New Ordering-Based Pruning Metrics for Ensembles of Classifiers in Imbalanced Datasets

Mikel Galar[1], Alberto Fernández[2], Edurne Barrenechea[1], Humberto Bustince[1], and Francisco Herrera[3]

[1] Departamento de Automática y Computación, ISC (Institute of Smart Cities), Universidad Pública de Navarra, Pamplona, Spain
{mikel.galar,edurne.barrenechea,bustince}@unavarra.es,
[2] Department of Computer Science, University of Jaén, Jaén, Spain
alberto.fernandez@ujaen.es
[3] Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain
herrera@decsai.ugr.es

Abstract. The task of classification with imbalanced datasets have attracted quite interest from researchers in the last years. The reason behind this fact is that many applications and real problems present this feature, causing standard learning algorithms not reaching the expected performance. Accordingly, many approaches have been designed to address this problem from different perspectives, i.e., data preprocessing, algorithmic modification, and cost-sensitive learning.
The extension of the former techniques to ensembles of classifiers has shown to be very effective in terms of quality of the output models. However, the optimal value for the number of classifiers in the pool cannot be known a priori, which can alter the behaviour of the system. For this reason, ordering-based pruning techniques have been proposed to address this issue in standard classifier learning problems. The hitch is that those metrics are not designed specifically for imbalanced classification, thus hindering the performance in this context.
In this work we propose two novel adaptations for ordering-based pruning metrics in imbalanced classification, specifically the margin distance minimization and the boosting-based approach. Throughout a complete experimental study, our analysis shows the goodness of both schemes in contrast with the unpruned ensembles and the standard pruning metrics in Bagging-based ensembles.

Keywords: Imbalanced Datasets, Ensembles, Ordering-Based Pruning, Bagging

1 Introduction

The unequal distribution among examples of different classes in classification tasks is known as the problem of imbalanced datasets [9, 22]. The use of standard algorithms in this framework lead to undesirable solutions as the model is usually biased towards the most represented concepts of the problem [13]. Therefore, several approaches have been developed for addressing this issue, which can be divided into three large groups including preprocessing for resampling the training set [3], algorithmic adaptation of standard methods [2], and cost-sensitive learning [25]. Additionally, all these schemes can be integrated into an ensemble learning algorithm, increasing the capabilities and performance of the baseline approach [8, 7, 13].

An ensembles is a set of classifiers where its components are supposed to complement each other, so that the learning space is completely covered and the generalization capability is enhanced with respect to the single baseline learning classifier [18, 21]. When classifying a new instance, all individual members are queried and their decision is obtained in agreement. The total number of classifiers that compose an ensemble is not a synonym of its quality and performance [27], since several issues that can degrade its behavior must be taken into account: (1) the time elapsed in the learning and prediction stages; (2) the memory requirements; and (3) contradictions and/or redundance among components of the ensemble.

In accordance with the above, several proposals have been developed to carry out a pruning of classifiers within the ensemble [26]. Specifically, ordering-based pruning is based on a greedy approach that adds classifiers iteratively to the final set with respect to the maximization of a given heuristic metric, until a preestablished number of classifiers are selected [10, 15].

In this contribution, we aim at developing an adaptation of two popular metrics towards the scenario of classification with imbalanced datasets, i.e. Margin Distance Minimization (MDM) and Boosting-Based pruning (BB) [6, 16]. Specifically, we consider that the effect of each classifier in both classes must be analyzed after the construction of the classifier and not only before (for example, rebalancing the dataset).

The goodness of this novel proposal is analyzed by means of a thorough experimental study, including a number of 66 different imbalanced problems. We have selected SMOTE-Bagging [23] and Under-Bagging [1] as ensemble learning schemes which, despite of being simple approaches, have shown to achieve a higher performance than many other more complex algorithms [8]. As in other related studies, we have selected the well-known C4.5 algorithm as baseline classifier [20]. Finally, our results are supported by means of non-parametric statistical tests [5].

