Statistica Sinica **16**(2006), 659-674

VARIABLE SELECTION FOR SUPPORT VECTOR MACHINES VIA SMOOTHING SPLINE ANOVA

Hao Helen Zhang

North Carolina State University

Abstract: It is well-known that the support vector machine paradigm is equivalent to solving a regularization problem in a reproducing kernel Hilbert space. The squared norm penalty in the standard support vector machine controls the smoothness of the classification function. We propose, under the framework of smoothing spline ANOVA models, a new type of regularization to conduct simultaneous classification and variable selection in the SVM. The penalty functional used is the sum of functional component norms, which automatically applies soft-thresholding operations to functional components, hence yields sparse solutions. We suggest an efficient algorithm to solve the proposed optimization problem by iteratively solving quadratic and linear programming problems. Numerical studies, on both simulated data and real datasets, show that the modified support vector machine gives very competitive performances compared to other popular classification algorithms, in terms of both classification accuracy and variable selection.

Key words and phrases: Classification, L_1 penalty, smoothing spline ANOVA, sparsity, support vector machine.

1. Introduction

In classification problems, we are given a training data set of n examples from two or more populations. For each example i, $i = 1, \ldots, n$, in the training set, we observe an input vector $\mathbf{x}_i \in \mathbb{R}^d$, and a label y_i indicating one of the classes to which the example belongs. The binary classification problem is considered in this paper, and two classes are the positive class (labeled as $+1$) and the negative class (labeled as -1). Support vector machine (SVM) classifiers developed by Boser, Guyon, and Vapnik (1992) and Vapnik (1995) have gained popularity due to promising performance in real-world applications such as text categorization, image recognition, gene expression array data analysis, etc. However, the standard SVM decision rule utilizes all the input variables, which is not desirable when some variables are not relevant or have too much noise. Hastie, Tibshirani, and Friedman (2001) demonstrated that the standard SVM can suffer from the presence of irrelevant variables. Appropriate variable selection is therefore needed to obtain a compact classifier with improved accuracy.

Several methods have been proposed for conducting variable selection in the SVM. In the linear SVM setting, Bradley and Mangasarian (1998) suggested the

1-norm SVM which imposes an absolute value penalty on the coefficients, hence produces a sparse directional vector for the separating plane. Recently Zhu, Rosset, Hastie and Tibshirani (2003) studied the solution property of the 1-norm SVM and suggested an algorithm to find the whole solution path over a range of tuning parameters. Fung and Mangasarian (2004) developed a fast Newton algorithm to solve the dual problem of the 1-norm SVM. Another class of methods are the kernel scaling methods proposed by Weston, Mukherjee, Chapelle, Pontil, Poggio and Vapnik (2000) and Grandvalet and Canu (2002). A special issue on variable and feature selection, published by *Journal of Machine Learning Research* in 2003, introduced other approaches like Bi, Bennett, Embrechts, Breneman and Song (2003) and Rakotomamonjy (2003). Recently Bach, Lanckriet and Jordan (2004) considered the block 1-norm regularization for learning a sparse conic combination of kernels. Their formulation can also be used for variable selection in nonparametric setting.

Apart from the methods above, we formulate the SVM as a regularization problem in the reproducing kernel Hilbert space (RKHS), with a novel form of the penalty functional. The optimization problem consists of two parts: the data fit is represented by the hinge loss function functional; the regularization penalty is defined as the sum of function component norms. In the Gaussian regression context, this penalty was proposed and studied by Lin and Zhang (2002), and named the component smoothing and selection operator (COSSO). Following their terminology, we refer to our method as the COSSO SVM. We show that the COSSO SVM inherits the desired properties of the SVM and, more importantly, it conducts variable selection and classification simultaneously. For the linear classification, the COSSO SVM reduces to the 1-norm SVM.

This paper is organized as follows. Section 2 gives an overview of the SVM regularization problem and the smoothing spline ANOVA framework for multivariate function estimation. Section 3 proposes the COSSO SVM method and studies the existence and finite representation of the optimal classifier. In Section 4, we give an iterative algorithm to solve the COSSO SVM, and show that only quadratic and linear programming problems are needed for implementation. We also discuss the issue of parameter tuning in this section. Simulation results and examples are presented in Section 5. The final discussion is given in Section 6. The proofs of the theorems are in the Appendix.

2. SVM and Smoothing Functional ANOVA

2.1. Regularization problem

In supervised learning problems, our task is to learn a classification rule $f : \mathbb{R}^d \to \{+1, -1\}$ from the training set, so that we can assign a class label to any new subject observed in the future. In the statistical framework, the training data $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, n$, are generally assumed to be independent realizations of the random pair $(\mathbf{X}, Y)$ that has a joint distribution $P(\mathbf{X}, Y)$. Define $p(\mathbf{x}) = \text{Prob}(Y = +1|\mathbf{X} = \mathbf{x})$. When the 0-1 loss is used, the optimal rule minimizing the expected loss is $\text{sign}[p(\mathbf{x}) - 1/2]$. This is known as the Bayes rule.

The linear SVM is a large margin classifier which separates two classes by maximizing the margin between them. When a linear separation is not plausible, the nonlinear SVM maps the data into a high dimensional feature space and then implements the linear classification. It is well-known that the nonlinear SVM can be cast as a regularization problem in a reproducing kernel Hilbert space (RKHS). Let $\mathcal{H}$ be the RKHS associated with some reproducing kernel, and $||\cdot||$ be the functional norm of any element in $\mathcal{H}$. The standard SVM amounts to solving the regularization problem

$$\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} [1 - y_i f(\mathbf{x}_i)]_+ + \lambda ||f||^2. \tag{2.1}$$

Here $[\tau]_+ = \tau$, for $\tau > 0$; $= 0$, otherwise. The hinge loss function $[1 - yf]_+$ is a convex upper bound on the misclassification rate and λ is the tuning parameter that can be adaptively chosen by the data. Once the solution $\hat{f}$ is obtained, the classification rule is $\text{sign}[\hat{f}(\mathbf{x})]$. See Wahba (1999) and Evgeniou, Pontil, and Poggio (1999) for more details. Lin (2002) showed that, if the reproducing kernel Hilbert space is rich enough (for example, associated with the Gaussian kernel or the spline kernel), the solution to (2.1) approaches the Bayes rule $\text{sign}[p(\mathbf{x}) - 1/2]$ when $n \to \infty$. This result provides a theoretical justification for the superior performances of the nonlinear SVM.

