Manuscript Number: 2643

Radius Margin Bounds for Support Vector Machines with the RBF Kernel

Kai-Min Chung, Wei-Chun Kao, Chia-Liang Sun, Li-Lun Wang, and Chih-Jen Lin

Department of Computer Science and

Information Engineering

National Taiwan University

Taipei 106, Taiwan

cjlin@csie.ntu.edu.tw

Abstract

An important approach for efficient support vector machine (SVM) model selection is to use differentiable bounds of the leave-one-out (loo) error. Past efforts focused on finding tight bounds of loo, for example, radius margin bounds, span bounds, etc. However, their practical viability is still not very satisfactory. In (Duan et al., 2003), it has been shown that radius margin bound gives good prediction for L2-SVM. In this paper, through the analyses why this bound performs well for L2-SVM, we show that finding a bound whose minima are in a region with small loo values may be more important than its tightness. Based on this principle we propose modified radius margin bounds for L1-SVM where the original bound is only applicable to the hard-margin case. Our modification for L1-SVM achieves comparable performance to L2-SVM. To study whether L1- or L2-SVM should be used, we further analyze other properties such as their differentiability, number of support vectors, and number of free support vectors. In this aspect, L1-SVM possesses the advantage of having fewer support vectors. Their implementations are also different so we discuss related issues in detail.

1 Introduction

Recently, support vector machines (SVM) (Vapnik, 1998) have been a promising tool for data classification. Its success depends on the tuning of several parameters which affect the generalization error.

The generalization error can be estimated by, for example, testing some data which are not used for training; cross-validation and leave-one-out (loo) estimate the error in this way. Loo is particularly of theoretical interest, because it makes use of the greatest possible data for training and does not involve random sampling. A survey on loo error and results for SVM and related algorithms can be found in (Elisseeff and Pontil, 2002). However, loo also exhibits serious limitations. In addition to the fact that it is computationally expensive, it also has a larger variance than cross-validation (Hastie et al., 2002, page 215).

A way of saving the computational cost of loo is by a bound from theoretical derivation. The goal of this paper is to make radius margin bound, a theoretical bound of loo error, a practical tool. There are some interesting results on the loo bounds of SVM and kernel machines in (Zhang, 2001).

First, we briefly describe the SVM formulation. Given training vectors $x_i \in R^n, i = 1, \ldots, l$, in two classes, and a vector $y \in R^l$ such that $y_i \in \{1, -1\}$, the SVM formulation is as follows:

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2} w^T w \\ \text{subject to} \quad & y_i(w^T \phi(x_i) + b) \geq 1, i = 1, \ldots, l, \end{aligned} \tag{1.1}$$

where training data x_i are mapped to a higher dimensional space by the function ϕ. Practically, we need only $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$, which is called the kernel function. In this paper, we focus on the RBF kernel

$$K(x_i, x_j) = e^{-\|x_i - x_j\|^2/(2\sigma^2)}.$$

The parameter σ can be determined by the loo bound described below. It was shown in (Vapnik, 1998) that the following radius margin bound holds for SVM without the bias term b

$$loo \leq 4R^2\|w\|^2, \tag{1.2}$$

where loo is the number of loo errors, w is the solution of (1.1), and R is the radius of the smallest sphere containing all $\phi(x_i)$. (Vapnik and Chapelle, 2000) further extends the bound for the general case where b is present. It has been shown (e.g. (Vapnik, 1998)) that R^2 is the objective value of the following optimization problem:

$$\begin{aligned}\max_{\beta} \quad & 1-\beta^T K\beta \\ \text{subject to} \quad & 0 \leq \beta_i, i=1,\ldots,l, \\ & e^T\beta = 1.\end{aligned} \tag{1.3}$$

Some early experiments on minimizing the right-hand side of (1.2) are in (Schölkopf et al., 1995) and (Cristianini et al., 1999).

However, (1.1) is not a form for practical use. It may not be feasible if $\phi(x_i)$ are not linearly separable. In addition, a highly nonlinear ϕ may lead to overfitting. Thus, practically we solve either

$$\min_{w,b,\xi} \frac{1}{2}w^T w + C\sum_{i=1}^{l}\xi_i, \tag{1.4}$$

or

$$\min_{w,b,\xi} \frac{1}{2}w^T w + \frac{C}{2}\sum_{i=1}^{l}\xi_i^2, \tag{1.5}$$

both subject to $y_i(w^T\phi(x_i)+b) \geq 1-\xi_i$ and $\xi_i \geq 0$, where ξ_i represents the training error and the parameter C adjusts the training error and the regularization term $w^T w/2$. We refer to the two cases as L1-SVM and L2-SVM, respectively. Note that for L2-SVM we use $C/2$ instead of C for easier analyses later. Then, if the RBF kernel is used, C and σ are the two tunable parameters. Usually they are solved through dual problems. For (1.4), its dual is

$$\begin{aligned}\max_{\alpha} \quad & e^T\alpha - \frac{1}{2}\alpha^T Q\alpha \\ \text{subject to} \quad & 0 \leq \alpha_i \leq C, i=1,\ldots,l, \\ & y^T\alpha = 0,\end{aligned} \tag{1.6}$$

where e is the vector of all ones and Q is an $l \times l$ matrix with $Q_{ij} = y_i y_j K(x_i, x_j)$. For primal and dual optimal solutions,

$$w = \sum_{i=1}^{l}\alpha_i y_i \phi(x_i) \qquad \text{and} \qquad \frac{1}{2}w^T w + C\sum_{i=1}^{l}\xi_i = e^T\alpha - \frac{1}{2}\alpha^T Q\alpha. \tag{1.7}$$

For (1.5), its dual is

$$\begin{aligned}\max_{\alpha} \quad & e^T\alpha - \frac{1}{2}\alpha^T(Q + \frac{I}{C})\alpha \\ \text{subject to} \quad & 0 \le \alpha_i, i = 1, \dots, l, \\ & y^T\alpha = 0.\end{aligned} \tag{1.8}$$

Unfortunately, for L1-SVM, the radius margin bound (1.2) does not hold as the optimization formulation is now different. However, L2-SVM can be reduced to a form of (1.1) using

$$\tilde{w} \equiv \begin{bmatrix} w \\ \sqrt{C}\xi \end{bmatrix} \text{ and the } i\text{th training data as } \begin{bmatrix} \phi(x_i) \\ \frac{y_i e_i}{\sqrt{C}} \end{bmatrix}, \tag{1.9}$$

where e_i is a zero vector of length l except the ith component is one. The kernel function becomes $\tilde{K}(x_i, x_j) = K(x_i, x_j) + \delta_{ij}/C$, where $\delta_{ij} = 1$ if $i = j$, 0 otherwise. Then, (1.2) can be directly used and so existing work on the radius margin bound mainly focus on L2-SVM (e.g. (Chapelle et al., 2002; Keerthi, 2002)). Note that R^2 is now also different so we will denote the new bound as

$$\tilde{R}^2\|\tilde{w}\|^2, \tag{1.10}$$

where $\tilde{R}^2$ is the objective value of the following problem:

$$\begin{aligned}\max_{\beta} \quad & 1 + \frac{1}{C} - \beta^T(K + \frac{I}{C})\beta \\ \text{subject to} \quad & 0 \le \beta_i, i = 1, \dots, l, \\ & e^T\beta = 1.\end{aligned} \tag{1.11}$$

(Chapelle et al., 2002) is the first to use the differentiability of (1.2) and develop optimization algorithms for finding optimal C and σ. This is faster than a two-dimensional grid search. More implementation issues for solving large problems are discussed in (Keerthi, 2002). There are several other bounds on the loo error, which may be used for model selection. (Joachims, 2000) developed the $\xi\alpha$-estimator, extending results of (Jaakkola and Haussler, 1999) to general SVMs. (Wahba et al., 2000) suggested that Generalized Approximate Cross Validation (GACV) is a reasonable estimate of the Generalized Comparative Kullback-Liebler Distance (GCKL) (Wahba, 1998). For both $\xi\alpha$ and GACV bounds, they may not be differentiable so efficient implementations have not been available. (Vapnik and Chapelle, 2000) proposed the span bound on loo, and there are also simplified approximations of the span bound. It is a very tight bound but requires more computational cost. A comparison of different methods for model selection is in (Duan et al., 2003) which shows the radius margin bound for L2-SVM performs quite well. In this paper, we will discuss only radius margin bounds.

For L1-SVM, we have no way to reduce it to a form of (1.1). (Zhang, 2001) shows that it is more difficult to obtain a quantitative loo bound for non-separable problems, which is the case for L1-SVM. Since (1.2) does not hold, some modifications are necessary. In (Duan et al., 2003), following the suggestion by Chapelle, they consider

$$D^2 e^T \alpha + \sum_{i=1}^{l} \xi_i, \tag{1.12}$$

where $D = 2R$.

However, experiments in (Duan et al., 2003) showed that comparing to other methods, this bound is not good. In addition, implementation issues such as the differentiability of (1.12) have not been addressed.

In this paper, we would like to investigate the following issues:

1. Why experiments in (Duan et al., 2003) show that the radius margin bound for L2-SVM is better than that for L1-SVM ?

   Our analysis in Section 2 shows that compared to the bound for L1-SVM, the one for L2-SVM possesses some nice properties so that minima of the radius margin bound happen in the region where the generalization error is small. We show that finding a bound whose minima are in a good region may be more important than its tightness.

2. Can we have better modified radius margin bounds for L1-SVM ?

   Based on the discussion for L2-SVM, in Section 3, we propose some modifications for L1-SVM which perform better than (1.12). However, these bounds, including (1.12), may not be differentiable so gradient-based optimization methods are not applicable. We propose some further modifications for L1-SVM bounds which are differentiable.

3. Should we use L1-SVM or L2-SVM after everything is considered ?

   In Section 4, we show that in terms of generalization as measured by testing accuracy, the proposed modification of the radius margin bound for L1-SVM is competitive with that for L2-SVM. As the number of support vectors as well as free support vectors affect computational time of gradient-based methods, we also conduct comparisons. Results indicate that L1-SVM requires fewer support vectors. However, we also show that an efficient implementation for L1-SVM is not as straightforward as that for L2-SVM. In Section 5, we discuss many implementation issues which were not studied before.

2 Radius Margin (RM) Bound for L2-SVM

In this section, we investigate why the radius margin (RM) bound performs well for L2-SVM. First, in Table 2.1 we list test accuracy given in (Duan et al., 2003) by comparing (1.12), a modified RM bound for L1-SVM, and the RM bound for L2-SVM. For each problem, we fix C (or σ^2) as they used and then search for the value of σ^2 (or C) that minimizes the bound. The (C, σ^2) is then used to train a model and predict the test data. Following them we present $\ln \sigma^2$ and $\ln C$ using the natural log. We give the description of data sets in Section 4.

Some immediate observations are as follows:

1. No matter C or σ is fixed, the RM bound for L2-SVM is better.

2. When C is fixed, except for problem `tree`, the modified RM bound for L1-SVM in (1.12) has minima at large σ. In other words, a good σ should be smaller.

3. When σ is fixed, for each problem, the modified RM bound for L1-SVM in (1.12) has the minimum at a smaller value than the best C.

Therefore, (1.12) suffers from the problem that the obtained C is too small and σ is too large. While it may not be easy to directly explain these phenomena, here, we have an alternative way of thinking: the RM bound for L2-SVM may have inherently avoided that the minimum happens at too small C or too large σ. In

C fixed

Data set	banana	image	splice	waveform	tree
Value of $\ln C$	5.2	4.0	0.4	1.4	8.6
L1 test error	0.1043(0.6)	0.0188(1.0)	0.0947(3.4)	0.1022(3.2)	0.1086(3.8)
$D^2(e^T\alpha)+\sum\xi_i$	0.5594(10.0)	0.2564(10.0)	0.1545(8.1)	0.1533(8.0)	0.1703(-2.5)
Value of $\ln C$	-0.9	0.44	6.91	0	9.80
L2 test error	0.1118(-1.4)	0.0238(0.5)	0.0947(3.3)	0.0989(2.7)	0.1049(4.6)
$\tilde{R}^2\|\tilde{w}\|^2$	0.1141(-1.6)	0.0297(-0.3)	0.0998(3.1)	0.1030(2.1)	0.1703(-2.5)

σ fixed

Data set	banana	image	splice	waveform	tree
Value of $\ln\sigma^2$	0.6	1.0	3.4	3.2	3.8
L1 test error	0.1043(5.2)	0.0188(3.9)	0.0947(0.4)	0.1022(1.4)	0.1086(8.6)
$D^2(e^T\alpha)+\sum\xi_i$	0.3843(-2.9)	0.1287(-2.9)	0.1945(-2.4)	0.1409(-2.5)	0.2609(-10.0)
Value of $\ln\sigma^2$	-1.39	-0.29	3.07	2.80	4.60
L2 test error	0.1120(-0.0)	0.0218(2.4)	0.1007(2.1)	0.0991(-0.0)	0.1049(9.7)
$\tilde{R}^2\|\tilde{w}\|^2$	0.1124(-0.9)	0.0297(0.4)	0.1016(10.0)	0.1007(-0.6)	0.1413(-1.4)

Table 2.1: Error rates with respect to the test set obtained using L1- and L2-SVM RM bounds at their minimal points. For the case of "C fixed," the value given in parentheses is the value of $\ln\sigma^2$ where the bound attained the minimum; and, for the case of "σ fixed," the value given in the parentheses is the value of $\ln C$ where the bound attained the minimum.

the following we derive two other bounds for L2-SVM and through the comparison we show that the RM bound for L2-SVM really possesses such mechanisms.