In order to do so, this work is organized as follows. Section 2 briefly introduces the problem of imbalanced datasets. Then, Section 3 presents ordering-based pruning methodology, in which we describe standard metrics for performing this process and our adaptations to imbalanced classification. Next, the details about the experimental framework are provided in Section 4. The analysis and discussion of the experimental results are carried out in Section 5. Finally, Section 6 summarizes and concludes the work.

2 Basic Concepts on Classification with Imbalanced Datasets

Classification with imbalanced datasets appears when the distribution of instances between the classes of a given problem is quite different [13, 19]. Therefore, this classification task needs a special treatment in order to carry out an accurate discrimination between both concepts, independently of their representation.

The presence of classes with few data can generate sub-optimal classification models, since there is a bias towards the majority class when the learning process is guided by the standard accuracy metric. Furthermore, recent studies have shown that other data intrinsic characteristics have a significant influence for the correct identification of the minority class examples [13]. Some examples are overlapping, small-disjuncts, noise, and dataset shift.

Solutions developed to address this problem can be categorized into three large groups [13]: (1) *data level solutions* [3], (2) *algorithmic level solutions* [2], and (3) *cost-sensitive solutions* [25]. Additionally, when the former approaches are integrated within an ensemble of classifiers, their effectiveness is enhanced [8, 13].

Finally, in order to evaluate the performance in such a particular classification scenario, the metrics used must be designed to take into account the class distribution. One commonly considered alternative is the Area Under the ROC curve (AUC) [11]. In those cases where the used classifier outputs a single solution, this measure can be simply computed by the following formula:

$$AUC = \frac{1 + TP_{rate} - FP_{rate}}{2} \tag{1}$$

where $TP_{rate} = \frac{TP}{TP+FN}$ and $FP_{rate} = \frac{FP}{TN+FP}$.

3 A Proposal for Ordering-Based Pruning Scheme for Ensembles in Imbalanced Domains

Ensemble-based classifiers [18] are composed by a set of so-called *weak learners*, i.e., low changes in data produce big changes in the induced models. Diversity is quite significant in the performance of this type of approach, implying that individual classifiers must be focused on different parts of the problem space [12]. There are mainly two types of ensemble techniques: Bagging [4] and Boosting [6].

In this work, we will focus on the first scheme, due to the simplicity for the integration of data preprocessing techniques [8]. In this methodology, an ensemble of classifiers is trained with different sets of random instances from the original training data. When classifying a new sample, all individual classifiers are fired and a majority or weighted vote is used to infer the class.

The first parameter to take into account when building these types of models is the number of classifiers considered in the ensemble. In this sense, pruning methods were designed to obtain the "optimal" number of classifiers by carrying out a selection from a given pool of components of the ensemble. The hypothesis is that accuracy generally increases monotonically as more elements are added to the ensemble [10, 15, 16]. Most of pruning techniques make use of an heuristic function to seek for the reduced set of classifiers. In the case of *ordering-based pruning*, a metric that measures the goodness of adding each classifier to the ensemble is defined and the classifier with the highest value is added to the final sub-ensemble. The same process is performed until the size of the sub-ensemble reaches the specified parameter value.

In this work, we study two popular pruning metrics MDM and BB [6, 16]. We describe both schemes and our adaptation to imbalanced classification below:

