Pairwise Costs in Multiclass Perceptrons

Sarunas Raudys and Aistis Raudys

Abstract—A novel loss function to train a net of K single-layer perceptrons (KSLPs) is suggested, where pairwise misclassification cost matrix can be incorporated directly. The complexity of the network remains the same; a gradient's computation of the loss function does not necessitate additional calculations. Minimization of the loss requires a smaller number of training epochs. Efficacy of cost-sensitive methods depends on the cost matrix, the overlap of the pattern classes, and sample sizes. Experiments with real-world pattern recognition (PR) tasks show that employment of novel loss function usually outperforms three benchmark methods.

Index Terms—Cost-sensitive learning, loss function, pairwise classification, perceptron.

1 Introduction

In two category situations, the single-layer perceptron (SLP)- based classifiers [1], [2], [3] possess a number of qualities. If specifically trained, SLP can approach seven classifiers of diverse complexity: euclidean distance, regularized and standard Fisher, robust, minimal empirical error, and maximum margin (support vectors) [3], [4]. If training is stopped in time, binary perceptron can save almost all useful information contained in a properly defined initial weight vector [5]. One can specify prices (costs) of incorrect decisions C_{ij} (it is a price of deciding in favor of class Π_j when the true class is Π_i) and modify an importance of training sets of opposite pattern classes (rescaling approach) [6], [7].

In a generalization of cost-sensitive training, one can assume that *the misclassification costs depend on the particular p-dimensional input vector* $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)^T$ (the superscript "T" denotes the transposition operation), and not only on the class the vector $\boldsymbol{x}$ belongs to. A nice illustration of the class-dependent cost is the classification of credit applicants to a bank as either being good or bad customers. Two misclassification costs depend on the credit size. So, the costs are diverse for different customers and corrupted by a noise-variation of the interest rate [8], [9].

Imperative direction in the cost-sensitive learning is the *multiclass case*. In practice, time and again, one faces problems where *misclassification costs depend on the pattern class the vector* $\boldsymbol{x}$ *is allocated to* [10], [11]. An example can be a medical diagnosis task. Here, the prices of incorrect decisions depend on a particular diagnosis. The cost-sensitive solution is characterized by $K \times K$ cost matrix $\boldsymbol{C} = ((C_{ij}))$, where, for the sake of simplicity, we assume that $C_{ii} = 0$.

A standard single or multilayer perceptron's loss function used to find the weights expresses a sum of squared differences between desired outputs and real ones averaged over all outputs, classes, and training vectors [12]. This way is beneficial for inclusion of misclassification costs $C_1, C_2, \ldots, C_K$ that depend only on *a class label the vector* $\boldsymbol{x}$ *belongs to*. It is not suited for simultaneous appraisal of all $K(K-1)$ C_{ij} values. A large number of novel algorithms and extensions to existing ones are dealing with class-dependent costs. Most of them are based on the multiclass (multicategory) classification task split into $K_{\mathrm{pw}} = K(K-1)/2$ binary (pairwise) problems. The pairs of the costs, C_{ij} and C_{ji}, are used to design each binary classifier. To make a final allocation of vector $\boldsymbol{x}$, one fuses K_{pw} binary solutions. Such solutions are developed for support vector, decision tree classifiers, and fit for the perceptrons as well [8], [9], [10], [11]. A comparison of decision making strategies constitutes a separate research topic. Our aim is to develop the perceptron training procedure suitable for pairwise costs inclusion.

Two solutions to deal with the cost diversity could be applied for *multicategory perceptron training*. In a standard rescaling approach, the pairwise costs are averaged

$$\bar{C}_i = \sum_j C_{ij}/(K-1), \tag{1}$$

and the decision making procedure based on the novel costs, $\bar{C}_i$, is applied later [6], [7]. It is equivalent to weighting of contributions of the input vectors by rescaling coefficients $[r_1, r_2, \ldots, r_K] = \boldsymbol{r}_B$, where vector $\boldsymbol{r}_{\mathrm{B}}$ is proportional to $\bar{C}_i$. Zhou and Liu (Z&L) [11] suggested another way to determine *optimal rescaling coefficients*. They assumed that *costs matrix is consistent*, i.e., values C_{ij} can be expressed as a ratio of coefficients $r_1, r_2, \ldots, r_K$:

$$C_{ij}/C_{ji} = r_i/r_j. \tag{2}$$

Usually, values C_{ij} are chosen arbitrarily. Strictly speaking, Z&L's suggestion to find rescaling coefficients should not be applied in such cases. If one ignores requirements in (2), for certain pattern recognition tasks and cost matrices, approximate solutions can be unsuccessful.

