A New Oversampling-Based Approach for Class Imbalance Problem

Alexandre Wagner Chagas Faria
Graduate Program in Electrical Engineering
Federal University of Minas Gerais
Av. Antônio Carlos 6627, 31270-901,
Belo Horizonte, MG, Brazil
Email:awcfaria@eng-ele.dout.ufmg.br

Cristiano Leite de Castro
Computing Department
Federal University of Lavras
University Campus, 37200-000,
Lavras, MG, Brazil
Email: crislcastro@gmail.com

Antônio de Pádua Braga
Graduate Program in Electrical Engineering
Federal University of Minas Gerais
Av. Antônio Carlos 6627, 31270-901,
Belo Horizonte, MG, Brazil
Email:apbraga@ufmg.br

***Abstract*—In this paper, a new oversampling method is proposed to improve the representativeness of minority groups in the training data set. Our methodology creates artificial (synthetic) examples on basis the spatial distribution of the classes. The original data are expanded (duplicated) along the lines connecting the class centroid and each minority pattern under consideration. In contrast to other methods known in literature (as SMOTE), our geometric approach for data generation has the advantage of being accomplished in a straightforward way, i.e., without the need of the definition of parameters by the user. Experiments conducted with real and synthetic data point out that the our solution to the class imbalance problem is able to improve the number of correct minority classifications and the balance between the class accuracies.**

I. Introduction

An important challenge in the design of supervised models for pattern recognition is to obtain training sets that are both representative and balanced. In most cases, due to the inherent difficulty to acquire examples belonging to a particular group, it is very common to observe a high level of imbalance in the class distributions. Such a characteristic is prevalent in real applications, such as medical diagnosis, detection of frauds and faults, text categorization, among others. In this kind of problem, in which the training sets are usually complex and highly imbalanced, traditional learning algorithms have presented difficulty in distinguishing among the class examples. They tend to produce decision rules that favor the majority (negative) class.

This bias induced by the dominant class mainly occurs because the traditional algorithms consider different errors as equally important, assuming that class distributions are relatively balanced [11], [14], [12]. Furthermore, the degradation of classification performance in the presence of uneven data has also been attributed to other factors, such as the level of class overlapping and the lack of representativeness of the minority class [1], [2].

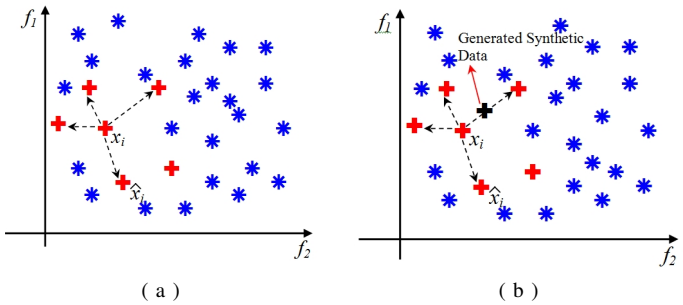
Two main approaches have been proposed to solve the class imbalance problem. In the first approach, existing learning algorithms are adapted to improve both the minority class recognition ability and the general accuracy of the classifier. Along this line of work, the main solutions that have been developed include those based on: (i) single class learning, such as the autoassociator [3] and *one-class SVMs* [4], [5]; (ii) *cost-sensitive Boosting* such as AdaCost [6] and, AdaC1, AdaC2 and AdaC3 [7], and (iii) modifications in the cost function optimized by the learning machine (algorithm) [8], [9], [10], [12].

In the second approach, called data preprocessing, the goal is to change (balance) the class distributions in the training set through mechanisms of data sampling in the input space, which includes undersampling of the majority class, oversampling of the minority class or a combination of both techniques [15], [16], [17]. The undersampling involves elimination of examples of the majority class. The examples to be eliminated can be chosen either randomly or from some a priori information.

