



---

**Título artículo / Títol article:** A hybrid method to face class overlap and class imbalance on neural networks and multi-class scenarios

**Autores / Autors** Alejo, R. ; Valdovinos, R. M. ; García Jiménez, Vicente ; Pacheco-Sanchez, J. H.

**Revista:** Pattern Recognition Letters, 2013, Marzo, Vol. 34, nº 4

**Versión / Versió:** Pre-print

**Cita bibliográfica / Cita bibliogràfica (ISO 690):** ALEJO, Roberto, et al. A hybrid method to face class overlap and class imbalance on neural networks and multi-class scenarios. *Pattern Recognition Letters*, 2013, vol. 34, no 4, p. 380-388.

**url Repositori UJI:** <http://hdl.handle.net/10234/91710>

---

# A hybrid method to face with class overlap and class imbalance on multi-class scenarios

\*R. Alejo<sup>a</sup>, R.M. Valdovinos<sup>b</sup>, V. García<sup>c</sup>, J.H., Pacheco-Sanchez<sup>d</sup>

<sup>a</sup>*Tecnológico de Estudios Superiores de Jocotitlán  
Carretera Toluca-Atlacomulco KM. 44.8, Col. Ejido de San Juan y San Agustín, 50700 Jocotitlán  
(Mexico)*

<sup>b</sup>*Centro Universitario Valle de Chalco, Universidad Autónoma del Estado de México  
Hermenegildo Galena No.3, Col. Ma. Isabel, 56615 Valle de Chalco (Mexico)*

<sup>c</sup>*Institute of New Imaging Technologies, Universitat Jaume I  
Av. Sos Baynat s/n, 12071 Castelló de la Plana (Spain)*

<sup>d</sup>*Instituto Tecnológico de Toluca, Av. Tecnológico s/n Ex-Rancho La Virgen, 52140, Metepec,  
(Mexico)*

---

## Abstract

Class imbalance and class overlap are two of the major problems in data mining and machine learning. Several studies have shown that these data complexities may affect the performance or behavior of artificial neural networks. Strategies proposed to face with both challenges have been separately applied. In this work, we introduce a hybrid method for handling both class imbalance and class overlap simultaneously in multi-class learning problems. Experimental results on three remote sensing data show that the combined approach is a promising method.

*Keywords:* Multi-class imbalance, overlapping, back-propagation, cost function, editing techniques.

---

\*Corresponding author Tel.: +52(712)1231313; Fax:+52(712)1210113; E-mail address:ralejoll@hotmail.com.

---

1 **1. Introduction**

2 In supervised classification learning, the intrinsic difficulties in the data may  
3 significantly affect generalization performance of standard classifier algorithms.  
4 An important issue that has been identified into the 10 challenging problems is  
5 when the datasets suffer from skewed class distributions, that is, the number of  
6 samples of one class out numbers the other classes (class imbalance) [1]. Ex-  
7 isting research indicates that class imbalance problem causes seriously negative  
8 effects on the classification performance [2], since the classifier algorithms are of-  
9 ten biased towards the majority classes [3]. This phenomenon appears with high  
10 frequency in many real-world applications where it is often costly misclassified  
11 examples of the minority class. Typical examples are remote sensing [4], medical  
12 diagnosis [5], biological data analysis [6], fraud detection [7] and credit assess-  
13 ment [8].

14 Most of the research addressing the imbalance problem can be grouped into  
15 three categories: (i) Assigning distinct costs to the classification errors for pos-  
16 itive and negative samples [9, 2], (ii) Resampling the original training data set,  
17 either by over-sampling the minority class and/or under-sampling the majority  
18 class until the classes are approximately equally represented [10, 11, 12], and (iii)

19 Internally biasing the discrimination-based process so as to compensate for the  
20 class imbalance [13, 4, 14].

21 It is generally accepted that imbalance is the main responsible for a significant  
22 degradation of the performance on individual classes. However, recent works have  
23 pointed out that there does not exist a direct correlation between class imbalance  
24 and the loss of performance. These studies suggest that the class imbalance is not  
25 a problem by itself, but the degradation of performance is also related to other  
26 factors, i.e., the degree of class overlapping [15, 16, 17]

27 The class overlapping occurs in those zones where the decision boundary re-  
28 gions intersect. The overlapped samples have a high probability of being mis-  
29 classified for any classifier. Hence, several Instance Selection (IS) methods has  
30 been developed to address this challenging task [18]. The IS approaches that seek  
31 to remove points that are noisy or do not agree with their neighbours are called  
32 Edition algorithms. The most popular editing methods are based on the nearest  
33 neighbour rule.

34 Class overlap and class imbalance has been widely studied in the literature  
35 and treated separately. Rarely, however, both at the same time. There are also  
36 very few approaches facing with this complexities in multi-class scenarios. In this  
37 paper, we introduce a novel hybrid algorithm to face with class imbalance and

38 class overlapping simultaneously on multi-class problems.

39 This method is based on using a Gabriel graphs editing technique to remove  
40 noisy and border-line negative samples to reduce the class overlapping, and then  
41 a modified the back-propagation algorithm to face with imbalanced classes. Our  
42 main contributions in this paper are: *a*) to propose a new cost function (based in  
43 the mean square error) to deal with the class imbalance problem, *b*) we adapted  
44 the Gabriel graphs editing (GGE) to become it effective to reduce the class overlap  
45 in the neural network context and *c*) to combine the point *a* and *b* to generate an  
46 effective strategy dealing with the class overlap and class imbalance.

47 The rest of this paper is organized as follows. Related works are briefly re-  
48 viewed in Section 2. In Section 3 we introduce the modified back-propagation  
49 algorithm for tackling the class imbalance problem. The editing algorithm is de-  
50 scribed in Section 4. In section 5 we present a hybrid method dealing with the  
51 class overlap and class imbalance. Section 6 and 7 we show the experimental set  
52 up and results respectively. Finally, section 8 is the conclusion.

## 53 **2. Related Works**

54 Back-propagation is now the most widely used tool in the field of artificial  
55 neural networks (NN). However, despite the general success of back-propagation,

56 several major deficiencies are still needed to be solved. The major disadvantage  
57 of back-propagation is the slow rate of convergence of net output error; this is  
58 especially a major difficulty in “imbalanced” classification problems [19, 3], i.e.,  
59 where the training set contains many more samples of some “dominants” classes  
60 (majority classes) than the other “subordinates” classes (minority classes).

61 In the back-propagation algorithm, the class imbalance poses severe problems  
62 in training stage as the learning process becomes biased towards the majority  
63 classes, ignoring the minority classes and leaving them poorly trained at the end  
64 of the training stage. The learning process also becomes slower as it takes a longer  
65 time to converge to expected solution [19].