The rescaling and two-stage decision making methodologies possess shortcomings. In the rescaling approach, one does not pay attention to the data. Only the cost matrix is used to find the rescaling coefficients. Actually, some of the pairwise probabilities of misclassification, P_{ij}, can be minor and have a small impact on the final loss [10]. In the two-stage decision making, each of the K_{pw} pairwise classifiers is focused on the single pair of the costs, C_{ij}, and C_{ij}. The amount of computations increases quadratically with an increase in the number of the classes.

To have an optimal procedure, one ought to take into account the costs C_{ij} and classification errors P_{ij}. In the classic Amari paper [13], integration over the patterns misclassified was considered. In the recent proposal of Santos-Rodríguez et al. [14] it was suggested to use an approximation, the Bregman divergences, to evaluate posterior probabilities P_i of vectors that are close to the decision boundaries. Excluding papers [13], [14], the problem of finding P_i in dependence of a liberally selected matrix (C_{ij}) was not considered so far. The inability to individuate pairwise costs is an important shortcoming of present day theory.

The paper focuses on the quality of pairwise decision boundaries. It improves ordinary loss function used to train KSLPs [12] and allows obtaining smaller misclassification error rates if the classes do not overlap drastically. It also allows incorporating the pairwise misclassification costs directly. Minimization of the loss function is fast. In Section 2, we analyze peculiarities of the novel loss function. In Section 3, we show that the efficacy of the novel method depends on the cost matrix, the pairwise classification error rates, and sample sizes. Finally, in Section 4, we discuss unsolved issues for future research.

2 Proposed Method

2.1 A Standard K-Category Single-Layer Perceptron

The K-class net of perceptrons [1], [2], [3], [12] performs classification according to the maximum of K outputs, $o_j = f(net_j)$ $(j = 1, 2, \ldots, K)$, where $net_j = \boldsymbol{w}_j^T \dot{\boldsymbol{x}}$, $\boldsymbol{w}_j = (v_{j0}, v_{j1}, v_{j2}, \ldots, v_{jg}, \ldots, v_{jp})^T$, $\dot{\boldsymbol{x}} = (1, x_1, x_2, \ldots, x_g, \ldots, x_p)^T$ are the "augmented" $(p+1)$-dimensional weight and input vectors, and $f(net_j)$

- *S. Raudys is with the Department of Mathematics and Informatics, Vilnius University, Didlaukio 47, Vilnius LT-08303, Lithuania. E-mail: sarunas.raudys@mif.vu.lt.*
- *A. Raudys is with the Institute of Mathematics and Informatics, Akademijos str. 4, Vilnius LT-08663, Lithuania. E-mail: aistis@raudys.com.*

Manuscript received 24 Mar. 2009; revised 6 Oct. 2009; accepted 12 Dec. 2009; published online 2 Mar. 2010.
Recommended for acceptance by M. Figueiredo.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2009-03-0189.
Digital Object Identifier no. 10.1109/TPAMI.2010.72.

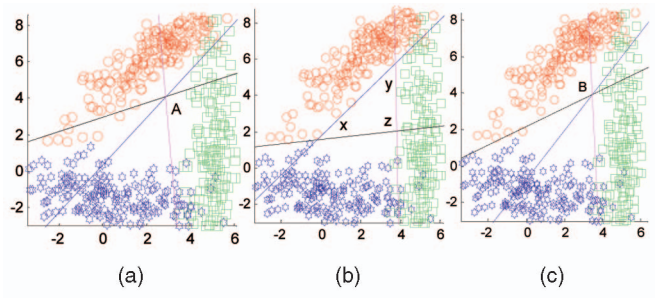

Fig. 1. Scatter diagrams of three pattern classes and three pairwise decision boundaries. (a) Standard KSLP, (b) three "ideal" pairwise perceptrons, and (c) modified KSLP.

