Chapter 2

Definition of Loss Functions for Learning from Imbalanced Data to Minimize Evaluation Metrics

Juan Miguel García-Gómez and Salvador Tortajada

Abstract

Most learning algorithms for classification use objective functions based on regularized and/or continuous versions of the 0-1 loss function. Moreover, the performance of the classification models is usually measured by means of the empirical error or misclassification rate. Nevertheless, neither those loss functions nor the empirical error is adequate for learning from imbalanced data. In these problems, the empirical error is uninformative about the performance of the classifier and the loss functions usually produce models that are shifted to the majority class. This study defines the loss function L_{BER} whose associated empirical risk is equal to the BER. Our results show that classifiers based on our L_{BER} loss function are optimal in terms of the BER evaluation metric. Furthermore, the boundaries of the classifiers were invariant to the imbalance ratio of the training dataset. The L_{BER}-based models outperformed the 0-1-based models and other algorithms for imbalanced data in terms of BER, regardless of the prevalence of the positive class. Finally, we demonstrate the equivalence of the loss function to the method of inverted prior probabilities, and we define the family of loss functions L_{WER} that is associated with any WER evaluation metric by the generalization of L_{BER}.

Key words Cost-sensitive learning, Imbalanced datasets, Machine learning, Loss function

Abbreviations

ACC	Accuracy
BER	Balanced Error Rate
CSL	Cost-Sensitive Learning
ERR	Error
ERR1	Error of the positive class (1-sensitivity)
FN	False Negative
FP	False Positive
TN	True Negative
TP	True Positive
WER	Weighted Error Rate

Carlos Fernández-Llatas and Juan Miguel García-Gómez (eds.), *Data Mining in Clinical Medicine*, Methods in Molecular Biology, vol. 1246, DOI 10.1007/978-1-4939-1985-7_2,

1 Introduction

Cost-Sensitive Learning (CSL) studies the problem of optimal learning with different types of loss [1]. It is based on the Bayesian decision theory that provides the procedure to perform optimal decision given a set of alternatives. CSL has been studied to solve learning from imbalanced datasets. Learning from imbalanced datasets is a difficult problem that is often found in real datasets and limits the performance and utility of predictive models when combined to other factors such as overlapping between classes. The current digitalization of massive data is uncovering this problem in multiple applications from different scopes, such as social media, biomedical data, massive sensorization, and quantum analytics. Moreover, incremental learning has to deal with changing prevalences of imbalanced datasets from which multi-center predictive analyses are required [2].

Chawla in [3] classified cost-sensitive learning within those solutions for learning from imbalanced data at algorithmic level. He compiled some advantages of using CSL for learning from imbalanced datasets. First, CSL is not encumbered by large sets of duplicated examples; second, CSL usually outperforms random resampling methods; and third, it is not always possible to apply other approaches such as smart sampling in data level algorithms. On the contrary, a general drawback of CSL is that it needs a cost-matrix to be known for different types of errors—or even examples—but this cost-matrix is usually unknown and some assumptions have to be made at design-time. Another characteristic of CSL is that it does not modify the class distribution of data the way re-sampling does, which can be considered an advantage or a drawback depending on the author or the application [3].

Breiman et al. [4] studied the connection among the distribution of training samples by class, the costs of mistakes on each class, and the placement of the decision threshold. Afterwards, Maloof [5] reviewed the connection of learning from imbalanced datasets and cost-sensitive learning. Specifically, he observed the same ROC curve when moving the decision threshold and adjusting the cost-matrix. Visa and Ralescu in [6] studied the concept learning in the presence of overlapping and imbalance in the training set and developed solutions based on fuzzy classifiers.

The conclusions of the AAAI-2000 workshop and ICML-2003 pointed out the relevance of designing classifiers which performs well across a wide range of costs and priors. The insensitiveness to class imbalance of learning algorithms may lead to better control of their behavior in critical applications and streaming data scenarios. Furthermore, He and García in [7] supported the proposition addressed by Provost in [8] to concentrate the research on the theoretical and empirical studies of how machine learning algorithms can deal most effectively with whatever data they are given.

Weiss in [9, 10] supported the idea that the use of error and accuracy lead to poor minority-class performance. Moreover, Weiss determined the utility of the area under the ROC curve such as a measure to assess overall classification performance, but useless to obtain pertinent information for the minority class. He suggested appropriate evaluation metrics that take rarity into account, such as the geometric mean and the F-measure. In conclusion, he pointed out the value of using appropriate evaluation metrics and cost-sensitive learning to address the evaluation of results and to guide the learning process, respectively.

Although recent literature focuses its attention on the characterization and use of evaluation metrics which are sensitive to the effect of imbalanced data, the evaluation metrics used in class imbalance problems have not been studied in terms of the loss function under the empirical risk that defines them. To our concern, this is the first time a loss function is defined to equal its associated empirical risk to an evaluation metric different from the empirical error. Furthermore, it is our objective to observe its optimal behavior in terms of the selected evaluation metric, illustrate its stability, and compare its performance to other approaches for learning from imbalanced datasets.

2 Theoretical Framework

A predictive model (or classifier in classification problems), $\hat{y} = f(\mathbf{x}, \alpha)$, is a function with parameters α that gives a decision from the discrete domain $\hat{y} \in \hat{y}$ defined by the supervisor, given the observation of a sample represented by $\mathbf{x} \in \hat{y}$.

In Bayesian decision theory, the loss (or cost) function $L(y, \hat{y})$ measures the consequence of deciding $\hat{y}$ given the sample $\mathbf{x}$ that actually belongs to class y. When a predictive model decides $\hat{y}$ after observing $\mathbf{x}$, it assumes a *conditional risk*,

$$R(\hat{y} \mid \mathbf{x}) = E_{y|\mathbf{x}}[L(y, \hat{y})]. \tag{1}$$

In the case of classification problems, (1) is the sum of the weighted loss over the space of possible classes,

$$R(\hat{y} \mid \mathbf{x}) = \sum_{y \in \hat{y}} L(y, \hat{y}) p(y \mid \mathbf{x}). \tag{2}$$

As a consequence, the prediction model assumes a *functional risk* that is equal to the expected conditional risk over the possible values of $\mathbf{x}$,

$$R(\alpha) = E_{\mathbf{x}}[R(\hat{y} \mid \mathbf{x})] \tag{3}$$

$$= E_{\mathbf{x}}[E_{y|\mathbf{x}}[L(y,\hat{y})]] \tag{4}$$

$$= \int_{\mathbf{x}} E_{y|\mathbf{x}}[L(y,\hat{y})]p(\mathbf{x})d\mathbf{x} \tag{5}$$

$$= \int_{\mathbf{x}} \sum_{y\in\hat{y}} L(y,\hat{y})p(y \mid \mathbf{x})p(\mathbf{x})d\mathbf{x}. \tag{6}$$

