Journal of Machine Learning Research 11 (2010) 3313-3332 Submitted 1/10; Revised 6/10; Published 11/10

Maximum Likelihood in Cost-Sensitive Learning: Model Specification, Approximations, and Upper Bounds

Jacek P. Dmochowski JDMOCHOWSKI@CCNY.CUNY.EDU
Department of Biomedical Engineering
City College of New York – City University of New York
New York, NY 10031, USA

Paul Sajda PSAJDA@COLUMBIA.EDU
Department of Biomedical Engineering
Columbia University
New York, NY 10027, USA

Lucas C. Parra PARRA@CCNY.CUNY.EDU
Department of Biomedical Engineering
City College of New York – City University of New York
New York, NY 10031, USA

Editor: Charles Elkan

Abstract

The presence of asymmetry in the misclassification costs or class prevalences is a common occurrence in the pattern classification domain. While much interest has been devoted to the study of *cost-sensitive learning* techniques, the relationship between cost-sensitive learning and the specification of the model set in a parametric estimation framework remains somewhat unclear. To that end, we differentiate between the case of the model including the true posterior, and that in which the model is misspecified. In the former case, it is shown that thresholding the maximum likelihood (ML) estimate is an asymptotically optimal solution to the risk minimization problem. On the other hand, under model misspecification, it is demonstrated that thresholded ML is suboptimal and that the risk-minimizing solution varies with the misclassification cost ratio. Moreover, we analytically show that the negative weighted log likelihood (Elkan, 2001) is a tight, convex upper bound of the empirical loss. Coupled with empirical results on several real-world data sets, we argue that weighted ML is the preferred cost-sensitive technique.

Keywords: empirical risk minimization, loss function, cost-sensitive learning, imbalanced data sets

1. Introduction

Pattern classifiers make decisions; when those decisions are wrong, a loss is incurred. Thus, the ultimate goal of a classifier is to minimize the loss. When put into probabilistic terms, the mathematical expectation of the loss is called the *risk*, and is related to the classifier's error rates. In the case of a binary classification this can be written as (Duda et al., 2001):

$$\text{risk} = p(+1)\,c(+1)\,p(\text{error}|+1) + p(-1)\,c(-1)\,p(\text{error}|-1)\,, \tag{1}$$

where $c(+1)$ and $c(-1)$ denote the costs of a false negative and false positive, respectively, $p(+1)$ and $p(-1)$ are the prior probabilities for classes $y=+1$ and $y=-1$, $p(\text{error}|+1)$ is the false

negative rate, and $p(\text{error}|-1)$ is the false positive rate. Notice that the false positive and negative rates are the only terms which depend on the classifier parameters, whereas the misclassification costs and class priors are typically constants of the classification problem (later, we consider the case of example-dependent costs). The class priors are coupled with the costs of misclassification in the expression for expected loss. Thus, the risk minimization problem is uniquely defined by the ratio $\frac{p(+1)c(+1)}{p(-1)c(-1)}$; that is, even though the priors and costs may vary, as long as this ratio stays constant, the optimization problem is unchanged.

The term *cost-sensitive learning* (Elkan, 2001) has been attached to classification environments in which $c(+1) \neq c(-1)$. On the other hand, *classification with imbalanced data sets* (Chawla and Japkowicz, 2004) refers to the case where $p(+1) \neq p(-1)$. The presence of at least one of these asymmetries has been referred to by some as the "nonstandard" case (Lin et al., 2002), even though the situation is rather common in practice. In any case, these two problems may be unified simply by stating that the goal of the classification is to minimize the risk, as opposed to the conventional error rate:

$$\text{error rate} = p(+1)\, p(\text{error}|+1) + p(-1)\, p(\text{error}|-1).$$

A classifier that is designed to minimize the error rate will generally yield a high expected loss when applied to the case $c(+1) \neq c(-1)$, as the error-minimizing classifier will under-emphasize the more costly class. The problem may be exacerbated if the class prevalences are also skewed, and in the extreme case, the algorithm yields a trivial classifier which always selects the common class.

Minimizing risk is synonymous with optimally trading off the false negative and false positive rates. The trade-off between the false positive rate and false negative rate is precisely depicted by receiver operating characteristic (ROC) curves (Provost and Fawcett, 1997; Fawcett, 2004; Egan, 1975). Thus, ROC curves are well-suited to evaluating the expected loss of a classifier across the range of misclassification costs. However, "reading off" the expected loss from an ROC graph is not straightforward, and Drummond and Holte (2000) proposed cost curves as an explicit visualization of a classifier's risk for varying misclassification costs and class priors. Since the ratio $\frac{p(+1)c(+1)}{p(-1)c(-1)}$ is unbounded, the curves instead show the risk as a function of the *probability cost function* (pcf):

$$\text{pcf} = \frac{p(+1)c(+1)}{p(+1)c(+1) + p(-1)c(-1)}.$$

Cost curves facilitate the quantification of the reduction in loss offered by a cost-sensitive learning algorithm.

Several methodologies have been developed in the effort to design risk-minimizing classifiers. The simplest approach is to modify the threshold of an existing, cost-insensitive classifier. If the classifier is based on the log of the ratio of true class posterior probabilities, the threshold should be modified by a value equal to the log of the ratio of misclassification costs (Duda et al., 2001). In practice, the true class-conditional probabilities are unknown. Nevertheless, shifting the threshold by the corresponding amount has become a common heuristic (Elkan, 2001; Lin et al., 2002). Elkan (2001) proposes handling asymmetric misclassification costs by retraining the classifier on a training set in which the proportion of positive and negative examples is matched to the ratio of misclassification costs. Alternatively, if an algorithm may apply weights to the training examples, the negative examples should be weighted by a value corresponding to the asymmetry in misclassification costs. Maloof (2003) points out that although the problems of imbalanced data sets and

varying misclassification costs are separate problems, they may be tackled in very similar ways. He shows empirically that oversampling the less prevalent class (or undersampling the more prevalent class) is a procedure which yields results virtually identical to adjusting the decision threshold.

Domingos (1999) proposes a technique to relabel the training data in such a way that the relabeled data set may be trained using a standard (cost-insensitive) technique to yield a cost-sensitive classifier. The posteriors for the *labeled* examples are estimated via bagging and then used in conjunction with the Bayesian minimum risk criterion to assign new labels to the supervised data. Margineantu (2000) analyzes the approach of Domingos (1999) and suggests ways of improving the class probability estimates of the training data. Dudik and Phillips (2009) address the class imbalance problem by proposing a method which attempts to minimize loss assuming the worst-case class proportions. Masnadi-Shirazi and Vasconcelos (2010) describe a cost-sensitive version of the popular support vector machine.

Some work has been devoted to the case of example-dependent costs (Zadrozny and Elkan, 2001; Zadrozny et al., 2003). Moreover, some authors have advocated for maximizing benefits rather than minimizing costs (Elkan, 2001).

In Guerrero-Curieses et al. (2004), the authors examine loss functions which are minimized by the true class posterior probabilities; moreover, it is pointed out that the corresponding optimization algorithms should focus on training points near the decision boundary.

It is also important to point out that risk minimization is a diverse problem spanning multiple research communities; in particular, significant contributions to the problem have been made in the econometrics literature. To that end, Elliott and Lieli (2007) examine a problem analogous to cost-sensitive learning, namely the determination of a profit maximizing decision scheme by a lender. It is noted therein that to construct the minimum risk decision, the model density need not match the true density; rather, it is only required that the classifier output from the model density falls on the same side of the threshold as the classifier output using the true density. Moreover, the authors use a new loss function, namely an affine transformation of the expected utility (risk), and show an empirical advantage over traditional methods.

