Pattern Recognition 135 (2023) 109158

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Optimum Bayesian thresholds for rebalanced classification problems using class-switching ensembles

Aitor Gutiérrez-López, Francisco-Javier González-Serrano*, Aníbal R. Figueiras-Vidal

Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Avda. Universidad, 30, Leganés 28911, Spain

ARTICLE INFO

Article history:
Received 8 May 2022
Revised 3 November 2022
Accepted 5 November 2022
Available online 7 November 2022

Keywords:
Bayesian framework
Ensembles
Rebalancing techniques
Imbalanced classification
Label switching

ABSTRACT

Asymmetric label switching is an effective and principled method for creating a diverse ensemble of learners for imbalanced classification problems. This technique can be combined with other rebalancing mechanisms, such as those based on cost policies or class proportion modifications. In this study, and under the Bayesian theory framework, we specify the optimal decision thresholds for the combination of these mechanisms. In addition, we propose using a gating network to aggregate the learners contributions as an additional mechanism to improve the overall performance of the system.

© 2022 The Author(s). Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license
(http://creativecommons.org/licenses/by-nc-nd/4.0/)

1. Introduction

Data imbalance occurs in many real-world application areas, where the decision system is aimed at detecting rare but important cases. They can be found in information technology area [1,2]; biomedical data [3,4]; industrial applications [5]; and financial areas [6].

This imbalance implies difficulty in learning algorithms because they will be biased towards the most frequent (and usually less important) cases. To overcome such bias towards the majority class examples, specific machine learning algorithms must be applied. Even if we limit the search to recent years, it would be pretentious to list all relevant works related to these algorithms (there have been more than 6500 papers listed in Google Scholar in the last 3 years). Therefore, we prefer to suggest tutorials [7–9] (and references therein) to present a complete overview to the interested reader.

In general, approaches that address imbalance can be sorted into three categories:

1) Data-level methods concentrate on modifying the training set to make it suitable for a standard learning algorithm. Balancing distributions by creating new objects for minority classes (oversampling and variations such as SMOTE [10]) and removing examples from majority classes (undersampling [11]) belong to this category. A recently published study [12] proposed an improved version of SMOTE for high-dimensional datasets, which supports the relevance of these methods for imbalanced classification.

2) Algorithm-level methods modify existing learning algorithms to alleviate bias towards majority class examples. A recent example is [13], where the authors used modified SVMs to deal with imbalanced data and can be extended to multi-class problems. Cost-sensitive approaches [14,15] fall into this category.

3) Ensemble learning [16], in which multiple base learners are trained using diverse examples, and their complementary (or uncorrelated) predictions are fused to yield a final decision. According to [17], adequate diversity-increasing techniques may significantly improve the performance of ensemble methods for imbalanced problems. A more recent study [18] used Random Forest ensembles in combination with neutral data resampling.

According to [19], algorithm-level and, in particular, cost-sensitive approaches are more problem-dependent, whereas data-level and ensemble learning approaches are more versatile. Obviously, these methods can be combined resulting in hybrid approaches [20,21], where the capabilities and limitations of each method can be respectively exploited and mitigated, respectively.

This is precisely what we present in this work: an ensemble learning method, based on discriminative machines with universal approximation capabilities, that combines the aforementioned mechanisms. Our main contribution is to specify, under the Bayesian decision theory, the optimum decision thresholds that consider the intensities of the partial rebalance provided by

* Corresponding author.
E-mail addresses: aitorgl@tsc.uc3m.es (A. Gutiérrez-López), fran@ing.uc3m.es (F.-J. González-Serrano), arfv@ing.uc3m.es (A.R. Figueiras-Vidal).

https://doi.org/10.1016/j.patcog.2022.109158
0031-3203/© 2022 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license
(http://creativecommons.org/licenses/by-nc-nd/4.0/)

a neutral[1] data-level method, the (mis)classification costs at the algorithm-level, and the ensemble diversity.

The remainder of this study is organized as follows. Starting from the classical Bayes classification theory, in Section 2, we justify that the combination of neutral rebalancing procedures and Bregman divergences [22] as surrogate training costs for the ensemble's machines permits the estimation of an equivalent likelihood ratio that solves the original imbalanced problem. Sections 3 and 4 discuss the implications of the asymmetric label switching [23], as a method for creating diverse learners using the presented Bayesian approach. In Section 5, we propose the use of an example-dependent weighted combination of base learners (i.e., a Mixture-of-Experts) in the ensemble output layer. Section 6 presents the experimental results for the proposed ensemble method. The main conclusions and directions for further research close this study.

2. Likelihood ratio equivalent classification problems

Bayes theory establishes that the minimization of the average classification cost in a binary problem leads to the Likelihood Ratio Test (LRT) [24]:

$$q(\mathbf{x}) = \frac{p(\mathbf{x}|C_1)}{p(\mathbf{x}|C_0)} \underset{C_0}{\overset{C_1}{\gtrless}} \frac{c_{10}-c_{00}}{c_{01}-c_{11}}\frac{P_0}{P_1} = Q_C Q_P = Q \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^d$ is the observed sample, $\{C_0, C_1\}$ $(\equiv \{-1, +1\})$ the classes (we assume that C_1 is the minority class), $p(\mathbf{x}|C_i)$ the class C_i likelihood, $q(\mathbf{x})$ the Likelihood Ratio, LR, c_{ji} the cost of attributing a class C_i sample to class C_j, P_i the ' a priori' probability of the class C_i, $Q_C = [c_{10} - c_{00}]/[c_{01} - c_{11}]$, and $Q_P = P_0/P_1$ (the Imbalance Ratio (IR)).

This means that two ' problems' with the same LR, $q(\mathbf{x})$, and different values of Q (i.e., different cost policies, Q_C, or/and ' a priori' probabilities, Q_P) are solved equivalently, the only difference being the classification threshold (Q). In other words, $q(\mathbf{x})$ allows the construction of the NP-ROC)[2] for the problem model.

The above serves to conceive a principled method to deal with imbalanced problems when working with classification machines: If one of such machines is able to provide a good estimate of $q(\mathbf{x})$ and there is a way of rebalancing the problem without modifying its LR (neutral rebalancing), we can transform the imbalanced problem, in which $Q \gg 1$ (because $Q_P \gg 1$ and/or $Q_C \gg 1$), into an equivalent (more) "balanced" problem, and, finally, use the corresponding LR estimate to solve the original imbalanced problem.

