Multicategory Support Vector Machines

Yoonkyung Lee, Yi Lin, & Grace Wahba*
Department of Statistics
University of Wisconsin-Madison
yklee,yilin,wahba@stat.wisc.edu

Abstract

The Support Vector Machine (SVM) has shown great performance in practice as a classification methodology. Oftentimes multicategory problems have been treated as a series of binary problems in the SVM paradigm. Even though the SVM implements the optimal classification rule asymptotically in the binary case, solutions to a series of binary problems may not be optimal for the original multicategory problem. We propose multicategory SVMs, which extend the binary SVM to the multicategory case, and encompass the binary SVM as a special case. The multicategory SVM implements the optimal classification rule as the sample size gets large, overcoming the suboptimality of the conventional one-versus-rest approach. The proposed method deals with the equal misclassification cost and the unequal cost case in unified way.

1 Introduction

This paper concerns Support Vector Machines (SVM) for classification problems when there are more than two classes. The SVM paradigm has a nice geometrical interpretation of discriminating one class from the other by a separating hyperplane with maximum margin in the binary case. See Boser, Guyon, & Vapnik (1992), Vapnik (1998), and Burges (1998). Now, it is commonly known that the SVM paradigm can be cast as a regularization problem. See Wahba (1998) and Evgeniou, Pontil, & Poggio (1999) for details. In the statistical point of view, it becomes natural to ask about statistical properties of SVMs, such as what is the asymptotic limit of the solution to the SVM or what is the relation between SVMs and the Bayes rule, the optimal rule we can get when we know the underlying distribution. Lin (1999) has shed a fresh light on SVMs by answering these questions. Let $X \in R^d$ be covariates used for classification, and Y be the class label, either 1 or -1 in the binary case. We define (X, Y) as a random sample from the underlying distribution $P(\mathbf{x}, y)$, and $p_1(\mathbf{x}) = P(Y = 1|X = \mathbf{x})$ as the probability of a random sample being in the positive class given $X = \mathbf{x}$. In the paper, it was shown that the solution of SVMs, $f(\mathbf{x})$ targets directly at $sign(p_1(\mathbf{x}) - 1/2) = sign[\log \frac{p_1(\mathbf{x})}{1 - p_1(\mathbf{x})}]$ and, implements the Bayes rule asymptotically. The estimated $f(\mathbf{x})$ in the SVM paradigm is given by sparse linear combinations of some basis functions depending only on data points near the classification boundary or the misclassified data points.

*Research partly supported by NSF Grant DMS0072292 and NIH Grant EY09946.

For the multicategory classification problem, assume the class label $Y \in \{1, \cdots, k\}$ without loss of generality. k is the number of classes. To tackle the problem, one may take one of two strategies in general: reducing the multicategory problem to a series of binary problems or considering all the classes at once. Constructing pairwise classifiers or one-versus-rest classifiers corresponds to the former approach. The pairwise approach has the disadvantage of potential variance increase since smaller samples are used to learn each classifier. Regarding its statistical validity, it allows only a simple cost structure when unequal misclassification costs are concerned. See Friedman (1997) for details. For SVM, the one-versus-rest approach has been widely used to handle the multicategory problem. So, the conventional recipe using an SVM scheme is to train k one-versus-rest classifiers, and to assign a test sample the class giving the largest $f_j(\mathbf{x})$, the SVM solution from training class j versus rest for $j = 1, \cdots, k$. Even though the method inherits the optimal property of SVMs for discriminating one class from the rest of the classes, it does not necessarily imply the best rule for the original k-category classification problem. Define $p_j(\mathbf{x}) = P(Y = j|X = \mathbf{x})$. Leaning on the insight that we have from two category SVMs, $f_j(\mathbf{x})$ will approximate $sign[p_j(\mathbf{x}) - 1/2]$. If there is a class j with $p_j(\mathbf{x}) > 1/2$ given $\mathbf{x}$, then we can easily pick the majority class j by comparing $f_\ell(\mathbf{x})$'s for $\ell = 1, \cdots, k$ since $f_j(\mathbf{x})$ would be near 1, and all the other $f_\ell(\mathbf{x})$ would be close to -1, making a big contrast. However, if there is no dominating class, then all $f_j(\mathbf{x})$'s would be close to -1, having no discrimination power at all. Indeed the one-versus-rest scheme doesn't make use of the class mutual exclusiveness. It is different from the Bayes rule which assigns a test sample $\mathbf{x}$ to the class with the largest $p_j(\mathbf{x})$. Thus there is a demand for a rightful extension of SVMs to the multicategory case, which would inherit the optimal property of the binary case, and solve the problem not by breaking it into unrelated pieces like k one-versus-rest classifiers, but in a simultaneous fashion. In fact, there have been alternative formulations of multicategory SVM considering all the classes at once, such as Vapnik (1998), Weston & Watkins (1998), and Crammer & Singer (2000). However, the relation of the formulations to the Bayes rule is unclear. So, we devise a loss function for the multicategory classification problem, as an extension of the SVM paradigm, and show that under the loss function, the solution to the problem directly targets the Bayes rule in the same fashion as for the binary case. For unequal misclassification costs, we generalize the loss function to incorporate the cost structure in a unified way so that the solution to the generalized loss function would implement the Bayes rule for the unequal costs case again. This would be another extension of existing two category SVMs for the nonstandard case in Lin, Lee, & Wahba (2000) to the multicategory case.

The outline of the paper is as follows. We briefly review the Bayes rule, the optimal classification rule in Section 2. The equal cost case and unequal cost case are both covered. Section 3 is the main part of the paper where we present a formulation of multicategory SVMs given as a rightful extension of ordinary SVMs for the standard case. Section 4 merely concerns modifications of the formulation to accommodate the nonstandard case, followed by the derivation of the dual problem in Section 5. A simulation study and discussions for further study follow.

2 The multicategory problem and the Bayes rule

In the classification problem, we are given a training data set that consists of n data points $(\mathbf{x}_i, y_i)$ for $i = 1, \cdots, n$. $\mathbf{x}_i \in R^d$ represents covariates and y_i denotes the class label of the ith data point. The task is to learn a classification rule $\phi(\mathbf{x})$: $R^d \to \{1, \cdots, k\}$ that well matches attributes $\mathbf{x}_i$ to a class label y_i. We assume that each $(\mathbf{x}_i, y_i)$ is an independent random sample from a target population with probability distribution $P(\mathbf{x}, y)$. Let (X, Y) denote a generic pair of a random sample from $P(\mathbf{x}, y)$, and $p_j(\mathbf{x}) = P(Y = j|X = \mathbf{x})$ be the conditional probability of class j given $X = \mathbf{x}$ for $j = 1, \cdots, k$. When the misclassification costs are all equal, the loss function is

$$l(y, \phi(\mathbf{x})) = I(y \neq \phi(\mathbf{x})) \tag{2.1}$$

where $I(\cdot)$ is the indicator function, which assumes 1 if its argument is true, and 0 otherwise. In a decision theoretic formulation, the best classification rule would be the one that minimizes the expected misclassification rate given by

$$\phi_B(\mathbf{x}) = arg \min_{j=1,\cdots,k} [1 - p_j(\mathbf{x})] = arg \max_{j=1,\cdots,k} p_j(\mathbf{x}). \tag{2.2}$$

If we knew the conditional probabilities $p_j(\mathbf{x})$, we can implement the best classification rule $\phi_B(\mathbf{x})$, often called the Bayes rule. Since we rarely know $p_j(\mathbf{x})$'s in reality, we need to approximate the Bayes rule by learning from a training data set. A common way to approximate it is to estimate $p_j(\mathbf{x})$'s or equivalently the log odds $\log[p_j(\mathbf{x})/p_k(\mathbf{x})]$ from data first and to plug them into the rule. Different from such conventional approximation, Lin (1999) showed that SVMs target directly at the Bayes rule without estimating the component $p_1(\mathbf{x})$ when $k = 2$. Note that the representation of class label Y in the SVMs literature for $k = 2$ is either 1 or -1, instead of 1 or 2 as stated here, and the Bayes rule is then $\phi_B(\mathbf{x}) = sign(p_1(\mathbf{x}) - 1/2)$ in the symmetric representation.

