A First Approach to Deal with Imbalance in Multi-label Datasets

Francisco Charte[1], Antonio Rivera[2],
María José del Jesus[2], and Francisco Herrera[1]

[1] Dep. of Computer Science and Artificial Intelligence,
University of Granada, Granada, Spain
[2] Dep. of Computer Science, University of Jaén, Jaén, Spain
{fcharte,herrera}@ugr.es, {arivera,mjjesus}@ujaen.es
http://simidat.ujaen.es, http://sci2s.ugr.es

Abstract. The process of learning from imbalanced datasets has been deeply studied for binary and multi-class classification. This problem also affects to multi-label datasets. Actually, the imbalance level in multi-label datasets uses to be much larger than in binary or multi-class datasets. Notwithstanding, the proposals on how to measure and deal with imbalanced datasets in multi-label classification are scarce.

In this paper, we introduce two measures aimed to obtain information about the imbalance level in multi-label datasets. Furthermore, two preprocessing methods designed to reduce the imbalance level in multi-label datasets are proposed, and their effectiveness is validated experimentally. Finally, an analysis for determining when these methods have to be applied depending on the dataset characteristics is provided.

Keywords: Multi-label Classification, Imbalanced Datasets, Preprocessing, Measures.

1 Introduction

Classification is one of the most important tasks in the field of supervised learning. Multi-label classification (MLC) [1] is a generalization of binary and multi-class classification, as it does not impose an a priori limit to the number of elements that the set of outputs can hold. This type of classification is receiving significant attention lately, and it is being applied in fields such as text categorization [2] and music labeling [3], among others.

The data used for learning a classifier is often imbalanced, as the class labels assigned to each instance are not equally represented. This is a profoundly examined problem [4], but almost limited to binary datasets and to a lesser extent to multi-class datasets. That most multi-label datasets (MLDs) suffer from a large level of imbalance is a commonly accepted fact in the specialized literature [5], but there is a lack of measures to obtain information about it. In addition, and to the best of our knowledge, the proposals made until now to deal with imbalance in MLC have been focused in algorithmic adaptations of MLC algorithms [5–7], but none of them provides a general way of handling this problem.

J.-S. Pan et al. (Eds.): HAIS 2013, LNAI 8073, pp. 150–160, 2013.

In this paper two measures directed to determine the level of imbalance in MLDs are introduced, and two preprocessing methods aimed at reducing the imbalance in MLDs are proposed. The usefulness of the measures and effectiveness of the methods are proven experimentally, using different MLDs and MLC algorithms. The analysis of classification results provides a convenient guide in order to decide when an MLD suffers of imbalance and, therefore, could benefit from the preprocessing.

The rest of this paper is structured as follows: Section 2 briefly describes the MLC and the learning from imbalanced data problems. Section 3 introduces the imbalance problem in MLC, and presents the main proposals of the study which are the measures and preprocessing methods cited above. In Section 4 the experimental framework is described, and the results obtained are analyzed. Finally, the conclusions are presented in Section 5.

2 Preliminaries

2.1 Multi-label Classification

In many application domains [2,3,8] each data sample is associated with a set of labels, instead of only one class label as in binary and multi-class classification. Therefore, Y being the total set of labels in an MLD D, a multi-label classifier must produce as output a set $Z_i \subseteq Y$ with the predicted labels for the *i-th* sample. As each distinct label in Y could appear in Z_i, the total number of potential different combinations would be $2^{|Y|}$. Each one of these combinations is called a *labelset*. The same labelset can appear in several instances of D.

There are two main approaches [1] to accomplish an MLC task: data transformation and algorithm adaptation. The former aims to produce from an MLD a dataset or group of datasets which can be processed with traditional classifiers, while the latter has the objective of adapting existent classification algorithms in order to work with MLDs. Among the transformation methods the most popular are those based in the binarization of the MLD, such as Binary Relevance (BR) [9] and Ranking by Pairwise Comparison [10], and the Label Powerset (LP) [11] transformation, which produces a multi-class dataset from an MLD. In the algorithm adaptation approach there are proposals of multi-label C4.5 trees [12], algorithms based in nearest neighbors such as ML-kNN [13], multi-label neural networks [2,14], and multi-label SVMs [15], among others.

