Support Vector Machines with Example Dependent Costs

Ulf Brefeld, Peter Geibel, and Fritz Wysotzki

TU Berlin, Fak. IV, ISTI, AI Group, Sekr. FR5-8
Franklinstr. 28/29, D-10587 Berlin, Germany
{geibel,wysotzki}@cs.tu-berlin.de

Abstract. Classical learning algorithms from the fields of artificial neural networks and machine learning, typically, do not take any costs into account or allow only costs depending on the classes of the examples that are used for learning. As an extension of class dependent costs, we consider costs that are example, i.e. feature and class dependent. We present a natural cost-sensitive extension of the support vector machine (SVM) and discuss its relation to the Bayes rule. We also derive an approach for including example dependent costs into an arbitrary cost-insensitive learning algorithm by sampling according to modified probability distributions.

1 Introduction

The consideration of cost-sensitive learning has received growing attention in the past years ([9,4,5,8]). As it is stated in the Technological Roadmap of the MLnetII project (European Network of Excellence in Machine Learning, [10]), the inclusion of costs into learning and classification is one of the most relevant topics of future machine learning research.

The aim of the inductive construction of classifiers from training sets is to find a hypothesis that minimizes the mean predictive error. If costs are considered, each example not correctly classified by the learned hypothesis may contribute differently to the error function. One way to incorporate such costs is the use of a cost matrix, which specifies the misclassification costs in a class dependent manner (e.g. [9,4]). Using a cost matrix implies that the misclassification costs are the same for each example of the respective class.

The idea we discuss in this paper is to let the cost depend on the single example and not only on the class of the example. This leads to the notion of example dependent costs, which was to our knowledge first formulated in [6]. Besides costs for misclassification, we consider costs for correct classification (gains are expressed as negative costs).

One application for example dependent costs is the classification of credit applicants to a bank as either being a "good customer" (the person will pay back the credit) or a "bad customer" (the person will not pay back parts of the credit loan).

N. Lavrač et al. (Eds.): ECML 2003, LNAI 2837, pp. 23–34, 2003.

The gain or the loss in a single case forms the (mis-) classification cost for that example in a natural way. For a good customer the cost for correct classification is the negative gain of the bank. I.e. the cost for correct classification is not the same for all customers but depends on the amount of money borrowed. Generally there are no costs to be expected (or a small loss related to the handling expenses), if the customer is rejected, since he or she is incorrectly classified as a bad customer. For a bad customer, the cost for misclassification corresponds to the actual loss that has been occured. The gain of correct classification is zero (or small positive, if one considers handling expenses of the bank).

As opposed to the construction of a cost matrix, we claim that using the example costs directly is more natural and will lead to the production of more accurate classifiers. If the real costs are example dependent as in the credit risk problem, learning with a cost matrix means that in general only an approximation of the real costs is used. When using the classifier based on the cost matrix e.g. in the real bank, the real costs as given by the example dependent costs will occur and not the costs specified by the cost matrix. Therefore using example dependent costs is better than using a cost matrix for theoretical reasons, provided that the learning algorithm used is able to use the example dependent costs in an appropriate manner.

In this paper, we consider the extension of support vector machines (SVMs, [11,2,3]) by example dependent costs, and discuss its relationship to the cost-sensitive Bayes rule. In addition we provide an approach for including example-dependent costs into an arbitrary learning algorithm by using modified example distributions.

This article is structured as follows. In section 2 the Bayes rule in the case of example dependent costs is discussed. In section 3, the cost-sensitive SVM for non-separable classes is described. Experiments on some artificial domains can be found in section 5. In section 4, we discuss the inclusion of costs by resampling the dataset. The conclusion is presented in Section 6.

2 Example Dependent Costs

In the following we consider binary classification problems with classes –1 (negative class) and +1 (positive class). For an example $\mathbf{x} \in \mathbf{R}^d$ of class +1, let

- $c_{+1}(\mathbf{x})$ denote the cost of misclassifying $\mathbf{x}$
- and $g_{+1}(\mathbf{x})$ the cost of classifying $\mathbf{x}$ correctly.

The functions c_{-1} and g_{-1} are equivalently given for examples of class –1. In our framework, gains are expressed as negative costs. I.e. $g_y(\mathbf{x}) < 0$, if there is a gain for classifying $\mathbf{x}$ correctly into class y. $\mathbf{R}$ denotes the set of real numbers. d is the dimension of the input vector.