The oversampling is based on the replication of preexisting examples or on the generation of synthetic data. In the first case, the selection of examples to be replicated can be made randomly or it can be guided. Regarding the generation of synthetic data, most the oversampling methods that have been proposed are based on the following interpolation technique: for each positive example $\mathbf{x}_i$ under consideration, a new synthetic example is generated along one of the line segments that connect $\mathbf{x}_i$ to its K nearest neighbors. This technique was originally proposed in the SMOTE method [18] and it is still used by recent methods, such as ADASYN [19] and RAMO [20], which are different from SMOTE in relation to the regions of the positive class that should be more intensely oversampled.

This paper investigates a new approach for the generation of synthetic data. Differently from the oversampling methods aforementioned, our solution does not depend on the choice of one parameter K (number of nearest neighbors), since it generates new examples from the spatial projection of the minority class. Experiments using *MultiLayer Perceptron networks* (MLPs) as well as *Support Vector Machines* (SVMs) were conducted to test the efficiency of our methodology on imbalanced data. The results are promising and show that the proposed method is able to improve the number of correct positive classifications and the balance between the class accuracies.

The reminder of this paper is organized as follow: Section 2 describes the SMOTE method and the foundations of the proposed oversampling algorithm. Section 3 presents the methodology used in the experiments as well as the results that were obtained from those experiments. Finally, the discussion and conclusion are provided in Section 4.



(a) (b)

Fig. 1. (a) Example of application of KNN for K=4. (b) New synthetic example.

II. Oversampling-based Approach

Before presenting the details of our approach, the following section discusses briefly the mechanism of generation of synthetic data that is used by the SMOTE method. The description of the SMOTE method aims to provide a better understanding of the foundations of our method as well as to emphasize the difference between both methods.

A. SMOTE

In the SMOTE algorithm [18], for a subset $S_{min} \in S$ (training set), the K-nearest neighbors (KNN) are considered for each one of the examples $x_i \in S_{min}$ (minority class); the KNN are defined as the K elements of S_{min} whose Euclidian distance between itself and x_i under consideration exhibits the smallest magnitude along the n-dimensions of feature (input) space X. To create a synthetic data, randomly select one of the KNN, then multiplied the corresponding feature vector difference with a random number between [0 - 1], and finally, add this vector to x_i

$$x_{new} = x_i + (\widehat{x}_i - x_i) \times \delta \tag{1}$$

where x_i is the minority example under consideration, $\widehat{x}_i$ is one of the KNN of x_i (randomly chosen) and δ is a random number between 0 and 1. Thus, the resulting synthetic example is some point along the segment joining x_i under consideration and the randomly selected $\widehat{x}_i$ [14].

Figure 1 illustrates the SMOTE's interpolation mechanism. Figure 1.a shows an imbalanced data set where the minority class is represented by "crosses" and the majority class by "stars". The number of KNN is 4. Figure 1.b shows the synthetic example created along the line segment that joins x_i and $\widehat{x}_i$.

One important limitation of the SMOTE algorithm is that it does not take into consideration the proximity of examples belonging to different classes during the process of synthetic data generation. This can result in increasing the level of class overlapping and variance [14], [12].

B. Our Proposal

In contrast with the well known SMOTE algorithm and its variants [18], [20], our oversampling-based approach is not dependent on the K-nearest neighbor (KNN) method. It creates artificial (synthetic) data considering the spatial distribution of the classes in the training set. Let S_{min} be the minority class drawn from the training set $S = \{S_{min}, S_2, \ldots, S_m\}$, where m is the total number of classes. An synthetic example is generated from the following steps: (i) obtain the centroid C of S_{min}; (ii) select an arbitrary example x_i from S_{min}; (iii) calculate the corresponding difference vector $(C - x_i)$; (iv) multiply this vector by a factor ε and, add it to x_i. This procedure can be summarized by Equation 2 and should be repeated for each positive example belonging to S_{min}.

$$x_{new} = x_i + (C - x_i) * \varepsilon \tag{2}$$

the parameter ϵ, in the Equation 2, represents a constant in the range $0 \leq \epsilon \leq 1$ (for the Proposal 1) or $1 \leq \epsilon \leq 2$ (for the Proposal 2) - The two proposals will be discussed below. The value of ϵ can be set according to the minority class distribution. The centroid coordinates (C_x, C_y) is given by the following expressions, Equations 3 and 4.