- *MDM* is based on certain distances among the output vectors of the ensembles. These output vectors have the length equal to the training set size, and their value at the i^{th} position is either 1 or -1 depending on whether the i^{th} example is classified or misclassified by the classifier. The signature vector of a sub-ensemble is computed as the sum of the vectors of the selected classifiers. To summarize, the aim is to add those classifiers with the objective of obtaining a signature vector of the sub-ensemble where all the components are positive, i.e., all examples are correctly predicted. For a wider description please refer to [15].
  This method selects the classifier to be added depending on the closest Euclidean distance between an objective point (where every components are positive) and the signature vector of the sub-ensemble after adding the corresponding classifier. As a consequence, every example has the same weight in the computation of the distance, which can bias the selection to those classifiers favoring the majority class. Therefore, we compute the distance for the majority class examples and minority class examples independently. Then, distances are normalized by the number of examples used to compute them and added afterwards. That is, the same weight is given to both classes in the distance. This new metric is noted as *MDM-Imb*.
- *BB* selects the classifier that minimizes the cost with respect to the boosting scheme. This means that boosting algorithm is applied to compute the weights (costs) for each example in each iteration, but instead of training a classifier with these weights, the one that obtains the lowest cost from those in the pool is added to the sub-ensemble and weights are updated accordingly. Hence, it makes no difference whether classifiers were already learned using a boosting scheme or not. Different from the original boosting method, when no classifier has a weighted training error below 50%, weights are reinitialized (equal weights for all the examples) and the method continues (whereas in boosting it is stopped). Once classifiers are selected the scores assigned to each classifier by boosting are forgotten and not taken into account in the aggregation phase.
  It is well-known that boosting by itself is not capable of managing class imbalance problem [8]. For this reason, we have also adapted this approach in a similar manner as in the case of *MDM*. In boosting, every example has

initially the same weight and these are updated according to whether they are correctly classified or not. Even though minority class instances should get larger weights if they are misclassified, these weights can be negligible compared with those of the majority class examples. Hence, before finding the classifier that minimizes the total cost, we normalize the weights of the examples of each class by half of their sum, so that both classes has the same importance when selecting the classifier (even though each example of each class would have a different weight). This is only done before selecting the classifier, and then weights are updated according to the original (non-normalized ones). This working procedure tries to be similar to that successfully applied in several boosting models such as EUS-Boost [7]. This second weighting approach is noted as *BB-Imb*.

4 Experimental Framework

Table 1 shows the benchmark problems selected for our study, in which the name, number of examples, number of attributes, and IR (ratio between the majority and minority class instances) are shown. Datasets are ordered with respect to their degree of imbalance. Multi-class problems were modified to obtain two-class imbalanced problems, defining the joint of one or more classes as positive and the joint of one or more classes as negative, as defined in the name of the dataset. A wider description for these problems can be found at http://www.keel.es/datasets.php.

The estimates of AUC measure are obtained by means of a Distribution Optimally Balanced Stratified Cross-Validation (DOB-SCV) [17], as suggested in the specialized literature for working in imbalanced classification [14]. Cross-validation procedure is carried out using 5 folds, aiming to include enough positive class instances in the different folds. In accordance with the stochastic nature of the learning methods, these 5 folds are generated with 5 different seeds, and each one of the 5-fold cross-validation is run 5 times. Therefore, experimental results are computed with the average of 125 runs.

As ensemble techniques, we will make use of SMOTE-Bagging [23] and Under-Bagging [1]. In order to apply the pruning procedure, we will learn a number of 100 *classifiers* for each ensemble, choosing a subset of only 21 *classifiers* as suggested in the specialized literature [15]. The baseline ensemble models for comparison will use 40 classifiers as recommended in [8]. For *SMOTE-Bagging*, SMOTE configuration will be the standard with a 50% class distribution, 5 neighbors for generating the synthetic samples, and Heterogeneous Value Difference Metric for computing the distance among the examples. Finally, both learning approaches include the C4.5 decision tree [20] as baseline classifier, using a confidence level at 0.25, with 2 as the minimum number of item-sets per leaf, and the application of pruning will be used to obtain the final tree. Reader may refer to [8] in order to get a thorough description of the former ensemble methods.