Let $r : \mathbf{R}^d \longrightarrow \{+1, -1\}$ be a classifier (decision rule) that assigns $\mathbf{x}$ to a class. According to [11] the risk of r with respect to the distribution function P of $(\mathbf{x}, y)$ is given by

$$R(r) = \int Q(\mathbf{x}, y, r) dP(\mathbf{x}, y)\,. \tag{1}$$

The loss function Q is defined by

$$Q(\mathbf{x}, y, r) = \begin{cases} g_y(\mathbf{x}) \text{ if } y = r(\mathbf{x}) \\ c_y(\mathbf{x}) \text{ else.} \end{cases} \tag{2}$$

We assume that the density $p(\mathbf{x}, y)$ exists. Let $X_y = \{\mathbf{x} \,|\, r(\mathbf{x}) = y\}$ the region of decision for class y. Then the risk can be rewritten with $p(\mathbf{x}, y) = p(\mathbf{x}|y)P(y)$ as

$$\begin{aligned} R(r) = &\int_{X_{+1}} g_{+1}(\mathbf{x})p(\mathbf{x}|{+1})P({+1})d\mathbf{x} + \int_{X_{+1}} c_{-1}(\mathbf{x})p(\mathbf{x}|{-1})P({-1})d\mathbf{x} \\ &+ \int_{X_{-1}} g_{-1}(\mathbf{x})p(\mathbf{x}|{-1})P({-1})d\mathbf{x} + \int_{X_{-1}} c_{+1}(\mathbf{x})p(\mathbf{x}|{+1})P({+1})d\mathbf{x}. \end{aligned} \tag{3}$$

$P(y)$ is the prior probability of class y, and $p(\mathbf{x}|y)$ is the class conditional probability density of class y. The first and the third integral express the costs for correct classification, whereas the second and the fourth integral express the costs for misclassification. We assume, that the integrals defining R exist. This is the case, if the cost functions are integrable and bounded.

The risk $R(r)$ is minimized, if $\mathbf{x}$ is assigned to class +1, if

$$\begin{aligned} &g_{+1}(\mathbf{x})p(\mathbf{x}|{+1})P({+1}) + c_{-1}(\mathbf{x})p(\mathbf{x}|{-1})P({-1}) \\ &\leq g_{-1}(\mathbf{x})p(\mathbf{x}|{-1})P({-1}) + c_{+1}(\mathbf{x})p(\mathbf{x}|{+1})P({+1}) \end{aligned}$$

holds, and to class –1 otherwise. From this, the following proposition is derived.

Proposition 1 (Bayes Classifier). *The function*

$$\begin{aligned} r^*(\mathbf{x}) = \text{sign}[&(c_{+1}(\mathbf{x}) - g_{+1}(\mathbf{x}))p(\mathbf{x}|{+1})P({+1}) \\ &-(c_{-1}(\mathbf{x}) - g_{-1}(\mathbf{x}))p(\mathbf{x}|{-1})P({-1})] \end{aligned} \tag{4}$$

minimizes R.

r^* is called the Bayes classifier (see e.g. [1]). As usual, we define $\text{sign}(0) = +1$. We assume $c_y(\mathbf{x}) - g_y(\mathbf{x}) > 0$ for every example $\mathbf{x}$, i.e. there is a real benefit for classifying $\mathbf{x}$ correctly.

From (4) it follows that the classification of examples depends on the *difference* of the costs for misclassification and correct classification, not on their actual values. Therefore we will assume $g_y(\mathbf{x}) = 0$ and $c_y(\mathbf{x}) > 0$ without loss of generality.

Given a training sample $(\mathbf{x}^{(1)}, y^{(1)}, c^{(1)}), \ldots, (\mathbf{x}^{(l)}, y^{(l)}, c^{(l)})$ with $c^{(i)} = c_{y^{(i)}}(x^{(i)})$, the empirical risk is defined by

$$R_{\text{emp}}(r) = \frac{1}{l} \sum Q(\mathbf{x}^{(i)}, y^{(i)}, r).$$

It holds $Q(\mathbf{x}^{(i)}, y^{(i)}, r) = c^{(i)}$, if the example is misclassified and $Q(\mathbf{x}^{(i)}, y^{(i)}, r) = 0$ otherwise. In our case, R_{emp} corresponds to the mean misclassification costs defined using the example dependent costs.

Proposition 2 ([11]). *If both cost functions are bounded by a constant B, then it holds with a probability of at least $1-\eta$*

$$R(r) \leq R_{emp}(r) + B\sqrt{\frac{h(ln\frac{2l}{h}+1) - ln\frac{\eta}{4}}{l}},$$

where h is the VC-dimension of the hypothesis space of r.

