ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, IL 60439

Noam Goldberg, Sven Leyffer, and Todd Munson

Mathematics and Computer Science Division

Preprint ANL/MCS-P2049-0212

June 11, 2012

This work was supported by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract DE-AC02-06CH11357.

A new perspective on convex relaxations of sparse SVM

Noam Goldberg * Sven Leyffer† Todd Munson‡

June 11, 2012

Abstract

This paper proposes a convex relaxation of a sparse support vector machine (SVM) based on the perspective relaxation of mixed-integer nonlinear programs. We seek to minimize the zero-norm of the hyperplane normal vector with a standard SVM hinge-loss penalty and extend our approach to a zero-one loss penalty. The relaxation that we propose is a second-order cone formulation that can be efficiently solved by standard conic optimization solvers. We compare the optimization properties and classification performance of the second-order cone formulation with previous sparse SVM formulations suggested in the literature.**Keywords:** SVM, second order cone optimization, sparsity

AMS-MSC2010: 90C90, 90C25.

1 Introduction

Given a dataset $(A, y) \in \mathbb{R}^{m\times n} \times \{-1, 1\}^m$ we consider binary classification by support vector machines (SVMs), computing a hyperplane $\left\{a \in \mathbb{R}^n \;\middle|\; w^\top a = b\right\}$ in order to classify any $x \in \mathbb{R}^n$ based on $\operatorname{sgn}(w^\top x - b)$. Letting $[k] = \{1, \ldots, k\}$, we seek a hyperplane to separate a subset of $\{i \in [m] \mid y_i = 1\}$ from a subset of $\{i \in [m] \mid y_i = -1\}$; a separator that (we hope) generalizes well for the test data. In the standard SVM formulation this is achieved by minimizing $||w||_2^2$ or, equivalently maximizing the margin of separation. Since generally the data is inseparable, one usually minimizes the sum of $||w||_2^2$ and the hinge-loss penalty $c\sum_{i=1}^m [1 - y_i(w^\top x + b)]$ for some $c \geq 0$. The hinge-loss, however, is only a surrogate for the quantity of interest: the number of misclassifications that is measured by the zero-one loss; see Höffgen et al. (1995) and the discussion and references in Bennett and Bredensteiner (1997). We denote the zero-norm of a vector $a \in \mathbb{R}^n$ by $||a||_0 = \{j \in [n] \mid a_j \neq 0\}$. Next we consider sparse SVM formulations that minimize the sum of $||w||_0$ and the hinge-loss with penalty.

Our formulations are based on the perspective reformulation of mixed-integer nonlinear programs (MINLPs), a nonlinear MINLP formulation and relaxation technique that may eliminate in some cases the need for big-M constants that may otherwise be needed for modeling indicator variables Günlük and Linderoth (2010). The weakness of big-M relaxations in the context of classifier ensemble mixed-integer programming formulations was demonstrated by Goldberg and Eckstein (2012).

2 Convex relaxations of sparse SVM

Chan et al. (2007) consider a sparse SVM formulation that applies a constraint $||w||_0 \leq r$ to the standard

*Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, noamgold@mcs.anl.gov.
†Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, leyffer@mcs.anl.gov.
‡Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, munson@mcs.anl.gov.

SVM formulation:

$$\min_{\xi,w,b} \{||w||_2 + c\,||\xi||_1 \mid Y(Aw+b) + \xi \geq \mathbb{1}\}, \tag{1}$$

where Y denotes the $m \times m$ diagonal matrix with diagonal y.

Chan et al. (2007) propose a quadratically constrained quadratic program (QCQP) and a semidefinite programming (SDP) as convex relaxations of a sparse SVM (SSVM) formulation. Both relaxations are inspired by and rely on the vector norm inequality $||a||_1 \leq \sqrt{||a||_0}\,||a||_2$. The QCQP-SSVM is

$$\begin{aligned}
\min_{\xi,w,b,t} \quad & t + c\sum_{i=1}^{m} \xi_i && \text{(2a)}\\
\text{s.t.} \quad & Y(Aw + \mathbb{1}b) + \xi \geq \mathbb{1} && \text{(2b)}\\
& ||w||_2^2 \leq t && \text{(2c)}\\
& ||w||_1^2 \leq rt && \text{(2d)}\\
& \xi \geq 0. && \text{(2e)}
\end{aligned}$$

This formulation is equivalent to replacing the standard SVM objective by

$$\min_{\xi,w,b} \left\{ \max\left\{ \frac{1}{r}\,||w||_1^2, ||w||_2^2 \right\} + c\sum_{i=1}^{m} \xi_i \right\}.$$

For the sake of brevity we omit the formulation of the SDP relaxation (SDP-SSVM) which can be found in Chan et al. (2007).