There are some specific measures to characterize MLDs, such as label cardinality *Card* and label density *Dens*. The former is the average number of active labels per sample in an MLD, while the latter is calculated as $Card/|Y|$ in order to obtain a dimensionless measure.

2.2 Classification with Imbalanced Data

The learning from imbalanced data problem is founded on the different distributions of class labels in the data [4], and it has been thoroughly studied in the context of binary classification. In this context, the measurement of the imbalance level in a dataset is obtained as the ratio of the number of samples of the majority class and the number associated to the minority class, being known as *imbalance ratio* (IR) [16]. The higher the IR, the larger is the imbalance level. The difficulty in the learning process with this kind of data is due to the design of most classifiers, as their main goal is to reduce some global error rate [16]. This approach tends to penalize the classification of the minority classes.

In binary and multi-class classification the imbalance problem has been mainly faced using two different approaches: data preprocessing [17] and cost sensitive classification [18]. The former is based on the rebalancing of class distributions, either deleting instances of the most frequent class (undersampling) or adding new instances of the least frequent one (oversampling). Random undersampling (RUS) [19], random oversampling (ROS) and SMOTE [20] are among the most used preprocessing methods to equilibrate imbalanced datasets. The advantage of the preprocessing approach is that it can be applied as a general method to solve the imbalance problem, independently of the classification algorithms applied once the datasets have been preprocessed.

3 The Imbalance Problem in MLC

Most MLDs [21] have hundreds of labels, being each instance associated with a subset of them. Intuitively, it is easy to see that the more different labels exist, the more possibilities there are that some of them have a very low presence. In Figure 1, which represents the sample distribution per label of CAL500 dataset, this fact can be verified. However, as will be seen in Section 4, it is not straightforward

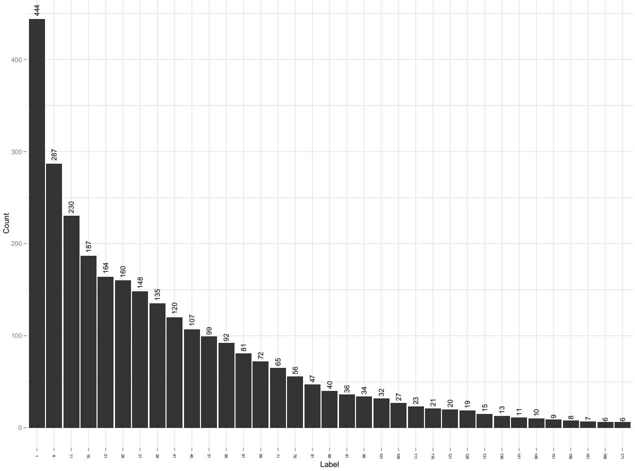

Fig. 1. Number of instances per label in CAL500 dataset

to infer the imbalance level from measures such as *Card* and *Dens*, which are the most widely used in the literature in order to characterize MLDs.

Many of the proposals made in the literature [5–7] for dealing with imbalanced datasets in MLC claim the imbalanced nature of MLDs, but none of them offer a procedure to measure it. Furthermore, most of these proposals aim to deal with the imbalance problem by means of algorithmic adaptations of MLC classifiers or the use of ensembles of classifiers. Therefore, there is a need for specific measures which can be used to obtain information about the imbalance level in MLDs, as well as some way able to face this problem while maintaining the use of the usual MLC algorithms.

3.1 Proposals on How to Measure the Imbalance Level in MLC

In traditional classification the imbalance level is measured taking into account only two classes: the majority class and the minority class. However, many MLDs have hundreds of labels, and several of them may have a very low presence. For that reason, it is important to define the level of imbalance in MLC considering not only two labels, but all of them. In this scenario, we propose the use of the following measures:

- *IRperLabel*: It is calculated for each label as the ratio between the majority and the considered labels, as shown in Equation 1. This value will be 1 for the most frequent label and a greater value for the rest. The higher *IRperLabel* is, the larger will be the imbalance level for the considered label.