Equation 6 requires knowing the joint distribution $p(\mathrm{x},y) = p(y \mid \mathrm{x})p(\mathrm{x})$, which is not always possible. Hence, it is common to estimate an *empirical risk* by means of an observed sample $\hat{y} = \{(\mathbf{x}_i, y_i)\}, i = 1,\ldots,N, \mathbf{x} \in \hat{y}, y_i \in \hat{y}$,

$$R_{\hat{y}}(\alpha) = \frac{1}{N}\sum_{i=1}^{N} E_{y|\mathbf{x}}[L(y,\hat{y})] \tag{7}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\sum_{y\in\hat{y}} L(y,\hat{y})p(y \mid \mathbf{x}), \tag{8}$$

where $p(y_i \mid \mathrm{x}_i) = 1$ and $p(y_{j\neq i} \mid \mathrm{x}_i) = 0$ are assumed to be observed in supervised learning. Thus, the empirical risk can be calculated as

$$R_{\hat{y}}(\alpha) = \frac{1}{N}\sum_{i=1}^{N} L[y_i, f(\mathbf{x}_i,\alpha)]. \tag{9}$$

Furthermore, the evaluation metric that has historically been used in classification is the ERR, or its positive equivalent, accuracy. Nevertheless, when dealing with class imbalance problems, it is necessary to evaluate the performance using metrics that take into account the prevalence of the datasets. In this paper, we focus our attention on the Balanced Error Rate (BER) and on the Weighted Error Rate (WER) family to define their associated loss functions.

The evaluation of a predictive model implies the estimation of its performance in future samples by means of an evaluation metric. Ideally, this evaluation metric is the estimation of the empirical risk given an independent and representative set of test cases. For instance, when the loss function used for the evaluation is the 0-1 loss function, then the *functional risk* is the *generalization error* and the *empirical risk* is the *test error* (or their equivalents in terms of accuracy). Similarly, the L_{BER} loss function defined in Subheading 3 and the family of loss functions defined in Subheading 5 ensure the equality of their respective empirical risks with the BER and WER evaluation metrics, respectively.

Without loss of generality, we define the evaluation metrics for a two-class discrimination problem $\hat{y} = \{y_1, y_2\}$, with y_1 as the positive class and y_2 as the negative class. The problems with imbalanced data are usually defined such that the positive class is under represented (minority class) compared to the negative class (majority

class) [11]. Let the test sample be a sample of N cases, where n_1 cases are from class y_1 and n_2 cases are from class y_2. The *confusion matrix* of a predictive model takes the form:

	$\hat{y}_1$	$\hat{y}_2$	
y_1	n_{11} (TP)	n_{12} (FN)	n_1
y_2	n_{21} (FP)	n_{22} (TN)	n_2
	$\hat{n}_1$	$\hat{n}_2$	N

where n_{11} is the number of positive cases that are correctly classified (True Positive (TP)), and n_{21} is the number of negative cases that are misclassified (False Positive (FP), or type I errors). Similarly, n_{22} is the number of negative cases that are correctly classified (True Negative (TN)), and n_{12} is the number of positive cases that are misclassified (False Negative (FN), or type II errors). The evaluation metrics for a model with parameters can be defined in terms of the values from the confusion matrix:

1. Err3or (ERR)

$$\mathrm{ERR}(\alpha) = \frac{n_{12} + n_{21}}{N} \tag{10}$$

2. Error of the positive class (1-sensitivity) (ERR1)[1]

$$\mathrm{ERR}_1(\alpha) = \frac{n_{12}}{n_1} \tag{11}$$

3. Balanced Error Rate (BER)

$$\mathrm{BER}(\alpha) = \frac{1}{2}\left(\frac{n_{12}}{n_1} + \frac{n_{21}}{n_2}\right) \tag{12}$$

4. Weighted Error Rate (WER)

$$\mathrm{WER}(\alpha) = w\frac{n_{12}}{n_1} + (1-w)\frac{n_{21}}{n_2}, 0 \le w \le 1 \tag{13}$$

Observe that WER is a convex combination with parameter w that defines the family and the BER is the specific case of this family when $w = 1\hat{y}2$. Figure 1 shows the BER loss function with respect to the errors by class. It is worth noting that this function takes the value 0 when both errors are 0, 1 when both errors are 1 and the value 0.5 when one of them is 0 and the other is 1. We have used colored lines to highlight the results obtained by different imbalances.

[1] Similarly, the Error of the negative class is defined by $\mathrm{ERR}_2(\alpha) = \frac{n_{21}}{n_2}$.

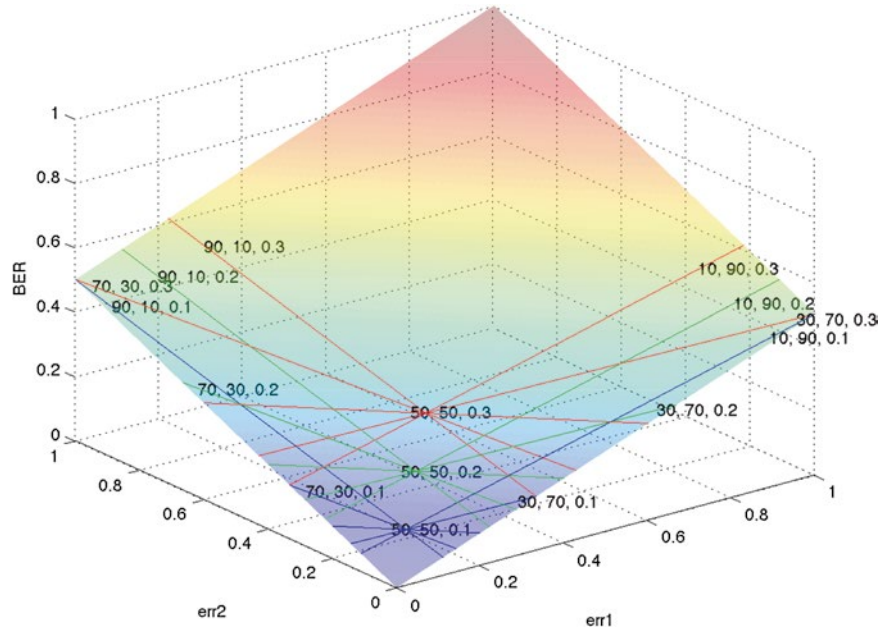

Fig. 1 BER loss function in relation to the error by class ($\mathrm{ERR}_1 = \frac{n_{12}}{n_1}$ and $\mathrm{ERR}_2 = \frac{n_{21}}{n_2}$). The *colored lines* correspond to the evaluation of class imbalance problems with the ratios [10,90], [30,70], [50,50], [70,30], and [90,10]. The labels are composed by three values: the first two show the prevalence of each class and the third shows the result of the evaluation in terms of ERR. The third label of each line is the ERR of the classifier

