Journal of Machine Learning Research 7 (2006) - Submitted 8/05; Published 6/06

Exact 1-Norm Support Vector Machines via Unconstrained Convex Differentiable Minimization

Olvi L. Mangasarian[*] OLVI@CS.WISC.EDU
Computer Sciences Department
University of Wisconsin
Madison, WI 53706, USA

Editor: Kristin Bennett and Emilio Parrado-Hernández

Abstract

Support vector machines utilizing the 1-norm, typically set up as linear programs (Mangasarian, 2000; Bradley and Mangasarian, 1998), are formulated here as a completely unconstrained minimization of a convex differentiable piecewise-quadratic objective function in the dual space. The objective function, which has a Lipschitz continuous gradient and contains only one additional finite parameter, can be minimized by a generalized Newton method and leads to an exact solution of the support vector machine problem. The approach here is based on a formulation of a very general linear program as an unconstrained minimization problem and its application to support vector machine classification problems. The present approach which generalizes both (Mangasarian, 2004) and (Fung and Mangasarian, 2004) is also applied to nonlinear approximation where a minimal number of nonlinear kernel functions are utilized to approximate a function from a given number of function values.

1. Introduction

One of the principal advantages of 1-norm support vector machines (SVMs) is that, unlike 2-norm SVMs, they are very effective in reducing input space features for linear kernels and in reducing the number of kernel functions (Bradley and Mangasarian, 1998; Fung and Mangasarian, 2004) for nonlinear SVMs. With few exceptions, the simplex method (Dantzig, 1963) has been the exclusive algorithm for solving 1-norm SVMs. The interesting paper (Zhu et al., 2004) which treats the 1-norm SVM uses standard linear programming packages for solving their formulation. To the best of our knowledge there has not been an exact completely unconstrained differentiable minimization formulation of 1-norm SVMs, which is the principal concern of the present rather theoretical contribution which we outline now.

In Section 2 we show how a very general linear program can be solved as the minimization of a completely unconstrained differentiable piecewise-quadratic convex function that contains a single finite parameter. This result generalizes (Mangasarian, 2004) where linear programs with millions of constraints were solved as unconstrained minimization problems by a generalized Newton method. In Section 3 we show how to set up 1-norm SVMs, with linear and nonlinear kernels as unconstrained minimization problems and state a general-

[*] For commercial use of the algorithms described in this work, please contact the author.

ized Newton method for their solution. In Section 4 we show how to solve the problem of approximating an unknown function based on a given number of function values using a minimal number of kernel functions. We achieve this by again converting a 1-norm approximation problem to an unconstrained minimization problem. Computational results given in Section 5 show that the proposed approach is faster than a conventional linear programming solver, CPLEX (ILO, 2003), and faster than another related method as well as having better input space feature suppression for a linear classifier and mostly better kernel function suppression for a nonlinear classifier. Section 6 concludes the paper.

We now describe our notation and give some background material. All vectors will be column vectors unless transposed to a row vector by a prime $'$. For a vector x in the n-dimensional real space R^n, x_+ denotes the vector in R^n with all of its negative components set to zero. This corresponds to projecting x onto the nonnegative orthant. For a vector $x \in R^n$, x_* denotes the vector in R^n with components $(x_*)_i = 1$ if $x_i > 0$ and 0 otherwise (i.e. x_* is the result of applying the step function component-wise to x). For $x \in R^n$, $\|x\|_1$, $\|x\|$ and $\|x\|_\infty$, will denote the $1-$, $2-$ and $\infty-$ norms of x. For simplicity we drop the 2 from $\|x\|_2$. The notation $A \in R^{m\times n}$ will signify a real $m \times n$ matrix. For such a matrix A' will denote the transpose of A, A_i will denote the i-th row of A and A_{ij} will denote the ij-th element of A. A vector of ones or zeroes in a real space of arbitrary dimension will be denoted by e or 0, respectively. For a piecewise-quadratic function such as, $f(x) = \frac{1}{2}||(Ax-b)_+||^2 + \frac{1}{2}x'Px$, where $A \in R^{m\times n}$, $P \in R^{n\times n}$, $P = P'$, P positive semidefinite and $b \in R^m$, the ordinary Hessian does not exist because its gradient, the $n \times 1$ vector $\nabla f(x) = A'(Ax-b)_+ + Px$, is not differentiable but is Lipschitz continuous with a Lipschitz constant of $\|A'\|\,\|A\| + \|P\|$. However, one can define its **generalized Hessian** (Hiriart-Urruty et al., 1984; Facchinei, 1995; Mangasarian, 2001) which is the $n \times n$ symmetric positive semidefinite matrix:

