Mach Learn (2009) 76: 271–285
DOI 10.1007/s10994-009-5132-8

Cost-sensitive learning based on Bregman divergences

Raúl Santos-Rodríguez · Alicia Guerrero-Curieses · Rocío Alaiz-Rodríguez · Jesús Cid-Sueiro

Received: 17 June 2009 / Revised: 17 June 2009 / Accepted: 21 June 2009 / Published online: 23 July 2009

Abstract This paper analyzes the application of a particular class of Bregman divergences to design cost-sensitive classifiers for multiclass problems. We show that these divergence measures can be used to estimate posterior probabilities with maximal accuracy for the probability values that are close to the decision boundaries. Asymptotically, the proposed divergence measures provide classifiers minimizing the sum of decision costs in non-separable problems, and maximizing a margin in separable MAP problems.

Keywords Cost sensitive learning · Bregman divergence · Posterior class probabilities · Maximum margin

1 Introduction

The general problem of cost-sensitive learning consist of designing decision or regression machines that take into account the costs involved in the whole decision/estimation process: this includes the cost of data acquisition, which may depend on the attributes, the cost of

Editors: Aleksander Kołcz, Dunja Mladenić, Wray Buntine, Marko Grobelnik, and John Shawe-Taylor.

R. Santos-Rodríguez · J. Cid-Sueiro (✉)
Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Leganés (Madrid), Spain
e-mail: jcid@tsc.uc3m.es

R. Santos-Rodríguez
e-mail: rsrodriguez@tsc.uc3m.es

A. Guerrero-Curieses
Department of Signal Theory and Communications, Universidad Rey Juan Carlos, Fuenlabrada (Madrid), Spain
e-mail: alicia.guerrero@urjc.es

R. Alaiz-Rodríguez
Department of Electrical and Electronic Engineering, Universidad de León, León, Spain
e-mail: rocio.alaiz@unileon.es

labeling training samples, and the cost of each possible decision error. This paper is focused in the latter case, though we believe that the proposed approach to the problem could be extended to some more general situations, like those where the cost may depend on the selected features.

Three main general approaches have been proposed to deal with multiclass cost-sensitive problems:

1. Methods based on modifying the training data set. The most popular technique lies in rescaling the original class distribution of the training data set according to the cost decision matrix by means of subsampling/oversampling, modifying decision thresholds or assigning instance weights. These modifications have shown to be effective in many binary problems and can also be applied to any cost insensitive learning algorithm (Zadrozny et al. 2003; Liu and Zhou 2006).
2. Other methods change the learning process in order to build a binary cost-sensitive classifier, such as those proposed for neural networks (Kukar and Kononenko 1998) decision trees (Bradford et al. 1998) or boosting-based ensemble machines like AdaCost (Fan et al. 1999). Finally,
3. Methods based on the Bayes decision theory that assign instances to the class with minimum expected cost. Obtaining calibrated probability estimates at the classifier output requires a suitable learning machine, a large enough representative data set as well as an adequate loss function to be minimized during learning. Nonetheless, real-valued scores from any classifier can also be transformed into well calibrated probabilities by methods like Platt Scaling (Platt 1999) or Isotonic Regression (Zadrozny and Elkan 2002). Though less popular (Zadrozny and Elkan 2001a), this is the approach that uses a natural way to cope with multiclass cost-sensitive problems.

Cost-sensitive learning in multiclass domains becomes a challenging task due to the number of misclassification costs involved the decision making process. Abe et al. (2004) propose an iterative method for these problems that can be used by any binary classification algorithm. Other works tackle this issue by decomposing the original problem into multiple two-class classification tasks (Marrocco and Tortorella 2004; Lozano and Abe 2008) or converting the cost matrix with $L \times L$ elements (where L is the number of classes) into a cost vector (Kukar and Kononenko 1998; Liu and Zhou 2006) with L components.[1]

Our proposal belongs to the third category (based on Bayes decision theory) and focus on the unequal costs that result from the different misclassification errors. Classical decision theory shows that cost matrices define class boundaries determined by posterior class probability estimates. So, accurate posterior class probabilities estimates should be achieved to optimize decisions.

In a binary problem, the empirical threshold can be found with the ROC (Receiver Operating Characteristics) curve plotted for different thresholds (Provost and Fawcett 2001). Recently, it has also been extended for multiclass problems (O'Brien and Gray 2005; O'Brien et al. 2008) using a greedy optimization approach that may lead in some cases to a local optima. Another alternative is to improve the overall quality of probability estimates. Zadrozny and Elkan propose several post-processing methods to transform classifier scores into calibrated probability estimates for binary (Zadrozny and Elkan 2001b) and multiclass problems (through a decomposition into binary classification problems) (Zadrozny and Elkan 2002).

[1] Note, however, that its effectiveness depends on the cost information which is lost with the transformation.

Strictly speaking, in order to make optimal decisions, accurate probability estimates are only required near the decision boundaries. This paper is grounded on some previous works (Miller et al. 1993; Cid-Sueiro et al. 1999) on the analysis and description, in the context of machine learning, of proper loss functions, which are those minimized at calibrated probabilities. The idea of designing proper loss functions to increase the estimation accuracy for some pre-defined probability values was initially suggested in Cid-Sueiro and Figueiras-Vidal (2001), further explored in Guerrero-Curieses et al. (2004) for binary classification, and extended to multiclass problems in Guerrero-Curieses et al. (2005).

In this paper, we reformulate some of these previous results by using Bregman divergences (Bregman 1967). Our first purpose is to establish some links between several results published in the machine learning literature, concerning the estimation of posterior class probabilities, with some general results on the problem of probability elicitation, which has been widely studied in the context of subjective probability: general conditions on proper loss functions can be dated back to Savage (1971), and it is also well known (see Gneiting and Raftery 2007 and the references therein) that any proper loss function is essentially characterized by a Bregman divergence.

Bregman divergences have attracted recent attention in the machine learning literature (Dhillon et al. 2005). The utility of these measures to define tailored loss functions for cost-sensitive classification has been explored in Stuetzle et al. (2005) for binary problems. The application of Bregman divergences (though under the name of *strict sense Bayesian* divergences) was also proposed in Guerrero-Curieses et al. (2005), which is, up to our knowledge, the only published work on the multiclass case.

