well

their values distinguish between instances that are near each other. In MIL, the distance between two patterns has to be calculated taking into account that each pattern contains one or more instances. Therefore, a new concept has to be introduced concerning the similarity function and the calculation of the differences between attribute A in patterns R and H , $diff(A, R, H)$. The literature proposes different distance-based approaches to solve MI problems [8,9,10]. The most extensively used metric is Hausdorff distance [11], which measures the distance between two sets. Three adaptations of this measurement have been proposed in MIL literature: maximal Hausdorff, minimal Hausdorff and average Hausdorff. Here, a new metric is also designed based on previous ones that has been called the *Adapted Hausdorff*. With each different metric a different version of ReliefF-MI is proposed.

In continuation the metric and differential function in different versions is shown. In all cases R_i is considered the bag selected in the current iteration which contains three instances (R_i^1, R_i^2, R_i^3), H_j is the j^{th} bag of the k nearest hit selected in the current iteration which contains four instances, ($H_j^1, H_j^2, H_j^3, H_j^4$) and M_j is the j^{th} bag of the k nearest misses selected in the current iteration which contains six instances, ($M_j^1, M_j^2, M_j^3, M_j^4, M_j^5, M_j^6$).

ReliefF-MI with maximal Hausdorff distance. This extension of ReliefF for MIL uses the *Maximal Hausdorff Distance* [11]. This distance is classical Hausdorff distance and can be specified as: $H_{max}(R_i, H_j) = \max\{h_{max}(R_i, H_j), h_{max}(H_j, R_i)\}$ where $h_{max}(R_i, H_j) = \max_{r \in R_i} \min_{h \in H_j} \|r - h\|$

To calculate the difference between the attributes of two patterns, the instance selected are the maximal distance of the minimal distance between the different instances of one bag and another: $diff_{bag-max}(A, R_i, H_j) = diff_{instance}(A, R_i^3, H_j^4)$ being R_i^3 and H_j^4 instances which satisfy this condition. Similarly, the $diff_{bag-max}(A, R_i, M_j)$ is computed but with the instances of R_i and M_j .

ReliefF-MI with minimal Hausdorff distance. This extension of ReliefF for MIL uses the *Minimal Hausdorff Distance* [8]. This distance instead of choosing the maximum distance, the distance is ranked first and the lowest distance value is selected. Formally, it modifies the Hausdorff Distance definition as follows: $H_{min}(A, B) = \min_{a \in A} \min_{b \in B} \|a - b\|$

To calculate the difference between the attributes of two patterns, the instance of each bag selected to calculate the distances is the minimal distance between all instances of each bag: $diff_{bag-min}(A, R_i, H_j) = diff_{instance}(A, R_i^1, H_j^3)$ being R_i^1 and H_j^3 instances that satisfy this condition. Similarly, $diff_{bag-min}(A, R_i, M_j)$ would be computed but with the instances of R_i and M_j .

ReliefF-MI with average Hausdorff distance. This extension of ReliefF for MIL uses the *Average Hausdorff distance* [10] proposed by Zhang and Zhou to measure the distance between two bags. It is defined as follows: $H_{avg}(R_i, H_j) = (\sum_{r \in R_i} \min_{h \in H_j} \|r - h\| + \sum_{h \in H_j} \min_{r \in R_i} \|h - r\|) / (|R_i| + |H_j|)$

where $|\cdot|$ measures the cardinality of a set. $H_{avg}(A, B)$ averages the distances between each instance in one bag and its nearest instance in the other bag. Conceptually speaking, average Hausdorff distance takes more geometric relationships between two bags of instances into consideration than the maximal and minimal Hausdorff ones.

To calculate the difference between the attributes of two patterns, there are several instances involved to update the weights of the features. If we suppose

- $d(R_i^1, H_j^2)$, $d(R_i^2, H_j^1)$ and $d(R_i^3, H_j^4)$ are the minimal distance between each instance $r \in R_i$ with respect to instances $h \in H_j$.
- $d(H_j^1, R_i^1)$, $d(H_j^2, R_i^1)$, $d(H_j^3, R_i^2)$ and $d(H_j^4, R_i^3)$, are the minimal distances between each instance $h \in H_j$ with respect to the instances $r \in R_i$.