Guan et al. (2009) proposed a MINLP for optimally solving a closely related problem where $||w||_0$ is penalized in the objective instead of being subject to a hard constraint. Because of the intractability of the problem, however, they were limited to solving the problem only for small datasets. Although solving the discrete problem to optimality may better optimize the tradeoff of hinge-loss and $||w||_0$, tractable convex relaxations may suffice to improve on classification performance and/or sparsity; see for example Chan et al. (2007); Goldberg and Eckstein (2010). Recently, Tan et al. (2010) proposed an algorithm for sparse SVM to solve a continuous relaxation with an infinite number of constraints. Their method for solving that formulation is specialized for problems with a large number of features. Here we consider finite conic formulations that are closer to the approach Chan et al. (2007); these can be solved by standard conic optimization solvers. Further, with the advent of first-order methods for conic optimization Zhao et al. (2010) our models may apply with a large number of observations and potentially in online settings.

Alternatively, smooth approximations of $||w||_0$ Bradley and Mangasarian (1998); Weston et al. (2003) and LP-SVM Bradley and Mangasarian (2000); Fung and Mangasarian (2004) have been used for sparse classification. However, smooth approximation techniques give rise to nonconvex optimization problems requiring one to settle for local minima. On the other hand, LP-SVM and linear programming formulations in general are poor relaxations of the zero-norm minimization problem because of the required big-M constants. The sparsity however can still be controlled by setting larger hinge-loss penalties (i.e., larger values of c); see Goldberg and Eckstein (2012) for a detailed analysis of the setting where $A \in \{-1, 0, 1\}^{m\times n}$. In the following we consider a convex (conic) relaxation that avoids the use of big-M constants that may otherwise be needed to formulate $||w||_0$ within a mathematical program.

3 Combining the 2-norm and zero-norm penalties

We now consider a sparse SVM formulation that minimizes a linear combination of the two-norm and zero-norm of w:

$$\min_{\xi,w} \left\{ ||w||_2^2 + c\,||\xi||_1 + d\,||w||_0 \mid Y(Aw - \mathbb{1}b) + \xi \geq \mathbb{1} \right\}.$$

This problem can be motivated by test error (or generalization) bounds that appear in the literature, some given in terms of the margin of separation (the inverse of the 2-norm of w), some given in terms of sparsity, and some combining both; see, for example, Koltchinskii and Panchenko (2005).

Guan et al. (2009) considered this problem[1] and formulated it as a MINLP, using binary indicator variables $z \in \{0,1\}^n$ and auxiliary variables $u \in \mathbb{R}^n_+$

$$\min_{\xi,w,b,u,z} \quad \sum_{j=1}^{n} u_j + c\sum_{i=1}^{m} \xi_i + d\sum_{j=1}^{n} z_j \tag{3a}$$

$$\text{s.t.} \quad Y(Aw - \mathbb{1}b) + \xi \geq \mathbb{1} \tag{3b}$$

$$w_j^2 - z_j u_j \leq 0 \qquad j \in [n] \tag{3c}$$

$$\xi \geq 0, z \in \{0,1\}^n. \tag{3d}$$

The constraints (3c) ensure that

$$w_j^2 \begin{cases} \leq u_j & z_j = 1 \\ = 0 & z_j = 0. \end{cases}$$

We note that for sufficiently large values of c and d, the u_j terms of (3a) become negligible and an optimal solution of (3) is also optimal in

$$\min_{w,\xi} \{ ||w||_0 + \tilde{c}\,||\xi||_1 \;\mid\; Y(Aw - \mathbb{1}b) + \xi \geq \mathbb{1} \} \tag{4}$$

for $\tilde{c} = c/d$. Hence, (3) generalizes previously considered zero-norm minimization problems (e.g., Weston et al. (2003) and Amaldi and Kann (1998)), known to be $\mathcal{NP}$-hard, implying that (3a) is NP-hard.

In addition to the fact that this problem is a computationally challenging MINLP, another obstacle is that the left-hand side of (3c) is not convex. Further, even general nonlinear programming solvers have difficulty in computing local minima for continuous relaxations of (3) because the constraints (3c) violate constraint qualification. Consequently, to solve small instances of (3), Guan et al. (2009) resorted to replacing the constraints (3c) by big-M constraints of the form $|w_j| \leq M z_j$.