In this paper, we propose a novel parametric family of Bregman divergences that may be used to train cost-sensitive classifiers in multi-class situations. The proposed divergence measures are in general non-convex functions of the model parameters, but we show some connections between the minimization of the divergence measures and some kind of large margin classifiers, which opens the door to some convex optimization algorithms.

The structure of this paper is as follows: Sect. 2 states the learning and decision problem and shows the fundamentals of entropy and divergence measures. Section 3 presents a new family of entropy functions used to design a Bregman divergence that achieves maximal sensitivity near the decision boundaries defined by unequal costs. The asymptotic behavior of this divergence measure is analyzed in Sect. 4. Its application to some different real data sets is exposed in Sect. 5. Finally, we summarize the main conclusions and suggest further research lines in Sect. 6.

2 Decision and learning

2.1 Cost-sensitive decision problems

Let $\mathcal{X}$ be an observation space and $\mathcal{U}_L$ a finite set of L classes or labels. For mathematical convenience, we assume that the i-th class in $\mathcal{U}_L$ is a binary unit vector $\mathbf{u}_i$ with components $u_{ij} = \delta_{i-j}$ (that is, a unique "1" at the i-th position).

In a general classification problem, a pair $(\mathbf{x}, \mathbf{d}) \in \mathcal{X} \times \mathcal{U}_L$ is generated according to a probability model $p(\mathbf{x}, \mathbf{d})$. The goal is to predict class vector $\mathbf{d}$ when only $\mathbf{x}$ is observed.

In a general setting, a cost $c(\hat{\mathbf{d}}, \mathbf{d}, \mathbf{x})$ can be associated to decide in favor of class $\hat{\mathbf{d}}$ when the true class is $\mathbf{d}$ and the observation is $\mathbf{x}$. The general decision problem consists on making decisions minimizing the mean risk $E\{c(\hat{\mathbf{d}}, \mathbf{d}, \mathbf{x})\}$.

It is well-known that such minimum is reached by taking, for every sample $\mathbf{x}$, class $\hat{\mathbf{d}}^*$ such that

$$\hat{\mathbf{d}}^* = \arg\min_{\hat{\mathbf{d}}} \left\{ \sum_{j=1}^{L} E\{c(\hat{\mathbf{d}}, \mathbf{u}_j, \mathbf{x})|\mathbf{x}\} p_j \right\} \tag{1}$$

where $p_j = P\{\mathbf{d} = \mathbf{u}_j \mid \mathbf{x}\}$ is the posterior probability of class j given sample $\mathbf{x}$. In this paper we assume that c is deterministic, and it does not depend on the observation, so that, defining $c_{ij} = c(\hat{\mathbf{u}}_i, \mathbf{u}_j, \mathbf{x})$, we can write the optimal decision as $\hat{\mathbf{d}}^* = \mathbf{u}_{i^*}$ such that

$$i^* = \arg\min_{i} \left\{ \sum_{j=1}^{L} c_{ij} p_j \right\} \tag{2}$$

In particular, taking $c_{ij} = (1 - \delta_{i-j})$, we get $i^* = \arg\max_i\{p_i\}$, which is the decision rule of the *Maximum A Posteriori* (MAP) classifier.

2.2 Posterior probability estimation

In a general learning problem, the probability model $p(\mathbf{x}, \mathbf{d})$ is unknown, and only a training set $\mathcal{S} = \{(\mathbf{x}_k, \mathbf{d}_k), k = 1, \ldots, K\}$ of statistically independent samples (drawn from model p) is available. The classical discriminative approach to the problem consists on estimating a posterior probability map $\mathbf{y} = \mathbf{f}_\mathbf{w}(\mathbf{x})$, where $\mathbf{f}_\mathbf{w} : \mathcal{X} \to \mathcal{P}_L$ is a function with parameters $\mathbf{w}$, transforming every element of the observation space into an element of the set of probability vectors $\mathcal{P}_L = \{\mathbf{p} : 0 \leq p_i \leq 1, \sum_{i=1}^{L} p_i = 1\}$, and replace the true probabilities p_i in (1) by their estimates y_i.

Estimating posterior probabilities may be inefficient. If the goal is to optimize decisions, accurate estimates of posterior probabilities far from the decision boundaries are actually not needed, and focusing learning on these estimates may be suboptimal.

Some previous definitions are required. Following Kapur and Kesavan (1993), we define generalized entropy and divergence measures as follows

Definition 1 Function $h : \mathcal{P}_L \to \Re$, is an *entropy* if $h(\mathbf{u}_i) = 0$, for every $\mathbf{u}_i \in \mathcal{U}_L$ and h is strictly concave[2] in $\mathcal{P}_L$

Note that any entropy verifies $H(\mathbf{p}) \geq 0$, for every $\mathbf{p} \in \mathcal{P}_L$.

Definition 2 Function $D : \mathcal{P}_L \times \mathcal{P}_L \to \mathbb{R}$, is a *divergence* among probability vectors $\mathbf{p}$ and $\mathbf{y}$ if it satisfies the following properties:

1. Nonnegativity: $D(\mathbf{p}, \mathbf{y}) \geq 0$;
2. Identity: $D(\mathbf{p}, \mathbf{y}) = 0$ iff $\mathbf{p} = \mathbf{y}$;
3. Convexity: $D(\mathbf{p}, \mathbf{y})$ is a strictly convex function of $\mathbf{p}$.

[2] Since concavity and convexity are not unanimously defined in the literature, let us make clear that, in this paper, a function is strictly concave (convex) if its Hessian matrix is negative definite (positive).

Our approach in this paper is based on the estimation of posterior class probabilities by minimizing divergence sums given by

$$O(\mathbf{w}) = \sum_{k=1}^{K} D(\mathbf{d}^k, \mathbf{y}^k) \tag{3}$$

where $\mathbf{y}^k = f_{\mathbf{w}}(\mathbf{x}^k)$. One may wonder if parameters $\mathbf{w}^*$ minimizing $O(\mathbf{w})$ provide an estimate of posterior probabilities $\mathbf{p}$. The answer is positive for a particular class of divergence measures.