While a plethora of cost-sensitive methods has been investigated, it remains unclear under what conditions shifting the threshold of an existing cost-insensitive classifier is an appropriate solution. The distinction between the case of the model family including the true posterior, versus that of "misspecification" (the model does not contain the truth), has large implications on the resulting cost-sensitive learning process.

In the former case, shifting the threshold of the maximum likelihood (ML) solution is an asymptotically optimal solution to the risk minimization problem, and in the following we provide a proof of this important point. This means that when employing an expressive family which contains the true posterior, the cost-sensitive learning problem becomes one of density estimation, and the costs affect only the threshold, not the estimator. This may lead one to use a rich model set leading to complex classifiers. However, the choice to employ a simple classifier brings many advantages: ease of implementation, a lesser number of parameters to estimate, a reduced risk of over-fitting, and consequently simplified regularization procedures. Coupled with the complexity of real-world data sets, misspecified models are frequently encountered in practice. In this case, we demonstrate that thresholded ML is suboptimal, and that the minimum risk solution varies with the ratio of misclassification costs.

The problems with minimizing the true empirical risk, a non-smooth function, are well-known: for zero-one loss, the idea of smoothing out the indicator function appears in Horowitz (1992). In

this paper, we employ a sigmoidal approximation of the empirical risk to yield a novel minimizer of the loss under asymmetric misclassification cost values. Rather than argue for its optimality, this estimator is used as a basis for comparison and to argue for the relative merits of existing cost-sensitive techniques. We show analytically that the negative weighted log likelihood serves as an upper bound to the sigmoidal empirical risk. Based on the convexity of the negative weighted log likelihood and forthcoming numerical results, we will argue that weighted ML is generally the preferred technique.

2. Classification Model

In the following, we adopt a probabilistic model for the classification task: assume that the true posterior probability $p(y|\mathbf{x})$ of class $y \in \{-1,+1\}$ given received feature vector $\mathbf{x} \in \mathbb{R}^D$ is known. Let $c(y,\mathbf{x})$ denote the cost of a misclassification when the true class is y for feature $\mathbf{x}$, minus the cost of a correct prediction. Note that in general, c is feature-dependent, although in many applications, $c(y,\mathbf{x}) = c(y)$. If there is also no cost for a correct decision, then $c(y)$ is simply the cost of a false positive ($y=-1$) or false negative ($y=+1$). The optimal Bayesian decision rule is to predict $\hat{y}=+1$ if (Duda et al., 2001):

$$\frac{p(+1|\mathbf{x})}{p(-1|\mathbf{x})} > c_o(\mathbf{x}),$$

where $c_o(\mathbf{x}) = \frac{c(-1,\mathbf{x})}{c(+1,\mathbf{x})} > 0$. The optimal decision rule may be written as:

$$\hat{y}(\mathbf{x}) = \operatorname{sgn}\left[f(\mathbf{x}) - \ln c_o(\mathbf{x})\right], \tag{2}$$

where $\hat{y}(\mathbf{x})$ is the predicted class given feature vector $\mathbf{x}$, sgn is the signum function $\operatorname{sgn}(x) = \begin{cases} 1 & x > 0 \\ -1 & x \leq 0 \end{cases}$, and $f(\mathbf{x})$ is the discriminant function:

$$f(\mathbf{x}) = \ln \frac{p(+1|\mathbf{x})}{p(-1|\mathbf{x})}.$$

It should be noted that the argument of the signum function may be written in log units due to the nonnegativity of the ratio of posteriors and the optimal threshold $c_o(\mathbf{x})$.

In practice, we do not have access to the true class posteriors, but rather estimate their values from available training data. The estimate is denoted by $p(y|\mathbf{x};\theta)$, where $\theta \in \Theta$ is a vector parameterizing the *model* posterior, and Θ is termed the model set. If the true posterior is in the model set, denote the true value of θ by θ^*, such that $p(y|\mathbf{x}) = p(y|\mathbf{x};\theta^*)$. The model discriminant is written as $f(\mathbf{x},\theta) = \ln \frac{p(+1|\mathbf{x};\theta)}{p(-1|\mathbf{x};\theta)}$, and the classifier takes the form:

$$\hat{y}(\mathbf{x},\theta) = \operatorname{sgn}\left[f(\mathbf{x},\theta) - \ln c_o(\mathbf{x})\right]. \tag{3}$$

This paper is concerned with methods of estimating θ to minimize risk, and their relation to the specification of the model set. In order to treat these estimation methods, we briefly outline the risk minimization framework which allows for the subsequent analysis of the various cost-sensitive loss functions.

3. Risk Minimization for Cost-Sensitive Learning

Risk minimization is concerned with choosing a function from a set $\{\hat{y}(\mathbf{x},\theta),\theta\in\Theta\}$ to minimize the resulting *risk functional*

$$R(\theta)=\int\int L(y,\mathbf{x},\theta)\,p(\mathbf{x},y)\,d\mathbf{x}dy,$$

where $L(y,\mathbf{x},\theta)$ quantifies the loss incurred by the classifier $\hat{y}(\mathbf{x},\theta)$ in response to labeled data $(\mathbf{x},y)$. Note that the loss L varies with the feature $\mathbf{x}$. To ease notation throughout the rest of the paper, the dependence of $\hat{y}$ on $\mathbf{x}$ and θ is implied.

The problems of regression, density estimation, and pattern recognition may all be formulated within the context of risk minimization, simply by altering the loss function L, as outlined in Vapnik (1998, 1999). In the case of error-minimizing pattern recognition, the classical zero-one loss function is given by:

$$L(y,\mathbf{x},\theta)=\mathbb{1}(y\neq\hat{y}),$$

where $\mathbb{1}(\Phi)$ is the indicator function which equals one when Φ is true and zero otherwise.

Since we do not have access to the true density $p(\mathbf{x},y)$, the empirical risk minimization (ERM) approach substitutes the empirical density:

$$p_{\text{emp}}(\mathbf{x},y)=\frac{1}{N}\sum_{n=1}^{N}\mathbb{1}(\mathbf{x}=\mathbf{x}_n)\,\mathbb{1}(y=y_n),$$

where $\mathcal{D}=(\mathbf{x}_n,y_n)_{n=1}^{N}$ is a set of N labeled observations which are independent and identically distributed samples drawn from the true joint density $p(\mathbf{x},y)$, leading to the following expression for the *empirical risk*:

$$R_{\text{emp}}(\theta)=\frac{1}{N}\sum_{n=1}^{N}L(y_n,\mathbf{x}_n,\theta). \tag{4}$$

In order to design a cost-sensitive classifier, a loss function modeling the asymmetry in misclassification costs is required. Several alternatives exist. In the following subsections, we describe these loss functions.