Now consider the case when misclassification costs are not equal, which may be more useful in solving real world problem. First, we define $C_{j\ell}$ for $j, \ell = 1, \cdots, k$ as the cost of misclassifying an example from class j to class ℓ. C_{jj} for $j = 1, \cdots, k$ are all zero since the correct decision should not be penalized. The loss function is

$$l(y, \phi(\mathbf{x})) = \sum_{j=1}^{k} I(y = j) \left(\sum_{\ell=1}^{k} C_{j\ell} I(\phi(\mathbf{x}) = \ell) \right). \tag{2.3}$$

Analogous to the equal cost case, the best classification rule is given by

$$\phi_B(\mathbf{x}) = arg \min_{j=1,\cdots,k} \sum_{\ell=1}^{k} C_{\ell j} p_\ell(\mathbf{x}). \tag{2.4}$$

Notice that when the misclassification costs are all equal, say, $C_{j\ell} = 1$, $j \neq \ell$ then the Bayes rule derived just now is nicely reduced to the Bayes rule in the equal cost case. Besides the concern with different misclassification costs, sampling bias is an issue that needs special attention in the classification problem. So far, we assume that training data are truly from the general population that would generate future samples. However, it's often the case that while we collect data, we tend to balance each class by oversampling minor class examples and downsampling major class examples. The sampling bias leads to distortion of the class proportions, which would influence the classification rule. If we know the prior class proportions, then there is a remedy for the sampling bias by incorporating the discrepancy between the sample proportions and the population proportions into the cost component. Let π_j be the prior proportion of class j in the general population, and π_j^s be the prespecified proportion of class j examples in a training data set. π_j^s may be different from π_j if the sampling bias has occurred. Define $g_j(\mathbf{x})$ the probability density of X for class j population, $j = 1, \cdots, k$, and let (X^s, Y^s) be a random sample obtained by the sampling mechanism used in the data collection stage. Then the difference between (X^s, Y^s) in the training data and (X, Y) in the general population is clear when we look at the conditional probabilities. While

$$p_j(\mathbf{x}) \quad = P(Y = j|X = \mathbf{x}) \quad = \frac{\pi_j g_j(\mathbf{x})}{\sum_{\ell=1}^k \pi_\ell g_\ell(\mathbf{x})}, \tag{2.5}$$

$$p_j^s(\mathbf{x}) \quad = P(Y^s = j|X^s = \mathbf{x}) \quad = \frac{\pi_j^s g_j(\mathbf{x})}{\sum_{\ell=1}^k \pi_\ell^s g_\ell(\mathbf{x})}. \tag{2.6}$$

Since we learn a classification rule only through the training data, it is better to express the Bayes rule in terms of the quantities for (X^s, Y^s) and π_j which we assume to know a priori. One can verify that the following is equivalent to (2.4).

$$\phi_B(\mathbf{x}) = arg \min_{j=1,\cdots,k} \sum_{\ell=1}^k \frac{\pi_\ell}{\pi_\ell^s} C_{\ell j} p_\ell^s(\mathbf{x}) = arg \min_{j=1,\cdots,k} \sum_{\ell=1}^k l_{\ell j} p_\ell^s(\mathbf{x}) \tag{2.7}$$

where $l_{\ell j}$ is defined as $(\pi_\ell/\pi_\ell^s)C_{\ell j}$, which is a modified cost that takes the sampling bias into account together with the original misclassification cost. For more details on the two-category case, see Lin, Lee & Wahba (2000). In the paper, we call the case when misclassification costs are not equal or there is a sampling bias, nonstandard, as opposed to the standard case when misclassification costs are all equal, and there is no sampling bias. In the following section, we will develop an extended SVMs methodology to approximate the Bayes rule for multicategory standard case. Then we will modify it for the nonstandard case.

3 The standard multicategory SVM

Throughout this section, we assume that all the misclassification costs are equal and there is no sampling bias in the training data set. We briefly go over the standard SVMs for $k = 2$. SVMs have their roots in a geometrical interpretation of the classification problem as a problem of finding a separating hyperplane in multidimensional input space. The class labels y_i are either 1 or -1 in the SVM setting. The symmetry in the representation of y_i is very essential in the mathematical formulation of SVMs. Then SVM methodology seeks a function $f(\mathbf{x}) = h(\mathbf{x}) + b$ with $h \in H_K$ a reproducing kernel Hilbert space (RKHS) and b, a constant minimizing

$$\frac{1}{n}\sum_{i=1}^n (1 - y_i f(\mathbf{x}_i))_+ + \lambda \|h\|_{H_K}^2 \tag{3.1}$$

where $(x)_+ = x$ if $x \geq 0$ and 0 otherwise. $\|h\|_{H_K}^2$ denotes the norm of the function h defined in the RKHS with the the reproducing kernel function $K(\cdot, \cdot)$, measuring the complexity or smoothness of h. For more information on RKHS, see Wahba (1990). λ is a given tuning parameter which balances the data fit and the complexity of $f(\mathbf{x})$. The classification rule $\phi(\mathbf{x})$ induced by $f(\mathbf{x})$ is $\phi(\mathbf{x}) = sign[f(\mathbf{x})]$. The function $f(\mathbf{x})$ yields the level curve defined by $f(\mathbf{x}) = 0$ in R^d, which is the classification boundary of the rule $\phi(\mathbf{x})$. Note that the loss function $(1 - y_i f(\mathbf{x}_i))_+$, often called the hinge loss, is closely related to the misclassification loss function, which can be reexpressed as $[-y_i\phi(\mathbf{x}_i)]_* = [-y_i f(\mathbf{x}_i)]_*$ where $[x]_* = I(x \geq 0)$. Indeed, the former is an upper bound of the latter, and when the resulting $f(\mathbf{x}_i)$ is close to either 1 or -1, the hinge loss function is close to 2 times the misclassification loss. Let us consider the simplest case to motivate the SVM loss function. Take the function space to be $\{f(\mathbf{x}) = \mathbf{x}\cdot\mathbf{w}+b \mid \mathbf{w} \in R^d \text{ and } b \in R\}$. If the training data set is linearly separable, there exists linear $f(\mathbf{x})$ satisfying the following condition for $i = 1, \cdots, n$:

$$f(\mathbf{x}_i) \quad \geq 1 \quad \text{if } y_i = 1 \tag{3.2}$$

$$f(\mathbf{x}_i) \quad \leq -1 \quad \text{if } y_i = -1 \tag{3.3}$$

Or more succinctly, $1-y_i f(\mathbf{x}_i) \leq 0$ for $i = 1, \cdots, n$. Then the separating hyperplane $\mathbf{x}\cdot\mathbf{w} + b = 0$ separates all the positive examples from the negative examples, and SVM looks for the hyperplane with maximum margin that is the sum of the shortest distance from the hyperplane to the closest positive example and the closest negative example. In the nonseparable case, the SVM loss function measures the data fit by $(1 - y_i f(\mathbf{x}_i))_+$, which could be zero for all data points in the separable case. The notion of separability can be extended for a general RKHS. A training data set is said to be separable if there exists such $f(\mathbf{x})$ in the function space we assume, satisfying the condition (3.2) and (3.3). Notice that the data fit functional $\sum_{i=1}^n (1-y_i f(\mathbf{x}_i))_+$ penalizes violation of the separability condition, while the complexity $\|h\|^2_{H_K}$ of $f(\mathbf{x})$ is also penalized to avoid overfitting. Lin (1999) showed that, if the reproducing kernel Hilbert space is rich enough, the solution $f(\mathbf{x})$ approaches the Bayes rule $sign(p_1(\mathbf{x}) - 1/2)$, as the sample size n goes to ∞ for appropriately chosen λ. For example, Gaussian kernel is one of typically used kernels for SVMs, RKHS induced by which is flexible enough to approximate the Bayes rule. The argument in the paper is based on the fact that the target function of SVMs can be identified as the minimizer of the limit of the data fit functional. Bearing this idea in mind, we extend SVMs methodology by devising a data fit functional for the multicategory case which would encompass that of two-category SVMs.