Table 1. Summary of imbalanced datasets used

Name	#Ex.	#Atts.	IR	Name	#Ex.	#Atts.	IR
glass1	214	9	1.82	glass04vs5	92	9	9.22
ecoli0vs1	220	7	1.86	ecoli0346vs5	205	7	9.25
wisconsin	683	9	1.86	ecoli0347vs56	257	7	9.28
pima	768	8	1.87	yeast05679vs4	528	8	9.35
iris0	150	4	2.00	ecoli067vs5	220	6	10.00
glass0	214	9	2.06	vowel0	988	13	10.10
yeast1	1484	8	2.46	glass016vs2	192	9	10.29
vehicle2	846	18	2.52	glass2	214	9	10.39
vehicle1	846	18	2.52	ecoli0147vs2356	336	7	10.59
vehicle3	846	18	2.52	led7digit02456789vs1	443	7	10.97
haberman	306	3	2.78	ecoli01vs5	240	6	11.00
glass0123vs456	214	9	3.19	glass06vs5	108	9	11.00
vehicle0	846	18	3.25	glass0146vs2	205	9	11.06
ecoli1	336	7	3.36	ecoli0147vs56	332	6	12.28
newthyroid2	215	5	4.92	cleveland0vs4	1771	13	12.62
newthyroid1	215	5	5.14	ecoli0146vs5	280	6	13.00
ecoli2	336	7	5.46	ecoli4	336	7	13.84
segment0	2308	19	6.01	shuttle0vs4	1829	9	13.87
glass6	214	9	6.38	yeast1vs7	459	8	13.87
yeast3	1484	8	8.11	glass4	214	9	15.47
ecoli3	336	7	8.19	pageblocks13vs4	472	10	15.85
pageblocks0	5472	10	8.77	abalone918	731	8	16.68
ecoli034vs5	200	7	9.00	glass016vs5	184	9	19.44
yeast2vs4	514	8	9.08	shuttle2vs4	129	9	20.50
ecoli067vs35	222	7	9.09	yeast1458vs7	693	8	22.10
ecoli0234vs5	202	7	9.10	glass5	214	9	22.81
glass015vs2	506	8	9.12	yeast2vs8	482	8	23.10
yeast0359vs78	172	9	9.12	yeast4	1484	8	28.41
yeast0256vs3789	1004	8	9.14	yeast1289vs7	947	8	30.56
yeast02579vs368	1004	8	9.14	yeast5	1484	8	32.73
ecoli046vs5	203	6	9.15	yeast6	1484	8	41.40
ecoli01vs235	244	7	9.17	ecoli0137vs26	281	7	39.15
ecoli0267vs35	244	7	9.18	abalone19	4174	8	129.44

Finally, we will make use of Wilcoxon signed-rank test [24] to find out whether significant differences exist between a pair of algorithms.

5 Experimental Study

Our analysis is focused on determining whether the new proposed metrics, specifically designed for dealing with class imbalance, are well-suited for this problem with respect to the original metrics, i.e., *BB* and *MDM*. Additionally, we will analyze the improvement in the performance results with respect to the original ensemble model. The average values for the experimental results are shown in Table 2, whereas full results are shown in Table 3.

Table 2. Average test results for the standard ensemble approach (Base) and the ordering-based pruning with the original (BB and MDM) and imbalanced pruning metrics (BB-Imb and MDM-Imb).

Ensemble	Base	BB	BB-Imb	MDM	MDM-Imb
SMOTE-Bagging	.8645 $\pm$ .0587	.8602 $\pm$ .0632	**.8635 $\pm$ .0610**	.8596 $\pm$ .0629	**.8625 $\pm$ .0622**
Under-Bagging	.8647 $\pm$ .0516	**.8755 $\pm$ .0564**	.8734 $\pm$ .0544	.8653 $\pm$ .0563	**.8699 $\pm$ .0558**

Regarding the comparison between the pruning schemes, in the case of *BB* and *BB-Imb* we find that for SMOTE-Bagging the metric adapted for imbalanced

Table 3. Test results for the standard ensemble (Base) and ordering-based pruning schemes (BB, BB-Imb, MDM, and MDM-Imb) using AUC metric.