4 A second-order cone relaxation

As $u, z \geq 0$, constraint (3c) can be rewritten as a convex second-order cone constraint, for each $j \in [n]$,

$$||(2w_j, u_j - z_j)|| \leq u_j + z_j.$$

Let $\mathcal{Q}^n$ denote the n-dimensional second-order (Lorentz) cone; see Ben-Tal and Nemirovski (2001) for a definition and related results concerning second-order cone programming. Now, relaxing the variables and letting $z_j \in [0,1]$ in place of $z_j \in \{0,1\}$, a second-order cone relaxation of (3) in which we also replace the

[1] In their formulation, Guan et al. (2009) introduce an additional constraint for enforcing $||w||_0 \geq 1$. However, one may set c and d to enforce $||w||_0 \geq 1$ endogenously.

$d\,||w||_0$ penalty by a hard constraint for a given parameter $r \geq 0$ is

$$\min_{\xi,w,b,u,z} \quad \sum_{j=1}^{n} u_j + c\sum_{i=1}^{m} \xi_i \tag{5a}$$

$$\text{s.t.} \quad Y(Aw - \mathbb{1}b) + \xi \geq \mathbb{1} \tag{5b}$$

$$(2w_j, u_j - z_j, u_j + z_j) \in \mathcal{Q}^3 \qquad j \in [n] \tag{5c}$$

$$\sum_{j=1}^{n} z_j \leq r \tag{5d}$$

$$\xi \geq 0, z \in [0,1]^n. \tag{5e}$$

This is a convex optimization problem that can be solved efficiently and also rapidly in practice by specialized interior-point conic optimization solvers. We refer to this novel formulation as CQ-SSVM. Note that for a sufficiently large values of r, in particular for $r = n$, the problem reduces to the standard SVM problem (1).

5 Extending the formulation for the zero-one loss

We now consider a formulation that minimizes the zero-one loss in place of the standard SVM hinge-loss:

$$\min_{\xi,w,b} \left\{ ||w||_0 + c\,||\xi||_0 \;\middle|\; Y(Aw - \mathbb{1}b) + \xi \geq \mathbb{1} \right\}. \tag{6}$$

Similar formulations that attempt to minimize the zero-one loss in conjunction with penalizing $||w||_0$ have been considered in the context of SVMs and boosting Weston et al. (2003); Goldberg and Eckstein (2010).

Appending the corresponding perspective variables and constraints to (3) for each of the variables ξ_i, for $i \in [n]$, we may formulate this problem as a MINLP:

$$\min_{\xi,w,b,u,s,q,z} \quad \sum_{j=1}^{n}(u_j + dz_j) + \sum_{i=1}^{m}(s_i + cq_i) \tag{7a}$$

$$\text{s.t.} \quad Y(Aw - \mathbb{1}b) + \xi \geq \mathbb{1} \tag{7b}$$

$$w_j^2 - z_j u_j \leq 0 \qquad j \in [n] \tag{7c}$$

$$\xi_i^2 - q_i s_i \leq 0 \qquad i \in [m] \tag{7d}$$

$$\xi \geq 0, z \in \{0,1\}^n, q \in \{0,1\}^m. \tag{7e}$$

The following proposition establishes values of c and d for which an optimal solution of (7) is optimal to (6). It is assumed that the data is integer; however, note that every rational matrix can be scaled so that its entries are integer.

Proposition 1. *Suppose $A \in \mathbb{Z}^{m\times n}$, $A_{ij} \leq U$ for all i, j, and $c, d \geq U^{2m}m^{m+2}$. Then, for every $(w^*, b^*, \xi^*, u^*, s^*, q^*, z^*)$ that is optimal to* (7)*, (w^*, b^*, ξ^*) must be optimal to* (6)*.*

Proof. Clearly, (w^*, b^*, ξ^*) is feasible for (6). Also note that by optimality to (7) we have $u_j^* = {w_j^*}^2$ for $j \in [n]$, and $s_i^* = {\xi_i^*}^2$ for $i \in [m]$. Now, assume for the sake of deriving a contradiction some $(\bar{w}, \bar{\xi})$ that is feasible for (6) (and hence also to (7b)) with $||\bar{w}||_0 + c\left|\left|\bar{\xi}\right|\right|_0 < ||w^*||_0 + c\,||\xi^*||_0$. Now, the system of inequalities (7b), for $i \in [m]$, with $\xi_i = 0$ fixed for i with $\bar{\xi}_i = 0$, and $w_j = 0$ fixed for j with $\bar{w}_j = 0$, has a basic feasible solution. Let B denote a submatrix of $\begin{pmatrix} Y \begin{pmatrix} A & \mathbb{1} \end{pmatrix} & I \end{pmatrix}$ that forms a basis. This basis has a corresponding set of active inequalities given by $B \begin{pmatrix} \hat{w} & b & \hat{\xi} \end{pmatrix}^\top = \mathbb{1}$, for some $\hat{w} \in \mathbb{R}^k$ and $\hat{\xi} \in \mathbb{R}^\ell$ with $k \leq n$ and $\ell \leq m$.