The RM bound for L2-SVM is derived using inequality (2.1) on the hard-margin SVM (see, for example, Lemma 3 of (Vapnik and Chapelle, 2000)) : If in the leave-one-out procedure a support vector x_p corresponding to a non-zero dual variable $\alpha_p > 0$ is recognized incorrectly, then

$$\frac{1}{2\max_{i\neq j}\|\tilde{z}_i-\tilde{z}_j\|^2} \quad \leq \quad \frac{\alpha_p^2}{2}\min_{\tilde{z}\in\Lambda_p^*}\|\tilde{z}_p-\tilde{z}\|^2, \tag{2.1}$$

where

$$\tilde{z}_i \equiv [\phi(x_i)^T, e_i^T/\sqrt{C}]^T. \tag{2.2}$$

(2.2) changes L2-SVM to a hard-margin SVM formulation and Λ_p^* is a subset of the convex hull by $\{\tilde{z}_1,\ldots,\tilde{z}_l\}\backslash\{\tilde{z}_p\}$. Notice that in (Vapnik and Chapelle, 2000), the left-hand side of (2.1) is bounded below by $1/(2\tilde{D}^2)$. Checking the derivation we realize that the proof of Lemma 3 in (Vapnik and Chapelle, 2000) remains valid when substituting $1/(2\tilde{D}^2)$ with a larger number $1/(2\max_{i\neq j}\|\tilde{z}_i-\tilde{z}_j\|^2)$, and the inequality still holds.

By the definition of Λ_p^*,

$$\min_{\tilde{z}\in\Lambda_p^*} \|\tilde{z}_p - \tilde{z}\|^2 \leq \max_{i\neq j} \|\tilde{z}_i - \tilde{z}_j\|^2. \tag{2.3}$$

Since $\tilde{D}$ is the diameter of the smallest sphere containing all $\tilde{z}_1, \ldots, \tilde{z}_l$,

$$\max_{i\neq j} \|\tilde{z}_i - \tilde{z}_j\|^2 \leq \tilde{D}^2 = 4\tilde{R}^2. \tag{2.4}$$

Therefore, if x_p is recognized incorrectly, (2.1), (2.3), and (2.4) imply $\alpha_p^2\tilde{D}^4 \geq 1$. Then, $\alpha_p\tilde{D}^2 \geq 1$ so with $\sum_p \alpha_p = \|\tilde{w}\|^2$, we have

$$loo \leq \sum_{p=1}^{l} \alpha_p \tilde{D}^2 = 4\|\tilde{w}\|^2\tilde{R}^2.$$

Instead of using (2.4), from (2.2), we consider

$$\|\tilde{z}_i - \tilde{z}_j\|^2 \;=\; \frac{2}{C} + \|\phi(x_i) - \phi(x_j)\|^2$$

where $i \neq j$. With

$$\|\phi(x_i) - \phi(x_j)\|^2 \leq D^2 = 4R^2,$$

where R^2 is the objective value of (1.3), we obtain a different bound:

$$loo \leq (4R^2 + \frac{2}{C})\|\tilde{w}\|^2. \tag{2.5}$$

Some immediate comparisons between the two bounds are as follows. When σ^2 is fixed, we can easily prove that

$$\lim_{C\to\infty} \tilde{R}^2 = \lim_{C\to\infty} R^2 = \lim_{C\to\infty} (R^2 + \frac{0.5}{C}). \tag{2.6}$$

Furthermore, we have

Theorem 1 *With fixed* σ^2

$$(R^2 + \frac{0.5}{C})\|\tilde{w}\|^2 \leq \tilde{R}^2\|\tilde{w}\|^2$$

and $\tilde{R}^2/(R^2 + \frac{0.5}{C})$ *is a monotonically decreasing function of* C.

Thus, (2.5) is a tighter bound. The proof is in Appendix A.

When C is small, a large $1/C$ causes $\tilde{R}^2$ to be dominated by the term $1/C - \beta^T\beta/C$ in the objective function of (1.11). If we consider $\beta_i = 1/l, i = 1, \ldots, l$, a feasible solution of (1.11), then with $1 - \beta^T K\beta \geq 0$,

$$1 + \frac{1}{C} \geq \tilde{R}^2 \geq 1 + \frac{1}{C} - \beta^T(K + \frac{I}{C})\beta \geq \frac{1}{C}(1 - \frac{1}{l}). \tag{2.7}$$

Thus, if $l \gg 1$ and C is small,

$$\tilde{R}^2 \approx \frac{1}{C}$$

but

$$(R^2 + \frac{0.5}{C}) \approx \frac{0.5}{C}. \tag{2.8}$$

Therefore, when C is small, (1.10) overestimates the loo error. Interestingly this becomes a good property due to the following reason. As we can see in (2.1), the RM bound seriously overestimates the loo if α_p is large. This happens only when C is not small. It is very easy to see this for L1-SVM due to the constraint $\alpha_i \leq C$. For L2-SVM, when C is small, we can show that $\alpha_i \to \frac{l_2}{l_1}C$ if $y_i = 1$ and $\alpha_i \to \frac{l_1}{l_2}C$ if $y_i = -1$, where l_1 and l_2 are numbers of data with $y_i = 1$ and -1, respectively. The derivation is similar to (2.7). Thus, large α happens only when C is large. Therefore, the overestimation of loo at large C pushes the minimum to be at a smaller value of C. Therefore, we can see that $\tilde{R}^2$ puts penalty at small C so the minimum of the RM bound may be pushed back to the correct position. In Table 2.2 described later, we will see that when σ is fixed, $(R^2 + \frac{0.5}{C})\|\tilde{w}\|^2$ always returns a smaller C than $\tilde{R}^2\|\tilde{w}\|^2$.

Next, we present another bound for L2-SVM. If the problem is separable (i.e., hard-margin SVM) and the set of support vectors remains the same during the leave-one-out procedure, Theorem 3 of (Vapnik and Chapelle, 2000) shows:

$$y_p(f^0(x_p) - f^p(x_p)) = \alpha_p^0 S_p^2, \tag{2.9}$$

where x_p is any support vector, α_p^0 is the pth element of the dual solution α^0 by training the whole set, f^0 and f^p are the decision functions trained, respectively, on the whole set and after the point x_p has been removed, and

$$S_p^2 \equiv \min\{(\sum_{i=1}^{l} \lambda_i \phi(x_i))^2 \mid \lambda_p = -1, \sum_{i=1}^{l} \lambda_i = 0, \alpha_i^0 + y_i y_p \alpha_p^0 \lambda_i \geq 0\}.$$

C fixed

Data set	banana	image	splice	waveform	tree
Value of $\ln C$	-0.9	0.44	6.91	0	9.80
L2 test error	0.1118(-1.4)	0.0238(0.5)	0.0947(3.3)	0.0989(2.7)	0.1049(4.6)
$\tilde{R}^2\|\tilde{w}\|^2$	0.1141(-1.6)	0.0297(-0.3)	0.0998(3.1)	0.1030(2.1)	0.1703(-2.5)
$(R^2+0.5/C)\|\tilde{w}\|^2$	0.1141(-1.6)	0.0297(-0.3)	0.0998(3.1)	0.1267(5.6)	0.1703(-2.5)
$(R^2+0.25/C)\|\tilde{w}\|^2$	0.1141(-1.6)	0.2723(9.5)	0.0998(3.1)	0.1354(6.5)	0.1703(-2.5)

σ fixed

Data set	banana	image	splice	waveform	tree
Value of $\ln \sigma^2$	-1.39	-0.29	3.07	2.80	4.60
L2 test error	0.1120(-0.0)	0.0218(2.4)	0.1007(2.1)	0.0991(-0.0)	0.1049(9.7)
$\tilde{R}^2\|\tilde{w}\|^2$	0.1124(-0.9)	0.0297(0.4)	0.1016(10.0)	0.1007(-0.6)	0.1413(-1.4)
$(R^2+0.5/C)\|\tilde{w}\|^2$	0.1145(-1.5)	0.0366(-0.7)	0.1251(-1.4)	0.1048(-1.4)	0.1763(-2.3)
$(R^2+0.25/C)\|\tilde{w}\|^2$	0.1178(-2.1)	0.0564(-1.8)	0.4800(-10.0)	0.1124(-2.2)	0.2609(-3.1)

Table 2.2: Error rates with respect to the test set obtained using three L2-SVM RM bounds. For the case of "C fixed," the value given in parentheses is the value of $\ln \sigma^2$ where the bound attained the minimum; and, for the case of "σ fixed," the value given in the parentheses is the value of $\ln C$ where the bound attained the minimum.

Note that the assumption that support vectors do not change might not be always true so we must be cautious that things derived from (2.9) might not be real bounds of loo.

For L2-SVM, by (1.9), it can be reduced to the hard-margin form. Equation (2.9) becomes

$$y_p(\tilde{f}^0(\tilde{z}_p) - \tilde{f}^p(\tilde{z}_p)) = \alpha_p^0 \tilde{S}_p^2,$$

where $\tilde{z}_p$ is defined in (2.2). We then have the following lemma:

Lemma 1 If *the set of support vectors remains the same during the leave-one-out procedure, we have*

$$\tilde{S}_p^2 \leq D^2 + \frac{1}{C} + \begin{cases} \frac{1}{(l_{sv1}-1)C} & \text{if } y_p = 1, \\ \frac{1}{(l_{sv2}-1)C} & \text{if } y_p = -1. \end{cases}$$

where l_{sv1} and l_{sv2} are the number of support vectors with $y_i = 1$ and -1, respectively.

The proof is left in Appendix B. When there is an loo error, $y_p\tilde{f}^p(\tilde{z}_p) \leq 0$ and

$y_p\tilde{f}^0(\tilde{z}_p) = 1$. Thus, if $y_p = 1$ and $\alpha_p^0 < C$,

$$\begin{aligned} 1 &\leq y_p(\tilde{f}^0(\tilde{z}_p) - \tilde{f}^p(\tilde{z}_p)) \\ &= \alpha_p^0 \tilde{S}_p^2 \\ &\leq \alpha_p^0(D^2 + \frac{1}{C} + \frac{1}{(l_{sv1}-1)C}). \end{aligned} \tag{2.10}$$

Similarly, if $y_p = -1$ and $\alpha_p^0 < C$, then

$$1 \leq \alpha_p^0(D^2 + \frac{1}{C} + \frac{1}{(l_{sv2}-1)C}). \tag{2.11}$$

It is trivial that

$$1 \leq \alpha_p^0(D^2 + \frac{1}{C}), \tag{2.12}$$

if $\alpha_p^0 \geq C$.

Therefore, from (2.10) to (2.12), we have

$$\begin{aligned} loo &\leq \sum_{p=1}^{l} \alpha_p(D^2 + \frac{1}{C}) + \sum_{y_p=1,\alpha_p^0<C} \frac{\alpha_p^0}{(l_{sv1}-1)C} + \sum_{y_p=-1,\alpha_p^0<C} \frac{\alpha_p^0}{(l_{sv2}-1)C} \\ &\leq \sum_{p=1}^{l} \alpha_p^0(D^2 + \frac{1}{C}) + \frac{l_{sv1}}{l_{sv1}-1} + \frac{l_{sv2}}{l_{sv2}-1}. \end{aligned}$$

Since $l_{sv1}/(l_{sv1}-1)$ and $l_{sv2}/(l_{sv2}-1)$ are constants, $e^T\alpha(D^2 + 1/C)$ can be considered as a bound of loo for L2-SVM. With $\|\tilde{w}\|^2 = e^T\alpha$ for L2-SVM, a new bound is:

$$(R^2 + \frac{0.25}{C})\|\tilde{w}\|^2. \tag{2.13}$$

Note that using $\alpha = C\xi$ for L2-SVM, this bound can be rewritten as

$$D^2 e^T\alpha + \sum_{i=1}^{l} \xi_i, \tag{2.14}$$

the same form as (1.12) for L1-SVM.

Interestingly, it is an even tighter bound than (2.5) as $R^2 + \frac{0.25}{C}$ is smaller than $R^2 + \frac{0.5}{C}$. In Table 2.2 we present results using the two new bounds for L2-SVM. It can be clearly seen that both new bounds perform more poorly than the original RM bound, and (2.13) is particularly bad. In the following we list some further observations:

1. When C is fixed, the accuracy of using (2.5) for waveform is not good because it obtains a too large σ. Then, for (2.13), it returns an even larger σ so the error rate further increases. Especially for the problem image a very large $\ln \sigma^2 = 9.5$ is obtained.

   If we check the limiting behavior of these three bounds, when $\sigma \to \infty$,

$$\tilde{R}^2 \|\tilde{w}\|^2 \approx \frac{4 l_1 l_2}{l}, \tag{2.15}$$

$$(R^2 + \frac{0.5}{C}) \|\tilde{w}\|^2 \approx \frac{2 l_1 l_2}{l}, \tag{2.16}$$

$$(R^2 + \frac{0.25}{C}) \|\tilde{w}\|^2 \approx \frac{l_1 l_2}{l}, \tag{2.17}$$

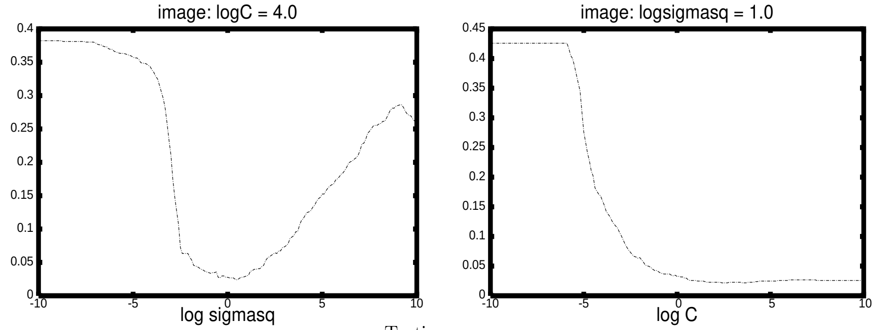
   where l_1 and l_2 are the number of data with $y_i = 1$ and -1, respectively. We leave details of (2.15)-(2.17) in Appendix C.