	SMOTE-Bagging					Under-Bagging				
Dataset	Base	BB	BB-Imb	MDM	MDM-Imb	Std.	BB	BB-Imb	MDM	MDM-Imb
glass1	.7675	.8021	.7925	.7866	.7900	.7686	.7979	.7927	.7918	.7928
ecoli0vs1	.9812	.9750	.9763	.9802	.9788	.9805	.9806	.9764	.9826	.9809
wisconsin	.9707	.9692	.9700	.9662	.9666	.9691	.9698	.9704	.9678	.9672
pima	.7568	.7451	.7546	.7500	.7558	.7598	.7561	.7548	.7532	.7539
iris0	.9880	.9888	.9880	.9880	.9880	.9900	.9900	.9900	.9900	.9900
glass0	.8347	.8517	.8464	.8413	.8430	.8264	.8469	.8438	.8399	.8352
yeast1	.7312	.7192	.7321	.7301	.7315	.7304	.7333	.7310	.7331	.7307
vehicle2	.9723	.9752	.9734	.9686	.9691	.9704	.9750	.9744	.9680	.9686
vehicle1	.7848	.7691	.7918	.7898	.7934	.8016	.8020	.7983	.7959	.7985
vehicle3	.7784	.7593	.7827	.7795	.7808	.8060	.7979	.7976	.7966	.7974
haberman	.6627	.6517	.6476	.6500	.6498	.6627	.6616	.6486	.6488	.6620
glass0123vs456	.9405	.9318	.9357	.9308	.9378	.9335	.9432	.9379	.9264	.9337
vehicle0	.9635	.9630	.9636	.9609	.9614	.9492	.9558	.9595	.9539	.9544
ecoli1	.9053	.8988	.9067	.9044	.9107	.8988	.8981	.9101	.9043	.9123
newthyroid2	.9642	.9540	.9586	.9567	.9577	.9605	.9572	.9696	.9614	.9692
newthyroid1	.9558	.9460	.9486	.9456	.9467	.9490	.9479	.9550	.9594	.9613
ecoli2	.9145	.9153	.9128	.9131	.9099	.9054	.9057	.8996	.9017	.8996
segment0	.9917	.9917	.9924	.9922	.9926	.9866	.9881	.9887	.9872	.9878
glass6	.9291	.9164	.9213	.9157	.9203	.9096	.9277	.9248	.9228	.9190
yeast3	.9330	.9308	.9325	.9315	.9329	.9311	.9326	.9305	.9311	.9295
ecoli3	.8462	.8508	.8560	.8506	.8514	.8830	.8702	.8670	.8793	.8707
pageblocks0	.9580	.9552	.9585	.9572	.9581	.9610	.9631	.9626	.9612	.9615
ecoli034vs5	.9129	.9032	.9018	.9029	.8948	.8922	.9148	.9203	.8701	.9037
yeast2vs4	.9277	.9192	.9155	.9123	.9223	.9445	.9408	.9482	.9383	.9536
ecoli067vs35	.8576	.8651	.8626	.8653	.8630	.8582	.8624	.8578	.8670	.8523
ecoli0234vs5	.9007	.9008	.9036	.8935	.8939	.8641	.9053	.9027	.8404	.8784
glass015vs2	.7041	.7004	.7015	.7052	.7025	.7412	.7117	.7604	.7553	.7628
yeast0359vs78	.7173	.7023	.7174	.7016	.7134	.7373	.7414	.7386	.7394	.7387
yeast02579vs368	.8028	.7982	.7995	.7927	.7993	.8159	.8090	.8068	.8136	.8075
yeast0256vs3789	.9183	.9173	.9176	.9150	.9185	.9149	.9136	.9099	.9140	.9098
ecoli046vs5	.9132	.9086	.9114	.9046	.9083	.8869	.9188	.9238	.8666	.9123
ecoli01vs235	.8988	.8665	.8815	.8789	.8883	.8815	.9031	.9047	.8893	.8942
ecoli0267vs35	.8617	.8544	.8611	.8664	.8642	.8573	.8623	.8556	.8662	.8483