66 Many researches have been done in addressing the class imbalance problem  
67 [2]. In the NN field, the modified learning algorithm has been proposed for deal-  
68 ing with this problem. In reference [19] a modified back-propagation is proposed,  
69 this consists of calculating a direction in weight-space which decreases the error  
70 for each class (majority and minority class) in the same magnitude, in order to  
71 accelerate the learning rate for two-class imbalance problems. In the reference  
72 [4, 20, 3, 14], the error function was modified by introducing different costs asso-  
73 ciated with making errors in different classes. Basically, when the sum of square  
74 errors is calculated, each term is multiplied by a class dependent (regularization)

75 factor. This compensates class imbalance [4, 20, 14] and accelerates the conver-  
76 gence of the NN [3]. However, the main drawback of these approaches is the  
77 use of free parameters, because these parameters control the updating amount of  
78 weights whether training samples are in the minority or majority classes.

79 The most popular strategies to deal with the class imbalance problem have  
80 been at the data level. These methods for balancing the classes are the most inves-  
81 tigated because they are independent of the underlying classifier and can be easily  
82 implemented for any problem. The data level methods resampling the original  
83 dataset, either by over-sampling the minority class or by under-sampling the ma-  
84 jority class, until the classes are approximately equally represented. Both strate-  
85 gies can be applied in any learning system since they act as a preprocessing phase,  
86 thus allowing the system to receive the training instances as if they belonged to  
87 a well-balanced dataset. By using this strategy, any bias of the learning system  
88 towards the majority class due to the skewed class priors will hopefully be elimi-  
89 nated.

90 The simplest method to increase the size of the minority class corresponds  
91 to random over-sampling, that is, a non-heuristic method that balances the class  
92 distribution through the random replication of positive examples. Nevertheless,  
93 since this method replicates existing examples in the minority class, overfitting is

94 more likely to occur. Chawla et al.[10] proposed an over-sampling technique that  
95 generates new synthetic minority samples by interpolating between several pre-  
96 existing positive examples that lie close together. This method, called SMOTE  
97 (Synthetic Minority Over-sampling TEchnique), allows to the classifier to build  
98 larger decision regions that contain nearby samples from the minority class.

99       On the other hand, random under-sampling [21] aims at balancing the dataset  
100 through the random removal of negative examples. Despite its simplicity, it has  
101 empirically been shown to be one of the most effective resampling methods. Un-  
102 like the random approach, many other proposals are based on a more intelligent  
103 selection of the negative examples to be eliminated.

104       Several works point out class imbalance as an obstacle when applying machine  
105 learning algorithms to real world domains. However, in some cases, learning  
106 algorithms perform well on several imbalanced domains [22]. Recent work shows  
107 that class imbalance is not always a problem [17, 16]. Japkowicz and Stephen [21]  
108 suggest that some classifiers are not sensitive to the class imbalance problem in  
109 cases where the classes are separable. In the same way some researchers [20, 23]  
110 affirm that the class imbalance is not an intrinsic problem if the distributions do  
111 not overlap.

112       The overlapping appears when the samples of the minority class share a region



113 with the majority one, where all the samples are intertwined (this is an intrinsic  
114 problem of the data). García et al. [17] have shown that overlap can play an  
115 even larger role in determining classifier performance than the class imbalance  
116 problem. Lawrence et al. [20] suggest that when distribution is overlapped, it  
117 is desirable to pre-process or editing the data in a manner that results in reduced  
118 overlap. The similar idea was studied in [22]. That work shows that data clean-  
119 ing strategies usually lead to a performance improvement for highly overlapped  
120 datasets. Tang and Gao [24] use the inverse k-nearest neighbor and k-nearest  
121 neighbor (K-NN) algorithms to eliminate potential noisy patterns, and extraction  
122 of boundary pattern. The goal of that work is to deal with the classification prob-  
123 lem, which involves class overlapping. Nevertheless, the main drawback of these  
124 approaches is that parameter setting in k-NN impacts directly on the classification  
125 performance. Kretzschmar et al. [25] introduce variance-controlled NN (VC-  
126 NNs), which were developed to handle class overlap. These VCNNs are feed for-  
127 ward models trained by minimizing an error function involving the class-specific  
128 variance (CSV) computed at their outputs. This minimization suppresses abrupt  
129 changes in the responses of the trained classifiers in regions of the input space  
130 occupied by overlapping classes. The main restriction is that VCNNs require the  
131 selection of additional free parameter (to adjust of influence of CSV) specified

132 empirically by the user.

### 133 **3. A Modified Back-Propagation (MBP)**

134 The multilayer perceptron (MLP) neural network [26] usually comprises one  
135 input layer, one or more hidden layers, and one output layer. Input nodes corre-  
136 spond to features, hidden layers are used for computations, and output nodes are  
137 related with the number of classes. A neuron is the elemental unit of each layer.  
138 It computes the weighted sum of its inputs, adds a bias term and drives the re-  
139 sult through a generally non-linear (commonly a sigmoid) activation function to  
140 produce a single output.

141 The most popular training algorithm for MLP is the back-propagation algo-  
142 rithm, which uses a set of training instances for the learning process. Given a  
143 feed-forward network, the weights are initialized to small random numbers. Each  
144 training instance sent through the network and the output from each unit is com-  
145 puted. The target output is compared with the estimated output of the network by  
146 calculating the error, which is fed-back through the network.

147 To adjust the weights, the back-propagation algorithm uses a gradient descent  
148 to minimize the squared error. At each unit in the network starting from the output  
149 unit and moving to the hidden units, its error value is used to adjust the weights of

150 its connections as well as to reduce the error. This process is repeated for a fixed  
 151 number of times, or until the error is small.

152 On other hand, in the back-propagation algorithm the class imbalance problem  
 153 generates unequal contributions to the mean square error (MSE) in the training  
 154 phase [19]. Clearly the major contribution to the MSE is produced by the majority  
 155 class.

156 Let us consider a training dataset (TDS) with two classes ( $J = 2$ ) such that  
 157  $N = \sum_j n_j$  and  $n_j$  is the number of samples from class  $j$ . Suppose that the MSE  
 158 by class can be expressed as

$$E_j(U) = \frac{1}{N} \sum_{n=1}^{n_j} \sum_{p=1}^J (t_p^n - z_p^n)^2, \quad (1)$$

159 where  $t_p^n$  is the desired output and  $z_p^n$  is the actual output of the network for the  
 160 sample  $n$ . Then the overall MSE can be expressed as

$$E(U) = \sum_{j=1}^J E_j(U) = E_1(U) + E_2(U). \quad (2)$$

161 If  $n_1 \ll n_2$  then  $E_1(U) \ll E_2(U)$  and  $\|\nabla E_1(U)\| \ll \|\nabla E_2(U)\|$ , conse-  
 162 quently  $\nabla E(U) \approx \nabla E_2(U)$ . So,  $-\nabla E(U)$  it is not always the best direction to  
 163 minimize the MSE in both classes. [19].