3.1 Thresholded Maximum Likelihood

The traditional (cost-insensitive) ML loss function is given by Vapnik (1998):

$$L(y,\mathbf{x},\theta)=-\ln p(y|\mathbf{x};\theta),$$

leading to the following expression for the empirical risk:

$$R_{\text{emp}}^{\text{ml}}(\theta)=-\frac{1}{N}\sum_{n}\ln p(y_n|\mathbf{x}_n;\theta). \tag{5}$$

The minimizer of (5) is the well-known ML estimate (Duda et al., 2001):

$$\begin{aligned}\hat{\theta}_{\mathrm{ML}} &= \arg\max_{\theta\in\Theta} \ln \prod_{n=1}^{N} p\left(y_n|\mathbf{x}_n;\theta\right) \\ &= \arg\max_{\theta\in\Theta} \ln \prod_{n=1}^{N} \frac{1}{1+\frac{p\left(-y_n|\mathbf{x}_n;\theta\right)}{p\left(y_n|\mathbf{x}_n;\theta\right)}} \\ &= \arg\max_{\theta\in\Theta} \sum_{n=1}^{N} \ln\left[\frac{1}{1+e^{-y_n f\left(\mathbf{x}_n,\theta\right)}}\right],\end{aligned}$$

where the second step follows from Bayes' rule. If the model set Θ contains the true parameter θ^*, it follows that in the asymptotic limit, we have $\lim_{N\to\infty}\hat{\theta}_{\mathrm{ML}} = \theta^*$ (Kay, 1993). From (2) and (3), if we have knowledge of θ^*, then a threshold shift of $\ln c_o(\mathbf{x})$ yields the optimal classifier. Assuming continuity of the log likelihood (in θ), we have that $\lim_{N\to\infty}\hat{y}\left(\mathbf{x},\hat{\theta}_{\mathrm{ML}}\right) = \hat{y}\left(\mathbf{x},\theta^*\right)$, and thus the thresholded ML estimate yields the minimum risk decision rule for all cost ratios. Once the ML estimate is available, the cost-sensitive classifier for any cost ratio may be formed by appropriately adjusting the threshold. There is no need to retrain the classifier if the cost ratio changes. In the case of a generalized linear model for $p(y|\mathbf{x},\theta)$, it may easily be shown (McCullagh and Nelder, 1989; Parra et al., 2005) that the risk function is convex, and an iteratively reweighted least squares (IRLS) algorithm locates the optimal linear classifier often within a few iterations.

Unfortunately, in many real-world classification problems, the model set (for example, the set of all hyperplanes) does not contain the true posterior. Notice, for example, that even in the simple case of Gaussian data, the linear discriminant is only optimal in the case of equal covariance matrices; nevertheless, linear classifiers are heavily used. (For a comprehensive treatment of misspecified models in ML, refer to White, 1982.) In such cases, the classifier in the model set which minimizes risk will vary with the pcf. As a result, a shift in threshold of the ML solution will yield a sub-optimal classifier.

3.2 Example: Minimum Risk Hyperplane for Gaussian Data

To illustrate this point, we consider the instructive case of Gaussian densities with unequal covariances and a linear classifier function. The purpose of this exercise is not to argue for a simple Gaussian model or a linear classifier but rather to demonstrate in an analytically tractable case the problem that arises with thresholded ML when the model is misspecified. It is assumed that $c\left(y,\mathbf{x}\right) = c(y)$.

Consider a linear classifier of the form $f\left(\mathbf{x};\theta\right) = \theta^T\mathbf{x} - b$, and assume Gaussian class-conditional densities:

$$p\left(\mathbf{x}|y\right) = \frac{1}{\left(2\pi\right)^{D/2}\left|\Sigma_y\right|^{1/2}} e^{-\frac{1}{2}\left(\mathbf{x}-\mu_y\right)^T\Sigma_y^{-1}\left(\mathbf{x}-\mu_y\right)},\ y\in\{-1,+1\}.$$

Note that by the normality of $\mathbf{x}|y$, $\theta^T\mathbf{x} \sim \mathcal{N}\left(\theta^T\mu_y, \theta^T\Sigma_y\theta\right)$. Thus, we have:

$$\begin{aligned}p\left(\text{error}|y\right) &= p\left[y\left(\theta^T\mathbf{x}-b\right)<0\right] \\ &= \frac{1}{2}\left[1+y\cdot\operatorname{erf}\left(\frac{b-\theta^T\mu_y}{\sqrt{2\theta^T\Sigma_y\theta}}\right)\right].\end{aligned} \tag{6}$$

Substituting (6) into (1), the expression for the expected loss takes the form:

$$R(\theta,b) = \frac{c(+1)p(+1)}{2}\left[1+\operatorname{erf}\left(\frac{b-\theta^T\mu_+}{\sqrt{2\theta^T\Sigma_+\theta}}\right)\right] + \frac{c(-1)p(-1)}{2}\operatorname{erfc}\left(\frac{b-\theta^T\mu_-}{\sqrt{2\theta^T\Sigma_-\theta}}\right). \tag{7}$$

The optimal hyperplane is the one which minimizes the risk: $\arg\min_{\theta,b} R(\theta,b)$.

Below, we illustrate an example where the parameters of the data are given by:

$$\mu_+ = \begin{bmatrix} 0.5 & 0 \end{bmatrix}^T,\ \mu_- = \begin{bmatrix} 0 & 0.5 \end{bmatrix}^T,\ \Sigma_+ = \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix},\ \Sigma_- = \begin{bmatrix} 0.5 & 0 \\ 0 & 1 \end{bmatrix}.$$

In this case of unequal covariance, the optimal ML classification function is not linear but instead a quadric (Duda et al., 2001, Chapter 2).

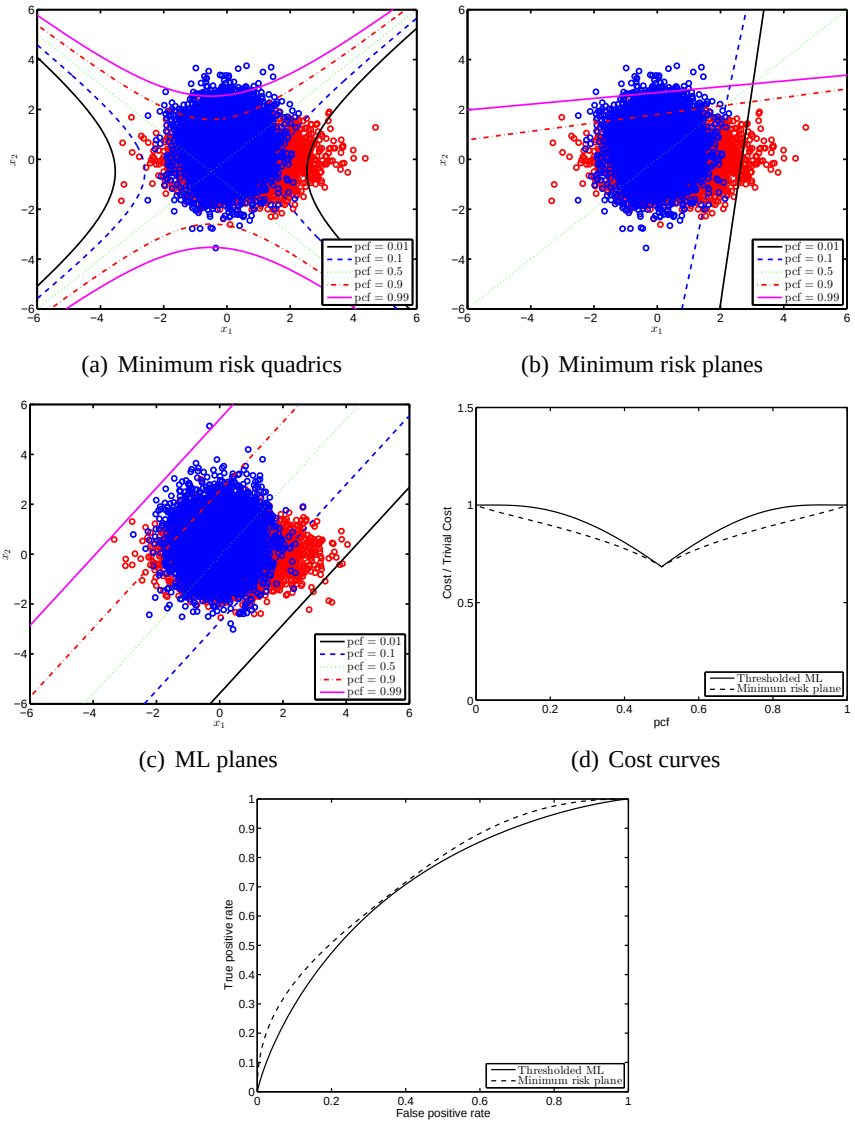
Figure 1(a) displays the minimum risk quadrics for various values of the pcf, where it is assumed that $p(+1) = p(-1)$. The quadrics are related to each other via a threshold shift. On the other hand, Figure 1(b) depicts the minimum risk planes for the same data, which were computed by minimizing (7) using numerical optimization techniques. It is clear that the direction of the optimal plane is a function of the pcf, and a threshold shift of the minimum error plane is not an optimal solution to the risk minimization problem. Figure 1 (c) displays the threshold-shifted ML solutions for various values of the pcf. The suboptimality of the ML approach is readily apparent by contrasting Figure 1(b) with Figure 1(c). Notice that at the extremes of the pcf, the optimal planes are orthogonal to each other. Meanwhile, the ML plane has unit slope for all pcf. The risks obtained by applying the ML and minimum risk planes to the given data are shown in Figure 1(d). In the figure, we normalize the raw risk of (1) by the "trivial risk", which is defined as the risk achieved by the scheme:

$$\hat{y}_{\text{trivial}} = \operatorname{sgn}\left[p(+1)c(+1) - p(-1)c(-1)\right].$$

We call this the "trivial risk" because the decision rule is feature-independent and is strictly a function of the class priors and misclassification costs. A normalized risk less than 1 indicates that the classification scheme yields a "savings" over the a priori decision rule. The normalization allows us to quantify the "percentage of savings" achieved by employing a "smart" decision rule.

The curves were generated by averaging over 1000 ensembles, where each ensemble consisted of $N = 1000$ training samples. The ML classifier was trained on each ensemble and the resulting risk computed by substituting the solution into (7). The risk margin between the threshold-shifted ML solution and that of the minimum risk plane is what is "available" for cost-sensitive learning algorithms to improve upon. These methods attempt to learn, for each pcf, the minimum risk plane shown in Figure 1(b), to achieve the dashed cost curve in Figure 1(d).

The difference between the threshold-shifted ML and cost-sensitive paradigms may be understood in terms of ROC analysis—Figure 1 (e) depicts the ROC curves for the thresholded ML and minimum risk classifiers. In the ML method, the ROC curve is generated by sweeping the threshold of the base classifier across the real line and computing the corresponding error rates. In the cost-sensitive paradigm, each point on the ROC curve corresponds to a distinct classifier which is computed by minimizing (7) for a specific ratio of misclassification costs, resulting in values for the true and false positive rates. Note that one may also produce a *family* of ROC curves by sweeping the threshold of each of these distinct cost-sensitive classifiers, although this is not shown in the figure.



(a) Minimum risk quadrics

(b) Minimum risk planes

(c) ML planes

(d) Cost curves

(e) ROC curves

Figure 1: Minimum risk classification of Gaussian data with unequal covariance matrices.

3.3 Relabeled Examples

One heuristic to cost-sensitive classification is to modify the training labels to achieve a balance in class prevalence. In terms of an ERM loss function, this may be written as:

$$L(y, \mathbf{x}, \theta) = -\ln p\left[g(\mathbf{x})|\mathbf{x};\theta\right],$$

where the function $g(\mathbf{x}) : \mathbb{R}^D \rightarrow \{-1, +1\}$, produces a new label for each training sample according to some criterion. Domingos (1999) proposes MetaCost, which reassigns labels according to the

Bayesian minimum risk criterion:

$$g(\mathbf{x}) = \arg \max_{y \in \{-1,+1\}} c(y) \hat{p}(y|\mathbf{x}), \tag{8}$$

where $\hat{p}(y|\mathbf{x})$ is an estimate of the class posterior probability which is obtained using bagging (Breiman, 1996). It is typically the examples near the boundary which are re-labeled. The empirical risk follows as:

$$R^{\text{rel}}_{\text{emp}}(\theta) = -\frac{1}{N} \sum_n \ln p\left[g(\mathbf{x}_n)|\mathbf{x}_n;\theta\right].$$

As the re-labeling of (8) varies with the ratio of misclassification costs, the resulting cost-sensitive classifier is a function of the pcf and thus has the ability to yield the minimum risk estimator. The success of the method hinges on the estimation of the posterior probabilities; in the best case scenario, the re-labeling results in the cost-sensitive boundary approaching the minimum risk boundary. In contrast to the forthcoming methods, MetaCost does not reweight examples, and thus the risk function will not dominated by the examples of a rare but costly class. The technique may be used as a cost-sensitive pre-processing to any classification technique, and not just ML estimation of the posterior. In the case of ML, we maximize the log-likelihood but with the altered labels.

3.4 Weighted Likelihood

A standard approach for developing cost-sensitive classifiers is to weight the training examples according to the "costliness" of misclassifying that example. This procedure may be viewed in terms of an ERM loss function (Elkan, 2001; Zadrozny et al., 2003):

$$L(y,\mathbf{x},\theta) = -c(y,\mathbf{x}) \ln p(y|\mathbf{x};\theta),$$

such that the corresponding empirical risk takes the form:

$$R^{\text{wml}}_{\text{emp}}(\theta) = -\frac{1}{N} \sum_n c(y_n,\mathbf{x}_n) \ln p(y_n|\mathbf{x}_n;\theta). \tag{9}$$

Weighting the log likelihood has previously been studied as a tool to handle misspecification (Shimodaira, 2000). Note that such weighting of examples is equivalent to modifying the proportion of examples in the training set according to the weightings $c(y,\mathbf{x})$. If these weightings change, so does the cost function, and thus the classifier needs to be retrained. In principle, this technique allows the classification to choose the model in Θ which minimizes risk for the specified cost matrix. Moreover, example-weighting may easily be incorporated into the IRLS algorithm, yielding an iterative reweighted *weighted* least-squares scheme (McCullagh and Nelder, 1989) which minimizes (9)—in the appendix, we provide a MATLAB implementation.

Notice that if the misclassification costs are highly asymmetric, the more "costly" examples will be heavily emphasized in the empirical risk function. Furthermore, if there are only a few such examples, the classifier is at an increased risk of overfitting, since the *effective* number of examples is much less than N. This issue plagues any cost-sensitive method which weights the examples based on cost.