Consider the k-category classification problem. To carry over the symmetry in representation of class labels, we use the following vector representation of class label. For notational convenience, we define $\mathbf{v}_j$ for $j = 1, \cdots, k$ as a k-dimensional vector with 1 in the jth coordinate and $-\frac{1}{k-1}$ elsewhere. Then, $\mathbf{y}_i$ is coded as $\mathbf{v}_j$ if example i belongs to class j. For example, if example i belongs to class 1, $\mathbf{y}_i = \mathbf{v}_1 = (1, -\frac{1}{k-1}, \cdots, -\frac{1}{k-1})$. Similarly, if it belongs to class k, $\mathbf{y}_i = \mathbf{v}_k = (-\frac{1}{k-1}, \cdots, -\frac{1}{k-1}, 1)$. Accordingly, we define a k-tuple of separating functions $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_k(\mathbf{x}))$ with the sum-to-zero constraint, $\sum_{j=1}^k f_j(\mathbf{x}) = 0$ for any $\mathbf{x} \in R^d$. Note that the constraint holds implicitly for coded class labels $\mathbf{y}_i$. Analogous to the two-category case, we consider $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_k(\mathbf{x})) \in \prod_{j=1}^k(\{1\} + H_{K_j})$, the product space of k reproducing kernel Hilbert spaces H_{K_j} for $j = 1, \cdots, k$. In other words, each component $f_j(\mathbf{x})$ can be expressed as $h_j(\mathbf{x}) + b_j$ with $h_j \in H_{K_j}$. Unless there is compelling reason to believe that H_{K_j} should be different for $j = 1, \cdots, k$, we will assume they are the same RKHS denoted by H_K. Define Q as the k by k matrix with 0 on the diagonal, and 1 elsewhere. It represents the cost matrix when all the misclassification costs are equal. Let L be a function which maps a class label $\mathbf{y}_i$ to the jth row of the matrix Q if $\mathbf{y}_i$ indicates class j. So, if $\mathbf{y}_i$ represents class j, then $L(\mathbf{y}_i)$ is a k dimensional vector with 0 in the jth coordinate, and 1 elsewhere. Now, we propose that to find $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_k(\mathbf{x})) \in \prod_1^k(\{1\} + H_K)$, with the sum-to-zero constraint, minimizing the following quantity is a natural extension of SVMs methodology.

$$\frac{1}{n}\sum_{i=1}^{n} L(\mathbf{y}_i)\cdot(\mathbf{f}(\mathbf{x}_i)-\mathbf{y}_i)_+ + \frac{1}{2}\lambda\sum_{j=1}^{k}\|h_j\|^2_{H_K} \tag{3.4}$$

where $(\mathbf{f}(\mathbf{x}_i)-\mathbf{y}_i)_+$ means $[(f_1(\mathbf{x}_i)-y_{i1})_+,\cdots,(f_k(\mathbf{x}_i)-y_{ik})_+]$ by taking the truncate function $(\cdot)_+$ componentwise, and $\cdot$ operation in the data fit functional indicates the Euclidean inner product.

As we observe in the hinge loss function of the binary case, the proposed loss function has analogous relation to the multicategory misclassification loss (2.1). If $\mathbf{f}(\mathbf{x}_i)$ itself is one of the class representations, $L(\mathbf{y}_i)\cdot(\mathbf{f}(\mathbf{x}_i)-\mathbf{y}_i)_+$ is $\frac{k}{k-1}$ times the misclassification loss. We can verify that the binary SVM formulation (3.1) is a special case of (3.4) when $k=2$. Check that if $\mathbf{y}_i=(1,-1)$ (1 in SVMs notation), then $L(\mathbf{y}_i)\cdot(\mathbf{f}(\mathbf{x}_i)-\mathbf{y}_i)_+=(0,1)\cdot[(f_1(\mathbf{x}_i)-1)_+,(f_2(\mathbf{x}_i)+1)_+]=(f_2(\mathbf{x}_i)+1)_+=(1-f_1(\mathbf{x}_i))_+$. Similarly, if $\mathbf{y}_i=(-1,1)$ (-1 in SVMs notation), $L(\mathbf{y}_i)\cdot(\mathbf{f}(\mathbf{x}_i)-\mathbf{y}_i)_+=(f_1(\mathbf{x}_i)+1)_+$. So the data fit functionals in (3.1) and (3.4) are identical, f_1 playing the same role as f in (3.1). Since $\frac{1}{2}\lambda\sum_{j=1}^{2}\|h_j\|^2_{H_K}=\frac{1}{2}\lambda(\|h_1\|^2_{H_K}+\|-h_1\|^2_{H_K})=\lambda\|h_1\|^2_{H_K}$, the remaining model complexity parts are also identical. The limit of the data fit functional for (3.4) is $E[L(Y)\cdot(\mathbf{f}(X)-Y)_+]$. Like the two-category case, we can identify the target function by finding a minimizer of the limit data fit functional. The following lemma shows the asymptotic target function of (3.4).

Lemma 3.1. *The minimizer of $E[L(Y)\cdot(\mathbf{f}(X)-Y)_+]$ under the sum-to-zero constraint is $\mathbf{f}(\mathbf{x})=(f_1(\mathbf{x}),\cdots,f_k(\mathbf{x}))$ with*

$$f_j(\mathbf{x})=\begin{cases}1 & \text{if } j=arg\max_{\ell=1,\cdots,k}p_\ell(\mathbf{x})\\ -\frac{1}{k-1} & \text{otherwise}\end{cases} \tag{3.5}$$

Proof: Since $E[L(Y)\cdot(\mathbf{f}(X)-Y)_+]=E(E[L(Y)\cdot(\mathbf{f}(X)-Y)_+|X])$, we can minimize $E[L(Y)\cdot(\mathbf{f}(X)-Y)_+]$ by minimizing $E[L(Y)\cdot(\mathbf{f}(X)-Y)_+|X=\mathbf{x}]$ for every $\mathbf{x}$. If we write out the functional for each $\mathbf{x}$, we have

$$\begin{aligned}E[L(Y)\cdot(\mathbf{f}(X)-Y)_+|X=\mathbf{x}] &= \sum_{j=1}^{k}\left(\sum_{\ell\neq j}(f_\ell(\mathbf{x})+\frac{1}{k-1})_+\right)p_j(\mathbf{x}) && (3.6)\\ &= \sum_{j=1}^{k}\left(\sum_{\ell\neq j}p_\ell(\mathbf{x})\right)(f_j(\mathbf{x})+\frac{1}{k-1})_+ && (3.7)\\ &= \sum_{j=1}^{k}(1-p_j(\mathbf{x}))(f_j(\mathbf{x})+\frac{1}{k-1})_+. && (3.8)\end{aligned}$$