The previous idea can be made possible by using discriminative machines with trainable transformations, such as MLPs. These machines must be trained using Bregman divergences [22,25], such that:

$$\frac{\partial \mathcal{L}_{\omega(\mathbf{x})}(y, o)}{\partial o} = -g(o)(y - o) \tag{2}$$

y is the target, o is the machine output, $\mathcal{L}_{\omega(\mathbf{x})}(y, o)$ is a weighted[3] Bregman Loss Function[4], and $g(o) > 0$ is an arbitrary (positive) function. After optimizing the machine weights using a Bregman Loss Function over the training samples $\{y_n, o(\mathbf{x}_n)\}_{n=1}^{N}$,

$$\mathbf{w}_{\text{opt}} = \underset{\mathbf{w}}{\operatorname{argmin}} \sum_n \mathcal{L}_{\omega(\mathbf{x}_n)}(y_n, o(\mathbf{x}_n; \mathbf{w})) , \tag{3}$$

the machine provides an estimate of the conditional mean of y, $\mathrm{E}\{y|\mathbf{x}\}$. This is a necessary and sufficient condition and its proof is immediate. For a binary classification problem with $y_n = \pm 1$,

$$o(\mathbf{x}) = \mathrm{E}(y|\mathbf{x}) = \Pr(C_1|\mathbf{x}) - \Pr(C_0|\mathbf{x}) = 2\Pr(C_1|\mathbf{x}) - 1 \tag{4}$$

and the posterior probability of minority class C_1

$$\Pr(C_1|\mathbf{x}) = [o(\mathbf{x}) + 1]/2 \tag{5}$$

is obtained[5]

Considering the one-to-one correspondence

$$\Pr(C_1|\mathbf{x}) = \frac{p(\mathbf{x}|C_1)P_1}{p(\mathbf{x}|C_1)P_1 + p(\mathbf{x}|C_0)P_0} = \frac{1}{1 + Q_P/q_L(\mathbf{x})} \tag{6}$$

and using (5), the LR can be expressed as

$$q(\mathbf{x}) = Q_P \frac{1 + o(\mathbf{x})}{1 - o(\mathbf{x})} \tag{7}$$

Alternative expressions for the LRT can be derived by applying Bayes' rule and simple mathematical transformations, such as

$$\Pr(C_1|\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} \frac{Q}{Q + Q_P} \tag{8a}$$

or, equivalently, as

$$o(\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} \frac{Q - Q_P}{Q + Q_P} = \eta \tag{8b}$$

In addition, it is necessary to use rebalancing mechanisms to transform the original problem into a new one. Thus, the classification machines provide better estimates of the LR and consequently better decisions. This requires a conveniently modified decision threshold that considers the impact of the rebalancing mechanism.

Among those methods, we will focus on the ' neutral' mechanisms, because they do not (essentially) change the LR of the problem. Namely:

Randomly resample (separately for each class) the training set examples, including standard subsampling and oversampling bootstrap-based processes, but not "informed" resampling[6]

Randomly generate (separately for each class) new samples, including SMOTE [10]. In both cases, the new population ratio is given by Q_{R_P}, where $1 \leq Q_{R_P} < Q_P = \text{IR}$.

Modify the cost policy, which can be described in terms of Q_{R_C}.

And, obviously, their combinations, $Q_R = Q_{R_P} \cdot Q_{R_C}$.

For random resampling and sample generation, ensembles are mandatory to approximate statistical neutrality [28].

If we follow the requirements stated above, the LR in the new rebalanced problem is given by

$$q_R(\mathbf{x}) = Q_R \frac{1 + o(\mathbf{x})}{1 - o(\mathbf{x})} \tag{9}$$

From this expression, the solution to the original imbalanced problem is

$$o(\mathbf{x}) \underset{C_0}{\overset{C_1}{\gtrless}} \frac{Q - Q_R}{Q + Q_R} = \eta_R \tag{10}$$

Reference [28] presented a more detailed discussion of the LR equivalence method.

[1] A neutral rebalance must keep invariant the likelihood ratio. The well-known bootstrap and SMOTE techniques fall into the category (see Section 2).

[2] The Neyman-Pearson Receiver Operating Characteristic, or NP-ROC, is the curve that presents the detection probability $P_D = \Pr(\text{decide } C_1|\mathbf{x} \in C_1)$ vs. the false alarm probability $P_{FA} = \Pr(\text{decide } C_1|\mathbf{x} \in C_0)$ (or true positive and false positive probabilities, respectively).

[3] The term $\omega(\mathbf{x})$ in $\mathcal{L}_{\omega(\mathbf{x})}(y, o)$ specifies an example-dependent cost (or weight).

[4] Common Bregman Loss Functions in machine learning are the (weighted) squared error, the Mahalanobis distance, the (negative) exponential loss, the logistic loss, or the Kullback-Leibler Divergence [26].

[5] It should be noted that machines trained with Bregman divergences produce consistent *estimates* of $\mathrm{E}(C_i|\mathbf{x})$. Therefore, we should have written $o(\mathbf{x}) = \widehat{\mathrm{E}}(C_i|\mathbf{x}) = 2\widehat{\Pr}(C_1|\mathbf{x}) - 1$, where the symbol $\widehat{\ }$ denotes estimation. In an abuse of notation, but for clarification purposes, henceforth, we will drop that symbol.

[6] Informed resampling [27] uses the local or global information of the class distribution to remove or generate instances. Therefore, it modifies the empirical distribution of the data.

3. Ensemble diversity by asymmetric label switching

An asymmetric switching mechanism [23] changes the labels of randomly selected samples at different rates for each class.

The application of a random label switching with rates α and β to the C_0 and C_1 samples, respectively. $\alpha > \beta$ results in a new and more balanced classification problem S, which will have " classes"C_1' and C_0'.

The new " class" probabilities are given by

$$\Pr_S(C_1'|\mathbf{x}) = \frac{p(\mathbf{x}|C_1)(1-\beta)P_1 + p(\mathbf{x}|C_0)\alpha P_0}{p(\mathbf{x}|C_1)P_1 + p(\mathbf{x}|C_0)P_0} \tag{11}$$

Moreover, by adding and subtracting $\alpha p(\mathbf{x}|C_1)P_1$ from the numerator of (11), we obtain

$$\Pr_S(C_1'|\mathbf{x}) = (1-\alpha-\beta)\Pr(C_1|\mathbf{x}) + \alpha \tag{12}$$

from which

$$\Pr(C_1|\mathbf{x}) = \frac{\Pr_S(C_1'|\mathbf{x}) - \alpha}{1-\alpha-\beta} \tag{13}$$

that is, we can recover (an estimate of) $\Pr(C_1|\mathbf{x})$ from (an estimate of) $\Pr_S(C_1'|\mathbf{x})$.

From (13) and (8a), we obtain

$$\Pr_S(C_1'|\mathbf{x}) \overset{C_1}{\underset{C_0}{\gtrless}} \alpha + (1-\alpha-\beta)\frac{Q}{Q+Q_P} \tag{14}$$

Applying (4) to (14) leads to

$$o_S(\mathbf{x}) \overset{C_1}{\underset{C_0}{\gtrless}} 2\left[\alpha + (1-\alpha-\beta)\frac{Q}{Q+Q_P}\right] - 1 = \eta_S \tag{15}$$

as the classification rule for ensemble learners that work with (consistently) estimated values. Clearly, (15) can be applied to each learner and then produce the output using a majority vote rule. However, this will close the door to the possibility of including other principled output aggregation methods, such as that proposed in Section 5.