164 Considering that the class imbalance problem affects negatively in the back-  
 165 propagation algorithm due to the disproportionate contributions in the MSE, it is

166 possible to consider a cost function ( $\gamma$ ) that balance the MSE as follows:

$$E(U) = \sum_{j=1}^J \gamma(j) E_j = \gamma(1) E_1(U) + \gamma(2) E_2(U) \quad (3)$$

$$= \frac{1}{N} \sum_{j=1}^J \gamma(j) \sum_{n=1}^{n_j} \sum_{p=1}^J (t_p^n - z_p^n)^2,$$

167 where  $\gamma(1) \|\nabla E_1(U)\| \approx \gamma(2) \|\nabla E_2(U)\|$  avoiding that the minority class be ig-

168 nored in the learning process. In this work, we propose a new cost function defined

169 as:

$$\gamma(j) = \frac{\|\nabla E_{max}(U)\|}{\|\nabla E_j(U)\|} \quad (4)$$

170 where  $\|\nabla E_{max}(U)\|$  corresponds to the largest majority class.

171 On the other hand, when a cost function is included in the training process, the

172 data probability distribution is altered [20]. Nevertheless, the cost function  $\gamma(j)$

173 (Eq. 4) reduces its impact in the data distribution probability because the cost

174 function value is diminished gradually. In this way, the class imbalance problem is

175 reduced in early iterations, and later  $\gamma(j)$  reduces its effect on the data distribution

176 probability.

#### 177 **4. Editing technique for handling class overlap**

178 The editing techniques have been proposed to remove noisy samples as well  
179 as close border cases (overlapping), leaving smoother decision boundaries [27].  
180 The aim is to improve the classifier accuracy. The most popular editing schemes  
181 are based on the well-know  $k$  Nearest Neighbour ( $k$ -NN) rule. However, this rule  
182 only takes into account the distances to a number of close neighbors. Alternative  
183 concepts of neighborhood have been proposed to consider geometrical relation  
184 between a sample and some of its neighbours [28].

185 The Gabriel Graph has recently been used for introducing a set of editing  
186 methods for the  $k$ -NN rule [29]. The Gabriel Graph Editing (GGE) consists of ap-  
187 plying the general idea of Wilson's algorithm [30], but using the graph neighbours  
188 of each sample instead of the Euclidean or other norm-based distance neighbour-  
189 hood. Two samples  $x$  and  $y$  are graph neighbours in a  $GG = (V, E)$  if there exists  
190 an edge  $(x, y) \in E$  between them. Taking into account the definitions of GG, the  
191 graph neighbourhood of a given point requires that no other point lies inside the  
192 union of the zones of influence (i.e. hypersphere of influence) corresponding to  
193 all its graph neighbours.

194 The application of GGE has some additional properties as compared to the  
195 conventional methods: first, they consider the number of neighbours as a variable

196 feature which depends upon every prototype. Secondly, since the graph neigh-  
197 bourhood of a sample always tends to be widely distributed around it, the infor-  
198 mation extracted from samples close to decision boundaries may be richer in the  
199 sense of the prototypes distribution [28].

200 The original GGE was proposed to improve the  $k$ -NN accuracy [29]. How-  
201 ever, in this work the original GGE was adapted to do it effective in the back-  
202 propagation context. The aim was to remove noisy and overlapping samples of  
203 the majority classes, but keeping all the positive samples. This task allows im-  
204 proving the back-propagation learning over the minority classes. The proposed  
205 GGE can be summarized as follows:

- 206 • For each sample  $p$ , constructs the corresponding GG.
- 207 • Consider  $p$  in the Training Dataset (TDS), if all its graph neighbours are of  
208 its same class.
- 209 • Other issue, if  $p$  belongs to some majority class, then discard  $p$  from TDS.

## 210 **5. Methodology for dealing with class imbalance and the class overlapping** 211 **on multi-class problems**

212 This section provides an overview of the method here proposed to deal with  
213 class imbalance and class overlapping simultaneously, which consists of combin-

214 ing an editing technique and a cost function. This strategy can be summarized as  
215 follows:

216 1. MBP: To deal with class imbalance problem.

217 (a) To modify the back-propagation (MBP) algorithm applying a cost func-  
218 tion (Eq. 4) in order to avoid that the minority classes would be ig-  
219 nored in the training process, and to accelerate the convergence of the  
220 neural network.

221 2. GGE: To deal with class overlapping problem.

222 (a) To edit the TDS with the GGE technique (sec. 4), removing only  
223 majority samples in the overlap region and producing a local balance  
224 of the classes.

225 3. MBP + GGE (Proposed strategy).

226 (a) To train the MLP with the modified back-propagation algorithm over  
227 the edited TDS.

## 228 **6. Experimental Protocol**

229 In this section we first provide details of the data sets chosen for the experi-  
230 mentation, the performance measures used to evaluate the classifiers and the re-

231 sampling methods. Finally, a briefly description of the configuration parameters  
232 of the methods.

### 233 *6.1. Database description*

234 We used in our experiments five remote sensing datasets: Cayo, Feltwell  
235 Satimage, Segment and 92AV3C. Feltwell is related to an agriculture region near  
236 to Felt Ville, Feltwell (UK) [31], Cayo represents a particular region in the gulf of  
237 Mexico, and Satimage consists of the multi-spectral values of pixels in 3x3 neigh-  
238 borhoods in a satellite image. Segment contains instances drawn randomly from  
239 a dataset of 7 outdoor images [32]. 92AV3C dataset<sup>2</sup> corresponds to a hyperspec-  
240 tral image (145x145 pixels) taken over Northwestern Indianas Indian Pines by the  
241 AVIRIS sensor.

242 In order to covert Cayo in a highly imbalanced dataset some of their classes  
243 were merged as follows: join the classes 1,3,6,7 and 10 for integrating the class  
244 1; join the classes 8, 9 and 11 for integrating the class 3, finally, the rest of classes  
245 (2,4 and 5) we obtain from original dataset. M92AV3C is a subset of 92AV3C,  
246 it contains six classes (2, 3, 4, 6,7 and 8) and 38 attributes. The attributes were

---

<sup>2</sup><https://engineering.purdue.edu/biehl/MultiSpec/hyperspectral.html>



247 selected using a common features selection algorithm (Best-First Search [33])  
248 implemented in WEKA<sup>3</sup>:

249 Feltwell, Satimage, Segment and 92AV3C were random under-sampling to  
250 generate severe class imbalance datasets. A brief summary of these multi-class  
251 imbalance datasets is shown in the Table 1. Note that are highly imbalanced  
252 datasets. For each database, a 10-fold cross-validation was applied. The datasets  
253 were divided into ten equal parts, using nine folds as training set and the remaining  
254 block as test set.