3 Definition of the L_{BER} Loss Function

Let L_{BER} be the loss function that defines the empirical risk that is equivalent to the BER evaluation metric:

$$L_{\mathrm{BER}}(y,\hat{y}) = \frac{N}{n_y \mid \hat{y} \mid}(1-\delta(y,\hat{y})), \tag{14}$$

where N is the number of cases, from which n_y is the number of cases of class y and

$$\delta(y,\hat{y}) = \begin{cases} 1, & \text{if} \quad y = \hat{y}, \\ 0, & \text{if} \quad y \neq \hat{y}. \end{cases} \tag{15}$$

For reasons of clarity, let us focus our demonstration on a classification problem with $\mid \hat{y} \mid = 2$, such that (14) can be specified as a 2×2 loss-matrix:

	$\hat{y}_1$	$\hat{y}_2$
y_1	0	$\frac{N}{2n_1}$
y_2	$\frac{N}{2n_2}$	0

By substitution of (14) in (9), we can demonstrate that its empirical risk is equal to the evaluation metric BER(α) (12):

$$
\begin{aligned}
R_{\hat{y}}(\alpha) &= \frac{1}{N}\sum_{N}^{i=1} L_{\mathrm{BER}}[y_i, f(\mathbf{x}_i,\alpha)] \\
&= \frac{1}{N}\sum_{N}^{i=1} \frac{N}{2n_{y_i}}(1-\delta(y_i, f(\mathbf{x}_i,\alpha))) \\
&= \frac{N}{2N}\sum_{N}^{i=1} \frac{1}{n_{y_i}}(1-\delta(y_i, f(\mathbf{x}_i,\alpha))) \\
&= \frac{1}{2}\left(\sum_{i:y_i=y_1} \frac{1}{n_1}(1-\delta(y_1, f(\mathbf{x}_i,\alpha))) \right. \\
&\quad \left. + \sum_{i:y_i=y_2} \frac{1}{n_2}(1-\delta(y_2, f(\mathbf{x}_i,\alpha))) \right) \\
&= \frac{1}{2}\left(\frac{1}{n_1}\sum_{i:y_i=y_1} (1-\delta(y_1, f(\mathbf{x}_i,\alpha))) \right. \\
&\quad \left. + \frac{1}{n_2}\sum_{i:y_i=y_2} (1-\delta(y_2, f(\mathbf{x}_i,\alpha))) \right) \\
&= \frac{1}{2}\left(\frac{1}{n_1} n_{12} + \frac{1}{n_2} n_{21} \right) = \mathrm{BER}(\alpha).
\end{aligned}
$$

4 Experiments

The following experiments are designed to (1) observe the behavior of the classifiers based on the L_{BER} from imbalanced datasets when varying the overlapping of the classes, (2) observe the sensitiveness or stability of the boundaries obtained by L_{BER} for different class imbalances, and (3) compare the performance of the *LBER-based* classifiers with SMOTE [12], which is a reference method for learning from imbalanced datasets using oversampling. Finally, we report the performance of predictive models based on the L_{BER} loss function in several real discrimination problems.

For our experiments with synthetic data, we studied two-class classification problems based on a $\mathbb{R}^2$ input space. Hence, we generated datasets following prior distributions and bidimensional Gaussian distributions that were parameterized for each experiment.

We compared classifiers based on the 0-1 loss function (c01) with those defined by the L_{BER} loss function (cLBER) which minimizes the conditional risk given the observation of **x**,

$$\hat{y}^* \leftarrow \arg\min_{\hat{y} \in \hat{y}} R(\hat{y} \mid \mathbf{x}). \tag{16}$$

Specifically, we compare our cost-sensitive learning classifiers based on L_{BER} with classical Gaussian classifiers based on generative models with free covariate matrices.

4.1 Behavior of L_{BER} When Varying the Overlapping Between Classes

The first experiment compared L_{BER}-based classifiers (cLBER) with the Gaussian classifier (c01) after training with imbalanced datasets in terms of ERR (10), BER (12), and ERR_1 (11). As pointed out by [3], the effect of the overlapping between classes is amplified when dealing with imbalanced problems. Hence, we studied the performance of the classifiers with respect to the overlapping ratio between classes.

We introduced a parameter D to control the overlapping between classes. Varying D from 1 to 5, we randomly generated a training sample composed by $N = 10{,}000$ cases with 10% of cases from class y_1 and the rest from class y_2 using the following distributions,

- Class 1, $p(y_1) = 0.1$, $p(x \mid y_1) = \hat{y}(\mu_1, \Sigma_1)$, $\mu_1 = (0,0)^T$, $\Sigma_1 = I$,
- Class 2, $p(y_2) = 0.9$, $p(x \mid y_2) = \hat{y}(\mu_2, \Sigma_2)$, $\mu_2 = (0, D)^T$, $\Sigma_2 = I$.

As can be observed, the parameter D controls the overlapping between the classes. Additionally, we randomly generated test samples composed by other $N = 10{,}000$ cases from the same distribution.

Figure 2 shows the results of the experiment. In general, cLBER always outperforms c01 in terms of BER, whereas c01 outperforms cLBER in terms of ERR. This shows the optimization that the L_{BER} loss function produces in terms of the evaluation metric BER. It is worth noting that for the cLBER classifiers the ERR and BER lines are equal.