$$C_x = (\max\ x_-) + \left(\frac{\max\ x_+ - \max\ x_-}{2}\right) \tag{3}$$

$$C_y = (\max\ y_-) + \left(\frac{\max\ y_+ - \max\ y_-}{2}\right) \tag{4}$$

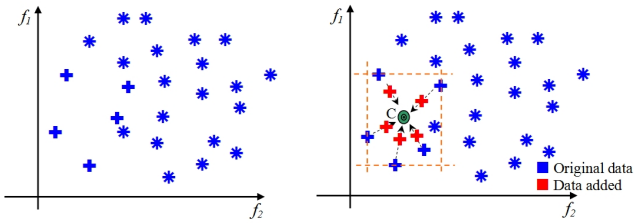
where $\max\ x_+$ and $\max\ x_-$ represent the x coordinates of the examples that are further along the x axis in both directions, $+$ and $-$; $\max\ y_+$ and $\max\ y_-$ represent the y coordinate of the examples that are further along the y axis in both directions $+$ and $-$.

According to Equation 2, the new example is a point along the line segment which passes by example x_i under consideration and the centroid C of the minority class. For the Proposal 1, the new point (example) should be located between the centroid and x_i, and for the Proposal 2 the new point should be on the same line, but beyond x_i. Thus, at the end of the oversampling process, each example of the minority class is duplicated towards the centroid or in the opposite direction, depending on which approach was chosen.

The Figures 2 (a) and (b) show an imbalanced data set and illustrate the application of our oversampling method. Firstly, we calculated the centroid of the square whose edges pass over the points that are furthest on the x and y axes. Then, each point of the minority class generates a new synthetic examples in two ways:

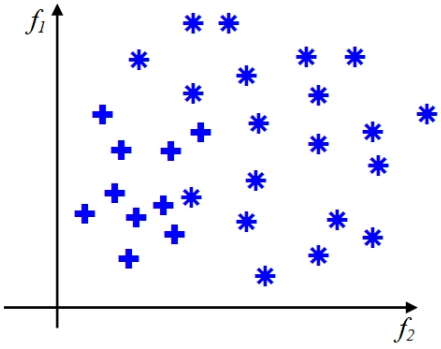
- Going towards the centroid. Figure 3 presents the new data with the minority class generated in the centroid direction. This approach was named as Proposal 1.
- After calculating the centroid, the generated data are expanded in the opposite direction of the centroid. This approach was named as Proposal 2.

We also evaluated the duplication of the original minority data in other directions: North West (NW), North East (NE), South West (SW) and South East (SE). All of them take the center as the centroid of the original minority class. In the experiments conducted, this last approach was tested only for the Thyroid database.



(a) (b)

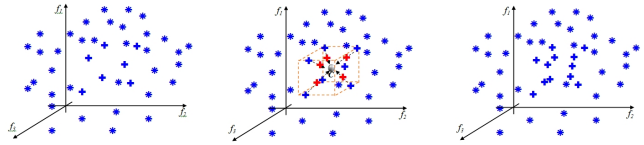
Fig. 2. (a) Original data, (b)Proposal methodology.



(a)

Fig. 3. (a) Final data.

An example of the application of our proposal on 3D data (3 features) can be seen in Figure 4. In this case, we created a cube whose edges pass through the maximum and minimum values in each axis (X,Y,Z).



(Original 3D Data) (Proposal) (Final 3D data)

Fig. 4. Illustration of our proposal on 3D data.

III. Experiments and Results

In this section we present the experiments accomplished to validate our method. For all experiments, was used two-fold cross validation [27]. Using traditional supervised learning algorithms as base classifiers, our oversampling proposals 1 and 2 were compared with SMOTE method. We also ran the learners with the original data sets (without any sampling strategy) to serve as a baseline for comparison.