Table 1: A brief summary of some basic characteristics of the datasets. The bold numbers represent the samples of minority classes.

dataset	Size	Attr.	Class	Class distribution
MCayo	6019	4	5	2941/ <b>293</b> /2283/ <b>322</b> / <b>133</b>
MFelt	10944	15	5	3531/2441/ <b>91</b> /2295/ <b>178</b>
MSat	6430	36	6	1508/1531/ <b>104</b> /1356/ <b>93</b> / <b>101</b>
MSeg	1470	19	7	330/ <b>50</b> /330/330/ <b>50</b> / <b>50</b> /330
M92AV3C	5062	38	6	<b>190</b> / <b>117</b> /1434/2468/747/ <b>106</b>

<sup>3</sup>Available in: <http://www.cs.waikato.ac.nz/ml/weka/>

255 *6.2. Classifier performance and Significance Statistical Test*

256 The most traditional metric for measuring the performance of learning systems  
257 is the accuracy which can be defined as the degree of fit (matching) between the  
258 predictions and the true classes of data. However, the use of plain accuracy to eval-  
259 uate the classifiers in imbalanced domains might produce misleading conclusions,  
260 since it is strongly biased to favour the majority classes [34, 14]. Shortcomings of  
261 this evaluator has motivated search for new measures. One the most widely-used  
262 techniques for the evaluation of binary classifiers in imbalanced domains is the  
263 Receiver Operating Characteristic curve (ROC), which is a tool for visualizing,  
264 organizing and selecting classifiers based on their trade-offs between true positive  
265 rates and false positive rates. Furthermore, a quantitative representation of a ROC  
266 curve is the area under it, which is known as AUC [35]. The AUC measure for  
267 multi-class problems can be defined as:

$$AUC = \frac{2}{\|J\|(\|J\| - 1)} \sum_{j_i, j_k \in J} AUC_R(j_i, j_k) \quad (5)$$

268 where  $AUC_R(j_i, j_k)$  is the AUC for each pair of classes  $j_i$  and  $j_k$ .

269 Kubat and Matwin [36] use the the geometric mean of accuracies measured  
270 separately on each class. For multi-class problems it can be computed as:

$$g - mean = \left( \prod_{i=1}^J acc_i \right)^{\frac{1}{J}}, \quad (6)$$

271 where  $acc_i$  is the accuracy on the class  $i$  and  $J$  the number of classes.

272 Statistical tests are used to evaluate whether the performance of a new method  
 273 or learning algorithm on the same problem is significantly different. Into the  
 274 framework of empirical analysis, the Student’s paired t-test is the most widely  
 275 used parametric statistical procedure. However, it is well-known that it is con-  
 276 ceptually inappropriate and statistically unsafe to require certain assumptions like  
 277 the data is normally distributed [37]. In this work, we adopt the non-parametric  
 278 statistical Friedman test to perform a multiple comparison, which is equivalent of  
 279 the repeated-measures ANOVA. This test used to check if all methods perform  
 280 equal on the selected datasets can be rejected. The first step in calculating the test  
 281 statistic is to rank the algorithms for each dataset separately; the best performing  
 282 algorithm should have the rank of 1, the second best rank 2, etc. The Friedman  
 283 test uses the average rankings to calculate the Friedman statistic, which can be  
 284 computed as,

$$\chi_F^2 = \frac{12N}{K(K+1)} \left( \sum_j R_j^2 - \frac{K(K+1)^2}{4} \right) \quad (7)$$

285 where  $K$  denotes the number of methods,  $N$  the number of data sets, and  $R_j$  the

286 average rank of method  $j$  on all datasets. Iman and Davenport [38] showed that  $\chi_F^2$   
 287 presents a conservative behaviour, so they proposed a better statistic distributed  
 288 according to the  $F$ -distribution with  $K - 1$  and  $(K - 1)(N - 1)$  degrees of  
 289 freedom,

$$F_F = \frac{(N - 1)\chi_F^2}{N(K - 1) - \chi_F^2} \quad (8)$$

290 When the null-hypothesis is rejected, we can use post-hoc tests in order to  
 291 find the particular pairwise comparisons that produce statistical significant dif-  
 292 ferences. The Bonferroni-Dunn post-hoc test is applied to report any significant  
 293 difference between individual methods here used. The test uses the average rank  
 294 of each method and compare it to each other if these differ by at least the critical  
 295 difference, which is given by

$$CD = q_\alpha \sqrt{\frac{K(K + 1)}{6N}} \quad (9)$$

296 where the value  $q_\alpha$  is based on the studentized range statistic divided by  $\sqrt{2}$ .

### 297 6.3. Resampling Methods

298 SMOTE, and random under sampling (RUS) are used in the empirical study,  
 299 because are a popular approaches to deal with the class imbalance problem. How-

300 ever, it methods have internal parameters that enable the user to set up the resulting  
301 class distribution obtained after the application of these methods. In this paper,  
302 we decided to add or remove examples until a balanced distribution was reached.  
303 This decision was motivated for two reasons: a) by simplicity (to avoid use many  
304 free parameters) and b) by effectiveness. Results obtained with the other classi-  
305 fiers [39], have shown that when AUC is used as a performance measure, the best  
306 class distribution for learning tends to be near the balanced class distribution.

#### 307 *6.4. Neural network configuration*

308 The MLP was trained with the standard back-propagation (SBP) and modi-  
309 fied back-propagation (MBP) algorithm in batch mode. For each TDS, MLP was  
310 initialized ten times with different weights. The results here included correspond  
311 to the average of those achieved in the ten different initialization and of ten par-  
312 titions. The learning rate ( $\eta$ ) was set to 0.1 and only one hidden layer was used.  
313 The stop criterion was established at 25000 epoch or an MSE below to 0.001.  
314 The number of neurons for the hidden layer was obtained from the trial and error  
315 strategy. So, the number of neurons was 7, 6, 12, 10, 10 for MCayo, MFelt, and  
316 MSat, MSement and M datasets respectively.

## 317 **7. Results and discussion**

318 In order to assess the performance of the proposed method, we have carried out  
319 an experimental comparison with respect to well-known resampling approaches.  
320 In total, seven strategies were examined: (i) Standard Back-Propagation Algo-  
321 rithm (SBP), (ii) Modified Back-Propagation Algorithm (MBP), (iii) Standard  
322 Back-Propagation with Grabel Graph Editing (SBP+GGE), (iv) Modified Back-  
323 Propagation with Grabel Graph Editing (MBP+GGE), (v) SMOTE, (vi) SMOTE  
324 with Grabel Graph Editing (SMOTE+GGE) and (vii) Random Under Sampling  
325 (RUS). The datasets that were preprocessed by the SMOTE, SMOTE+GGE and  
326 RUS strategies were applied to the SBP algorithm.

327 In this paper, we have omitted other neural networks approaches as the two-  
328 phase technique [4], threshold moving [2], or modified error function [14], be-  
329 cause these methods contain many prior free parameters, so it is difficult to make  
330 a fair comparison.

331 With the aim of show the effectiveness of combining the MBP and the GGE  
332 techniques, in Fig. 1, the performance by class of the SBP, the MBP, the SBP+GGE  
333 and the proposed strategy (MBP+GGE), are presented separably (the bold boxes  
334 belong to minority classes). The results show that the minority classes perfor-  
335 mance is severally affected by the class imbalance. In Fig. 1a, 1e, 1i, and 1q

336 are observed that the class imbalance problem cause that some minority classes  
337 are not enough learned. So these minority classes show 0% of accuracy. The ef-  
338 fects the class imbalance problem is slow down the convergence of the SBP due  
339 to disproportionate contribution in the MSE in the training phase (see section 3).  
340 An immediate consequence of this, is the difficulty of achieving effective perfor-  
341 mance (in terms of classification) in a “reasonable” time. Especially in situations  
342 where there is an extreme class imbalance.