4. Combining asymmetric label switching with LR equivalent rebalance

Using the LR equivalence $q(\mathbf{x})/Q_P = q_R(\mathbf{x})/Q_R$ (Eq. (7) and (9)), and the expression for the class probabilities (14) in the new switching problem, we obtain

$$\Pr_S(C_1'|\mathbf{x}) \overset{C_1}{\underset{C_0}{\gtrless}} \alpha + (1-\alpha-\beta)\frac{Q}{Q+Q_R} \ , \tag{16}$$

which leads to the classification rule:

$$o_S(\mathbf{x}) \overset{C_1}{\underset{C_0}{\gtrless}} 2\left[\alpha + (1-\alpha-\beta)\frac{Q}{Q+Q_R}\right] - 1 = \eta_{RS} \tag{17}$$

Additionally, if we apply a rebalancing cost policy, the loss function to be optimized is $\mathcal{L}'_{\omega(\mathbf{x})}$, which includes the weighting factor Q/Q_{R_C} in the minority samples.

$$\begin{aligned}\mathcal{L}'_{\omega(\mathbf{x})}(y_n, o(\mathbf{x}_n;\mathbf{w})) = &\sum_{n\in C_0}\mathcal{L}_{\omega(\mathbf{x})}(-1, o(\mathbf{x}_n;\mathbf{w})) \\ &+\frac{Q}{Q_{R_C}}\sum_{n\in C_1}\mathcal{L}_{\omega(\mathbf{x})}(1, o(\mathbf{x}_n;\mathbf{w}))\end{aligned} \tag{18}$$

Please note that weighting the majority samples by Q_{R_C}/Q is also a valid option.

5. Aggregating outputs

A key factor in the design of ensembles of classifiers is the choice of the aggregation method to combine the results of the base learners into a single result. The aggregation strategies proposed in the literature follow two models: fusion and selection [29]. In classifier fusion, each ensemble member is supposed to have knowledge of the whole feature space, so all base classifiers are involved in the final ensemble decision; methods such as (weighted) majority vote (assuming hard decisions) or averaging (if soft outputs are produced) belong to this group. In classifier selection, ensemble members are responsible for a subset of input examples lying in specific local regions of the feature space, according to their competencies.

An alternative lying between the "pure" fusion and selection strategies, is the Mixture-of-Experts model [30]. In this model, for any given input $\mathbf{x}$, the gating network is responsible for learning the appropriate weighted combination of estimations of the " a posteriori" class probability function obtained by each ensemble learner (or expert). The function is defined as follows:

$$p(y|\mathbf{x};\Theta) = \sum_{m=1}^{M} g_m(\mathbf{x};\Theta_g)p_m(y|\mathbf{x};\mathbf{w}_m), \tag{19}$$

where M is the number of experts, the function $g_m(\mathbf{x};\Theta_g)$ is the gate output for expert m given $\mathbf{x}$ (satisfying the usual constraints $\sum_{m=1}^{M} g_m(\mathbf{x}) = 1$, and $0 \le g_m(\mathbf{x}) \le 1$), with parameters Θ_g, $p_m(y|\mathbf{x};\mathbf{w}_m)$ is the centered estimation of the "a posteriori" class probability function obtained by each ensemble learner (see (13)); and $\mathbf{w}_m$ are the learner weights. The collection of both expert and gating network parameters is defined as $\Theta = \{\Theta_g, \mathbf{w}_1, \ldots \mathbf{w}_M\}$.

Even if we know the full statistical model of the data (features and labels), the estimation of the global optimum parameters Θ is a difficult (if not unfeasible) problem. Therefore, we use the Expectation-Maximization (E-M) approach. EM uses a two-step iterative procedure, which usually appears to be faster than gradient descent [31]. The first step, the Expectation step (E-step), involves computing the conditional expectation of the log-likelihood given the observed data and the current estimates. In the second step, called the Maximization step (M-step), new gating and ensemble parameters are determined.

In particular, the posterior probabilities $h_m^{(s)}(\mathbf{x}_n)$ for each data pair $(\mathbf{x}_n, y_n)$ are computed in the E-step (s-th epoch) as:

$$h_m^{(s)}(\mathbf{x}_n) = \frac{g_m(\mathbf{x}_n;\mathbf{v}_m^{(s)})p(y_n|\mathbf{x}_n;\mathbf{w}_m^{(s)})}{\sum_{k\neq m} g_k(\mathbf{x}_n;\mathbf{v}_k^{(s)})p(y_n|\mathbf{x}_n;\mathbf{w}_k^{(s)})} \tag{20}$$

The M-step is then divided into two parts. In the first, and for each learner, a modified version of Eq. (3) is minimized:

$$\mathbf{w}_{m,\text{opt}}^{(s+1)} = \underset{\mathbf{w}_m}{\text{argmin}} \sum_{n=1}^{N} \mathcal{L}_{h_m^{(s)}(\mathbf{x}_n)}(y_n, o(\mathbf{x}_n;\mathbf{w}_m)) \tag{21}$$

where the posterior probabilities $h_m^{(s)}(\mathbf{x}_n)$ act as an extra weighting factor, causing the gate network to reward experts that make good predictions with stronger error feedback updates. It is important to point out that a rebalancing cost policy, similar to that considered in (18), can also be included in (21).

In the second part of the M-step, the following maximization problem is solved for the gating network:

$$\mathbf{V}_{\text{opt}}^{(s+1)} = \underset{\mathbf{V}}{\text{argmax}} \sum_{n=1}^{N}\sum_{m=1}^{M} h_m^{(s+1)}(\mathbf{x}_n)\log g_m(\mathbf{x}_n;\mathbf{v}_m) \tag{22}$$

A simple heuristic in the M-step for the overall set of linear coefficients $\mathbf{V}$ reduces the optimization to a one-pass weighted least squares computation [30], which results in a computational complexity of $\mathcal{O}(N \cdot M \cdot (d+1)^2)$.

Table 1
Description of the datasets. N: Number of instances, d: Number of attributes, IR: Imbalance Ratio. The datasets were ordered according to their IR.

Datasets	Description	N	d	IR
EcoliImU	UCI, class imU vs rest	336	7	8.6
Satimage4	UCI, class 4 vs rest	6435	36	9.3
Abalone7	UCI, class 7 vs rest	4177	8	9.7
Ringnorm10	Synthetic dataset	1650	20	10
BalanceB	UCI, class B vs rest	625	4	11.8
Ecoli4	UCI, Keel, class 4 vs rest	336	7	15.8
Aba9vs18	UCI, Keel, class 9 vs 18	731	8	16.4
SolarflareM0	UCI, target: M->0	1389	32	19
Ringnorm20	Synthetic dataset	3150	20	20
Oil	UCI, target: minority class	937	49	22
Flare-F	UCI, Keel, class F vs rest	1066	11	23.8
Winequal4	UCI, target: <=4	4898	11	26
LetterimgZ	UCI, class Z vs rest	20,000	16	26
Yeast4	UCI, Keel, class 4 vs rest	1484	8	28.1
Aba17	UCI, Keel, class 17 vs 7,8,9,10	2338	8	39.3
Ringnorm40	Synthetic dataset	6150	20	40
Yeast6	UCI, Keel, class 4 vs rest	1484	8	41.4

6. Experiments

The imbalanced datasets are presented in Table 1. As it can be seen, they cover a wide range of Imbalance Ratios, dimensionality and sizes.