3.5 Sigmoidal Empirical Risk

In order to relate the risk of (1) with the empirical risk (4), the appropriate loss function is found to be:

$$L(y,\mathbf{x},\theta) = c(y,\mathbf{x})\mathbb{1}(\hat{y} \neq y).$$

Strict equivalence is achieved in the simplifying case of $c(y,\mathbf{x}) = c(y)\mathbb{1}(\hat{y} \neq y)$ assuming that there is no cost for a correct decision. Generally, the empirical risk follows as:

$$\begin{aligned} R_{\text{emp}}(\theta) &= \text{const.} + \frac{1}{N}\sum_n c(y_n,\mathbf{x}_n)\,\mathbb{1}(\hat{y}_n \neq y_n) \\ &= \text{const.} + \frac{1}{N}\sum_n c(y_n,\mathbf{x}_n)\,u[-y_n f(\mathbf{x}_n,\theta)] \end{aligned} \tag{10}$$

where the constant is a summation across the costs of a correct decision and $u(x)$ is the step function: $u(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases}$. Since our goal is to minimize (1), the optimization of the empirical risk under the direct loss of (10) is of great importance.

Elliott and Lieli (2007) propose to optimize a function closely related to (10) in an econometric context; employing the notation of this paper, the objective function maximized by Elliott and Lieli (2007) is written as:

$$R_{\text{el}}(\theta) = \sum_{n=1}^{N} y_n c(y_n,\mathbf{x}_n)\,\text{sgn}[f(\mathbf{x}_n,\theta) - \ln c_o(\mathbf{x}_n)], \tag{11}$$

and the authors propose simulated annealing to perform the optimization.

Note that both objective functions (10) and (11) are not differentiable due to the non-smoothness of u and sgn at zero, respectively. In the case of (10), we may approximate the step function with a sigmoid :

$$u(x) \approx \frac{1}{1+e^{-x}}. \tag{12}$$

Substituting (12) into (10), we obtain the following expression for the approximate empirical risk:

$$\tilde{R}_{\text{emp}}(\theta) = \frac{1}{N}\sum_n c(y_n,\mathbf{x}_n)\frac{1}{1+e^{y_n f(\mathbf{x}_n,\theta)}}.$$

The classifier θ which minimizes the empirical risk follows as:

$$\hat{\theta} = \arg\min_{\theta} \tilde{R}_{\text{emp}}(\theta). \tag{13}$$

The advantage of this approach is that it closely approximates (up to the ability of the sigmoid to approximate a step) the true empirical risk. On the other hand, the risk function is non-convex, complicating the minimization of (13).

3.6 Relating Sigmoidal Risk to Weighted ML

The need to minimize non-convex functions arises often in practice. A standard trick in optimizing a non-convex function is to optimize a convex upper bound of the original function. In this subsection, we show that the negative weighted log likelihood, a convex function, provides a tight upper bound of the sigmoidal empirical risk.

To see this, note the inequality:

$$z \leq -\ln(1-z), \;\; z \leq 1. \tag{14}$$

Substituting $z = \frac{1}{1+e^{y_n f(\mathbf{x}_n,\theta)}}$ into (14) results in:

$$\frac{1}{1+e^{y_n f(\mathbf{x}_n,\theta)}} \leq -\ln \frac{1}{1+e^{-y_n f(\mathbf{x}_n,\theta)}}.$$

Combining these inequalities over the training examples and assuming strict positivity of the weights $c(y_n,\mathbf{x}_n)$, we obtain:

$$\sum_n c(y_n,\mathbf{x}_n) \frac{1}{1+e^{y_n f(\mathbf{x}_n,\theta)}} \leq -\sum_n c(y_n,\mathbf{x}_n) \ln \frac{1}{1+e^{-y_n f(\mathbf{x}_n,\theta)}}.$$

As a result,

$$\tilde{R}_{\text{emp}}(\theta) \leq R^{\text{wml}}_{\text{emp}}(\theta).$$

This means that minimizing the weighted negative log-likelihood (9) minimizes an upper bound on the empirical risk. As will be shown numerically with upcoming examples, this bound is fairly tight (c.f., Figures 2 and 3). Since the negative weighted log-likelihood is convex, to circumvent the non-convexity of the sigmoidal empirical risk, one option is to employ the weighted likelihood loss function.

4. Experimental Evaluation

To assess the performance of the various cost-sensitive approaches (and its dependence on N), and to support the upper bound relationship of weighted ML to sigmoidal risk, we conducted an empirical evaluation of the various cost-sensitive learning approaches on several data sets. We first consider the case of example-independent and synthetic (i.e., exogenous to the features) costs. Later, we examine a data set where costs are endogenous and depend on the features. We employed a linear model set of the form $f(\mathbf{x},\theta) = \theta^T\mathbf{x} - b$. For all data sets, five-fold cross-validation was employed, and we plot the mean loss over the N examples (each example is used once for validation) along with standard errors of the mean.

The prevalence of the positive class is data-set dependent. The pcf was varied from 0.1 to 0.9 in increments of 0.1. The relationship between the misclassification cost ratio and the pcf is given by:

$$\frac{c(-1)}{c(+1)} = \frac{p(+1)}{p(-1)} \frac{(1-\text{pcf})}{\text{pcf}}.$$

Thus, a pcf of 0.5 corresponds to the case where the ratio of misclassification costs is inversely related to the ratio of class priors [i.e., $c(-1)p(-1) = c(+1)p(+1)$].

The generalization ability of all algorithms benefited from ℓ_2 regularization. Thus, the problem of cost-sensitive learning becomes one of penalized ERM:

$$\hat{\theta} = \arg\min_{\theta} \left\{ R_{\text{emp}}(\theta) + \frac{\lambda}{2} \|\theta\|^2 \right\}. \tag{15}$$

Where computationally feasible, the value of λ was determined using a nested cross-validation loop which tunes λ on the training set; the tuned value is then fed up to the outer cross-validation loop which evaluates performance on the test set.

The implementation of all cost-sensitive learning methods requires solving the optimization problem (15). For the 3 likelihood based methods, the Newton-Raphson IRLS algorithm (McCullagh and Nelder, 1989) was employed to solve the optimization. In order to solve the minimum risk optimization of (13), the Broyden-Fletcher-Goldfarb-Shanno (BFGS) quasi-Newton method (Fletcher, 2000) was employed in conjunction with multiple random restarts: a random starting vector is chosen, the BFGS algorithm is run, and the training risk evaluated. This process is repeated 100 times, and the final solution is selected as the classifier which yields the lowest training risk among the runs.