$$\mathit{IRperLabel(y)} = \frac{\underset{y'=Y_1}{\overset{Y_{|Y|}}{\operatorname{argmax}}}(\sum_{i=1}^{|D|} h(y', Y_i))}{\sum_{i=1}^{|D|} h(y, Y_i)}, \quad h(y, Y_i) = \begin{cases} 1 & y \in Y_i \\ 0 & y \notin Y_i \end{cases}. \tag{1}$$

- *MeanIR*: This measure will offer a value which represents the average level of imbalance in an MLD, obtained as shown in Equation 2.

$$\mathit{MeanIR} = \frac{1}{|Y|} \sum_{y=Y_1}^{Y_{|Y|}} (\mathit{IRperLabel(y)}). \tag{2}$$

- *CVIR*: This is the coefficient of variation of *IRperLabel*, and is calculated as shown in Equation 3. It will indicate if all labels suffer from a similar level of imbalance or, on the contrary, there are big differences in them. The higher is the *CVIR* the larger will be this difference.

$$\mathit{CVIR} = \frac{\mathit{IRperLabel\sigma}}{\mathit{MeanIR}}, \quad \mathit{IRperLabel\sigma} = \sqrt{\sum_{y=Y_1}^{Y_{|Y|}} \frac{\mathit{(IRperLabel(y)\ \text{-}\ MeanIR)}^2}{|Y| - 1}}. \tag{3}$$

Table 1 shows the *MeanIR* and *CVIR* for the datasets used in the experimentation conducted for the present study. As we will see in the discussion on Section 4, these values would be enough to get a first glimpse to know the imbalance level in MLDs.

3.2 LP-RUS and LP-ROS: Random Undersampling and Oversampling for MLC

The existent undersampling and oversampling methods cannot be directly used in MLC, as they are designed to work with one output class label only. Furthermore, these methods assume that there are only one minority label and one majority label. Thus, an approach to preprocess MLDs, which have a set of labels as output and several of them could be considered minority/majority labels, is needed. In this paper we propose two methods aimed to undersample and oversample MLDs, called LP-RUS and LP-ROS. Both are based on the LP transformation method [11], which has been used in order to transform MLDs, in classification algorithms such as RAkEL [22] and HOMER [23], and also to complete other kinds of tasks, such as the stratified partitioning of MLDs [24]. Therefore, LP-RUS and LP-ROS will interpret each labelset as class identifier while preprocessing an MLD.

LP-RUS is a multi-label undersampling method that deletes random samples of majority labelsets, until the MLD D is reduced to a 75% of its original size. This method works as follows:

```
procedure LP-RUS($D$)
    $samplesToDelete \leftarrow |D| * 0.25$                ▷ 25% size reduction
    for $i = 1 \rightarrow |labelsets|$ do    ▷ Group samples according to their labelsets
        $labelSetBag_i \leftarrow samplesWithLabelset(i)$
    end for
    ▷ Calculate the average number of samples per labelset
    $meanSize \leftarrow 1/|labelsets| * \sum_{i=1}^{|labelsets|} |labelSetBag_i|$
    ▷ Obtain majority labels bags
    for each $labelSetBag_i$ in $labelSetBag$ do
        if $|labelSetBag_i| > meanSize$ then
            $majBag_i \leftarrow labelSetBag_i$
        end if
    end for
    $meanReduction \leftarrow samplesToDelete/|majBag|$
    $majBag \leftarrow SortFromSmallestToLargest(majBag)$
    ▷ Calculate # of instances to delete and remove them
    for each $majBag_i$ in $majBag$ do
        $reductionBag_i \leftarrow min(|majBag_i| - meanSize, meanReduction)$
        $remainder \leftarrow meanReduction - reductionBag_i$
        $distributeAmongBags_{j>i}(remainder)$
        for $n = 1 \rightarrow reductionBag_i$ do
```

```
            $x \leftarrow random(1, |majBag_i|)$
            $deleteSample(majBag_i, x)$
        end for
    end for
end procedure
```

The procedure described above aims to achieve a labelset representation in the MLD as close as possible to an uniform distribution. However, since a limit on the minimum dataset size has been established, a certain degree of imbalance among the labelsets could remain in the MLD. In any case, the imbalance level always will be lower than in the original dataset.