$$\partial^2 f(x) = A' diag(Ax-b)_* A + P,$$

where $diag(Ax-b)_*$ denotes an $m \times m$ diagonal matrix with diagonal elements $(A_i x - b_i)_*$, $i = 1, \ldots, m$. The generalized Hessian has many of the properties of the regular Hessian (Hiriart-Urruty et al., 1984; Facchinei, 1995; Mangasarian, 2001) in relation to $f(x)$. If the smallest eigenvalue of $\partial^2 f(x)$ is greater than some positive constant for all $x \in R^n$, then $f(x)$ is a strongly convex piecewise-quadratic function on R^n. A separating plane, with respect to two given point sets $\mathcal{A}$ and $\mathcal{B}$ in R^n, is a plane that attempts to separate R^n into two halfspaces such that each open halfspace contains points mostly of $\mathcal{A}$ or $\mathcal{B}$. The notation $:=$ denotes a definition.

2. Linear Programs as Exact Unconstrained Differentiable Minimization Problems

We consider in this section a very general linear program (LP) that contains nonnegative and unrestricted variables as well as inequality and equality constraints. We will show how to obtain an exact solution of this LP by a single minimization of a completely unconstrained differentiable piecewise-quadratic function that contains a single finite parameter. We begin with the primal linear program:

$$\min_{(x,y)\in R^{n+\ell}} c'x + d'y \ \ s.t. \ \ Ax + By \geq b, \ Ex + Gy = h, \ x \geq 0, \tag{1}$$

where $c \in R^n$, $d \in R^\ell$, $A \in R^{m\times n}$, $B \in R^{m\times \ell}$, $E \in R^{k\times n}$, $G \in R^{k\times \ell}$, $b \in R^m$ and $h \in R^k$, and its dual:

$$\max_{(u,v)\in R^{m+k}} b'u + h'v \ \ s.t. \ \ A'u + E'v \leq c, \ B'u + G'v = d, \ u \geq 0. \tag{2}$$

The exterior penalty problem for the dual linear program is:

$$\min_{(u,v)\in R^{m+k}} \epsilon(-b'u - h'v) + \frac{1}{2}(\|(A'u + E'v - c)_+\|^2 + \|B'u + G'v - d\|^2 + \|(-u)_+\|^2). \tag{3}$$

Solving the exterior penalty problem for a positive sequence $\{\epsilon_i\}$ converging to zero will yield a solution to the dual linear program (2) (Fiacco and McCormick, 1968; Bertsekas, 1999). However, we will *not* do that here because of the inherent inaccuracies associated with asymptotic exterior penalty methods and the fact that this would merely yield an approximate *dual* solution but *not* a primal solution. Instead, we will solve the exterior penalty problem for some finite value of the penalty parameter ϵ and from this *inexact* dual solution we shall easily extract an *exact* primal solution by using the following proposition.

Proposition 1 Exact Primal Solution Computation *Let the primal LP (1) be solvable. Then the dual exterior penalty problem (3) is solvable for all $\epsilon > 0$. For any $\epsilon \in (0, \bar{\epsilon}]$ for some $\bar{\epsilon} > 0$, any solution (u, v) of (3) generates an exact solution to primal LP (1) as follows:*

$$x = \frac{1}{\epsilon}(A'u + E'v - c)_+, \ \ y = \frac{1}{\epsilon}(B'u + G'v - d). \tag{4}$$

In addition, this (x, y) minimizes:

$$\|x\|^2 + \|y\|^2 + \|Ax + By - b\|^2, \tag{5}$$

over the solution set of the primal LP (1).