2.2. Smoothing spline ANOVA

The smoothing spline analysis of variance (SS-ANOVA) models provide a general framework for high dimensional function estimation, as shown in Wahba (1990) and Gu (2002). In the SS-ANOVA, any function $f(\mathbf{x}) = f(x^{(1)}, \ldots, x^{(d)})$ on a product domain $\mathcal{X}$ has a functional ANOVA decomposition

$$f(\mathbf{x}) = b + \sum_{j=1}^{d} f_j(x^{(j)}) + \sum_{j<k} f_{jk}(x^{(j)}, x^{(k)}) + \text{all higher-order interactions}, \tag{2.2}$$

where b is constant, f_j's are the main effects, and f_{jk}'s are the two-factor interactions. Each main effect $f_j, j = 1, \cdots, d$, is estimated in a reproducing kernel Hilbert space, denoted by $\mathcal{H}_j = [1] \oplus \bar{\mathcal{H}}_j$. The entire model space $\mathcal{H}$ is then the tensor product space $\otimes_{j=1}^{d} \mathcal{H}_j$ which admits the following tensor sum decomposition

$$\otimes_{j=1}^{d} \mathcal{H}_j = [1] \oplus \sum_{j=1}^{d} \bar{\mathcal{H}}_j \oplus \sum_{j<k} [\bar{\mathcal{H}}_j \otimes \bar{\mathcal{H}}_k] \oplus \cdots . \tag{2.3}$$

The space $\otimes_{j=1}^{d}\mathcal{H}_j$ is also a RKHS, and its reproducing kernel is the sum of the reproducing kernels of those component spaces. Each functional component in (2.2) lies in a subspace in the orthogonal decomposition (2.3) of $\otimes_{j=1}^{d}\mathcal{H}_j$. The identifiability of the components is assured by side conditions through averaging operators. Without loss of generality, we assume $\mathcal{X} = [0,1]^d$. A typical choice of $\mathcal{H}_j$ is the ℓ-th order Sobolev Hilbert space: $W_\ell[0,1] = \{g : g, g', \ldots, g^{(\ell-1)}$ are absolutely continuous, $g^{(\ell)} \in \mathcal{L}_2[0,1]\}$. In particular, the space $W_2[0,1]$ is an RKHS when equipped with the norm

$$\|g\|^2 = \Big[\int_0^1 g(t)dt\Big]^2 + \Big[\int_0^1 g'(t)dt\Big]^2 + \int_0^1 [g''(t)]^2 dt, \quad \forall g \in W_2[0,1].$$

The reproducing kernel is $1 + R(s,t)$, where

$$R(s,t) = k_1(s)k_1(t) + k_2(s)k_2(t) - k_4(|s-t|), \tag{2.4}$$

$k_1(t) = t - 1/2$, $k_2(t) = \{k_1^2(t) - 1/12\}/2$, and $k_4(t) = \{k_1^4(t) - k_1^2(t)/2 + 7/240\}/24$.

In the applications, usually only low order interaction terms in the decomposition (2.2) are retained for easy computation and interpretation. Correspondingly, the space $\otimes_{j=1}^{d}\mathcal{H}_j$ represented in (2.3) is truncated to some proper subspace $\mathcal{F}$. We write $\mathcal{F}$ as

$$\mathcal{F} = \{1\} \oplus_{\alpha=1}^{q} \mathcal{F}^\alpha, \tag{2.5}$$

where $\mathcal{F}^1, \ldots, \mathcal{F}^q$ are q orthogonal subspaces of $\otimes_{j=1}^{d}\mathcal{H}_j$. The space $\mathcal{F}$ is an RKHS with the induced norm $\|\cdot\|$.

3. The COSSO SVM

3.1. Formulation

We propose a new type of regularization for the SVM, in the framework of smoothing spline ANOVA, by solving

$$\min_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}[1 - y_i f(\mathbf{x}_i)]_+ + \tau^2 \sum_{\alpha=1}^{q} \|P^\alpha f\|, \tag{3.1}$$

where $P^\alpha f$ is the projection of f onto the subspace $\mathcal{F}^\alpha$. The parameter τ is a tuning parameter which should be properly chosen, and we will discuss the tuning issue in Section 4. The penalty $\sum_{\alpha=1}^{q}\|P^\alpha f\|$ is a sum of RKHS component norms, different from the squared RKHS norm penalty used in the standard SVM. Lin and Zhang (2002) proposed and studied this type of regularization in the penalized likelihood regression setting. Two special cases of (3.1) will be considered in this paper. Assume $\mathcal{F} = \{1\} \oplus_{j=1}^{d} \bar{\mathcal{H}}_j$, then we have the additive model:

$$\min_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}[1 - y_i f(\mathbf{x}_i)]_+ + \tau^2 \sum_{j=1}^{d} \|f_j\|, \quad \text{where} \quad f(\mathbf{x}) = b + \sum_{j=1}^{d} f_j(x^{(j)}).$$

Each $\bar{\mathcal{H}}_j$ is the Sobolev space $W_2[0,1]$ associated with the reproducing kernel R given in (2.4). In this case, the selection of main effect components is equivalent to variable selection. The other important case is the two-way interaction model which includes all the main effects and the second-order interaction effects:

$$\min_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}[1 - y_i f(\mathbf{x}_i)]_+ + \tau^2 \Big[\sum_{j=1}^{d} \|f_j\| + \sum_{j<k} \|f_{jk}\|\Big],$$

where $f(\mathbf{x}) = b + \sum_{j=1}^{d} f_j(x^{(j)}) + \sum_{j<k} f_{jk}(x^{(j)}, x^{(k)})$. We have $q = d(d+1)/2$. The choice of the additive or two-way interaction form mainly depends on the nature of data. When additive models are not adequate, two-way or higher order interaction models should be considered.

In the context of linear classification, the COSSO SVM actually reduces to the 1-norm SVM suggested by Bradley and Mangasarian (1998). The argument is as follows. For the input space $\mathcal{X} = [0,1]^d$, the linear SVM has the separating hyperplane $f(x) = b + \sum_{j=1}^{d} w_j x^{(j)}$. Consider the linear function space $\mathcal{F} = \{1\} \oplus \{x^{(1)} - 1/2\} \oplus \cdots \oplus \{x^{(d)} - 1/2\}$ with the usual L_2 inner product on $\mathcal{F}$: $(f, g) = \int_{\mathcal{X}} fg$. For each f_j, its RKHS norm penalty becomes $\|f_j\| = (12)^{-1/2}|w_j|$, which is equivalent to the absolute value penalty used in the 1-norm SVM.