All datasets were obtained from Keel [32] and UCI [33]. Those from the UCI were configured in the same way as [34]. `Ringnorm` was replicated from Breiman's work [35].

6.1. Performance indicators

There is a debate about whether the F_1-score or Matthews Correlation Coefficient (MCC) is the most appropriate for imbalanced datasets [36]. F_1-score provides the harmonic mean between *precision* and *recall*. It can also be expressed as a function of TP (True Positives), FP (False Positives), and FN (False Negatives), as follows:

$$F_1 = 2\frac{Precision \cdot Recall}{Precision + Recall} = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

Unlike F_1-score, which does not take into account True Negatives (TN), MCC is a metric that includes all four confusion matrix categories [37]:

$$\mathrm{MCC} = \frac{\mathrm{TP}\cdot\mathrm{TN} - \mathrm{FP}\cdot\mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}$$

In our experiments, we observed that, from a qualitative point of view, the relative performance improvement of the proposed methods is similar, whether F_1-score or MCC is used. For this reason, and for simplicity, we primarily use the F_1-score as the base metric.

6.2. Description of the experiments

In our experiments, we used an ensemble of M=31 MLP base learners, each with a single hidden layer with n_h=4 neurons, a hyperbolic tangent as the activation function, and the modified output activation function `act1` described in [23] (Section 2).

For the loss function, we used the weighted squared error, which is a Bregman divergence. The learners' weights were optimized using the LBFGS-B method (see Eqs. (3) or (21)), which has a computational complexity of $\mathcal{O}(N\cdot M\cdot d\cdot n_h\cdot l_{mem}\cdot k_{LBFGS})$, where l_{mem} and k_{LBFGS} are the size of the memory and the number of iterations [38].

Ensemble diversity is achieved by random initialization of the weights and asymmetric label switching. It is important to point out that we have made no attempt to optimize the default implementations of `sckit-learn` or `Pytorch` for both the ensemble of MLPs and the LBFGS-B method (l_{mem}=10, and $k_{LBFGS} \leq 150$), so the presented results are simply a reference to demonstrate the relative performance gain resulting from the studied methods[7]

We averaged the results over 50 independent runs, each with a random 75%-25% train-test split.

We considered different rebalancing strategies, namely:

1. Asymmetric label switching uses an averaging output layer to fuse learners' estimations (`SW`).
2. A statistically neutral rebalancing technique was applied to the input to achieve more balanced populations (`RB_SW`). In this study, we used the standard SMOTE implementation of `imblearn`.
3. Applying a cost-sensitive training that weighs the minority samples `Cost_SW`.
4. Asymmetric label switching using a gating network at the output layer (`SW_Gate`). In our experiments we used a gating network with a linear structure at the input and softmax activation output layer. In our experiments, the gating network parameters were estimated after $N_{EM} = 5$ iterations of the E-M algorithm described in Section 5.

We explored switching rates (α, β) from 0 to 0.45 in 0.05 steps. The intensities of the neutral population and cost rebalance (Q_{R_P} and Q_{R_C}, respectively) ranged from 1 to IR. Note that, when $\alpha = 0$ and $\beta = 0$, the results obtained with RB_SW and Cost_SW describe the performance of the rebalance methods (SMOTE and cost weighting, respectively) with no asymmetric label switching.

6.3. Results

The first analysis concerns the ensemble output aggregation mechanism, that is, averaging versus gating network.

The first point to consider is computational complexity. Whereas averaging requires a single optimization of (21) (with complexity $\mathcal{O}(N\cdot M\cdot d\cdot n_h\cdot l_{mem}\cdot k_{LBFGS})$), each iteration of the E-M algorithm involves the sequential optimization of (21) and (22) (gating network coefficients), which has a complexity of $\mathcal{O}(N\cdot M\cdot (d+1)^2)$. Because the last term is usually much smaller than the previous one (especially for low dimensional datasets, $d < 20$), the computational complexity using a gating network (vs. averaging) is approximately N_{EM} (number of E-M iterations) times greater. To interpret the orders of magnitude in a practical manner, Table 2 presents the training time (mean and standard deviation) of an ensemble of 31 MLPs (4 neurons each) on a 6-core @ 3.5 GHz Intel Xeon processor with 32 Gb of RAM, averaged over 50 runs. The N_{EM}-fold increase in complexity is an important factor in fixed-computational-budget scenarios, in which it may be better to simply average over larger ensembles than to optimize the parameters of the gating network.

The second point concerns the relationship between IR and switching rates (α, β). To simplify the analysis, we used the synthetic `Ringnorm` dataset [35]. In addition, to simplify the interpretation of the results, we only consider the β value at which the maximum performance in F_1-score is reached ($\beta = 0.05$ for IR=10; and $\beta = 0.0$ for IR=20). The results are presented in Fig. 1, where, in addition to the mean value of F_1, the standard deviation is also shown (error bars at each point).

[7] We leave an (under development) implementation of the Asymmetric Label Switching algorithm at https://github.com/franjgs/LabelSwitching for the interested readers.

Table 2
Training Time (seconds and increase factor; (average ± standard deviation)) comparison over 50 runs for learners fusion by averaging and by gating network ($N_{EM} = 5$) on a 6-core 3.5 GHz Intel Xeon platform.

Datasets	N	d	Averaging	Gating Network	Inc. Factor
EcoliImU	336	7	4.06± 0.09	18.60±0.77	4.58±0.20
Satimage4	6435	36	10.92±0.97	59.80±0.87	5.48±0.45
Abalone7	4177	8	8.27± 0.18	41.2± 0.13	4.98±0.12
Ringnorm10	1650	20	5.92± 0.12	25.77± 0.70	4.35±0.24
BalanceB	625	4	4.56± 0.06	19.61±0.58	4.30±0.11
Ecoli4	336	7	4.21±0.24	19.18±0.73	4.56±0.23
Aba9vs18	731	8	4.98± 0.08	24.96±0.49	5.01±0.22
SolarflareM0	1389	32	6.07±0.14	30.62±0.64	5.04±0.10
Ringnorm20	3150	20	7.68±0.15	37.60±0.44	4.90±0.13
Oil	937	49	5.48±0.12	23.94±0.94	4.37±0.21
Flare-F	1066	11	5.52±0.06	26.03±0.20	4.72±0.13
Winequal4	4898	11	9.08±0.26	45.95±0.32	5.06±0.18
LetterimgZ	20,000	16	22.04±0.67	119.7±1.66	5.43±0.17
Yeast4	1484	8	5.77±0.07	28.86±0.19	5.00±0.09
Aba17	2338	8	6.57±0.25	33.89±0.19	5.16±0.19
Ringnorm40	6150	20	11.03±0.42	53.97±1.08	4.89±0.20
Yeast6	1484	8	5.94±0.17	29.07±0.39	4.89±0.11