Proof The dual exterior penalty minimization problem (3) can be written in the equivalent form:

$$\begin{array}{rcl} \displaystyle\min_{(u,v,z_1,z_2)\in R^{m+k+n+m}} \epsilon(-b'u - h'v) + \frac{1}{2}(\|z_1\|^2 + \|B'u + G'v - d\|^2 & + & \|z_2\|^2) \\ s.t. \ -A'u - E'v + c + z_1 & \geq & 0 \\ u + z_2 & \geq & 0. \end{array} \tag{6}$$

The justification for this is that at a minimum of (6) the variables z_1 and z_2 are nonnegative, else if any component of these variables is negative the objective function can be strictly decreased by setting that component to zero while maintaining constraint feasibility. Hence, at a solution of (6), $z_1 = (A'u + E'v - c)_+$ and $z_2 = (-u)_+$. The Wolfe dual (Mangasarian, 1994, Problem 8.2.2) for the convex quadratic program (6) is:

$$\begin{array}{rcl} \max\limits_{(u,v,z_1,z_2,r,s)\in R^{m+k+n+m+n+m}} -\dfrac{1}{2}((\|z_1\|^2 + \|B'u\|^2 + \|G'v\|^2 + 2v'GB'u - \|d\|^2 + \|z_2\|^2) & - & c'r \\ s.t. \; -\epsilon b + B(B'u + G'v - d) + Ar - s & = & 0 \\ -\epsilon h + G(B'u + G'v - d) + Er & = & 0 \\ z_1 = r & \geq & 0 \\ z_2 = s & \geq & 0, \end{array} \tag{7}$$

which can be written in the equivalent form:

$$\begin{array}{rcl} -\min\limits_{(u,v,r,s)\in R^{m+k+n+m}} \dfrac{1}{2}(\|r\|^2 + \|B'u\|^2 + \|G'v\|^2 + 2v'GB'u - \|d\|^2 + \|s\|^2) & + & c'r \\ s.t. \; -b + B(\frac{B'u+G'v-d}{\epsilon}) + A\frac{r}{\epsilon} = \frac{s}{\epsilon} & \geq & 0 \\ -h + G(\frac{B'u+G'v-d}{\epsilon}) + E\frac{r}{\epsilon} & = & 0 \\ r & \geq & 0. \end{array} \tag{8}$$

Note that at a solution of the exterior penalty problem (6) and the corresponding Wolfe dual (7) we have that:

$$\begin{array}{rcl} r = z_1 & = & (A'u + E'v - c)_+ \\ s = z_2 & = & (-u)_+. \end{array} \tag{9}$$

Define now:

$$\begin{array}{rcl} x & := & \frac{r}{\epsilon} = \frac{1}{\epsilon}(A'u + E'v - c)_+ \\ y & := & \frac{1}{\epsilon}(B'u + G'v - d), \end{array} \tag{10}$$

where the equality in (10) follows from (9). Substituting (10) in (8) gives, after some algebra, the optimization problem (11) below. It is easiest to see that (8) follows from (11) if we substitute for x and y from (10) in (11) below and note that $0 \leq r = \epsilon x$ and that $0 \leq s = \epsilon(Ax + By - b)$ which follow from the constraints of (8) and the definitions (10) of x and y.

$$\begin{array}{rcl} -\min\limits_{(x,y)\in R^{n+\ell}} c'x + d'y & + & \frac{\epsilon}{2}(\|x\|^2 + \|y\|^2 + \|Ax + By - b\|^2) \\ Ax + By & \geq & b \\ Ex + Gy & = & h \\ x & \geq & 0. \end{array} \tag{11}$$