LP-ROS is a multi-label oversampling method that works cloning random samples of minority labelsets, until the size of the MLD is a 25% larger than the original. The procedure followed is analogous to the described above for LP-RUS. In this case, a collection of minority groups *minBag*$_i$ with ($|$*labelsetBag*$_i| <$ *meanSize*) is obtained, a *meanIncrement* = *#samplesGenerate*/*#minBag* is calculated, and processing the minority groups from the largest to the smallest an individual increment for each *minBag*$_i$ is determined. If a *minBag*$_i$ reaches *meanSize* samples before *incrementBag*$_i$ instances have been added, the excess is distributed among the others *minBag*. Therefore, the labelsets with a lower representation will be benefited from a bigger number of clones, aiming to adjust the labelset representation to an uniform distribution as in the case of LP-RUS.

4 Experimentation and Analysis

4.1 Experimental Framework

Four MLDs from the MULAN repository [21] were selected in order to test the proposed preprocessing methods. These are shown in Table 1, along with some measures which characterize them: number of attributes, samples, and labels, the average number of labels per sample, and the previously proposed measures related to the imbalance level. As can be seen, there are datasets with a variety of values in *Card* and *Dens*, as well as some big differences in the number of labels, attributes, samples, and the imbalance measures. The goal is to analyze how the proposed preprocessing methods work with MLDs which are not similar, but quite different.

Table 1. Characterization measures of datasets used in experimentation

Dataset	#Attributes	#Samples	#Labels	Card	Dens	MeanIR	CVIR
CAL500	68	502	174	26.0438	0.1497	20.5778	1.0871
Corel5k	499	5000	374	3.5220	0.0094	189.5676	1.5266
genbase	1186	662	27	1.2523	0.0464	37.3146	1.4494
scene	294	2407	6	1.0740	0.1790	1.2538	0.1222

The high *MeanIR* and *CVIR* values for Corel5k and genbase suggest that these MLDs are the most imbalanced, and therefore they could be the more benefited from the preprocesing. The CAL500 measurements also indicate a certain level of imbalance, but $CVIR$ is significantly lower than in the case of Corel5k and genbase, as is $MeanIR$. Finally, the values associated to scene denote its nature of well balanced MLD, being a dataset which does not need any preprocessing.

These datasets have been partitioned using a 5x2 folds cross validation scheme, and the training partitions have been preprocessed with LP-RUS and LP-ROS.

Regarding the MLC algorithms, the following methods were selected: BR-C4.5 [9], CLR [25], RAkEL [22], and IBLR-ML [26]. Each MLC algorithm was run over the base datasets, without any preprocessing, as well as using the datasets once they had been processed with LP-RUS and LP-ROS, respectively.

In the MLC field more than a dozen evaluation measures have been defined [1]. In this study to assess the influence of the preprocessing methods the following have been used: accuracy (Equation 4), precision (Equation 5) and recall (Equation 6). In these expressions Y_i is the set of real labels associated to the instance x_i, whereas $h(x_i)$ would be the set of labels predicted by the multi-label classifier.

$$Accuracy = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \cap h(x_i)|}{|Y_i \cup h(x_i)|} \tag{4}$$

$$Precision = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \cap h(x_i)|}{|h(x_i)|} \tag{5}$$

$$Recall = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \cap h(x_i)|}{|Y_i|} \tag{6}$$

Accuracy is a measure which assess the positive and negative predictive performance of the classifier, while precision is focused in the positive predictive performance only. Recall is a measure used often in conjunction with precision, and in this context will be useful to know the proportion of true active labels which has been predicted.

4.2 Results and Analysis

In Table 2 the accuracy for each configuration without preprocessing (noted as Base), with LP-RUS, and with LP-ROS are shown, and best values are highlighted in bold. It can be observed that LP-ROS always improves the results of Corel5k, and almost always in the case of genbase. These are the datasets with highest *MeanIR* and *CVIR* values. This implies that they are the most imbalanced in average, and that the differences in imbalance level among their samples is bigger than in the other MLDs, and that is something which LP-ROS is able to partially fix. The *MeanIR* of CAL500 is significantly lower, as is its *CVIR*. LP-ROS considerably improves the result of this MLD when processed with CLR, while losing narrowly with the others MLC algorithms. LP-RUS achieves some ties and slightly improves the result of CAL500/IBLR-ML. The scene dataset, characterized by a very low *MeanIR* and *CVIR* which denotes its nature of more balanced MLD, is the only one without any improvements.