Vapnik's result from [11] (p. 80) holds in our case, since the only assumption he made on the loss function is its non-negativity and boundedness.

Let $\bar{c}_{+1}$ and $\bar{c}_{-1}$ be the *mean* misclassification costs for the given distributions. Let r^+ be the Bayes optimal decision rule with respect to these class dependent costs. Then it is easy to see that $R(r^*) \leq R(r^+)$, where $R(r^*)$ (see above) and $R(r^+)$ are evaluated with respect to the example dependent costs. I.e. because the example dependent costs can be considered to be the real costs occuring, their usage can lead to decreased misclassification costs. Of course this is only possible if the learning algorithm is able to incorporate example dependent costs.

In the following, we will discuss the cost-sensitive construction of an r using the SVM approach. In the presentation we assume that the reader is familiar with SVM learning.

3 Support Vector Machines

If the class distributions have no overlap there is a decision rule r^* with zero error. It holds $R(r^*) = 0$, independent of the cost model used. Since the cost model does not influence the optimal hypothesis, we will not consider hard margin SVMs in this paper. For soft margin SVMs the learning problem can be stated as follows.

Let $S = \{(\mathbf{x}^{(i)}, y^{(i)}) | i = 1, \ldots, l\} \subset \mathbf{R}^d \times \{+1, -1\}$ be a training sample and $c_{y^{(i)}}(\mathbf{x}^{(i)}) = c^{(i)}$ the misclassification costs defined above. For learning from a finite sample, only the sampled values of the cost functions need to be known, not their definition. We divide S into subsets $S_{\pm 1}$ which contain the indices of all positive and negative examples respectively. By means of $\phi : \mathbf{R}^d \to \mathcal{H}$ we map the input data into a feature space $\mathcal{H}$ and denote the corresponding kernel by $K(\cdot, \cdot)$. The optimization problem can now be formulated as

$$\begin{aligned} \min_{\mathbf{w},b,\xi} \ & \tfrac{1}{2}\|\mathbf{w}\|^2_{\mathcal{H}} + C \sum_{i \in S_{+1}} c_{+1}(\mathbf{x}^{(i)})\, \xi_i^k \\ & + C \sum_{i \in S_{-1}} c_{-1}(\mathbf{x}^{(i)})\, \xi_i^k \end{aligned} \tag{5}$$

$$\text{s.t.} \qquad y^{(i)} \left(\mathbf{w} \cdot \phi(\mathbf{x}^{(i)}) + b \right) \geq 1 - \xi_i \tag{6}$$

$$\xi_i \geq 0, \tag{7}$$

where the regularization constant $C > 0$ determines the trade-off between the weighted empirical risk and the complexity term.

$\mathbf{w}$ is the weight vector that together with the threshold b defines the classification function $f(\mathbf{x}) = \text{sign}(h(\mathbf{x}) + b)$ with $h(\mathbf{x}) = \mathbf{w} \cdot \phi(x)$. The slack variable ξ_i is zero for objects, that have a functional margin of more than 1. For objects with a margin of less than 1, ξ_i expresses how much the object fails to have the required margin, and is weighted with the cost value of the respective example. ξ is the margin slack vector containing all ξ_i. $\|\mathbf{w}\|_{\mathcal{H}}$ can be interpreted as the norm of h.

With $k = 1, 2$ we obtain the soft margin algorithms including individual costs (1-norm SVM and 2-norm SVM). Both cases can be extended to example dependent costs.

1-Norm SVM. Introducing non-negative Lagrange multipliers α_i, $\mu_i \geq 0$, $i = 1, \ldots, l$, we can rewrite the optimization problem with $k = 1$ and resolve the following primal Lagrangian

$$\begin{aligned} L_P(\mathbf{w}, b, \xi, \alpha, \mu) &= \frac{1}{2}\|\mathbf{w}\|_{\mathcal{H}}^2 \\ &+ C \sum_{i \in S_{+1}} c_{+1}(\mathbf{x}^{(i)})\xi_i + C \sum_{i \in S_{-1}} c_{-1}(\mathbf{x}^{(i)})\xi_i \\ &- \sum_{i=1}^{l} \alpha_i \left[y^{(i)} \left(\mathbf{w} \cdot \phi(\mathbf{x}^{(i)}) + b \right) - 1 + \xi_i \right] - \sum_{i=1}^{l} \mu_i \, \xi_i. \end{aligned}$$