glass04vs5	.9910	.9836	.9879	.9876	.9869	.9940	.9900	.9940	.9940	.9940
ecoli0346vs5	.8921	.8888	.8929	.8762	.8884	.8799	.8961	.9051	.8618	.8956
ecoli0347vs56	.8595	.8701	.8707	.8590	.8643	.8762	.8875	.8897	.9009	.8800
yeast05679vs4	.8177	.8152	.8133	.8088	.8124	.8209	.8287	.8189	.8018	.8182
ecoli067vs5	.8897	.8894	.8888	.8909	.8886	.8820	.8883	.8888	.9028	.8779
vowel0	.9878	.9874	.9880	.9838	.9853	.9588	.9671	.9684	.9689	.9685
glass016vs2	.7009	.7083	.7176	.7168	.7214	.7025	.7185	.7291	.7265	.7323
glass2	.7425	.7390	.7436	.7458	.7458	.7569	.7394	.7691	.7452	.7702
ecoli0147vs2356	.8685	.8637	.8719	.8673	.8793	.8328	.8625	.8536	.8665	.8468
led7digit02456789vs1	.8466	.8547	.8407	.8500	.8383	.8268	.8397	.8322	.8449	.8399
ecoli01vs5	.8881	.8786	.8782	.8688	.8755	.8726	.9142	.9174	.8795	.8937
glass06vs5	.9926	.9954	.9954	.9916	.9912	.9151	.9910	.9940	.9940	.9940
glass0146vs2	.6961	.7161	.7295	.7189	.7254	.7214	.7335	.7336	.7323	.7434
ecoli0147vs56	.8703	.8848	.8804	.8682	.8750	.8738	.9035	.8870	.8819	.8756
cleveland0vs4	.7894	.7933	.8004	.7815	.7835	.8492	.8714	.8305	.7917	.8069
ecoli0146vs5	.8875	.9037	.9022	.8828	.8994	.8933	.9197	.9273	.8639	.8988
ecoli4	.9245	.9220	.9247	.9094	.9135	.8952	.9357	.9349	.9017	.8969
shuttle0vs4	.9999	.9999	.9999	.9999	.9999	1.0000	1.0000	1.0000	1.0000	1.0000
yeast1vs7	.7458	.7354	.7349	.7368	.7303	.7661	.7869	.7852	.7463	.7824
glass4	.9069	.8795	.8788	.8716	.8675	.9065	.9182	.8903	.8943	.8882
pageblocks13vs4	.9952	.9932	.9964	.9963	.9963	.9804	.9937	.9946	.9928	.9928
abalone9vs18	.7120	.7140	.7076	.7090	.7085	.7560	.7490	.7388	.7222	.7354
glass016vs5	.9865	.9493	.9747	.9675	.9674	.9429	.9698	.9675	.9670	.9663
shuttle2vs4	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
yeast1458vs7	.6330	.6175	.6144	.6059	.6153	.6374	.6530	.6263	.6009	.6315
glass5	.9769	.9533	.9619	.9586	.9626	.9488	.9596	.9639	.9631	.9621
yeast2vs8	.8064	.7916	.7946	.8014	.8068	.7526	.7846	.7608	.7579	.7629
yeast4	.8211	.8117	.8114	.8046	.8124	.8420	.8534	.8537	.8416	.8543
yeast1289vs7	.7046	.6818	.6905	.6831	.7004	.7370	.7194	.7392	.6918	.7433
yeast5	.9622	.9536	.9581	.9525	.9585	.9593	.9689	.9673	.9623	.9625
yeast6	.8375	.8354	.8446	.8369	.8431	.8673	.8736	.8570	.8706	.8514
ecoli0137vs26	.8347	.8273	.8336	.8363	.8400	.7807	.8774	.7874	.8060	.7789
abalone19	.5432	.5380	.5447	.5375	.5462	.7121	.7034	.7251	.7213	.7307
Average	.8645	.8602	.8635	.8596	.8625	.8647	.8755	.8734	.8653	.8699