343 On other hand, the Fig. 1 shows that when the class imbalance is compensated  
344 (MBP) the minority classes performance is improve (Fig. 1b, 1f, 1j, 1n, and 1r).  
345 However, in hight overlapped TDS is not enough (Fig. 1j and 1r).

346 GGE technique is used to reduce the overlapping between classes. Fig. 1c,  
347 1g, 1k, 1o, and 1s, present the results obtained to apply the GGE technique. Note  
348 that it archive improve the minority classes performance, specially in overlapped  
349 TDS (see class 5 in Fig. 1k and 1o). Nevertheless, the class imbalance problem  
350 continues to affect. For example, observe Mfelt, and M92AV3C datasets (Fig.1g  
351 and 1s respectively). A negative consequence of GGE technique is that when  
352 increase the minority classes accuracy, the majority classes performs is affected.

353 The four column of the Fig. 1 presents the combining the MBP and GGE  
354 (MBP+GGE). These results show a remarkable improvement in minority classes

355 performance and exhibit a better performance than to apply individually the MBP  
356 and GGE techniques.

357 The modification of the training algorithm including a cost function (MBP)  
358 increases the recognition rate of less represented classes, accelerating the conver-  
359 gence of the network, and to apply GGE technique reduce the confusion of the  
360 minority classes in the overlap region. So the results presented in Fig. 1d, Fig. 1h,  
361 Fig. 1l, Fig. 1p and Fig. 1t, demonstrate the effectiveness of combining the MBP  
362 and GGE techniques.

363 Fig. 2 shows experimental results of compare the proposed method with re-  
364 spect to others well-known resampling approaches. The experimental results are  
365 presented in graphics where boxes represent the accuracy by class, and the bold  
366 boxes belong to minority classes. Fig. 2 exhibits that, the worst accuracy for the  
367 minority classes is shown by the RUS strategy (mainly over MFelt and MSat, see  
368 Fig. 2h and 2l). This means that when TDS is severely imbalanced removes sam-  
369 ples to balance the class distribution, and it is not effective on back-propagation,  
370 because the RUS involves a loss of useful information that could be important for  
371 the training process. In the M92AV3C and MSeg datasets, the RUS shows a good  
372 minority classes performance, however, is not a tendency.

373 SMOTE was very successful in MCayo and MFelt, but in MSeg, Sat, and



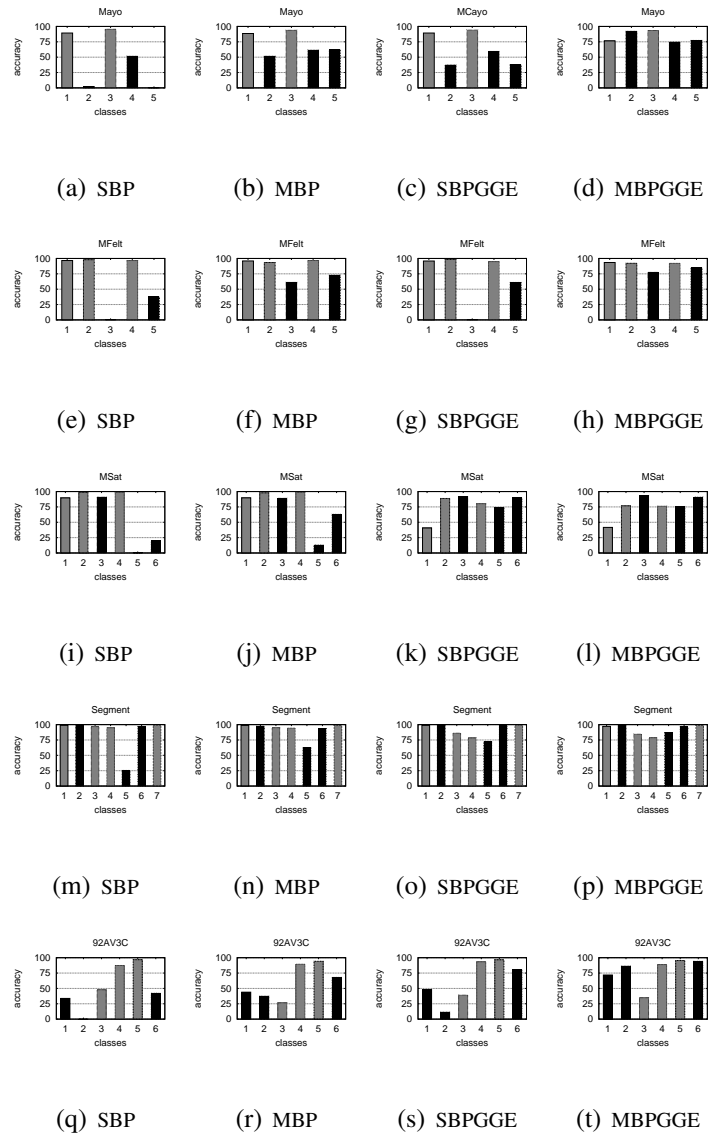


Figure 1: The comparison of methods deal with class imbalance problem and class overlap. The graphics shows accuracy by class. The bold boxes belong to minority classes. The acronyms SMO, it mean SMOTE.

374 M92AV3C datasets the minority classes performance is worst than the proposed  
375 method (see class 5 in Fig.2j and 2n, and class 1 in Fig. 2r). We believe that the  
376 explanation is that these datasets present high level of overlapping. For example  
377 MSat dataset shows high level of overlapping between the C-01 and C-05 classes,  
378 in other words, it is not enough to balance the TDS for improving the classifier  
379 performance over minority classes when the TDS overlaps. This is the reason of  
380 the low accuracy in class C-05 for RUS, MBP and SMOTE.

381 On other hand, MBP+GGE presents better results than the SMOTE for over-  
382 lapping datasets (see MSeg MSat, and M92AV3C, Fig. 2 *j, n* and *r*), this is due  
383 to the data cleaning method (GGE) is more efficient in highly overlapped regions.  
384 The MBP+GGE method starts to be less effective as overlapping is reduced (for  
385 example see MCayo and MFelt in Fig. 2).

386 The accuracy showed by SMOTE+GGE was very similar at SMOTE, how-  
387 ever, despite of that SMOTE+GGE include GGE technique this method was in-  
388 effective on overlapped datasets (see MSat, Fig. 2 *l*). The explanation is that,  
389 as SMOTE was firstly applied the overlap level was increased too, thus GGE  
390 was not able to remove the enough overlap for improving the accuracy of minor-  
391 ity class. To prove this, we repeat the experiment: we first applied GGE over  
392 MSat, and then MSat was over-sampled using SMOTE. The results obtained were

393 very successful and similar at achieved by MBP+GGE. The AUC= 0.756(0.050),  
394  $g$ -mean= 0.713(0.071), C-05 accuracy = 0.91(0.02). This results show the ef-  
395 fective of the GGE technique to reduce the class overlap and for improving the  
396 accuracy of the classifier over minority classes.