Here, we claim that it is sufficient to search over $\mathbf{f}(\mathbf{x})$ with $f_j(\mathbf{x})\geq-\frac{1}{k-1}$ for all $j=1,\cdots,k$, to minimize (3.8). If any $f_j(\mathbf{x})<-\frac{1}{k-1}$, then we can always find another $\mathbf{f}^*(\mathbf{x})$ which is better than or as good as $\mathbf{f}(\mathbf{x})$ in reducing the expected loss as follows. Set $f_j^*(\mathbf{x})$ to be $-\frac{1}{k-1}$ and subtract the surplus $-\frac{1}{k-1}-f_j(\mathbf{x})$ from other component $f_\ell(\mathbf{x})$'s which are greater than $-\frac{1}{k-1}$. Existence of such other components is always guaranteed by the sum-to-zero constraint. Determine $f_i^*(\mathbf{x})$ in accordance with the modifications. By doing so, we get $\mathbf{f}^*(\mathbf{x})$ such that $(f_j^*(\mathbf{x})+\frac{1}{k-1})_+\leq(f_j(\mathbf{x})+\frac{1}{k-1})_+$ for each j. Since the expected loss is a nonnegatively weighted sum of $(f_j(\mathbf{x})+\frac{1}{k-1})_+$, it is sufficient to consider $\mathbf{f}(\mathbf{x})$ with $f_j(\mathbf{x}) \geq -\frac{1}{k-1}$ for all $j = 1, \cdots, k$. Dropping the truncate functions from (3.8), and rearranging, we get

$$\begin{aligned}
& E[L(Y) \cdot (\mathbf{f}(X) - Y)_+ | X = \mathbf{x}] \\
= & \sum_{j=1}^{k} (1 - p_j(\mathbf{x}))(f_j(\mathbf{x}) + \frac{1}{k-1}) && (3.9) \\
= & 1 + \sum_{j=1}^{k-1} (1 - p_j(\mathbf{x})) f_j(\mathbf{x}) + (1 - p_k(\mathbf{x}))(-\sum_{j=1}^{k-1} f_j(\mathbf{x})) && (3.10) \\
= & 1 + \sum_{j=1}^{k-1} (p_k(\mathbf{x}) - p_j(\mathbf{x})) f_j(\mathbf{x}). && (3.11)
\end{aligned}$$

Without loss of generality, we may assume that $k = arg \max_{j=1,\cdots,k} p_j(\mathbf{x})$ by the symmetry in the class labels. This implies that to minimize the expected loss, $f_j(\mathbf{x})$ should be $-\frac{1}{k-1}$ for $j = 1, \cdots, k-1$ because of the nonnegativity of $p_k(\mathbf{x}) - p_j(\mathbf{x})$. Finally, we have $f_k(\mathbf{x}) = 1$ by the sum-to-zero constraint. □

Indeed, Lemma 3.1 is a multicategory extension of Lemma 3.1 in Lin (1999) which was the key idea to show that $f(\mathbf{x})$ in ordinary SVMs approximates $sign(p_1(\mathbf{x}) - 1/2)$ asymptotically. So, if the reproducing kernel Hilbert space is flexible enough to approximate the minimizer in Lemma 3.1, and λ is chosen appropriately, the solution $\mathbf{f}(\mathbf{x})$ to (3.4) approaches it as the sample size n goes to ∞. Notice that the minimizer is exactly the representation of the most probable class. Hence, the classification rule induced by $\mathbf{f}(\mathbf{x})$ is naturally $\phi(\mathbf{x}) = arg \max_j f_j(\mathbf{x})$. If $\mathbf{f}(\mathbf{x})$ is the minimizer in Lemma 3.1, then the corresponding classification rule is $\phi_B(\mathbf{x}) = arg \max_j p_j(\mathbf{x})$, the Bayes rule (2.2) for the standard multicategory case.

4 The nonstandard multicategory SVM

In this section, we allow different misclassification costs and the possibility of sampling bias mentioned in Section 2. Necessary modification of the multicategory SVM (3.4) to accommodate such differences is straightforward. First, let's consider different misclassification costs only, assuming no sampling bias. Instead of the matrix Q used in the definition of $L(\mathbf{y}_i)$, define a k by k cost matrix C with entry $C_{j\ell}$ for $j, \ell = 1, \cdots, k$ meaning the cost of misclassifying an example from class j to class ℓ. All the diagonal entries C_{jj} for $j = 1, \cdots, k$ would be zero. Modify $L(\mathbf{y}_i)$ in (3.4) to the jth row of the cost matrix C if $\mathbf{y}_i$ indicates class j. When all the misclassification costs $C_{j\ell}$ are equal to 1, the matrix C becomes the matrix Q. So, the modification of the map $L(\cdot)$ encompasses Q for standard case. Now, we consider the sampling bias concern together with unequal costs. As illustrated in Section 2, we need a transition from (X, Y) to (X^s, Y^s) to differentiate a "training example" population from the general population. In this case, with little abuse of notation we redefine a generalized cost matrix L whose entry $l_{j\ell}$ is given by $(\pi_j/\pi_j^s)C_{j\ell}$ for $j, \ell = 1, \cdots, k$. Accordingly, define $L(\mathbf{y}_i)$ to be the jth row of the matrix L if $\mathbf{y}_i$ indicates class j. When there is no sampling bias, in other words, $\pi_j = \pi_j^s$ for all j, the generalized cost matrix L reduces to the ordinary cost matrix C. With the finalized version of the cost matrix L and the map $L(\mathbf{y}_i)$, the multicategory SVM formulation (3.4) still holds as the general scheme. The following lemma identifies the minimizer of the limit of the data fit functional, which is $E[L(Y^s) \cdot (\mathbf{f}(X^s) - Y^s)_+]$.

Lemma 4.1. *The minimizer of $E[L(Y^s)\cdot(\mathbf{f}(X^s)-Y^s)_+]$ under the sum-to-zero constraint is $\mathbf{f}(\mathbf{x})=(f_1(\mathbf{x}),\cdots,f_k(\mathbf{x}))$ with*

$$f_j(\mathbf{x})=\begin{cases} 1 & \textit{if } j=\textit{arg}\min_{\ell=1,\cdots,k}\sum_{m=1}^k l_{m\ell}p_m^s(\mathbf{x}) \\ -\frac{1}{k-1} & \textit{otherwise} \end{cases} \tag{4.1}$$

Proof: Parallel to all the arguments used for the proof of Lemma 3.1, it can be shown that

$$\begin{aligned} & E[L(Y^s)\cdot(\mathbf{f}(X^s)-Y^s)_+|X^s=\mathbf{x}] \\ = \quad & \frac{1}{k-1}\sum_{j=1}^k\sum_{\ell=1}^k l_{\ell j}p_\ell^s(\mathbf{x})+\sum_{j=1}^k\left(\sum_{\ell=1}^k l_{\ell j}p_\ell^s(\mathbf{x})\right)f_j(\mathbf{x}) \end{aligned} \tag{4.2}$$

We can immediately eliminate the first term which does not involve any $f_j(\mathbf{x})$ from our consideration. To make the equation simpler, let $W_j(\mathbf{x})$ be $\sum_{\ell=1}^k l_{\ell j}p_\ell^s(\mathbf{x})$ for $j=1,\cdots,k$. Then the whole equation reduces to the following up to a constant.

$$\begin{aligned} \sum_{j=1}^k W_j(\mathbf{x})f_j(\mathbf{x}) &= \sum_{j=1}^{k-1} W_j(\mathbf{x})f_j(\mathbf{x})+W_k(\mathbf{x})(-\sum_{j=1}^{k-1}f_j(\mathbf{x})) && (4.3) \\ &= \sum_{j=1}^{k-1}(W_j(\mathbf{x})-W_k(\mathbf{x}))f_j(\mathbf{x}) && (4.4) \end{aligned}$$

Without loss of generality, we may assume that $k=\textit{arg}\min_{j=1,\cdots,k}W_j(\mathbf{x})$. To minimize the expected quantity, $f_j(\mathbf{x})$ should be $-\frac{1}{k-1}$ for $j=1,\cdots,k-1$ because of the nonnegativity of $W_j(\mathbf{x})-W_k(\mathbf{x})$ and $f_j(\mathbf{x})\geq-\frac{1}{k-1}$ for all $j=1,\cdots,k$. Finally, we have $f_k(\mathbf{x})=1$ by the sum-to-zero constraint. □

It is not hard to see that Lemma 3.1 is a special case of the above lemma. Like the standard case, Lemma 4.1 has its existing counterpart when $k=2$. See Lemma 3.1 in Lin, Lee & Wahba (2000) with the caution that $\mathbf{y}_i$, and $L(\mathbf{y}_i)$ are defined differently than here. Again, the lemma implies that if the reproducing kernel Hilbert space is rich enough to approximate the minimizer in Lemma 4.1, for appropriately chosen λ, we would observe the solution to (3.4) to be very close to the minimizer for a large sample. A classification rule induced by $\mathbf{f}(\mathbf{x})$ is $\phi(\mathbf{x})=\textit{arg}\max_j f_j(\mathbf{x})$ by the same reasoning as in the standard case. Especially, the classification rule derived from the minimizer in Lemma 4.1 is $\phi_B(\mathbf{x})=\textit{arg}\min_{j=1,\cdots,k}\sum_{\ell=1}^k l_{\ell j}p_\ell^s(\mathbf{x})$, the Bayes rule (2.7) for the nonstandard multicategory case.