   Therefore, we suspect that smaller values when $\sigma \to \infty$ cause the bound (2.13) to have minima at large σ. To confirm this, in Figure 1 we present the value of (2.13), using the problem image.

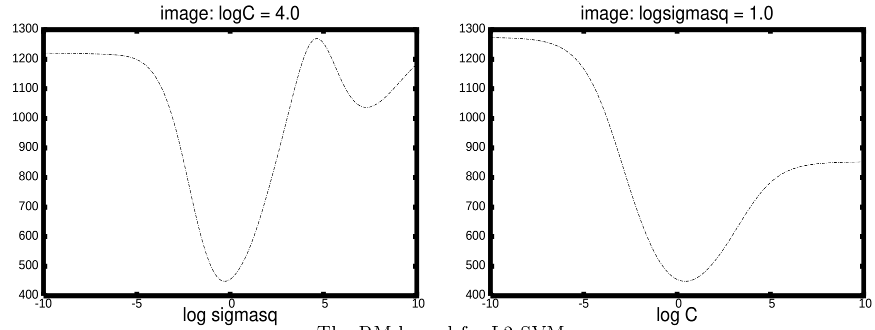
   Clearly there are two local minima where the left one is better but is not chosen. This seems to suggest that as (1.10) has larger values when $\sigma \to \infty$, it can avoid that the global minimum happens at a wrong place.

2. When σ^2 is fixed, the situation is also similar, as is shown in Figure 1. The tighter the bound is, the worse the test accuracy is as the optimal C becomes too small. If $R^2 > 1/2$, we can further prove that changing the bound from $\tilde{R}^2$ to $R^2 + \frac{0.5}{C}$ and then $R^2 + \frac{0.25}{C}$, the optimal C can only be smaller. In the proof of Theorem 1, we show that $R^2 \geq 1/2$ and then $\tilde{R}^2/(R^2 + \frac{0.5}{C})$ is a decreasing function of C. If we further assume $R^2 > 1/2$, both $\tilde{R}^2/(R^2 + \frac{0.5}{C})$ and $(R^2 + \frac{0.5}{C})/(R^2 + \frac{0.25}{C})$ are strictly decreasing functions of C so this result on the position of optimal C follows from the following theorem (proof in Appendix E):

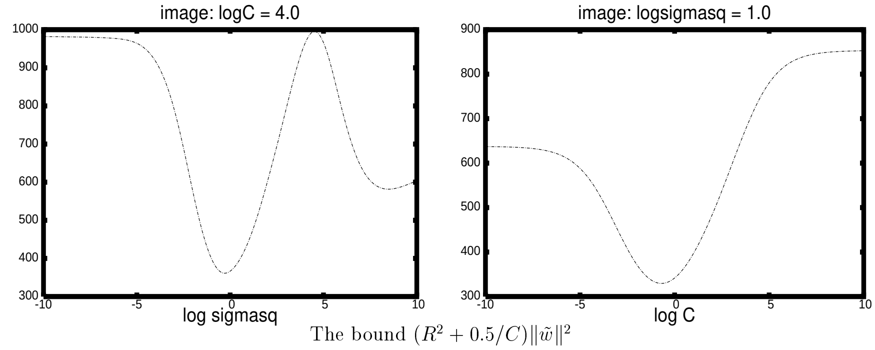
   Theorem 2 *If $f_1, f_2, g > 0$ are positive functions on $(0, \infty)$ and f_1/f_2 is strictly decreasing, then any $f_2 g$'s global optimum is greater than or equals to that of $f_1 g$.*



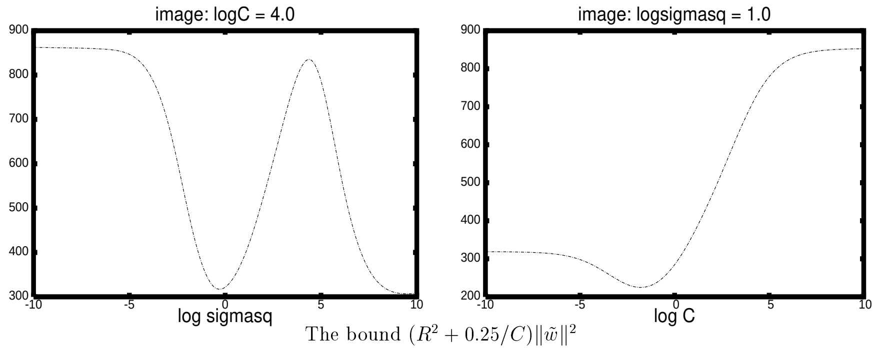
Testing error



The RM bound for L2-SVM



The bound $(R^2 + 0.5/C)\|\tilde{w}\|^2$



The bound $(R^2 + 0.25/C)\|\tilde{w}\|^2$

Figure 1: image data set. The figures on the left-hand side show the values of the bounds with respect to $\ln \sigma^2$ with fixed $\ln C$. The figures on the right-hand side show the values of the bounds with respect to $\ln C$ with fixed $\ln \sigma^2$. For the case of $\ln \sigma^2$ fixed, we can see that the local minimum of larger C becomes dominant as the bound becomes tighter. For the case of $\ln C$ fixed, the tighter the bound is, the smaller the optimal C is.

Therefore, as we have mentioned earlier, there is a benign overestimation for $\tilde{R}$ when C is small. In summary, experiments in this section tell us that

1. Finding a bound whose minima are in a good region may be more important than its tightness.

2. A good bound should avoid minima that happen at the boundary (i.e., too small or too large C and σ^2).

3 Some Heuristic Bounds for L1-SVM

Based on the experience in the previous section, we search for better radius margin bounds for L1-SVM which can replace the $D^2 e^T \alpha + \sum_{i=1}^{l} \xi_i$ used in (Duan et al., 2003). Our strategy is to consider that (2.13) for L2-SVM is the counterpart of $D^2 e^T \alpha + \sum_{i=1}^{l} \xi_i$ for L1-SVM as the explanation in (2.14) shows they have the same form. Then, by investigating the difference between (1.10) and (2.13), we seek for the counterpart of (1.10) for L1-SVM.

If we consider

$$\tilde{R}^2 \approx R^2 + \frac{1}{C},$$

$\tilde{R}^2 \|\tilde{w}\|^2$ is similar to

$$(R^2 + \frac{1}{C})\|\tilde{w}\|^2 = R^2 e^T \alpha + \sum_{i=1}^{l} \xi_i$$

as $\|\tilde{w}\|^2 = e^T \alpha$ and $\alpha = C\xi$ for L2-SVM.

Thus, for L1-SVM,

$$R^2 e^T \alpha + \sum_{i=1}^{l} \xi_i \tag{3.1}$$

may be a good bound. Now using (1.7),

$$C \sum_{i=1}^{l} \xi_i = e^T \alpha - \|w\|^2 \leq e^T \alpha,$$

so another possibility is

$$(R^2 + \frac{1}{C}) e^T \alpha = (R^2 + \frac{1}{C})(\|w\|^2 + C \sum_{i=1}^{l} \xi_i). \tag{3.2}$$

Later on we will conduct experiments on these two new bounds. Another issue about these bounds is their differentiability. We would like them to be differentiable so gradient-based methods can be used to find a local minimum. Unfortunately, as we show in the Appendix F, the two new bounds for L1-SVM may not be differentiable.

Therefore, if we require differentiability at any C and σ^2, the above two new bounds for L1-SVM are not applicable. Thus, we propose a further modification of (3.2):

$$(R^2 + \frac{\Delta}{C})(\|w\|^2 + 2C\sum_{i=1}^{l}\xi_i), \tag{3.3}$$

where Δ is a positive constant close to one. As we discussed in the paragraph after (2.8), Δ/C can be thought as penalty term for small C. If we take $\Delta = 1$, the main change from (3.2) is to replace $e^T\alpha$ by a differentiable term.

This modified bound is still an loo bound. It even need not the assumption that support vectors do not change. From Lemma 3 of (Vapnik and Chapelle, 2000) and $S_p \leq D$, we have

$$\begin{aligned} 1 &\leq \alpha_p^0 S_p \max(D, \frac{1}{\sqrt{C}}) \\ &\leq \alpha_p^0 \max(D, \frac{1}{\sqrt{C}})^2 \\ &\leq \alpha_p^0 (D^2 + \frac{1}{C}), \end{aligned}$$

where α_p^0 is the pth element of the dual solution α^0 by training the whole set. Therefore,

$$\begin{aligned} loo &\leq \sum_{p=1}^{l} \alpha_p^0 (D^2 + \frac{1}{C}) \\ &= (\|w\|^2 + C\sum_{i=1}^{l}\xi_i)(D^2 + \frac{1}{C}) \\ &\leq (\|w\|^2 + 2C\sum_{i=1}^{l}\xi_i)(D^2 + \frac{4\Delta}{C}) \end{aligned}$$

when $\Delta \geq 1/4$.

Table 3.1 presents a comparison of different bounds for L1-SVM: (1.12), two improvements (3.1) and (3.2), and the differentiable bound (3.3) with $\Delta = 1$ and

C fixed					
Data set	banana	image	splice	waveform	tree
Value of $\ln C$	5.2	4.0	0.4	1.4	8.6
L1 test error	0.1043(0.6)	0.0188(1.0)	0.0947(3.4)	0.1022(3.2)	0.1086(3.8)
$D^2(e^T\alpha) + \sum \xi_i$	0.5594(10.0)	0.2564(10.0)	0.1545(8.1)	0.1533(8.0)	0.1703(-2.5)
$R^2(e^T\alpha) + \sum \xi_i$	0.3520(-6.6)	0.0376(-1.2)	0.0975(3.2)	0.1354(6.8)	0.1703(-2.5)
$(R^2 + 1/C)e^T\alpha$	0.3520(-6.6)	0.0376(-1.2)	0.1421(6.1)	0.1143(1.4)	0.1703(-2.5)
$(R^2 + 1/C)(w^2 + 2C\sum \xi_i)$	0.3520(-6.6)	0.0376(-1.2)	0.0998(3.1)	0.1143(1.4)	0.1703(-2.5)
$(R^2 + 0.5/C)(w^2 + 2C\sum \xi_i)$	0.3520(-6.6)	0.0376(-1.2)	0.0998(3.1)	0.1143(1.4)	0.1703(-2.5)

σ fixed					
Data set	banana	image	splice	waveform	tree
Value of $\ln \sigma^2$	0.6	1.0	3.4	3.2	3.8
L1 test error	0.1043(5.2)	0.0188(3.9)	0.0947(0.4)	0.1022(1.4)	0.1086(8.6)
$D^2(e^T\alpha) + \sum \xi_i$	0.3843(-2.9)	0.1287(-2.9)	0.1945(-2.4)	0.1409(-2.5)	0.2609(-10.0)
$R^2(e^T\alpha) + \sum \xi_i$	0.2284(-1.2)	0.0802(-1.3)	0.1136(-0.7)	0.1143(-1.1)	0.1437(-1.5)
$(R^2 + 1/C)e^T\alpha$	0.5594(-10.0)	0.0723(-1.0)	0.0966(10.0)	0.1128(-0.6)	0.2609(-10.0)
$(R^2 + 1/C)(w^2 + 2C\sum \xi_i)$	0.2022(-0.8)	0.0485(-0.4)	0.0966(10.0)	0.1130(-0.5)	0.2609(-10.0)
$(R^2 + 0.5/C)(w^2 + 2C\sum \xi_i)$	0.3618(-2.3)	0.0891(-1.4)	0.0966(10.0)	0.1157(-1.2)	0.2609(-10.0)

Table 3.1: Error rates with respect to the test set obtained using various L1-SVM bounds at their minimal points. For the case of "C fixed," the value given in parentheses is the value of $\ln \sigma^2$ where the bound attained the minimum; and, for the case of "σ fixed," the value given in the parentheses is the value of $\ln C$ where the bound attained the minimum.

$\Delta = 0.5$. It can be clearly seen that bounds proposed in this section are better than (1.12). Besides the differentiability, the derivative of (3.3) can be easily computed. Details will be in Section 5.1.

4 Overall Performance of Bounds for L1-SVM and L2-SVM

In previous sections we have discussed some bounds for L1- and L2-SVMs. However, our experiments were based on fixing one parameter and testing the other one. This may not reveal how bounds perform when C and σ are considered together. In this section, we obtain the minimal point of each bound in the $(\ln C, \ln \sigma^2)$ plane. Then a model based on this (C, σ) is trained to predict the test data.

In addition to the five data sets used in the previous sections, there are three more data sets here. Most of them are from (Rätsch, 1999), as in (Duan et al., 2003). The `tree` data set was originally used in (Bailey et al., 1993). Table 4.1 shows some statistics about the data sets.