Applying standard techniques using Cramer's rule, Hadamard inequality, and the fact that A is integer matrix, one has that $\hat{w}_j, \hat{\xi}_j \leq U^m m^{m/2}$. It follows that $\left|\left|(\hat{w}, \hat{\xi})\right|\right|_1 \leq U^m m^{m/2+1}$. Hence, by a standard norm inequality and, respectively, the supposition of the proposition

$$\left|\left|(\hat{w}^\top, \hat{\xi}^\top)\right|\right|_2^2 \leq \left|\left|(\hat{w}^\top, \hat{\xi}^\top)\right|\right|_1^2 \leq U^{2m} m^{m+2} \leq c, d.$$

By construction $||\hat{w}||_0 + c\left|\left|\hat{\xi}\right|\right|_0 \leq ||\bar{w}||_0 + c\left|\left|\bar{\xi}\right|\right|_0$, so

$$\begin{aligned}\left|\left|(\hat{w}^\top, \hat{\xi}^\top)\right|\right|_2^2 + c\left|\left|\hat{\xi}\right|\right|_0 + d\,||\hat{w}||_0 &\leq U^{2m} m^{m+3} + c\left|\left|\hat{\xi}\right|\right|_0 + d\,||\hat{w}||_0 \\ &< \left|\left|({w^*}^\top, {\xi^*}^\top)\right|\right|_2^2 + c\,||\xi^*||_0 \\ &\quad + d\,||w^*||_0\,,\end{aligned}$$

thereby establishing a contradiction. □

Through a similar second-order cone reformulation of the constraints (7d), relaxing the variables, letting $\xi_i, w_j \in [0,1]$, we arrive at the second-order cone relaxation. We also replace the penalty $\sum_{j\in[n]} z_j$ in the objective by a hard constraint in order to facilitate comparison with the other sparse SVM relaxations (see the following section for details).

$$\min_{\xi,w,b,r,u,q,z} \quad \sum_{j=1}^{n}(u_j + dz_j) + \sum_{i=1}^{m}(r_i + cq_i) \tag{8a}$$

$$\text{s.t.} \quad Y(Aw - \mathbb{1}b) + \xi \geq \mathbb{1} \tag{8b}$$

$$(2w_j, u_j - z_j, u_j + z_j) \in \mathcal{Q}^3 \qquad j \in [n] \tag{8c}$$

$$(2\xi_i, r_i - q_i, r_i + q_i) \in \mathcal{Q}^3 \qquad i \in [m] \tag{8d}$$

$$\sum_{j\in[n]} z_j \leq r \tag{8e}$$

$$z \in [0,1]^n, q \in [0,1]^m, \xi \geq 0. \tag{8f}$$

We refer to the method of solving formulation (8) as CQ01-SSVM.

6 Evaluating the quality of the relaxation

Ideally one can solve the MINLP (3) in order to compare its optimal solution with the solution of the relaxation. However, the MINLP becomes increasingly difficult to solve with more than a small number of features.

The perspective (see Boyd and Vandenberghe (2004); Günlük and Linderoth (2010)) of $g(w_j, u_j) = w_j^2 - u_j$ is illustrated in Figure 1. If $u_j = w_j^2$, then the constraint (7c) implies that $z_j = 1$ if and only $w_j > 0$. Otherwise, if $u_j > w_j^2$, then for a solution (ξ, w, u, z) optimal to (5) with (5d) binding, it must be that $0 < z_j = w_j^2/u_j < 1$. Further, it is precisely when c is large compared with unity (the objective coefficients of the u_j's), and when r is small, that for each $j \in [n]$, u_j may tend to overestimate w_j^2 in order to allow $z_j < 1$. Intuitively, the larger the values of c, and the smaller r is, the larger the Lagrangian multipliers that are associated with the constraints (5c) and that "push against" z_j being large, for each $j \in [n]$. Empirical evidence for quality of the relaxation is given in Section 7.

Formulations (5) and (8) facilitate comparison with (2) and also with an integer solution: if (ξ, w, u, z) is optimal to (5) with $||w||_0 \leq r$, then since z_j for $j \in [n]$ is constrained from above by 1 or by constraint (5d), it follows by optimality to (5) that $u_j = w_j^2$ for all $j \in [n]$. Thus, if $||w||_0 \leq r$, it follows for $j \in [n]$ that $z_j \in \{0, 1\}$.