This is due to the equilibrium that L_{BER} produces in the number of false negative and false positive cases. Meanwhile, c01 classifiers tend to classify most of the new cases as the majority class, obtaining different ERR and BER lines as a result. As expected, the evaluations of the two approaches and the two metrics converge when the overlapping decreases. Nevertheless, it is more interesting to observe that the discrepancy of the ERR and the BER for the c01

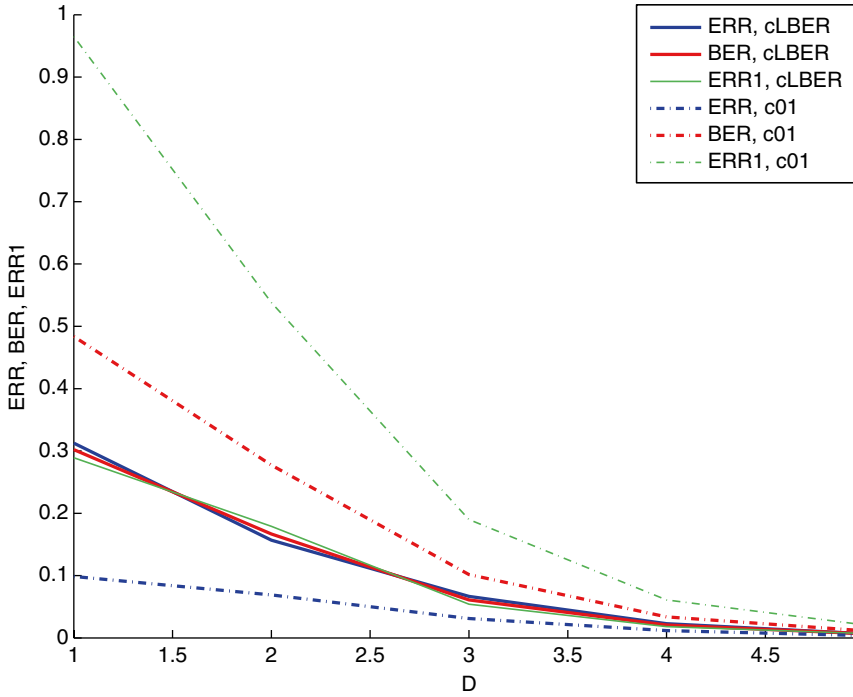

Fig. 2 ERR, BER, and ERR_1 of the cLBER and c01 classifiers with respect to D (grade of overlapping). cLBER classifiers obtain optimal results in terms of BER and stability between ERR and BER. The good behavior of cLBER classifiers is due to the control of ERR1 obtained with the L_{BER} loss function

classifiers increases with the overlapping. However, the ERR and the BER of the cLBER classifiers stay the same. This behavior is mainly explained by ERR_1, whereas the behavior is balanced for the L_{BER}-based classifiers, it is extremely high for the c01 classifiers.

4.2 Stability of the Boundaries When Varying the Class Imbalance

In the second experiment, we studied the behavior of the decision boundary of the L_{LBER}-based classifiers when varying the imbalance ratio. We generated 10^6 samples from the following distributions,

- Class 1, $p(y_1)=p$, $p(x \mid y_1)=\hat{y}(\mu_1,\Sigma_1)$, $\mu_1=(0,0)^T$, $\Sigma_1=I$
- Class 2, $p(y_2)=1-p$, $p(x \mid y_1)=\hat{y}(\mu_1,\Sigma_1)$, $\mu_2=(0,2)^T$, $\Sigma_2=I$,

where the prior probability of the positive class (y_1) took the values $[0.01, 0.1, 0.3, 0.5]$. We compared our results with those obtained when using c01 classifiers.

Figure 3 shows the bidimensional space with the 1,000 cases following the previous distribution when $p=0.1$. The boundaries obtained by the cLBER classifier are represented by thick lines, whereas the boundaries obtained by the c01 classifier are represented by thin lines. This figure clearly shows the stability of the cLBER

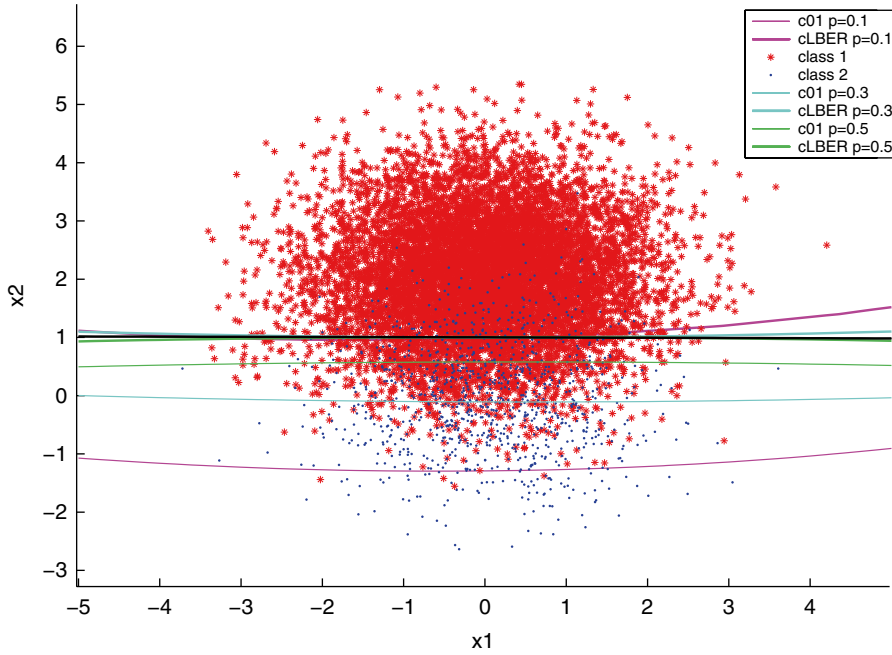

Fig. 3 Decision boundaries obtained by the cLBER classifiers (*thick lines*) and by the c01 classifiers (*thin lines*). The stability of the boundaries shows that our approach is invariant to the imbalance of the training sample and that the location of the boundary can be controlled by the loss function

with respect to the variation of the number of samples used for training. Moreover, the boundaries obtained by the cLBER classifier correspond to the boundary of the c01 classifier that is trained with a balanced dataset ($p=0.5$). This result shows that our approach is invariant to the imbalance of the training sample and that the location of the boundary can be controlled by the loss function.

4.3 Performance of L_{BER} Classifiers Compared to SMOTE

After characterizing our approach in terms of optimality and stability, we are interested in comparing its behavior with other approaches for learning from imbalanced datasets. SMOTE is a well-known algorithm that deals with imbalanced datasets by applying a synthetic minority oversampling. Specifically, we have used the implementation of SMOTE by Manohar at MathWorks based on [12] with the default parameters, which also performs a random subsampling of the majority class.

A characteristic effect of SMOTE is the local directionality of the samples in the oversampling distribution.