Data set	banana	diabetes	image	ringnorm
Number of features	2	8	18	20
Number of training samples	400	468	1300	400
Number of testing samples	4900	300	1010	7000
Number of positive samples in training set	183	170	560	207
Number of positive samples in test set	2741	98	430	3457
Data set	splice	tree	twonorm	waveform
Number of features	60	18	20	21
Number of training samples	1000	700	400	400
Number of testing samples	2175	11692	7000	4600
Number of positive samples in training set	517	502	207	268
Number of positive samples in test set	1131	8642	3496	3085

Table 4.1: Some statistics about the data sets.

We consider the bounded region that $\ln C$ and $\ln \sigma^2$ are both from -10 to 10. For L1-SVM our new bounds tend to increase the C value at the minimal point due to the change from R^2 to $R^2 + \frac{\Delta}{C}$. Sometimes it pushes the minimal point to the boundary at $\ln C = 10$. However, we observe that at a smaller C, with the same σ, the model generated already has no bounded support vectors. That is, $\xi_i = 0, i = 1, \ldots, l$, so the training data are fully separated. Therefore, the model using larger C will be the same and there is no reason to further increase C even though the bound can still be decreased. This can be seen in Table 4.3 described later. In our experiment, once at a (C, σ) if data are fully separated, for the same σ, larger C will not be considered.

Table 4.2 presents the results of using different bounds. It shows in each block the optimal parameters $(\ln C, \ln \sigma^2)$ and the error rate with respect to the test set for each bound and data set. Here, we use $\Delta = 1$ and $\Delta = 0.5$ for our differentiable modifications of the L1-SVM bound (3.3). We discretize $[-10, 10]$ to 21 points so at most 441 parameters are tried. Some observations are as follows:

1. Compared with the results in Section 2, the difference in performance among different bounds becomes less significant when C and σ are considered together. This shows that fixing one parameter may not be adequate all the time.

2. The modified RM bound $D^2 e^T \alpha + \sum \xi_i$ performs worse than our modifications for many data sets. Similarly, its counterpart for L2-SVM, $(R^2 + \frac{0.25}{C})\|\tilde{w}\|^2$, is also the worst except for diabetis, ringnorm, and twonorm.

Data set	banana	diabetes	image	ringnorm
L1 test error	(7, 1) 0.1084	(6, 6) 0.2267	(4, 1) 0.0188	(-2, 2) 0.0143
L1 $D^2e^T\alpha + \sum \xi_i$	(-3, -1) 0.1482	(-10, 10) 0.3267	(-3, 1) 0.1297	(-2, 2) 0.0143
L1 $R^2e^T\alpha + \sum \xi_i$	(-2, -2) 0.1198	(-2, 2) 0.2467	(-1, 0) 0.0535	(-1, 2) 0.0161
L1 $(R^2 + 1/C)e^T\alpha$	(-1, -1) 0.1182	(-10, 10) 0.3267	(4, -1) 0.0356	(1, 2) 0.0199
L1 $(R^2 + 1/C)(\|w\|^2 + 2C\sum \xi_i)$	(-1, -2) 0.1135	(2, -1) 0.3233	(4, -1) 0.0356	(1, 2) 0.0199
L1 $(R^2 + 0.5/C)(\|w\|^2 + 2C\sum \xi_i)$	(-1, -2) 0.1135	(-10, 10) 0.3267	(4, -1) 0.0356	(1, 2) 0.0199
L2 test error	(0, -1) 0.1112	(6, 4) 0.2233	(9, 4) 0.0188	(-3, 2) 0.0139
L2 RM bound	(-1, -2) 0.1141	(-2, 2) 0.2400	(0, 0) 0.0297	(2, 2) 0.0190
L2 $(R^2 + 0.5/C)\|\tilde{w}\|^2$	(-2, -1) 0.1212	(-3, 2) 0.2367	(-1, 0) 0.0406	(0, 2) 0.0161
L2 $(R^2 + 0.25/C)\|\tilde{w}\|^2$	(-2, -1) 0.1212	(-3, 2) 0.2367	(-2, 1) 0.0861	(-1, 2) 0.0147
Data set	splice	tree	twonorm	waveform
L1 test error	(1, 4) 0.0970	(9, 4) 0.1089	(-2, 3) 0.0234	(1, 3) 0.1035
L1 $D^2e^T\alpha + \sum \xi_i$	(-2, 5) 0.1480	(-3, 1) 0.1483	(-2, 4) 0.0243	(-2, 3) 0.1322
L1 $R^2e^T\alpha + \sum \xi_i$	(-1, 4) 0.1200	(-2, 0) 0.1378	(-1, 3) 0.0240	(-1, 2) 0.1096
L1 $(R^2 + 1/C)e^T\alpha$	(2, 3) 0.1007	(-2, 1) 0.1401	(2, 2) 0.0349	(-1, 3) 0.1111
L1 $(R^2 + 1/C)(\|w\|^2 + 2C\sum \xi_i)$	(2, 3) 0.1007	(3, -2) 0.1497	(2, 2) 0.0349	(3, 2) 0.1098
L1 $(R^2 + 0.5/C)(\|w\|^2 + 2C\sum \xi_i)$	(2, 3) 0.1007	(-2, 1) 0.1401	(2, 2) 0.0349	(-1, 3) 0.1111
L2 test error	(1, 4) 0.0979	(10, 5) 0.1066	(-2, 2) 0.0223	(0, 3) 0.1011
L2 RM bound	(10, 3) 0.1007	(-2, 1) 0.1354	(0, 3) 0.0260	(0, 2) 0.1039
L2 $(R^2 + 0.5/C)\|\tilde{w}\|^2$	(-1, 5) 0.1292	(-2, 1) 0.1354	(0, 4) 0.0251	(-1, 3) 0.1041
L2 $(R^2 + 0.25/C)\|\tilde{w}\|^2$	(-2, 5) 0.1338	(-3, 1) 0.1433	(-1, 4) 0.0240	(-2, 3) 0.1139

Table 4.2: Error rates with respect to the test set obtained using various L1- and L2-SVM bounds at their minimal points. The value given in parentheses is the value of $(\ln C, \ln \sigma^2)$ where the bound attained the minimum.

3. Our new bounds $R^2e^T\alpha + \sum \xi_i$ and $(R^2 + \frac{1}{C})e^T\alpha$ for L1-SVM have generated results competitive with those by the RM bound for L2-SVM.

4. Since the L1-SVM counterparts of $\tilde{R}^2\|\tilde{w}\|^2$ for L2-SVM are not differentiable, we further modify them to $(R^2 + \frac{1}{C})(\|w\|^2 + 2C\sum \xi_i)$ or $(R^2 + \frac{0.5}{C})(\|w\|^2 + 2C\sum \xi_i)$. Results show that fortunately the test accuracy does not change much.

Next, we compare the number of support vectors for optimal models generated by these bounds. In particular, we are interested in the difference between L1- and L2-SVM. If L1-SVM does not possess an advantage in this aspect, it may be better to consider L2-SVM as its RM bound is derived in a more natural way.

Table 4.3 presents the total number of support vectors and the number of free support vectors for the optimal models. There are some points to note:

1. It can be seen that the numbers of support vectors are fewer for L1-SVM than for L2-SVM. This is consistent with the common understanding that the quadratic cost function leads to more data with small nonzero ξ_i.

Data set	banana	diabetes	image	ringnorm
L1 test error	(88, 16)	(239, 18)	(276, 245)	(212, 45)
L1 $D^2e^T\alpha + \sum \xi_i$	(345, 6)	(340, 0)	(895, 29)	(212, 45)
L1 $R^2e^T\alpha + \sum \xi_i$	(262, 21)	(326, 12)	(622, 215)	(145, 63)
L1 $(R^2 + 1/C)e^T\alpha$	(175, 21)	(340, 0)	(708, 708)	(101, 101)
L1 $(R^2 + 1/C)(\|w\|^2 + 2C\sum \xi_i)$	(185, 35)	(435, 435)	(708, 708)	(101, 101)
L1 $(R^2 + 0.5/C)(\|w\|^2 + 2C\sum \xi_i)$	(185, 35)	(340, 0)	(708, 708)	(101, 101)
L2 test error	(254, 254)	(356, 356)	(225, 225)	(400, 400)
L2 RM bound	(378, 378)	(447, 447)	(886, 886)	(120, 120)
L2 $(R^2 + 0.5/C)\|\tilde{w}\|^2$	(382, 382)	(468, 468)	(1120, 1120)	(188, 188)
L2 $(R^2 + 0.25/C)\|\tilde{w}\|^2$	(382, 382)	(468, 468)	(1182, 1182)	(269, 269)

Data set	splice	tree	twonorm	waveform
L1 test error	(601, 481)	(197, 62)	(201, 16)	(120, 52)
L1 $D^2e^T\alpha + \sum \xi_i$	(909, 13)	(401, 59)	(242, 18)	(226, 12)
L1 $R^2e^T\alpha + \sum \xi_i$	(728, 130)	(443, 180)	(136, 32)	(203, 55)
L1 $(R^2 + 1/C)e^T\alpha$	(877, 877)	(345, 67)	(159, 159)	(181, 21)
L1 $(R^2 + 1/C)(\|w\|^2 + 2C\sum \xi_i)$	(877, 877)	(663, 663)	(159, 159)	(151, 151)
L1 $(R^2 + 0.5/C)(\|w\|^2 + 2C\sum \xi_i)$	(877, 877)	(345, 67)	(159, 159)	(181, 21)
L2 test error	(761, 761)	(290, 290)	(380, 380)	(198, 198)
L2 RM bound	(892, 892)	(671, 671)	(179, 179)	(238, 238)
L2 $(R^2 + 0.5/C)\|\tilde{w}\|^2$	(956, 956)	(671, 671)	(191, 191)	(240, 240)
L2 $(R^2 + 0.25/C)\|\tilde{w}\|^2$	(998, 998)	(700, 700)	(286, 286)	(297, 297)

Table 4.3: Number of support vectors and number of free support vectors obtained using various L1- and L2-SVM bounds at their minimal points. For the L2 cases, the number of free support vectors equals the number of support vectors because there is no upper bound for α. For the L1 cases of which the number of free support vectors equals the number of support vectors, the training data are fully separated.

2. The numbers of free support vectors are also listed here because they affect the training time. When we use the shrinking technique in (Joachims, 1998), we only select the working set from the free support vectors, and thus fewer support vectors generally yields shorter training time.

3. For L2-SVM, the number of free support vectors equals the number of support vectors because there is no upper bound for α. For some problems in L1, the number of free support vectors also equals the number of support vectors because the training data are fully separated in the model generated.

5 Efficient Implementation by Gradient Methods

Experiments in the previous section have shown the viability of some possible bounds for L1-SVM. In this section, we further discuss their efficient implementations by numerical optimization techniques. We first address the differentiability of the bounds in Section 5.1. With the differentiability, we are able to apply gradient-based optimization methods. These methods iteratively find a descent direction p by ensuring that $\nabla f(x^k)^T p < 0$, where x^k is the solution of the kth iteration and $f(x)$ is the function to be minimized. If f is not continuously differentiable, $\nabla f(x^k)^T p < 0$ does not imply that p is a descent direction. Therefore, algorithms have troubles to strictly decrease the function value. We will check it and compute the derivative in Section 5.1. However, none of the bounds is twice differentiable. This will also be discussed in Section 5.1. Therefore, Newton's method cannot be used unless the Hessian (second derivative) is approximated by finite difference. Then quasi-Newton and nonlinear conjugate gradient methods are the remaining major candidates. Following (Keerthi, 2002), we decide to work on quasi-Newton methods in Section 5.2.

On the other hand, even if the second derivative is available, we still may not consider Newton's method. The advantage of using the second order information is to have fast convergence for a high precision. It has been pointed out in (Keerthi, 2002) that a high precision is not important on improving the test accuracy. Both (Keerthi, 2002) and experiments here indicate that under appropriate stopping conditions, the number of quasi-Newton iterations is already small. Furthermore, the computational cost on the Hessian matrix is expensive compared to the gradient. Thus, the effort of each Newton iteration is more than the quasi-Newton one. Details will be in Section 5.1.

5.1 Differentiability of Bounds for L1-SVM

We have proposed to use (3.3) as the bound for L1-SVM. First, we denote it as $f(C, \sigma^2)$ and calculate its partial derivatives:

$$\begin{aligned} \frac{\partial f}{\partial C} &= \frac{\partial (R^2 + \frac{\Delta}{C})}{\partial C}(\|w\|^2 + 2C\sum_{i=1}^{l}\xi_i) + \frac{\partial(\|w\|^2 + 2C\sum_{i=1}^{l}\xi_i)}{\partial C}(R^2 + \frac{\Delta}{C}), \\ \frac{\partial f}{\partial(\sigma^2)} &= \frac{\partial (R^2 + \frac{\Delta}{C})}{\partial(\sigma^2)}(\|w\|^2 + 2C\sum_{i=1}^{l}\xi_i) + \frac{\partial(\|w\|^2 + 2C\sum_{i=1}^{l}\xi_i)}{\partial(\sigma^2)}(R^2 + \frac{\Delta}{C}). \end{aligned}$$

For experiments in this section, we consider only $\Delta = 1$. Note that $f(C, \sigma^2)$ may not be convex so we are solving a non-convex optimization problem. We will present an example in Appendix F.