The function *diff* would be specified as following,

$$diff_{bag-avg}(A, R_i, H_j) = \frac{1}{r+h} * [diff_{instance}(A, R_i^1, H_j^2) + diff_{instance}(A, R_i^2, h_j^1) + diff_{instance}(A, R_i^3, h_j^4) + diff_{instance}(A, H_j^1, R_i^1) + diff_{instance}(A, H_j^2, R_i^1) + diff_{instance}(A, H_j^3, R_i^2) + diff_{instance}(A, H_j^4, R_i^3)]$$

The same process is used to calculate $diff_{bag-avg}(A, R_i, M_j)$, but considering the pattern M_j .

ReliefF-MI with adapted Hausdorff distance. This extension of ReliefF for MIL uses the *Adapted Hausdorff distance*. Due to the particularities of this learning, we propose this new distance combining the previous ones to measure the distance between two bags. This metric represents a different calculation depending on the class of the pattern because the information about instances in each pattern depends on the class that it belongs to. Thus, the metric is different if we evaluate the distance between two positive or negative patterns or between one positive and one negative pattern.

- If both patterns are negative, we can be sure that there is no instance in the pattern that represents the concept that we want to learn. Therefore, an average distance will be used to measure the distance between these bags because all instances are guaranteed to be negative: $H_{adapted}(R_i, H_j) = H_{avg}(R_i, H_j)$.
- If both patterns are positive. The correct information is that at least one instance in each of them represents the concept that we want to learn, but there is no information about which particular instance or set represents the concept. Therefore, we use the minimal distance to measure their distance because the positive instance has more probability of being near: $H_{adapted}(R_i, H_j) = H_{min}(R_i, H_j)$
- Finally, if we evaluate the distance between patterns where one of them is a positive bag and the other is a negative one, the measurement considered will be the maximal Hausdorff distance because the instances in the different classes are probably outliers between the two patterns: $H_{adapted}(R_i, M_j) = H_{max}(R_i, M_j)$

In this case, the calculation of the *diff* function also depends on the pattern class. Therefore, the pattern label will determine how the function *diff* will be evaluated.

- If R_i is positive and H_j is positive,

$$diff_{bag-adapted}(A, R_i, H_j) = diff_{bag-min}(A, R_i, H_j)$$

- If R_i is negative and H_j is negative,

$$diff_{bag-adapted}(A, R_i, H_j) = diff_{bag-avg}(A, R_i, H_j)$$

- Finally, if R_i is positive and M_j is negative or viceverse,

$$diff_{bag-adapted}(A, R_i, M_j) = diff_{bag-max}(A, R_i, M_j)$$

The function *diff_{instance}* used in the previous calculations is the difference between two particular instances for a given attribute. The total distance is simply the sum of distances throughout all attributes (Manhattan distance, [12]). When dealing with nominal attributes, function *diff*(A, R_i, H_j) is defined as:

$$diff_{instance}(A, I_x, I_y) = \begin{cases} 0; & \text{value}(A, R_i) = \text{value}(A, H_j) \\ 1; & \text{otherwise} \end{cases}$$

and for numerical attributes as:

$$diff_{instance}(A, R_i, H_j) = \frac{|value(A, R_i) - value(A, H_j)|}{max(A) - min(A)}$$

3 Experimental Results

The experimental study is aimed to evaluate the performance of ReliefF-MI. This evaluation has been broken down into two parts: a comparative study between the results obtained by the different versions designed and a comparison between algorithm performance that does or does not use dimensionality reduction to show the relevance of this filter method in the MIL framework. In the experimentation, seventeen of the most popular proposals in MIL are considered with three applications of categorization of images based on content [13,14], whose names and characteristics are given in Table 1. All the experiments were executed using 10-fold cross validation, and a statistical test was adopted to analyze the experimental results.

Table 1. General Information about Data Sets

DATASET	BAGS			ATTRIBUTES	INSTANCES	AVERAGE BAG SIZE
	Positive	Negative	Total			
Elephant	100	100	200	230	1391	6.96
Tiger	100	100	200	230	1220	6.10
Fox	100	100	200	230	1320	6.60

3.1 Comparison of Different Metrics for ReliefF-MI

To determine which metric is more interesting, representative paradigms in MIL have been considered. The different paradigms consider seventeen algorithms including: methods based on Diverse Density: MIDD, MIEMDD and MDD; methods based on Logistic Regression: MILR; methods based on Support Vector Machines: SMO and MISMO; distance-based Approaches: CitationKNN and MIOptimalBall; methods based on Rules: such as PART, Bagging with PART and AddaBoost with PART using MIWrapper and MISimple approach (they are different adaptations for working with MIL); method based on decision trees: MIBoost and methods based on Naive Bayes. More information about the algorithms considered could be consulted at the WEKA workbench [15] where these techniques have been designed.