This convex quadratic program (11) is feasible, because the linear program (1) is feasible. It is solvable for any $\epsilon > 0$ (Frank and Wolfe, 1956) because its objective function is bounded below since it is a strongly convex quadratic function in (x, y). Since the dual exterior penalty minimization problem objective (3) or equivalently (6) is bounded below by the negative of the objective function of (11) by the weak duality theorem (Mangasarian, 1994, Theorem 8.2.3), hence (3) is solvable for any $\epsilon > 0$. By the perturbation theory of linear programs (Mangasarian and Meyer, 1979), it follows that for $\epsilon \in (0, \bar{\epsilon}]$, for some $\bar{\epsilon} > 0$, (x, y) as defined in (10) or equivalently (4), solve the linear program (1) and additionally minimize the expression (5) over the solution set of the original linear program (1).□

A more direct, but just as laborious and rather unintuitive proof of Proposition 1 can be given by showing that the KKT necessary and sufficient optimality conditions for (11) follow from the necessary and sufficient optimality conditions of setting the gradient of the exterior penalty problem (3) equal to zero. We do not give that proof here because it does not justify how the quadratic perturbation terms of (11) arose, but it merely starts with these terms as given.

We turn now to an implementation of this result for various 1-norm SVMs.

3. 1-Norm SVMs as Unconstrained Minimization Problems

We consider first the 1-norm linear SVM binary classification problem (Mangasarian, 2000; Bradley and Mangasarian, 1998; Fung and Mangasarian, 2004):

$$\begin{array}{rrcl} \displaystyle\min_{(w,\gamma,y)} & \nu\|y\|_1 + \|w\|_1 & & \\ \text{s.t.} & D(Aw - e\gamma) + y & \geq & e \\ & y & \geq & 0, \end{array} \tag{12}$$

where, with some abuse of notation by multiple representation, we let the $m \times n$ matrix A in this section represent m points in R^n to be separated to the best extent possible by a separating plane:

$$x'w = \gamma, \tag{13}$$

according to the class of each row of A as given by the $m \times m$ diagonal matrix D with elements $D_{ii} = \pm 1$. The objective term $\|y\|_1$ minimizes the classification error weighted with the positive parameter ν while the term $\|w\|_1$ maximizes the ∞-norm margin (Mangasarian, 1999) between the bounding planes $x'w = \gamma \pm 1$ that approximately bound each of the two classes of points represented by A. It is well known (Bradley and Mangasarian, 1998; Fung and Mangasarian, 2004) that using $\|w\|_1$ in the objective function of (12) instead of the standard 2-norm squared term $\|w\|^2$ (Vapnik, 2000; Schölkopf and Smola, 2002) results in input space feature selection by suppressing many components of w, whereas the standard 2-norm SVM does not suppress any components of w in general. We convert (12) to an explicit linear program as in (Fung and Mangasarian, 2004) by setting:

$$w = p - q, \;\; p \geq 0, \; q \geq 0, \tag{14}$$

which results in the linear program:

$$\begin{array}{rrcl} \displaystyle\min_{(p,q,\gamma,y)} & \nu e'y + e'(p+q) & & \\ \text{s.t.} & D(A(p-q) - e\gamma) + y & \geq & e \\ & p, q, y & \geq & 0. \end{array} \tag{15}$$

We note immediately that this linear program is solvable because it is feasible and its objective function is bounded below by zero. Hence, Proposition 1 can be utilized to yield the following unconstrained reformulation of the problem.