Accuracy assesses positive and negative predictive performance. Table 3 shows the evaluation of results in terms of precision, a measure which only quantifies the positive predictive performance. This measure is generally used in conjunction

Table 2. Accuracy values on test sets

Dataset	Algorithm	Base	LP-RUS	LP-ROS
CAL-500	BR-J48	**0.2135**	**0.2135**	0.2060
CAL-500	RAkEL-BR	**0.2135**	**0.2135**	0.2060
CAL-500	CLR	0.1787	0.1787	**0.2116**
CAL-500	IBLR-ML	0.1922	**0.1926**	0.1900
Corel5k	BR-J48	0.0586	0.0480	**0.0607**
Corel5k	RAkEL-BR	0.0586	0.0480	**0.0607**
Corel5k	CLR	0.0360	0.0292	**0.0446**
Corel5k	IBLR-ML	0.0315	0.0235	**0.0368**
genbase	BR-J48	0.9842	0.9839	**0.9844**
genbase	RAkEL-BR	0.9842	0.9839	**0.9844**
genbase	CLR	**0.9837**	0.9812	0.9754
genbase	IBLR-ML	0.9790	0.9770	**0.9804**
scene	BR-J48	**0.5318**	0.5294	0.4648
scene	RAkEL-BR	**0.5318**	0.5294	0.4648
scene	CLR	**0.5242**	0.5194	0.4662
scene	IBLR-ML	**0.6786**	0.6683	0.6088

Table 3. Precision values on test sets

Dataset	Algorithm	Base	LP-RUS	LP-ROS
CAL-500	BR-J48	**0.4398**	**0.4398**	0.3448
CAL-500	RAkEL-BR	**0.4398**	**0.4398**	0.3448
CAL-500	CLR	**0.6364**	**0.6364**	0.5756
CAL-500	IBLR-ML	0.2859	**0.2864**	0.2776
Corel5k	BR-J48	**0.3643**	0.3638	0.1968
Corel5k	RAkEL-BR	**0.3643**	0.3638	0.1968
Corel5k	CLR	**0.4620**	0.4294	0.3624
Corel5k	IBLR-ML	0.0598	0.0451	**0.0805**
genbase	BR-J48	**0.9947**	**0.9947**	0.9939
genbase	RAkEL-BR	**0.9947**	**0.9947**	0.9939
genbase	CLR	**0.9946**	**0.9946**	0.9916
genbase	IBLR-ML	0.9899	0.9895	**0.9922**
scene	BR-J48	0.6752	**0.6811**	0.5989
scene	RAkEL-BR	0.6752	**0.6811**	0.5989
scene	CLR	0.6926	**0.6998**	0.6412
scene	IBLR-ML	**0.8230**	0.8164	0.7116

Table 4. Recall values on test sets

Dataset	Algorithm	Base	LP-RUS	LP-ROS
CAL-500	BR-J48	0.2964	0.2964	**0.3446**
CAL-500	RAkEL-BR	0.2964	0.2964	**0.3446**
CAL-500	CLR	0.2016	0.2016	**0.2584**
CAL-500	IBLR-ML	0.3722	0.3723	**0.3782**
Corel5k	BR-J48	0.0640	0.0516	**0.0789**
Corel5k	RAkEL-BR	0.0640	0.0516	**0.0789**
Corel5k	CLR	0.0378	0.0307	**0.0491**
Corel5k	IBLR-ML	0.0721	0.0690	**0.0856**
genbase	BR-J48	0.9896	0.9892	**0.9904**
genbase	RAkEL-BR	0.9896	0.9892	**0.9904**
genbase	CLR	**0.9885**	0.9858	0.9820
genbase	IBLR-ML	**0.9867**	0.9854	**0.9867**
scene	BR-J48	**0.6295**	0.6222	0.5826
scene	RAkEL-BR	**0.6295**	0.6222	0.5826
scene	CLR	**0.6574**	0.6454	0.6178
scene	IBLR-ML	0.6884	0.6809	**0.6956**

with recall (shown in Table 4), which is defined as the number of positive predictions versus real positives ratio.