classification achieves a higher average performance. Regarding Under-Bagging, the relative differences are below 1% in favour of the standard approach. On the other hand, the analysis for *MDM* and *MDM-Imb* metrics shows the need for the imbalanced approach, as it stands out looking at the experimental results. Finally, the robustness of the imbalanced metrics must be stressed in accordance with the low standard deviation shown with respect to the standard case.

In order to determine statistically the best suited metric, we carry out a Wilcoxon pairwise test in Table 4. We have included a symbol for stressing whether significant differences are found at 95% confidence degree (*) or at 90% (+). Results of these tests agree with our previous remarks. The differences in the case of MDM are clear in favour of the imbalanced version. In the case of BB the behaviour vary depending on the ensemble technique, where significant differences are obtained for SMOTE-Bagging whereas none are found for Under-Bagging.

Table 4. Wilcoxon test for pruning metrics: standard [R^+] and imbalanced [R^-].

Ensemble	Comparison	R^+	R^-	p-value
SMOTE-Bagging	BB vs. BB-Imb	540.0	1671.0	0.00028*
	MDM vs. MDMimb	436.0	1775.0	0.00002
Under-Bagging	BB vs. BB-Imb	1277.0	934.0	0.27939
	MDM vs. MDMimb	831.5	1379.5	0.07246+

Finally, when we contrast these results versus the standard ensemble approach, we also observe a two-fold behaviour: in the case of SMOTE-Bagging the pruning approach enables the definition of a simpler ensemble with a low decrease of the performance, especially when the imbalanced metric is selected. On the other hand, for Under-Bagging we observe a notorious improvement of the results in all cases when the ordering-based pruning is applied, showing a better behaviour for MDM-Imb and especially in BB-Imb (see Tables 2 and 3). These findings are complemented by means of a Wilcoxon test (shown in Table 5), for which we observe significant differences in favour of the ordering-based pruning for the Under-Bagging approach.

Table 5. Wilcoxon test to compare the standard ensemble approach (Std.) [R^+] and the one with imbalanced ordering-based pruning [R^-].

Ensemble	Comparison	R^+	R^-	p-value
SMOTE-Bagging	Std. vs. BB-Imb	1261.5	883.5	0.215579
	Std. vs. MDMimb	1386.5	758.5	0.039856*
Under-Bagging	Std. vs. BB-Imb	502.0	1709.0	0.000114*
	Std. vs. MDMimb	637.0	1574.0	0.002735*

6 Concluding Remarks

Ordering-based pruning in ensembles of classifiers consists of carrying out a selection of those elements of the ensemble set that are expected to work with better synergy. The former process is guided by a given metric of performance which is focused on different capabilities of the ensemble. However, they have not been previously considered within been developed within the scenario of imbalanced datasets.

In this work, we have proposed two adaptations of metrics for ordering-based pruning in imbalanced classification, namely BB-Imb and MDM-Imb. The experimental analysis has shown the success of these novel metrics with respect to their original definition, especially in the case of the SMOTE-Bagging approach. Additionally, we have point out that a significant improvement in the behaviour of the Under-Bagging ensemble is achieved by means of the application of the ordering-based pruning, outperforming the results with respect to the original model.

As future work, we plan to include a larger number of pruning metrics and ensemble learning methodologies, aiming at giving additional support and strength to the findings obtained in this contribution.

7 Acknowledgement

This work was supported by the Spanish Ministry of Science and Technology under projects TIN-2011-28488, TIN-2012-33856, TIN2013-40765-P; the Andalusian Research Plans P11-TIC-7765 and P10-TIC-6858; and both the University of Jaén and Caja Rural Provincial de Jaén under project UJA2014/06/15.