Definition 3 Bregman Divergence (Bregman 1967)

Given entropy $h : \mathcal{P}_L \to \mathbb{R}$, the *Bregman divergence* $D : \mathcal{P}_L \times \mathcal{P}_L \to \mathbb{R}$ relative to h is defined as

$$D_h(\mathbf{p}, \mathbf{y}) = h(\mathbf{y}) - h(\mathbf{p}) + (\mathbf{p} - \mathbf{y})^T \nabla_{\mathbf{y}} h(\mathbf{y}) \tag{4}$$

where $\nabla_{\mathbf{y}} h(\mathbf{y})$ represents the gradient vector of h evaluated at $\mathbf{y}$.

The main result is the following

Theorem 1 *Let* $(\mathbf{x}, \mathbf{d}) \subset \mathcal{X} \times \mathcal{U}_L$ *a pair of random variables with arbitrary joint distribution* $p(\mathbf{x}, \mathbf{d})$, *and let* $\mathbf{p}$ *be the posterior probability map given by* $p_i = P\{\mathbf{d} = \mathbf{u}_i | \mathbf{x}\}$. *The divergence measure* $D : \mathcal{P}_L \times \mathcal{P}_L \to \mathbb{R}$ *satisfies*

$$\arg\min_{\mathbf{y}} E\{D(\mathbf{d}, \mathbf{y}) | \mathbf{x}\} = \arg\min_{\mathbf{y}} E\{D(\mathbf{p}, \mathbf{y}) | \mathbf{x}\} \tag{5}$$

for any distribution $p(\mathbf{x}, \mathbf{d})$ *if and only if* D *is a Bregman divergence for some entropy measure* h.

The theorem shows that probability estimates minimizing the mean divergence can be found by minimizing $E\{D(\mathbf{d}, \mathbf{y})\}$, which, in practice, can be estimated from samples as in (3). Moreover, since $\arg\min_{\mathbf{y}} E\{D(\mathbf{p}, \mathbf{y}) | \mathbf{x}\} = \mathbf{p}$, the posterior class probability vector is the minimizer of the expected divergence.

As a particular case, if $h(\mathbf{y}) = -\sum_{i=1}^{L} y_i \log(y_i)$ (i.e., the Shannon entropy), $D_h(\mathbf{p}, \mathbf{y})$ is the Kullback-Leibler divergence, and $D_h(\mathbf{d}, \mathbf{y})$ is the cross Entropy.

Theorem 1 is a reformulation of Theorem 1 in Cid-Sueiro et al. (1999) by using Bregman divergences (details of the proof can be found there), though the role of these divergences in the calibration of probabilities is well known in the area of subjective probability (see, for instance, a similar result in Gneiting and Raftery 2007). A recent generalization can be found in Banerjee et al. (2005).

Our approach in this paper is based on the idea (also explored in Guerrero-Curieses et al. 2005) of optimizing Bregman divergences which are very sensitive to deviations of $\mathbf{y}$ from values of $\mathbf{p}$ close to the decision boundaries. The strategy that we follow in the next section is to design specific divergence measures for each decision problem.

2.3 Sensitivity of a divergence measure

In general, posterior probability vector $\mathbf{p}$ is an unknown function of observation $\mathbf{x}$. If the final goal is to minimize a mean risk function, the accuracy of the probability estimates near the decision regions should be maximized. To do so, the Bregman divergence should have maximum *sensitivity* to changes at probability vectors near the decision regions. The sensitivity can be defined as follows:

Definition 4 The *sensitivity* of a Bregman divergence at $\mathbf{p} \in \mathcal{P}_L$ in direction $\mathbf{a}$ (with $\|\mathbf{a}\| = 1$ and $\sum_i a_i = 0$) is

$$s(\mathbf{p}, \mathbf{a}) = \left.\frac{\partial^2 D_h(\mathbf{p}, \mathbf{p} + \alpha\mathbf{a})}{\partial\alpha^2}\right|_{\alpha=0} = -\mathbf{a}^T \mathbf{H}_{\mathbf{yy}}(\mathbf{p})\mathbf{a} \tag{6}$$

where $\mathbf{H}_{\mathbf{yy}}$ is the Hessian matrix of the corresponding entropy $h(\mathbf{y})$.

(note that condition $\sum_i a_i = 0$ is necessary for $\mathbf{p} + \alpha\mathbf{a}$ in (6) to be a probability vector). The sensitivity measures the velocity of change of the divergence around $\mathbf{p}$. It is always non negative, since $D_h(\mathbf{p}, \mathbf{y})$ is a convex function of $\mathbf{y}$ at $\mathbf{y} = \mathbf{p}$, for any $\mathbf{p} \in \mathcal{P}_L$.

3 Designing Bregman divergences

3.1 A parametric family of entropies

If decision rule in (1) is based on estimates of posterior probabilities p_i, small estimation errors near the decision boundaries may change decisions and reduce the overall performance. This is the motivation to search for Bregman divergences with the highest sensitivity at probability values close to the decision boundaries and in the direction orthogonal to the boundary.

Since, according to Definition 2, a Bregman divergence can be specified from an entropy function (4), we define the family of entropies given by

$$h_R(\mathbf{y}) = -\|\mathbf{s} - \mathbf{C}\mathbf{y}\|_R + \mathbf{b}^T\mathbf{y} \tag{7}$$

where $\mathbf{s} = \max_{\mathbf{y}}\{\mathbf{u_i^T C y}\} = \max_{\mathbf{j}}\{\mathbf{c_{ij}}\}$, $\|\cdot\|_R$ is the R-norm (i.e., for any $\mathbf{z} \in \mathbb{R}$, $\|\mathbf{z}\|_R = (\sum_{i=1}^L z_i^R)^{1/R}$), $\mathbf{C}$ is the cost matrix with components c_{ij} (the cost of deciding in favor of class i when the true class is j), and R is a smooth parameter.

Parameter vector $\mathbf{b}$ should be adjusted so that $h_R(\mathbf{u}_i) = 0$, for any $\mathbf{u}_i \in \mathcal{U}_L$, though, as we will see later, it has no influence on the Bregman divergence. It is easy to see that $b_i = \|\mathbf{s} - \mathbf{C}\mathbf{u}_i\|_R$.

The concavity of h_R arises from the fact that the R-norm is strictly convex for any finite R, and convexity is preserved after any affine transformation of the variables. Moreover, if $\mathbf{C}$ is invertible, h_R is strictly concave so that it satisfies Definition 1, and the divergence D_R emanated from h_R using (4) is actually a Bregman divergence.

