SOUP-Bagging: a new approach for multi-class imbalanced data classification

Mateusz Lango and Jerzy Stefanowski

Institute of Computer Science, Poznan University of Technology, Poznań, Poland
{mlango,jstefanowski}@cs.put.poznan.pl

1 Introduction

Learning from imbalanced data is an important and prevalent issue in machine learning research and applications [2]. Class-imbalanced data occur in many applications such as fraud detection, network intrusion detection, sentiment analysis, predictive maintenance and in the medical domain. As the standard classifiers fail to sufficiently recognize the minority classes, many novel algorithms have been developed in recent years. They are usually categorized into three groups: data-level, algorithmic-level and ensemble methods. The first data-level methods modify the original distribution of the data in order to improve the classification of minority classes. They can be used with virtually any classification algorithm. Algorithmic techniques modify a particular classification algorithm trying to make it more accurate for class-imbalanced problems. The ensemble methods exploit both these directions in the construction of combined classifier set, e.g. they generalize bagging and boosting schema and additionally incorporate data pre-preprocessing of the training subsets. Following [1] they are quite efficient in improving prediction measures for minority classes.

Most research has been devoted to a binary version of imbalanced data with a single minority class and a single majority class. There are also imbalanced problems with several important minority classes and in the last years, the increasing research interest has been observed on that issue [7]. It has been addressed mainly by adapting binary decomposition strategies (e.g. pairwise ensembles) or by proposing simple modifications of re-sampling methods. However, some of these researchers questioned the initial belief that multi-class imbalanced learning can be solved by simple decomposition into binary problems [4]. In particular, they postulated that these techniques are insufficiently dealing with complex interrelations which occur between classes. For instance, a class of average size can act as a minority class in the region dominated by majority class, at the same time causing difficulties in the recognition of other, smaller classes.

In our previous research, we introduced a new approach for examining the interrelations of multiple classes in imbalanced data [6]. It is based on analyzing the neighborhood of minority class examples and on the additional information about similarities between classes. Recently, we exploited this idea in the new resampling approach called Similarity Oversampling and Undersampling Preprocessing (SOUP) [3]. Even though in the experimental evaluation SOUP proved

to be an efficient approach for dealing with multiple imbalanced classes, the possibility of constructing an ensemble classifier was not considered. In this paper, we put forward the proposition of SOUP-Bagging - a bagging-based ensemble algorithm which leverages SOUP during its training.

2 SOUP Pre-processing Algorithm

Due to page limits we will only provide a brief description of Similarity Oversampling and Undersampling Preprocessing (SOUP) [3] which is directly related to this paper. A reader interested in more details has to consult [6, 3].

SOUP, as its name suggests, combine oversampling with undersampling to achieve a balanced class distribution in the training set. After SOUP resampling all classes have the same cardinality being equal the mean of the biggest minority and the smallest majority class sizes. This causes that (except corner cases) all the minority classes are oversampled and all the majority classes are undersampled by the algorithm. Both under- and oversampling is not performed randomly and the weight based selection of instances to be resampled is the key ingredient of SOUP algorithm, where inspirations from [6] are utilised to establish the level of difficulty of each example.

During undersampling, SOUP tries to clear the decision boundary from majority instances at the same time strengthening the minority class concepts with oversampling. To this end, a notion of a safe level is used. The safe level of an instance x belonging to class C_i is defined as

$$safe(x_{C_i}) = \frac{1}{k} \sum_{j=1}^{l} n_{C_j} \mu_{ij} \tag{1}$$

where n_{C_j} is the number of k-nearest neighbors of x which belong to C_j class and μ_{ij} is the special *degree of similarity between classes* C_i and C_j, which allows us to model interrelations between classes [6]. This degree is defined[1] by

$$\mu_{ij} = \frac{\min(|C_i|, |C_j|)}{\max(|C_i|, |C_j|)} \tag{2}$$

where $|C_i|$ is the size of C_i class. Safe level of an examples is higher in the clear homogenuous regions, dominated by the example's class. In the presence of instances from other classes the safe level decreases, taking into account sizes of surrounding classes. If the classes are of roughly the same size, safe level does not drop significantly, but together with bigger discrepancies between class sizes the decrease is more notable. SOUP uses this properties of the safe level to clean decision boundary from majority examples by undersampling instances with the lowest safe level values. On the other hand, safe regions of minority class are enlarged by duplicating examples with highest safe levels.

[1] In the original SOUP paper, authors suggest that μ_{ij} should be provided by a domain expert. Here, for simplicity we provide a heuristic which is used in SOUP-Bagging.