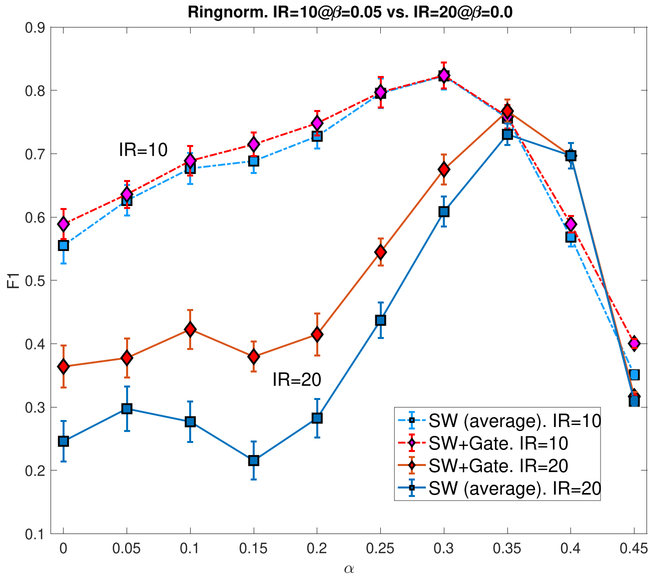

Fig. 1. F_1-score evolution for `Ringnorm` (IR=10 vs. IR=20).

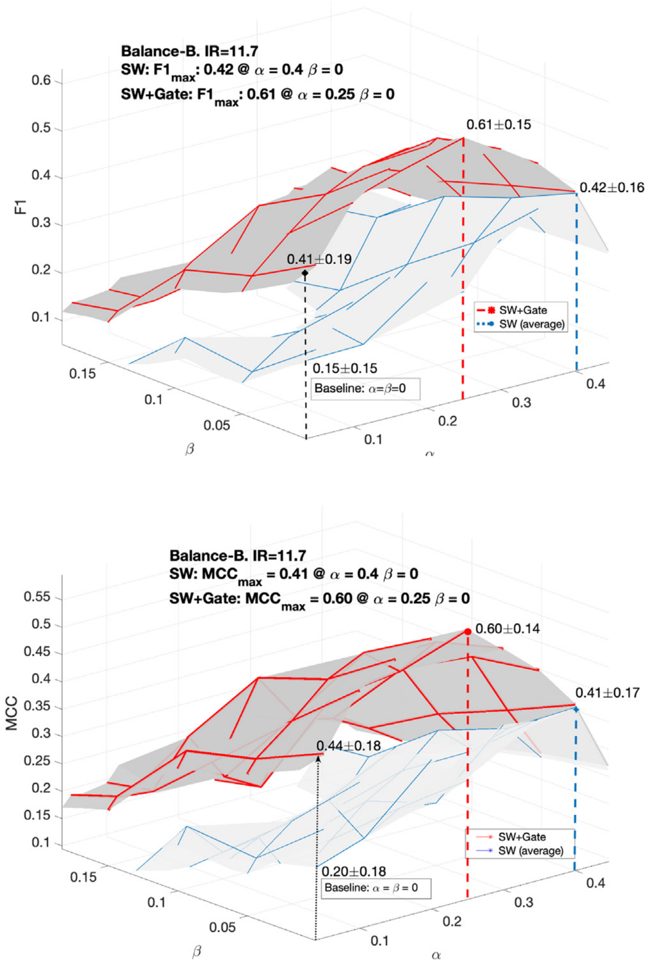

Fig. 2. F_1-score and MCC evolution for `Balance-B` (IR=11.7). The darker-colored surface corresponds to the Switching+Gate ensemble.

It is clearly seen that the α values (majority class switching rate; $C_0 \rightarrow C_1$, or $-1 \rightarrow +1$) that achieve the best performance increase with the imbalance ($\alpha_{\text{opt}} = 0.3$ for IR=10 versus $\alpha_{\text{opt}} = 0.35$ for IR=20), which makes sense because the higher the IR, the harder it is for machines to detect minority samples. Therefore, increasing the majority class switching rate improves the performance. Clearly, if α is too aggressive ($\alpha > 0.35$), the performance degrades again. Another important aspect is that the improvement produced by the gating network is more significant when the Imbalance Ratio is greater, and the experts' performance is worse (for low α).

In Fig. 2, we represent the behavior in terms of F_1-score and MCC of the output aggregation mechanism on the UCI dataset `Balance-B` (which was chosen only for illustration purposes). As mentioned previously, from a qualitative point of view (and for this dataset, quantitative as well), the relative improvement in performance for both metrics is similar. It can also be seen how, in this case, the contribution of the gate is significant throughout the range of α and β values. Thus, without label switching ($\alpha = \beta = 0$), the ensemble with aggregation by averaging, or "baseline" (diversity by initialization of weights), achieves a mean F_1-score of 0.15 (with a standard deviation of 0.15), while the weighted aggregation raises this value to 0.41 (standard deviation 0.18), which is comparable to the maximum reached by SW (0.42 ± 0.16 at $\alpha = 0.4$, $\beta = 0.0$). The SW_Gate optimum is 0.61 ± 0.15, which is

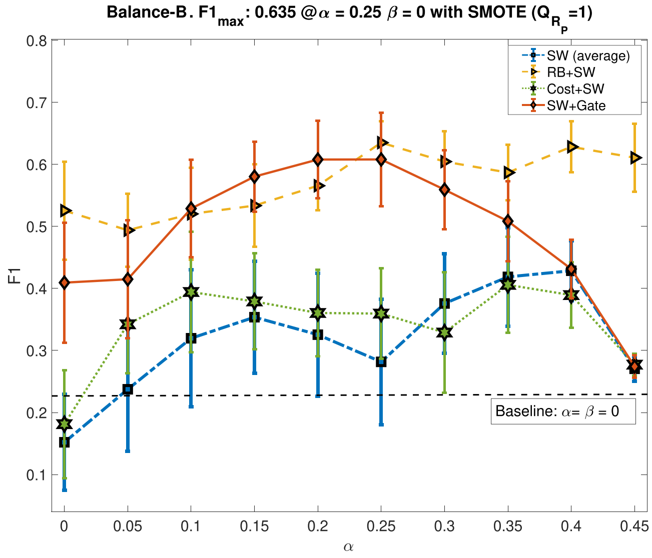

Fig. 3. F$_1$-score evolution for `Balance-B`. Comparison of methods.

Table 3
F$_1$-score confidence intervals (average standard deviation) for the datasets considered.