5 Dual problem for the multicategory SVM

We now switch to a Lagrangian formulation of the problem (3.4). The problem of finding constrained functions $(f_1(\mathbf{x}),\cdots,f_k(\mathbf{x}))$ minimizing (3.4) is then transformed into that of finding finite dimensional coefficients instead, with the aid of a variant of the representer theorem. For the representer theorem in a regularization framework, see Kimeldorf & Wahba (1971) or Wahba (1998). The following lemma says that we can still apply the representer theorem to each component $f_j(\mathbf{x})$ with, however some restrictions on the coefficients due to the sum-to-zero constraint.

Lemma 5.1. *To find $(f_1(\mathbf{x}), \cdots, f_k(\mathbf{x})) \in \prod_1^k(\{1\} + H_K)$, with the sum-to-zero constraint, minimizing (3.4) is equivalent to find $(f_1(\mathbf{x}), \cdots, f_k(\mathbf{x}))$ of the form*

$$f_j(\mathbf{x}) = b_j + \sum_{i=1}^{n} c_{ij} K(\mathbf{x}_i, \mathbf{x}) \quad \textit{for } j = 1, \cdots, k \tag{5.1}$$

with the sum-to-zero constraint only at $\mathbf{x}_i$ for $i = 1, \cdots, n$, minimizing (3.4).

Proof. Consider $f_j(\mathbf{x}) = b_j + h_j(\mathbf{x})$ with $h_j \in H_K$. Decompose $h_j(\cdot) = \sum_{\ell=1}^n c_{\ell j} K(\mathbf{x}_\ell, \cdot) + \rho_j(\cdot)$ for $j = 1, \cdots, k$ where c_{ij}'s are some constants, and $\rho_j(\cdot)$ is the element in the RKHS orthogonal to the span of $\{K(\mathbf{x}_i, \cdot), i = 1, \cdots, n\}$. $f_k(\cdot) = -\sum_{j=1}^{k-1} b_j - \sum_{j=1}^{k-1}\sum_{i=1}^n c_{ij} K(\mathbf{x}_i, \cdot) - \sum_{j=1}^{k-1} \rho_j(\cdot)$ by the sum-to-zero constraint. By the definition of the reproducing kernel $K(\cdot, \cdot)$, $(h_j, K(\mathbf{x}_i, \cdot))_{H_K} = h_j(\mathbf{x}_i)$ for $i = 1, \cdots, n$. Then,

$$\begin{aligned} f_j(\mathbf{x}_i) &= b_j + h_j(\mathbf{x}_i) = b_j + (h_j, K(\mathbf{x}_i, \cdot))_{H_K} && (5.2)\\ &= b_j + (\sum_{\ell=1}^n c_{\ell j} K(\mathbf{x}_\ell, \cdot) + \rho_j(\cdot), K(\mathbf{x}_i, \cdot))_{H_K} && (5.3)\\ &= b_j + \sum_{\ell=1}^n c_{\ell j} K(\mathbf{x}_\ell, \mathbf{x}_i) && (5.4) \end{aligned}$$

So, the data fit functional in (3.4) does not depend on $\rho_j(\cdot)$ at all for $j = 1, \cdots, k$. On the other hand, we have $\|h_j\|_{H_K}^2 = \sum_{i,\ell} c_{ij} c_{\ell j} K(\mathbf{x}_\ell, \mathbf{x}_i) + \|\rho_j\|_{H_K}^2$ for $j = 1, \cdots, k-1$, and $\|h_k\|_{H_K}^2 = \| \sum_{j=1}^{k-1} \sum_{i=1}^n c_{ij} K(\mathbf{x}_i, \cdot)\|_{H_K}^2 + \| \sum_{j=1}^{k-1} \rho_j\|_{H_K}^2$. To minimize (3.4), obviously $\rho_j(\cdot)$ should vanish. It remains to show that minimizing (3.4) under the sum-to-zero constraint at the data points only is equivalent to minimizing (3.4) under the constraint for every $\mathbf{x}$. With some abuse of notation, let K be now the n by n matrix with $i\ell$ th entry $K(\mathbf{x}_i, \mathbf{x}_\ell)$. Let $\mathbf{e}$ be the column vector with n ones, and $\mathbf{c}_{\cdot j} = (c_{1j}, \cdots, c_{nj})^t$. Given the representation (5.1), consider the problem of minimizing (3.4) under $(\sum_{j=1}^k b_j)\mathbf{e} + K(\sum_{j=1}^k \mathbf{c}_{\cdot j}) = 0$. For any $f_j(\cdot) = b_j + \sum_{i=1}^n c_{ij} K(\mathbf{x}_i, \cdot)$ satisfying $(\sum_{j=1}^k b_j)\mathbf{e} + K(\sum_{j=1}^k \mathbf{c}_{\cdot j}) = 0$, define the centered solution $f_j^*(\cdot) = b_j^* + \sum_{i=1}^n c_{ij}^* K(\mathbf{x}_i, \cdot) = (b_j - \bar{b}) + \sum_{i=1}^n (c_{ij} - \bar{c}_i) K(\mathbf{x}_i, \cdot)$ where $\bar{b} = \frac{1}{k}\sum_{j=1}^k b_j$ and $\bar{c}_i = \frac{1}{k}\sum_{j=1}^k c_{ij}$. Then $f_j(\mathbf{x}_i) = f_j^*(\mathbf{x}_i)$, and

$$\sum_{j=1}^k \|h_j^*\|_{H_K}^2 = \sum_{j=1}^k \mathbf{c}_{\cdot j}^t K \mathbf{c}_{\cdot j} - k\bar{\mathbf{c}}^t K \bar{\mathbf{c}} \leq \sum_{j=1}^k \mathbf{c}_{\cdot j}^t K \mathbf{c}_{\cdot j} = \sum_{j=1}^k \|h_j\|_{H_K}^2. \tag{5.5}$$

Since the equality holds only when $K\bar{\mathbf{c}} = 0$, that is, $K(\sum_{j=1}^k \mathbf{c}_{\cdot j}) = 0$, we know that at the minimizer, $K(\sum_{j=1}^k \mathbf{c}_{\cdot j}) = 0$, and therefore $\sum_{j=1}^k b_j = 0$. Observe that $K(\sum_{j=1}^k \mathbf{c}_{\cdot j}) = 0$ implies $(\sum_{j=1}^k \mathbf{c}_{\cdot j})^t K(\sum_{j=1}^k \mathbf{c}_{\cdot j}) = \| \sum_{i=1}^n (\sum_{j=1}^k c_{ij}) K(\mathbf{x}_i, \cdot)\|_{H_K}^2 = \| \sum_{j=1}^k \sum_{i=1}^n c_{ij} K(\mathbf{x}_i, \cdot)\|_{H_K}^2 = 0$. It means $\sum_{j=1}^k \sum_{i=1}^n c_{ij} K(\mathbf{x}_i, \mathbf{x}) = 0$ for every $\mathbf{x}$. Hence, minimizing (3.4) under the sum-to-zero constraint at the data points is equivalent to minimizing (3.4) under $\sum_{j=1}^k b_j + \sum_{j=1}^k \sum_{i=1}^n c_{ij} K(\mathbf{x}_i, \mathbf{x}) = 0$ for every $\mathbf{x}$. □