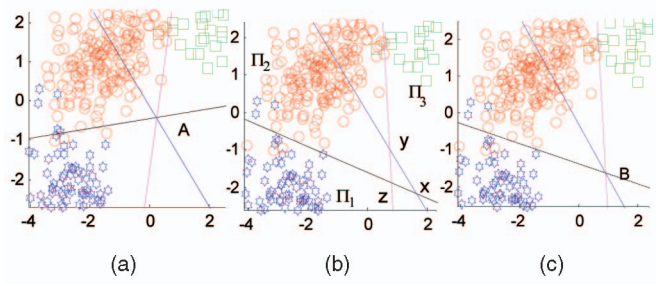

Fig. 2. Scatter diagrams of three pattern classes and three pairwise decision boundaries. (a) Standard KSLP, (b) ideal case, and (c) modified KSLP.

is a nonlinear soft limiting activation function, e.g., $f(net) = 1/(1 + exp(-net))$. A standard loss function regularly used to find perceptron weights is

$$Loss_{\text{stand}} = \sum_{h=1}^{K}\sum_{j=1}^{K}\sum_{s=1}^{N_h}\left(t_{hj} - f\left(w_j^T \dot{\boldsymbol{x}}_s^{(h)}\right)\right)^2, \tag{3}$$

where t_{hj} is a desired output corresponding to the jth output and the hth class training vector, $\dot{\boldsymbol{x}}_s^{(h)}$. For the sigmoid function, we assume: $t_{hj} = 1$ if $j = h$, and $t_{hj} = 0$ if $j \neq h$.

While minimizing loss function (3), we work with K separate sums $\sum_{h=1}^{K}\sum_{s=1}^{N_h}(t_{hj} - f(w_j^T \dot{\boldsymbol{x}}_s^{(h)}))^2$ independently $(j = 1, \ldots, K)$. To take into account pairwise costs, we have to evaluate influences of the pairwise classification error rates. Training vectors of two classes have to be used in such a scenario. Consequently, minimization of (3) could lead to nonoptimal results. In Fig. 1, we present two-dimensional (2D) example with three Gaussian classes.

We see three pairwise decision lines $(\boldsymbol{w}_1 - \boldsymbol{w}_2)^T \dot{\boldsymbol{x}} = 0$, $(\boldsymbol{w}_1 - \boldsymbol{w}_3)^T \dot{\boldsymbol{x}} = 0$, and $(\boldsymbol{w}_2 - \boldsymbol{w}_3)^T \dot{\boldsymbol{x}} = 0$ formed by standard KSLP (Fig. 1a) and that formed by three separately trained SLPs (Fig. 1b). In Fig. 1b, intersections of three lines form *a triangle* xyz. Inside the triangle, classification is ambiguous. In this example, we have no training vectors inside the triangle. That is the reason why we have low misclassification error rate: 2.5 percent. KSLP, however, generates 6.7 percent of classification errors. One can say that it is a consequence of an unspoken requirement that three decision lines of KSLP intersect in a single point (point A in Fig. 1a).

While minimizing single terms, $\sum_{h=1}^{K}\sum_{s=1}^{N_h}(t_{hj} - f(\boldsymbol{w}_i^T \dot{\boldsymbol{x}}_s^{(h)}))^2$, of function (3), we solve K one-against-all classification problems with imbalanced training sets: We have N_h training vectors in one group and $\sum_{i=1}^{K} N_i - N_h$ vectors in another one. If sample sizes and prior probabilities are not proportional, i.e., $q_h \neq N_h / \sum_{i=1}^{K} N_i$, the class imbalance should be included into the loss function. Instead of loss (3), one needs to use a modified one

$$Loss_{\text{imbalance}} = \sum_{h=1}^{K}\frac{C_h q_h}{N_h}\sum_{j=1}^{K}\sum_{s=1}^{N_h}\left(t_{hj} - f\left(\boldsymbol{w}_j^T \dot{\boldsymbol{x}}_s^{(h)}\right)\right)^2. \tag{4}$$

2.2 A Novel Loss Function

In order to force the KSLP to pay more attention to the pairwise decision boundaries, instead of minimizing weight vectors $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_j, \ldots, \boldsymbol{w}_K$ directly, we focus our attention on the K_{pw} difference weight vectors, $\boldsymbol{w}_{jh} = \boldsymbol{w}_j - \boldsymbol{w}_h$ $(j \neq h)$, and *replace regular cost function* (4) with

$$\begin{aligned} Loss_{\text{modified}} &= \sum_{h=1}^{K}\frac{C_h q_h}{N_h}\sum_{j\neq h\ j=1}^{K}\sum_{s=1}^{N_h}\left(t_{hj} - f\left((\boldsymbol{w}_j - \boldsymbol{w}_h)^T \dot{\boldsymbol{x}}_s^{(h)}\right)\right)^2 = \\ &= \sum_{h=1}^{K}\frac{C_h q_h}{N_h}\sum_{j\neq h\ j=1}^{K}\sum_{s=1}^{N_h}\left(f(\boldsymbol{w}_j - \boldsymbol{w}_h)^T \dot{\boldsymbol{x}}_s^{(h)}\right)\right)^2, \end{aligned} \tag{5}$$

where we used the earlier assumption that $t_{hj} = 0$ if $j \neq h$.