Datasets	Baseline	SW (α, β)
EcoliImU	0.57 ± 0.12	0.68 ± 0.05 (0.25, 0.05)
Satimage4	0.60 ± 0.03	0.63 ± 0.03 (0.2, 0)
Abalone7	0.00 ± 0.00	0.43 ± 0.04 (0.35, 0.1)
Ringnorm10	0.59 ± 0.08	0.80 ± 0.05 (0.25, 0)
BalanceB	0.15 ± 0.15	0.42 ± 0.16 (0.4, 0)
Ecoli4	0.82 ± 0.12	0.88 ± 0.12 (0.15, 0.15)
Aba9vs18	0.51 ± 0.13	0.61 ± 0.12 (0.25, 0)
SolarflareM0	0.08 ± 0.07	0.24 ± 0.07 (0.4, 0.05)
Ringnorm20	0.28 ± 0.07	0.73 ± 0.06 (0.35, 0)
Oil	0.57 ± 0.13	0.62 ± 0.12 (0.15, 0)
Flare-F	0.20 ± 0.11	0.34 ± 0.09 (0.4, 0.1)
Winequal4	0.21 ± 0.05	0.27 ± 0.06 (0.4, 0)
LetterimgZ	0.92 ± 0.02	0.94 ± 0.01 (0.05, 0.05)
Yeast4	0.32 ± 0.10	0.44 ± 0.12 (0.35, 0)
Aba17	0.28 ± 0.12	0.44 ± 0.09 (0.4, 0.05)
Ringnorm40	0.03 ± 0.04	0.56 ± 0.06 (0.45, 0.15)
Yeast6	0.54 ± 0.13	0.61 ± 0.11 (0.3, 0)

reached at $\alpha = 0.25$ and $\beta = 0$. Another important aspect is that there are improvements for a wide range of α and β values close to the optimum, and that the increase in the beta factor (switching from minority to majority) tends to degrade the performance.

In our second analysis, we graphically compared the performance of the four rebalancing techniques proposed in this work on the `Balance-B` dataset (the selection of both `Balance-B` and `Ringnorm` datasets is merely based on visualization purposes).

The results are represented in Fig. 3, where the error bars indicate standard deviation. As a baseline, we used the ensemble with diversity by random weight initialization with no label switching ($\alpha = \beta = 0$; F$_1$-score = 0.15 ± 0.15). It was observed that the gating network produced a significant improvement in mid and high α rates ($0 \leq \alpha \leq 0.35$), worsening performance for more aggressive α values ($\alpha > 0.35$), while the population rebalancing `RB_SW` (using SMOTE with $Q_{R_p} = 1$) resulted in improvements in the entire practical range of α ($0 \leq \alpha < 0.5$).

Finally, in Tables 3 and 4, we show the results of all tested methods on the datasets considered. It is observed that asymmetric switching achieves a notable increase in performance with respect to the Baseline ensemble (with diversity only by random weight initialization) (see Table 3).

Confidence intervals in bold in Table 4 are those in which the rebalancing methods achieve statistically consistent improvements (the difference in average is at least equal to the semi-sum of the standard deviation) with respect to the Baseline. It is observed that, in general, they slightly outperform the asymmetric label switching, but there is no clear winning method. To support this, we performed a statistical analysis of the performance of the methods across the 17 datasets considered. In particular, we used the Python package `autorank` [39], which yielded the results indicated in Table 5.

Another important aspect is that SMOTE and cost weighting techniques are powerful rebalancing methods (their performances correspond to the $\alpha = \beta = 0$ case). However, their effectiveness can be improved by combining them with asymmetric label switching. It is also important to point out that both mechanisms (rebalance and switching) can interfere with each other, and not always in a constructive manner. Evidence of this interference in learning is that the optimum rebalance factors are slightly smaller than the original Imbalance Ratio when combined with asymmetric label switching. For example, we can see in

Table 4
F$_1$-score. The SW_Gate results indicated with * are those in which the gating network is equivalent to averaging. Please note that Q_{R_c} (rebalancing cost policy) and Q_{R_p} (rebalanced population ratio) are those introduced in Section 2.

Datasets	Cost_SW (α, β, Q_{R_c})	SW_Gate (α, β)	RB_SW (α, β, Q_{R_p})
EcoliImU	**0.72** ± **0.09** (0.35, 0.09, 6)	0.68 ± 0.05 (0.25, 0.05)*	0.69 ± 0.10 (0.45, 0.1, 4)
Satimage4	**0.63** ±**0.03** (0.25, 0, 8)	**0.63** ± **0.03** (0.1, 0)	**0.63** ± **0.03** (0.25, 0, 7)
Abalone7	0.44 ± 0.03 (0.35, 0, 5)	0.44 ± 0.02 (0.35, 0)	**0.45** ± **0.03** **(0.35, 0, 8)**
Ringnorm10	0.81 ± 0.05 (0.25, 0, 8)	**0.82** ± **0.04** (0.3, 0.05)	0.82 ± 0.06 (0.45, 0.1, 6)
BalanceB	0.41 ± 0.16 (0.35, 0, 10)	0.61 ± 0.15 (0.25, 0)	**0.64** ± **0.07** (0.25, 0, 1)
Ecoli4	**0.92** ± **0.08** (0.15, 0.05, 8)	0.88 ± 0.12 (0.15, 0.15)*	0.90 ± 0.08 (0.05, 0.1, 10)
Aba9vs18	0.62 ± 0.12 (0.3, 0, 14)	0.61 ± 0.12 (0.25, 0)*	0.62 ± 0.12 (0.35, 0, 8)
SolarflareM0	**0.26** ± **0.09** (0.35, 0, 16)	0.24 ± 0.07 (0.35, 0.05)*	0.24 ± 0.07 (0.4, 0.05, 16)
Ringnorm20	0.75 ± 0.06 (0.4, 0.05, 18)	**0.77** ± **0.04** (0.35, 0)	0.76 ± 0.06 (0.45, 0, 14)
Oil	0.63 ± 0.10 (0.1, 0, 12)	0.62 ± 0.12 (0.15, 0)	0.65 ± 0.10 (0.2, 0.05, 18)
Flare-F	**0.36** ± **0.06** (0.4, 0.05, 20)	0.34 ± 0.08 (0.4, 0)	0.36 ± 0.11 (0.4, 0.05, 20)
Winequal4	0.27 ± 0.04 (0.4, 0, 12)	0.28 ± 0.06 (0.4, 0)	**0.30** ± **0.06** (0.45, 0, 22)
LetterimgZ	**0.95** ±**0.01** (0.05, 0, 20)	0.94 ± 0.01 (0.05, 0)	**0.95** ± **0.01** (0.05, 0, 16)
Yeast4	0.44 ± 0.11 (0.35, 0, 24)	**0.47** ± **0.08** (0.35, 0)	0.45 ± 0.06 (0.4, 0, 24)
Aba17	**0.48** ±**0.08** (0.45, 0.05, 32)	0.46 ± 0.09 (0.4, 0.05)	0.46 ± 0.11 (0.45, 0.05, 32)
Ringnorm40	0.65 ± 0.08 (0.45, 0.05, 36)	0.62 ± 0.08 (0.4, 0)	**0.69** ± **0.06** (0.45, 0, 36)
Yeast6	0.60 ± 0.09 (0.3, 0, 28)	0.61 ± 0.11 (0.3, 0)*	**0.65** ± **0.08** (0.35, 0, 36)

Table 5
Comparison of the performance of rebalancing methods.