Proposition 2 Exact 1-Norm SVM Solution via Unconstrained Minimization *The unconstrained dual exterior penalty problem for the 1-norm SVM (15):*

$$\min_{u \in R^m} \; -\epsilon e'u + \frac{1}{2}(\|(A'Du - e)_+\|^2 + \|(-A'Du - e)_+\|^2 + (-e'Du)^2 + \|(u - \nu e)_+\|^2 + \|(-u)_+\|^2), \tag{16}$$

is solvable for all $\epsilon > 0$. For any $\epsilon \in (0, \bar{\epsilon}]$ for some $\bar{\epsilon} > 0$, any solution u of (16) generates an exact solution of the 1-norm SVM classification problem (12) as follows:

$$\begin{array}{lcl} w = p - q = & = & \frac{1}{\epsilon}((A'Du - e)_+ - (-A'Du - e)_+), \\ \gamma & = & -\frac{1}{\epsilon}e'Du, \\ y & = & \frac{1}{\epsilon}(u - \nu e)_+. \end{array} \tag{17}$$

In addition this (w, γ, y) minimizes:

$$\|w\|^2 + \gamma^2 + \|y\|^2 + \|D(Aw - e\gamma) + y - e\|^2, \tag{18}$$

over the solution set of the 1-norm SVM classification problem (12).

We note here the similarity between our unconstrained penalty minimization problem (16) and the corresponding problem of (Fung and Mangasarian, 2004, Equation 23). But, we also note a major difference. In the latter, a penalty parameter α multiplies the term $\|(-u)_+\|^2$ of equation (16) above and is required to approach ∞ in order to obtain an exact solution to the original problem (12). Thus, the solution obtained by (Fung and Mangasarian, 2004, Equation 23) for any finite α is only approximate, as pointed out there. Furthermore, our solution to (16) here minimizes the expression (18) rather than being merely an approximate least 2-norm solution as is the case in (Fung and Mangasarian, 2004, Equation 11). However the generalized Newton method prescribed in (Fung and Mangasarian, 2004) for a sequence $\{\alpha \uparrow \infty\}$, is applicable here with $\alpha = 1$. For completeness we state that result here. To do that we let $f(u)$ denote the exterior penalty function (16). Then the gradient and generalized Hessian as defined in the Introduction are given as follows.

$$\begin{array}{rcl} \nabla f(u) & = & -\epsilon e + DA(A'Du - e)_+ - DA(-A'Du - e)_+ \\ & & +Dee'Du + (u - \nu e)_+ - (-u)_+. \end{array} \tag{19}$$

$$\begin{array}{rcl} \partial^2 f(u) & = & DA(diag((A'Du - e)_* + (-A'Du - e)_*)A'D \\ & & +Dee'D + diag((u - \nu e)_* + (-u)_*) \\ & = & DA(diag(|A'Du| - e)_*)A'D \\ & & +Dee'D + diag((u - \nu e)_* + (-u)_*), \end{array} \tag{20}$$

where the last equality follows from the equality:

$$(a - 1)_* + (-a - 1)_* = (|a| - 1)_*. \tag{21}$$

To handle a nonlinear symmetric kernel $K(A, B)$ that maps $R^{m \times n} \times R^{n \times \ell}$ into $R^{m \times \ell}$ and which generates, instead of the separating plane (13), the nonlinear separating surface:

$$K(x', A')Dv = \gamma, \tag{22}$$

all we need to do is essentially to make the replacement:

$$A \longrightarrow K(A, A')D, \tag{23}$$

which we justify now. For a linear kernel $K(A, A') = AA'$, we have that $w = A'Dv$, where v is a dual variable (Mangasarian, 2000) and the primal linear programming SVM (15) becomes upon using $w = p - q = A'Dv$ and minimizing the 1-norm of v in the objective instead that of w:

$$\begin{array}{rrcl} \displaystyle\min_{(v,\gamma,y)} & \nu e'y + \|v\|_1 & & \\ \text{s.t.} & D(AA'Dv - e\gamma) + y & \geq & e \\ & y & \geq & 0. \end{array} \tag{24}$$

Setting:

$$v = r - s, \;\; r \geq 0, \; s \geq 0, \tag{25}$$

the linear program (24) becomes:

$$\begin{array}{rrcl} \displaystyle\min_{(r,s,\gamma,y)} & \nu e'y + e'(r+s) & & \\ \text{s.t.} & D(AA'D(r-s) - e\gamma) + y & \geq & e \\ & r, s, y & \geq & 0, \end{array} \tag{26}$$

which is the linear kernel SVM in terms of the dual variable $v = r - s$. If we replace the linear kernel AA' in (26) by the nonlinear kernel $K(A, A')$ we obtain the nonlinear kernel linear program:

$$\begin{array}{rrcl} \displaystyle\min_{(r,s,\gamma,y)} & \nu e'y + e'(r+s) & & \\ \text{s.t.} & D(K(A,A')D(r-s) - e\gamma) + y & \geq & e \\ & r, s, y & \geq & 0. \end{array} \tag{27}$$

We immediately note that the linear program (15) is identical to the linear program (27) if we make the replacement (23).