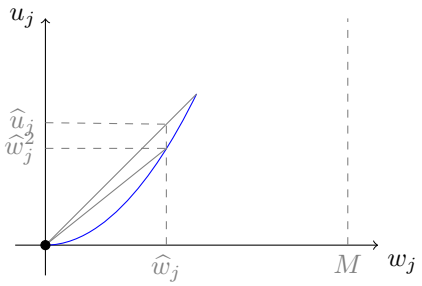


Figure 1: Illustration of the perspective relaxation: the optimum (with respect to feature j) $(\widehat{w}_j, \widehat{u}_j, \widehat{z}_j)$ is taken over the convex hull of $(0,0,0)$ and $(t, t^2, 1)$ for $t \geq 0$. Such an optimal solution satisfies $\widehat{z}_j = \widehat{w}_j^2/\widehat{u}_j$. In a big-$M$ formulation one would have $\widehat{z}_j = |\widehat{w}_j|/M$ for a potentially very large M.

7 Computational experiments

Chan et al. (2007) suggested that investigating the tradeoff of $||w||_0$ and accuracy was interesting but not in the scope of their work. Here we more closely examine this tradeoff for both SDP-SSVM and QCQP-SSVM as well as for our formulations CQ-SSVM and CQ01-SSVM. We compare the quality of the different relaxations for different values of the penalty parameters. We also compare the classification performance and generalization of our two novel formulations CQ-SSVM and CQ01-SSVM with previous relaxations.

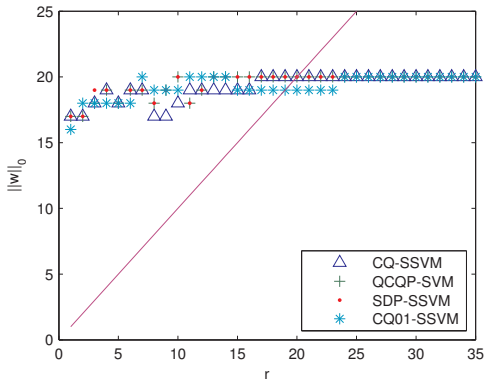
We solve the optimization models using the SDPT3 solver Toh et al. (1999) version 4.0. SDPT3 is a specialized interior-point solver for conic optimization problems. We note that all the formulations considered in this paper can be cast as conic optimization problems.

7.1 Optimization and relaxation quality

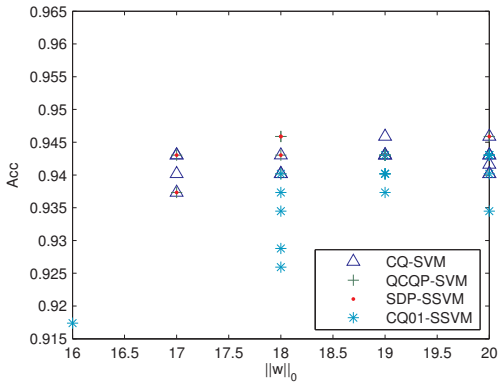
We ran experiments on the entire datasets in order to examine the quality of the relaxation of the MINLP by applying each type of relaxation. In Figures 2(a) and 2(c) we show the actual values of $||w||_0$ as r is varied in formulations (5), (8), (2), and SDP-SSVM using the UCI-Ionosphere dataset (Frank and Asuncion, 2010). Following the discussion of Section 6, points that lie on or below the diagonal line correspond to an integer solution. Figures 2(b) and 2(d) show the training accuracy vs. the density of w as r is varied, for $c = 2^6$ and $c = 2^{-6}$, respectively. In the case that c is "optimally selected" for the dataset and for the QCQP-SSVM method, as in the case of $c = 2^6$, then all four methods perform nearly the same. However, for most other choices of the parameter c, such as the case of $c = 2^{-6}$ in Figure 2(d), then there appears to be a clear advantage of CQ-SSVM and CQ01-SSVM over SDP-SSVM and QCQP-SSVM in terms of integrality of the indicator variable vector z, and more importantly in the training accuracy–$||w||_0$ space.

The average SDP-SSVM running time in the experiments of Figures 2(a)–2(d) was 40.99 seconds compared with 1.87 seconds on average to solve the CQ01-SSVM formulation, a formulation with $m+n$ second-order cone constraints. Due to the computational cost of SDP-SSVM and the fact that Figures 2(a)–2(d) demonstrate that the SDP-SSVM solutions are similar to those of QCQP-SSVM, we did not further consider SDP-SSVM.