3.2 Bregman divergence

According to (4), and defining

$$\mathbf{z}(\mathbf{y}) = \mathbf{s} - \mathbf{C}\mathbf{y} \tag{8}$$

the Bregman divergence corresponding to h_R is

$$\begin{aligned} D_R(\mathbf{p},\mathbf{y}) &= \|\mathbf{z}(\mathbf{p})\|_R - \|\mathbf{z}(\mathbf{y})\|_R + \|\mathbf{z}(\mathbf{y})\|_R^{1-R}(\mathbf{z}^{R-1}(\mathbf{y}))^T\mathbf{C}(\mathbf{p}-\mathbf{y}) \\ &= \|\mathbf{z}(\mathbf{p})\|_R - \|\mathbf{z}(\mathbf{y})\|_R + \|\mathbf{z}(\mathbf{y})\|_R^{1-R}(\mathbf{z}^{R-1}(\mathbf{y}))^T(\mathbf{z}(\mathbf{y})-\mathbf{z}(\mathbf{p})) \\ &= \|\mathbf{z}(\mathbf{p})\|_R - \|\mathbf{z}(\mathbf{y})\|_R^{1-R}(\mathbf{z}^{R-1}(\mathbf{y}))^T\mathbf{z}(\mathbf{p}) \end{aligned} \tag{9}$$

Before analyzing the asymptotic behavior of the sample divergence, we show that, for large R, D_R has maximal sensitivity near the decision regions defined by $\min_i\{\sum_{j=1}^L c_{ij}p_j\}$.

3.3 Sensitivity analysis

According to (6), the sensitivity is a function of the Hessian matrix of the divergence. Since the gradient vector of h_R has components

$$\frac{\partial h_R}{\partial y_i} = \|\mathbf{z}\|_R^{1-R}\sum_{n=1}^L |z_n|^{R-1}c_{ni} + b_i \tag{10}$$

the second-order derivatives are

$$\begin{aligned} \frac{\partial^2 h_R}{\partial y_i\partial y_j} &= (R-1)\|\mathbf{z}\|_R^{1-2R}\sum_{m=1}^L |z_m|^{R-1}c_{mi}\sum_{n=1}^L |z_n|^{R-1}c_{nj} \\ &\quad - (R-1)\|\mathbf{z}\|_R^{1-R}\sum_{n=1}^L |z_n|^{R-2}c_{ni}c_{nj} \end{aligned} \tag{11}$$

Thus, the Hessian matrix of h_R can be expressed as

$$\mathbf{H}_{\mathbf{yy}} = (R-1)\|\mathbf{z}\|_R^{1-2R}\left(\mathbf{C}^T\mathbf{z}^{R-1}(\mathbf{z}^{R-1})^T\mathbf{C} - \|\mathbf{z}\|_R^R\mathbf{C}^T\mathbf{D}_{\mathbf{z}}^{R-2}\mathbf{C}\right) \tag{12}$$

where $\mathbf{D}_{\mathbf{z}}$ is a diagonal matrix with $\mathrm{diag}(\mathbf{D}_{\mathbf{z}}) = \mathbf{z}$ and $\mathbf{z}^{R-1}$ denotes a vector whose i-th component is z_i^{R-1}, for $i = 1, \ldots, L$.

Using (12) the sensitivity defined in (6) is

$$\begin{aligned} s(\mathbf{y},\mathbf{a}) &= -(R-1)\|\mathbf{z}\|_R^{1-2R}\left(\mathbf{a}^T\mathbf{C}^T\mathbf{z}^{R-1}(\mathbf{z}^{R-1})^T\mathbf{C}\mathbf{a} - \|\mathbf{z}\|_R^R\mathbf{a}^T\mathbf{C}^T\mathbf{D}_{\mathbf{z}}^{R-2}\mathbf{C}\mathbf{a}\right) \\ &= -(R-1)\|\mathbf{z}\|_R^{1-2R}\left(\left((\mathbf{z}^{R-1})^T\mathbf{C}\mathbf{a}\right)^2 - \|\mathbf{z}\|_R^R\mathbf{a}^T\mathbf{C}^T\mathbf{D}_{\mathbf{z}}^{R-2}\mathbf{C}\mathbf{a}\right) \end{aligned} \tag{13}$$

For any decision problem given by cost matrix $\mathbf{C}$ and posterior probability vector $\mathbf{y}$, any class m satisfying

$$\sum_j c_{mj}y_j = \min_n\left\{\sum_j c_{nj}y_j\right\} \tag{14}$$

is optimal (because it minimizes the expected cost). Let k be the number of optimal classes for some $\mathbf{y}$. Note that, if $k = 1$, $\mathbf{y}$ is an interior point of a decision region. If $k > 1$, $\mathbf{y}$ is a point in the boundary between k decision regions. For large R, the powers of z_i for any non-optimal class i can be neglected, and we can approximate

$$\begin{aligned} s(\mathbf{y},\mathbf{a}) &\approx -(R-1)k^{\frac{1-2R}{R}} z_m^{1-2R} \left(\left(k z_m^{R-1}\mathbf{u}^T\mathbf{C}\mathbf{a}\right)^2 - k z_m^R z_m^{R-2}\mathbf{a}^T\mathbf{C}^T\mathbf{D}_\mathbf{u}\mathbf{C}\mathbf{a}\right) \\ &\approx -(R-1)k^{\frac{1-R}{R}} z_m^{-1} \left(k\left(\mathbf{u}^T\mathbf{C}\mathbf{a}\right)^2 - \mathbf{a}^T\mathbf{C}^T\mathbf{D}_\mathbf{u}\mathbf{C}\mathbf{a}\right) \end{aligned} \tag{15}$$

where $\mathbf{u}$ is a vector with components equal to 1 at the optimal classes, and zero otherwise, and $\mathbf{D}_\mathbf{u}$ is a diagonal matrix with $\mathbf{u}$ in the diagonal.

Analyzing the value of (15), it is not difficult to see that:

1. Far from the boundary: when $R \to \infty$, then $\|z\|_R \to \max_i\{z_i\}$ and

$$s(\mathbf{y},\mathbf{a}) \to 0 \tag{16}$$

2. At the boundary between two or more decision regions, the sensitivity goes to infinity for any direction $\mathbf{a}$, (because of the factor $R-1$ in (15)), unless some other factor is zero: it is not difficult to see that, for any vector $\mathbf{a}$ along the boundary decision, the right hand side of (15) is zero. Thus, at each point $\mathbf{y}$ in the boundary between several decision regions, the sensitivity to directions along the boundary tend to zero, while it tends to ∞ for any orthogonal direction.