The differentiability of both R^2 and $\|w\|^2 + 2C\sum_{i=1}^{l}\xi_i$ relies on results of perturbation analysis of optimization problems. Theorem 4.1 of (Bonnans and Shapiro, 1998) gives a general result:

Theorem 3 *For an optimization problem of the form* $f^*(v) = \min_{\alpha \in A} f(\alpha, v)$ *with optimization variable vector* α*, parameter vector* v*, and* A *independent of* v*, if for each* v *there exists an unique* $\alpha^*(v) \in A$ *such that* $f^*(v) = f(\alpha^*(v), v)$ *then* $\frac{df^*}{dv}(v)$ *exists and* $\frac{\partial f^*}{\partial v_i}$ *is given by* $\frac{\partial f}{\partial v_i}(\alpha^*, v)$ *for any parameter* v_i*.*

Note that $\frac{df^*}{dv}(v)$ does not depend on the existence of $\frac{d\alpha^*}{dv}(v)$. However, though $\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i$ is the objective function of the primal L1-SVM, the theorem does not apply as

1. The optimal ξ may not be unique since the optimal b is not unique when there is no free support vector.

2. Constraints involve with the kernel parameter σ. That is, A is not independent of v.

Therefore, we look at the dual problem as at any optimal solution,

$$\|w\|^2 + 2C\sum_{i=1}^{l}\xi_i = 2(e^T\alpha - \frac{1}{2}\alpha^T Q\alpha). \tag{5.1}$$

For the RBF kernel, if no training data are at the same point (i.e., $x_i \neq x_j$), Q is positive definite (Micchelli, 1986). Then, (1.6) is a strictly convex quadratic programming problem, so the optimal α is unique under any given parameters. Without much loss of generality from now on we assume $x_i \neq x_j$. Then a remaining difficulty is that constraints of the dual form of L1-SVM are related to C. Therefore, we transform the dual to a problem whose constraints are independent of parameters:

From (1.6), let $\alpha = C\bar{\alpha}$:

$$\begin{aligned} \min_{\bar{\alpha}} \quad & \frac{1}{2}\bar{\alpha}^T Q\bar{\alpha} - \frac{e^T\bar{\alpha}}{C} \\ \text{subject to} \quad & 0 \leq \bar{\alpha} \leq 1, i = 1, \ldots, l, \\ & y^T\bar{\alpha} = 0. \end{aligned} \tag{5.2}$$

Then,

$$\frac{1}{2}\alpha^T Q\alpha - e^T\alpha = C^2(\frac{1}{2}\bar{\alpha}Q\bar{\alpha} - \frac{e^T\bar{\alpha}}{C}). \tag{5.3}$$

Finally, we can apply Theorem 3 on (5.2) so

$$\begin{aligned} \frac{\partial(\frac{1}{2}\alpha^T Q\alpha - e^T\alpha)}{\partial C} &= 2C(\frac{1}{2}\bar{\alpha}Q\bar{\alpha} - \frac{e^T\bar{\alpha}}{C}) + C^2(\frac{e^T\bar{\alpha}}{C^2}) \\ &= \frac{2}{C}(\frac{1}{2}\alpha^T Q\alpha - e^T\alpha) + \frac{1}{C}(e^T\alpha) \\ &= \frac{1}{C}(\alpha^T Q\alpha - e^T\alpha). \end{aligned}$$

With (5.1),

$$\frac{\partial(\|w\|^2 + 2C\sum_i \xi_i)}{\partial C} = \frac{2}{C}(e^T\alpha - \alpha^T Q\alpha) = 2\sum \xi_i. \tag{5.4}$$

Similarly,

$$\frac{\partial(\|w\|^2 + 2C\sum \xi_i)}{\partial(\sigma^2)} = -\sum \alpha_i\alpha_j y_i y_j \frac{\partial K(x_i, x_j)}{\partial(\sigma^2)}, \tag{5.5}$$

$$\frac{\partial R^2 + \Delta/C}{\partial C} = \frac{-\Delta}{C^2}, \frac{\partial R^2 + \Delta/C}{\partial(\sigma^2)} = -\sum \beta_i\beta_j \frac{\partial K(x_i, x_j)}{\partial(\sigma^2)}, \tag{5.6}$$

where

$$\frac{\partial K(x_i, x_j)}{\partial(\sigma^2)} = K(x_i, x_j)\frac{\|x_i - x_j\|^2}{2\sigma^4}.$$

Note that the differentiability is special to the RBF kernel since the uniqueness of α may not hold in the hyperspace for other kernels. However, for L2-SVM, as the Hessian of (1.8) and (1.11) are positive definite, they always have unique solutions. Thus, RM bound for L2-SVM is differentiable no matter which kernel is used.

In contrast to (3.3), the derivatives of other bounds may require the partial derivatives of α with respect to the hyperparameters C and σ^2. This dependency raise two drawbacks — non-differentiability and high computational cost. For the non-differentiability, we give two examples for L1-SVM and L2-SVM in Appendices F and G respectively, showing that $\frac{\partial \alpha}{\partial C}$ may not be differentiable. To compute the derivatives of (3.3) we need to compute (5.4)–(5.6). Since only nonzero α_i contribute to (5.5), the computational cost is $O(l_{sv}^2 n)$ where l_{sv} denotes the number of nonzero elements in α. The situation of (5.6) is similar. However, as shown in (Chapelle et al., 2002), under the assumption that the support vectors do not change, to compute $\frac{\partial \alpha}{\partial v}$ where v denotes hyperparameters requires the inverse of Q_{sv}, which is the submatrix of Q involving with support vectors, and the inversion of Q_{sv} usually takes $O(l_{sv}^3)$ time.

Although we have differentiable bounds for both L1- and L2-SVMs, they are not twice differentiable since the second order derivatives require $\frac{\partial \alpha}{\partial v}$ also. For example, the second order derivative of our RM bound contains the partial derivative of (5.4) with respect to C, which is

$$\frac{\partial}{\partial C}\frac{2}{C}(e^T\alpha - \alpha^T Q\alpha) = -\frac{2}{C^2}(e^T\alpha - \alpha^T Q\alpha) + \frac{2}{C}(e^T - 2\alpha^T Q)\frac{\partial \alpha}{\partial C},$$

dependent with the non-differentiable term $\frac{\partial \alpha}{\partial C}$. Similarly, the need of $\frac{\partial \alpha}{\partial C}$ also forbid us using second derivatives due to its expensive computational cost discussed before.

5.2 Quasi-Newton Methods

Following earlier experiments, we search on the $(\ln C, \ln \sigma^2)$ space. An advantage is that the optimization problem becomes unconstrained. Otherwise, $C \geq 0$ and the property $\sigma^2 = (-\sigma)^2$ both cause problems. However, practically we still have to specify upper and lower bounds for $\ln C$ and $\ln \sigma^2$. As in Section 4, we restrict them to be in $[-10, 10]$. Once the optimization procedure has reached the boundary and still tends to move out, we stop it.

Working on the $(\ln C, \ln \sigma^2)$ space, the gradient is also different. We must modify (5.4)-(5.6) according to the following chain rules:

$$\frac{\partial}{\partial \ln C} = \frac{\partial}{\partial C} \times C, \text{ and } \frac{\partial}{\partial \ln \sigma^2} = \frac{\partial}{\partial (\sigma^2)} \times \sigma^2.$$

We consider the Broyden-Fletcher-Goldfarb-Shanno (BFGS) quasi-Newton method (see (Nocedal and Wright, 1999, page 198)) which has the following procedure: Assume x^k is the current iterate and $f(x)$ is the function to be minimized:

1. Compute a search direction $p = -H_k \nabla f(x^k)$.
2. Find $x^{k+1} = x^k + \lambda p$ using a line search to ensure sufficient decrease.
3. Obtain H_{k+1} by

$$H_{k+1} = (I - \frac{sy^T}{y^T s}) H_k (I - \frac{ys^T}{y^T s}) + \frac{ss^T}{y^T s},$$

   where

$$s = x^{k+1} - x^k \text{ and } y = \nabla f(x^{k+1}) - \nabla f(x^k).$$

Here, H_k serves as the inverse of an approximate Hessian. The sufficient decrease by the line search usually means

$$f(x^k + \lambda p) \leq f(x^k) + \sigma_1 \lambda \nabla f(x^k)^T p, \tag{5.7}$$

where $0 < \sigma_1 < 1$ is a positive constant. Since $p = -H_k \nabla f(x^k)$, we need H_k to be positive definite to ensure that p is a descent direction. A good property of the BFGS formula is that H_{k+1} inherits the positive definiteness of H_k as long as $y^T s > 0$. The condition $y^T s > 0$ is guaranteed to hold if the initial Hessian is positive definite (e.g. the identity) and the step size is determined by satisfying the second Wolfe condition:

$$\nabla f(x^k + \lambda p) \geq \sigma_2 \nabla f(x^k)^T p, \tag{5.8}$$

where $0 < \sigma_1 < \sigma_2 < 1$. Note that (5.7) is usually called the first Wolfe condition.

The main disadvantage of considering (5.8) is that the line search becomes more complicated. In addition, (5.8) involves the calculation of $\nabla f(x^k + \lambda p)$ so for each trial step size, a gradient evaluation is needed. Though $\nabla f(x^k + \lambda p)$ is easily computed once $f(x^k + \lambda p)$ is computed (as pointed out in (Keerthi, 2002)), this still contributes some additional cost.

We consider an alternative approach to avoid the more complicated line search. If $y^T s < 0$, H_k is not updated. More specifically, H_{k+1} is determined by

$$H_{k+1} = \begin{cases} (I - \frac{sy^T}{y^T s})H_k(I - \frac{ys^T}{y^T s}) + \frac{ss^T}{y^T s} & \text{if } y^T s > \eta, \\ H_k & \text{otherwise,} \end{cases} \tag{5.9}$$

where η is usually a small constant. Here, we simply use $\eta = 0$. Then the second Wolfe condition is not needed. The disadvantage of (5.9) is that it may lengthen the procedure if in many iterations H_k is not updated. Here, we do not have this problem as in our experiments for only very few iterations H_k is not updated.

It is unfortunate that the global convergence of BFGS algorithm for non-convex problems is still an open issue. Existing convergence results (e.g. (Li and Fukushima, 2001)) require either further modification of the algorithm or more conditions on the function.

Regarding different trials of step size to ensure the sufficient decrease condition (5.7), we can simply find the largest value in a set $\{\gamma^i \mid i = 0, 1, ...\}$ such that (5.7) holds ($\gamma = 1/2$ used in this paper). More advanced techniques such as quadratic or cubic interpolation (Nocedal and Wright, 1999) can also be used. However, we do not recommend them here as for most iterations, an initial step size $\lambda = 1$ already satisfies (5.8). Also note that in early iterations, the search direction p may be a vector that is large in size so, using the initial $\lambda = 1$, sometimes $x^k + p$ is far beyond the region considered. Thus, numerical instability may occur. Therefore, if $x^k + \lambda p$ is outside the $[-10, 10] \times [-10, 10]$ region, we project it back by

$$P(x_i^k + \lambda p_i) = \max(-10, \min(x_i^k + \lambda p_i, 10)).$$

We further avoid a too large step size by requiring the initial λ to satisfy

$$\|P(x^k + \lambda p) - x^k\| \le 2. \tag{5.10}$$

This reduces the chance of going to a wrong region in the beginning.

5.3 Initial Points and Stopping Criteria

(Keerthi, 2002) proposed to use either

$$\ln C = 0, \ln \sigma^2 = 0, \tag{5.11}$$

or

$$\ln C = 0, \ln \sigma^2 = \ln \hat{R}^2 \tag{5.12}$$

as the initial points, where $\hat{R}^2$ is the radius of the smallest sphere that contains all training examples in their original space. It is mentioned in (Keerthi, 2002) that the latter performs better. However, from experiments, we observed that both settings are appropriate. Thus, in Figures 2–4, we use (5.11) as the initial point in our implementation. Indeed most points on the $(\ln C, \ln \sigma^2)$ plane as initials lead to the convergence toward the local minimum which we would like to identify. The only exception is that when initial $\ln C$ and $\ln \sigma^2$ are too large or too small, the procedure goes toward other local minimum. Another reason to avoid larger initial $\ln C$ and $\ln \sigma^2$ is that methods used to solve each individual SVM usually take more time on such parameters.

For the stopping condition, (Keerthi, 2002) considers

$$| f(x^{k+1}) - f(x^k) | \leq 10^{-5} f(x^k).$$

However, the value of $f(x^k)$ ranges too much to be applicable to our procedure. A large $f(x^k)$ causes early stop of our procedure, whereas a small $f(x^k)$ easily leads to too many iterations. Instead, we check the gradient, as frequently used in many other iterative procedures. Following (Lin and Moré, 1999), we use

$$\frac{\|\nabla f(x^k)\|}{\|\nabla f(x^0)\|} \leq 10^{-3} \text{ or } \|\nabla f(x^k)\| \leq 10^{-3},$$

where x^0 is the initial solution. Usually the relative condition is achieved first.

5.4 Implementation

We use LIBSVM (Chang and Lin, 2001b), which implements a decomposition method, to calculate $\|w\|^2$, $\|\tilde{w}\|^2$, R^2, and $\tilde{R}^2$. The quasi-Newton implementation is written in Python by modifying the optimization subroutine by Travis Oliphant (available in Scientific Python at `http://www.scipy.org`). Tools such as numerical python are employed so it is easy to conduct matrix and vector operations. The python program is then linked to LIBSVM through its python interface. Some advantages of using python include the powerful data preprocessing and the easy link with other graphic tools. The computational experiments for this section were done on a Pentium III-1000 with 1024MB RAM using the gcc compiler.

We keep all the default settings of LIBSVM except using a smaller stopping tolerance 10^{-6} (default 10^{-3}) and increasing the cache size. It was pointed out in (Keerthi, 2002) that near the minimizer, $\|\nabla f\|$ is small so the error associated with finding $\|w\|^2, \|\tilde{w}\|^2, R^2$ or $\tilde{R}^2$ may strongly affect the search direction. We have the same observation so decide to use a smaller stopping tolerance.