Finally, a word regarding the choice of ϵ in Propositions 1 and 2. Computationally in (Fung and Mangasarian, 2004) this does not seem to be critical and is effectively addressed as follows. By (Lucidi, 1987, Corollary 3.2), if for two successive values of ϵ: $\epsilon^1 > \epsilon^2$, the corresponding solutions of the ϵ-perturbed quadratic programs (11) are equal, then under certain assumptions these equal successive solutions constitute a solution of the linear programs (1) or (12) that also minimize the quadratic perturbations (5) or (18). This result can be implemented computationally by using an ϵ, which when decreased by some factor yields the same solution to (1) or (12). In our computational results this turned out to either 4×10^{-4} or 10^{-6}.

We state now our generalized Newton algorithm for solving the unconstrained minimization problem (16) as follows.

Algorithm 3 Generalized Newton Algorithm for (16) *Let $f(u)$, $\nabla f(u)$ and $\partial^2 f(u)$ be defined by (16),(19) and (20). Set the parameter values ν, ϵ, δ, tolerance* tol, *and* imax *(typically: $\epsilon \in [10^{-6}\,, 4 \times 10^{-4}]$ for linear SVMs and $\epsilon \in [10^{-9}\,, 1]$ nonlinear SVMs, tol $= 10^{-3}$, imax $= 50$, while ν and δ are set by a tuning procedure). Start with any $u^0 \in R^m$. For $i = 0, 1, \ldots$:*

(I) $u^{i+1} = u^i - \lambda_i(\partial^2 f(u^i) + \delta I)^{-1}\nabla f(u^i) = u^i + \lambda_i d^i$,
where the Armijo stepsize $\lambda_i = \max\{1, \frac{1}{2}, \frac{1}{4}, \ldots\}$ *is such that:*

$$f(u^i) - f(u^i + \lambda_i d^i) \geq -\frac{\lambda_i}{4}\nabla f(u^i)'d^i, \tag{28}$$

and d^i is the modified Newton direction:

$$d^i = -(\partial^2 f(u^i) + \delta I)^{-1} \nabla f(u^i). \tag{29}$$

In other words, start with $\lambda_i = 1$ and keep multiplying λ_i by $\frac{1}{2}$ until (28) is satisfied.

(II) Stop if $\|u^i - u^{i+1}\| \leq tol$ or $i = imax$. Else, set $i = i + 1$ and go to (I).

(III) Define the solution of the 1*-norm SVM (12) with least quadratic perturbation (18) by (17) with $u = u^i$.*

We state a convergence result for this algorithm now.

Proposition 4 *Let $tol = 0$, $imax = \infty$ and let $\epsilon > 0$ be sufficiently small. Each accumulation point $\bar{u}$ of the sequence $\{u^i\}$ generated by Algorithm 3 solves the exterior penalty problem (16). The corresponding $(\bar{w}, \bar{\gamma}, \bar{y})$ obtained by setting u to $\bar{u}$ in (17) is an exact solution to the primal* 1*-norm SVM (12) which in addition minimizes the quadratic perturbation (18) over the solution set of (12).*

Proof That each accumulation point $\bar{u}$ of the sequence $\{u^i\}$ solves the minimization problem (13) follows from exterior penalty results (Fiacco and McCormick, 1968; Bertsekas, 1999) and standard unconstrained descent methods such as (Mangasarian, 1995, Theorem 2.1, Examples 2.1(i), 2.2(iv)) and the facts that the direction choice d^i of (24) satisfies, for some $c > 0$:

$$\begin{array}{rcl} -\nabla f(u^i)' d^i & = & \nabla f(u^i)'(\delta I + \partial^2 f(u^i))^{-1} \nabla f(u^i) \\ & \geq & c\|\nabla f(u^i)\|^2, \end{array} \tag{30}$$

and that we are using an Armijo stepsize (28). The last statement of the theorem follows from Proposition 2.□

We turn now to minimal kernel function approximation.

