A Voronoi Diagram Based Classifier for Multiclass Imbalanced Data Sets

Evandro J. R. Silva
Centro de Informática - CIn
Universidade Federal de Pernambuco - UFPE
Recife-PE, Brazil
Email: ejrs@cin.ufpe.br

Cleber Zanchettin
Centro de Informática - CIn
Universidade Federal de Pernambuco - UFPE
Recife-PE, Brazil
Email: cz@cin.ufpe.br

***Abstract*—The imbalance problem is receiving an increasing attention in the literature. Studies in binary cases are recurrent, however there still are several real world problems with more than two classes. The known solutions for binary datasets may not be applicable in this case. Some efforts are being applied in decomposition techniques which transforms a multiclass problem into some binary problems. However it is also possible to face a multiclass problem with an *ad hoc* approach, i.e., a classifier able to handle all classes at once. In this work a method able to handle several classes is proposed. This new method is based on the Voronoi diagram. We try to dynamically divide the feature space into several regions, each one assigned to a different class. It is expected for the method to be able to construct a complex classification model. However, as it is in its beginning, some tests need to be performed in order to evaluate its feasibility. Experiments with some classical classifiers confirm its feasibility, and comparisons with *ad hoc* methods found in literature show its potentiality.**

I. Introduction

The imbalance data problem occurs when a dataset has a different distribution of samples among the classes. In the literature this problem is well studied in the case of binary datasets [1], [2]. However there exist several multiclass imbalanced datasets which also need to be addressed. In this case the instance distribution may be different in more than two classes.

Unfortunately when multiple classes are present, the literature solutions proposed for the binary case may not be directly applicable, or may achieve a lower performance than expected [1]. A multiclass problem may also require a different focus. For example, in a binary case researchers focus on the correct classification of the minority class, as normally the classifier is skewed towards the majority class, and the minority is usually the most important one. With multiple classes it is possible the non existence of a *main* class.

In the literature there are three approaches to address multiclass problems. The first one deals with classifiers able of handling all classes at once. Classical algorithms in machine learning area such as decision trees and artificial neural networks are examples. The second one is called one-vs-one (OVO) [1], [3], [4] or one-against-one (OAO) [2]. The OVO strategy consists of dividing the problem into as many binary problems as the possible combinations between pairs of classes, so one classifier is trained to discriminate between each pair, and then the outputs of these base classifiers are combined in order to predict the output class [4]. The third approach is called one-vs-all (OVA) [1], [4] or one-against-all (OAA) [2]. The OVA approach trains a classifier for each class, where the class is distinguished from all other classes, so the base classifier with a positive answer indicates the output class [4].

Usually it is easier to build a classifier to distinguish between only two classes. Techniques that transform multiclass problems in binary (OVO and OVA) became the most common strategy to face multiclass imbalance problems [4]. These binarization strategies have in common the construction of an ensemble of classifiers and, consequently, share the inherent benefits on performance due to the use of these ensembles. However they also have some drawbacks. For example, in OVO each classifier is only trained with instances from the two classes the classifier must distinguish. The instances belonging to other classes are unknown to the classifier. During the classification process a new sample is presented to all binary classifiers, which must set a score for each one of the two trained classes. Since all outputs are aggregated, in the decision process both the competent and non-competent classifiers are taken into account, possibly misleading the correct sample label [5]. Hence, the use of the most appropriate classifier, or the competent one for a given class, becomes a problem itself.

Therefore, the use of the first approach, i.e., the construction of a classifier able to handle with several classes at once, is still practicable, and desirable as it is not the most common used technique [1], [2]. This type of classifier also allows the use of ensembles, which may improve its performance.

In this work we propose a different classification model to multiclass imbalanced datasets. Its base is the Voronoi diagram idea [6]. In this diagram a set of points $S = \{s_1, s_2, ..., s_n\}$, also known as seeds, is used to split regions. Each seed s_i has a corresponding region (Voronoi cell) where all points are closer to s_i than to any other seed.