	Mean(F1)	Std(F1)	Conf. Interval	Cohen's D	Improvement
Baseline	0.396	0.270	[0.292, 0.501]	0.000	-
SW	0.564	0.202	[0.460, 0.669]	−0.705	medium
Cost_SW	0.585	0.208	[0.480, 0.689]	−0.782	medium
SW_Gate	0.589	0.201	[0.485, 0.694]	−0.812	large
RB_SW	0.604	0.202	[0.499, 0.708]	−0.869	large

Table 4 that for Aba17, Ringnorm40, or Yeast6 (all of them with IR $\simeq 40$), the optimum Q_{R_C} and Q_{R_P} factors are between 28 and 36.

7. Conclusions and future lines

It is relatively common for classification machines to handle practical scenarios that present imbalanced class populations. For these scenarios, we analyzed different methods that partially rebalance the original problem, allowing us to increase the effectiveness of machine learning and consequently improve the final performance with respect to conventional models. Our starting point was the Bayesian formulation for optimal (binary) classification and the use of an ensemble of machines trained with a Bregman divergence as a surrogate loss function, which provides a consistent estimate of the posterior class probabilities.

The analyzed methods fall into three categories: At the data-level, neutral rebalancing methods, that is, those that do not modify the form of the class likelihoods; at the algorithmic-level, the use of cost policies in training; and at the ensemble-level, the use of asymmetric label switching as the source of diversity and weighted aggregation of base learner's outputs. These methods transform the original problem by introducing a rebalancing effect that facilitates base learner training. Transformed problems require new decision thresholds, which have been specified in an optimal Bayesian sense.

In the experimentation part, we considered these principled neutral rebalancing mechanisms separately to better understand their contribution to performance improvement. Obviously, it is reasonable to expect that the results listed in this work could be improved by jointly optimizing the architecture of the ensemble (number, type, and training of the base learners) and the global combination of methods, a possibility that remains to be explored in future studies.

Among the strengths of this study, we have proposed a Bayesian framework that allows the design of ensembles of classifiers that simultaneously combine different mechanisms to address imbalance classification problems. We leave open further research on the analysis of the adequate proportion of each of these mechanisms to optimize their performance in a particular classification problem.

Regarding weaknesses, we point out that the actual formulation is limited to binary classification problems. Currently, we have studied the application of label switching to the (still imbalanced) dichotomic problems derived in multi-class classification, but more research is needed. The analysis of other neutral rebalancing methods other than SMOTE is also pending, as well as the use of other gating network architectures not necessarily linear, and the reduction of the training complexity, especially for fixed-computational-budget scenarios.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

We thank the anonymous reviewers for their valuable suggestions and comments. This work is partially funded by Project PID2021-125652OB-I00 from the "Ministerio de Ciencia e Innovación" of Spain. Funding for APC: Universidad Carlos III de Madrid (Read & Publish Agreement CRUE-CSIC 2022). In memoriam: Prof. Aníbal R. Figueiras-Vidal (1950–2022)

References

[1] C. Zhao, Y. Xin, X. Li, Y. Yang, Y. Chen, A heterogeneous ensemble learning framework for spam detection in social networks with imbalanced data, Applied Sciences 10 (3) (2020) 936, doi:10.3390/app10030936.

[2] G. Karatas, O. Demir, O.K. Sahingoz, Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset, IEEE Access 8 (2020) 32150–32162, doi:10.1109/ACCESS.2020.2973219.

[3] H. Lu, Y. Xu, M. Ye, K. Yan, Z. Gao, Q. Jin, Learning misclassification costs for imbalanced classification on gene expression data, BMC Bioinformatics 20 (25) (2019) 1–10, doi:10.1186/s12859-019-3255-x.

[4] S. Fotouhi, S. Asadi, M.W. Kattan, A comprehensive data level analysis for cancer diagnosis on imbalanced data, J. of Biomedical Informatics 90 (2019) 103089, doi:10.1016/j.jbi.2018.12.003.

[5] T. Lee, K.B. Lee, C.O. Kim, Performance of machine learning algorithms for class-imbalanced process fault detection problems, IEEE Trans. on Semiconductor Manufacturing 29 (4) (2016) 436–445, doi:10.1109/TSM.2016.2602226.

[6] J. Sun, H. Li, H. Fujita, B. Fu, W. Ai, Class-imbalanced dynamic financial distress prediction based on adaboost-SVM ensemble combined with SMOTE and time weighting, Information Fusion 54 (2020) 128–144, doi:10.1016/j.inffus.2019.07.006.

[7] H. He, E.A. Garcia, Learning from imbalanced data, IEEE Trans. on Knowledge and Data Engineering 21 (9) (2009) 1263–1284, doi:10.1109/TKDE.2008.239.

[8] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, G. Bing, Learning from class-imbalanced data: review of methods and applications, Expert Syst Appl 73 (2017) 220–239, doi:10.1016/j.eswa.2016.12.035.

[9] A. Fernández, S. García, M. Galar, R.C. Prati, B. Krawczyk, F. Herrera, Learning from imbalanced data sets, Springer, Cham, 2018, doi:10.1007/978-3-319-98074-4.

[10] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: Synthetic minority over-sampling technique, J. of Artificial Intelligence Research 16 (2002) 321–357, doi:10.1613/jair.953.

[11] L.I. Kuncheva, Á. Arnaiz-González, J.-F. Díez-Pastor, I.A. Gunn, Instance selection improves geometric mean accuracy: a study on imbalanced data classification, Progress in Artificial Intelligence 8 (2) (2019) 215–228, doi:10.1007/s13748-019-00172-4.

[12] S. Maldonado, C. Vairetti, A. Fernández, F. Herrera, FW-SMOTE: A feature-weighted oversampling approach for imbalanced classification, Pattern Recognit 124 (2022) 108511, doi:10.1016/j.patcog.2021.108511.

[13] C. Jiménez-Castaño, A. Álvarez-Meza, A. Orozco-Gutiérrez, Enhanced automatic twin support vector machine for imbalanced data classification, Pattern Recognit 107 (2020) 107442, doi:10.1016/j.patcog.2020.107442.

[14] V. López, A. Fernández, J.G. Moreno-Torres, F. Herrera, Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification, Expert Syst Appl 39 (7) (2012) 6585–6608, doi:10.1016/j.eswa.2011.12.043.