Taking the derivative with respect to $\mathbf{w}$, b and ξ leads to

$$\frac{\partial L_P}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{l} \alpha_i \, y^{(i)} \phi(\mathbf{x}^{(i)}) = \mathbf{0} \tag{8}$$

$$\frac{\partial L_P}{\partial b} = - \sum_{i=1}^{l} \alpha_i y^{(i)} = 0 \tag{9}$$

$$\frac{\partial L_P}{\partial \xi_i} = C \, c_{+1}(\mathbf{x}^{(i)}) - \alpha_i - \mu_i = 0 \, , \forall i \in S_{+1} \tag{10}$$

$$\frac{\partial L_P}{\partial \xi_i} = C \, c_{-1}(\mathbf{x}^{(i)}) - \alpha_i - \mu_i = 0 \, , \forall i \in S_{-1} \tag{11}$$

Substituting (8)-(11) into the primal, we obtain the dual Lagragian that has to be maximized with respect to the α_i

$$L_D(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \, \alpha_j \, y^{(i)} y^{(j)} K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}). \tag{12}$$

Equation (12) is called the 1-norm soft margin SVM. Note that the values of the cost function c_y do not occur in L_D.

The Karush-Kuhn-Tucker conditions hold, and the corresponding complementary conditions are

$$\xi_i \, (C \, c_{+1}(\mathbf{x}^{(i)}) - \alpha_i) = 0, \; \forall i \in S_{+1} \tag{13}$$

$$\xi_i \, (C \, c_{-1}(\mathbf{x}^{(i)}) - \alpha_i) = 0, \forall i \in S_{-1}. \tag{14}$$

Thus the α_i are bounded within the so called box constraints

$$0 \leq \alpha_i \leq C\, c_{+1}(\mathbf{x}^{(i)}),\ \forall i \in S_{+1} \tag{15}$$

$$0 \leq \alpha_i \leq C\, c_{-1}(\mathbf{x}^{(i)}),\ \forall i \in S_{-1}. \tag{16}$$

I.e. in the case of example dependent costs, the box constraints depend on the cost value for the respective example.

2-Norm SVM. The optimization problem in (7) leads with $k = 2$ to the minimization of the primal Lagrangian

$$\begin{aligned} L_P(\mathbf{w}, b, \xi, \alpha) &= \frac{1}{2}\|\mathbf{w}\|^2_{\mathcal{H}} \\ &+ \frac{C}{2} \sum_{i \in S_{+1}} c_{+1}(\mathbf{x}^{(i)})\, \xi_i^2 + \frac{C}{2} \sum_{i \in S_{-1}} c_{-1}(\mathbf{x}^{(i)})\, \xi_i^2 \\ &- \sum_{i=1}^{l} \alpha_i \left[y^{(i)} \left(\mathbf{w} \cdot \phi(\mathbf{x}^{(i)}) + b \right) - 1 + \xi_i \right]. \end{aligned}$$

Analogous to the 1-norm case, the minimization of the primal is equivalent to maximizing the dual Lagrangian given by

$$\begin{aligned} L_D(\alpha) = &\sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j\, K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \\ &- \frac{1}{2} \sum_{i \in S_{+1}} \frac{\alpha_i^2}{C\, c_{+1}(\mathbf{x}^{(i)})} - \frac{1}{2} \sum_{i \in S_{-1}} \frac{\alpha_i^2}{C\, c_{-1}(\mathbf{x}^{(i)})}. \end{aligned}$$

In contrast to the 1-norm SVM, L_D depends on the values of the costs functions c_y. The quadratic optimization problem can be solved with slightly modified standard techniques, e.g. [3].

3.1 Convergence to the Bayes Rule

In [7] the cost free SVM learning problem is treated as a regularization problem in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}_K$

$$\min_{h,b,\xi} \quad \frac{1}{l} \sum_{i=1}^{l} \xi_i^k + \lambda \|h\|^2_{\mathcal{H}_K}, \tag{17}$$

with $f(\mathbf{x}) = h(\mathbf{x}) + b$ subject to (6),(7). Lin showed in [7] that the solution to (17) approximates the Bayes rule for large training sets, if $\lambda = \frac{1}{2lC}$ is chosen in an optimal manner, and the kernel is rich enough (e.g. spline kernels).