The build of a Voronoi diagram is like creating a mosaic of regions and it does not need to learn/discover a function as artificial neural networks and other classifiers do. It is expected that a classifier based on Voronoi cells will be able to construct a complex classification model, independently of the number

Fig. 1. Pseudo-code of VDBC

```
/*Given all training instances*/
for each instance do
    Verify the nearest neighbor
    if nearest neighbor is not a seed then
        if nearest neighbor is from same class then
            Create seed between them
        else
            This instance becomes a seed
        end if
    else
        if nearest neighbor is from different class then
            This instance becomes a seed;
        end if
    end if
end for
/*Given all test instances*/
for each instance do
    Classify it accordingly to its nearest seed
end for
```

of classes. At the same time the proposed classification model will be very simple to build.

The proposed classifier, named as VDBC (Voronoi Diagram Based Classifier) is evaluated in different UCI [7] databases and compared to classical approaches to deal with imbalanced problems as well as with some approaches found in literature.

The remainder of this paper is organized as follows. In Sect. II we present some related works. Next, in Sect. III the proposed classification model is presented in details. In Sect. IV experiments and discussions are exposed. Sect. V presents some final remarks.

II. Related Work

In this section we present some related *ad hoc* classifiers for multiclass imbalanced problems.

In [8] the authors present the SMOTEBoost. This technique consists of applying SMOTE (Synthetic Minority Oversampling TEchnique) [9] after each boosting iteration. The authors justify that a boosting algorithm sampling from a pool of data that predominantly consists of the majority class, will be probably skewed towards the majority class. Moreover, although boosting reduces the variance and the bias in the final ensemble, it might not be as effective for datasets with skewed class distributions. Therefore, introducing SMOTE in each iteration of boosting will enable each learner to sample more of the minority class cases, and also learn better and broader decision regions for the minority class. The authors also imply that introducing the SMOTE procedure also increases the diversity amongst the classifiers in the ensemble, as in each iteration there is a production of a different set of synthetic examples, and therefore different classifiers. The used boosting procedure was a variant of the AdaBoost.M2 [10].

Wang and Yao [2] developed a study regarding the extension of boosting techniques for imbalance problems. Their approach is an AdaBoost algorithm [10] in combination with negative correlation learning [11]. The main procedure is quite similar to any boosting approach. However, the update of the weights depends on the classification or misclassification given by both the classifiers in the current iteration and the global ensemble. The initial weights in this boosting approach are assigned in inverse proportion to the number of instances in the corresponding class. The base learner used is the C4.5 classifier [12].

Sokol Koço and Cécile Capponi [13] introduced CoMBo (Confusion Matrix Boosting), an extension of AdaBoost.MM [14], that greedily minimizes the empirical norm of the confusion matrix. This process is done in such way that poorly represented classes are performed as well as majority classes within the overall learning process, independently from any prior misclassification cost. In other words the norm of the *empirical confusion matrix*[1] is used as a metric to be minimized by a boosting-based method.

CoMBo is also a cost sensitive method (as it uses a cost matrix) where the classification costs are given for each sample and class. However, contrary to usual cost sensitive methods, the matrix is not given *a priori* to the learning process. The matrix is updated after each iteration so that the misclassification cost reflects the difficulty of correctly classifying an sample. The update rule was built to depend not only on the ability of a classifier to correctly classify a hard example, but also on the number of samples that have the same class. The output hypothesis is a simple weighted majority vote over the whole set of weak classifiers. So, for a given example, the final prediction is the class that obtains the highest score.

III. Proposed Classification Model

The proposed classification model is based on the Voronoi diagram idea. The algorithm tries to dynamically create regions through the sample space, assigning to each cell a respective class. The seeds are created as follows. For each training instance its nearest neighbor is found. A class verification is executed. If the neighbor is from the same class a seed (or centroid) is created between them. Otherwise the current instance becomes the seed. However, if the neighbor is a seed, an action will be performed only if the seed belongs to a different class. In this case the current instance also becomes a seed.

All seeds are automatically assigned to the training sample class. After this *training* phase, the sample space is divided into several cells, each one assigned to a different class. With the diagram constructed the testing phase begins. In the testing phase, for each unknown instance its nearest neighbor, i.e., the nearest seed is found. The unknown instance is classified accordingly to the nearest seed. The pseudo-code of the proposed classification model is shown in Figure 1.