Table 2. Results with Reduced Feature Data Set

ALGORITHMS	Maximal			Minimal			Average			Adapted		
	Eleph	Tiger	Fox	Eleph	Tiger	Fox	Eleph	Tiger	Fox	Eleph	Tiger	Fox
citationKNN	0.750	0.830	0.615	0.745	0.850	0.630	0.745	0.840	0.610	0.745	0.815	0.615
MDD	0.725	0.810	0.620	0.710	0.800	0.600	0.710	0.790	0.605	0.705	0.805	0.660
RepTree ¹	0.825	0.870	0.655	0.840	0.845	0.665	0.840	0.865	0.700	0.840	0.855	0.710
DecisionStump ¹	0.825	0.800	0.655	0.820	0.785	0.695	0.820	0.800	0.660	0.830	0.805	0.700
MIDD	0.755	0.780	0.600	0.750	0.780	0.645	0.750	0.780	0.595	0.755	0.770	0.695
MIEMDD	0.725	0.775	0.530	0.685	0.720	0.605	0.685	0.745	0.530	0.715	0.770	0.615
MILR	0.815	0.855	0.600	0.840	0.825	0.630	0.840	0.840	0.615	0.835	0.875	0.635
MIOptimalBall	0.795	0.740	0.575	0.765	0.715	0.495	0.765	0.735	0.525	0.775	0.740	0.535
RBF Kernel ²	0.765	0.835	0.615	0.800	0.865	0.655	0.800	0.830	0.655	0.785	0.855	0.650
Polynomial Kernel ²	0.765	0.825	0.620	0.780	0.825	0.685	0.780	0.830	0.665	0.770	0.820	0.655
AdaBoost&PART ³	0.830	0.840	0.615	0.830	0.825	0.745	0.830	0.820	0.620	0.840	0.860	0.665
Bagging&PART ³	0.830	0.850	0.585	0.810	0.865	0.595	0.810	0.860	0.610	0.830	0.865	0.605
PART ³	0.830	0.815	0.580	0.815	0.830	0.615	0.815	0.810	0.570	0.835	0.840	0.620
SMO ³	0.705	0.815	0.660	0.715	0.835	0.675	0.715	0.830	0.655	0.705	0.820	0.690
Naive Bayes ³	0.655	0.820	0.590	0.675	0.815	0.650	0.675	0.825	0.585	0.660	0.820	0.680
AdaBoost&PART ⁴	0.800	0.855	0.570	0.840	0.830	0.600	0.840	0.840	0.560	0.830	0.845	0.650
PART ⁴	0.770	0.795	0.620	0.765	0.730	0.670	0.765	0.740	0.660	0.775	0.780	0.665
RANKING	2.676			2.520			2.794			2.010		

¹ MIBoost ³ MIWrapper
² MISMO ⁴ MISimple

The average results of accuracy for tiger, fox and elephant data sets obtained by the algorithms using the different proposals for ReliefF-MI are reported in Table 2. To check which metric achieves the most relevant features for different algorithms a statistical test is carried out (Friedman test). This test is a non parametric test that compares the average ranks of the proposals considered, where the metric that achieves the highest accuracy for one algorithm is given a rank of 1, the metric with the next highest accuracy value for this algorithm has the rank of 2, and so on with all algorithms and data sets used. These ranks let us know which metric obtains the best results considering all algorithms and data sets. In this way, the metric with the value closest to 1 indicates that the algorithms considered in this study generally obtain better results using the

reduced feature set it provided. The ranks obtained by each metric can be seen in Table 2. The Friedman test determines at a confidence level of 95% that there are significant differences in the results when use different metrics because its value is 10.965 and the $\chi^2(n = 3)$ is 6.251, therefore the null hypothesis is rejected and it is obvious that there are differences between them. The Bonferroni test is carried out to determine which metric selects the most relevant features. Results of this test show that the methods with a threshold over 2.638 (confidence at 95%) are considered worse proposals than the control algorithm. In this case, this algorithm would be ReliefF-MI with metric designed because it gets the lowest ranking value and therefore it is the best option. Statically, the worst proposals would be average Hausdorff distance and a maximum of Hausdorff distance which has a higher ranking than the threshold set by this test.