In this paper, the experiments was conducted by two Machine Learning algorithms [29],[28]:SVM (Support Vector Machine) [21] and ANN (Artificial Neural Network) [27], both in MatLab implementations. Since we are dealing with non-linear data, the standard Matlab SVM algorithm was modified to use a polynomial kernel function of order 3. The ANN topology chosen was a MultiLayer Perceptron (MLP) trained with a Levenberg-Marquardt algorithm [22]. The MLP architecture used in all runs was 2:3:1. Hyperbolic tangent was chosen as activation function for all neurons (units). The training stop criterion was a MSE (mean squared error) smaller than 10^{-2} or 500 epochs.

In order to evaluate the algorithms, the experiments were performed over the following databases:

- A bidimensional synthetic data set generated from two Gaussian distributions with mean vectors (0,0) (minority class) e (1,1) (majority class) and covariance matrices equal to the identity matrix. The minority class contains only 8% of the majority class examples. The results are reported in Tables I and II.
- Thyroid dataset from the UCI Repository [23]. This dataset contain 1762 examples in the majority class and 238 examples in the minority class. The minority class represents 13.5% of the majority class (Figure 5). In this work we used only two features for this data set; such features were selected using the T-test procedure, as recommended in [24]. The results using this dataset are reported int Table III

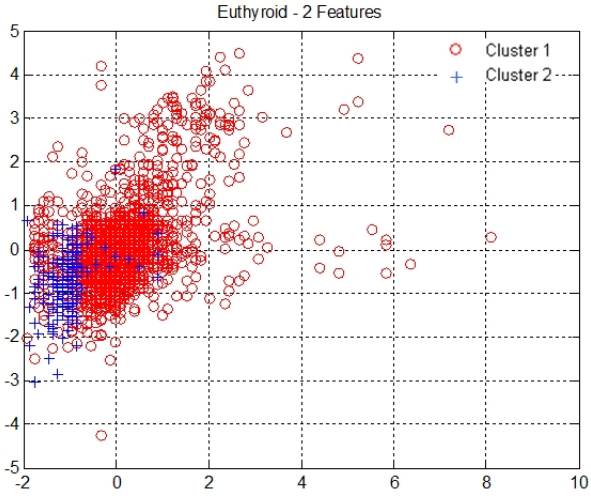


Fig. 5. Thyroid data base - using 2 features

The algorithms were evaluated using metrics extracted from the confusion matrix [25]. With this matrix, it is possible to analyze the relation between success and misclassification rates (Figure 6). Each element (k, j) of this matrix provides the number of the examples whose true (real) class was k and that were actually classified as j. Hence, the elements along the major diagonal represent the correct decisions made: number of true positives (TP) and true negatives (TN); and the elements off this diagonal represent the errors: number of false negatives (FN) and false positives (FP). From the confusion matrix, it is possible to determine some performance scores. A score commonly used in classification tasks is given in Equation 5

$$Accuracy = \frac{TP + TN}{Total} \tag{5}$$

where $Total$ corresponds to the total number of cases evaluated.

		CLASSIFICATION +	CLASSIFICATION -	
REAL	+	TP (true positive)	FN (false negative)	P
	-	FP (false positive)	TN (true negative)	N
		TP + FP	FN + TN	P + N

Fig. 6. Confusion matrix

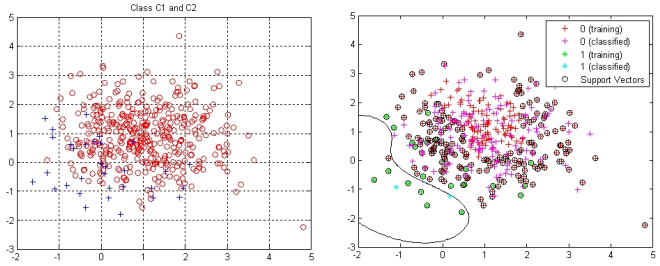
However, when the data set is highly imbalanced, the use of global measures, such as *Accuracy*, might lead to misleading conclusions. For instance, it is straightforward to create a classifier having 98% *Accuracy* if the data set has a majority class with 98% of the total number of examples, by simply classifying every new example as belonging to the majority class. An efficient way of evaluating a classifier in imbalanced scenarios is to consider metrics that disassociates the errors (or hits) that occurred in each class. This can be achieved using the following expressions, Equations 6 and 7.

$$AccuracyofPositiveClass = \frac{TP}{TP + FN} \tag{6}$$

$$AccuracyofNegativeClass = \frac{TN}{FP + TN} \tag{7}$$

A. Results

Figure 7 (b) shows the SVMs' decision boundary yielded from the imbalanced synthetic data set of Figure 7 (a), in which the majority and minority classes are represented by circles and crosses, respectively. It is worth noting that since the training set is highly imbalanced, the SVM boundary bends towards the positive class.