4 Asymptotic analysis

Replacing probability vector $\mathbf{p}$ by the label vector, $\mathbf{d}$, we obtain

$$D_R(\mathbf{d},\mathbf{y}) = \|\mathbf{z}(\mathbf{d})\|_R - \|\mathbf{z}(\mathbf{y})\|_R^{1-R}(\mathbf{z}^{R-1}(\mathbf{y}))^T\mathbf{z}(\mathbf{d}) \tag{17}$$

The sum of the above expression computed over a set of training samples (as in (3)) is the objective function that should be minimized.

In order to analyze the behavior of D_R for large values of R, we will use an alternative expression. Let m be the index of the true class (i.e., $\mathbf{d} = \mathbf{u}_m$) and $\hat{m}$ the index of the classifier decision given $\mathbf{y}$, i.e.,

$$\hat{m} = \arg\min_i \left\{ \sum_{j=1}^{L} c_{ij} y_j \right\} = \arg\max_i z_i(\mathbf{y}) \tag{18}$$

Then, D_R can be written as

$$D_R(\mathbf{d},\mathbf{y}) = \|\mathbf{z}(\mathbf{u}_m)\|_R - \|\mathbf{z}(\mathbf{y})\|_R \frac{\sum_{i=1}^{L} z_i^{R-1}(\mathbf{y}) z_i(\mathbf{u}_m)}{\sum_{i=1}^{L} z_i^R(\mathbf{y})} \tag{19}$$

4.1 Non-separable data

For large R, (19) becomes

$$\lim_{R\to\infty} D_R(\mathbf{d},\mathbf{y}) = \max_i z_i(\mathbf{u}_m) - z_{\hat{m}}(\mathbf{y}) \frac{z_{\hat{m}}(\mathbf{u}_m)}{z_{\hat{m}(\mathbf{y})}} = c_{\hat{m}m} - \min_i c_{im} \tag{20}$$

(Usually, $\min_i c_{im} = c_{mm} = 0$ and the above limit is $c_{\hat{m}m}$.) Thus, the divergence converges to the difference between the cost of the classifier decision and the cost of the correct decision. If the classifier makes the correct decision (i.e., the one minimizing c_{im}), the divergence is zero. Thus, in the limit, the objective function given by (3) converges to

$$\lim_{R\to\infty} O_R(\mathbf{w}) = \sum_{k=1}^{K} \left(c_{\hat{m}^k m^k} - \min_i c_{im^k} \right) \tag{21}$$

where m^k and $\hat{m}^k$ represent the index of the true class and the assigned class for sample $\mathbf{x}^k$, respectively. That is, the divergence converges to the difference in the total classification cost and the minimum achievable cost. In the MAP case, this equals the number of decision errors.

4.2 Separable data

If data are separable, then the limit in (21) is zero for any separating boundary. In this section we analyze which zero-error boundary is obtained when the loss in (3) is minimized.

It is interesting to analyze the behavior of this classifier for large R, when the sample is correctly classified. Though we will restrict our analysis to the MAP case, we provide a formula for the asymptotic divergence for an arbitrary cost matrix $\mathbf{C}$. Using (19), we can write

$$D_R(\mathbf{d}, \mathbf{y}) = \|\mathbf{z}(\mathbf{u}_m)\|_R - \left(\sum_{j=1}^{L} z_j^R(\mathbf{y}) \right)^{\frac{1}{R}} \sum_{i=1}^{L} \frac{z_i^R(\mathbf{y})}{\sum_{j=1}^{L} z_j^R(\mathbf{y})} \frac{z_i(\mathbf{u}_m)}{z_i(\mathbf{y})} \tag{22}$$

Consider an arbitrary sample, $\mathbf{x}$, from class m, that is out of any decision boundary. If decision $\hat{m}$ in (18) is correct, then $\max_i\{z_i(\mathbf{y})\} = z_m(\mathbf{y})$, and we can make first order approximations

$$\frac{z_i^R(\mathbf{y})}{\sum_{j=1}^{L} z_j^R(\mathbf{y})} = \frac{z_i^R(\mathbf{y})}{z_m^R(\mathbf{y})} \frac{1}{1 + \sum_{j\neq m}^{L} \frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})}} \approx \frac{z_i^R(\mathbf{y})}{z_m^R(\mathbf{y})} \left(1 - \sum_{j\neq m}^{L} \frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})} \right) \tag{23}$$

and,

$$\begin{aligned} \left(\sum_{j=1}^{L} z_j^R(\mathbf{y}) \right)^{\frac{1}{R}} &= z_m(\mathbf{y}) \left(1 + \sum_{j\neq m} \frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})} \right)^{\frac{1}{R}} \\ &\approx z_m(\mathbf{y}) \left(1 + \frac{1}{R} \sum_{j\neq m} \frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})} \right) \end{aligned} \tag{24}$$

Using (23) and (24) in (22), we get

$$\begin{aligned} D_R(\mathbf{d}, \mathbf{y}) \approx{}& \|\mathbf{z}(\mathbf{u}_m)\|_R \\ & - z_m(\mathbf{y}) \left(1 + \frac{1}{R} \sum_{j\neq m} \frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})} \right) \sum_{i=1}^{L} \frac{z_i^R(\mathbf{y})}{z_m^R(\mathbf{y})} \left(1 - \sum_{j\neq m}^{L} \frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})} \right) \frac{z_i(\mathbf{u}_m)}{z_i(\mathbf{y})} \end{aligned}$$

$$
\begin{aligned}
&= \|\mathbf{z}(\mathbf{u}_m)\|_R \\
&\quad - \left(1 + \frac{1}{R}\sum_{j\neq m}\frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})}\right)\left(1 - \sum_{j\neq m}\frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})}\right)\sum_{i=1}^{L}\frac{z_i^{R-1}(\mathbf{y})}{z_m^{R-1}(\mathbf{y})}z_i(\mathbf{u}_m) \\
&\approx \|\mathbf{z}(\mathbf{u}_m)\|_R - \left(1 - \frac{R-1}{R}\sum_{j\neq m}\frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})}\right)\sum_{i=1}^{L}\frac{z_i^{R-1}(\mathbf{y})}{z_m^{R-1}(\mathbf{y})}z_i(\mathbf{u}_m) \\
&\approx \|\mathbf{z}(\mathbf{u}_m)\|_R - z_m(\mathbf{u}_m) + \frac{R-1}{R}z_m(\mathbf{u}_m)\sum_{j\neq m}\frac{z_j^R(\mathbf{y})}{z_m^R(\mathbf{y})} \\
&\quad - \sum_{i\neq m}\frac{z_i^{R-1}(\mathbf{y})}{z_m^{R-1}(\mathbf{y})}z_i(\mathbf{u}_m)
\end{aligned} \tag{25}
$$