4. Minimal Kernel Function Approximation as Unconstrained Minimization Problems

We consider here the problem of constructing a kernel function approximation from a given number of function values using the 1-norm to minimize both the error in the approximation as well as the weights of the kernel functions. Utilizing the 1-norm in minimizing the kernel weights suppresses unnecessary kernel functions similar to the approach of (Mangasarian et al., 2004) except that we shall solve the resulting linear program here through an unconstrained minimization reformulation. Also, for simplicity we shall not incorporate prior knowledge as was done in (Mangasarian et al., 2004).

We consider m given function values $b \in R^m$ associated with m n-dimensional vectors represented by the m rows of the $m \times n$ matrix A. We shall fit the data points by a linear combination of symmetric kernel functions as follows:

$$K(A, A')v + e\gamma \approx b, \tag{31}$$

where the unknown parameters $v \in R^m$ and $\gamma \in R$ are determined by minimizing the 1-norm of the approximation error weighted by $\nu > 0$ and the 1-norm of v as follows:

$$\min_{(v,\gamma) \in R^{n+1}} \nu\|K(A, A')v + e\gamma - b\|_1 + \|v\|_1. \tag{32}$$

Setting

$$\begin{array}{rcl} v & = & r-s,\ r\geq 0,\ s\geq 0, \\ K(A,A')v+e\gamma-b & = & y-z,\ y\geq 0,\ z\geq 0, \end{array} \tag{33}$$

we obtain the following linear program:

$$\begin{array}{rrcl} \displaystyle\min_{(r,s,\gamma,y,z)} & \nu e'(y+z)+e'(r+s) & & \\ \text{s.t.} & K(A,A')(r-s)+e\gamma-y+z & = & b \\ & r,s,y,z & \geq & 0, \end{array} \tag{34}$$

which is similar to the nonlinear kernel SVM classifier linear programming formulation (27) with equality constraints replacing inequality constraints. We also note that this linear program is solvable because it is feasible and its objective function is bounded below by zero. Hence, Proposition 1 can be utilized to yield the following unconstrained reformulation of the problem.

Proposition 5 Exact 1-Norm Nonlinear SVM Approximation via Unconstrained Minimization *The unconstrained dual exterior penalty problem for the 1-norm SVM approximation (34):*

$$\begin{array}{l} \displaystyle\min_{u\in R^m}\ -\epsilon b'u+\frac{1}{2}(\|(K(A,A')u-e)_+\|^2+\|(-K(A,A')u-e)_+\|^2+ \\ \qquad\qquad\qquad (e'u)^2+\|(-u-\nu e)_+\|^2+\|(u-\nu e)_+\|^2), \end{array} \tag{35}$$

is solvable for all $\epsilon>0$. For any $\epsilon\in(0,\bar{\epsilon}]$ for some $\bar{\epsilon}>0$, any solution u of (35) generates an exact of the 1-norm SVM approximation problem (32) as follows:

$$\begin{array}{lcl} v=r-s= & = & \frac{1}{\epsilon}((K(A,A')u-e)_+-(-K(A,A')u-e)_+), \\ \gamma & = & \frac{1}{\epsilon}e'u, \\ y & = & \frac{1}{\epsilon}(-u-\nu e)_+, \\ z & = & \frac{1}{\epsilon}(u-\nu e)_+ \end{array} \tag{36}$$

In addition this (r,s,γ,y,z) minimizes:

$$\|r\|^2+\|s\|^2+\gamma^2+\|y\|^2+\|z\|^2, \tag{37}$$

over the solution set of the 1-norm SVM classification problem (34).