For experimental purposes three modifications were made in the original algorithm. In the next section all versions will be

[1] Please check [13] for the definition of *empirical confusion matrix*.

TABLE I
SUMMARY OF THE USED DATASETS

Name	#Classes	#	IR	MIR
Gene	3	762/765/1648	2.1627	0.4134
Glass	6	70/76/17/13/09/29	8.4444	5.0133
Horse	3	224/88/52	4.3077	1.2538
Page-Blocks	5	4913/329/28/88/115	175.4643	59.5996
Satimage	6	1533/703/1358/626/707/1508	2.4488	0.9564
Thyroid	3	166/368/6666	40.1566	18.3396
Yeast	10	463/05/35/44/51/163/244/429/20/30	92.6	44.7543

compared, and the best of them will be compared to classical algorithms.

The first modification corresponds to the addition of two or more neighbors in the NN-rule. We investigate the use of two, three, four and five neighbors. The remain of the algorithm does not suffer any modification, i.e., if the neighbors of an instance belong to the same class a seed is created among them, otherwise the instance becomes the seed, itself.

The second modification consists in the use of the SMOTE [9] in the training phase. Differently from Wang and Yao [2], we did not considered the existence of "multi-minority" or "multi-majority" classes. Wang and Yao created artificial datasets with majority classes and minority classes with the same sizes. For the "multi-minority" case they gathered together one majority class with several minority classes. In the "multi-majority" case the authors gathered together one minority class with several majority classes. However, as can be seen in Table I, in real world problems we may not have such clear distinction between classes. It is even possible to a dataset present both "multi-minority" and "multi-majority" cases. With this information in mind we decided to automatize the choice of which classes will be increased with SMOTE.

The SMOTE strategy is used as follows. Let $C = \{c_1, c_2, ..., c_n\}$ be the set of classes and $p(c_n)$ be the probability of a sample belong to class n. The average of all classes' probabilities is calculated and then compared to each single probability. The class with single probability lesser than the average is chosen to be increased via SMOTE. During the training phase, synthetic examples are created between each instance and all others of the same class.

The third modification is related to the degree of overlapping among classes. As is stated in Section IV VDBC decreases its performance when overlapping is present. To overcome this situation the following steps were implemented:

1) Calculate generalized Fisher Ratio (Table IV) [15], [16] for each pair of classes of the dataset;
2) Select the pairs in which Fisher Ratio values are less or equal to 0.15;
3) Find a centroid for each class belonging to the pair;
4) Create new instances, randomly, between centroids[2];
5) For each new synthetic instance assign one of two possible classes randomly.

IV. EXPERIMENTS AND DISCUSSION

In this section comparisons among the proposed algorithm and others algorithms (classical and *ad hoc*) are performed.

As can be seen in Section II all presented related works are based in boosting methods, which may give them some expected advantages inherent to the use of ensembles. However, as the proposed classification model is quite simple, a comparison with classical classifiers might show its feasibility, whereas comparisons with *ad hoc* methods might show how distant its performance is from some state-of-the art methods.

The experiments were performed in seven datasets from UCI repository [7]. Table I shows a summary of the datasets. One important thing in this table is the concept of MIR (Multiclass Imbalance Ratio), which is proposed in this paper.

In a binary case the Imbalanced Ratio (IR) is defined as the ratio between the number of instances of the majority class and the minority class [1]. However, as we have more than two classes, and there is no more a well defined concept of majority and minority classes, a different measure of imbalance need to be defined considering all classes and not only the extreme cases.

Let $C = \{c_1, c_2, ..., c_n\}$ be the set of classes and x_c the number of elements of the class c. The MIR can be defined as

$$MIR = \sum_{c=1}^{n} \frac{\frac{\sum\limits_{c=1}^{n} x_c}{n}}{x_c} - n \qquad (1)$$

The numerator part finds the size each class should have in such way the data set would be balanced. Furthermore this value is divided for each class size, creating an imbalanced ratio. The sum of all ratios less the number of classes creates the value of MIR. Notice that a completely balanced case has a MIR value of 0.