Define the penalty functional $J(f) = \sum_{\alpha=1}^{q} \|P^{\alpha} f\|$. It is straightforward to show that $J(f)$ is convex in f and defines a pseudo-norm in the space $\mathcal{F}$. The following theorem guarantees the existence of the solution to (3.1).

Theorem 3.1. *Let $\mathcal{F}$ be a reproducing kernel Hilbert space of functions over an input space $\mathcal{X}$. Assume that $\mathcal{F}$ can be decomposed as in* (2.5). *Then there exists a minimizer of* (3.1) *in $\mathcal{F}$.*

Though the model space $\mathcal{F}$ is infinite dimensional, the solution to (3.1) can be shown to lie in a finite dimensional subspace of $\mathcal{F}$. This is an important property also satisfied by the standard smoothing spline. We demonstrate later that the finite representation of the solution makes it feasible to implement the COSSO SVM in practice. Here is the representer theorem, its proof is similar to that of the smoothing spline (Kimeldorf and Wahba (1971)).

Theorem 3.2. *Let $\hat{f} = \hat{b} + \sum_{\alpha=1}^{q} \hat{f}_\alpha$ be the minimizer of* (3.1) *with $\hat{f}_\alpha \in \mathcal{F}^\alpha$. Then $\hat{f}_\alpha \in span\{R_\alpha(\mathbf{x}_i, \cdot), i = 1, \ldots, n\}$, where $R_\alpha(\cdot,\cdot)$ is the reproducing kernel of $\mathcal{F}^\alpha$.*

3.2. Equivalent optimization problem

In this section we derive an equivalent formulation of (3.1) which naturally leads to an iterative algorithm. We introduce a new vector $\boldsymbol{\theta} = (\theta_1, \dots, \theta_q)^{\mathrm{T}}$ and consider the optimization problem

$$\min_{f\in\mathcal{F},\boldsymbol{\theta}} \frac{1}{n}\sum_{i=1}^{n}[1-y_i f(\mathbf{x}_i)]_+ + \lambda_0 \sum_{\alpha=1}^{q}\theta_\alpha^{-1}\|P^\alpha f\|^2 + \lambda \sum_{\alpha=1}^{q}\theta_\alpha, \qquad \text{subject to} \quad \theta_\alpha \ge 0, \quad \alpha = 1, \dots, q. \tag{3.2}$$

Note there is only one tuning parameter τ in (3.1) while there are two parameters (λ_0, λ) in (3.2). In fact, $\lambda > 0$ is the real tuning parameter while $\lambda_0 > 0$ is a constant that can be fixed at any positive value. In Section 4, we will explain that the redundancy of λ_0 is intentional for computational convenience. We have the following theorem.

Theorem 3.3. *Set* $\lambda = \tau^4/(4\lambda_0)$. (i) *If* $\hat{f}$ *minimizes* (3.1), *setting* $\hat{\theta}_\alpha = \lambda_0^{1/2}\lambda^{-1/2}\|P^\alpha \hat{f}\|$ *for* $\alpha = 1, \dots, q$, *then the pair* $(\hat{\boldsymbol{\theta}}, \hat{f})$ *minimizes* (3.2). (ii) *On the other hand, if a pair* $(\hat{\boldsymbol{\theta}}, \hat{f})$ *minimizes* (3.2), *then* $\hat{f}$ *minimizes* (3.1).

Theorem 3.3 states that, with proper choice of parameters, solving (3.1) and solving (3.2) always give the same optimal classifier $\hat{f}$. In practice, we choose to solve (3.2) since its objective function can be easily handled by standard quadratic programming and linear programming techniques.

The non-negative θ_α's can be regarded as scaling parameters and they are interpretable for the purpose of variable selection. Using the standard smoothing spline results, we can show that the solution to (3.2) has the following form

$$\hat{f}(\mathbf{x}) = b + \sum_{\alpha=1}^{q}\hat{\theta}_\alpha \sum_{i=1}^{n}\hat{c}_i R_\alpha(\mathbf{x}_i, \mathbf{x}) = b + \sum_{\alpha=1}^{q}\hat{\theta}_\alpha \hat{f}_\alpha(\mathbf{x}). \tag{3.3}$$

The expression in (3.3) suggests that whether $\hat{\theta}_\alpha = 0$ or not directly determines the appearance of the αth component of the classification function. For the additive model, if $\hat{\theta}_j = 0$, the minimizer is then taken to satisfy $\|\hat{f}_j\| = 0$, implying that the variable X_j is not selected. In this paper we use the convention 0/0=0.

4. Algorithm

Given any $\boldsymbol{\theta}$, we note that solving (3.2) is equivalent to solving the standard SVM problem (2.1) with the reproducing kernel $R_\theta = \sum_{\alpha=1}^{q}\theta_\alpha R_\alpha$, where R_α is the reproducing kernel of $\mathcal{F}^\alpha$. Lin, Wahba, Zhang and Lee (2002) showed that the solution f is given by

$$f(\mathbf{x}) = b + \sum_{i=1}^{n} c_i R_\theta(\mathbf{x}_i, \mathbf{x}).$$

Define $\mathbf{c} = (c_1, \dots, c_n)^{\mathrm{T}}$, $\mathbf{f} = (f(\mathbf{x}_1), \dots, f(\mathbf{x}_n))^{\mathrm{T}}$, $\mathbf{y} = (y_1, \dots, y_n)^{\mathrm{T}}$, and the diagonal matrix $Y = \mathrm{diag}[y_1, \dots, y_n]$. Let $\mathbf{1}_n$ and $\mathbf{0}_n$ be the column vector of ones and zeros with length n, and I_n be the identity matrix of dimension n. With some abuse of notations, we also use R_α for the $n \times n$ matrix $\{R_\alpha(\mathbf{x}_i, \mathbf{x}_j)\}$, $i = 1, \dots, n$, $j = 1, \dots, n$, and use R_θ for the matrix $\sum_{\alpha=1}^{q} \theta_\alpha R_\alpha$. Then we have $\mathbf{f} = R_\theta \mathbf{c} + b\mathbf{1}_n$, and (3.2) becomes

$$\min_{\boldsymbol{\theta} > \mathbf{0}, b, \mathbf{c}} \frac{1}{n} \sum_{i=1}^{n} [1 - y_i f(\mathbf{x}_i)]_+ + \lambda_0 \sum_{\alpha=1}^{q} \theta_\alpha \mathbf{c}^{\mathrm{T}} R_\alpha \mathbf{c} + \lambda \sum_{\alpha=1}^{q} \theta_\alpha. \tag{4.1}$$

4.1. Quadratic and linear programming

It is possible to minimize the objective function in (4.1) with respect to $\boldsymbol{\theta}$ and $(b, \boldsymbol{c})$ simultaneously. We propose alternatively solving $(b, \mathbf{c})$ with $\boldsymbol{\theta}$ fixed and solving $\boldsymbol{\theta}$ with $(b, \mathbf{c})$ fixed, since both problems can be easily solved using standard software.