References

1. Barandela, R., Valdovinos, R., Sánchez, J.: New applications of ensembles of classifiers. Pattern Analysis Applications 6(3), 245–256 (2003)
2. Barandela, R., Sánchez, J.S., García, V., Rangel, E.: Strategies for learning in class imbalance problems. Pattern Recognition 36(3), 849–851 (2003)
3. Batista, G.E.A.P.A., Prati, R.C., Monard, M.C.: A study of the behaviour of several methods for balancing machine learning training data. SIGKDD Explorations 6(1), 20–29 (2004)
4. Breiman, L.: Bagging predictors. Machine Learning 24(2), 123–140 (1996)
5. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 7, 1–30 (2006)
6. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences 55(1), 119–139 (1997)
7. Galar, M., Fernández, A., Barrenechea, E., , Herrera, F.: Eusboost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling. Pattern Recognition 46(12), 3460–3471 (2013)

8. Galar, M., Fernández, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for class imbalance problem: Bagging, boosting and hybrid based approaches. IEEE Transactions on System, Man and Cybernetics Part C: Applications and Reviews 42(4), 463–484 (2012)
9. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering 21(9), 1263–1284 (2009)
10. Hernández-Lobato, D., Martínez-Muñoz, G., Suárez, A.: Statistical instance-based pruning in ensembles of independent classifiers. IEEE Transactions On Pattern Analysis And Machine Intelligence 31(2), 364–369 (2009)
11. Huang, J., Ling, C.X.: Using AUC and accuracy in evaluating learning algorithms. IEEE Transactions on Knowledge and Data Engineering 17(3), 299–310 (2005)
12. Kuncheva, L.I.: Diversity in multiple classifier systems. Information Fusion 6(1), 3–4 (2005)
13. López, V., Fernández, A., García, S., Palade, V., Herrera, F.: An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. Information Sciences 250(20), 113–141 (2013)
14. López, V., Fernández, A., Herrera, F.: On the importance of the validation technique for classification with imbalanced datasets: Addressing covariate shift when data is skewed. Information Sciences 257, 1–13 (2014)
15. Martínez-Muñoz, G., Hernández-Lobato, D., Suárez, A.: An analysis of ensemble pruning techniques based on ordered aggregation. IEEE Transactions On Pattern Analysis And Machine Intelligence 31(2), 245–259 (2009)
16. Martínez-Muñoz, G., Suárez, A.: Using boosting to prune bagging ensembles. Pattern Recognition Letters 28(1), 156–165 (2007)
17. Moreno-Torres, J.G., Sáez, J.A., Herrera, F.: Study on the impact of partition-induced dataset shift on k-fold cross-validation. IEEE Transactions On Neural Networks And Learning Systems 23(8), 1304–1313 (2012)
18. Polikar, R.: Ensemble based systems in decision making. IEEE Circuits and Systems Magazine 6(3), 21–45 (2006)
19. Prati, R.C., Batista, G.E.A.P.A., Silva, D.F.: Class imbalance revisited: a new experimental setup to assess the performance of treatment methods. Knowledge and Information Systems in press, 1–25 (2014)
20. Quinlan, J.: C4.5: Programs for Machine Learning. Morgan Kauffman (1993)
21. Rokach, L.: Ensemble-based classifiers. Artificial Intelligence Review 33(1), 1–39 (2010)
22. Sun, Y., Wong, A.K.C., Kamel, M.S.: Classification of imbalanced data: A review. International Journal of Pattern Recognition and Artificial Intelligence 23(4), 687–719 (2009)
23. Wang, S., Yao, X.: Diversity analysis on imbalanced data sets by using ensemble models. In: Proceedings of the 2009 IEEE Symposium on Computational Intelligence and Data Mining (CIDM'09). pp. 324–331 (2009)
24. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics Bulletin 1(6), 80–83 (1945)
25. Zadrozny, B., Langford, J., Abe, N.: Cost–sensitive learning by cost–proportionate example weighting. In: Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM'03). pp. 435–442 (2003)
26. Zhang, Y., Burer, S., Street, W.N.: Ensemble pruning via semi-definite programming. Journal of Machine Learning Research 7, 1315–1338 (2006)
27. Zhou, Z.H., Wu, J., Tang, W.: Ensembling neural networks: Many could be better than all. Artificial Intelligence 137(1-2), 239–263 (2002)