[15] S.H. Khan, M. Hayat, M. Bennamoun, F.A. Sohel, R. Togneri, Cost-sensitive learning of deep feature representations from imbalanced data, IEEE Trans. on Neural Networks and Learning Systems 29 (8) (2017) 3573–3587, doi:10.1109/TNNLS.2017.2732482.

[16] L. Rokach, Ensemble-based classifiers, Artif Intell Rev 33 (1) (2010) 1–39, doi:10.1007/s10462-009-9124-7.

[17] J.F. Díez-Pastor, J.J. Rodríguez, C.I. García-Osorio, L.I. Kuncheva, Diversity techniques improve the performance of the best imbalance learning ensembles, Inf Sci (Ny) 325 (2015) 98–117, doi:10.1016/j.ins.2015.07.025.

[18] R. OBrien, H. Ishwaran, A random forests quantile classifier for class imbalanced data, Pattern Recognit 90 (2019) 232–249, doi:10.1016/j.patcog.2019.01.036.

[19] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, F. Herrera, A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches, IEEE Trans. on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 42 (4) (2012) 463–484, doi:10.1109/TSMCC.2011.2161285.

[20] T. Le, M.T. Vo, B. Vo, M.Y. Lee, S.W. Baik, A hybrid approach using oversampling technique and cost-sensitive learning for bankruptcy prediction, Complexity 2019 (2019), doi:10.1155/2019/8460934.

[21] K. Yang, Z. Yu, X. Wen, W. Cao, C.L.P. Chen, H.-S. Wong, J. You, Hybrid classifier ensemble for imbalanced data, IEEE Trans. on Neural Networks and Learning Systems 31 (4) (2020) 1387–1400, doi:10.1109/TNNLS.2019.2920246.

[22] L.M. Bregman, The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming, USSR Computational Mathematics 7 (3) (1967) 200–217, doi:10.1016/0041-5553(67)90040-7.

[23] A. Gutiérrez-López, F.-J. González-Serrano, A.R. Figueiras-Vidal, Asymmetric label switching resists binary imbalance, Information Fusion 60 (2020) 20–24, doi:10.1016/j.inffus.2020.02.004.

[24] H.L. Van Trees, in: Classical Detection and Estimation Theory, John Wiley & Sons, Ltd, 2001, pp. 19–165.

[25] J. Cid-Sueiro, J.I. Arribas, S. Urbán-Munoz, A.R. Figueiras-Vidal, Cost functions to estimate a posteriori probabilities in multiclass problems, IEEE Trans. on Neural Networks 10 (3) (1999) 645–656, doi:10.1109/72.761724.

[26] A. Banerjee, S. Merugu, I.S. Dhillon, J. Ghosh, J. Lafferty, Clustering with Bregman divergences, J. of Machine Learning Research 6 (58) (2005) 1705–1749.

[27] G. Douzas, F. Bacao, F. Last, Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE, Inf Sci (Ny) 465 (2018) 1–20, doi:10.1016/j.ins.2018.06.056.

[28] A. Benítez-Buenache, L. Álvarez-Pérez, V.J. Mathews, A.R. Figueiras-Vidal, Likelihood ratio equivalence and imbalanced binary classification, Expert Syst Appl 130 (2019) 84–96, doi:10.1016/j.eswa.2019.03.050.

[29] L.I. Kuncheva, Combining classifiers: soft computing solutions, in: Pattern recognition: From classical to modern approaches, World Scientific, 2001, pp. 427–451, doi:10.1142/9789812386533_0015.

[30] M.I. Jordan, R.A. Jacobs, Hierarchical mixtures of experts and the EM algorithm, Neural Comput 6 (2) (1994) 181–214, doi:10.1162/neco.1994.6.2.181.

[31] M.I. Jordan, L. Xu, Convergence results for the EM approach to mixtures of experts architectures, Neural Networks 8 (9) (1995) 1409–1431, doi:10.1016/0893-6080(95)00014-3.

[32] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, F. Herrera, KEEL data-mining software tool, J. of Multiple-Valued Logic & Soft Computing 17 (2011) 255–287.

[33] D. Dua, C. Graff, UCI Machine Learning Repository (2019) http://archive.ics.uci.edu/ml.

[34] Z. Ding, Diversified Ensemble Classifiers for Highly Imbalanced Data Learning and their Application in Bioinformatics, Georgia State University, 2011 Ph.D. thesis.

[35] L. Breiman, Randomizing outputs to increase prediction accuracy, Mach Learn 40 (3) (2000) 229–242, doi:10.1023/A:1007682208299.

[36] A. Luque, A. Carrasco, A. Martín, A. de las Heras, The impact of class imbalance in classification performance metrics based on the binary confusion matrix, Pattern Recognit 91 (2019) 216–231, doi:10.1016/j.patcog.2019.02.023.

[37] D. Chicco, G. Jurman, The advantages of the Matthews Correlation Coefficient (MCC) over F1 score and accuracy in binary classification evaluation, BMC Genomics 21 (1) (2020) 1–13, doi:10.1186/s12864-019-6413-7.

[38] V. Asirvadam, S. McLoone, G. Irwin, Memory efficient BFGS neural-network learning algorithms using MLP-network: a survey, in: Proceedings of the 2004 IEEE International Conference on Control Applications, 2004., volume 1, 2004, pp. 586–591Vol.1, doi:10.1109/CCA.2004.1387275.

[39] S. Herbold, Autorank: a python package for automated ranking of classifiers, J. of Open Source Software 5 (48) (2020) 2173, doi:10.21105/joss.02173.

Aitor Gutiérrez-López received the Telecommunications Engineering degree from Universidad Carlos III de Madrid, Spain, in 2018, where he is currently pursuing his Ph.D. in Telecommuncations Engineering. His current research interests are focused on new algorithms and their application to multimedia, signal processing and data analysis.

Francisco J. González-Serrano received the Ph.D. degree in Telecommunication Engineering from Universidad de Vigo, Spain, in 1997. Since 2000, he is Associate Professor of the Signal Theory and Communications Department, Universidad Carlos III de Madrid, Spain. His current research interests include sensor networks (particularly localization and tracking tasks), the application of signal processing and machine learning methods to encrypted data, and the design of adaptive architectures in prediction and anomaly detection problems.

Aníbal R. Figueiras-Vidal received the Telecommunication Engineer (Hons.) degree from Universidad Politécnica de Madrid, Madrid, Spain, in 1973, and the Doctor (Hons.) degree from Universidad Politécnica de Barcelona, Barcelona, Spain, in 1976. He also received Honoris Causa Doctor degrees from University of Vigo, Vigo, Spain, in 1999, and University San Pablo, Arequipa, Perú, in 2011. He was a Professor of Signal Theory and Communications Department with University Carlos III de Madrid, Madrid, Spain. He has authored or co-authored more than 300 journal and conference papers, and he has been the principal researcher in almost 100 research projects and contracts. Prof. Figueiras-Vidal was member of the Royal Academy of Engineering of Spain, being the President from 2007 to 2011. Sadly, Prof. Figueiras-Vidal passed away at the age of 71 while this paper was in preparation.