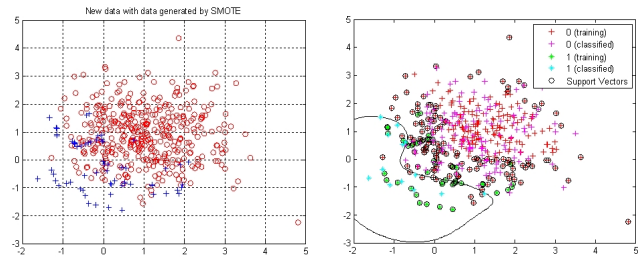


(a) (b)

Fig. 7. Test with original imbalanced data: a)Original imbalanced data b) SVMs' decision boundary

In order to provide a fair comparison with our oversampling-based approach, the SMOTE method was configured in order to duplicate the number of minority class examples (see Figure 8 a)). The decision boundary yielded by an SVM on the new data set generated with SMOTE is illustrated in Figure 8 (b).

Figure 9 (a) shows the data set after the application of our oversampling-based Proposal I. It is should be noted that



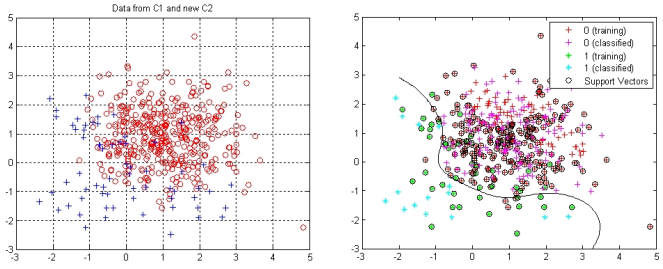
(a) (b)

Fig. 8. Test with SMOTE algorithm: a) Original minority class + synthetic data generated with SMOTE, b) SVMs' decision boundary yielded from SMOTE dataset.

the new examples were generated in the centroid direction of the minority class (crosses). The decision boundary yielded by an SVM is shown in the Figure 9 (b). Observe that by improving the representativeness of minority class, our approach leads to a decision surface which is more similar to a linear discriminant. This result points out that our approach can be used to diminish the bias in the presence of uneven data, improving the discrimination ability of the classifier.

Regarding the synthetic imbalanced data set, Tables I (using SVM) and II (using MLP) compares the results achieved with the following methods: original (without any sampling strategy), SMOTE, Proposal I and Proposal II. It should be noticed that our approach (specially the proposal 2) achieved a better accuracy for the positive class than SMOTE. One can state that the separation surfaces learned in the input space (by SVM and MLP) were set to maximize the number of correct positive classifications. A better balance between the class accuracy rates was also obtained.

The results obtained with Thyroid database (using SVM) are presented in Table III. For this database, other directions for the process of generation of synthetic data were evaluated: Proposal NE (North-East), Proposal SW (South-West), Proposal NW (North-West) and Proposal SE (South-East). In terms of the positive accuracy, the results suggest that our approach performs better than the original dataset. In comparison with the SMOTE algorithm, only Proposals II, SW and NW were better. Moreover, one can observe again a better balance between the positive and negative accuracies.



(a) (b)

Fig. 9. Test with our oversampling approach: a) Original minority class + synthetic data generate with Proposal 1; b) SVMs' decision boundary yielded from new data.

It is worth noting that all values provided in Tables I, II and III are based on the average of 10 different runs.