When $\boldsymbol{\theta}$ is fixed, we need to solve the SVM associated with the kernel R_θ:

$$\min_{b, \mathbf{c}} \frac{1}{n} \sum_{i=1}^{n} [1 - y_i f(\mathbf{x}_i)]_+ + \lambda_0 \mathbf{c}^{\mathrm{T}} R_\theta \mathbf{c}. \tag{4.2}$$

Typically the dual problem of (4.2) is solved using the quadratic programming problem (QP). Introducing the dual variables $\boldsymbol{a} = (a_1, \dots, a_n)^{\mathrm{T}}$ and the matrix $H = (2n\lambda_0)^{-1} Y R_\theta Y$, the dual problem is

$$\max L = -\frac{1}{2} \boldsymbol{a}^{\mathrm{T}} H \boldsymbol{a}, \quad \text{subject to} \quad \mathbf{y}^{\mathrm{T}} \boldsymbol{a} = 0, \quad \mathbf{0}_n \leq \boldsymbol{a} \leq \mathbf{1}_n.$$

From the dual solution, we compute $\mathbf{c} = (2n\lambda_0)^{-1} Y \boldsymbol{a}$. Define $A = \mathrm{diag}[a_1, \dots, a_n]$. The constant b is derived using the Karash-Kuhn-Tucker optimality condition as $b = [\mathbf{1}_n^{\mathrm{T}} A (I_n - A)(\mathbf{y} - R_\theta \mathbf{c})] / [\boldsymbol{a}^{\mathrm{T}} (\mathbf{1}_n - \boldsymbol{a})]$,

When $(b, \mathbf{c})$ is fixed, we need to solve the linear programming (LP) under linear inequality constraints. Let $\mathbf{g}_\alpha = R_\alpha \mathbf{c}$ and G be the matrix with αth column $\mathbf{g}_\alpha$, $\alpha = 1, \dots, q$. The objective function in (4.1) becomes $n^{-1} \sum_{i=1}^{n} \big[1 - y_i f(\mathbf{x}_i)\big]_+ + \lambda_0 \mathbf{c}^{\mathrm{T}} G \boldsymbol{\theta} + \lambda \boldsymbol{\theta}^{\mathrm{T}} \mathbf{1}_q$. Using slack variables $\mathbf{z} = (z_1, \dots, z_n)^{\mathrm{T}}$, the hinge loss function can be replaced by $\mathbf{z}^{\mathrm{T}} \mathbf{1}_n$ plus the constraints $\mathbf{z} \geq \mathbf{0}_n$ and $\mathbf{z} \geq \mathbf{1} - Y(G\boldsymbol{\theta} + b\mathbf{1}_n)$. In addition, the term $\lambda \boldsymbol{\theta}^{\mathrm{T}} \mathbf{1}_q$ can be changed into the constraint

$\boldsymbol{\theta}^{\mathrm{T}}\mathbf{1}_q \leq M$. The parameter M is a tuning parameter which replaces λ, and there is one-to-one corresponding relationship between them. The final optimization problem is

$$\begin{aligned}&\min_{\mathbf{z},\boldsymbol{\theta}} \quad \frac{1}{n}\mathbf{z}^{\mathrm{T}}\mathbf{1}_n + \lambda_0 \mathbf{c}^{\mathrm{T}} G\boldsymbol{\theta} \\ &\text{subject to } \mathbf{z} + \boldsymbol{\theta}^{\mathrm{T}}(YG) \geq (I - Yb)\mathbf{1}_n, \quad \mathbf{z} \geq \mathbf{0}_n, \quad \boldsymbol{\theta}^{\mathrm{T}}\mathbf{1}_q \leq M, \quad \boldsymbol{\theta} \geq \mathbf{0}_q.\end{aligned} \tag{4.3}$$

This is a linear optimization problem with polyhedral constraints. Popular algorithms for solving the LP include the simplex method and the interior-point method. Many optimization packages are in wide use as well, such as CPLEX, MATLAB, GAMS, and MINOS. In our implementations, we used the MATLAB optimization toolbox. The following algorithm is proposed to solve the COSSO SVM.

1. Initialization: $\boldsymbol{\theta} = \mathbf{1}_q$.
2. With $\boldsymbol{\theta}$ fixed at current values, solve the dual problem of (4.2) for $(b, \mathbf{c})$.
3. With $(b, \mathbf{c})$ fixed at current values, solve (4.3) for $\boldsymbol{\theta}$.
4. With the new $\boldsymbol{\theta}$, go to Step 2 until convergence.

When R_θ is strictly positive definite, this algorithm is guaranteed to converge because the objective function in (4.1) is bounded below by zero, and each iteration between Step 2 and Step 3 results in improved updates. It is well-known that the LP is solvable in polynomial time, the algorithm given by Anstreicher (1999) has the computational complexity $O(n^3/\log(n)L)$ where L is the bit length of input variables for example. Since the complexity of the QP is $O(n^3)$, the overall computation for one iteration is cubic in time. Numerical studies showed that one-step update was often sufficient to give good approximate solutions.