Analogous to Lin we can rewrite the optimization problem in (5) to get

$$\min_{h,b,\xi} \quad \frac{1}{l} \sum_{i=1}^{l} c_{y^{(i)}}(\mathbf{x}^{(i)})\, \xi_i^k + \lambda \|h\|^2_{\mathcal{H}_K}, \tag{18}$$

subject to (6),(7), where (6),(7) can be rewritten to

$$1 - y^{(i)} f(\mathbf{x}^{(i)}) \leq \xi_i \tag{19}$$

$$\xi_i \geq 0. \tag{20}$$

We define the function $(z)_+ = 0$, if $z < 0$, and $(z)_+ = z$, else. Then (19) and (20) can be integrated into the single inequality

$$(1 - y^{(i)} f(\mathbf{x}^{(i)}))_+ \leq \xi_i. \tag{21}$$

With this inequality, the minimization problem can be rewritten to

$$\min_{h,b,\xi} \quad \frac{1}{l} \sum_{i=1}^{l} c_{y^{(i)}}(\mathbf{x}^{(i)})((1 - y^{(i)} f(\mathbf{x}^{(i)}))_+)^k + \lambda \|h\|^2_{\mathcal{H}_K}. \tag{22}$$

For $l \to \infty$, the data driven term converges to

$$E_{\mathbf{X},Y}[c_Y(\mathbf{X})((1 - Y f(\mathbf{X}))_+)^k] \tag{23}$$

with random variables Y and $\mathbf{X}$. Equation (23) is equivalent to

$$E_{\mathbf{X}}[E_Y[c_Y(\mathbf{X})((1 - Y f(\mathbf{X}))_+)^k | \mathbf{X}]]. \tag{24}$$

(24) can be minimized, by minimizing $E_Y[.]$ for every fixed $\mathbf{X} = \mathbf{x}$ giving the expression to be minimized

$$c_{-1}(\mathbf{x})((1 + f(\mathbf{x}))_+)^k (1 - p(\mathbf{x})) + c_{+1}(\mathbf{x})((1 - f(\mathbf{x}))_+)^k p(\mathbf{x}), \tag{25}$$

where $p(\mathbf{x}) := p(+1|\mathbf{x})$.

According to the proof in [7] it can be shown that the function f that minimizes (25) minimizes the modified expression

$$g = c_{-1}(\mathbf{x})(1 + f(\mathbf{x}))^k (1 - p(\mathbf{x})) + c_{+1}(\mathbf{x})(1 - f(\mathbf{x}))^k p(\mathbf{x}). \tag{26}$$

By setting $z := f(\mathbf{x})$ and solving $\frac{\partial g}{\partial z} = 0$, we derive the decision function

$$f^*(\mathbf{x}) = \frac{[c_{+1}(\mathbf{x})p(\mathbf{x})]^{\frac{1}{k-1}} - [c_{-1}(\mathbf{x})(1 - p(\mathbf{x}))]^{\frac{1}{k-1}}}{[c_{+1}(\mathbf{x})p(\mathbf{x})]^{\frac{1}{k-1}} + [c_{-1}(\mathbf{x})(1 - p(\mathbf{x}))]^{\frac{1}{k-1}}}.$$

A random pattern is assigned to class +1 if $f^*(\mathbf{x}) \geq 0$ and to class –1 otherwise. The above proves the following proposition.

Proposition 3. *In the case $k = 2$, $sign(f^*(\mathbf{x}))$ is a minimizer of R, and it minimizes (23). It holds*

$$sign(f^*(\mathbf{x})) = r^*(\mathbf{x}).$$

$\text{sign}(f^*(\mathbf{x}))$ can be shown to be equivalent to (4) in the case $k = 2$ by using the definition of the conditional density and by simple algebraic transformations.

It can be conjectured from proposition 3 that SVM learning approximates the Bayes rule for large training sets. For $k = 1$ the corresponding cannot be shown.

4 Re-sampling

Example dependent costs can be included into a cost-insensitive learning algorithm by re-sampling the given training set. First we define the mean costs for each class by

$$B_y = \int_{\mathbf{R}^d} c_y(x) p(x|y) d\mathbf{x} \,. \tag{27}$$

We define the global mean cost $b = B_{+1}P(+1) + B_{-1}P(-1)$. From the cost-sensitive definition of the risk in (3) it follows that

$$\frac{R(r)}{b} = \int_{X_{+1}} \frac{c_{-1}(\mathbf{x}) p(\mathbf{x}|-1)}{B_{-1}} \frac{B_{-1}P(-1)}{b} d\mathbf{x} + \int_{X_{-1}} \frac{c_{+1}(\mathbf{x}) p(\mathbf{x}|+1)}{B_{+1}} \frac{B_{+1}P(+1)}{b} d\mathbf{x}.$$

I.e. we now consider the new class conditional densities

$$p'(\mathbf{x}|y) = \frac{1}{B_y} c_y(\mathbf{x}) p(\mathbf{x}|y)$$

and new priors

$$P'(y) = P(y) \frac{B_y}{B_{+1}P(+1) + B_{-1}P(-1)} \,.$$

It is easy to see that $\int p'(\mathbf{x}|y) d\mathbf{x} = 1$ holds, as well as $P'(+1) + P'(-1) = 1$.