A further approximation can be made if we note that, as R grows, only the terms with the highest values of z_j/z_m are relevant. Let n be the index of a "2nd-best" class, such that $n = \arg\max_{i\neq m} z_i(\mathbf{y})$, and q the number of classes satisfying this condition, and Q the set of indices of such classes. For large R, we can further approximate

$$
\begin{aligned}
D_R(\mathbf{d},\mathbf{y}) &\approx \|\mathbf{z}(\mathbf{u}_m)\|_R - z_m(\mathbf{u}_m) + \frac{R-1}{R}q z_m(\mathbf{u}_m)\frac{z_n^R(\mathbf{y})}{z_m^R(\mathbf{y})} \\
&\quad - \frac{z_n^{R-1}(\mathbf{y})}{z_m^{R-1}(\mathbf{y})}\sum_{i\in Q} z_i(\mathbf{u}_m)
\end{aligned} \tag{26}
$$

4.3 Maximum margin as a limit classifier

Starting from (26), we will show that, in the Maximum A Posteriori (MAP) case and using an exponential probability map, the classifier minimizing the asymptotic divergence tends to behave like a maximum margin classifier. To do so, let us assume that $\mathbf{C} = \mathbf{1}\mathbf{1}^T - \mathbf{I}$ (the MAP case), so that $\mathbf{z}(\mathbf{y}) = \mathbf{y}$, and (26) becomes

$$D_R(\mathbf{d},\mathbf{y}) \approx \frac{R-1}{R}q\frac{y_n^R}{y_m^R} \tag{27}$$

Consider the exponential posterior probability estimate given by

$$\mathbf{y} = \mathbf{f}_{\mathbf{W}}(\mathbf{x}) = \frac{\exp(\mathbf{d}^T(\mathbf{W}\phi(\mathbf{x}) + \mathbf{b}))}{\sum_i \exp(\mathbf{u}_i^T(\mathbf{W}\phi(\mathbf{x}) + \mathbf{b}))} \tag{28}$$

where $\mathbf{W}$ is a parameter matrix, $\mathbf{b}$ is a parameter vector and $\phi : \mathcal{X} \to \mathbb{R}^{N'}$ is a nonlinear feature map. In such case, (27) reduces to

$$D_R(\mathbf{d},\mathbf{y}) \approx q\exp(R(\mathbf{w}_n - \mathbf{w}_m)\phi(\mathbf{x}) + b_n - b_m) \tag{29}$$

where $\mathbf{w}_n$ is the n-th row in $\mathbf{w}$. If $P_{n,m}$ is the hyperplane defined by the equation $(\mathbf{w}_n - \mathbf{w}_m)\phi(\mathbf{x}) + b_n - b_m = 0$, and $d(\mathbf{x}, P_{n,m})$ is the euclidean distance (in the feature space) from $\phi(\mathbf{x})$ to $P_{n,m}$, we can write

$$D_R(\mathbf{d},\mathbf{y}) \approx q\exp(R\|\mathbf{w}_n - \mathbf{w}_m\|_2 d(\mathbf{x}, P_{n,m})) \tag{30}$$

For the whole training set, we get

$$O(\mathbf{W}) \approx \sum_{k=1}^{K} q^k \exp(-R\|\mathbf{w}_{n^k} - \mathbf{w}_{m^k}\|_2 d(\mathbf{x}^k, P_{n^k,m^k})) \\ \approx q^\ell \exp(-R\|\mathbf{w}_{n^\ell} - \mathbf{w}_{m^\ell}\|_2 d(\mathbf{x}^\ell, P_{n^\ell,m^\ell})) \tag{31}$$

where ℓ is the index of the sample in the training set that minimizes the negative of the exponent,

$$\ell = \arg\max_k \left\{\|\mathbf{w}_{n^k} - \mathbf{w}_{m^k}\|_2 d(\mathbf{x}^k, P_{n^\ell,m^k})\right\} \tag{32}$$

(if several samples attain this minimum, q_ℓ must be replaced by its sum over all that samples). This expression can be maximized by making $\|\mathbf{w}_{n^k} - \mathbf{w}_{m^k}\|_2$ large (which is easy to do by multiplying matrix $\mathbf{W}$ and $\mathbf{b}$ by a constant factor, which does not modify the decision boundaries). However, imposing some constraints on the size of $\mathbf{W}$, the minimum of $O(\mathbf{W})$ is obtained by maximizing the distances from samples to decision boundaries. Thus, for large R, the classifier optimizing $O(\mathbf{W})$ tends to behave as a maximum margin classifier.

The analysis of the non-MAP case is more complex. However, (26) shows that, for large R, the asymptotical divergence depends critically on the factor $\frac{z_n^R(\mathbf{y})}{z_m^R(\mathbf{y})}$. Using an exponential model $\mathbf{z}(\mathbf{y}) \propto \exp(\mathbf{d}^T(\mathbf{W}\phi(\mathbf{x}) + \mathbf{b}))$, it is easy to see that the divergence sum is similar to (32) and the boundary decision of the optimal classifier (when data are separable) does not depend on the cost matrix. Though this may seem surprising, it is in accordance with the boundary decision provided by other maximum margin classifiers, such as cost-sensitive support vector machines, which usually include the costs parameters in the slack variables, without apparent influence when dealing with separable data.

5 Experiments

In this section we show the results of experiments carried out to test our approach. We conducted systematic experiments to compare the performance of the proposed method with a number of existing algorithms: an architecture based on the classical cross entropy objective function; oversampling and threshold-moving to train cost-sensitive neural networks (we refer the reader to Liu and Zhou (2006) for the detailed description of the comparison methods we use); using multiclass data sets from the UCI repository.