After introducing the pairwise costs C_{ij}, with assumption $C_{ii} = 0$, we have

$$Loss_{\text{novel}} = \sum_{h=1}^{K}\frac{q_h}{N_h}\sum_{j=1}^{K}C_{hj}\sum_{s=1}^{N_h}\left(f(\boldsymbol{w}_j - \boldsymbol{w}_h)^T \dot{\boldsymbol{x}}_s^{(h)}\right)\right)^2. \tag{6}$$

While minimizing loss (6), we calculate K gradients according to $\boldsymbol{w}_g$ $(g = 1, 2, \ldots, K)$:

$$\begin{aligned} \partial Loss_{\text{novel}}/\partial \boldsymbol{w}_g = 2 &\sum_{h\neq g\ h=1}^{K}\frac{q_h C_{hj}}{N_h}\sum_{s=1}^{N_h} f\left((\boldsymbol{w}_g - \boldsymbol{w}_h)^T \dot{\boldsymbol{x}}_s^{(h)}\right) \\ &\times f'\left((\boldsymbol{w}_g - \boldsymbol{w}_h)^T \dot{\boldsymbol{x}}_s^{(h)}\right)\dot{\boldsymbol{x}}_s^{(h)} - 2\sum_{j\neq g\ j=1}^{K}\frac{q_g C_{gj}}{N_g}\sum_{s=1}^{N_g} f\left((\boldsymbol{w}_j - \boldsymbol{w}_g)^T \dot{\boldsymbol{x}}_s^{(g)}\right) \\ &\times f'\left((\boldsymbol{w}_j - \boldsymbol{w}_g)^T \dot{\boldsymbol{x}}_s^{(g)}\right)\dot{\boldsymbol{x}}_s^{(g)}. \end{aligned} \tag{7}$$

A major part of the calculation is devoted to find weighted sums $\boldsymbol{w}_j^T \dot{\boldsymbol{x}}_s^{(h)}$. For conventional and novel loss, we have to compute $K\sum_{h=1}^{K} N_h$ such sums. To find derivatives, $f'(\boldsymbol{w}_j^T \dot{\boldsymbol{x}}_s^{(h)} - \boldsymbol{w}_j^T \dot{\boldsymbol{x}}_s^{(h)})$, we use a lookup table. Thus, the amounts of the calculations necessary to find the gradients of loss functions (4) and (6) are similar. Moreover, direct minimization of pairwise loss routinely requires smaller number of training epochs.

3 Experiments

An objective is to compare efficacies of the employment of novel loss function with three benchmark methods: traditional 0-1 loss-based KSLP training and the two variants of handling the pairwise costs, cost averaging (1) and the Z&L rescaling.

3.1 Two-Dimensional Artificial Data

In Figs. 1a and 1b, we showed the situation where KSLP was ineffective in comparison with the application of three pairwise hyperplanes. Pairwise decision boundaries formed by KSLP trained with novel cost function (Fig. 1c), however, are closer to the ideal pairwise decision lines (Fig. 1b) than minimization of standard loss (Fig. 1a). The classification error is smaller: 4.6 percent versus 6.7 percent of error. To show the pros of the new method, we generated another 2D example, where it performs notably better (Fig. 2). We consider three rules, as follows:

1. Standard KSLP: In training, it generates 4.2 percent of errors (Fig. 2a).
2. Ideal case with pairwise decision boundary: It generates 3.3 percent of errors (Fig. 2b).
3. Novel cost function KSLP: It generates 3.3 percent of errors (Fig. 2c).

Like in the earlier example, the standard KSLP is not successful: The decision boundaries of the KSLP (Fig. 2a) and three ideal pairwise classifiers (Fig. 2b) are distant from each other in the areas of intersection of the adjacent pattern classes. The intersection

point A (see Fig. 2a) and the ambiguity triangle xyz (Fig. 2b) are rather distant too. In comparison with point A, the intersection point B (Fig. 2c) formed by novel KSLP is much closer to triangle xyz. Decision lines of novel KSLP (Fig. 2c) and ideal pairwise classifiers (Fig. 2a) discriminate the neighboring pattern classes similarly. Consequently, the classification error rates of the pairwise classifiers and the new KSLP are comparable: We have approximately 3.3 percent misclassifications. It is lower than traditional KSLP (4.2 percent) error rate.