4.1 Gaussian Data

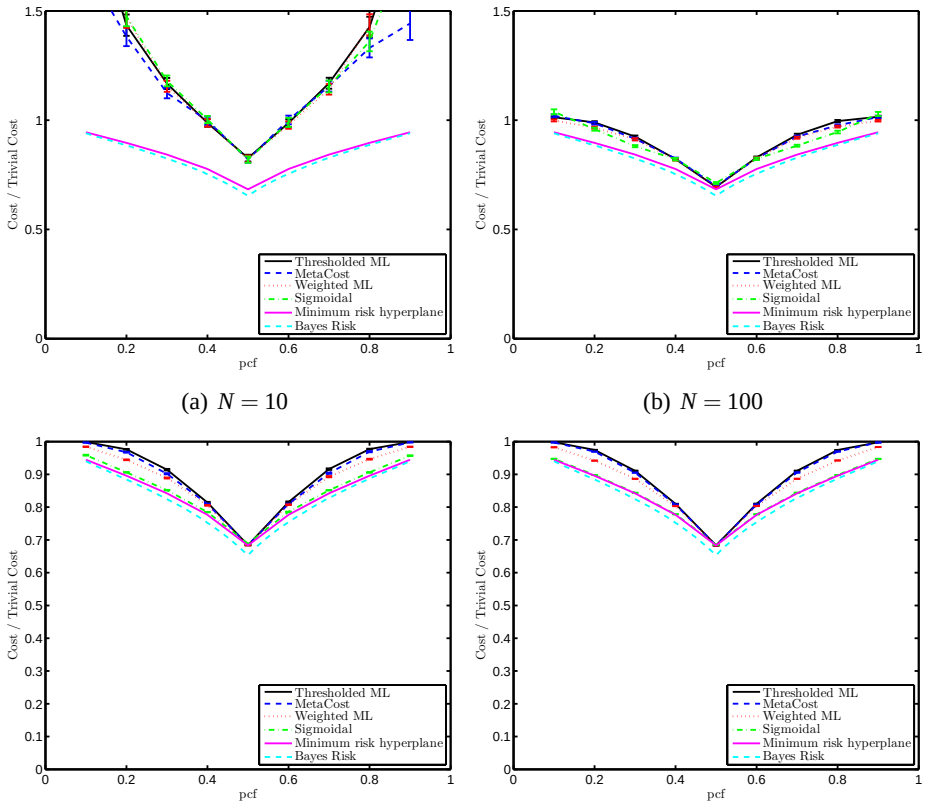
Before delving into real-world data sets, we evaluated the various cost-sensitive approaches on the Gaussian data described in Section 3. This example is instructive as we have a closed-form expression for the minimum attainable value of risk, and thus can evaluate the convergence properties with an increased sample size. Figure 2 depicts the cost curves[1] for various training sizes N. At $N = 10$, there is a substantial loss margin between the minimum risk plane and that which is achieved by the thresholded ML technique. However, it is clear that the cost-sensitive techniques are not able to provide reliable estimates of the minimum risk plane direction with such limited data. As the size of the training set increases, the sigmoidal risk estimator converges to the minimum risk plane. Notice, however, that with such large sample sizes, the thresholded ML technique is relatively adept at yielding risk values comparable to the true minimum. The reason for this is that the examples which are misclassified by thresholded ML and classified correctly by the the other techniques are mostly the low-cost examples (compare Fig. 1(b) with Fig. 1(c), for example). Also shown in all plots is the Bayes risk, which is the risk attained by the minimum risk quadric.

4.2 UCI Data

Next, we evaluate the classifiers on several real-world data sets obtained from the UCI database (Asuncion and Newman, 2007) as well as our previous work on the classification of electroencephalographic (EEG) data in a real-time detection task (Parra et al., 2008).[2] Table 1 summarizes the parameters used in the evaluation of these data sets.

From Figs. 3 (a) and (b), it is once again apparent that given a modest value of N, the benefits provided by cost-sensitive learners over the thresholded ML approach are not substantial. However, in Figs. 3 (c) and (d), one observes a tangible loss reduction of the sigmoidal risk estimator over

1. In addition to the ensemble-averaged risk, we also report standard errors of the mean, which follow as the sample standard deviation of the ensemble-averaged mean, divided by the square root of the number of ensembles.
2. Since only one "ensemble" is available in the experiments with real data, we treat the cost (at test time) of each example as an iid realization of the risk and report standard errors of the mean across examples.



(a) $N = 10$ (b) $N = 100$

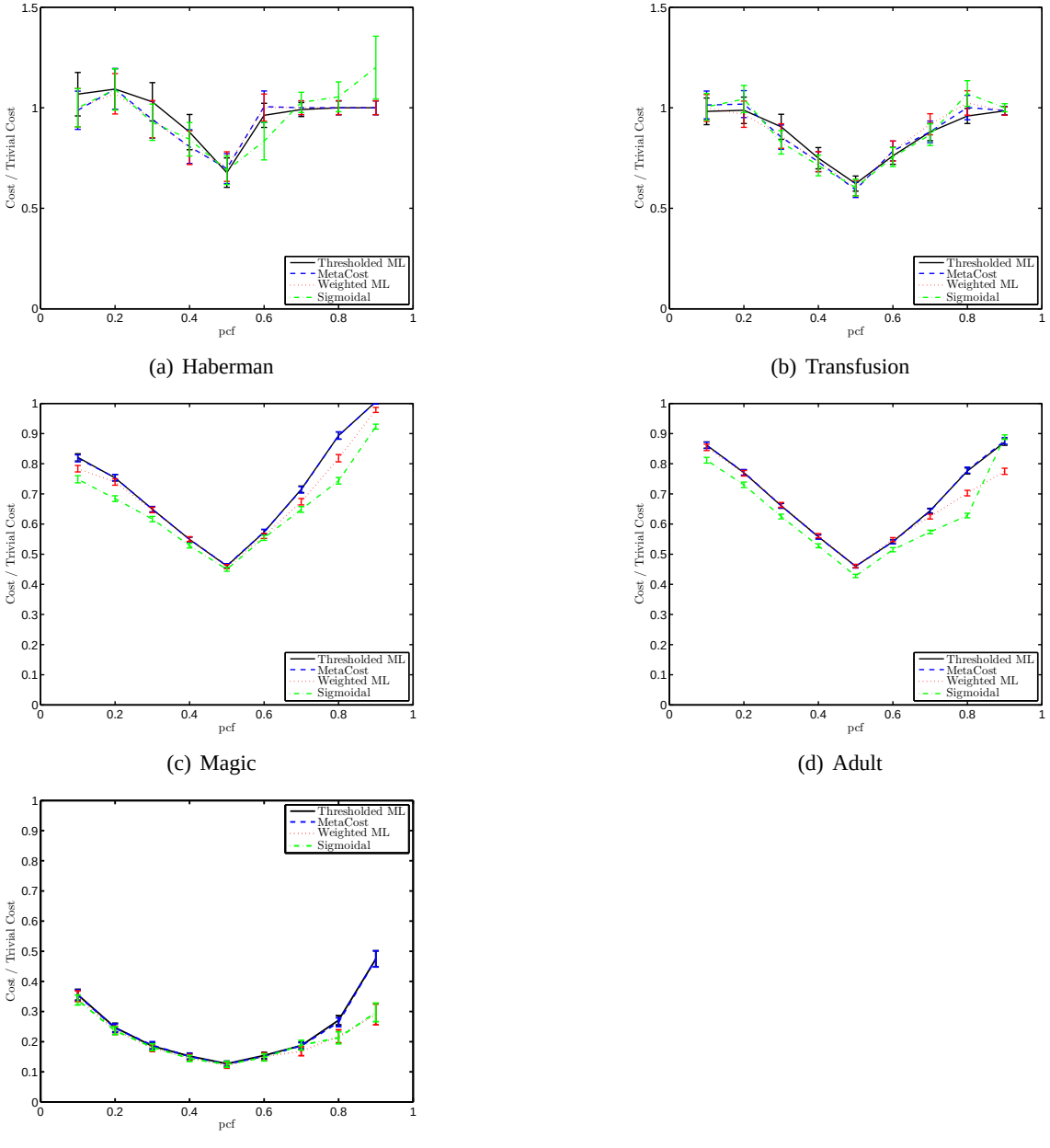
(c) $N = 1000$ (d) $N = 10000$

Figure 2: Cost curves for Gaussian data with varying training set sizes. In (d), the thresholded ML and MetaCost curves are nearly equivalent.