The MIR shows how much heterogeneous are the size of classes in the dataset. As higher is the value of MIR higher is the difference among classes regarding their sizes. Therefore a multiclass data set with a high value for MIR probably will face imbalance issues.

[2] The number of instances to be created in Step 4 is equal to the size of the biggest class in the pair.

TABLE II

AVERAGE MAUC OF VDBC WITH EACH NEIGHBORHOOD CONFIGURATION

Dataset	1NN	2NN	3NN	4NN	5NN
Gene	0.7172±0.0301	0.7570±0.0122	0.7601±0.0278	**0.7761**±0.0278	0.7640±0.0447
Glass	0.8497±0.0347	0.8575±0.0321	0.8640±0.0305	0.8621±0.0313	**0.8644**±0.0322
Horse	0.5917±0.0286	0.5940±0.0299	0.5961±0.0295	**0.5999**±0.0273	0.5977±0.0306
Page-Blocks	**0.7504**±0.0299	0.6850±0.0302	0.6861±0.0258	0.6927±0.0279	0.7063±0.0359
Satimage	0.9412±0.0040	**0.9476**±0.0038	0.9465±0.0035	0.9473±0.0043	0.9467±0.0035
Thyroid	**0.7624**±0.0224	0.7406±0.0193	0.7452±0.0168	0.7492±0.0227	0.7490±0.0212
Yeast	0.6622±0.0225	0.6670±0.0222	**0.6736**±0.0213	0.6687±0.0250	0.6696±0.0232

TABLE III

AVERAGE MAUC OF THREE VERSIONS OF VDBC

Dataset	Original	+ SMOTE	+ Fisher Ratio
Gene	0.7172±0.0301	0.6450±0.0110	**0.8837**±0.0107
Glass	0.8497±0.0347	0.8448±0.0329	**0.9381**±0.0291
Horse	0.5917±0.0286	0.5851±0.0262	**0.8429**±0.0393
Page-Blocks	0.7504±0.0299	0.7568±0.0308	**0.9039**±0.0211
Satimage	0.9412±0.0040	0.9453±0.0035	**0.9816**±0.0023
Thyroid	0.7624±0.0224	0.7566±0.0204	**0.8999**±0.0174
Yeast	0.6622±0.0225	0.6667±0.0250	**0.8297**±0.0376

However, in some cases the value of MIR appears to be lower than it should to be. This happens because the classes sizes are relatively homogeneous. For instance in the *Satimage* data set classes may be separated in two groups in which the first is the group of classes of size near 1500 instances and the other group with classes of size near 700 instances. Therefore its MIR value is low despite being imbalanced.

The experiments are organized as follows. A total of 100 runs were performed with each classifier[3], in each data set. In each run the training and test sets were built randomly, with a division of $2/3$ and $1/3$, respectively, but following the natural distribution of the classes. The results shown hereafter are the average of these 100 runs.

The metric used to evaluate the classifiers performances was the MAUC [17], an extension of AUC (Area Under ROC Curve) for multiclass problems. In previous work we investigated the Balanced Accuracy [18] and it showed to be completely equivalent to AUC in binary datasets. However with more than two classes the Balanced Accuracy behavior is completely different.

In multiclass scenario the Balanced Accuracy metric also shows another problem, specifically with the presence of rare classes. For example, if one class has only 3 instances for testing, and the classifier is able to correctly classify 2 of them, it will have a bad performance according to Balanced Accuracy. Using AUC and consequently MAUC this problem does not occurs.

The experiments were divided in four steps. The first one investigates the use of more than one neighbor of training instances. The second experiment compares the four versions of the proposed algorithm, as presented in Sect. III. The third experiment compares the best version of the proposed algorithm to classical algorithms, namely a Multilayer Perceptron with 100 hidden neurons (MLP100) and CART (Classification and Regression Tree) [19] classifier. The fourth experiment is the comparison among the proposed algorithm and *ad hoc* methods.

Table II shows the results of the proposed method with different neighborhood configuration. Values in **bold** are the highest achieved value for each data set, while values after ± symbol means the standard deviation. The values underlined are those statistically equivalent[4] to the highest value.