The above two examples demonstrate that *the effectiveness of novel loss function is problem dependent*. Following the 2D examples, an interested reader could construct a toy data model, where the novel method outperforms the traditional KSLP several times or be practically equivalent. Both the geometry of distributions of the pattern classes in multidimensional input space and the values of the pairwise costs are affecting differences between loss functions (4) and (6). It is problematic to determine which loss function to use in each practical case. However, the good news is that the new method gives as good as or better results than the traditional KSLP. We will consider this question in our experiments.

3.2 Experiments with Three Real-World Data Sets

3.2.1 The Data

The Chromosomes data set [15] is based on 30 geometrical measurements and describes 24 classes, 500 vectors in each of them ($n = 12{,}000$). To look at a variety of pattern class configurations, we also considered 100 randomly selected quadruplets of the classes.

The Satimage data set [16] describes 36 spectral values of pixels in a satellite image. Six classes contain 1,072, 479, 961, 415, 470, and 1,038 vectors, respectively ($n = 4{,}435$).

The Yeast data set describes 10 types of yeast infections. The classes contain 113, 84, 116, 83, 120, 56, 90, 97, 113, and 129 vectors, respectively ($n = 1{,}001$). Originally, we had 1,500 spectral features. To form 20 similarity features, we determined two cluster centers in each pattern class.

3.2.2 Cost Matrices

The relative effectiveness of the cost evaluation methods depends on the $K \times K$-dimensional cost matrices themselves. Following previous research papers [10], [11], [14], we considered a variety (a hundred in each experiment) of the cost matrices. In a constraint-free matrices model, the costs C_{ij} ($j \neq i$) were generated at random in the interval (1 10). To satisfy the Z&L model [11], C_{ij} values ($j > i$) were generated in the interval (1 10), while elements C_{ij} ($j > i$) were determined from C_{ij} to satisfy requirements (2). To compare the conventional and novel loss functions, we considered also 0-1 costs ($C_{ij} = 1$ if $j \neq i$).

3.2.3 Benchmark Methods

Three baseline methods were used in evaluating novel cost function as follows:

1. the standard cost blind K class single-layer perceptron (loss function (3)),
2. the cost-sensitive KSLPs with rescaling based on averaging (1), and
3. the cost-sensitive KSLPs developed for the costs satisfying consistency requirements (2) (Z&L method). Here, rescaling vector $\boldsymbol{r}_Z L = [1, \boldsymbol{r}^*]$ is expressed as a solution of equation

$$C_K \boldsymbol{r}_{ZL}^T = \mathbf{0}, \tag{8}$$

where $\boldsymbol{r}^* = [r_2, \ldots, r_K]$. For $K = 4$, the $K_{\mathrm{pw}} \times K$-dimensional matrix $\boldsymbol{C}_K$ is constructed as [11]

$$C_K = \begin{bmatrix} C_{21} & -C_{12} & 0 & 0 \\ C_{31} & 0 & -C_{13} & 0 \\ C_{41} & 0 & 0 & -C_{14} \\ 0 & C_{32} & -C_{23} & 0 \\ 0 & C_{42} & 0 & -C_{24} \\ 0 & 0 & C_{43} & -C_{34} \end{bmatrix}. \tag{9}$$

If a rank of the matrix $\boldsymbol{C}_K$, $ra \leq K - 1$, requirements in (2) are satisfied. Then, Z&L rescaling can be carried out exactly. If $ra = K$, we have an approximate solution.