Data set	$N(+)$	$N(-)$	D	λ	$p(+1)$
Haberman	81	225	3	optimized	0.26
Transfusion	178	570	4	optimized	0.24
Magic	6688	12332	10	fixed (0.2)	0.35
Adult	7841	24720	14	fixed (0.2)	0.24
EEG	830	40608	15	fixed (2)	0.02

Table 1: Data set and regularization parameters. $N(+)$ and $N(-)$ refer to the number of positive and negative examples, respectively.

thresholded ML and MetaCost, whose curves overlap. Lastly, Fig. 3 (e) demonstrates the near-optimality of weighted ML and its close approximation of the sigmoidal risk minimizing solution. Note that for this heavily skewed data set, while the total number of examples is $N = 41438$, only 830 of these are positive exemplars. Note also that a skew in class prevalence leads to asymmetry in the resulting cost curves.



(a) Haberman

(b) Transfusion

(c) Magic

(d) Adult

(e) EEG

Figure 3: Cost curves (with standard errors of the means) for various real data sets with synthetic costs.

4.3 German Banking Data

Finally, we evaluated the risk minimizing classifiers on a publicly available data set collected by a German bank: http://www.stat.uni-muenchen.de/service/datenarchiv/kredit/kredit.html. The data set details the credit history and biographical information of $N = 1000$ past loan applicants, as well

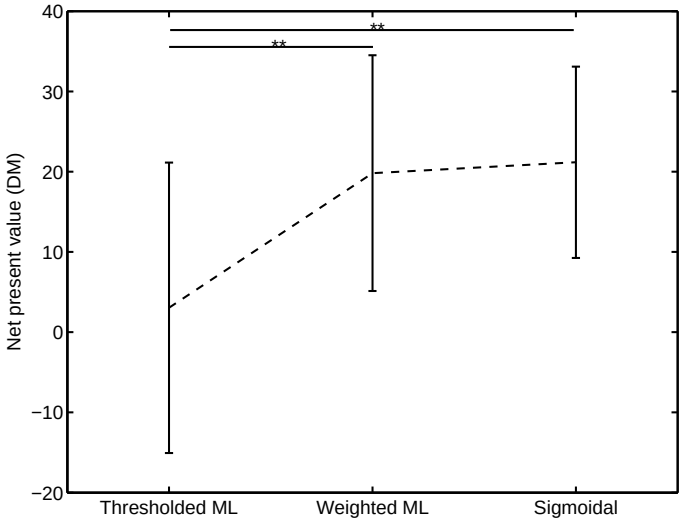


Figure 4: NPV per applicant: means and standard errors. Solid horizontal lines indicate statistical significance at the $p = 0.01$ level.

as whether the loan was repaid. This data set has been used previously to evaluate a novel profit-maximizing decision rule in the econometrics literature (Lieli and White, 2010).

Upon receiving a loan request, the bank decides either to grant the loan at a certain interest rate, or rather to invest the loan amount in a risk-free government bond. In this application, it is easier to work with benefits rather than costs; as such, the net present value (NPV), measured in Deutsche Marks (DM) of extending the loan must be compared with the NPV of rejecting the application, which is zero as outlined in Lieli and White (2010). The NPV of extending the loan depends on whether the loan is repaid, and thus the optimal decision rule takes into account the probability of repayment as well as the potential profit to be made on the loan. Thus, the aim of the evaluation is to predict, from an applicant's credit history and biographical information, the likelihood of the applicant repaying the loan which he/she is seeking.

In the framework described in the earlier sections, we have $c(y,\mathbf{x}) = y \cdot \pi(y,\mathbf{x})$, where $\pi(y,\mathbf{x})$ denotes the NPV associated with predicting $\hat{y} = +1$ when the truth is y (repayment: $y = +1$, default: $y = -1$). Please refer to Lieli and White (2010) for the precise mathematical relationship between the NPV π and the individual features in $\mathbf{x}$. In the evaluation, we used the $D = 5$ dimensional feature set chosen by Lieli and White (2010), as well as their proxies for the interest and risk-free government rates. Note that the "costs" are both example-dependent and endogenous to the problem. We conducted a leave-one-out cross-validation of the thresholded ML, weighted ML, and sigmoidal risk estimators on this $N = 1000$ example data set (MetaCost is not applicable to problems with example-dependent costs).

Figure 4 displays the mean NPV per applicant, along with standard errors. On average, the means obtained by the thresholded ML, weighted ML, and sigmoidal risk estimators are DM 3.0 , DM 19.8 , and DM 21.2 . The standard errors are given by DM 18.1 , DM 14.7, and DM 11.9, respectively (a negative value of NPV indicates an overall loss for the classification scheme). We

performed pairwise sign-tests of statistical significance to determine if performance differs significantly for the classifiers. A statistically significant improvement in NPV is achieved by both WML and sigmoidal risk estimators over the thresholded ML solution ($p < 0.01$). On the other hand, statistical significance cannot be established between WML and sigmoidal risk ($p \approx 0.9$). This empirical finding supports the analytical result of negative weighted log likelihood upper bounding empirical loss.

5. Discussion

It is interesting to point out the process which led to the findings presented in this paper. Initially, we were motivated by the observation that with a misspecified model, the direction of the minimum risk hyperplane is a function of the ratio of misclassification costs and the class priors. Since the ML approach is inherently to shift the minimum error hyperplane, we sought to develop an estimator which, given a ratio of misclassification costs, will find the direction required to minimize risk rather than maximizing likelihood. The expectation was that such an estimator would provide large performance gains over the ML approach. This led us to the development of the sigmoidal empirical risk minimizer. During the algorithm evaluation process, several findings emerged. Firstly, the search for the minimum risk hyperplane is non-trivial: regularization techniques proved to be necessary, particularly in the case of a limited training set. Moreover, both the existing and proposed cost-sensitive learning techniques yield the greatest benefits over thresholded ML when presented with large amounts of training data. When abundant data is available, the sigmoidal risk estimator typically outperforms all other methods, but weighted ML yields quite comparable values.

When the model set includes the true posterior, the threshold-shifted ML approach is optimal. This naturally brings us to the following question: why not employ a rich model set (for example, a multi-layer neural network), estimate its parameters using ML, and then shift the threshold by the log of the misclassification cost ratio? With an infinite amount of training data, we are sure to arrive at the lowest attainable risk. However, there are a few reasons why this procedure may not be desirable: a rich model set consists of many parameters, which in turn requires a large amount of training data to prevent over-fitting. From the so-called *structural risk minimization* principle, it is well-known that a simpler model set yields empirical risks that are closer to the true risk (Vapnik, 1998). Moreover, the optimality of the ML solution is not guaranteed for a finite amount of data. Thus, rates of convergence are key to determining the best approach.

In general, the choice of model complexity hinges upon several factors: the dimensionality of the feature space in relation to the number of available examples, the signal-to-noise ratio, and also the skew in class prevalence. For example, in applications involving a rare and expensive class, the key is to yield accurate decisions for this infrequent class. If the number of such examples is low, then even if the number of overall examples is high, a complex model will generally be undesirable. In other words, the effective sample size is closer to the number of costly examples than the entire sample size N. Consequently, the number of free parameters needs to be limited to prevent overfitting. The design issue in cost-sensitive learning is thus how best to use these few degrees of freedom: whether to "prioritize" correct decisions on the costly training examples, or rather to "spend" the degrees of freedom on achieving the best model fit.