397 SMOTE and SMOTE+GGE strategies have made great improvement on the  
398 minority classes. However, they add information to the TDS by introducing new  
399 (non-replicate) minority classes samples, which involves a larger TDS and longer  
400 training times for the same number of training epochs. In addition, when the  
401 dataset present high overlapping the SMOTE can be not good choice, because can  
402 be increase the class overlapping. Meanwhile SMOTE+GGE is recommendable  
403 to apply first GGE and after the SMOTE, i.e., GGE+SMOTE.

404 Fig. 2 shows that the results obtained by MBP+GGE are very competitive  
405 with the results obtained by SMOTE and SMOTE+GGE. As well as MBP+GGE  
406 does not have internal parameters that the user needs to set up before to apply it  
407 and use of a TDS sight more reduced (much less training time). These are main  
408 advantages of MBP+GGE over SMOTE and SMOTE+GGE.

409 Table 2 summarize the experimental results in terms of AUC and  $g$ -mean on  
410 the five datasets when using six different strategies previously enumerated. For  
411 each method, the average ranking is shown. As can be seen in the table, the orig-

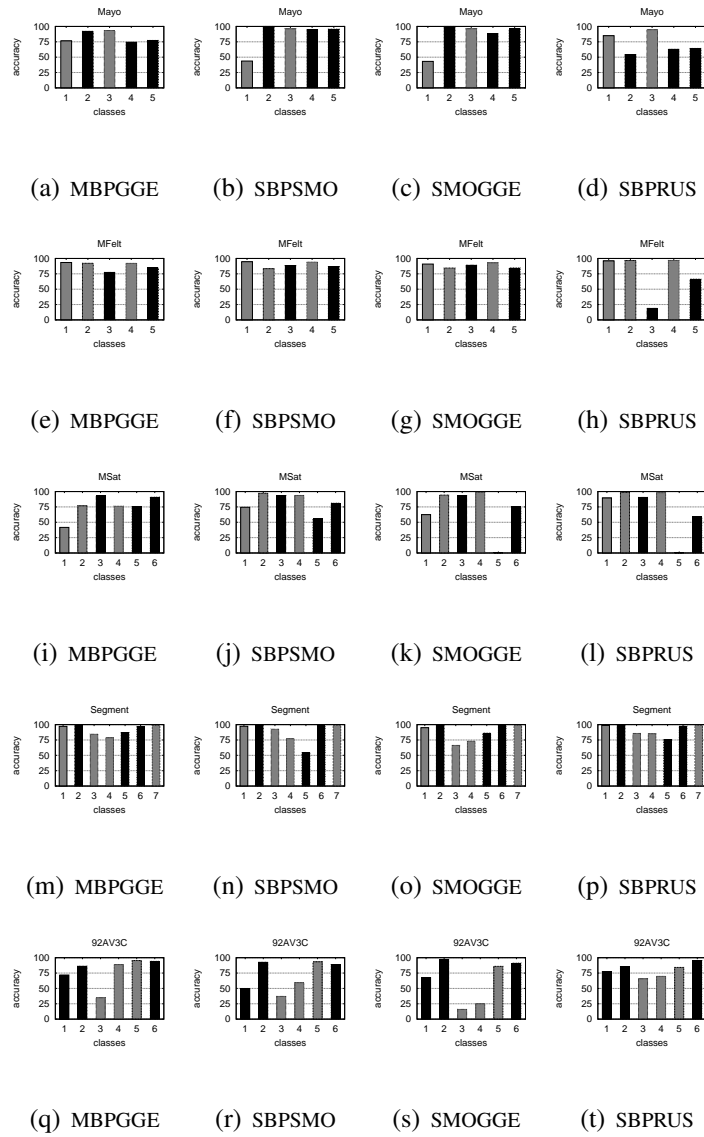


Figure 2: The comparison of methods deal with class imbalance problem and class overlap. The graphics shows accuracy by class. The bold boxes belong to minority classes. The acronyms SMO, it mean SMOTE.

412 inal (imbalanced) training set has the highest Friedman score (AR), which means  
 413 that this strategy performs worse than other methods, whereas MBP+GGE is the  
 414 best performing algorithm for AUC an  $g$ -mean. Note, that SMOTE performs  
 415 equal to MBP+GGE when the results are evaluated with AUC.

Table 2: Performance on three datasets measured using AUC,  $g$ -mean and average rank (AR)

AUC							
Dataset	Imbalanced <sup>1</sup>	MBP	GGE	MBP+GGE	SMOTE <sup>1</sup>	SMOTE+GGE <sup>1</sup>	RUS <sup>1</sup>
MCayo	0.477 (0.020)	0.715 (0.034)	0.636(0.064)	0.828(0.040)	0.860 (0.040)	0.847 (0.024)	0.722 (0.035)
MFelt	0.658 (0.022)	0.839 (0.033)	0.700(0.017)	0.880(0.031)	0.895 (0.046)	0.884 (0.027)	0.749 (0.028)
MSat	0.663 (0.026)	0.752 (0.044)	0.774(0.049)	0.757 (0.041)	0.826 (0.038)	0.705 (0.038)	0.726 (0.031)
MSeg	0.871(0.032)	0.916(0.098)	0.905(0.030)	0.918(0.095)	0.880(0.053)	0.882(0.031)	0.914(0.027)
M92AV3C	0.512(0.061)	0.589(0.106)	0.615(0.039)	0.780(0.086)	0.690(0.136)	0.638(0.079)	0.796(0.054)
AR	7.0	4.2	4.6	2.4	2.4	3.8	3.6
$g$ -mean							
Dataset	Imbalanced <sup>1</sup>	MBP	GEE	MBP+GGE	SMOTE <sup>1</sup>	SMOTE+GGE <sup>1</sup>	RUS <sup>1</sup>
MCayo	0.00 (0.00)	69.18 (4.18)	48.38(24.67)	81.99 (4.18)	82.24 (2.48)	80.63 (2.86)	70.22 (4.10)
MFelt	0.00 (0.00)	82.29 (4.10)	0.00(0.00)	87.54 (3.42)	89.05 (5.30)	88.14 (2.88)	53.05 (27.77)
MSat	0.00 (0.00)	49.36 (28.5)	73.89(6.71)	72.27 (5.38)	80.12 (5.28)	0.000 (0.00)	0.00 (0.00)
MSeg	66.60(31.81)	90.09(9.91)	89.21(3.82)	91.29(9.46)	78.83(24.62)	85.57(9.53)	90.37(3.46)
M92AV3C	0.00(0.00)	49.40(12.73)	32.17(26.39)	73.33(8.51)	54.79(26.28)	41.40(23.12)	77.77(6.39)
AR	6.7	4.0	4.9	2.2	2.4	4.2	3.6

<sup>1</sup> Classification using SBP

416 The Iman and Davenport statistic computed using Equation 8 yields  $F_F = 4.43$   
417 and  $F_F = 4.06$ , for AUC and  $g$ -mean respectively. The critical value of the  $F$ -  
418 Distribution with 6 and 24 degrees of freedom for  $\alpha = 0.05$  is 2.51. Given that  
419 the Iman and Davenport statistics are clearly greater than their associated critical  
420 value, the null-hypothesis that all methods perform equally can be rejected with  
421 a level of significance  $\alpha = 0.05$ . Then a post-hoc statistical analysis was used  
422 to detect significant differences for the control algorithm (method with the lowest  
423 ranking) in each measure.