For our current implementation, solving each of $\|w\|^2, \|\tilde{w}\|^2, R^2$ or $\tilde{R}^2$ is considered an independent problem. In the future, $\|w\|^2$ and R^2 ($\|\tilde{w}\|^2$ and $\tilde{R}^2$) should be considered together. How to effectively pass information under one given parameter set to another is also worthy of investigation.

5.5 Experiments

We compare the quasi-Newton implementation for L1- and L2-SVM. For L1-SVM, we use (3.3) with $\Delta = 1$. The same five problems are considered. To demonstrate the viability for solving large problems, we include the problem ijcnn1 which has 49,990 training and 91,701 testing samples in two classes. It is from the first problem of IJCNN challenge 2001 (Prokhorov, 2001). Note that we use the winner's transformation of raw data (Chang and Lin, 2001a).

Tables 5.1 and 5.2 present the results using different initial points: $(0, 0)$ and $(-3, -3)$. We list the number of function and gradient evaluations, the optimal $(\ln C, \ln \sigma^2)$, and the test accuracy. Results for L2-SVM are consistent with Table 1 of (Keerthi, 2002). Note that the number of gradient evaluations is the same as the number of quasi-Newton iterations. Hence, it can be seen that the average number of line searches in each iteration is very close to one. For problems image, splice, and tree, using L1-SVM, the algorithm reaches a point with $\ln C = 10$. At that point there are no bounded support vectors so we can set the resulting $\ln C$ to the largest element of the dual variable α without affecting the model produced. Actually in final iterations for these problems, there are already no bounded support vectors in the models. But unlike the grid search in Section 4, where we can stop increasing C for the same σ^2, here C and σ^2 can both change in a single iteration and must be considered together. We have not been able to develop early stopping criteria for such a situation so decided to let the algorithm continue until it reaches the boundary. Thus, for image and waveform, L1-SVM takes more iterations than L2-SVM. On the other hand, for splice, L2-SVM also reaches $\ln C = 10$. Overall except this difference, in terms of accuracy as well as computational cost, the bound for L1-SVM is competitive with that for L2-SVM.

For the large problem ijcnn1, RM bounds for both L1- and L2-SVM reach points with error rate around 0.0291 and 0.0217, respectively. Unfortunately, this is worse than 0.0141 by $C = 16$ and $\sigma^2 = 0.125$ using cross-validation (Chang and Lin, 2001a). In addition, the number of support vectors is very different. There are about 17,000 support vectors comparing to 3370 in (Chang and Lin, 2001a). However, the total computational time is around 26,000 seconds (using the default 10^{-3} as the stopping tolerance of LIBSVM), much shorter than doing cross-validation.

When $(-3, -3)$ is used as the initial solution, in Table 5.2, for splice and waveform, the algorithm does not converge to desired points. This seems to show that for these problems, $(-3, -3)$ is already too far away from the good region. For L1 bound (3.3), the vertical contour lines in the bottom of Figures 2–5, shows that the bound is almost independent of σ when $\sigma^2 \to 0$. In Appendix D, we further show that when $\sigma^2 \to 0$, the algorithm tends to find a too large or too small C. Thus, we should avoid a too small σ^2 as the initial solution.

To further compare bounds for L1- and L2-SVM, in Figures 2–5, we present contour plots of tree, waveform, and splice, with searching paths on them. Note that in the L1 case of Figure 3, the final solution is projected from $(8.87, 1.38)$ to $(1.25, 1.38)$ because there is already no bounded support vectors. Figure 5 is an example that the algorithm does not converge to the desired region, where the initial solution is too far away from the good region.

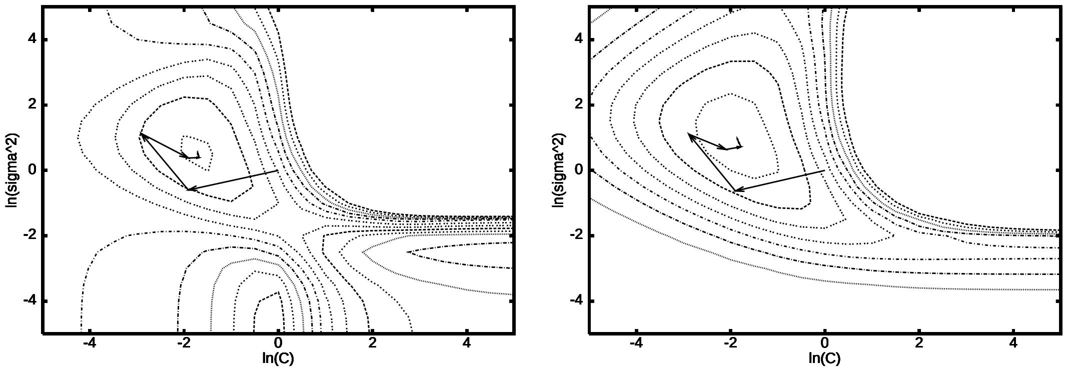


Figure 2: Contour plots and searching paths of tree (left: L1 bound, right: L2 bound). The initial solution is $(0, 0)$.

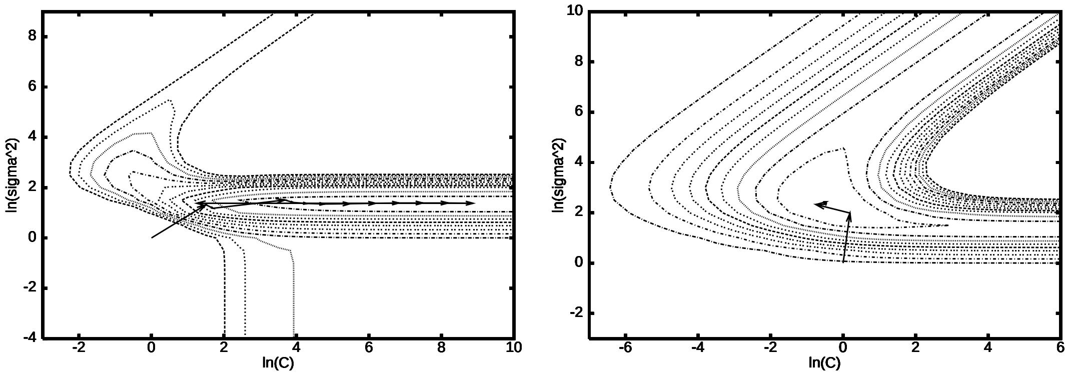


Figure 3: Contour plots and searching paths of waveform (left: L1 bound, right: L2 bound). The initial solution is $(0, 0)$. The final solution for the L1 bound is projected from $(8.87, 1.38)$ to $(1.25, 1.38)$.

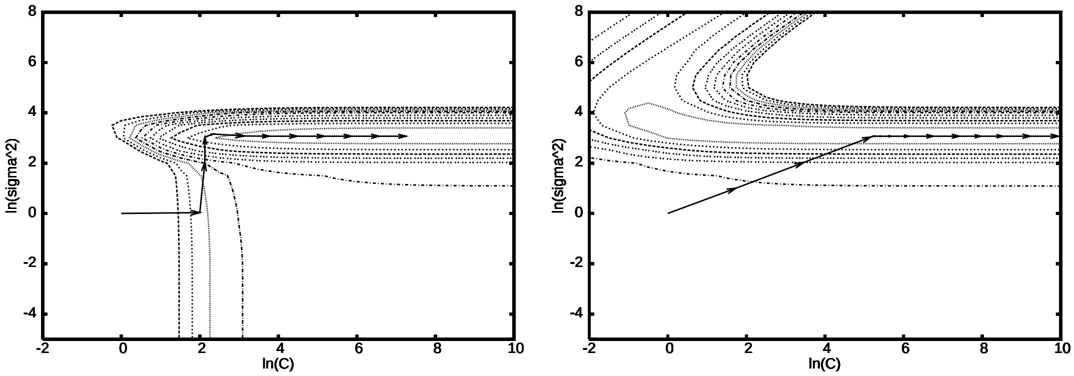


Figure 4: Contour plots and searching paths of splice (left: L1 bound, and right: L2 bound). The initial solution is $(0, 0)$.

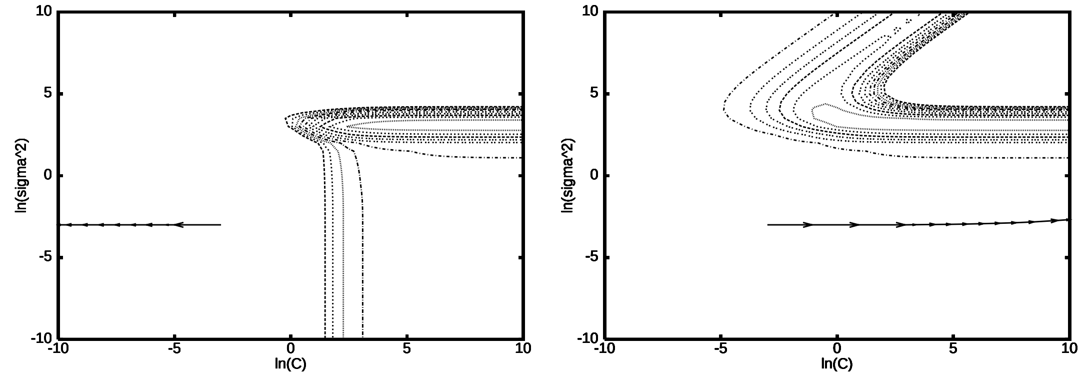


Figure 5: Contour plots and searching paths of splice (left: L1 bound, and right: L2 bound). The initial solution is $(-3, -3)$. Quasi-Newton iterations fail to converge to the desired regions.

	L1				L2			
	#fun	#grad	$(\ln C, \ln \sigma^2)$	accuracy	#fun	#grad	$(\ln C, \ln \sigma^2)$	accuracy
banana	9	6	(-0.83,-1.69)	88.96	8	5	(-0.91,-1.64)	88.53
image	17	13	(2.99,-1.20)	96.24	11	6	(0.44,-0.29)	97.03
splice	13	12	(1.09,3.07)	89.84	21	19	(10.00,3.07)	89.84
tree	8	8	(-1.71,0.53)	86.50	8	8	(-1.86,0.97)	86.54
waveform	16	13	(1.20,1.38)	88.57	8	7	(-0.50,2.36)	89.83
ijcnn1	9	9	(2.06,-4.05)	97.09	7	7	(-0.10,-2.58)	97.83

Table 5.1: The results of the experiments using RM bounds with $(\ln C, \ln \sigma^2) = (0, 0)$ as the initial solution. The bound used for L1-SVM is (3.3) with $\Delta = 1$. #fun and #grad represent the number of function and gradient evaluations, respectively.

	L1				L2			
	#fun	#grad	$(\ln C, \ln \sigma^2)$	accuracy	#fun	#grad	$(\ln C, \ln \sigma^2)$	accuracy
banana	8	7	(-0.83,-1.69)	88.96	8	7	(-0.91,-1.64)	88.53
image	25	23	(2.99,-1.20)	96.24	8	8	(0.44, -0.29)	97.03
splice	10	10	(-10.00,-3.00)	52.00	14	14	(10.00,-2.68)	56.92
tree	13	11	(-1.71,0.53)	86.50	7	7	(-1.86, 0.97)	86.54
waveform	10	10	(-10.00,-3.00)	67.07	1	1	(-3.00,-3.00)	67.07
ijcnn1	13	12	(2.06,-4.05)	97.09	6	6	(-0.10,-2.58)	97.83

Table 5.2: The results of the experiments using RM bounds with $(\ln C, \ln \sigma^2) = (-3, -3)$ as the initial solution. The bound used for L1-SVM is (3.3) with $\Delta = 1$. #fun and #grad represent the number of function and gradient evaluations, respectively.

6 Discussion and Conclusion

6.1 SVM Model Selection Software

People frequently say that an advantage of SVM over Neural Networks is that SVM solves a convex problem. However, this is true only in the sense that parameters are given. Once model selection is considered, we still deal with difficult non-convex problems.

Therefore, it is fair to say that SVM elegantly separate things to different stages where the innermost one solves simple convex problems. Easy-to-use implementations for solving these convex problems have been available. However, if we joggle everything together, the SVM software will be more complicated. In particular, there may be a need to provide different model selection strategies. For small problems where cross-validation is affordable, a complete search may be more reliable. But for large problems using heuristic bounds is essential. Whether users can adapt to use these different techniques and conduct appropriate model selection remains to be a main challenge for SVM software development.

6.2 Remaining Implementation Issues

As the search direction of quasi-Newton method depends on accurately solving each SVM dual, we have used a smaller stopping tolerance for LIBSVM. However, this also lengthen the total computational time. The requirement of high accuracy on solving SVM duals happens in final iterations of the quasi-Newton procedure. How to effectively adapt the stopping tolerance for SVM duals in the quasi-Newton procedure is a future issue.

6.3 Heuristic Bounds for Other Problems

In this paper we only discuss two-class classification SVM. However, it is possible to extend our results and apply similar ideas on multi-class classification problems and SVM regression (Chang and Lin, 2003). Moreover, the heuristic bounds for L1-SVM developed in this paper may be useful for tuning many hyperparameters, such as doing automatic relevance determination of input features.

Acknowledgments

This work was supported in part by the National Science Council of Taiwan via the grant NSC 90-2213-E-002-111. We thank S. Sathiya Keerthi, Jorge Moré, and Olivier Chapelle for very useful discussion. In particular, Olivier Chapelle helped to remove an assumption of Theorem 1.