4.2. Parameter tuning

Smoothing parameters balance the tradeoff between the hinge loss fit and the penalty on the functional components. The choice of parameters is typically done by minimizing either an estimate of generalization error, or other related performance measure. We consider minimizing the generalized comparative Kullback-Liebler (GCKL) distance proposed in Wahba (1999). Given a fitted classifier f_λ, the GCKL is defined as:

$$\begin{aligned}GCKL(\lambda) &= E_p\Big[\frac{1}{n}\sum_{i=1}^{n}(1 - Y_i f_{\lambda i})_+\Big] \\ &= \frac{1}{n}\sum_{i=1}^{n}\big[p_i(1 - f_{\lambda i})_+ + (1 - p_i)(1 + f_{\lambda i})_+\big].\end{aligned} \tag{4.4}$$

Here f_λ is fixed and the expectation is taken over the true conditional probability $p(\mathbf{x}) = P(Y = +1|\mathbf{x} = \mathbf{x})$. The GCKL can be seen as an upper bound on the misclassification rate. Since the GCKL depends on the underlying distribution $P(\mathbf{X}, Y)$, it is only computable in simulations. For real data with unknown $p(\mathbf{x})$, the leave-out-one cross validation proxy of GCKL, $1/n \sum_{i=1}^{n}[1 - y_i f_\lambda^{[-i]}(\mathbf{x}_i)]_+$, can be used as a tuning criterion. Here $f_\lambda^{[-i]}$ is the solution with the ith data point deleted. In practice, we suggest using the five-fold cross validation (CV) estimate of the GCKL. The training set is randomly split into five subsets of approximately equal sizes. Then one subset is left out, and the COSSO SVM is fitted using the other four subsets and the hinge loss is evaluated on the left-out subset. This procedure is repeated five times in this fashion with each subset being left out once. For the COSSO SVM, we need to tune λ, or equivalently M in (4.3). The parameter λ_0 seems redundant in addition to M, however the proper choice of λ_0 in (4.2) helps to solve the SVM in Step 2 more stably. In practice, we suggest tuning λ_0 once when Step 2 is executed for the first time, and fixing it thereafter.

5. Numerical Examples

In this section we study the empirical performances of the COSSO SVM through simulated examples and some data sets. The fitted classifier is evaluated in its classification and variable selection performances. We simulate the examples for both the additive model and two-way interaction model. In each experimental setting, we generated 100 data sets for fitting and tuning, and an extra test set of $10,000$ points to compute the expected misclassification rate (EMR) of the classifier. We summarized the average EMR, model size, the frequency of each variable appearing in the COSSO SVM classifier over the 100 runs.

5.1. Example 1: additive model

Consider an additive model with $\mathcal{X} = [0,1]^{10}$. We generated $\mathbf{X}$ uniformly, and the binary response Y with the conditional logit function

$$f(\mathbf{x}) = 3x^{(1)} + \pi \sin(\pi x^{(2)}) + 8(x^{(3)})^5 + \frac{2}{e-1}e^{x^{(4)}} - 6.$$

Therefore $X^{(5)}, \ldots, X^{(10)}$ are uninformative in this example. The tuning parameter M was tuned by the GCKL. The Bayes misclassification rate was 0.216, which is the optimal rate based on the true p. Consider the two settings $n = 250$ and $n = 500$. In Table 1 we report the average EMR and model size of the additive COSSO SVM in 100 runs. The values in the parentheses are the standard errors of the corresponding mean values. Table 2 shows the appearance frequency of each variable in the final model. When $n = 250$, the COSSO SVM always selected $X^{(1)}$, $X^{(2)}, X^{(3)}$, and selected $X^{(4)}$ in 98 runs. Sometimes some of uninformative variables were selected as well. When the sample size increased to 500, the COSSO SVM never missed any important variable, and it selected unimportant variables with a much lower frequency. The left plot in Figure 5.1 shows that the COSSO SVM gave the correct true model in 74 of 100 runs.

Table 1. The average EMR and model size of the additive COSSO SVM classifier.

n	EMR	Model Size
250	0.234 (0.009)	5.07
500	0.225 (0.006)	4.46

Table 2. The appearance frequency of the input variables in the COSSO SVM classifiers.

n	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
250	100	100	100	98	14	18	18	17	23	19
500	100	100	100	100	9	12	10	8	2	5

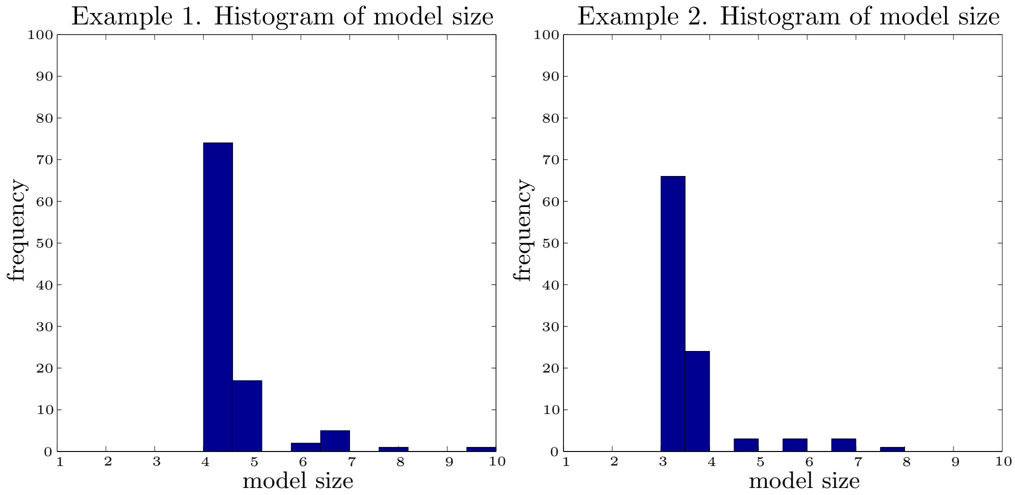

Figure 5.1. The histogram of the model size given by the COSSO SVM in the 100 runs when $n = 500$. The plot on the left is for the additive model, and the plot on the right is for the two-way interaction model.

5.2. Example 2. two-way interaction model

In this example, we generated four input variables independently from Unif$[0, 1]$. The true logit function contains the important main effects $X^{(1)}, X^{(2)}$

and their interaction:

$$f(\mathbf{x}) = 4x^{(1)} + \pi \sin\left(\pi x^{(1)}\right) + 6x^{(2)} - 8\left(x^{(2)}\right)^3 + 3\cos\left(2\pi(x^{(1)} - x^{(2)})\right) - 5.$$

The Bayes error is 0.155. We fit the two-way interaction model with $n = 300$ and $n = 500$. Table 3 shows that the COSSO SVM never missed any important main or interaction term in the 100 runs under each setting. When n increased from 300 to 500, the COSSO SVM selected the correct model size more precisely, as shown in Table 4. The distribution of the model size in 100 runs is depicted in Figure 5.1 (right plot), showing that the correct model was chosen by the COSSO SVM in 66 of 100 runs.