Algorithm 1 Similarity Oversampling and Undersampling Preprocessing (SOUP)

Input: D: original training set of $|D|$ examples with c classes; C_{min}: indexes of minority classes; C_{maj}: indexes of majority classes
Output: D': balanced training set

1: Split dataset D into c homogeneous parts $D_1, D_2, ..., D_c$. Each D_i contains all examples from i class
2: $D' = \emptyset$
3: $m \leftarrow mean(\min_{i \in C_{maj}} |D_i|, \max_{j \in C_{min}} |D_j|)$
4: **for all** $i \in C$ **do**
5: **for all** $x \in D_i$ **do**
6: find k nearest neighbours of x
7: calculate safe level of x, according to Eq. 1
8: **end for**
9: **if** $|D_i| > m$ **then**
10: remove $|D_i| - m$ examples with the lowest safe level values from D_i
11: **else**
12: duplicate $m - |D_i|$ examples with the highest safe level values in D_i
13: **end if**
14: $D' \leftarrow D' \cup D_i$
15: **end for**
16: **return** D'

The experimental evaluation of SOUP was performed over 19 imbalanced datasets [3] and it was compared against decomposition ensembles, resampling methods, and Multi-class Roughly Balanced Bagging (MRBB) [5]. SOUP stood out as the best performing method for decision trees (J48) and k-nearest neighbour classifier, loosing only to MRBB while using PART rules. Nevertheless, this result raises a question about the possibility of further improvement by ensembling techniques.

3 SOUP-Bagging

We investigate the possibility of improving SOUP by combining it with bagging which was often successfully generalized for binary complex imbalanced datasets. Moreover, bagging-based MRBB proved to be useful in multi-class problems [5] and it worked better than decomposition-based ensembles [3].

We introduce SOUP-Bagging algorithm whose pseudocode is presented in Alg. 2. The method iteratively resamples the original dataset with replacement, applies SOUP preprocessing technique and constructs a classifier. While resampling the dataset, stratified sampling is used. Predictions of the component classifiers are aggregated by the majority voting.

We carry out its experimental evaluation using the same real datasets used in [3]. Table 1 presents average ranks (as in the Friedman test) of G-mean measure while using J48 classifier. SOUP-Bagging stood out as the best-performing

Algorithm 2 SOUP-Bagging

Input: D: original training set of examples of size N, k: number of bootstrap samples, LA: learning algorithm;
Output: C^* bagging ensemble with k component classifiers
Learning phase:

1: **for** $i = 1 \rightarrow k$ **do**
2: $S_i \leftarrow N$-element sample drawn with replacement from D
3: $S_i \leftarrow$ SOUP (S_i)
4: $C_i \leftarrow LA(S_i)$
5: **end for**

Prediction phase:

$$C^*(x) = \arg\max_y \sum_{i=1}^{k} p_{C_i}(y|x)$$

Table 1. Average rank (like in the Friedman test) of G-mean obtained by algorithms with J48 classifier.

Algorithm	SOUP-Bagging	SOUP OVO	RUS OVO	ROS MRBB	Global-CS	Static-SMOTE
Average rank	2.80	3.30	3.30	3.56	3.90	4.93	6.20

approach, however, the difference between SOUP and SOUP-Bagging is not statistically significant according to the pairwise Wilcoxon test. Nevertheless, the difference between SOUP-Bagging and the other bagging-based approach which achieved the best results in our previous studies is statistically significant.

Summary: In this paper we promote new approach to deal with complex interrelation between multiple imbalanced classes. We partly summarize earlier research and introduce their new generalization into SOUP-Bagging.

References

1. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F.: A review on ensembles for the class imbalance problem: bagging-, boosting, and hybrid-based approaches. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(4), 463–484, (2011).
2. Haibo He, Edwardo A. Garcia: Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263–1284, (2009).
3. Janicka M., Lango M., Stefanowski J.: Using information on class interrelations to improve classification of multi-class imbalanced data: a new re-sampling algorithm, International Journal of Applied Mathematics and Computer Science, 29 (4), (2019).
4. Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. Progress in Artificial Intelligence, 5(4), 221–232 (2016).
5. Lango M., Stefanowski J.: Multi-class and Feature Selection Extensions of Roughly Balanced Bagging for Imbalanced Data. JIIS, 50 (1), 97–127 (2017) .
6. Lango, M., Napierala, K. and Stefanowski, J.: Evaluating difficulty of multi-class imbalanced data, Proc. 23rd Inter- national Symposium ISMIS, pp. 312–322, (2017).
7. Wang, S., Yao, X: Multiclass imbalance problems: Analysis and potential solutions. IEEE Trans. on Systems, Man, and Cybernetics, Part B, 42(4), 1119–1130 (2012).