The increasing in the number of neighbors does not mean an increasing of performance. Also, we did not identified a configuration as definitively preferred. Nevertheless in four of the seven data sets, the best performances were achieved with a low number of neighbors and in two of these four the best performance were achieved with only one neighbor. As this is the simplest configuration, its result will be used in the second experiment.

Table III shows the results of three versions of VDBC. The second modification, i.e., VBDC + SMOTE did not have performance improvement. Meanwhile the third modification showed to be a very good complement to the algorithm.

The last modification to VDBC was motivated by the fact that it shows to be sensitive to overlapping. This fact can be easily noticed crossing values of Tables III and IV. The latter shows the generalized Fisher Ratio (FR) [15], [16]. This measure verify the degree of separation between classes. As higher the value more separated are the classes. Among the classes used in this work, those with higher degrees of separation are the same in which VDBC had its best performance. Thus, overcoming overlapping problem resulted in a better performance, as expected.

There is, however, an outlier. The FR value for *Thyroid* shows a data set with almost non separated classes. Nevertheless VDBC was able to better generalize its instances than instances of other data sets with more separated classes. This happens because the biggest class in *Thyroid* data set is huge in comparison with other classes. Thus, the biggest class part in the formula has an enormous influence in the final value. A similar case occurs with *Page-Blocks* data set, however in smaller proportions. Hence, the FR formula should

[3] Except *ad hoc* methods. Their results were extracted from the literature.

[4] Statistical tests were performed with $\alpha = 0.05$, t-student for parametric tests and Wilcoxon's ranksum for non-parametric tests.

TABLE IV
LEVEL OF OVERLAPPING FOR EACH DATASET

Dataset	Generalized Fisher Ratio
Gene	0.1349
Glass	0.7534
Horse	0.1700
Page-Blocks	0.2497
Satimage	1.4764
Thyroid	0.0352
Yeast	0.5980

be modified due to these cases, in which it shows classes less separated than they probably are.

Further, it is necessary to point out that overlapping degree is not the only factor that influences the performance of classifiers in imbalanced domains, e.g., complexity of classes [20], [21].

Table V shows the comparison among VDBC (with FR), MLP with 100 hidden neurons (MLP100) and CART classifiers. These classifiers were chosen after a previous study [22], which showed MLP100 and CART as very sensitive and moderately sensitive (respectively) to class imbalance.

The MLP classifier showed a different behavior than the obtained in [22]. It did not show to be very sensitive to class imbalance with the presence of multiple classes. This behavior might be related to the increasing of different class concepts. As some pairs of classes are naturally far from each other, the effects of imbalance is not "widespread". However the MLP was worse than VDBC in all datasets, even comparing to the original version of VDBC.

CART behaved as expected for a non tuned and moderately sensitive classifier. VDBC lost in only two of seven datasets, however it still had a good performance — near 90% of success. It is interesting to notice that CART only performs better when the number of classes is small. Although *Horse* dataset has also three classes it showed challenging for classical classifiers.

Table VI presents the comparisons with *ad hoc* methods. The values for AdaBoost.NC were those with random oversampling and λ parameter set to 9, as in the original paper [2]. From this paper we also drawn the MAUC values for SMOTEBoost.

As expected *ad hoc* methods were able to achieve better performances. However VDBC did not performed too worse. On the contrary it achieved, sometimes, better or equivalent results than some of the classifiers, although never better than all of them at the same time. These results show the potentiality of VDBC as it does not use more than one classifier in ensemble fashion, as its *ad hoc* competitors.

A graphical summary of comparisons is shown in Fig. 2. In this figure the performance of VDBC is shown in x-axis, while classical and *ad hoc* methods are shown in y-axis. Points below the diagonal (y = x) correspond to data sets where the proposed approach performs better than the compared methods. This figure confirms VDBC's potentiality as it shows clearly that the proposed classification model was able to reach ensemble *ad hoc* methods performances.