Table 3. The average EMR and model size of the two-way interaction COSSO SVM classifier.

n	EMR	Model Size
300	0.198 (0.016)	4.56 (1.73)
500	0.182 (0.010)	3.56 (1.03)

Table 4. The appearance frequency of the input variables in the COSSO SVM classifiers.

n	X_1	X_2	X_3	X_4	X_1, X_2	X_1, X_3	X_1, X_4	X_2, X_3	X_2, X_4	X_3, X_4
300	100	100	23	20	100	22	24	28	24	15
500	100	100	6	5	100	13	8	7	5	12

5.3. Data examples

Gestel, Suykens, Baesens, Viaene, Vantheienen, Dedene, Moor and Vandewalle (2004) conducted a benchmark study comparing a number of commonly used machine learning techniques including the SVM; least squares SVM (LS-SVM); linear discriminant analysis (LDA); quadratic discriminant analysis (QDA); logistic regression (Logit); the decision tree algorithm C4.5; Holte's one-rule classifier (oneR); instance based learners (IB); and the Naive Bayes method. They found the least squares SVM (LS-SVM) with the radial basis function (RBF) kernel performed best among six types of LS-SVMs. Thus we only include the LS-SVM with RBF kernel and linear kernel for our comparison. They considered two instance based learners (IB1 and IB10) and two types of Naive Bayes methods, and we only report the better performance of the two. There are five binary classification datasets with continuous predictors in their study, and we test the performance of the COSSO SVM on these datasets. The results for the other algorithms are taken from their paper.

The datasets are the BUPA Liver Disorder data, the Johns Hopkins University Ionosphere data; the PIMA Indian Diabetes; the Sonar, Mines vs. Rocks data; and the Wisconsin Breast Cancer data. The basic features of the datasets and the performances of different algorithms are summarized in Table 5. Following Gestel, Suykens, Baesens, Viaene, Vantheienen, Dedene, Moor, and Vandewalle (2004), for each dataset we randomly select 2/3 of the data for training and tuning, and test on the remaining 1/3 of the data. We do this randomization 10 times and report the average test set performance and sample standard deviation for the COSSO SVM. The best average test set performances are denoted in bold face for each dataset in Table 5. The additive COSSO SVM is fitted and its performances on these benchmark datasets are very competitive to that of the other algorithms.

Table 5. Comparison of the test set performance of the COSSO SVM with those of SVM, LS-SVM, LDA, QDA, Logit, C4.5, oneR, IB, Naive Bayes, and the Majority Rule. The results of the other algorithms are taken from the paper Gestel et al. (2004).

	BUPA	Ionosphere	Pima Indian	Sonar MR	Wisc. BC
n	345	351	768	208	683
d	6	33	8	60	9
COSSO SVM	**72.0** (5.0)	89.6 (2.6)	**77.3** (2.3)	**78.6** (2.6)	95.8 (1.2)
SVM (linear)	67.7 (2.6)	87.1 (3.4)	77.0 (2.4)	74.1 (4.2)	96.3 (1.0)
SVM (RBF)	70.4 (3.2)	95.4 (1.7)	**77.3** (2.2)	75.0 (6.6)	96.4 (1.0)
LS-SVM (linear)	65.6 (3.2)	87.9 (2.0)	76.8 (1.8)	72.6 (3.7)	95.8 (1.0)
LS-SVM (RBF)	70.2 (4.1)	**96.0** (2.1)	76.8 (1.7)	73.1 (4.2)	96.4 (1.0)
LDA	65.4 (3.2)	87.1 (2.3)	76.7 (2.0)	67.9 (4.9)	95.6 (1.1)
QDA	62.2 (3.6)	90.6 (2.2)	74.2 (3.3)	53.6 (7.4)	94.5 (0.6)
Logit	66.3 (3.1)	86.2 (3.5)	77.2 (1.8)	68.4 (5.2)	96.1 (1.0)
C4.5	63.1 (3.8)	90.6 (2.2)	73.5 (3.0)	72.1 (2.5)	94.7 (1.0)
oneR	56.3 (4.4)	83.6 (4.8)	71.3 (2.7)	62.6 (5.5)	91.8 (1.4)
IB	61.3 (6.2)	87.2 (2.8)	73.6 (2.4)	77.7 (4.4)	96.4 (1.2)
Naive Bayes	63.7 (4.5)	92.1 (2.5)	75.5 (1.7)	71.6 (3.5)	**97.1** (0.9)
Majority Rule	56.5 (3.1)	64.4 (2.9)	66.8 (2.1)	54.4 (4.7)	66.2 (2.4)

The number of interaction terms in the two-way interaction model is $d(d-1)/2$, which can be very large even for a moderate d. For example, in the Sonar, Mine, Rock data there are 60 variables and the full two-way ANOVA model has $1,770$ interaction terms. This can cause great difficulty in both model fitting and interpretation. Therefore, when d is large, additive models are often preferred, and maybe sufficient. We also fitted the two-way interaction models for three data sets having fewer than 10 variables (BUPA, Pima Indian and Wisconsin BC), and they gave similar classification performances as additive models, but took much longer to fit. For example, for the BUPA data, the average accuracy of the two-way interaction model was 73.0% and that of the additive model was 72.0%.

6. Discussion

The COSSO SVM is attractive in its compact mathematical formulation and nice solution properties. The novel regularization setup naturally combines smoothing and shrinkage-type operations on the ANOVA components of the classifier. In addition, the COSSO SVM includes the 1-norm SVM as a special case. Numerical studies demonstrate its desirable performances when compared with other classification schemes.

The proposed idea provides a general framework for variable selection in the SVM. We focus on the two-class classification in this paper, however, the idea can be used in multi-class classification problems. See the technical report of Lee, Kim, Lee, and Koo (2004). In addition, it is straightforward to generalize the COSSO SVM to nonstandard classification situations where: (i) different costs are used for different types of misclassification; (ii) the proportions of two classes in samples do not represent those in populations. Let the false positive and the false negative cost be c^+ and c^- respectively, the proportions of two classes in populations be π_0^+ and π_0^-, and those in samples be π^+ and π^-. Following Lin, Lee, and Wahba (2002), we define the weight w function on the label by $w(+1) = c^-\pi^-\pi_0^+$ and $w(-1) = c^+\pi^+\pi_0^-$. The non-standard COSSO SVM can be posed as

$$\min_f \quad \frac{1}{n}\sum_{i=1}^{n} w(y_i)[1 - y_i f(\mathbf{x}_i)]_+ + \tau^2 \sum_{\alpha=1}^{q} \|P^\alpha f\|. \tag{5.1}$$

The algorithm suggested in Section 4 can be adapted to solve this problem.