In this experiment, we compared the performance of our L_{BER} approach with SMOTE and c01 classifiers in terms of ERR, BER, and ERR_1. The learning process after applying SMOTE was the estimation of the Gaussian distributions, which is similar to the process for c01 classifiers. We were interested in seeing the stability of the performance when varying the imbalance ratios, i.e.,

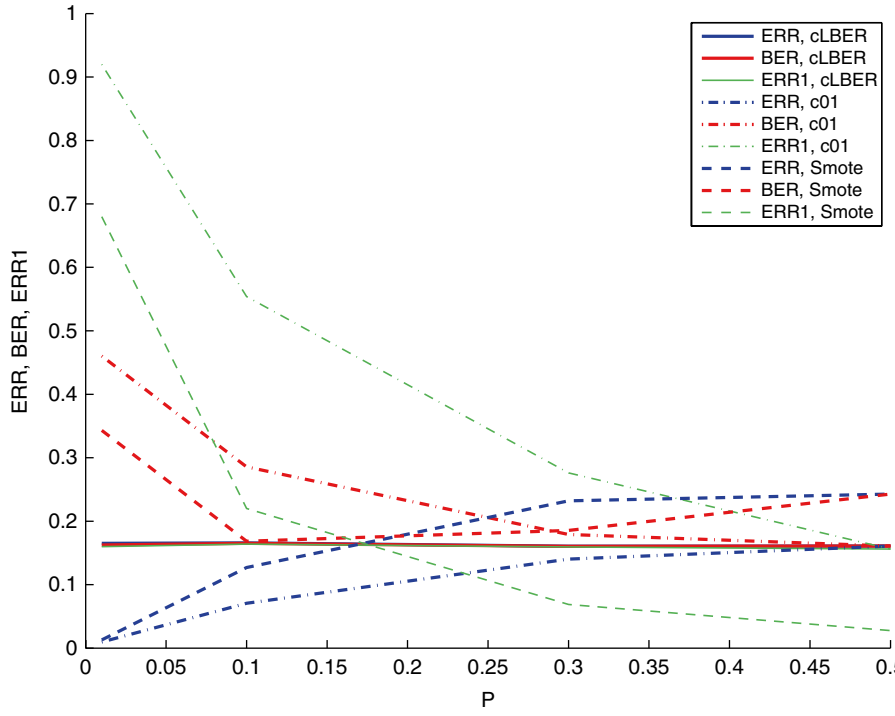

Fig. 4 ERR, BER, and ERR_1 of the cLBER, c01, and SMOTE classifiers with respect to the imbalance ratio (P). The L_{BER} is stable and invariant to the imbalance ratio in terms of ERR, BER, and ERR_1 (*solid lines*) in contrast to SMOTE, which shows a low performance for extreme and low imbalance datasets

$p=[0.01, 0.1, 0.3, 0.5]$. In order to complement our previous results, we used the same distribution as the one in Subheading 4.1 but fixing $D=2$. Figure 4 shows the results of the experiment. Both cBER and SMOTE overperformed the c01 classifiers in terms of BER and ERR_1 for moderate ($p=[0.1, 0.3[$) and extreme ($p=[0.01, 0.1[$) imbalance ratios.

The most important result of this experiment is the stability of the cLBER classifiers to changes in the imbalance ratio. In fact, the L_{BER} approach obtained constant values in the three evaluation metrics, ERR, BER, and ERR_1 (solid lines) that are directly relative to the overlapping between the distributions. This good result contrasts with the behavior obtained by SMOTE. In terms of ERR_1 (the green dashed line), the SMOTE algorithm is able to compensate moderate imbalances ($p=[0.1, 0.3[$), but it fails for extreme imbalances ($p=[0.01, 0.1[$) and low imbalances ($p=[0.3, 0.5]$). This results in a BER function (the red dashed line) with a minimum at $p=0.1$ but with worse behavior for extreme and low imbalances. Moreover, when approaching extreme imbalances, the slope of the BER function is high. As in the first experiment, we consider ERR (the blue lines) not to be

Table 1
Computational time of the c01, SMOTE, and cLBER approaches. SMOTE is computationally costly in comparison with cLBER approach. This is due to its re-sampling strategy, whereas our CSL approach is based on the modification of the conditional risk, which minimizes the learning algorithm

Training	Computational time (s)
Gaussian 0-1 (c01)	0.0013
Smote + Gaussian 0-1 (Smote)	19.0955
Gaussian + L_{BER} (cLBER)	0.0016

significant of the study the performance of the classifiers, especially for extreme imbalances.

We also studied the computational time required by each approach to perform the learning process of a classifier. Specifically, we measured the time of the learning processes of the c01, SMOTE, and cLBER classifiers when the imbalance is $p = 0.1$ on a 1.8 GHz dual-core Intel Core i5 with Mac OS X v10.7.5 and Matlab R2011a 64 bit. The results presented in Table 1 clearly show a significant difference between the SMOTE algorithm and the L_{BER} approach. This is due to the different strategy of each approach. Whereas SMOTE is a re-sampling method that involves generating a new dataset, L_{BER} implies the estimation of fix number of parameters to be used in the decision process or during a minimization process.

4.4 Performance of L_{BER} Classifiers in Real Datasets

We trained predictive models based on the L_{BER} loss function for three real datasets. One of them is the reference Contraceptive dataset from UCI [13], whereas the other two are biomedical datasets previously studied by machine-learning techniques: Brain Tumor [14, 15, 16, 17] and Postpartum depression [18]. All of them are two-class datasets. The Contraceptive dataset contains 1,473 cases ($p_1 = 0.427$) and 9 variables. The Brain Tumor dataset includes 571 cases ($p_1 = 0.257$) and 15 variables, and the Postpartum dataset is composed by 1,008 cases ($p_1 = 0.128$) represented by 19 variables. The predictive models for Contraceptive and Brain were trained with different subsets to study the response of the L_{BER} for different prevalences. Finally, we report the evaluation of the predictive models for the Postpartum depression dataset. The evaluation results include the evaluation metrics ERR, BER, and ERR_1 which were estimated by a bootstrap strategy with 200 repetitions. For the Contraceptive and the Brain Tumor datasets, we repeated the bootstrap estimation for each prevalence ten times, in order to avoid spurious results from specific subsets.