TABLE I. SYNTHETIC DATA SET - AVERAGE RESULTS ACHIEVED WITH SVM CLASSIFIER

Approach	Pos. Class Acc.	Neg. Class Acc.
Original	31.1%	97.6%
SMOTE	49.5%	94.6%
Proposal I	44.28%	95.7%
Proposal II	58.5%	96.6%

TABLE II. SYNTHETIC DATA SET - AVERAGE RESULTS ACHIEVED WITH MLP CLASSIFIER

Approach	Pos. Class Acc.	Neg. Class Acc.
Original	33.3%	98.7%
SMOTE	40.8%	96.6%
Proposal I	50.1%	96.6%
Proposal II	48.7%	97.5%

TABLE III. THYROID DATA SET - AVERAGE RESULTS ACHIEVED WITH SVM CLASSIFIER

Approach	Pos. Class Acc.	Neg. Class. Acc.
Original	16.28%	99.16%
SMOTE	20.43%	98.94%
Prop. I	20.09%	99.37%
Prop. II	24.74%	97.89%
Prop. NE	18.04%	98.52%
Prop. SW	22.07%	99.16 %
Prop. NW	20.48%	98.74%
Prop. SE	19.97%	98.74%

IV. CONCLUSION

In this paper, a new oversampling approach was presented in order to circumvent the problem of lack of representativeness such as sparse and imbalance of class distributions in training sets. It is important noting that although our method is based on the spatial locations of class centroids, it is not restricted to the unimodal assumption. The method assumes the existence of a class centroid which is representative of the overall class distribution even if there are multiple modes in the underlying distribution. Oversampling process is then accomplished along the line connecting the corresponding centroid and the input pattern to be sampled. According to the sparsity of the data and the relative position of the pattern, sampling can be accomplished towards the centroid or the separation margin between classes. This geometric notion of the class imbalance problem points out the need to understand data distribution in order to decide upon the method to be used. This approach, however, has shown to be efficient in all experiments accomplished in this paper.

Further works point out to the understanding of the geometric problem in higher dimensions and to the consideration of the relative positions of opposite class centroids in the sampling procedures.

REFERENCES

[1] Prati R., Batista G., Monard M.: Class Imbalances versus Class Overlapping: An Analysis of a Learning System Behavior. In: Advances in Artificial Intelligence, Third Mexican International Conference on Artificial Intelligence, series Lecture Notes in Computer Science, v. 2972, pgs 312–321. (2004)

[2] Khoshgoftaar, T.M., Hulse, J.V., Napolitano, A.: Supervised neural network modeling: An empirical investigation into learning from imbalanced data with labeling errors. IEEE Trans. on Neural Networks, 21, pp. 813–830. (2010)

[3] Japkowicz, N.: Supervised Versus Unsupervised Binary-Learning by Feedforward Neural Networks, Mach. Learn., v. 42, pgs. 97–122. (2001)

[4] Scholkopf, Bernhard and Platt, John C. and Shawe-Taylor, John C. and Smola, Alex J. and Williamson, Robert C., Estimating the Support of a High-Dimensional Distribution, Neural Comput., volume 13, number 7, pages 1443–1471, publisher MIT Press, address Cambridge, MA, USA, (2001)

[5] Raskutti, B., Kowalczyk, A.: Extreme re-balancing for SVMs: a case study, SIGKDD Explor. Newsl., v. 6, n. 1, pgs 60–69. (2004)

[6] Wei Fan and Salvatore J. Stolfo and Junxin Zhang and Philip K. Chan, AdaCost: misclassification cost-sensitive boosting, book Proceedings of IEEE International Conference on Machine Learning, pages 97-105, publisher Morgan Kaufmann (1999)

[7] Sun Y., Kamel M.S., Wong A.K.C., Wang Y.: Cost-sensitive boosting for classification of imbalanced data, Pattern Recognition, v. 40, n. 12, pgs. 3358–3378. (2007)

[8] K. Veropoulos and C. Campbell and N. Cristianini, Controlling the sensitivity of support vector machines, book Proceedings of the International Joint Conference on Artificial Intelligence, pages 55-60 (1999)