Because b is a constant, minimizing the cost-sensitive risk $R(r)$ is equivalent to minimizing the cost-free risk

$$\frac{R(r)}{b} = R'(r) = \int_{X_{+1}} p'(\mathbf{x}|-1) P'(-1) d\mathbf{x} + \int_{X_{-1}} p'(\mathbf{x}|+1) P'(+1) d\mathbf{x}.$$

The following proposition holds.

Proposition 4. *A decision rule r minimizes R' if it minimizes R.*

The proposition follows from $R(r) = bR'(r)$.

In order to minimize R', we have to draw a new training sample from the given training sample. Assume that a training sample $(\mathbf{x}^{(1)}, y^{(1)}, c^{(1)}), \ldots, (\mathbf{x}^{(l)}, y^{(l)}, c^{(l)})$ of size l is given. Let C_y the total cost for class y in the sample. Based on the given sample, we form a second sample of size lN by random sampling from the given training set, where $N > 0$ is a fixed real number.

It holds for the compound density

$$p'(\mathbf{x}, y) = p'(\mathbf{x}|y) P'(y) = \frac{c_y(\mathbf{x})}{b} p(\mathbf{x}, y). \tag{28}$$

Therefore, in each of the $\lfloor Nl \rfloor$ independent sampling steps, the probability of including example i in this step into the new sample should be determined by

$$\frac{c^{(i)}}{C_{+1} + C_{-1}}$$

i.e. an example is chosen according to its contribution to the total cost of the fixed training set. Note that $\frac{C_{+1}+C_{-1}}{l} \approx b$ holds. Because of $R(r) = bR'(r)$, it holds $R_{\text{emp}}(r) \approx b \cdot R'_{\text{emp}}(r)$, where $R_{\text{emp}}(r)$ is evaluated with respect to the given sample, and $R'_{\text{emp}}(r)$ is evaluated with respect to the generated cost-free sample. I.e. a learning algorithm that tries to minimize the expected cost-free risk by minimizing the mean cost-free risk will minimize the expected cost for the original problem. From the new training set, a classifier for the cost-sensitive problem can be learned with a cost-insensitive learning algorithm.

Re-Sampling from a fixed sample is only sensible, if the original sample is large enough. Especially a multiple inclusion of the same example into the new training set can cause problems, e.g. when estimating the accuracy using cross validation, where the example may occur in one of the training sets *and* in the respective test set. We assume that the re-sampling method is inferior to using the example dependent costs directly. Thorough experiments on this point have to be conducted in the future.

5 Experiments

We have shown in section 2 that the usage of example dependent costs will in general lead to decreased costs for classifier application. In section 3 we showed that the inclusion of example dependent costs into the SVM is possible and sound. To demonstrate the effects of the example dependent costs and the convergence to the Bayes classifier, we have conducted experiments on two artificial domains. The two classes of the first data set where defined by Gaussian distributions having means $\mu_{\pm 1} = (0, \pm 1)^T$ and equal covariance matrices $\Sigma_{\pm 1} = \mathbf{1}$ respectively. The cost functions $c_{\pm 1}$ are defined as follows

$$c_y(\mathbf{x}) = \frac{2}{1 + \exp(-yx_1)}, \; y \in \{+1, -1\}, \tag{29}$$

see figure 1.a. We used radial basis function kernels for learning. The result of learning is also displayed in fig. 1.b-d for different number of training examples ($l = 128, 256, 512$).

For the given distributions, and the given cost functions, the expected risk is given by

$$\begin{aligned} R = &\frac{1}{2}\int_{X_{-1}} \frac{2}{2\pi(1+e^{-x_1})} e^{-\frac{1}{2}\left(x_1^2+(x_2-1)^2\right)} d\mathbf{x} \\ &+\frac{1}{2}\int_{X_{+1}} \frac{2}{2\pi(1+e^{x_1})} e^{-\frac{1}{2}\left(x_1^2+(x_2+1)^2\right)} d\mathbf{x}. \end{aligned}$$

The decision boundary is determined by the equality of the two integrands. After simple transformations it can be seen that the class boundary is defined by the hyperplane $x_1 + 2x_2 = 0$ and the optimal Bayes classifier decides in favour of class +1 if $x_1 + 2x_2 \geq 0$ and –1 otherwise. Figure 1 shows the approximation of the Bayes classifier for data sets containing 128, 256 and 512 examples with