424 Fig. 3 display a graphical representation of the results of Bonferroni-Dunn's  
425 post-hoc test, where for each method on the  $y$ -axis (ordered in ascending rank),  
426 the AR is plotted on the  $x$ -axis. For each AR we sum the critical difference ob-  
427 tained by the Bonferroni method,  $CD = 3.60$  for  $\alpha = 0.05$  in the two measures  
428 considered. The vertical dashed line segment represents the end of the best per-  
429 forming algorithm and the start of the next significantly method. MBP+GGE is  
430 the best algorithm, although according to Bonferroni-Dunn's test, only the differ-  
431 ence to the Imbalanced approach is different<sup>4</sup>.

432 The effects of the MBP+GGE can be better analysed by considering the num-

---

<sup>4</sup>Other powerful tests, such as Holm and Hochbergs ones would be necessary, for comparing the control algorithm with the rest of algorithms.

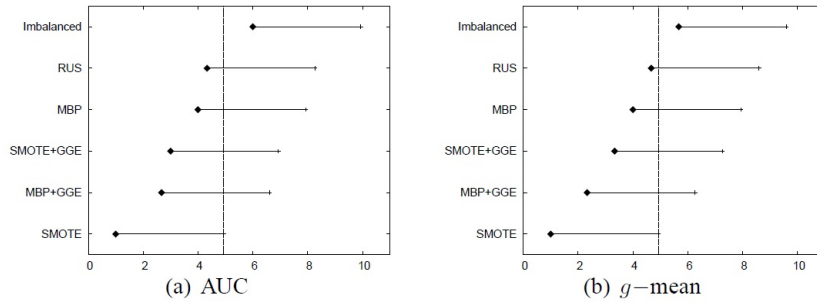


Figure 3: Bonferroni-Dunn's Critical Difference Diagram for AUC and  $g$ -mean

433 ber of samples that remain in the TDS after its application. Results in Figure 4  
 434 suggest a higher decrease in the size of the dataset when it is processed with the  
 435 GGE, whereas using SMOTE increase twice of the original size. RUS reduce  
 436 more the TDS size, however, not always present a good classifier performance.  
 437 Reducing the dataset involve to reach a better neural network learning time and  
 438 reduce storage requirements.

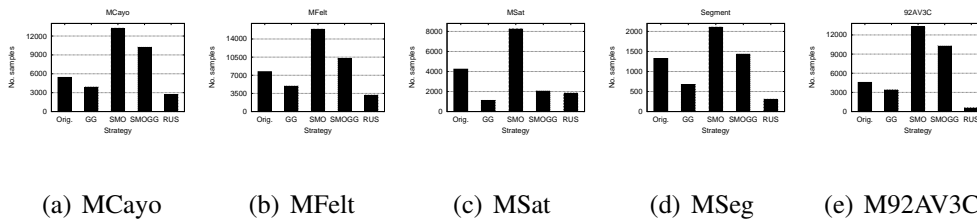


Figure 4: Training size after resampling TDS with the techniques GGE, SMOTE, SMOTE+GGE and RUS. The acronyms Orig., GG and SMO, they mean Imbalance TDS, GGE and SMOTE respectively.

## 439 **8. Conclusions**

440 In this work, we propose an hybrid method (MBP+GGE) for dealing with class  
441 imbalance and the class overlapping on multi-class problems. The MBP+ GGE is  
442 based on combination of modified back-propagation (MBP) with a Gabriel graph  
443 editing technique (GGE). For modified back-propagation algorithm we proposed  
444 to include a new cost function (based on MSE) in the algorithm, and for doing  
445 effective the Gabriel graph editing we adapted it in the back-propagation context.  
446 MBP+GGE generates two effects: a) MBP: to compensate the class imbalance  
447 during the training process and b) GGE: to reduce the confusion of the minority  
448 classes in the overlap region. With the edition of the majority classes it is possible  
449 to reduce the confusion between the minority and majority classes.

450 The MBP+GGE strategy was compared with the conventional class imbalance  
451 techniques: RUS, SMOTE, MBP, GGE and SMOTE+GGE. Results show  
452 that SMOTE and SMOTE+GGE are very effective even with highly imbalanced  
453 datasets, but inadequate on overlapped datasets. MBP+GGE show a better performance  
454 on class overlap problems. The data cleaning step used in the MBP+GGE  
455 seems to be specially suitable in situations having a high degree of overlapping,  
456 moreover, GGE produces a small training dataset.

457 The SMOTE is needed to find the most appropriate re-sampling rate, i.e., to



458 determine the number of samples when we introduce them in the minority classes  
459 before applying it. So the main advantages of MBP+GGE over SMOTE and  
460 SMOTE+GGE are: a) does not have internal parameters that the user needs to set  
461 up before applying it and b) use of a TDS sight more reduced (much less training  
462 time). As we see from the results, MBP+GGE is a very competitive strategy for  
463 dealing with class imbalance and the class overlapping on multi-class problems.

464 Further research is required to investigate the potential of the strategy pro-  
465 posed in this paper in “severe” multi-class imbalance and highly class overlap-  
466 ping problems. So, the exploration of the other editing strategies is necessary  
467 when approaching the graph based on editing scheme. Also, the study of new cost  
468 functions which help to speed up the neural network convergence in order to avoid  
469 altering the data probability distribution.

## 470 **References**

471 [1] Q. Yang, X. Wu, 10 challenging problems in data mining research, Inter-  
472 national Journal of Information Technology and Decision Making 5 (4)  
473 (2006) 597–604.

474 [2] Z.-H. Zhou, X.-Y. Liu, Training cost-sensitive neural networks with methods

- 475           addressing the class imbalance problem., *IEEE Transactions on Knowl-*  
476           edge and Data Engineering. 18 (2006) 63–77.
- 477 [3] S. Ramanan, T. Clarkson, J. Taylor, Adaptive algorithm for training pram  
478           neural networks on unbalanced data sets, *Electronics Letters* 34 (13)  
479           (1998) 1335–1336.
- 480 [4] L. Bruzzone, S. Serpico, Classification of imbalanced remote-sensing data  
481           by neural networks., *Pattern Recognition Letters* 18 (1997) 1323–1328.
- 482 [5] S. C. Mohammed Khalilia, M. Popescu, Predicting disease risks from highly  
483           imbalanced data using random forest, *BMC Medical Informatics and De-*  
484           cision Making 11 (2011) 1–13.
- 485 [6] L. Al-Haddad, C. W. Morris, L. Boddy, Training radial basis function neural  
486           networks: effects of training set size and imbalanced training sets., *Journal*  
487           of Microbiological Methods 43 (1) (2000) 33–44.
- 488 [7] T. Fawcett, F. Provost, Adaptive fraud detection, *Data Min. Knowl. Discov.*  
489           1 (3) (1997) 291–316.
- 490 [8] I. Brown, C. Mues, An experimental comparison of classification algorithms