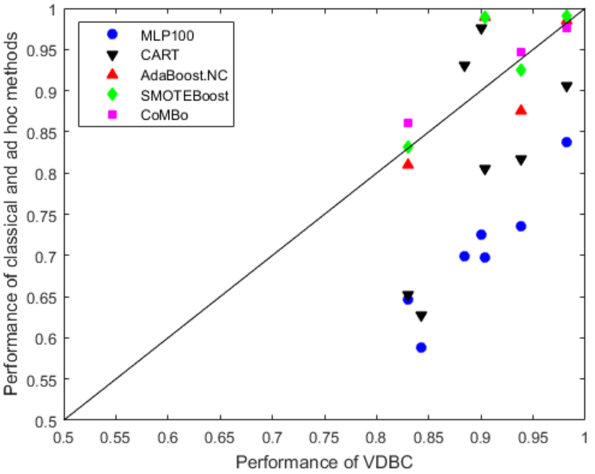

Fig. 2. Comparison of performance results among the Proposed approach (x-axis), the classical and *ad hoc* methods (y-axis).

To finish we wish to point that VDBC is feasible for its purpose. Observing its performance in comparison with *ad hoc* methods, which use ensemble, we can see that this simpler algorithm has a great potentiality. Nevertheless, new approaches are still necessary to be evaluated, including the use of ensembles.

V. CONCLUSION

In machine learning the study on imbalance problems is receiving an increased attention, as many real world data sets show at least a minimum degree of imbalance among their classes. This imbalance is responsible for several problems, mainly the misdirection of results.

In the literature there exist lots of works related to the imbalance binary case. However, as there are imbalanced data sets with more than two classes, this case is also receiving attention from researches. The most common approach to multiclass problem is the use of binarization techniques, also called decomposition techniques. However this case may be treated with classifiers that are able to handle more than two classes at once.

In this paper we presented a new approach to classify multiclass imbalanced data sets called VDBC. The classifier is based on the idea of Voronoi diagrams. With this classification model the sample space may be divided in several cells, in which each cell is assigned to a different class. The goal of this paper was then verify the feasibility of this new approach.

In the training phase we considered the construction of cells with at least one neighbor and at most five neighbors. Results showed that increasing the number of neighbors do not improves performance. The use of SMOTE to increase the size of the smallest classes also did not result in improvement of performance. However, creating synthetic training data between some classes resulted in better performance.

TABLE V
AVERAGE MAUC OF VDBC AND CLASSICAL CLASSIFIERS

Dataset	MLP100	CART	VDBC
Gene	0.6983±0.0179	**0.9304**±0.0081	0.8837±0.0107
Glass	0.7352±0.0511	0.8175±0.0396	**0.9381**±0.0291
Horse	0.5884±0.0379	0.6276±0.0356	**0.8429**±0.0393
Page-Blocks	0.6981±0.0585	0.8057±0.0338	**0.9039**±0.0211
Satimage	0.8375±0.0070	0.9066±0.0061	**0.9816**±0.0023
Thyroid	0.7245±0.0459	**0.9754**±0.0207	0.8999±0.0174
Yeast	0.6466±0.0547	0.6524±0.0235	**0.8297**±0.0376

TABLE VI
AVERAGE MAUC FOR THE PROPOSED CLASSIFICATION MODEL AND *ad hoc* CLASSIFIERS

Dataset	AdaBoost.NC	SMOTEBoost	CoMBo	VDBC
Glass	0.876±0.037	0.925±0.030	**0.947**±0.027	0.9381±0.0107
Page-Blocks	0.989±0.004	0.989±0.005	—	0.9039±0.0211
Satimage	0.984±0.002	**0.991**±0.001	0.976±0.003	0.9816±0.0023
Yeast	0.810±0.020	0.831±0.021	**0.861**±0.025	0.829±0.0376

Among classical classifiers (MLP100 and CART), VDBC showed its superiority. Compared with *ad hoc* methods, it was observed that our classification model has a great potentiality. However VDBC still needs more investigation.

Although VDBC did not achieved better performances than compared *ad hoc* classifers we must stress that its simplicity makes it already competitive. Probably few modifications or tuning shall result in a very efficient classifier for imbalance data sets.

ACKNOWLEDGMENT

The authors would like to thank CNPq and FACEPE (Brazilian research agencies) for the financial support.

REFERENCES

[1] A. Fernández, V. López, M. Galar, M. J. del Jesus, and F. Herrera, "Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches," *Knowledge-Based Systems*, vol. 42, pp. 97–110, apr 2013.