The results with Gaussian data presented above appear to indicate that the sigmoidal risk minimizer tends to the true minimum risk model given enough data. However, the weighted ML estimator provides a tight upper bound on the sigmoidal empirical risk and thus this solution is not far

from optimal. Given that the negative weighted likelihood is convex, weighted ML thus becomes the preferred cost-sensitive technique.

Lastly, as seen in Figure 2, both the opportunity and the challenge in cost-sensitive learning lies in the ability to estimate the minimum risk model with limited data. Thus, the focus going forward should be on sparse (i.e., with few examples relative to the dimensionality of the feature space) inference of the minimum risk classifier.

6. Conclusion

This paper has elucidated the role of the specification of the model set in the problem of learning with asymmetric costs or class prevalences. It was shown that in the case of the model family including the true posterior, thresholding the ML solution is guaranteed to asymptotically minimize risk. In this case, cost-sensitive learning is synonymous with threshold adjustment. On the other hand, with a misspecified model, the risk minimizing solution is a function of the misclassification cost ratios, and thresholding the ML estimate is sub-optimal. A novel estimator based on a sigmoidal estimation of the empirical risk was presented and shown to outperform conventional techniques provided enough data; however, the negative weighted log likelihood was analytically and empirically shown to tightly upper bound the sigmoidal loss. Thus, we advocated for the weighted ML as the preferred cost-sensitive learning technique.

Acknowledgments

The authors are grateful to the editor and all of the anonymous reviewers for each providing numerous useful comments and suggestions. We would also like to thank Barak Pearlmutter for some insightful discussions.

This work was supported by The Defense Advanced Research Projects Agency (government contract no. NBCHC080029). We would also like to acknowledge support for this project from the National Science and Engineering Research Council of Canada (NSERC).

The views, opinions, and/or findings contained in this article are those of the authors and should not be interpreted as representing the official views or policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the Department of Defense.

Appendix A. MATLAB Code for Weighted Maximum Likelihood

Provided below is a basic MATLAB implementation of the weighted ML approach to cost-sensitive learning using iteratively reweighted weighted least squares. For a more detailed version as well as implementations of thresholded ML and sigmoidal risk minimization, please refer to `http://bme.ccny.cuny.edu/faculty/lparra/cost`.

```
function v = wml(x,y,c,lambda)
% x - N-by-D matrix of input samples \in (-inf,inf)
% y - N-by-1 vector of binary labels \in {0,1}
% c - N-by-1 vector of costs \in (0,inf)
% lambda - regularization parameter \in (0,inf) (defaults to 0)
% v - v(1:D) normal to separating hyperplane, v(D+1) threshold
% (c) Lucas C. Parra, Jacek P. Dmochowski
```

```
if nargin<4; lambda=0; end;
[N,D]=size(x);
s = std(x); x = x./repmat(s,[N 1]);
x = [x ones(N,1)];
v = zeros(D+1,1);
lambda = [0.5*lambda*ones(1,D) 0]';
while 1
    vold=v;
    mu = exp(x*v - log(1+exp(x*v)));
    w = ( mu.*(1-mu) ).*c;
    e = (y - mu).*c;
    grad = x'*e - lambda .* v;
    inc = inv(x'*(repmat(w,1,D+1).*x)+diag(lambda)) * grad;
    v = v + inc;
    if norm(vold) & subspace(v,vold)<10^-10, break, end;
end;
v(1:end-1) = v(1:end-1)./s';
```

References

A. Asuncion and D. J. Newman. Uci machine learning repository, 2007. URL `http://archive.ics.uci.edu/ml/`.

L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

N. V. Chawla and N. Japkowicz. Editorial: Special issue on learning from imbalanced data sets. *SIGKDD Explorations*, 6:2004, 2004.

P. Domingos. Metacost: A general method for making classifiers cost-sensitive. In *In Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining*, pages 155–164. ACM Press, 1999.

C. Drummond and R. C. Holte. Explicitly representing expected cost: An alternative to roc representation. In *In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 198–207. ACM Press, 2000.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification.* Wiley, New York, 2001.

M. Dudik and S. J. Phillips. Generative and discriminative learning with unknown labeling bias. *Advances in Neural information Processing Systems*, 21, 2009.

J. P. Egan. *Signal Detection Theory and ROC Analysis*. Academic Press, New York, NY, 1975.

C. Elkan. The foundations of cost-sensitive learning. In *In Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 973–978, 2001.

G. Elliott and R. P. Lieli. Predicting binary outcomes. Technical report, 2007.

T. Fawcett. Roc graphs: notes and practical considerations for researchers. Technical Report HPL-2003-4, 2004.

R. Fletcher. *Practical Methods of Optimization*. Wiley, West Sussex, 2000.

A. Guerrero-Curieses, J. Cid-Sueiro, R. Alaiz-Rodriguez, and A. R. Figueras-Vidal. Local estimation of posterior class probabilities to minimize classification errors. *IEEE Transactions on Neural Networks*, 15:309–317, 2004.

J. L. Horowitz. A smoothed maximum score estimator for the binary response model. *Econometrica*, 60:505–531, 1992.

S. M. Kay. *Fundamentals of statistical signal processing: estimation theory*. 1993.

R. P. Lieli and H. White. The construction of empirical credit scoring rules based on maximization principles. *Journal of Econometrics*, 157:110–119, 2010.

Y. Lin, Y. Lee, and G. Wahba. Support vector machines for classification in nonstandard situations. *Machine Learning*, 46:191–202, 2002.

M. A. Maloof. Learning when data sets are imbalanced and when costs are unequal and unknown. In *ICML-2003 Workshop on Learning from Imbalanced Data Sets II*, 2003.

D. D. Margineantu. On class-probability estimates and cost-sensitive evaluation of classifiers. In *In Workshop on Cost-Sensitive Learning at the Seventeenth International Conference on Machine Learning (WCSL at ICML2000)*, 2000.

H. Masnadi-Shirazi and N. Vasconcelos. Risk minimization, probability elicitation, and cost-sensitive svms. In *Proceedings of the International Conference on Machine Learning*, pages 204–213. ACM Press, 2010.

P. McCullagh and J. A. Nelder. *Generalized Linear Models, 2nd ed.* Chapman and Hall, London, 1989.

L. Parra, C. Spence, A. Gerson, and P. Sajda. Recipes for the linear analysis of eeg. *NeuroImage*, 28:326–341, 2005.

L. Parra, C. Christoforou, A. Gerson, M. Dyrholm, A. Luo, M. Wagner, M. Philiastides, and P. Sajda. Spatio-temporal linear decoding of brain state: Application to performance augmentation in high-throughput tasks. *IEEE Signal Processing Magazine*, 25:95–115, 2008.

F. Provost and T. Fawcett. Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In *In Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pages 43–48. AAAI Press, 1997.

H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Statistical Planning and Inference*, 90:227–244, 2000.

V. N. Vapnik. *Statistical Learning Theory*. Wiley, New York, NY, 1998.

V. N. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999.

H. White. Maximum likelihood estimation of misspecified models. *Econometrica*, 50:1–25, 1982.

B. Zadrozny and C. Elkan. Learning and making decisions when costs and probabilities are both unknown. In *In Proceedings of the Seventh International Conference on Knowledge Discovery and Data Mining*, pages 204–213. ACM Press, 2001.

B. Zadrozny, J. Langford, and N. Abe. Cost-sensitive learning by cost-proportionate example weighting. In *n Proceedings of the Third IEEE International Conference on Data Mining, ICDM 2003.*, pages 435–442. IEEE, 2003.