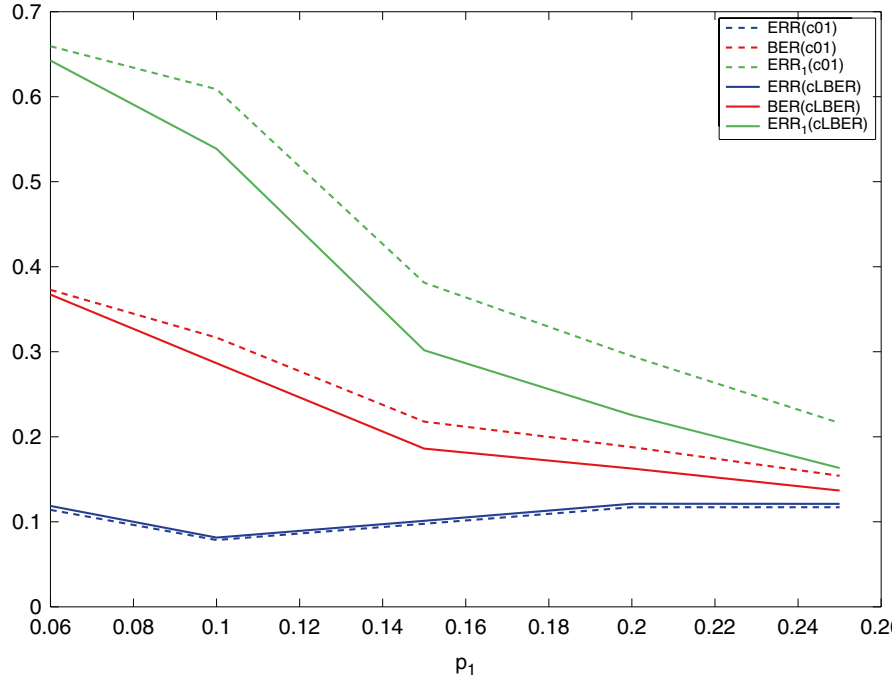

Fig. 5 BER, BER, and ERR_1 of the Brain Tumor problem obtained by the cLBER and c01 predictive models trained with datasets with different $p(y_1)$. The cLBER models outperform the c01 models in terms of BER and ERR1 independently of the prevalence

Figure 5 shows the BER, BER, and ERR_1 of the Brain Tumor problem obtained by the cLBER and c01 predictive models trained with datasets with different prevalences of the positive class ($[0.06, 0.10, 0.15, 0.20, 0.25]$). Table 2 shows the relative improvement in the Brain Tumor problem obtained by the predictive models based on the LBER loss function with respect to the 0-1 loss function in terms of BER and ERR_1.

Figure 6 shows the *BER*, *LBER*, and ERR_1 of the Contraceptive problem obtained by the cLBER and c01 predictive models trained with datasets with different prevalences of the positive class ($[0.05, 0.10, 0.15, 0.2, 0.3, 0.4]$). Table 3 shows the relative improvement in the Contraceptive problem obtained by the predictive models based on the LBER loss function with respect to the 0-1 loss function in terms of BER and ERR_1.

For the Postpartum Depression dataset, we prepared a predictive model with the full dataset $p(y_1) = 0.0128$ based on LBER, and we compared its performance with 0-1-based models. BER improved 5.6 % and ERR_1 improved 25.8 % using cLBER models with respect to the c01 models.

Table 2
Relative improvement of the *cBER* models in the Brain Tumor problem. The greatest improvement in terms of BER is obtained for a prevalence of $p(y_1) = 0.15$

$p(y_1)$	0.06	0.10	0.15	0.20	0.25
(BER(c01)-BER(cLBER))/BER(c01)	0.015	0.095	0.146	0.134	0.112
(ERR1(c01)-ERR1(cLBER))/ERR1(c01)	0.0256	0.115	0.208	0.235	0.244

ε

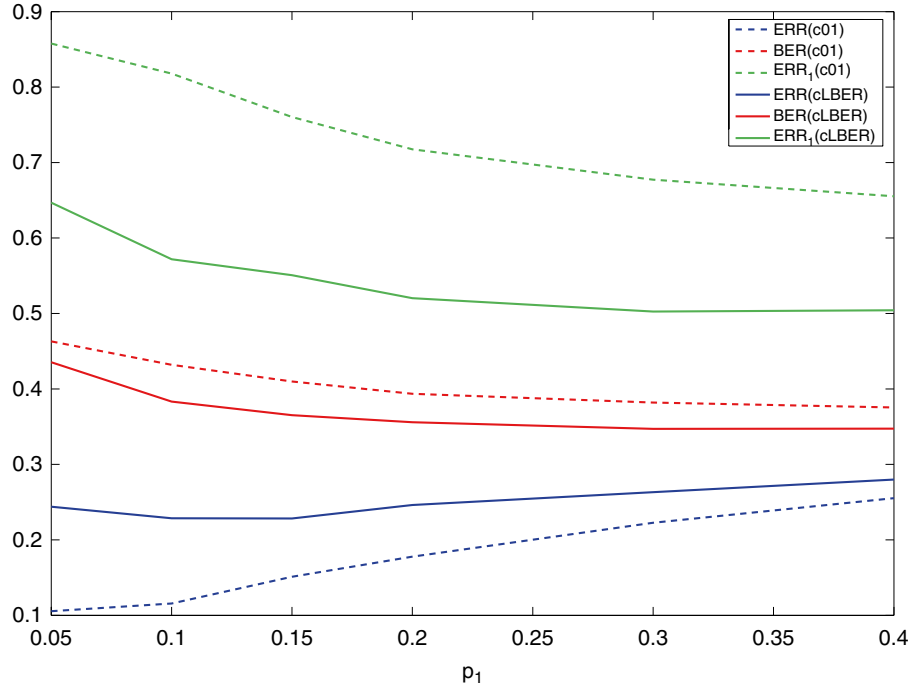

Fig. 6 BER, LBER, and ERR_1 of the Contraceptive problem obtained by the cLBER and c01 predictive models trained with datasets with different $p(y_1)$. cLBER models overperform c01 models in terms of BER and ERR1 independently of the prevalence

5 Discussion

Our experiments have shown that L_{BER} is a loss function that minimizes an empirical risk that is equal to the BER evaluation metric. As a result, the cost-sensitive classifiers based on the L_{BER} loss-function obtain the best performance in terms of their associated evaluation metric, which is BER. Furthermore, learning with L_{BER} has the theoretical property

Table 3
Relative improvement of the cBER models in the Contraceptive problem. The highest improvement in terms of BER is obtained for a prevalence of $p(y_1) = 0.10$

$p(y_1)$	0.05	0.1	0.15	0.2	0.3	0.4
(BER(c01)-BER(cLBER))/BER(c01)	0.0595	0.1128	0.1090	0.0960	0.0913	0.0746
(ERR1(c01)-ERR1(cLBER))/ERR1(c01)	0.2459	0.3008	0.2756	0.2751	0.2581	0.2307

of being insensitive to the imbalance of the datasets in terms of the challenges summarized by Chawla et al. [3]. Specifically, for the Gaussian models used in our experiments, the different values of the loss function cause a shift in the decision boundary.