3.2.4 Experimental Setup

Before training, all features of the three data sets were decorrelated and normalized according to their eigenvectors and eigenvalues [3]. To evaluate the classification error rates, we used twofold cross-validation technique. Half of the data was used for training and the remaining half was used for testing. Afterward, the training and testing sets were interchanged. If the data size is not sufficiently large, random split introduces considerable errors in estimation of classification performance. To obtain reliable estimates, we performed cross-validation experiments $n_{CV} = 10$ times, reshuffling the data in each single pattern class every time. To stop training optimally, pseudovalidation data sets were formed from each class training set by means of a colored noise injection [17]. Here, for each single training vector $\boldsymbol{x}_s^{(h)}$, one finds its k-nearest neighbors, $\boldsymbol{x}_{s1}^{(h)}, \ldots \boldsymbol{x}_{sk}^{(h)}$, from the same pattern class. Then, one adds Gaussian $\mathrm{N}(0, \sigma^2)$ noise n_{nn} times along k lines connecting $\boldsymbol{x}_s^{(h)}$, and $\boldsymbol{x}_{s1}^{(h)}, \ldots \boldsymbol{x}_{sk}^{(h)}$. We used $k = 2$ and a noise standard deviation was equal to distance $|\boldsymbol{x}_s^{(h)} - \boldsymbol{x}_{sj}^{(h)}|$. The number of artificial vectors generated around each single training vector is $n_{nn} = 2$. We remind the reader that a noise injection introduces certain supplementary information by filling a space between nearest vectors of one pattern class with vectors of the same category.

3.2.5 Results

Simulations confirmed that the efficacy of diverse pairwise cost assessment techniques highly depends on the cost matrices. Usage of the novel loss function resulted in a lower or approximately the same loss as the best from the benchmark methods (Table 1, see also a scatter diagram in Fig. 3a). In the rightmost column of Table 1, we present average gain ratio, between generalization errors of the novel and the best of three benchmark methods. The gain was obtained even in cases where conventional 0-1 cost (*cost blind*) was used. The gains for the Z&L model are smaller than that for freely generated cost matrices. For certain data and cost configurations, blind application of the Z&L rescaling leads to much worse results. Strictly speaking, the Z&L rescaling could be applied only if requirements (2) are satisfied. Therefore, in cases of violation of requirements (2), Zhou and Liu [11] used the two-stage decision making procedure with $K(K-1)/2$ pairwise classifiers.

We focused on the development of the loss function useful for the straight pairwise costs inclusion. Therefore, in the fourth series of the experiments (the lowest three rows of Table 1), we concentrated our attention on the diversities of the PR tasks and the cost matrices. In 100 experiments, we considered data sets characterized by the same number of features and the type of pattern classes. To have assorted overlaps of the classes, we formed 100 four-category PR tasks by randomly selecting the quadruplets of the pattern classes. The experiments with the 100 diverse PR tasks and different cost matrices definitely confirmed the conclusions obtained with three real-world data sets. We remind that we compared averages of $n_{CV} = 10$ twofold cross-validation trials (20 learn/test trials) and that while learning, only training set information was used.

In comparison with the conventional loss function ((3) and (4)), *the novel loss function* (6) *is more complex* since it requires minimization of $K_{\mathrm{pw}} = K(K-1)/2$ pairwise loss functions in an

TABLE 1
Average Costs and the Ratio $C_{\text{best}}/C_{\text{novel}}$

Data \ cost evaluation method		Cost blind	Averaging	Zhou&Liu	Novel (6)	$C_{\text{best}}/C_{\text{novel}}$
Yeast, K=10 classes, p=20, n=1001	Fixed, *cost blind*	**0.092**	**0.092**	**0.092**	0.088	1.04
	Rand, *Z&L model*	0.439	0.402	**0.400**	0.391	1.02
	Rand, free *model*	0.507	**0.500**	0.514	0.478	1.06
Sattelite, K=6 classes, p=10, n= 4,435	Fixed, *cost blind*	**0.173**	**0.173**	**0.173**	0.152	1.14
	Rand,*Z&Lmodel*	0.836	0.802	**0.779**	0.637	1.24
	Rand, *free model*	0.995	0.984	**0.983**	0.763	1.30
Chromosomes, K=24 classes, p=30, n=12,000	Fixed, *cost blind*	**0.255**	**0.255**	**0.255**	0.241	1.06
	Rand, *Z&L model*	1.218	**1.116**	1.139	1.033	1.09
	Rand, *free model*	1.394	**1.396**	1.454	1.294	1.11
Chromosomes random selection of 100 quadruplets of the classes, p=30, n=2000	Fixed, *cost blind*	**0.079**	**0.079**	**0.079**	0.074	1.07
	Rand,*Z&Lmodel*	0.383	**0.355**	0.364	0.339	1.05
	Rand, *free model*	0.435	**0.430**	0.432	0.398	1.08

The best from the benchmark methods are marked in boldface.

indirect way. Moreover, while minimizing novel loss function, we are using training data to optimize the effects of diverse pairwise costs. In determining the rescaling coefficients, one pays attention to the cost matrix only, and does not pay attention to the data. As a result, *novel loss function requires larger training sets*. Table 1 shows that the lowest gain was obtained for the Yeast data, where the sample size/dimensionality ratio was the smallest. Additional experiments performed in a resubstitution regime (all available data were used for training, and later, the same data were used for testing) also confirmed the complexity/sample size issue: in large sample situations, the gain due to applying the novel cost function is larger.