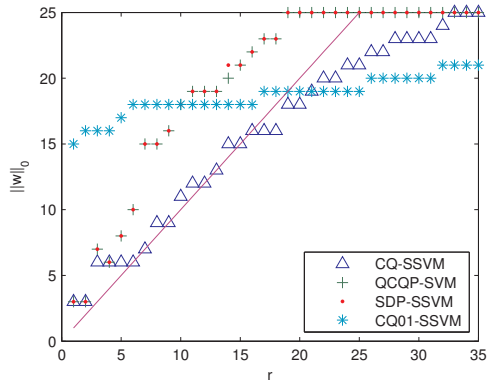
Table 1 indicates the datset sizes and shows the running times of CQ-SSVM and CQ01-SSVM compared with LP-SVM and QC-SSVM. The running times apply to runs using roughly 80% of the dataset, which is used as the training set. The results of in table indicate that in all cases considered CQ-SSVM is faster to compute than QCQP-SSVM.



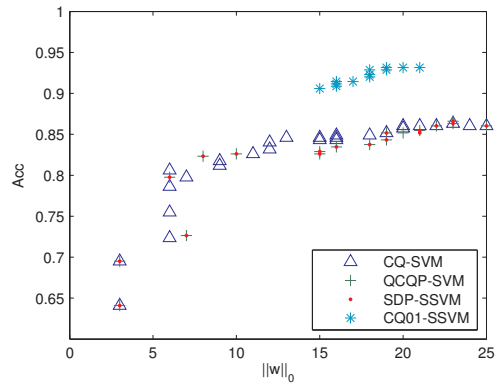
(a) Showing $||w||_0$ versus r with $c = 2^6$.



(b) Showing the tradeoff of $||w||_0$ and training accuracy while r is varied and $c = 2^6$; the optimal parameter value for QCQP-SSVM in the grid search described in Section 7.2



(c) Showing $||w||_0$ versus r with $c = 2^{-6}$.



(d) Showing the tradeoff of $||w||_0$ and training accuracy while r is varied and $c = 2^{-6}$.

Figure 2: Integrality and sparsity experiments using the Ionosphere dataset

7.2 Classification performance evaluation

For the classification experiments we did not further consider the SDP-SSVM method because of its computational cost and the fact that it seems to compute solutions that are similar to QCQP-SSVM. To set the parameter c we performed an internal 3-fold stratified cross validation (CV) for all datasets. The parameter setting was chosen to provide the best accuracy with $||w||_0$ as a tiebreaker. We experimented with the range of parameter values $c \in \{2^{-10}, 2^{-8}, \ldots, 2^8, 2^{10}\}$. The parameter r is set to 2 for CQ-SSVM and CQ01-SSVM: this setting seemed to provide a reasonable tradeoff of accuracy and sparsity. More careful fine-tuning of the parameter can be applied to improve the classification results even further. For QCQP-SSVM we applied the setting of $r = 0.1$ as recommended by Chan et al. (2007).

Table 2 displays the classification performance of the three methods compared also with the LP-SVM Fung and Mangasarian (2004) formulation. The results summarized in the table indicate that CQ01-SSVM provides the highest accuracy on average. In pairwise comparisons CQ01-SSVM and CQ-SSVM tie with each other and outperform the competing methods in most cases with respect to accuracy. QCQP-SSVM provided the sparsest classifiers while CQ01-SSVM seemed to provide the best balance of accuracy and sparsity. It should be noted however that QCQP-SSVM and CQ01-SSVM are more computationally expensive.

Table 1: CPU time statistics of the solver on 5-fold CV instances solved (25 in total for each dataset, as detailed in Section 7.2). The mean CPU-seconds is given plus and minus a standard deviation. The UCI dataset sizes are given after preprocessing; categorical attributes are converted into a several features, one for each attribute value, and observations with missing numerical attribute values are removed. Below Φ denotes the positive label proportion of the data.

Dataset	m	Φ	n	LP-SVM	QCQP-SSVM	CQ-SSVM	CQ01-SSVM
Voting	435	0.38	48	0.8 ± 0.4	5.5 ± 1.0	4.3 ± 0.7	13.3 ± 1.3
Wisc.	683	0.35	10	0.5 ± 0.1	9.7 ± 0.9	7.6 ± 0.9	13.2 ± 0.7
WDBC	569	0.37	31	0.6 ± 0.1	8.0 ± 0.7	7.8 ± 1.0	13.9 ± 1.0
WPBC	396	0.78	32	0.4 ± 0.1	5.4 ± 0.4	2.9 ± 0.3	11.5 ± 0.7
SPECT	80	0.50	22	0.2 ± 0.0	1.2 ± 0.2	1.1 ± 0.2	2.7 ± 0.2
SPECTF	80	0.50	44	0.3 ± 0.0	1.6 ± 0.2	1.6 ± 0.2	3.3 ± 0.3
Ionosphere	351	0.64	34	0.7 ± 0.2	4.8 ± 0.7	3.1 ± 0.3	9.7 ± 1.2
PIMA	768	0.65	8	0.3 ± 0.0	7.8 ± 0.6	7.4 ± 0.8	14.5 ± 1.1
Spam	4601	0.61	57	3.5 ± 0.8	163.3 ± 18.8	104.7 ± 19.7	477.1 ± 74.2
KDD CUP*	5039	0.73	121	103.2 ± 14.0	328.5 ± 37.5	209.8 ± 36.4	

* the value of m indicated is for samples of roughly 20% of the entire dataset that we used for the experiments described in detail in Section 7.3.