Analyzing the effect of preprocessing methods with respect to precision and recall measures, the following can be observed: LP-ROS improves the recall in 12 of 16 configurations, but decreasing the precision. This means that LP-ROS produces a better coverage of labels which are present in the MLDs, but introducing false positives. On the contrary, LP-RUS improves the precision in some of the configurations, but the results with respect to recall are worse than the obtained by LP-ROS. This is due to the removing of false positives, but it also reduces the coverage of labels which should be present. When the preprocessing methods are applied to the scene dataset the results are not improved because of, as *MeanIR* and *CVIR* show, it could be considered as a balanced MLD.

From the analysis of these results, considering accuracy, precision and recall, it is possible to see that the LP-RUS preprocessing method, which reduces samples of the majority labelsets, obtains a slight improvement in precision but with significant costs. Intuitively, a labelset with a high representation in the MLD has to be conformed by frequent labels, but the results show that frequent labelsets can include individual labels with low presence in other samples of the MLD. Thus, this preprocessing method reduces the presence of the most frequent labels, but also deletes samples in which not so frequent labels appear.

On the other hand, LP-ROS is a preprocessing method able to produce a general improvement, taking into account both positive and negative performance prediction (determined by means of accuracy, precision and recall measures), when applied over imbalanced MLDs. LP-ROS is a first approach to face with the imbalance problem for MLDs, and can be considered as a simple and efficient approach to improve the results of different MLC algorithms for imbalanced MLDs, i.e. with high *MeanIR* and *CVIR* values.

5 Conclusions

The classification with imbalanced datasets problem has been deeply studied, but almost limited until now to binary and multi-class contexts. In this paper two measures aimed to evaluate the imbalance level in MLDs, together with two preprocessing algorithms, have been proposed, and the experimentation made to validate them has been described. LP-RUS is a random undersampling algorithm, whereas LP-ROS does random oversampling, in both cases taking as class value the labelset assigned to each data instance.

The proposed measures can be used to assess the imbalance level, and being able to decide if a certain MLD could be benefited from the proposed preprocessing methods. We advanced in subsection 4.1, with the information offered by this measures, that Corel5k and genbase would be the most benefited MLDs, and that scene should not be preprocessed as it do not suffered from imbalance. The results discussed in subsection 4.2 have endorsed our hypothesis.

Among the two preprocessing algorithms proposed, LP-ROS obtains the best results considering different quality measures. We conclude that the multi-label oversampling accomplished by LP-ROS is able to improve classification results when it is applied to MLDs with large level of imbalance, such as Corel5k and genbase, whatever MLC algorithm is used.

Acknowledgments. F. Charte is supported by the Spanish Ministry of Education under the F.P.U. National Program (Ref. AP2010-0068). This paper is partially supported by the projects TIN2012-33856 and TIN2011-28488 of the Spanish Ministry of Science and Technology, FEDER founds, and the projects TIC-3928 and P10-TIC-6858 of the Andalusian Research Plan, FEDER founds.

References

1. Tsoumakas, G., Katakis, I., Vlahavas, I.: Mining Multi-label Data. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, ch. 34, pp. 667–685. Springer US, Boston (2010)