3.3 Influence of an Overlap of the Pattern Classes

Both the conventional and the novel cost functions are aimed to find K $(p+1)$-dimensional weight vectors used to discriminate the K pattern classes and K_{pw} pairs of the classes as well. If the pattern classes overlap considerably, we cannot avoid a situation where a notable part of incorrectly classified vectors is participating in determining *three and more* weight vectors. To trace an influence of the overlap on the effectiveness of standard and novel loss functions, we performed an artificial experiment with a partially synthetic data.

We have chosen the Yeast data, where the differences between effectiveness of conventional and novel costs were the smallest. To have small dimensionality/sample size ratio, we reduced dimensionality up to $p = 8$ by the principal component method. After subtracting a mean of the data set from all 8D vectors, we multiplied the means of each pattern class by an "overlap coefficient" α_{overlap} ranging in interval (0.1 2.5). In such a way, we were changing the overlap of pseudoartificial data. Fig. 3b shows that the generalization errors of novel KSLPs varied between 74 percent ($\alpha_{\text{overlap}} = 0.1$) and 0.8 percent ($\alpha_{\text{overlap}} = 2.5$). A fraction of vectors incorrectly classified as vectors of two or more pattern classes decreased from 70 percent up to 0.7 percent. A ratio between generalization errors of conventional and novel methods varied between 0.993 and 1.56 (see Fig. 3b). It means that a *small overlap of the pattern classes is more favorable to novel cost function*. Detailed analysis of the influence of overlap of the classes as well as problems associated with complexity and sample size issues deserve special studies.

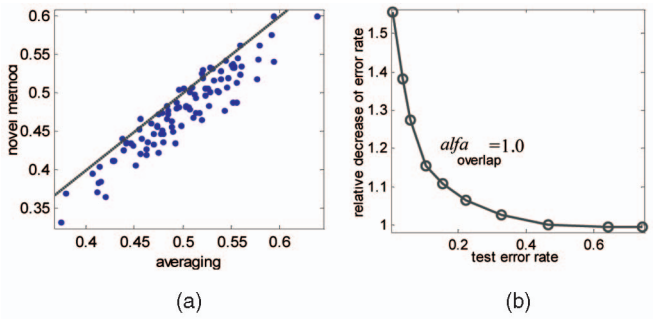

Fig. 3. (a) Scatter diagram of distribution of loss values in the experiment with Yeast data. (b) Dependence of the relative gain due to the application of the novel loss function instead of the conventional one in a sequence of PR tasks with varying overlap of the pattern classes.

4 Concluding Remarks

The standard loss function aimed at finding K weight vectors of the net of perceptrons [12] is not suited to minimize pairwise classification error rates and take into account their costs. Up to now, a general solution of how to deal with the pairwise misclassification costs in multicategory classification did not exist. The known solutions in the rescaling approach were based either on the cost averaging or require explicit relationships between the pairwise cost values. Besides, the rescaling pays no attention to multiclass training data. A popular alternative way to take into account the pairwise costs is to convert the multiclass problem into $K(K-1)/2$ binary classification problems. Such a way, however, requires developing prudent fusion rules to make the final allocation. Moreover, the computational resources increase quadratically with an increase in the number of the pattern classes.

Values of the pairwise costs can play essential role in the training process. To access the pairwise costs in SLP training, we suggested the novel loss function, where one pays attention to *pairwise sums of square errors*. The relationship between network complexity and the number of classes remains linear. Training is fast.

We found that relative efficacies of cost assessment methods depend on the matrix of pairwise costs, on the degree of overlap of the pattern classes, and training set size. Analysis of specially designed artificial data models showed that the nets of K SLPs can fail against the K-class pairwise classification in certain multiclass tasks. Nevertheless, the experiments with real-world data sets

demonstrated that in almost all PR tasks and for all cost matrices considered, the novel approach outperformed the benchmark ones.