We deal with two different objective functions: cross entropy (CE) versus the Bregman divergence loss function obtained from (7) (BD), in both cases using a neural network which computes the probability model given by

$$y_i = \sum_j y_{ij} \tag{33}$$

being

$$y_{ij} = \frac{\exp(\mathbf{w}_{ij}^T \mathbf{x})}{\sum_l \sum_m \exp(\mathbf{w}_{lm}^T \mathbf{x})} \tag{34}$$

Learning consists of estimating parameters $\mathbf{w}$ by means of the stochastic gradient minimization of the Bregman divergence. For instance, the stochastic gradient learning rule to

Table 1 UCI data sets description (C: continuous)

Data set	Size	Attribute	Class distribution
German	1000	24C	700/300
Heart	303	13C	164/139

minimize BD with a probabilistic model with parameters **w** is given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \rho\nabla_{\mathbf{w}} L_r(\mathbf{y}, \mathbf{d}) = \mathbf{w}(k) - \rho(\mathbf{d} - \mathbf{y})^T \mathbf{H}_{\mathbf{yy}} \nabla_{\mathbf{w}} \mathbf{y} \tag{35}$$

where $\mathbf{H}_{\mathbf{yy}}$ is the Hessian matrix given by (12). This shows that $\mathbf{H}_{\mathbf{yy}}$ modulates the error correcting term in the learning rule. Initial adaptation step is set to μ_0 and decreases according to $\mu_k = \mu_0/(1 + k/k_0)$, where k is the training number. (μ_0 and k_0 determine the convergence velocity).

Both, oversampling and threshold moving algorithms were coupled with the CE network scheme described above. They were selected due to its simplicity and the fact that in two-class tasks were shown to be effective in cost-sensitive learning, reducing the misclassification costs. The results are obtained using a network architecture with $m = 2$ in (34); it is the configuration chosen to be trained with both CE and BD loss functions.

Two data sets from the UCI Machine Learning Repository are used to evaluate the algorithms: heart disease and German credit data. The description of each data set is shown in Table 1.

In the same way as Liu and Zhou (2006), three types of cost matrices are suitable with the selected UCI databases, defined as:

1. $1.0 < c_{ij} \leq 10.0$ only for a single value $j = v$ and $c_{ij \neq v} = 1\ \forall j \neq i$.
2. $1.0 \leq c_{ij} = V_i \leq 10.0$ for each $j \neq i$. At least one $V_i = 1$.
3. $1.0 \leq c_{ij} \leq 10.0$ for each $j \neq i$. At least one $c_{ij} = 1$.

The three conditions are the same in case we work with a binary classification task. As an example, cost matrices (**C**) used in the experiments are chosen similar to the next one:

$$\mathbf{C} = \begin{pmatrix} 0 & 5 \\ 1 & 0 \end{pmatrix} \tag{36}$$

The experiments are carried out in the following way: first of all, we generate ten random cost matrices to estimate the average misclassification cost. Then, a 10-fold cross validation scheme is implemented: each data set is partitioned into ten subsets with similar sizes and distributions, using nine of them as the training set and the remaining subset as the test set. This procedure is repeated ten times to use each set as test set at least once. The whole process is then performed for ten random permutations of the dataset and the average results are recorded as the final results.

Tables 2 and 3 summarize the results of our experiments, giving the average test set error, misclassification cost and standard error for each of the data sets, and for each of four methods considered.[3] Table 2 compares the average error of all comparison methods. From these results, it appears convincing that the designed Bregman divergences family outperform all of the comparison methods we have considered. Table 3 compares the performance, in misclassification cost, of the algorithms for both data sets, which is the main point of interest

[3] In these and subsequent tables, the merit figures that correspond to best performance are shown in bold font.

Table 2 Average error rate for different classification procedures (NN-BD, NN-CE, NN-Oversampling and NN-Th. Moving) and different data sets

Dataset	Error rate (Test)			
	NN-BD	NN-CE	NN-Oversampling	NN-Th. Moving
German	**0.232 ± 0.032**	0.247 ± 0.031	0.244 ± 0.061	0.253 ± 0.041
Heart	**0.144 ± 0.049**	0.187 ± 0.053	0.193 ± 0.044	0.226 ± 0.060

Table 3 Average cost and standard error for different classification procedures (NN-BD, NN-CE, NN-Oversampling and NN-Th. Moving) and different data sets

Dataset	Cost (Test)			
	NN-BD	NN-CE	NN-Oversampling	NN-Th. Moving
German	**43.2 ± 1.7**	57.9 ± 3.3	45.9 ± 4.1	47.7 ± 1.5
Heart	**9.1 ± 0.9**	12.1 ± 1.3	11.7 ± 2.1	12.3 ± 1.4

Table 4 Error rates in training and test sets for different data sets (Decision Criterion: MAP). German2/3 and German1/3, contain 2/3 and 1/3 of the samples of the original German data set, respectively

Dataset	Loss Function	Error rate	
		Train	Test
German	CE	0.217 ± 0.019	0.247 ± 0.031
	BD	**0.186 ± 0.014**	**0.232 ± 0.032**
German2/3	CE	0.207 ± 0.022	0.262 ± 0.018
	BD	**0.192 ± 0.011**	**0.211 ± 0.035**
German1/3	CE	0.223 ± 0.032	0.269 ± 0.024
	BD	**0.221 ± 0.019**	**0.240 ± 0.044**

of our approach. It is confirmed that using BD in cost-sensitive learning, for high values of R, seems to be a good alternative to be further developed, which coincides with what was expected by our previous motivation.

It is interesting to note that the results shown in Table 4 try to support the relationship between the behavior of the proposed model and some kind of large margin classifying. We evaluate CE and BD loss functions, using MAP as decision criterion, while reducing the number of training samples to probe that using BD provides a robust classifier in scarce data scenarios. It also achieves slightly better generalization, as we can realize comparing training and test error rate results.

The main conclusion of the performed experiments is that the improvement in the obtained error rate results is not statistically noteworthy but we can highlight its behavior in average cost.