CQ-SSVM's classification performance provided a consistent balance of sparsity and accuracy over the datasets and also improved classification accuracy on average.

7.3 Experiments with designated test sets

Here we consider classification performance with designated test sets for SPECT, SPECTF (Frank and Asuncion, 2010), and the KDD CUP 1999 network intrusion datasets; we use a processed version of the dataset that corrects some of the flaws of the original network intrusion dataset; see Tavallaee et al. (2009). For this dataset, because of its size, we sampled randomly to select roughly 20% of the training set and repeated the experiment ten times. Here the parameters were chosen as follows: For QCQP-SSVM we used $r = .1$ as set by Chan et al. (2007). For CQ-SSVM $r = 2$. The parameter $c \in \{2^{-10}, 2^{-8}, \ldots, 2^{10}\}$ in each method is selected by 3-fold CV using only the sampled training data. We report the average accuracy and density over the 10 repetitions 10% sample size for these datasets in Table 3. The experiments show competitive classification results for our method. It also becomes apparent from the running times in Table 1 that the advantage of LP-SVM with respect to running time is not as significant for larger datasets such as the KDD network intrusion dataset. In all cases CQ-SSVM produced significantly sparser classifiers than LP-SVM.

8 Conclusions and future work

We propose novel second-order cone relaxations of sparse SVM. In empirical tests this relaxation is tighter than the norm-bound based convex relaxation of Chan et al. (2007), it is faster to compute, and it yields competitive classification performance. The formulations we propose appear to be more robust to the choice of the penalty parameters, obtaining a reasonable tradeoff of accuracy and sparsity without extensive fine-tuning of the penalty parameters. The improvement in the overall tradeoff of sparsity and classification

Table 2: Classification performance results with CQ-SSVM and CQ01-SSVM: 5 repetitions of 5-fold stratified CV experiments.

Dataset	LP-SVM		QCQP-SSVM		CQ-SSVM		CQ01-SSVM	
	$\|\|w\|\|_0$	Acc (%)	$\|\|w\|\|_0$	Acc (%)	$\|\|w\|\|_0$	Acc (%)	$\|\|w\|\|_0$	Acc (%)
Voting	12.5	95.50	9.9	95.29	12.1	95.37	15.8	95.74
Wisc.	8.8	96.63	8.2	96.72	8.5	96.66	8.2	96.60
WDBC	13.4	96.57	10.5	96.88	12.6	96.67	10.8	96.67
WPBC	0.0	78.05	0.0	78.05	0.2	78.05	2.0	78.05
SPECT	9.4	68.50	4.1	69.00	8.2	70.00	10.3	73.50
SPECTF	18.1	75.50	9.2	79.25	10.5	77.38	7.0	79.00
Ionosphere	25.6	87.68	21.6	87.30	25.0	87.99	21.8	87.64
PIMA	7.5	76.95	6.9	76.61	7.4	76.93	6.8	76.82
SPAM	54.9	92.89	35.0	90.46	53.2	92.79	39.6	90.85
Average	16.7	85.36	11.7	85.51	15.3	85.76	13.6	86.10

Table 3: Classification performance results using designated test sets averaged over 10 repetitions. For the KDD CUP dataset we performed random sampling in each run to select 20% of the training set.

Dataset	LP-SVM		QCQP-SSVM		CQ-SSVM	
	$\|\|w\|\|_0$	Acc (%)	$\|\|w\|\|_0$	Acc (%)	$\|\|w\|\|_0$	Acc (%)
SPECT	14.2 ± 4.0	73.80 ± 0.03	5.7 ± 4.0	70.75 ± 0.07	11.1 ± 4.3	71.34 ± 0.04
SPECTF	20.2 ± 9.4	75.19 ± 0.03	9.0 ± 4.2	74.06 ± 0.04	14.9 ± 6.4	76.04 ± 0.03
KDD CUP	85.0 ± 26.7	$^*87.20 \pm 0.62$	51.9 ± 34.4	86.27 ± 0.69	57.5 ± 13.6	86.98 ± 0.51

* 2 out of the 10 SDPT3 runs failed for LP-SVM due to numerical errors so that its accuracy is reported only for the 8 successful runs.

performance is especially apparent for CQ01-SSVM, which applies a similar relaxation technique to both the margin deviation variables and the hyperplane normal vector variables. Although more computationally expensive than alternative quadratic programming formulations considered herein, it remains less computationally expensive than SDP alternatives such as SDP-SSVM.