In the output layers of Multilayer perceptrons (MLPs) and Radial basic functions (RBFs) networks, we also have nets of K class SLPs. No doubt, the novel loss function could be incorporated into MLP and RBF loss functions. When KSLPs' weights are large, the novel loss function (6) starts characterizing pairwise classification error rates. An intelligent weight magnitude control could possibly help control complexities of the pairwise decision rules and reduce the loss in finite sample size situations. These are topics for future research.

In spite of the fact that inclusion of the costs into learning has been regarded as one of the most relevant topics of future machine learning research [11], [18], due to the lack of general methodology to design cost-sensitive classification rules, choosing the values of pairwise misclassification costs remains an unsolved imperative question. Free determination of the costs on the basis of "common-sense" is possibly not the best strategy. A theory of "approximately consistent" pairwise costs is necessary. Analysis of imbalanced training sets, wider diversity of the pairwise costs, the influences of the overlap of the pattern classes, and small sample size issues also remain as important topics for future studies.

ACKNOWLEDGMENTS

The authors thank R.P.W. Duin and R. Somorjai for providing data for the experiments and the anonymous reviewers for their useful and challenging remarks.

REFERENCES

[1] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification and Scene Analysis,* second ed. Wiley, 2000.
[2] C.M. Bishop, *Pattern Recognition and Machine Learning.* Springer-Verlag, 2006.
[3] S. Raudys, *Statistical and Neural Classifiers: An Integrated Approach to Design.* Springer-Verlag, 2001.
[4] S. Raudys, "Evolution and Generalization of a Single Neurone: I. SLP as Seven Statistical Classifiers," *Neural Networks,* vol. 11, pp. 283-296, 1998.
[5] S. Raudys and S. Amari, "Effect of Initial Values in Simple Perception," *Proc. IEEE World Congress Computational Intelligence,* pp. 1530-1535, May 1998.
[6] L. Breiman, J.H. Friedman, R.A. Olsen, and C.J. Stone, *Classification and Regression Trees.* Wadsworth, 1984.
[7] P. Domingos, "MetaCost: A General Method for Making Classifiers Cost-Sensitive," *Proc. ACM SIGKDD,* pp. 155-164, 1999.
[8] P. Geibel and F. Wysotzki, "Learning Perceptrons and Piecewise Linear Classifiers Sensitive to Example Dependent Costs," *Applied Intelligence,* vol. 21, no. 1, pp. 45-56, 2004.
[9] P. Geibel, U. Brefeld, and F. Wysotzki, "Perceptron and SVM Learning with Generalized Cost Models," *Intelligent Data Analysis,* vol. 8, no. 5, pp. 439-455, 2004.
[10] N. Abe, B. Zadrozny, and J. Langford, "An Iterative Method for Multi-Class Cost-Sensitive Learning," *Proc. ACM SIGKDD,* pp. 3-11, 2004.
[11] Z.-H. Zhou and X.-Y. Liu, "On Multi-Class Cost-Sensitive Learning," *Proc. 21st Nat'l Conf. Artificial Intelligence,* pp. 567-572, 2006.
[12] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning Internal Representations by Error Propagation," *Parallel Distributed Processing: Explorations in the Microstructure of Cognition,* D.E. Rumelhart and J.L. McClelland, eds., vol. 1, pp. 318-362, Bradford, 1986.
[13] S. Amari, "A Theory of Adaptive Pattern Classifiers," *IEEE Trans. Electronic Computers,* vol. 16, pp. 299-307, 1967.
[14] R. Santos-Rodríguez, A. Guerrero-Curieses, R. Alaiz-Rodríguez, and J. Cid-Sueiro, "Cost-Sensitive Learning Based on Bregman Divergences," *Machine Learning,* vol. 76, pp. 271-285, 2009.
[15] E. Pekalska and R.P.W. Duin, "Dissimilarity Representations Allow for Building Good Classifiers," *Pattern Recognition Letters,* vol. 23, pp. 943-956, 2002.
[16] A. Asuncion and D.J. Newman, *UCI Machine Learning Repository,* http://www.ics.uci.edu/~mlearn/MLRepository.html, 2007.
[17] M. Skurichina, S. Raudys, and R.P.W. Duin, "K-NN Directed Noise Injection in Multilayer Perceptron Training," *IEEE Trans. Neural Networks,* vol. 11, no. 2, pp. 504-511, Mar. 2000.
[18] "Machine Learning: A Technological Roadmap," L. Saitta, ed., technical report, Univ. of Amsterdam, 2000.