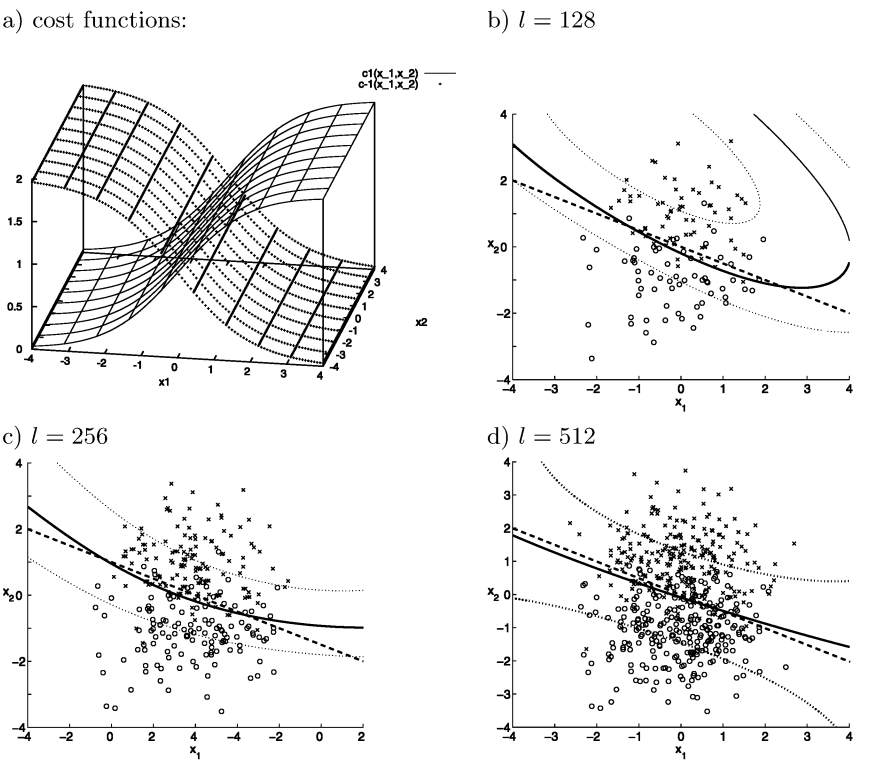

Fig. 1. Cost functions and approximation of the Bayes optimal classifier (drawn through and dashed line) with $l = 128, 256, 512$. The projections of the points on the dotted lines lie on the margin hyperplanes.

individual costs given in (29). The optimal parameter settings were determined by cross validation. We do not present the parameter settings, because they are not interesting for the purpose of this article.

The Bayes classifier *without* costs is defined by the line $x_2 = 0$. Using class dependent instead of example dependent costs results in lines $x_2 = -\frac{1}{2} ln(\frac{\bar{c}_{+1}}{\bar{c}_{-1}})$, where $\bar{c}_{\pm 1}$ denote the costs for positive and negative examples respectively. In contrast to example dependent costs, a rotation of the line is not possible for class dependent costs.

The decision based on class dependent costs is suboptimal for points between the lines $x_1 + 2x_2 \geq 0$ and $x_2 = -\frac{1}{2} ln(\frac{\bar{c}_{+1}}{\bar{c}_{-1}})$. For the cost functions in (29), the theoretical mean costs are given by $\bar{c}_{+1} = \bar{c}_{-1} = 1.0$. I.e. the decision based on class dependent costs is suboptimal with respect to the example dependent costs for points between the lines $x_1 + 2x_2 \geq 0$ and $x_2 = 0$.

An example for using class dependent costs computed as mean costs is shown in fig. 2.a. Here the individual costs in (29) were averaged for both classes and the resulting means interpreted as class dependent costs $\bar{c}_{+1} = 0.989$ and $\bar{c}_{-1} = 0.984$

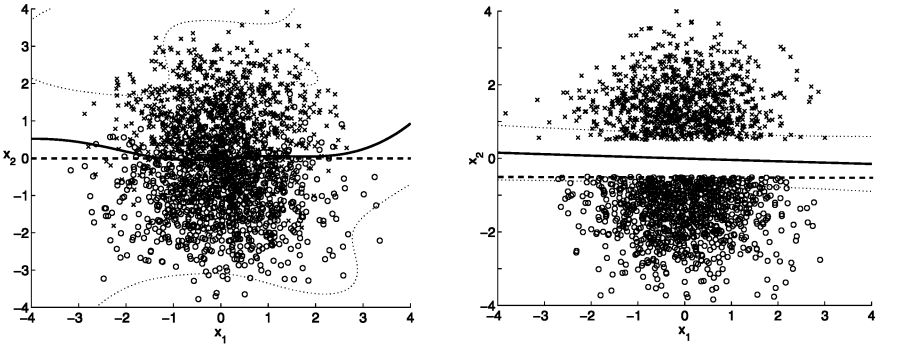

Fig. 2. a) Using class dependent, i.e. mean costs (left figure). b) Result for separable dataset with the example costs in (29) (right figure).