Remark 5.1. *If the reproducing kernel K is strictly positive definite, then the sum-to-zero constraint at the data points can be replaced by the equality constraints $\sum_{j=1}^k b_j = 0$ and $\sum_{j=1}^k \mathbf{c}_{\cdot j} = 0$.*

We introduce a vector of nonnegative slack variables $\xi_i \in R^k$ for the term $(\mathbf{f}(\mathbf{x}_i) - \mathbf{y}_i)_+$. By the above lemma, we can write the primal problem in terms of b_j and c_{ij}. Since the problem has k class components involved in a symmetrical way, we can rewrite it more succinctly in vector notation. Let $L_j \in R^n$ for $j = 1, \cdots, k$ be the jth column of the n by k matrix with the ith row $L(\mathbf{y}_i)$. Let $\xi_{\cdot j} \in R^n$ for $j = 1, \cdots, k$ be the jth column of the n by k matrix with the ith row ξ_i. Similarly, $\mathbf{y}_{\cdot j}$ denotes the jth column of the n by k matrix with the ith row $\mathbf{y}_i$. Then, the primal problem in vector notation is

$$\min L_P = \sum_{j=1}^k L_j^t \xi_{\cdot j} + \frac{1}{2} n\lambda \sum_{j=1}^k \mathbf{c}_{\cdot j}^t K \mathbf{c}_{\cdot j} \tag{5.6}$$

$$\text{subject to} \qquad b_j \mathbf{e} + K\mathbf{c}_{\cdot j} - \mathbf{y}_{\cdot j} \le \xi_{\cdot j} \qquad \text{for } j = 1, \cdots, k \tag{5.7}$$

$$\xi_{\cdot j} \ge 0 \qquad \text{for } j = 1, \cdots, k \tag{5.8}$$

$$(\textstyle\sum_{j=1}^k b_j)\mathbf{e} + K(\sum_{j=1}^k \mathbf{c}_{\cdot j}) = 0 \tag{5.9}$$

To derive its Wolfe dual problem, we introduce nonnegative Lagrange multipliers $\alpha_j \in R^n$ for (5.7), nonnegative Lagrange multipliers $\gamma_j \in R^n$ for (5.8), and unconstrained Lagrange multipliers $\delta_{\mathbf{f}} \in R^n$ for (5.9), the equality constraints. Then, the dual problem becomes a problem of maximizing

$$\begin{aligned} L_D \quad = \quad & \sum_{j=1}^k L_j^t \xi_{\cdot j} + \frac{1}{2} n\lambda \sum_{j=1}^k \mathbf{c}_{\cdot j}^t K \mathbf{c}_{\cdot j} + \sum_{j=1}^k \alpha_j^t (b_j \mathbf{e} + K\mathbf{c}_{\cdot j} - \mathbf{y}_{\cdot j} - \xi_{\cdot j}) \\ & - \sum_{j=1}^k \gamma_j^t \xi_{\cdot j} + \delta_f^t \left((\sum_{j=1}^k b_j)\mathbf{e} + K(\sum_{j=1}^k \mathbf{c}_{\cdot j}) \right) \end{aligned} \tag{5.10}$$

subject to for $j = 1, \cdots, k$,

$$\frac{\partial L_D}{\partial \xi_{\cdot j}} = L_j - \alpha_j - \gamma_j = 0 \tag{5.11}$$

$$\frac{\partial L_D}{\partial \mathbf{c}_{\cdot j}} = n\lambda K \mathbf{c}_{\cdot j} + K\alpha_j + K\delta_{\mathbf{f}} = 0 \tag{5.12}$$

$$\frac{\partial L_D}{\partial b_j} = (\alpha_j + \delta_f)^t \mathbf{e} = 0 \tag{5.13}$$

$$\alpha_j \ge 0 \tag{5.14}$$

$$\gamma_j \ge 0 \tag{5.15}$$

Let $\bar{\alpha}$ be $\frac{1}{k}\sum_{j=1}^k \alpha_j$. Since δ_f is unconstrained, one may take $\delta_f = -\bar{\alpha}$ from (5.13). Accordingly, (5.13) becomes $(\alpha_j - \bar{\alpha})^t \mathbf{e} = 0$. Eliminating all the primal variables in L_D by the equality constraint (5.11) and using relations from (5.12) and (5.13), we have the following dual problem.

$$\min_{\alpha_j} L_D = \frac{1}{2} \sum_{j=1}^k (\alpha_j - \bar{\alpha})^t K (\alpha_j - \bar{\alpha}) + n\lambda \sum_{j=1}^k \alpha_j^t \mathbf{y}_{\cdot j} \tag{5.16}$$

$$\text{subject to} \qquad 0 \le \alpha_j \le L_j \qquad \text{for } j = 1, \cdots, k \tag{5.17}$$

$$(\alpha_j - \bar{\alpha})^t \mathbf{e} = 0 \qquad \text{for } j = 1, \cdots, k \tag{5.18}$$

Once we solve the quadratic problem, we can take $\mathbf{c}_{\cdot j} = -\frac{1}{n\lambda}(\alpha_j - \bar{\alpha})$ for $j = 1, \cdots, k$ from (5.12). Note that if the matrix K is not strictly positive definite, then $\mathbf{c}_{\cdot j}$ is not uniquely determined. b_j can be found from any of the examples with $0 < \alpha_{ij} < l_{ij}$. By the Karush-Kuhn-Tucker complementarity conditions, the solution should satisfy

$$\alpha_j \perp (b_j \mathbf{e} + K\mathbf{c}_{\cdot j} - \mathbf{y}_{\cdot j} - \xi_{\cdot j}) \quad \text{for } j = 1, \cdots, k \tag{5.19}$$

$$\gamma_j = (L_j - \alpha_j) \perp \xi_{\cdot j} \quad \text{for } j = 1, \cdots, k \tag{5.20}$$

where $\perp$ means that componentwise products are all zero. If $0 < \alpha_{ij} < l_{ij}$ for some i, then ξ_{ij} should be zero from (5.20), and accordingly we have $b_j + \sum_{\ell=1}^{n} c_{\ell j} K(\mathbf{x}_\ell, \mathbf{x}_i) - y_{ij} = 0$ from (5.19). If there is no example satisfying $0 < \alpha_{ij} < l_{ij}$ for some class j, $\mathbf{b} = (b_1, \cdots, b_k)$ is determined as the solution to the following problem :

$$\min_{b_j} \tfrac{1}{n} \textstyle\sum_{i=1}^{n} L(\mathbf{y}_i) \cdot (\mathbf{h}_i + \mathbf{b} - \mathbf{y}_i)_+ \tag{5.21}$$

$$\text{subject to} \quad \textstyle\sum_{j=1}^{k} b_j = 0 \tag{5.22}$$

where $\mathbf{h}_i = (h_{i1}, \cdots, h_{ik}) = (\sum_{\ell=1}^{n} c_{\ell 1} K(\mathbf{x}_\ell, \mathbf{x}_i), \cdots, \sum_{\ell=1}^{n} c_{\ell k} K(\mathbf{x}_\ell, \mathbf{x}_i))$. It is worth noting that if $(\alpha_{i1}, \cdots, \alpha_{ik}) = 0$ for the i th example, then $(c_{i1}, \cdots, c_{ik}) = 0$, so removing such example $(\mathbf{x}_i, \mathbf{y}_i)$ would not affect the solution at all. In two-category SVM, those data points with nonzero coefficient are called support vectors. To carry over the notion of support vectors to multicategory case, we define support vectors as examples with $\mathbf{c}_i = (c_{i1}, \cdots, c_{ik}) \neq 0$ for $i = 1, \cdots, n$. Thus, the multicategory SVM retains the sparsity of the solution in the same way as the two-category SVM.