2. Zhang, M.-L.: Multilabel Neural Networks with Applications to Functional Genomics and Text Categorization. IEEE Trans. Knowl. Data Eng. 18(10), 1338–1351 (2006)
3. Wieczorkowska, A., Synak, P., Raś, Z.: Multi-Label Classification of Emotions in Music. In: Intel. Inf. Proces. and Web Mining, ch. 30, vol. 35, pp. 307–315 (2006)
4. Chawla, N.V., Japkowicz, N., Kotcz, A.: Editorial: special issue on learning from imbalanced data sets. SIGKDD Explor. Newsl. 6(1), 1–6 (2004)
5. Tahir, M.A., Kittler, J., Bouridane, A.: Multilabel classification using heterogeneous ensemble of multi-label classifiers. Pattern Recognit. Letters 33(5), 513–523 (2012)
6. Tahir, M.A., Kittler, J., Yan, F.: Inverse random under sampling for class imbalance problem and its application to multi-label classification. Pattern Recognit. 45(10), 3738–3750 (2012)
7. He, J., Gu, H., Liu, W.: Imbalanced multi-modal multi-label learning for subcellular localization prediction of human proteins with both single and multiple sites. PloS One 7(6), 7155 (2012)
8. Diplaris, S., Tsoumakas, G., Mitkas, P.A., Vlahavas, I.: Protein Classification with Multiple Algorithms. In: Bozanis, P., Houstis, E.N. (eds.) PCI 2005. LNCS, vol. 3746, pp. 448–456. Springer, Heidelberg (2005)
9. Godbole, S., Sarawagi, S.: Discriminative Methods for Multi-Labeled Classification. In: Dai, H., Srikant, R., Zhang, C. (eds.) PAKDD 2004. LNCS (LNAI), vol. 3056, pp. 22–30. Springer, Heidelberg (2004)
10. Hüllermeier, E., Fürnkranz, J., Cheng, W., Brinker, K.: Label ranking by learning pairwise preferences. Artificial Intelligence 172(16), 1897–1916 (2008)
11. Boutell, M., Luo, J., Shen, X., Brown, C.: Learning multi-label scene classification. Pattern Recognit. 37(9), 1757–1771 (2004)
12. Clare, A., King, R.D.: Knowledge discovery in multi-label phenotype data. In: Siebes, A., De Raedt, L. (eds.) PKDD 2001. LNCS (LNAI), vol. 2168, pp. 42–53. Springer, Heidelberg (2001)
13. Zhang, M., Zhou, Z.: ML-KNN: A lazy learning approach to multi-label learning. Pattern Recognit. 40(7), 2038–2048 (2007)
14. Zhang, M.-L.: Ml-rbf: RBF Neural Networks for Multi-label Learning. Neural Process. Lett. 29, 61–74 (2009)
15. Elisseeff, A., Weston, J.: A Kernel Method for Multi-Labelled Classification. In: Adv. Neural Inf. Processing Systems 14, vol. 14, pp. 681–687. MIT Press (2001)
16. Japkowicz, N., Stephen, S.: The class imbalance problem: A systematic study. Intell. Data Anal. 6(5), 429–449 (2002)
17. Japkowicz, N.: Learning from imbalanced data sets: A comparison of various strategies, pp. 10–15. AAAI Press (2000)
18. Provost, F., Fawcett, T.: Robust classification for imprecise environments. Machine Learning 42, 203–231 (2001)
19. Kotsiantis, S.B., Pintelas, P.E.: Mixture of expert agents for handling imbalanced data sets. Annals of Mathematics, Computing & Teleinformatics 1, 46–55 (2003)
20. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: Synthetic minority over-sampling technique. J. Artif. Intell. Res. 16, 321–357 (2002)
21. Tsoumakas, G., Xioufis, E.S., Vilcek, J., Vlahavas, I.: MULAN multi-label dataset repository, `http://mulan.sourceforge.net/datasets.html`
22. Tsoumakas, G., Vlahavas, I.: Random k-labelsets: An ensemble method for multilabel classification. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) ECML 2007. LNCS (LNAI), vol. 4701, pp. 406–417. Springer, Heidelberg (2007)

23. Tsoumakas, G., Katakis, I., Vlahavas, I.: Effective and Efficient Multilabel Classification in Domains with Large Number of Labels. In: Proc. ECML/PKDD Workshop on Mining Multidimensional Data, pp. 30–44 (2008)
24. Sechidis, K., Tsoumakas, G., Vlahavas, I.: On the stratification of multi-label data. In: Gunopulos, D., Hofmann, T., Malerba, D., Vazirgiannis, M. (eds.) ECML PKDD 2011, Part III. LNCS, vol. 6913, pp. 145–158. Springer, Heidelberg (2011)
25. Fürnkranz, J., Hüllermeier, E., Loza Mencía, E., Brinker, K.: Multilabel classification via calibrated label ranking. Mach. Learn. 73, 133–153 (2008)
26. Cheng, W., Hüllermeier, E.: Combining instance-based learning and logistic regression for multilabel classification. Mach. Learn. 76(2-3), 211–225 (2009)