3.2 Effectiveness of ReliefF-MI

Results obtained by algorithms when they use the reduced data set provided by ReliefF-MI using adapted Hausdorff distance are compared to results using the full data set to show the effectiveness of ReliefF-MI. Table 3 show the results of the seventeen algorithms for different data sets in two cases.

A statistical study is carried out to check whether the use of ReliefF-MI improves the results of algorithms with respect to results when no dimensionality reduction is done. The Wilcoxon rank sum test is used to find whether there are or are not differences between accuracy values obtained by different algorithms using the feature set provided by ReliefF-MI. This test is a non-parametric recommended in Demsar’s study [16]. The null hypothesis of this test maintains that there are no significant differences between accuracy values obtained by algorithms when they use different feature sets, while the alternative hypothesis assures that there are. Table 4 shows the mean ranks and the sum of ranks for

Table 3. Results with Full Feature Data Set

ALGORITHMS	Reduced Set			Full Set		
	Eleph	Tiger	Fox	Eleph	Tiger	Fox
citationKNN	0.745	0.815	0.615	0.500	0.500	0.500
MDD	0.705	0.805	0.660	0.800	0.755	0.700
MIBoost (RepTree)	0.840	0.855	0.710	0.815	0.825	0.670
MIBoost (DecisionStump)	0.830	0.805	0.700	0.815	0.780	0.650
MIDD	0.755	0.770	0.695	0.825	0.740	0.655
MIEMDD	0.715	0.770	0.615	0.730	0.745	0.600
MILR	0.835	0.875	0.635	0.780	0.840	0.510
MIOptimalBall	0.775	0.740	0.535	0.730	0.625	0.530
MISMO (RBF Kernel)	0.785	0.855	0.650	0.800	0.795	0.590
MISMO (Polynomial Kernel)	0.770	0.820	0.655	0.790	0.785	0.580
MIWrapper (AdaBoost&PART)	0.840	0.860	0.665	0.840	0.790	0.685
MIWrapper (Bagging&PART)	0.830	0.865	0.605	0.845	0.810	0.600
MIWrapper (PART)	0.835	0.840	0.620	0.790	0.780	0.550
MIWrapper (SMO)	0.705	0.820	0.690	0.715	0.800	0.635
MIWrapper (Naive Bayes)	0.660	0.820	0.680	0.680	0.760	0.590
MISimple (AdaBoost&PART)	0.830	0.845	0.650	0.840	0.795	0.625
MISimple (PART)	0.775	0.780	0.665	0.765	0.765	0.635

Table 4. Sum of Ranks and Mean Rank of the two proposals

	Mean Rank Sum of Ranks	
ReliefF-MI Method ¹	57.03	2908.50
Not Reducing features	45.97	2344.50

¹ Adapted Hausdorff Distance.

each of the two options. The scores are ranked from lowest to highest. Therefore, we can see that algorithms not using feature selection have a lower mean rank than algorithms using the ReliefF-MI method. This information can be used to ascertain a priori that ReliefF-MI is a better proposal.

The results of the Wilcoxon statistical test are 2344 and the corresponding z-score is -1.888. According to these values, the results are significant (p -value = $0.059 < 0.1$) to a 90% level of confidence when we reject the null hypothesis and determine that there are significant differences between the results obtained by algorithms when they use the feature selection method. Consequently, ReliefF-MI has significantly higher accuracy values than the option that does not use feature reduction. This conclusion is reached by noting that for ReliefF-MI scores, the mean rank is higher in the algorithms using selection feature as pre-processing step (for example, at a value of 57.03) than the other option (at a value of 45.97).

In general, we can conclude that the use of this method benefits the results achieved by the algorithms. That is, the usefulness of this method of feature selection in MIL is demonstrated because it manages to optimize the results obtained by different algorithms. Thus, the results with a lower number of features minimize the classification error, and the efficiency of this method is shown in the case of MIL. The study considers seventeen different algorithms and three different data sets, so it is a representative enough study to reach this conclusion.