respectively. The learned classifier therefore coincides approximately with the cost-free Bayes classifier, see fig. 2.a. I.e. the information about the costs is lost by using class dependent costs.

An example of a separable data set with example dependent costs given in (29) is shown in fig. 2.b. As expected the resulting classifier is not influenced by using the cost functions (29). Note that due to prop. 1 the Bayes classifier r^* in (4) is defined by the class boundary $x_2 = -0.5$. Since we defined $\text{sign}(0) = +1$ and decide in favour of class $+1$ if $r^* \geq 0$, all points within the tube $-0.5 \leq x_2 \leq 0.5$ are assigned to class $+1$ by r^*. Allowing an arbitrary choice of the class, if the argument of sign in (4) equals to zero, yields a whole set of Bayes decision rules. From this set, the SVM has constructed one with maximum margin.

6 Conclusion

In this article, we discussed a natural cost-sensitive extension of SVMs by example dependent classification and misclassification costs. The cost-insensitive SVM can be obtained as a special case of the SVM with example dependent costs.

We showed, that the Bayes rule only depends on differences between costs for correct classification and for misclassification. This allows us to define a simplified learning problem where the costs for correct classification are assumed to be zero. For the simplified problem, we stated a bound for the cost-sensitive risk. A bound for the original problem with costs for correct classification can be obtained in a similar manner.

We have stated the optimization problems for the soft margin support vector machine with example dependent costs and derived the dual Lagrangians. For the case $k = 2$, we discussed the approximation of the Bayes rule using SVM learning. However a formal proof of convergence is still missing.

We suspect that the inclusion of example dependent costs may be sensible in the hard margin case too, i.e. for separable classes (fig. 2). It may lead to more robust classifiers and will perhaps allow the derivation of better error bounds.

Independently from the SVM framework, we have discussed the inclusion of example dependent costs into a cost-insensitive learning algorithm by resampling the original examples in the training set according to their costs. This way example dependent costs can be incorporated into an arbitrary cost-insensitive learning algorithm.

The usage of example dependent costs instead of class dependent costs will lead to a decreased misclassification cost in practical applications, e.g. credit risk assignment.

References

1. Ch. M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, Oxford, 1995.
2. C. J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Knowledge Discovery and Data Mining*, 2(2), 1998.
3. N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines (and Other Kernel-Based Learning Methods).* Cambridge University Press, 2000.
4. Charles Elkan. The foundations of Cost-Sensitive learning. In Bernhard Nebel, editor, *Proceedings of the seventeenth International Conference on Artificial Intelligence (IJCAI-01)*, pages 973–978, San Francisco, CA, August 4–10 2001. Morgan Kaufmann Publishers, Inc.
5. M. Kukar and I. Kononenko. Cost-sensitive learning with neural networks. In Henri Prade, editor, *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98)*, pages 445–449, Chichester, 1998. John Wiley & Sons.
6. A. Lenarcik and Z. Piasta. Rough classifiers sensitive to costs varying from object to object. In Lech Polkowski and Andrzej Skowron, editors, *Proceedings of the 1st International Conference on Rough Sets and Current Trends in Computing (RSCTC-98)*, volume 1424 of *LNAI*, pages 222–230, Berlin, June 22–26 1998. Springer.
7. Yi Lin. Support vector machines and the bayes rule in classification. *Data Mining and Knowledge Discovery*, 6(3):259–275, 2002.
8. Yi Lin, Yoonkyung Lee, and Grace Wahba. Support vector machines for classification in nonstandard situations. *Machine Learning*, 46(1-3):191–202, 2002.
9. Dragos D. Margineantu and Thomas G. Dietterich. Bootstrap methods for the cost-sensitive evaluation of classifiers. In *Proc. 17th International Conf. on Machine Learning*, pages 583–590. Morgan Kaufmann, San Francisco, CA, 2000.
10. Lorenza Saitta, editor. *Machine Learning – A Technological Roadmap.* University of Amsterdam, 2000. ISBN: 90-5470-096-3.
11. V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer, New York, 1995.