6 Simulations

In this section, we demonstrate the effectiveness of the multicategory SVM through a couple of simulated examples. Let us consider a simple three-class example in which x lies in the unit interval $[0, 1]$. Let the conditional probabilities of each class given x be $p_1(x) = 0.97 \exp(-3x)$, $p_3(x) = \exp(-2.5(x - 1.2)^2)$, and $p_2(x) = 1 - p_1(x) - p_3(x)$. As shown in the top left panel of Figure 1, the conditional probabilities set up a situation where class 1 is likely to be observed for small x, and class 3 is more likely for large x. Inbetween interval would be a competing zone for three classes though class 2 is slightly dominant for the interval. The subsequent three panels depict the true target function $f_j(x)$, $j = 1, 2, 3$ defined in Lemma 3.1 for this example. It assumes 1 when $p_j(x)$ is maximum, and $-1/2$ otherwise, whereas the target functions under one-versus-rest schemes are $f_j(x) = sign(p_j(x) - 1/2)$. $f_2(x)$ of the one-versus-rest scheme would be relatively hard to estimate because dominance of class 2 is not strong. To compare the multicategory SVM and one-versus-rest scheme, we applied both methods to a data set of the sample size $n = 200$. The attribute x_i's come from the uniform distribution on $[0, 1]$, and given x_i, the corresponding class label y_i is randomly assigned according to the conditional probabilities $p_j(x)$, $j = 1, 2, 3$. The Gaussian kernel function, $K(s, t) = \exp\left(-\frac{1}{2\sigma^2}\|s - t\|^2\right)$ was used. The tuning parameters λ, and σ are jointly tuned to minimize GCKL (generalized comparative Kullback-Liebler) distance of the estimate $\hat{\mathbf{f}}_{\lambda,\sigma}$ from the true distribution, defined as

$$GCKL(\lambda, \sigma) = E_{true} \frac{1}{n} \sum_{i=1}^{n} L(\mathbf{Y}_i) \cdot (\hat{\mathbf{f}}_{\lambda,\sigma}(x_i) - \mathbf{Y}_i)_+ \tag{6.1}$$

$$= \quad \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k}\left(\hat{f}_j(x_i)+\frac{1}{k-1}\right)_+(1-p_j(x_i)). \qquad (6.2)$$

Note that GCKL is available only in simulation settings, and we will need a computable proxy of the GCKL for real data application. Figure 2 shows the estimated functions for both methods. We see that one-versus-rest scheme fails to recover $f_2(x) = sign(p_2(x) - 1/2)$, and results in the null learning phenomenon. That is, the estimated $f_2(x)$ is almost -1 at any x in the unit interval, meaning that it could not learn a classification rule associating the attribute x with the class distinction (class 2 vs the rest, 1 or 3). Whereas, the multicategory SVM was able to capture the relative dominance of class 2 for middle values of x. Presence of such indeterminate region would amplify the effectiveness of the proposed multicategory SVM. Over 10000 newly generated test samples, multicategory SVM has misclassification rate 0.3890, while that of the one-versus-rest approach is 0.4243.

Now, the second example is a four-class problem in 2 dimensional input space. We generate uniform random vectors $\mathbf{x}_i = (x_{i1}, x_{i2})$ on the unit square $[0,1]^2$. Then, assign class labels to each $\mathbf{x}_i$ according to the following conditional probabilities : $p_1(x) = C(x)\exp(-8[x_1^2+(x_2-0.5)^2]), p_2(x) = C(x)\exp(-8[(x_1-0.5)^2+(x_2-1)^2])$, $p_3(x) = C(x)\exp(-8[(x_1-1)^2+(x_2-0.5)^2]), p_4(x) = C(x)\exp(-8[(x_1-0.5)^2+x_2^2])$ where $C(x)$ is a normalizing function at x so that $\sum_{j=1}^4 p_j(x) = 1$. Note that four peaks of the conditional probabilities are at the middle points of the four sides of unit square, and by the symmetry the ideal classification boundaries are formed by two diagonal lines joining the opposite vertices of the unit square. We generated a data set of size $n = 300$, and the Gaussian kernel function was used again. The estimated classification boundaries derived from $\hat{f}_j(x)$ are illustrated in Figure 3 together with the ideal classification boundary induced by the Bayes rule.

7 Discussion

We have proposed a loss function deliberately tailored to target the representation of a class with the maximum conditional probability for multicategory classification problem. It is claimed that the proposed classification paradigm is a rightful extension of binary SVMs to the multicategory case. However, it suffers the common shortcoming of the approaches that consider all the classes at once. It has to solve the problem only once, but the size of the problem is bigger than that of solving a series of binary problems. See Hsu & Lin (2001) for the comparison of several methods to solve multiclass problems using SVM in terms of their performance and computational cost. To make the computation amenable to large data sets, we may borrow implementation ideas successfully exercised in binary SVMs. Studies have shown that slight modification of the problem gives fairly good approximation to a solution in binary case, and computational benefit incurred by the modification is immense for massive data. See SOR (Successive Overrelaxation) in Mangasarian & Musicant (1999), and SSVM (Smooth SVM) in Lee & Mangasarian (1999). We may also apply SMO (Sequential Minimal Optimization) in Platt (1999) to the multicategory case. Another way to make the method computationally feasible for massive datasets without modifying the problem itself would be to make use of the specific structure of the QP (quadratic programming) problem. Noting that the whole issue is approximating some sign functions by basis functions determined by kernel functions evaluated at data points, we may consider a reduction in the number of basis functions. For a large dataset, subsetting basis functions would not lead to any significant loss in accuracy, while we get a computational gain by doing so. How to ease computational burden of the multiclass approach is an ongoing research problem. In addition, as mentioned in the previous section, a data adaptive tuning procedure for the multicategory SVM is in demand, and a version of GACV (generalized approximate cross validation), which would be a computable proxy of GCKL is under development now. For the binary case, see Wahba, Lin, & Zhang (1999). Furthermore, it would be interesting to compare various tuning procedures including GACV and the k-fold crossvalidation method, which is readily available for general settings.

References

[1] Boser, B. E., Guyon, I. M., & Vapnik, V. (1992). A training algorithm for optimal margin classifiers. In *Fifth Annual Workshop on Computational Learning Theory*, Pittsburgh, 1992.

[2] Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2(2), 121-167.

[3] Crammer, K. & Singer, Y. (2000). On the learnability and design of output codes for multiclass problems. Computational Learning Theory, 35-46.

[4] Evgeniou, T., Pontil, M., & Poggio, T. (1999). A unified framework for regularization networks and support vector machines. Technical report, M.I.T. Artificial Intelligence Laboratory and Center for Biological and Computational Learning, Department of Brain and Cognitive Sciences.

[5] Friedman, J. H. (1997). Another approach to polychotomous classification. Technical report, Department of Statistics, Stanford University.

[6] Hsu, C.-W. & Lin, C.-J. (2001). A comparison of methods for multi-class support vector machines. To appear in IEEE Transactions on Neural Networks.

[7] Kimeldorf, G. & Wahba, G. (1971). Some results on Tchebycheffian spline functions. J. Math. Analysis Appl., 33, 82-95.

[8] Lee, Y.-J. & Mangasarian, O. L. (1999). SSVM: A Smooth Support Vector Machine for classification. Data Mining Institute Technical Report 99-03. Computational Optimization and Applications, 2000 to appear.

[9] Lin, Y. (1999). Support vector machines and the Bayes rule in classification. Technical Report 1014. Department of Statistics, University of Wisconsin, Madison. Submitted.