For high dimension, low sample size data, $d \gg n$, linear classifiers often give better performances than nonlinear ones (Hastie, Tibshirani, and Friedman (2001)). This fact is related to the asymptotic results in Hall and Marron (2004): when $d \to \infty$ with n fixed, the pairwise distances between any two points are asymptotically identical to each other, so the points form an n-simplex. Linear classifiers are natural choices to discriminate two simplices. In those situations, the L_1-norm SVM (Bradley and Mangasarian (1998) and Zhu, Rosset, Hastie and Tibshirani (2003)) may be sufficient for classification and variable selection.

Acknowledgement

Hao Helen Zhang's research was partially supported by National Science Foundation grant DMS-0405913.

Appendix 1. Proof of Solution Existence

Proof of Theorem 3.1. Denote the functional to be minimized in (3.1) by

$$A(f) = \sum_{i=1}^{n}[1 - y_i f(\mathbf{x}_i)]_+ + \tau^2 J(f).$$

Then $A(f)$ is convex and continuous, and

$$\sum_{\alpha=1}^{q} \|P^\alpha f\|^2 \leq J^2(f) \leq q \sum_{\alpha=1}^{q} \|P^\alpha f\|^2. \tag{A.1}$$

Without loss of generality, we assume $\tau = 1$.

Define $\mathcal{F}_1 = \oplus_{\alpha=1}^q \mathcal{F}^\alpha$. By (A.1) we have $J(f_1) \geq \|f_1\|$ for any $f_1 \in \mathcal{F}_1$. Let $R_{\mathcal{F}_1}$ be the reproducing kernel of $\mathcal{F}_1$ and $\langle \cdot, \cdot \rangle_{\mathcal{F}_1}$ be the inner product in $\mathcal{F}_1$. Write $a = \max_{i=1}^n R_{\mathcal{F}_1}^{1/2}(\mathbf{x}_i, \mathbf{x}_i)$. By the reproducing property of the kernel we have, for any $f_1 \in \mathcal{F}_1$ and $i = 1, \ldots, n$,

$$\begin{aligned} |f_1(\mathbf{x})| &= |\langle f_1(\cdot), R_{\mathcal{F}_1}(\mathbf{x}_i, \cdot)\rangle_{\mathcal{F}_1}| \leq \|f_1\| \langle R_{\mathcal{F}_1}(\mathbf{x}_i, \cdot), R_{\mathcal{F}_1}(\mathbf{x}_i, \cdot)\rangle_{\mathcal{F}_1}^{\frac{1}{2}} \\ &= \|f_1\| R_{\mathcal{F}_1}^{\frac{1}{2}}(\mathbf{x}_i, \mathbf{x}_i) \leq a\|f_1\| \leq aJ(f_1). \end{aligned} \tag{A.2}$$

Let n^+ and n^- be, respectively, the number of sample points from the $+1$ and -1 classes. Define $\rho = \min\{2n^+/n, 2n^-/n\}$. Consider the set

$$D = \{f \in \mathcal{F} : f = b + f_1, \text{with } \ b \in \{1\}, f_1 \in \mathcal{F}_1, J(f) \leq \rho, |b| \leq 1 + a\}.$$

Then D is a closed, convex, and bounded set. By Theorem 4 of Tapia and Thompson (1978, p.162), there exists a minimizer of (3.1) in D. Let the minimizer be $\bar{f}$. Direct calculation gives us

$$\sum_{i=1}^{n} [1 - y_i f(\mathbf{x}_i)]_+ = \sum_{y_i=+1} [1 - f(\mathbf{x}_i)]_+ + \sum_{y_i=-1} [1 + f(\mathbf{x}_i)]_+,$$

hence $A(+1) = 2n^-/n$ for the function $f(\mathbf{x}) \equiv +1$ and $A(-1) = 2n^+/n$ for the function $f(\mathbf{x}) \equiv -1$. Since the constant functions $+1$ and -1 are both in D, we must have $A(\bar{f}) < \min\{A(+1), A(-1)\} = \rho$.

On the other hand, for any $f \notin D$, one of the following must happen.

(i) When $J(f) > \rho$, we have $A(f) \geq J(f) > \rho$.

(ii) When $J(f) \leq \rho$, $f = b + f_1$, $f_1 \in \mathcal{F}$ and $b > 1 + a$, we use (A.2) to get that, for any $i = 1, \ldots, n$, $b + f_1(\mathbf{x}_i) \geq b - a > 1$ and

$$\sum_{y_i=+1} [1 - b_1 - f_1(\mathbf{x}_i)]_+ + \sum_{y_i=-1} [1 + b_1 + f(\mathbf{x}_i)]_+ \geq \sum_{y_i=-1} [1 + b_1 + f(\mathbf{x}_i)]_+ > 2n^-.$$

We then have $A(f) > 2n^-/n$.

(iii) When $J(f) \leq \rho$, $f = b + f_1$, $f_1 \in \mathcal{F}$ and $b < -1 - a$, we use (A.2) to get that, for any $i = 1, \ldots, n$, $b + f_1(\mathbf{x}_i) \leq b + a < -1$ and

$$\sum_{y_i=-1} [1 - b_1 - f_1(\mathbf{x}_i)]_+ + \sum_{y_i=-1} [1 + b_1 + f(\mathbf{x}_i)]_+ \geq \sum_{y_i=+1} [1 - b_1 - f(\mathbf{x}_i)]_+ > 2n^+.$$

We then have $A(f) > 2n^+/n$.

Hence for any $f \notin D$, we have $A(f) > A(\bar{f})$. Therefore $\bar{f}$ is a minimizer of (3.1) in $\mathcal{F}$.