As the results of the first experiment in Subheading 4.1 show, the classifiers based on the L_{BER} outperformed those based on the 0-1 loss functions in terms of BER. This result is compatible with the fact that the L_{BER}-based classifiers minimize a risk equivalent that is to the BER evaluation metric, whereas the 0-1 classifiers minimize the empirical error. The extension of this result to the L_{WER} family may allow designers to adapt their classifiers to the desired behavior in terms of their final results for class imbalance problems.

The bias (or independent term) of the boundaries of the *LBER-based* classifiers are independent of the prevalence of the classes in the training dataset. This fact can be easily observed in the results of our second experiment, in contrast to the behavior of the Gaussian classifier. To demonstrate this behavior, we can consider each term of the empirical risk of a classifier based on a generative model, where the product of $\frac{N}{n_y \mid \hat{y} \mid}$ and the posterior probability $p(y|\mathbf{x})$ (using the Bayes theorem), obtains the result $\frac{p(\mathbf{x} \mid y)}{\mid \hat{y} \mid p(\mathbf{x})}$, independent of the prior probability. Hence, an L_{BER} classifier that is trained with an imbalanced dataset is equivalent to a 0-1 classifier with equal prior probabilities.

The results of the third experiment in Subheading 4.3 demonstrate the good behavior of the approach in comparison with SMOTE. Moreover, our CSL approach has three advantages over SMOTE: (1) the computational time is significantly shorter; (2) the performance of the cLBER classifiers is insensitive to the imbalance of the classes; and (3) the estimation of the distributions (e.g., for generative models) are not disturbed by the assumptions

of the over-sampling or under-sampling procedures that modify the training sample.

In real datasets, as expected, the performance of the classifiers is dependent on the dataset; nevertheless, several features can be observed in all of the experiments. BER(cLBER) is equal to or less than BER(c01) in every dataset and every prevalence. ERR_1(cLBER) is lower than ERR_1(c01) in all the experiments. In return, ERR(cLBER) is worse than BER(c01); this is consistent with the trade-off effect produced by the BER metric.

The mean relative improvement in our experiments in terms of BER is about 10%. The best improvement in terms of LBER is obtained when the prevalence p_1 is in the interval $[0.10 \ldots 0.15]$. When the prevalence p_1 of the positive class is very small ($[0.05 \ldots 0.06]$), the improvement in terms of BER is small. Moreover, the improvement in terms of ERR_1 for Brain Tumor is also small. The cause is twofold. First, the improvement in ERR_1 is due to a great decrease in ERR_2; hence, the mean of both errors obtains a small improvement. Second, the y_1 class is not correctly represented by the small number of cases; hence, it is worse represented.

In our results with real datasets, the stability of LBER decreases with respect to the results obtained with synthetic data. This can be due to the limitation of the Gaussian models when applied to real problems. Nevertheless, our results demonstrate the improvement of the LBER approach in terms of BER for low to moderate class imbalance problems.

5.1 The L_{WER} Loss Function Family

We can generalize the L_{BER} loss function to define the L_{WER} loss function family that defines the empirical risk equivalent to WER for a given vector of weights **w**, such that $\sum_{y \in \hat{y}} w_y = 1$:

$$L_{WER}(y, \hat{y}) = w_y \frac{N}{n_y}(1 - \delta(y, \hat{y})), \tag{17}$$

where $\delta(y, \hat{y})$ is defined in (15). When $|\hat{y}| = 2$, we can establish $w_{y_1} = w$ y $w_{y_2} = 1 - w$, so the previous expression can be written as

	$\hat{y}_1$	$\hat{y}_2$
y_1	0	$w\frac{N}{n_1}$
y_2	$(1-w)\frac{N}{n_2}$	0

One more time, by substitution of (17) in (9):

$$R_{\hat{y}}(\alpha) = \frac{1}{N}\sum_{i=1}^{N} L_{WER}[y_i, f(\mathbf{x}_i, \alpha)]$$
$$= w\frac{n_{12}}{n_1} + (1-w)\frac{n_{21}}{n_2} = WER(\alpha),$$

so we demonstrate that the empirical risk given by the L_{WER} loss function (17) is equal to WER(α) (13).

5.2 Relation to Previous Studies

We have observed some connections of our methodology with previous studies about learning from imbalanced datasets by means of cost-sensitive learning. The Library for Support Vector Machines implemented by Chang and Lin in [19] implements a similar effect than our approach for training SVMs by assigning different soft-margin constants for the positive C_1 and negative C_2 cases [20, 21]. Specifically, similar effect to our proposal can be obtained choosing the ratio $\frac{C_1}{C_2} = \frac{n_1}{n_2}$ between constants. Nevertheless, compared to the Chang and Lin implementation, our approach is directly applicable to any approach to solve decision problems and to multi-class problems.

Raskutti and Kowalczyk in [11] investigated two methods of extreme imbalance compensation for SVM, one of which was based on cost-sensitive learning. They proposed a weight balancing through the regularization constants for the minority and majority class data. Their function is equivalent to our L_{BER} loss function when $B = 0$ and $C = N$. Nevertheless, they do not explain which risk function is minimized by the learning process, or the effect of their approach in terms of the risk function. Instead, they explain the effect in terms of the ROC curve.

One of the interesting conclusions obtained in [11] is the suitability of positive one-class classification for extremely imbalanced datasets. For this result, they observed that the performance of the classifiers was slightly better at higher values of C, which is connected to large L_{BER} loss for false negative cases when the training size N is very large.

Thai-Nghe et al. in [22] proposed two empirical methods to learn SVM from imbalanced datasets, one of them by optimizing the cost ratio. Their method is able to minimize user-selected evaluation metrics by means of a grid search where the cost ratio between positive and negative classes is a hyperparameter that must be adjusted. On the one hand, the advantage of their approach is that it can be used for any classification method. On the other hand, the disadvantages of the method are the need for a tuning dataset, the computational cost, and the sensitivity to the imbalance ratio that was demonstrated in their results.

6 Conclusions

We have defined the L_{BER} loss function to make the empirical risk of a classifier equal to the BER evaluation metric. To our concern, this is the first time a loss function is defined analytically to equal its associated empirical risk to an evaluation metric. The training of classifiers based on this loss function with imbalanced datasets is equivalent to the training of those based on the 0-1 function when they are trained with balanced data. The same concept has also been generalized to the L_{WER} loss function family. Our results, which are based on synthetic data, show that our approach obtains the optimal performance of the classifiers in terms of the evaluation metric associated with the loss metrics. Moreover, we have observed a trend to the stability of the classifiers with respect to the imbalance of the dataset. The approach is computationally efficient and allows the use of the available data. The classifiers based on the L_{BER} loss function outperformed the classifiers based on the 0-1 loss function in all our experiments with real data. Finally, we have also discussed some interesting properties of the loss function derived by its definition and its use in the calculation of conditional risks.