Another aspect to be stressed is the difficulty of finding out the optimum value of R, which is a crucial and decisive factor to get adequate results, as well as a high sensitive parameter. This problem, together with the drawbacks of the stochastic gradient learning rule used to minimize the loss function (in general the algorithm converges only to a local optimum), points at the necessity of exploring alternative optimization algorithms.

6 Conclusions

In this paper we propose a general procedure to train multiclass classifiers for particular cost-sensitive decision problems, which is based on estimating posterior probabilities using Bregman divergences. We have proposed a parametric family of Bregman divergences that can be tuned to a specific cost matrix. Our asymptotic analysis shows that the optimization of the Bregman divergence for large values of parameter R becomes equivalent to minimize the overall cost regret in non-separable problems, and to maximize a margin in separable problems. We show that using the learning algorithm based on Bregman divergences with a simple classifier, the error/cost results obtained are lower than those given by the cross entropy solely or combined with some well-known cost-sensitive algorithms.

As the linear combination of Bregman divergences is also a Bregman divergence, we are now investigating the possibility of combining divergences to adapt the classifier design to situations where the cost matrix may depend on the sample value, or even on the number of attributes. Another future line lead us to develop further study on different methods which could simplify the optimization stage, taking advantage of the properties of the designed Bregman divergences and its associated loss function.

Acknowledgements This work was partially funded by project TEC2008-01348 from the Spanish Ministry of Science and Innovation.

References

Abe, N., Zadrozny, B., & Langford, J. (2004). An iterative method for multi-class cost-sensitive learning. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 3–11). New York: ACM.

Banerjee, A., Guo, X., & Wang, H. (2005). On the optimality of conditional expectation as a Bregman predictor. *IEEE Transactions on Information Theory*, *51*(7), 2664–2669.

Bradford, J. P., Kunz, C., Kohavi, R., Brunk, C., & Brodley, C. E. (1998). Pruning decision trees with misclassification costs. In *Proceedings of the European conference on machine learning* (pp. 131–136). Berlin: Springer.

Bregman, L. M. (1967). The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, *7*(10), 200–217.

Cid-Sueiro, J., & Figueiras-Vidal, A. R. (2001). On the structure of strict sense Bayesian cost functions and its applications. *IEEE Transactions on Neural Networks*, *12*(3).

Cid-Sueiro, J., Arribas, J. I., Urbán-Muñoz, S., & Figueiras-Vidal, A. R. (1999). Cost functions to estimate a posteriori probabilities in multi-class problems. *IEEE Transactions on Neural Networks*, *10*(3), 645–656.

Dhillon, I. S., Banerjee, A., Merugu, S., & Ghosh, J. (2005). Clustering with Bregman divergences. *Journal of Machine Learning Research*, *6*, 1705–1749.

Fan, W., Stolfo, S. J., Zhang, J., & Chan, P. K. (1999). Adacost: misclassification cost-sensitive boosting. In *Proc. 16th international conf. on machine learning* (pp. 97–105). San Mateo: Morgan Kaufmann.

Gneiting, T., & Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, *102*(477), 359–378.

Guerrero-Curieses, A., Cid-Sueiro, J., Alaiz-Rodríguez, R., & Figueiras, A. (2004). Local estimation of posterior class probabilities to minimize classification errors. *IEEE Transactions on Neural Networks*, *15*(2), 309–317.

Guerrero-Curieses, A., Alaiz-Rodríguez, R., & Cid-Sueiro, J. (2005). Loss function to combine learning and decision in multiclass problems. *Neurocomputing*, *69*, 3–17.

Kapur, J. N., & Kesavan, H. K. (1993). *Entropy optimization principles with applications*. San Diego: Academic Press.

Kukar, M. Z., & Kononenko, I. (1998). Cost-sensitive learning with neural networks. In *Proceedings of the 13th European conference on artificial intelligence (ECAI-98)* (pp. 445–449). New York: Wiley.

Liu, X. Y., & Zhou, Z. H. (2006). Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Transactions on Knowledge and Data Engineering*, *18*(1), 63–77.

Lozano, A. C., & Abe, N. (2008). Multi-class cost-sensitive boosting with p-norm loss functions. In *KDD '08: proceeding of the 14th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 506–514). New York: ACM.

Marrocco, C., & Tortorella, F. (2004). *A cost-sensitive paradigm for multiclass to binary decomposition schemes*. *Lecture notes in computer science* (Vol. 3138, pp. 753–761). Berlin: Springer.

Miller, J. W., Goodman, R., & Smyth, P. (1993). On loss functions which minimize to conditional expected values and posterior probabilities. *IEEE Transactions on Information Theory*, *39*(4), 1404–1408.

O'Brien, D. B., & Gray, R. M. (2005). Improving classification performance by exploring the role of cost matrices in partitioning the estimated class probability space. In *Proceedings of the ICML workshop on ROC analysis* (pp. 79–86).

O'Brien, D. B., Gupta, M. R., & Gray, R. M. (2008). Cost-sensitive multi-class classification from probability estimates. In *ICML '08: proceedings of the 25th international conference on machine learning* (pp. 712–719). New York: ACM.

Platt, J. C. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in large margin classifiers* (pp. 61–74). Cambridge: MIT Press.

Provost, F., & Fawcett, T. (2001). Robust classification systems for imprecise environments. *Machine Learning*, *42*(3), 203–231.

Savage, L. J. (1971). Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association* (pp. 783–801).

Stuetzle, W., Buja, A., & Shen, Y. (2005). *Loss functions for binary class probability estimation and classification: Structure and applications* (Technical report). Department of Statistics, University of Pennsylvania.

Zadrozny, B., & Elkan, C. (2001a). Learning and making decisions when costs and probabilities are both unknown. In *Proceedings of the 7th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 204–213). New York: ACM.

Zadrozny, B., & Elkan, C. (2001b). Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. In *ICML '01: proceedings of the eighteenth international conference on machine learning* (pp. 609–616). San Francisco: Morgan Kaufmann.

Zadrozny, B., & Elkan, C. (2002). Transforming classifier scores into accurate multiclass probability estimates. In *KDD '02: proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 694–699). New York: ACM.

Zadrozny, B., Langford, J., & Abe, N. (2003). Cost-sensitive learning by cost-proportionate example weighting. In *ICDM '03: proc. of the 3rd IEEE int. conf. on data mining* (p. 435). Washington: IEEE Comput. Soc.