Appendix 2. Proof of Representer Theorem

Proof of Theorem 3.2. For any $f \in \mathcal{F}$, we can write $f = b + \sum_{\alpha=1}^{q} f_\alpha$ with $f_\alpha \in \mathcal{F}^\alpha$. Let the projection of f_α onto $\text{span}\{R_\alpha(\mathbf{x}_i, \cdot), i = 1, \ldots, n\} \subset \mathcal{F}^\alpha$ be denoted by g_α, and its orthogonal complement by h_α. Then $f_\alpha = g_\alpha + h_\alpha$, and $\|f_\alpha\|^2 = \|g_\alpha\|^2 + \|h_\alpha\|^2$, $\alpha = 1, \ldots, q$. Since the reproducing kernel of $\mathcal{F}$ is $1 + \sum_{\alpha=1}^{q} R_\alpha$ is, we have

$$f(\mathbf{x}_i) = \langle 1 + \sum_{\alpha=1}^{q} R_\alpha(\mathbf{x}_i, \cdot), b + \sum_{\alpha=1}^{q}(g_\alpha + h_\alpha)\rangle = b + \sum_{\alpha=1}^{q}\langle R_\alpha(\mathbf{x}_i, \cdot), g_\alpha\rangle,$$

where $\langle\cdot,\cdot\rangle$ is the inner product in $\mathcal{F}$. Therefore (3.1) can be written as

$$\frac{1}{n}\sum_{i=1}^{n}\left[1 - y_i(b + \sum_{\alpha=1}^{q}\langle R_\alpha(\mathbf{x}_i, \cdot), g_\alpha\rangle)\right]_+ + \tau^2\sum_{\alpha=1}^{q}(\|g_\alpha\|^2 + \|h_\alpha\|^2)^{\frac{1}{2}},$$

and any minimizer f satisfies $h_\alpha = 0$, $\alpha = 1, \ldots, q$. The theorem is proved.

Proof of Theorem 3.3. Denote the functional in (3.1) by $A(f)$, and the functional in (3.2) by $N(\theta, f)$. For any $\theta_\alpha \geq 0, f \in \mathcal{F}$, we have

$$\lambda_0\theta_\alpha^{-1}\|P^\alpha f\|^2 + \lambda\theta_\alpha \geq 2\lambda_0^{1/2}\lambda^{1/2}\|P^\alpha f\| = \tau^2\|P^\alpha f\|,$$

and the equality holds if and only if $\theta_\alpha = \lambda_0^{1/2}\lambda^{-1/2}\|P^\alpha f\|$. Therefore $N(\boldsymbol{\theta}, f) \geq A(f)$ for any $\theta_\alpha \geq 0$, $\alpha = 1, \ldots, q$, and $f \in \mathcal{F}$, and the equality holds if and only if $\theta_\alpha = \lambda_0^{1/2}\lambda^{-1/2}\|P^\alpha f\|$, $\alpha = 1, \ldots, q$.

References

Anstreicher, K. M. (1999). Linear programming in O((n3/ln n)L) operations. *SIAM J. Optim.* **9**, 803-812.

Bach, F., Lanckriet, G. R. and Jordan, M. I. (2004). Multiple kernel learning, conic duality, and the SMO algorithm. *Proceeding of the twenty-first International Conference on Machine Learning.*

Bi, J., Bennett, K. P., Embrechts, M., Breneman, C. M. and Song, M. (2003). Dimensionality Reduction via Sparse Support Vector Machines. *J. Mach. Learn. Res.* **3**, 1229-1243.

Boser, B. E., Guyon, I. M. and Vapnik, V. (1992). A training algorithm for optimal margin classifiers. In *Fifth Annual ACM Workshop on Computational Learning Theory*, 144-152. ACM press, Pittsburgh, PA.

Bradley, P. S. and Mangasarian, O. L. (1998). Feature selection via concave minimization and support vector machines. *Proceeding* 13*th International Conference on Machine Learning*, 82-90.

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

Fung, G. and Mangasarian, O. L. (2004). A Feature Selection Newton Method for Support Vector Machine Classification. *Computat. Optim. Appl.* **28**, 185-202.

Gestel, T. V., Suykens, J. A. K., Baesens, B., Viaene, S., Vanthienen, J., Dedene, G., Moor, B. D. and Vandewalle, J. (2004). Benchmarking least squares support vector machine classifiers. *Machine Learning* **54**, 5-32.

Grandvalet, Y. and Canu, S. (2002). Adaptive Scaling for Feature Selection in SVMs. *Neural Inf. Process. Systems.*

Gu, C. (2002). *Smoothing Spline ANOVA Models.* Springer-Verlag, New York.

Hall, P. and Marron, J. S. (2004). Geometric representation of high dimension low sample size data. *J. Roy. Statist. Soc.* To appear.

Hastie, T., Tibshirani, R. and Friedman, J. (2001). *The Elements of Statistical Learning: data mining, inference, and prediction.* Springer, New York.

Kimeldorf, G. and Wahba, G. (1971). Some results on Tchebycheffian spline functions. *J. Math. Anal. Appl.* **33**, 82-85.

Lee, Y., Kim, Y., Lee, S. and Koo, J. (2004). Structured multicategory support vector machine with ANOVA decomposition. Technical report No. 743, Dept. of Statistics, the Ohio State University.

Lin, Y. (2002). SVM and the Bayes rule in classification. *Data Mining and Knowledge Discovery* **6**, 259-275.

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

Lin, Y., Wahba, G., Zhang, H. H. and Lee, Y. (2002). Statistical Properties and Adaptive Tuning of Support Vector Machines. *Machine Learning* **48**, 115-136.

Lin, Y. and Zhang, H. H. (2002). Component selection and smoothing in smoothing spline analysis of variance models. Technical Report No. 1072, University of Wisconsin - Madison. Submitted.

Rakotomamonjy, A. (2003). Variable selection using SVM-based Criteria. *J. Mach. Learn. Res.*, **3**, 1357-1370.

Tapia, R. and Thompson, J. (1978). *Nonparametric Probability Density Estimation.* Johns Hopkins University Press, Baltimore, MD.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory.* Springer, New York.

Wahba, G. (1990). *Spline Models for Observational Data.* **59** SIAM. CBMS-NSF Regional Conference Series in Applied Mathematics.

Wahba, G. (1999). Support vector machines, reproducing kernel Hilbert spaces and the randomized GACV. In *Advances in Kernel Methods - Support Vector Learning* (Edited by B. Scholkopt, C. Burges and A. Smola), 69-88. MIT Press.

Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T. and Vapnik, V. (2000). Feature Selection for SVMs. *Adv. Neural Inf. Process. Systems* **13**, 668-674.

Zhu, J., Rosset, S., Hastie, T. and Tibshirani, R. (2003). 1-norm support vector machines. *Neural Inf. Process. Systems* **16**.

8203 Campus Box, North Carolina State University, Raleigh, NC 27695-8203, U.S.A.

E-mail: hzhang2@stat.ncsu.edu

(Received May 2005; accepted August 2005)