In further work, we plan to extend the approach to other evaluation metrics and introduce the loss function into other families of classification techniques, and study the stability of the classifiers with respect to the imbalance ratios. Our final objective is to incorporate the methodology in incremental learning frameworks for biomedical streaming data and multi-center repositories.

7 Acknowledgements

We thank Carlos Sáez and the rest of the IBIME group for their interesting discussions about biomedical problems and the need for clinical decision support systems that justify the application of this work to real environments. The work presented in this paper is funded by the Spanish grant *Modelo semántico y algoritmos de Data Mining aplicados al tratamiento del Cáncer de Mama en centros de Atención Especializada* (IPT-2011-1126-900000), Ministerio de Ciencia e Innovación (INNPACTO 2011), the Spanish and EU grant *Servicio remoto de atención sanitaria basado en la prevención, autonomía y autocontrol de los pacientes* (IPT-2011-1087-900000), Ministerio de Ciencia e Innovación (INNPACTO 2011) and FEDER (Fondo Europeo de Desarrollo Regional), and the EU grant *Help4Mood: A Computational Distributed System to Support the Treatment of Patients with Major Depression* (FP7-ICT-2009-4; 248765), European Commission (Seventh Framework Program).

References

1. Elkan C (2001) In: Proceedings of the seventeenth international joint conference on artificial intelligence, pp 973–978
2. Quiñonero-Candela J, Sugiyama M, Schwaighofer A (2009) Dataset shift in machine learning. MIT Press, Cambridge
3. Chawla NV, Japkowicz N, Kotcz A (2004) SIGKDD Explor Newslett 6(1):1. doi:10.1145/1007730.1007733. http://doi.acm.org/10.1145/1007730.1007733
4. Breiman L, Stone CJ, Friedman JH, Olshen RA (1984) Classification and regression trees. Chapman & Hall, New York
5. Maloof MA (2003) In: ICML-2003 workshop on learning from imbalanced data sets II
6. Visa S, Ralescu A (2003) Learning imbalanced and overlapping classes using fuzzy sets. University of Ottawa, Washington
7. He H, García E (2009) IEEE Trans Knowl Data Eng 21(9):1263. http://dx.doi.org/10.1109/TKDE.2008.239
8. Provost F (2000) In: Proceedings of the learning from imbalanced datasets: papers from the American Association for Artificial Intelligence workshop
9. Weiss GM, Provost F (2003) J Artif Intell Res 19:315
10. Weiss GM (2004) SIGKDD Explor Newslett 6(1):7. doi:10.1145/1007730.1007734. http://doi.acm.org/10.1145/1007730.1007734
11. Raskutti B, Kowalczyk A (2004) ACM Sigkdd Explor Newslett 6(1):60. http://dl.acm.org/citation.cfm?id=1007739
12. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) J Artif Intell Res 16(1):321. http://dl.acm.org/citation.cfm?id=1622407.1622416
13. Lim T-S, Loh W-Y, Shih Y-S (2000) A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. Mach Learn 40(3):203–228. doi:10.1023/A:1007608224229
14. García-Gómez JM, Tortajada S, Vidal C, Julià-Sape M, Luts J, Moreno-Torres À, Van Huffel S, Arus C, Robles M (2008) NMR Biomed 21(10):1112. doi:10.1002/nbm.1288. http://onlinelibrary.wiley.com/doi/10.1002/nbm.1288/abstract
15. García-Gómez JM, Luts J, Julià-Sape M, Krooshof P, Tortajada S, Robledo JV, Melssen W, Fuster-García E, Olier I, Postma G, Monleon D, Moreno-Torres À, Pujol J, Candiota AP, Martinez-Bisbal MC, Suykens J, Buydens L, Celda B, Van Huffel S, Arus C, Robles M (2009) Magma (New York, NY) 22(1):5. doi:10.1007/s10334-008-0146-y. http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2797843/. PMID: 18989714 PMCID: PMC2797843
16. Fuster-García E, Navarro C, Vicente J, Tortajada S, García-Gómez JM, Saez C, Calvar J, Griffiths J, Julià-Sape M, Howe FA, Pujol J, Peet AC, Heerschap A, Moreno-Torres À, Martinez-Bisbal MC, Martinez-Granados B, Wesseling P, Semmler W, Capellades J, Majos C, Alberich-Bayarri À, Capdevila A, Monleon D, Marti-Bonmati L, Arus C, Celda B, Robles M (2011) Magn Reson Mater Phys Biol Med 24(1):35. doi:10.1007/s10334-010-0241-8. http://link.springer.com/article/10.1007/s10334-010-0241-8
17. Fuster-García E, Tortajada S, Vicente J, Robles M, García-Gómez JM (2012) NMR Biomed. doi:10.1002/nbm.2895. http://onlinelibrary.wiley.com/doi/10.1002/nbm.2895/abstract
18. Tortajada S, García-Gómez JM, Vicente J, Sanjuán J, de Frutos R, Martín-Santos R, García-Esteve L, Gornemann I, Gutiérrez-Zotes A, Canellas F, Carracedo A, Gratacos M, Guillamat R, Baca-García E, Robles M (2009) Methods Inf Med 48(3):291. doi:10.3414/ME0562. PMID: 19387507
19. Chang CC, Lin CJ (2011) ACM Trans Intell SystTechnol2(3):27:1.doi:10.1145/1961189.1961199
20. Osuna E, Freund R, Girosi F (1997) Support vector machines: training and applications. Massachusetts Institute of Technology, Cambridge
21. Vapnik VN (1998) Statistical learning theory, 1st edn. Wiley-Interscience, New York
22. Thai-Nghe N, Gantner Z, Schmidt-Thieme L (2010) In: The 2010 international joint conference on neural networks (IJCNN), pp 1–8. doi:10.1109/IJCNN.2010.5596486

http://www.springer.com/978-1-4939-1984-0

Data Mining in Clinical Medicine
Llatas, C.F.; García-Gómez, J.M. (Eds.)
2015, XII, 270 p. 92 illus., 82 illus. in color., Hardcover
ISBN: 978-1-4939-1984-0
A product of Humana Press