[9] Alejo, R. and García, V. and Sotoca, J.M. and Mollineda, R. A. and Sánchez, J.S., Improving the Performance of the RBF Neural Networks Trained with Imbalanced Samples, book Proc. of Intell. Data Eng. Autom. Learn., publisher Springer, series Lecture Notes in Computer Science, volume 7, number 4, pages 720-747, (2006)

[10] Hong S, Chen S., Harris C.J.: A kernel-based two-class classifier for imbalanced data sets, IEEE Transactions on Neural Networks, v. 18, n. 1, pgs 28–41. (2007)

[11] Monard, M.C. and Batista, G.E.A.P.A.,Learning with Skewed Class Distribution,bookAdvances in Logic, Artificial Intelligence and Robotics,publisherIOS Press,pages 173–180, (2002)

[12] C.L. Castro and A.P. Braga,Artificial Neural Networks Learning in ROC Space,bookProc. of the 1st International Conference on Neural Computation (ICNC'09),publisherINSTICC, pages219–224, (2009)

[13] Weiss,, Gary M., Mining with rarity: a unifying framework, SIGKDD Explor. Newsl., volume 6, number 1, issn 1931-0145, pages 7–19, publisher ACM, address New York, NY, USA, (2004)

[14] He, Haibo and Garcia, Edwardo A., book IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Knowledge and Data Engineering, number 9, pages 1263–1284, posted-at 2009-11-10 13:45:05, Learning from Imbalanced Data, volume 21, (2009)

[15] Laurikkala, Jorma, Improving Identification of Difficult Small Classes by Balancing Class Distribution, book AIME '01: Proceedings of the 8th Conference on AI in Medicine in Europe, pages 63–66, publisher Springer-Verlag, address London, UK, (2001)

[16] Estabrooks, Andrew and Jo, Taeho and Japkowicz, Nathalie, A Multiple Resampling Method for Learning from Imbalanced Data Sets, Computational Intelligence, number 1, pages 18–36, volume 20, (2004)

[17] Batista, G., Prati, R., Monard, M.: A study of the behavior of methods for balancing machine learning training data. SIGKDD Expl. Newsl., 6, pp. 20–29 (2004)

[18] Chawla, N.V., Bowyer, K.W. and Kegelmeyer, P.W.: SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, v. 16, pgs 321–357. (2002)

[19] Haibo He and Yang Bai and Edwardo A. Garcia and Shutao Li, ADASYN: Adaptive synthetic sampling approach for imbalanced learning, Proceedings of the International Joint Conference on Neural Networks, IJCNN 2008, pages 1322-1328, (2008)

[20] Chen, S., He, H., Garcia, E.A.: Ramoboost: ranked minority oversampling in boosting. IEEE Trans. on Neural Networks, 21, pp. 1624–1642 (2010)

[21] B. Scholkopf and A. J. Smola , title Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond , edition, publisher The MIT Press, year (2002)

[22] Hagan, M. T., Menhaj, M. B. Training feedforward networks with the Marquardt algorithm journal. IEEE Trans. on Neural Networks, v. 5, n. 6, pgs. 989–993. (1994)

[23] UCI machine learning repository, *http://archive.ics.uci.edu/ml/*

[24] Guyon I., Gunn S., Nikravesh M., Zadeh L.: Feature Extraction, Foundations and Applications, Springer. (2006)

[25] Provost F. , and Fawcett T.: Robust classification for imprecise environments,Mach. Learn., vol. 42, no. 3, pp. 203–231 (2001).

[26] Provost, Foster J. and Fawcett, Tom and Kohavi,, Ron, The Case against Accuracy Estimation for Comparing Induction Algorithms, ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning, isbn 1-55860-556-8, pages 445–453, publisher Morgan Kaufmann Publishers Inc., address San Francisco, CA, USA, (1998)

[27] New York, Haykin, S, Pulisher:Macmillan, Neural Networks: A Comprehensive Foundation, (1994)

[28] R.O. Duda and P.E. Hart and D.G. Stork, Pattern Classification, 2nd edition, Wiley-Interscience,(2000)

[29] T. Mitchell, Machine Learning, McGraw Hill, (1997)