References

Bailey, R. R., Pettit, E. J., Borochoff, R. T., Manry, M. T., and Jiang, X. (1993). Automatic recognition of usgs land use/cover categories using statistical and neural networks classifiers. In *SPIE OE/Aerospace and Remote Sensing*, Bellingham, WA. SPIE.

Bonnans, J. F. and Shapiro, A. (1998). Optimization problems with perturbations: a guided tour. *SIAM Review*, 40(2):228–264.

Chang, C.-C. and Lin, C.-J. (2001a). IJCNN 2001 challenge: Generalization ability and text decoding. In *Proceedings of IJCNN*. IEEE.

Chang, C.-C. and Lin, C.-J. (2001b). *LIBSVM: a library for support vector machines.* Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Chang, M.-W. and Lin, C.-J. (2003). Properties of the dual svm solution as a function of parameters. Technical report, Department of Computer Science and Information Engineering, National Taiwan University.

Chapelle, O., Vapnik, V., Bousquet, O., and Mukherjee, S. (2002). Choosing multiple parameters for support vector machines. *Machine Learning*, 46:131–159.

Cristianini, N., Campbell, C., and Shawe-Taylor, J. (1999). Dynamically adapting kernels in support vector machines. In Kearns, M. S., Solla, S. A., and Cohn, D. A., editors, *Advances in Neural Information Processing Systems*, volume 11, pages 204–210, Cambridge, MA. MIT Press.

Duan, K., Keerthi, S. S., and Poo, A. N. (2003). Evaluation of simple performance measures for tuning SVM hyperparameters. *Neurocomputing*, 51:41–59.

Elisseeff, A. and Pontil, M. (2002). Leave-one-out error and stability of learning algorithms with applications. In Suykens, J., , Horvath, G., Basu, S., Micchelli, C., and Vandewalle, J., editors, *Learning Theory and Practice*, NATO ASI Series. IOS Press, Amsterdam; Washington, DC.

Hastie, T., Tibshirani, R., and Friedman, J. (2002). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer Series in Statistics. Springer Book.

Jaakkola, T. S. and Haussler, D. (1999). Probabilistic kernel regression models. In *Proceedings of the 1999 Conference on AI and Statistics.* Society for Artificial Intelligence in Statistics.

Joachims, T. (1998). Making large-scale SVM learning practical. In Schölkopf, B., Burges, C. J. C., and Smola, A. J., editors, *Advances in Kernel Methods - Support Vector Learning*, Cambridge, MA. MIT Press.

Joachims, T. (2000). Estimating the generalization performance of a SVM efficiently. In *Proceedings of the International Conference on Machine Learning*, San Francisco. Morgan Kaufman.

Keerthi, S. S. (2002). Efficient tuning of SVM hyperparameters using radius/margin bound and iterative algorithms. *IEEE Transactions on Neural Networks*, 13:1225–1229.

Keerthi, S. S. and Lin, C.-J. (2003). Asymptotic behaviors of support vector machines with Gaussian kernel. *Neural Computation*, 15(7). To appear.

Li, D.-H. and Fukushima, M. (2001). On the global convergence of the BFGS method for nonconvex unconstrained optimization problems. *SIAM Journal on Optimization*, 11(4):1054–1064.

Lin, C.-J. and Moré, J. J. (1999). Newton's method for large-scale bound constrained problems. *SIAM Journal on Optimization*, 9:1100–1127.

Micchelli, C. A. (1986). Interpolation of scattered data: distance matrices and conditionally positive definite functions. *Constructive Approximation*, 2:11–22.

Nocedal, J. and Wright, S. J. (1999). *Numerical Optimization*. Springer-Verlag, New York, NY.

Prokhorov, D. (2001). IJCNN 2001 neural network competition. Slide presentation in IJCNN'01, Ford Research Laboratory. `http://www.geocities.com/ijcnn/nnc_ijcnn01.pdf` .

Rätsch, G. (1999). Benchmark data sets. Available at `http://ida.first.gmd.de/~raetsch/data/benchmarks.htm`.

Schölkopf, B., Burges, C., and Vapnik, V. (1995). Extracting support data for a given task. In Fayyad, U. M. and Uthurusamy, R., editors, *Proceedings of the First International Conference on Knowledge Discovery and Data Mining*, Menlo Park, CA. AAAI Press.

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

Vapnik, V. and Chapelle, O. (2000). Bounds on error expectation for support vector machines. *Neural Computation*, 12(9):2013–2036.

Wahba, G. (1998). Support vector machines, reproducing kernel Hilbert spaces, and randomized GACV. In Schölkopf, B., Burges, C. J. C., and Smola, A. J., editors, *Advances in Kernel Methods: Support Vector Learning*, pages 69–88. MIT Press.

Wahba, G., Lin, Y., and Zhang, H. (2000). Generalized approximate cross validation for support vector machines. In Smola, A., Bartlett, P., Schölkopf, B., and Schuurmans, D., editors, *Advances in Large Margin Classifiers*, pages 297–309. MIT Press.

Zhang, T. (2001). A leave-one-out cross validation bound for kernel methods with applications in learning. In *Proceedings of the 14th Annual Conference on*

Computational Learning Theory, volume 2111, pages 427–443, Berlin, NY. Springer.

A Proof of Theorem 1

We introduce some required preliminaries before the proof. In fact, (1.3) and (1.11) are dual form of the following problem:

$$\begin{array}{ll} \min\limits_{a,R} & R^2 \\ \text{subject to} & R^2 - \|\phi(x_i) - a\|^2 \geq 0, i = 1, \ldots, l, \end{array}$$

where the vector a is the center of the sphere. From the KKT condition, we have

$$a = \sum_{j=1}^{l} \beta_j \phi(x_j), \tag{A.1}$$

and if $\beta_i \neq 0$,

$$R^2 - \|\phi(x_i) - a\|^2 = 0. \tag{A.2}$$

We then need a technical lemma:

Lemma 2 *For any kernel the vector β in dual form of R^2 satisfies $\beta^T\beta \leq 1/2$.*

Proof. It suffices to show that $\beta_i \leq 1/2$ for $i = 1, \ldots, l$. If it is true, with $\sum_{i=1}^{l} \beta_i = 1$ and $\beta_i \geq 0$ for $i = 1, \ldots, l$, we have $\beta^T\beta \leq 1/2$.

Assume that $\beta_i \neq 0$ for some i. (A.1) and (A.2) imply

$$\begin{aligned} R^2 &= \|\phi(x_i) - a\|^2 \\ &= \|\phi(x_i) - \sum_{j=1}^{l} \beta_j \phi(x_j)\|^2 \\ &= (1-\beta_i)^2 \|\phi(x_i) - \sum_{j \neq i} \frac{\beta_j}{1-\beta_i} \phi(x_j)\|^2 \\ &\equiv (1-\beta_i)^2 \|\phi(x_i) - z\|^2 \end{aligned}$$

Since $\beta_j \geq 0$ and $\sum_{j \neq i} \frac{\beta_j}{1-\beta_i} = 1$, z is in the convex hull of $\phi(x_j)$, $j \neq i$. Thus,

$$R^2 \leq (1-\beta_i)^2 D^2 = 4(1-\beta_i)^2 R^2,$$

so $\beta_i \leq 1/2$. $\square$

We now return to the proof of Theorem 1. First, we have

$$\beta^T(K + \frac{I}{C})\beta \geq \tilde{\beta}^T(K + \frac{I}{C})\tilde{\beta} \geq \beta^T K\beta + \frac{\tilde{\beta}\tilde{\beta}}{C},$$

where β and $\tilde{\beta}$ are optimal solutions of (1.3) and (1.11), respectively. Because $K_{ii} = 1$ and $K_{ij} \geq 0$,

$$\tilde{\beta}^T K\tilde{\beta} \geq \beta^T K\beta \geq \beta^T\beta \geq \tilde{\beta}^T\tilde{\beta}.$$

By the differentiability discussed in Section 5.1, we consider the partial derivative of $\tilde{R}^2/R^2 + \frac{0.5}{C}$.

$$\begin{aligned}
&\frac{\partial}{\partial C}\left(\frac{\tilde{R}^2}{R^2 + \frac{0.5}{C}}\right) \\
= \;& ((R^2 + \frac{0.5}{C})\frac{\partial \tilde{R}^2}{\partial C} - (\frac{-0.5}{C^2})\tilde{R}^2)/(R^2 + \frac{0.5}{C})^2.
\end{aligned}$$

We have

$$\begin{aligned}
& (R^2 + \frac{0.5}{C})\frac{\partial \tilde{R}^2}{\partial C} - (\frac{-0.5}{C^2})\tilde{R}^2 \\
= \;& R^2\frac{\partial \tilde{R}^2}{\partial C} + \frac{0.5}{C^2}(C\frac{\partial \tilde{R}^2}{\partial C} + \tilde{R}^2) \\
= \;& R^2\frac{\partial \tilde{R}^2}{\partial C} + \frac{0.5}{C^2}(\frac{-1 + \tilde{\beta}^T\tilde{\beta}}{C} + 1 + \frac{1}{C} - \tilde{\beta}^T(K + \frac{I}{C})\tilde{\beta}) \\
= \;& (1 - \beta^T K\beta)(\frac{-1}{C^2})(1 - \tilde{\beta}^T\tilde{\beta}) + \frac{0.5}{C^2}(1 - \tilde{\beta}^T K\tilde{\beta}) \\
= \;& \frac{1}{C^2}(\frac{-1}{2} + \beta^T K\beta + \tilde{\beta}^T\tilde{\beta} - (\beta^T K\beta)(\tilde{\beta}^T\tilde{\beta}) - 0.5\tilde{\beta}^T K\tilde{\beta}) \\
\leq \;& \frac{1}{C^2}(\frac{-1}{2} + 0.5\beta^T K\beta + (1 - \beta^T K\beta)\tilde{\beta}^T\tilde{\beta}) \\
= \;& \frac{1}{C^2}((\frac{-1}{2} + \tilde{\beta}^T\tilde{\beta})(1 - \beta^T K\beta)) \\
\leq \;& 0,
\end{aligned}$$

since $\tilde{\beta}^T\tilde{\beta} \leq 1/2$ from Lemma 2. Therefore, with (2.6), we conclude that $\tilde{R}^2/(R^2 + \frac{0.5}{C})$ monotonically decreases to 1 as $C \to \infty$ and

$$(R^2 + \frac{0.5}{C})\|\tilde{w}\|^2 \leq \tilde{R}^2\|\tilde{w}\|^2, \forall C.$$

B Proof of Lemma 1

Define

$$
\begin{aligned}
\Lambda_p &\equiv \{\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_l) \mid \lambda_p = -1, \sum_{i=1}^{l} \lambda_i = 0, \alpha_i^0 + y_i y_p \alpha_p^0 \lambda_i \geq 0\}, \\
\Lambda_p^+ &\equiv \{\lambda \in \Lambda_p \mid \lambda_i \geq 0, \forall i \neq p\}.
\end{aligned}
$$

For any $\lambda \in \Lambda_p^+$,

$$
(\sum_{i=1}^{l} \lambda_i \phi(x_i))^2 = \|\phi(x)_p - \sum_{i \notin p} \lambda_i \phi(x_i)\|^2 \leq D^2, \tag{B.1}
$$

where D is the diameter of the smallest ball containing all training data.

Select $\lambda^* \in \Lambda_p^+$ by

$$
\lambda_i^* = \begin{cases} -1 & \text{if } i = p, \\ \frac{1}{l_{sv1}-1} & \text{if } y_i = y_p = 1, \\ \frac{1}{l_{sv2}-1} & \text{if } y_i = y_p = -1, \\ 0 & \text{if } y_i \neq y_p. \end{cases}
$$

Now

$$
\tilde{S}_p^2 \equiv \min\{(\sum_{i=1}^{l} \lambda_i \tilde{z}_i)^2 \mid \lambda_p = -1, \sum_{i=1}^{l} \lambda_i = 0, \alpha_i^0 + y_i y_p \alpha_p^0 \lambda_i \geq 0\},
$$

where $\tilde{z}_i$ is defined in (2.2).