[10] Lin, Y., Lee, Y., & Wahba, G. (2000). Support vector machines for classification in nonstandard situations. Technical Report 1016. Department of Statistics, University of Wisconsin, Madison. Submitted.

[11] Mangasarian, O. L. & Musicant, D. (1999). Successive Overrelaxation for Support Vector Machines. Mathematical Programming Technical Report 98-18. IEEE Transactions on Neural Networks, 10, 1999, 1032-1037.

[12] Platt, J. (1999). Sequential minimal optimization: A fast algorithm for training support vector machines. In B. Schölkopf, C.J.C. Burges, and A.J. Smola, editors, Advances in Kernel Methods - Support Vector Learning, 185-208, 1999.

[13] Vapnik, V. (1998). Statistical learning theory. Wiley, New York.

[14] Wahba, G. (1990). Spline Models for Observational Data. Philadelphia, PA: Society for Industrial and Applied Mathematics.

[15] Wahba, G. (1998). Support vector machines, reproducing kernel Hilbert spaces, and randomized gacv, In B. Schölkopf, C.J.C. Burges, and A.J. Smola, editors, Advances in Kernel Methods - Support Vector Learning, MIT Press, chapter 6, pp. 69–87.

[16] Wahba, G., Lin, Y., & Zhang, H. (1999). *GACV* for support vector machines, or, another way to look at margin-like quantities. Technical Report 1006. Department of Statistics, University of Wisconsin, Madison. To appear in A. J. Smola, P. Bartlett, B. Scholkopf & D. Schurmans (Eds.), Advances in Large Margin Classifiers. Cambridge, MA & London, England: MIT Press.

[17] Weston, J. & Watkins, C. (1998). Multi-class support vector machines. Technical Report CSD-TR-98-04, Royal Holloway, University of London.

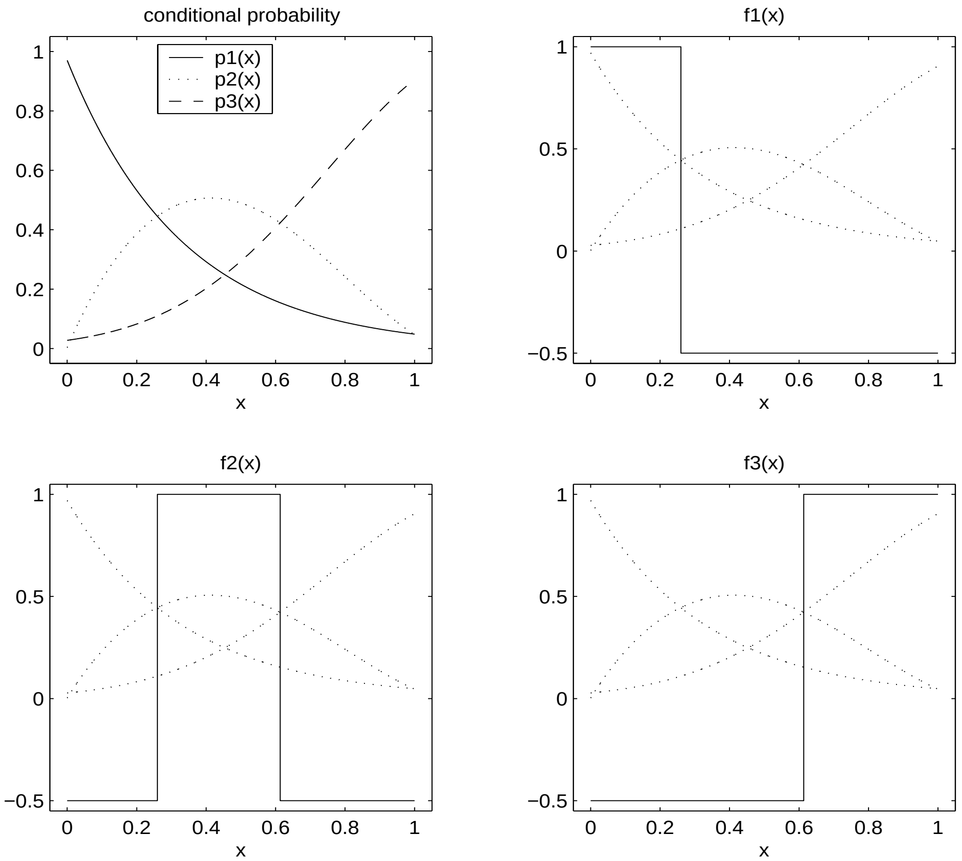

Figure 1: Conditional probabilities and multicategory SVM target functions for three-class example.

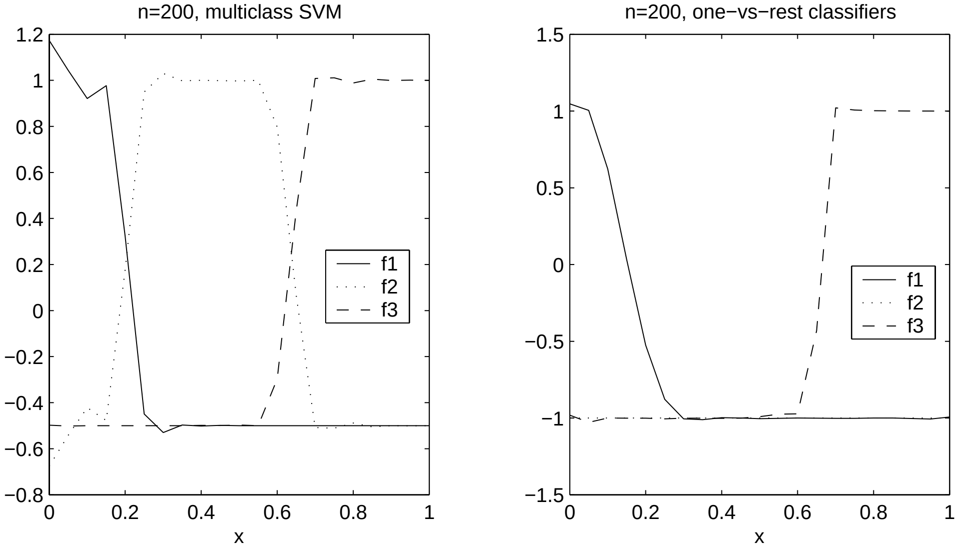

Figure 2: Comparison between multicategory SVM and one-versus-rest method. Gaussian kernel function is used, and the tuning parameters λ, and σ are simultaneously chosen via GCKL.

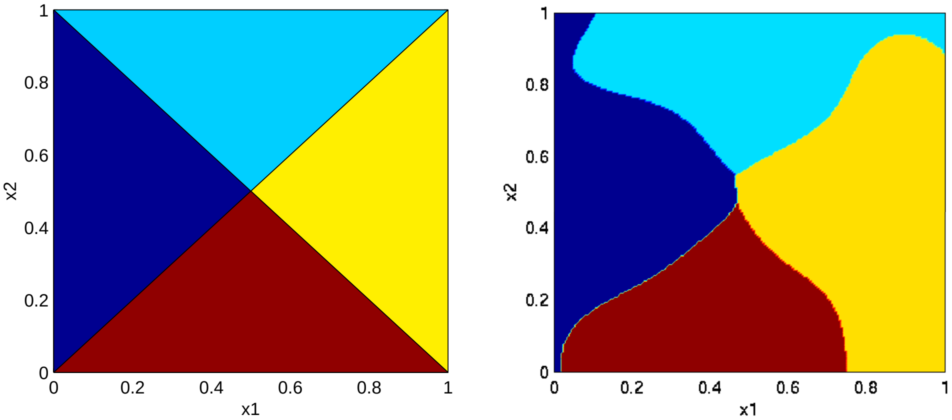

Figure 3: The classification boundaries determined by the Bayes rule (left) and the estimated classification boundaries by the multicategory SVM (right).