- 491 for imbalanced credit scoring data sets, *Expert Systems with Applications*  
492 39 (3) (2012) 3446 – 3453.
- 493 [9] P. Domingos, Metacost: a general method for making classifiers cost-  
494 sensitive, in: *Proceedings of the fifth ACM SIGKDD international con-*  
495 *ference on Knowledge discovery and data mining, KDD '99, 1999, pp.*  
496 155–164.
- 497 [10] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, Smote: Syn-  
498 thetic minority over-sampling technique, *Journal of Artificial Intelligence*  
499 *Research* 16 (2002) 321–357.
- 500 [11] A. Estabrooks, T. Jo, N. Japkowicz, A multiple resampling method for learn-  
501 ing from imbalanced data sets, *Computational Intelligence* 20 (1) (2004)  
502 1836.
- 503 [12] S. García, F. Herrera, Evolutionary undersampling for classification with im-  
504 balanced datasets: Proposals and taxonomy, *Evolutionary Computation*  
505 17 (2009) 275–306.
- 506 [13] R. Anand, K. Mehrotra, C. Mohan, S. Ranka, Efficient classification for  
507 multiclass problems using modular neural networks, *IEEE Transactions*  
508 *on Neural Networks* 6 (1) (1995) 117–124.

- 509 [14] S.-H. Oh, Error back-propagation algorithm for classification of imbalanced  
510 data, *Neurocomputing* 74 (6) (2011) 1058–1061.
- 511 [15] G. E. A. P. A. Batista, R. C. Prati, M. C. Monard, Balancing strategies and  
512 class overlapping, in: *IDA, 2005*, pp. 24–35.
- 513 [16] M. Denil, T. P. Trappenberg, Overlap versus imbalance, in: *Canadian Con-*  
514 *ference on AI, 2010*, pp. 220–231.
- 515 [17] V. García, R. A. Mollineda, J. S. Sánchez, On the k-nn performance in a  
516 challenging scenario of imbalance and overlapping, *Pattern Analysis and*  
517 *Applications* 11 (3) (2008) 269–280.
- 518 [18] J. Olvera-López, J. Carrasco-Ochoa, J. Martínez-Trinidad, J. Kittler, A re-  
519 view of instance selection methods, *Artificial Intelligence Review* 34  
520 (2010) 133–143.
- 521 [19] R. Anand, K. Mehrotra, C. Mohan, S. Ranka, An improved algorithm for  
522 neural network classification of imbalanced training sets., *IEEE Transac-*  
523 *tions on Neural Networks* 4 (1993) 962–969.
- 524 [20] S. Lawrence, I. Burns, A. Back, A. Tsoi, C. L. Giles, Neural network clas-  
525 sification and unequal prior class probabilities, in: G. Orr, K.-R. Müller,

- 526 R. Caruana (Eds.), *Neural Networks: Tricks of the Trade*, Lecture Notes  
527 in Computer Science, Springer Verlag, 1998, pp. 299–314.
- 528 [21] N. Japkowicz, S. Stephen, The class imbalance problem: A systematic study,  
529 *Intell. Data Anal.* 6 (5) (2002) 429–449.
- 530 [22] R. Prati, G. Batista, M. Monard, Class imbalances versus class overlapping:  
531 An analysis of a learning system behavior, in: *MICAI, 2004*, pp. 312–321.
- 532 [23] S. Visa, A. Ralescu, Issues in mining imbalanced data sets - a review paper,  
533 in: *Artificial Intelligence and Cognitive Science Conference, 2005*, pp.  
534 67–73.
- 535 [24] Y. Tang, J. Gao, Improved classification for problem involving overlapping  
536 patterns, *IEICE Transactions* 90-D (11) (2007) 1787–1795.
- 537 [25] R. Kretschmar, N. B. Karayiannis, F. Eggimann, Feedforward neural net-  
538 work models for handling class overlap and class imbalance, *Int. J. Neural*  
539 *Syst.* 15 (5) (2005) 323–338.
- 540 [26] C. M. Bishop, *Neural Networks for Pattern Recognition*, 1st Edition, Oxford  
541 University Press, USA, 1996.

- 542 [27] D. R. Wilson, T. R. Martinez, Reduction techniques for instance-based learn-  
543 ing algorithms, *Machine Learning* 38 (3) (2000) 257–286.
- 544 [28] B. V. Dasarathy, J. S. Sánchez, S. Townsend, Nearest neighbour editing and  
545 condensing tools-synergy exploitation, *Pattern Anal. Appl.* 3 (1) (2000)  
546 19–30.
- 547 [29] J. S. Sánchez, F. Pla, F. J. Ferri, Prototype selection for the nearest neighbour  
548 rule through proximity graphs, *Pattern Recognition Letters* 18 (6) (1997)  
549 507–513.
- 550 [30] D. Wilson, Asymptotic properties of nearest neighbor rules using edited data,  
551 *IEEE Transactions on Systems, Man and Cybernetics* 2 (4) (1972) 408–  
552 420.
- 553 [31] S. Serpico, F. Roli, P. Pellegritti, G. Vemazza, Structured neural networks for  
554 the classification of multisensor remote-sensing images., in: *Int. Geosci.*  
555 *Remote Sensing Symp.*, 1993, pp. 907–909.
- 556 [32] D. N. A. Asuncion, UCI machine learning repository (2007).  
557 URL [http://www.ics.uci.edu/~l  
sim\\$mllearn/{MLR}epository.html](http://www.ics.uci.edu/~l<br/>558 sim$mllearn/{MLR}epository.html)

- 559 [33] R. Kohavi, G. H. John, Wrappers for feature subset selection, *Artif. Intell.*  
560 97 (1-2) (1997) 273–324.
- 561 [34] H. He, E. Garcia, Learning from imbalanced data, *IEEE Transactions on*  
562 *Knowledge and Data Engineering In Knowledge and Data Engineering*  
563 21 (9) (2009) 1263–1284.
- 564 [35] T. Fawcett, An introduction to roc analysis, *Pattern Recogn. Lett.* 27 (2006)  
565 861–874.
- 566 [36] M. Kubat, S. Matwin, Addressing the curse of imbalanced training sets:  
567 one-sided selection, in: *Proc. 14th International Conference on Machine*  
568 *Learning*, Morgan Kaufmann, 1997, pp. 179–186.
- 569 [37] J. Demšar, Statistical comparisons of classifiers over multiple data sets, *Jour-*  
570 *nal of Machine Learning Research* 7 (1) (2006) 1–30.
- 571 [38] R. L. Iman, J. M. Davenport, Approximations of the critical region of the  
572 friedman statistic, *Communications in Statistics - Theory and Methods*  
573 9 (6) (1980) 571–595.
- 574 [39] G. M. Weiss, F. J. Provost, Learning when training data are costly: The

575 effect of class distribution on tree induction, *J. Artif. Intell. Res. (JAIR)*  
576 19 (2003) 315–354.