Since Λ_p^+ is a subset of Λ_p, $\lambda^* \in \Lambda_p$. With (B.1), we obtain

$$
\begin{aligned}
\tilde{S}_p^2 &\leq (\sum_{i=1}^{l} \lambda_i^* \tilde{z}_i)^2 \\
&= (\sum_{i=1}^{l} \lambda_i^* \phi(x_i))^2 + (\sum_{i=1}^{l} \frac{\lambda_i^* e_i}{\sqrt{C}})^2 \\
&\leq D^2 + \frac{1}{C} + \begin{cases} \frac{1}{(l_{sv1}-1)C} & \text{if } y_p = 1, \\ \frac{1}{(l_{sv2}-1)C} & \text{if } y_p = -1. \end{cases}
\end{aligned}
$$

C Proof of (2.15)-(2.17)

Consider the following problem:

$$
\begin{aligned}
\max_{\alpha} \quad & e^T \alpha - \frac{\alpha^T \alpha}{2C} \\
\text{subject to} \quad & 0 \leq \alpha_i, i = 1, \ldots, l, \\
& y^T \alpha = 0.
\end{aligned} \tag{C.1}
$$

We can easily see

$$\bar{\alpha}_i = \begin{cases} \frac{2l_2}{l}C & \text{if } y_i = 1, \\ \frac{2l_1}{l}C & \text{if } y_i = -1, \end{cases}$$

is an optimal solution. It satisfies the Karush-Kuhn-Tucker (KKT) condition that there is a number

$$b = \frac{l_1 - l_2}{l} \text{ such that } \frac{\alpha_i}{C} - 1 + by_i = 0, i = 1, \dots, l.$$

Under any fixed C, we assume $\alpha(\sigma)$ is the optimal solution of the dual L2-SVM (1.8) and $Q(\sigma)_{ij} = e^{-\|x_i - x_j\|^2/(2\sigma^2)}$. As (C.1) and (1.8) have the same feasible set and $Q(\sigma)$ is positive semi-definite,

$$\begin{aligned} & e^T\bar{\alpha} - \frac{\bar{\alpha}^T\bar{\alpha}}{2C} \\ \geq \;& e^T\alpha(\sigma) - \frac{1}{2}\alpha(\sigma)^T(Q(\sigma) + \frac{I}{C})\alpha(\sigma) \\ \geq \;& e^T\bar{\alpha} - \frac{1}{2}\bar{\alpha}^T(Q + \frac{I}{C})\bar{\alpha}. \end{aligned}$$

When $\sigma \to \infty$, $Q(\sigma) \to yy^T$ so with $y^T\bar{\alpha} = 0$,

$$\lim_{\sigma\to\infty} \|\tilde{w}\|^2 = 2(e^T\bar{\alpha} - \frac{\bar{\alpha}^T\bar{\alpha}}{2C}) = \frac{4Cl_1l_2}{l}.$$

Similarly, we can show

$$\lim_{\sigma\to\infty} \tilde{R}^2 = \frac{1}{C}(1 - \frac{1}{l}).$$

Thus, if $l \gg 1$, when $\sigma \to \infty$,

$$\tilde{R}^2\|\tilde{w}\|^2 \approx \frac{4l_1l_2}{l}.$$

We can also prove that when $\sigma \to \infty$, $R^2 \to 0$. Then, (2.16) and (2.17) follow.

D Limit Behavior of (3.3) When $\sigma^2 \to 0$

From (Keerthi and Lin, 2003), the optimal solution of (1.6) when $\sigma^2 \to 0$ is:

$$\alpha^- = \begin{cases} C_{lim} & \text{if } C \geq C_{lim} \\ C & \text{if } C < C_{lim} \end{cases}, \quad \alpha^+ = \begin{cases} \frac{2l_2}{l} & \text{if } C \geq C_{lim} \\ \frac{l_2C}{l_1} & \text{if } C < C_{lim} \end{cases}, \tag{D.1}$$

where $\alpha_i = \alpha^+$ if $y_i = +1$, $\alpha_i = \alpha^-$ if $y_i = -1$, $C_{lim} = \frac{2l_1}{l}$, and $l_1 > l_2 + 1 > 2$. Here l_1, l_2 is number of training examples which $y_i = +1$ and $y_i = -1$ respectively.

It is easy to show that when $\sigma^2 \to 0$, the optimal solution of (1.3) is $\beta = \frac{e}{l}$ where $e \in R^l$ is a vector with all its elements 1. Thus, when $\sigma^2 \to 0$, (3.3) becomes

$$(\frac{l-1}{l} + \frac{\Delta}{C})(\frac{4l_1l_2}{l}) \qquad \text{if } C \geq C_{lim}, \tag{D.2}$$

$$(\frac{l-1}{l} + \frac{\Delta}{C})(4l_2C - l_2C^2 - \frac{l_2^2C^2}{l_1}) \qquad \text{if } C < C_{lim}. \tag{D.3}$$

Thus, when $C < C_{lim}$, (3.3) is a concave function on C; when $C \geq C_{lim}$, (3.3) is a decreasing function on C. This explains the phenomenon that when $\sigma^2 \to 0$, the algorithm will find a too large or too small C in Section 5.

E Proof of Theorem 2

If x and x' are global optima of $f_1(x)$ and $f_2(x)$, respectively, using $g > 0$,

$$f_2(x')g(x') \leq f_2(x)g(x), \text{ and}$$
$$f_1(x)g(x) \leq f_1(x')g(x').$$

Therefore,

$$f_2(x')/f_1(x') \leq f_2(x)/f_1(x).$$

With the assumption that f_1/f_2 is a strictly decreasing function, $x' \leq x$.

F Differentiability of L1-SVM RM Bound: an Example

We will demonstrate four things by the following example:

1. Original modified RM bound is not differentiable.

2. Our RM bound is differentiable.

3. Our RM bound is not twice differentiable.

4. Our RM bound is not a convex function with respect to C and σ^2.

Given two different data points in two classes, if the RBF kernel is used, the dual problem has the following form:

$$\begin{aligned}
\min_{\alpha_1,\alpha_2} \quad & \frac{1}{2}\begin{bmatrix}\alpha_1 & \alpha_2\end{bmatrix}\begin{bmatrix}1 & -a\\ -a & 1\end{bmatrix}\begin{bmatrix}\alpha_1\\ \alpha_2\end{bmatrix} - e^T\alpha\\
\text{subject to} \quad & \alpha_1 - \alpha_2 = 0,\\
& 0 \le \alpha_1, \alpha_2 \le C,
\end{aligned}$$

where $0 < a < 1$.

At the optimal solution,

$$\alpha_1 = \alpha_2 = \begin{cases} \frac{1}{1-a} & \text{if } C \ge \frac{1}{1-a},\\ C & \text{if } 0 \le C \le \frac{1}{1-a}, \end{cases}$$

$$\xi_1 = \xi_2 = \begin{cases} 0 & \text{if } C \ge \frac{1}{1-a},\\ 1-(1-a)C & \text{if } 0 \le C \le \frac{1}{1-a}. \end{cases}$$

We have

$$e^T\alpha = \begin{cases} \frac{2}{1-a} & \text{if } C \ge \frac{1}{1-a},\\ 2C & \text{if } 0 \le C \le \frac{1}{1-a}, \end{cases}$$

$$e^T\xi = \begin{cases} 0, & \text{if } C \ge \frac{1}{1-a},\\ 2-2(1-a)C, & \text{if } 0 \le C \le \frac{1}{1-a}. \end{cases}$$

Therefore, $e^T\alpha$ is not differentiable at $C = 1/(1-a)$. We then claim that $(R^2 + \frac{1}{C})e^T\alpha$ is not differentiable. Otherwise, with the fact that $R^2 + 1/C$ is differentiable, we imply that $e^T\alpha$ is differentiable, a contradiction.

Now the optimization problem for finding R^2 is:

$$\begin{aligned}
\max_{\beta_1,\beta_2} \quad & e^T\beta - \frac{1}{2}\begin{bmatrix}\beta_1 & \beta_2\end{bmatrix}\begin{bmatrix}1 & a\\ a & 1\end{bmatrix}\begin{bmatrix}\beta_1\\ \beta_2\end{bmatrix}\\
\text{subject to} \quad & \beta_1 + \beta_2 = 1,\\
& 0 \le \beta_1, \beta_2.
\end{aligned}$$

Clearly,

$$\beta_1 = \beta_2 = \frac{1}{2}, R^2 = 1 - \frac{1}{2}(1+a) = \frac{1-a}{2}. \tag{F.1}$$

Hence, (3.1) becomes

$$R^2e^T\alpha + e^T\xi = \begin{cases} 1 & \text{if } C \ge \frac{1}{1-a},\\ 2-(1-a)C & \text{if } 0 \le C \le \frac{1}{1-a}. \end{cases}$$

Therefore,

$$\frac{\partial(R^2e^T\alpha+e^T\xi)}{\partial C}=\begin{cases}0 & \text{if } C>\frac{1}{1-a},\\ -(1-a) & \text{if } 0\le C<\frac{1}{1-a},\end{cases}$$

so $R^2e^T\alpha+e^T\xi$ is also not differentiable. We can further show that the original modified RM bound, $D^2e^T\alpha+\sum_{i=1}^l\xi_i$, is also not differentiable:

$$D^2e^T\alpha+e^T\xi=\begin{cases}4 & \text{if } C\ge\frac{1}{1-a},\\ 2+2(1-a)C & \text{if } 0\le C\le\frac{1}{1-a},\end{cases}$$

and

$$\frac{\partial(D^2e^T\alpha+e^T\xi)}{\partial C}=\begin{cases}0 & \text{if } C>\frac{1}{1-a},\\ 2(1-a) & \text{if } 0\le C<\frac{1}{1-a}.\end{cases}$$

For our RM bound, we can easily see that it is partially differentiable:

$$e^T\alpha+C\sum\xi_i=\begin{cases}\frac{2}{1-a} & \text{if } C\ge\frac{1}{1-a},\\ 4C-2(1-a)C^2 & \text{if } 0\le C\le\frac{1}{1-a}.\end{cases} \tag{F.2}$$

Then

$$\lim_{C\to(\frac{1}{1-a})^-}\frac{\partial(e^T\alpha+C\sum\xi_i)}{\partial C}=0=\lim_{C\to(\frac{1}{1-a})^+}\frac{\partial(e^T\alpha+C\sum\xi_i)}{\partial C}.$$

With the continuity, our RM bound is differentiable. However,

$$(R^2+\frac{\Delta}{C})(\|w\|^2+2C\sum\xi_i)=\begin{cases}1+\frac{2\Delta}{(1-a)C} & \text{if } C\ge\frac{1}{1-a},\\ \left(\frac{1-a}{2}+\frac{\Delta}{C}\right)(4C-2(1-a)C^2) & \text{if } 0<C\le\frac{1}{1-a}.\end{cases}$$

The second derivative is

$$\frac{\partial^2\left(R^2+\frac{\Delta}{C}\right)(\|w\|^2+2C\sum\xi_i)}{\partial C\partial C}=\begin{cases}\frac{4\Delta}{1-a}C^{-3} & \text{if } C>\frac{1}{1-a},\\ -2(1-a)^2 & \text{if } 0<C<\frac{1}{1-a}.\end{cases}$$

That is, at $C=1/(1-a)$, the second derivative is not available.

Finally, for convexity, when $C\le 1/(1-a)$, (F.1) and (F.2) imply $f(C,\sigma^2)=4-(1-a)^2C^2$. When σ^2 is fixed, a is a constant and f is a concave function of C.

G RM Bound for L2-SVM is not Twice Differentiable: an Example

Given two points in one class and one point in the other, if the RBF kernel is used, the dual problem of L2-SVM is:

$$
\begin{aligned}
\min_{\alpha_1,\alpha_2,\alpha_3} \quad & \frac{1}{2}\begin{bmatrix}\alpha_1 & \alpha_2 & \alpha_3\end{bmatrix}\begin{bmatrix}1+\frac{1}{C} & a & -b\\ a & 1+\frac{1}{C} & -c\\ -b & -c & 1+\frac{1}{C}\end{bmatrix}\begin{bmatrix}\alpha_1\\ \alpha_2\\ \alpha_3\end{bmatrix} - e^T\alpha \\
\text{subject to} \quad & \alpha_1+\alpha_2=\alpha_3,\\
& 0\le \alpha_1,\alpha_2,\alpha_3,
\end{aligned}
$$

and $\tilde{R}^2$ is the objective value of

$$
\begin{aligned}
\max_{\beta_1,\beta_2,\beta_3} \quad & 1+\frac{1}{C} - \begin{bmatrix}\beta_1 & \beta_2 & \beta_3\end{bmatrix}\begin{bmatrix}1+\frac{1}{C} & a & b\\ a & 1+\frac{1}{C} & c\\ b & c & 1+\frac{1}{C}\end{bmatrix}\begin{bmatrix}\beta_1\\ \beta_2\\ \beta_3\end{bmatrix} \\
\text{subject to} \quad & \beta_1+\beta_2+\beta_3=1,\\
& 0\le \beta_1,\beta_2,\beta_3,
\end{aligned}
$$

where $0 < a, b, c < 1$.

Consider a problem satisfying the following conditions:

$$
\begin{aligned}
a+c-b-1 &> 0,\\
c &> b,\\
a &> b.
\end{aligned}
\tag{G.1}
$$

Then, the optimal α and β *near* $C = 1/(a+c-b-1)$ are

$$
\alpha = \begin{cases}
\begin{bmatrix}\frac{2(1+\frac{1}{C}-a+b-c)}{\Delta} & \frac{2(1+\frac{1}{C}-a-b+c)}{\Delta} & \frac{4(1+\frac{1}{C}-a)}{\Delta}\end{bmatrix}^T & \text{if } C \le \frac{1}{a+c-b-1},\\
\begin{bmatrix}0 & \frac{1}{1+\frac{1}{C}-c} & \frac{1}{1+\frac{1}{C}-c}\end{bmatrix}^T & \text{if } C \ge \frac{1}{a+c-b-1},
\end{cases}
$$

and

$$
\beta = \begin{cases}
\begin{bmatrix}\frac{(1+\frac{1}{C}-c)(1+\frac{1}{C}-a-b+c)}{\Delta} & \frac{(1+\frac{1}{C}-b)(1+\frac{1}{C}-a+b-c)}{\Delta} & \frac{(1+\frac{1}{C}-a)(1+\frac{1}{C}+a-b-c)}{\Delta}\end{bmatrix}^T\\
\qquad \text{if } C \le \frac{1}{a+c-b-1},\\
\begin{bmatrix}\frac{1}{2} & 0 & \frac{1}{2}\end{bmatrix}^T\\
\qquad \text{if } C \ge \frac{1}{a+c-b-1},
\end{cases}
$$

where $\Delta = 3(1+\frac{1}{C})^2 - 2(a+b+c)(1+\frac{1}{C}) - (a^2+b^2+c^2) + (2ab+2bc+2ac)$.