[2] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions." *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 42, no. 4, pp. 1119–1130, 2012.

[3] T. Hastie and R. Tibshirani, "Classification by pairwise coupling," *Ann. Statist.*, vol. 26, no. 2, pp. 451–471, 1998.

[4] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, "An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes," *Pattern Recogn.*, vol. 44, no. 8, pp. 1761–1776, Aug. 2011. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2011.01.017

[5] ——, "Dynamic classifier selection for one-vs-one strategy: Avoiding non-competent classifiers," *Pattern Recogn.*, vol. 46, no. 12, pp. 3412–3424, Dec. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2013.04.018

[6] F. Aurenhammer, "Voronoi diagrams a survey of a fundamental geometric data structure," *ACM Computing Surveys*, vol. 23, no. 3, 1991.

[7] A. Asuncion and D. Newman, "Uci machine learning repository," University of California, Irvine, School of Information and Computer Sciences, 2007. [Online]. Available: http://www.ics.uci.edu/∼mlearn/MLRepository.html

[8] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "Smoteboost: improving prediction of the minority class in boosting," in *In Proceedings of the Principles of Knowledge Discovery in Databases, PKDD-2003*, 2003, pp. 107–119.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *J. Artif. Int. Res.*, vol. 16, no. 1, pp. 321–357, Jun. 2002. [Online]. Available: http://dl.acm.org/citation.cfm?id=1622407.1622416

[10] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Machine Learning, Proceedings of the Thirteenth International Conference (ICML '96), Bari, Italy, July 3-6, 1996*, 1996, pp. 148–156.

[11] Y. Liu and X. Yao, "Simultaneous training of negatively correlated neural networks in an ensemble," *Trans. Sys. Man Cyber. Part B*, vol. 29, no. 6, pp. 716–725, Dec. 1999. [Online]. Available: http://dx.doi.org/10.1109/3477.809027

[12] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[13] S. Koço and C. Capponi, "On multi-class classification through the minimization of the confusion matrix norm," in *Asian Conference on Machine Learning, ACML 2013, Canberra, ACT, Australia, November 13-15, 2013*, 2013, pp. 277–292. [Online]. Available: http://jmlr.org/proceedings/papers/v29/Koco13.html

[14] I. Mukherjee and R. E. Schapire, "A theory of multiclass boosting," *CoRR*, vol. abs/1108.2989, 2011. [Online]. Available: http://arxiv.org/abs/1108.2989

[15] R. A. Mollineda, J. S. Snchez, and J. M. Sotoca, "Data characterization for effective prototype selection," in *Proc. of the 2nd Iberian Conf. on Pattern Recognition and Image Analysis*. Springer, 2005, pp. 27–34.

[16] S. Maldonado and C. Montecinos, "Robust classification of imbalanced data using one-class and two-class svm-based multiclassifiers," *Intell. Data Anal.*, vol. 18, no. 1, pp. 95–112, Jan. 2014. [Online]. Available: http://dx.doi.org/10.3233/IDA-130630

[17] D. J. Hand and R. J. Till, "A simple generalisation of the area under the roc curve for multiple class classification problems," *Mach. Learn.*, vol. 45, no. 2, pp. 171–186, Oct. 2001. [Online]. Available: http://dx.doi.org/10.1023/A:1010920819831

[18] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, "The balanced accuracy and its posterior distribution," in *Proceedings of the 2010 20th International Conference on Pattern Recognition*, ser. ICPR '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 3121–3124. [Online]. Available: http://dx.doi.org/10.1109/ICPR.2010.764

[19] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Wadsworth, 1984.

[20] G. H. Nguyen, A. Bouzerdoum, and S. L. Phung, "Learning pattern classification tasks with imbalanced data sets," *In P. Yin (Eds.), Pattern recognition*, pp. 193–208, 2009.

[21] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intell. Data Anal.*, vol. 6, no. 5, pp. 429–449, oct 2002.

[22] E. J. R. Silva and C. Zanchettin, "On the existence of a threshold in class imbalance problems," *IEEE International Conference on Systems, Man, and Cybernetics, 2015, Hong Kong*, pp. 2714–2719, 2015.