4 Conclusions

The task of finding relevant features for multiple instance data is largely untouched. In this learning framework, the process is more complex because the class labels of particular instances in the patterns are not available. Thus, the classic method of feature selection for traditional supervised learning with single data does not work well in this scenario. For multiple instance data, where information shows uncertainty, only it has been proposed methods related with optimizing a particular algorithm (most of them are wrapper approaches). This paper addresses the problem of feature selection to reduce the dimensionality of data in MIL in a general way for any algorithm using filter method. Thus, we describe a new efficient algorithm based on ReliefF principles [7] that can be applied to continuous and discrete problems, is faster than wrapper methods and can be applied to any MIL algorithm designed previously because the method is applied as a preprocessing step. Experimental results shows the effectiveness of

our approach using three different applications and seventeen algorithms with the reduced data. First, the different metrics are compared to evaluate their effect on the algorithm developed. Results show that the new metric proposed is the metric that statistically achieves the best results. This metric is designed to be adapted to the variable information of each pattern according to the class of that pattern. So, the Wilcoxon test shows the benefits of applying data reduction in MIL and obtains better results in all algorithms in general when they only work with the most relevant features. Thus, the relevance of using feature selection in this scenario is established for improving the performance of algorithms with high-dimensional data.

More work that can be done in this area includes the designing of other metrics to measure the distance between bags in order to optimize the performance of this method, as well as the designing of other feature selection methods based on filtering in MIL scenario to study which methods work best in this learning context.

Acknowledgements

The authors gratefully acknowledge the financial support provided by the Spanish department of Research under TIN2008- 06681-C06-03, P08-TIC-3720 Projects and FEDER funds.

References

1. Dietterich, T.G., Lathrop, R.H., Lozano-Perez, T.: Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence* 89(1-2), 31–71 (1997)
2. Zhang, M.L., Zhou, Z.H.: Improve multi-instance neural networks through feature selection. *Neural Processing Letter* 19(1), 1–10 (2004)
3. Chen, Y., Bi, J., Wang, J.: Miles: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28(12), 1931–1947 (2006)
4. Yuan, X., Hua, X.S., Wang, M., Qi, G.J., Wu, X.Q.: A novel multiple instance learning approach for image retrieval based on adaboost feature selection. In: *ICME 2007: Proceedings of the IEEE International Conference on Multimedia and Expo.*, Beijing, China, pp. 1491–1494. IEEE, Los Alamitos (2007)
5. Raykar, V.C., Krishnapuram, B., Bi, J., Dundar, M., Rao, R.B.: Bayesian multiple instance learning: automatic feature selection and inductive transfer. In: *ICML 2008: Proceedings of the 25th International Conference on Machine Learning*, pp. 808–815. ACM, New York (2008)
6. Herman, G., Ye, G., Xu, J., Zhang, B.: Region-based image categorization with reduced feature set. In: *Proceedings of the 10th IEEE Workshop on Multimedia Signal Processing*, Cairns, Qld, pp. 586–591 (2008)
7. Kononenko, I.: Estimating attributes: analysis and extension of relief. In: Bergadano, F., De Raedt, L. (eds.) *ECML 1994*. LNCS, vol. 784, pp. 171–182. Springer, Heidelberg (1994)

8. Chevaleyre, Y.Z., Zucker, J.D.: Solving multiple-instance and multiple-part learning problems with decision trees and decision rules. Application to the mutagenesis problem. In: Stroulia, E., Matwin, S. (eds.) Canadian AI 2001. LNCS (LNAI), vol. 2056, pp. 204–214. Springer, Heidelberg (2001)
9. Zhang, D., Wang, F., Si, L., Li, T.: M3IC: Maximum margin multiple instance clustering, pp. 1339–1344 (2009)
10. Zhang, M.L., Zhou, Z.H.: Multi-instance clustering with applications to multi-instance prediction. *Applied Intelligence* 31, 47–68 (2009)
11. Edgar, G.: *Measure, topology, and fractal geometry*, 3rd edn. Springer, Heidelberg (1995)
12. Cohen, H.: Image restoration via n-nearest neighbour classification. In: *ICIP 1996: Proceedings of the International Conference on Image Processing*, pp. 1005–1007 (1996)
13. Yang, C., Lozano-Perez, T.: Image database retrieval with multiple-instance learning techniques. In: *ICDE 2000: Proceedings of the 16th International Conference on Data Engineering*, Washington, DC, USA, pp. 233–243. IEEE Computer Society, Los Alamitos (2000)
14. Pao, H.T., Chuang, S.C., Xu, Y.Y., Fu, H.: An EM based multiple instance learning method for image classification. *Expert Systems with Applications* 35(3), 1468–1472 (2008)
15. Witten, I.H., Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd edn. Morgan Kaufmann, San Francisco (2005)
16. Demsar, J.: Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* 17, 1–30 (2006)