Note that as the solution of (3) may have many zero components of ξ, we do not necessarily need to add all m constraints (8d) ahead of time. In this paper we experimented with an interior point solver (SDPT3) and hence dynamically generating the constraints and resolving may not have been sensible. Resolving the problem and dynamic generation of the constraints may be sensible when solving these formulations using first-order methods. Recent and ongoing development of first-order methods for conic optimization and in particular for second-order cone programming may allow the application of our methods to large-scale datasets. The ability to restart from any initial point could also make it suitable for an online setting (e.g., see Shalev-Shwartz et al. (2007); Ferris and Munson (2004) for specialized large-scale methods for standard SVMs).

Our extension of sparse SVM for the zero-one loss may be also be useful for cases in which labels of observations are subject to noise or when labels may be flipped by an adversary; see, for example, Biggio et al. (2011) and references therein for different approaches.

References

Amaldi, E. and Kann, V. (1998). On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 209:237–260.

Ben-Tal, A. and Nemirovski, A. (2001). *Lectures on Modern Convex Optimization*. MPS-SIAM Series on Optimization, SIAM, Philadelphia.

Bennett, K. and Bredensteiner, E. (1997). A parametric optimization method for machine learning. *INFORMS Journal on Computing*, 9(3):311–318.

Biggio, B., Nelson, B., and Laskov, P. (2011). Support vector machines under adversarial label noise. *Journal of Machine Learning Research*, 20:97–112.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

Bradley, P. and Mangasarian, O. (1998). Feature selection via mathematical programming. *INFORMS Journal on Computing*, 10:209–217.

Bradley, P. and Mangasarian, O. (2000). Massive data discrimination via linear support vector machines. *Optimization Methods and Software*, 13(1):1–10.

Chan, A., Vasconcelos, N., and Lanckriet, G. (2007). Direct convex relaxations of sparse SVM. In *Proceedings of the* 24^{th} *International Conference on Machine Learning*.

Ferris, M. and Munson, T. (2004). Semismooth support vector machines. *Mathematical Programming*, 101:185–204.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository.

Fung, G. and Mangasarian, O. (2004). A feature selection Newton method for support vector machine classification. *Computational Optimization and Applications*, 28:185–202.

Goldberg, N. and Eckstein, J. (2010). Boosting with tightened L0-relaxation penalties. In *Proceedings of the* 27^{th} *International Conference on Machine Learning*.

Goldberg, N. and Eckstein, J. (2012). Sparse weighted voting classifier selection and its linear programming relaxations. *Information Processing Letters*, 112(12):481–486.

Guan, W., Gray, A., and Leyffer, S. (2009). Mixed integer support vector machines. In *Neural Information Processing Systems (NIPS) Workshop on Optimization in Machine Learning*.

Günlük, O. and Linderoth, J. (2010). Perspective reformulations of mixed integer nonlinear programs with indicator variables. *Mathematical Programming*, 124:183–205.

Höffgen, K.-U., Simon, H., and Horn, K. V. (1995). Robust trainability of single neurons. *J. of Computer and Systems Sciences*, 50:114–125.

Koltchinskii, V. and Panchenko, D. (2005). Complexities of convex combinations and bounding the generalization error in classification. *The Annals of Statistics*, 33:1455–1496.

Shalev-Shwartz, S., Singer, Y., and Srebro, N. (2007). Pegasos: Primal estimated subgradient solver for SVM. In *Proceedings of the* 24^{th} *International Conference on Machine Learning*.

Tan, M., Wang, L., and Tsang, I. (2010). Learning sparse SVM for feature selection on very high dimensional datasets. In *Proceedings of the* 27^{th} *International Conference on Machine Learning*.

Tavallaee, M., Bagheri, E., Lu, W., and Ghorbani, A. (2009). A detailed analysis of the KDD CUP 99 data set. In *IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA)*.

Toh, K., Todd, M., and Tutuncu, R. (1999). SDPT3 — a matlab software package for semidefinite programming. *Optimization Methods and Software*, 11:545–581.

Weston, J., Elisseeff, A., Schölkopf, B., and Tipping, M. (2003). Use of the zero norm with linear models and kernel methods. *Journal of Machine Learning Research*, 3:1439–1461.

Zhao, X., Sun, D., and Toh, K.-C. (2010). A Newton-CG augmented lagrangian method for semidefinite programming. *SIAM Journal on Optimization*, 20:1737–1765.