Computational results utilizing the linear programming formulation (32) with prior knowledge in (Mangasarian et al., 2004) but using the simplex method of solution is effective for solving approximation problems. The unconstrained minimization formulation (35) is another method of solution which can also handle such problems without prior knowledge as well as with prior knowledge with appropriate but straightforward modifications.

We turn now to our computational results.

5. Computational Results

Computational testing was carried on a 3 Ghz Pentium 4 machine with 2GB of memory running CentOS 4 Linux and utilizing the CPLEX 7.1 (ILO, 2003) linear programming package within MATLAB 7.1 (MATLAB, 1994-2001). We tested our algorithm on six publicly available data sets. Five from the UCI Machine Learning Repository Murphy and Aha (1992): Ionosphere, Cleveland Heart, Pima Indians, BUPA Liver and Housing. The sixth data set, Galaxy Dim, is available from Odewahn et al. (1992). The results are summarized in Tables 1 and 2.

For the linear classifier (13) we compare in Table 1, NLPSVM (Fung and Mangasarian, 2004), CPLEX (ILO, 2003) and our Generalized LPNewton Algorithm for (16), on six public data sets using ten-fold cross validation. NLPSVM is essentially identical to our algorithm, except that it requires a penalty parameter multiplying the last term of (16) to approach infinity. CPLEX uses the standard linear programming package CPLEX (ILO, 2003) to solve (26). We note that our method LPNewton is faster than both NLPSVM and CPLEX on all six data sets and gives the best feature suppression based on the average number of features used by the linear classifier (13). NLPSVM has the best test set correctness on two of the data sets, and comparable correctness on the other four. The Armijo step size was not needed in either NLPSVM or LPNewton. Tuning on 10% of the training set was used to determine the parameters ν and δ from the sets $\{2^{-12}, \ldots, 2^{12}\}$ and $\{10^{-3}, \ldots, .10^{3}\}$ respectively. Epsilon was set to the value 4.00E-04 used in (Fung and Mangasarian, 2004) for NLPSM and to 1.00E-06 for our LPNewton algorithm.

For the nonlinear classifier (22) we compare in Table 2, NLPSVM (Fung and Mangasarian, 2004), CPLEX (ILO, 2003) and our Generalized LPNewton Algorithm 3 for (27), on three public data sets using ten-fold cross validation. We note again that our method LPNewton is faster than both NLPSVM and CPLEX on all three data sets and gives the best reduction in the number of kernel functions utilized, on two of the data sets, based on the cardinality of $v = r - s$ as defined in (25) and (27). Best test set correctness was achieved on two data sets by our method and it was a close second on the third data set. Again the Armijo step size was not needed in either NLPSVM or LPNewton. Tuning and choice of the parameters ν and ϵ was done as for the linear classifier above. A Gaussian kernel was used for all three methods and data sets with the Gaussian parameter tuned from the set $\{2^{-12}, \ldots, 2^{12}\}$.

6. Conclusion and Outlook

We have derived an unconstrained differentiable convex minimization reformulation of a most general linear program and have applied it to 1-norm classification and approximation problems. Very effective computational results of our method on special cases of general linear programs (Mangasarian, 2004) and an approximate version for support vector machine classification (Fung and Mangasarian, 2004), as well as computational results presented in Section 5, lead us to believe that the proposed unconstrained reformulation of very general linear programs and support vector machines is a very promising computational method for solving such problems as well as extensions to knowledge-based formulations (Mangasarian, 2005; Fung et al., 2003; Mangasarian et al., 2004).

Data Set/Size	Algorithm	Iters	Time	Train %	Test %	Feat	Eps
Ionosphere	NLPSVM	69	0.1796	92.6254	83.8016	20.6	4e-4
Ionosphere	CPLEX		0.179	**92.6255**	85.4841	25.1	
Ionosphere	LPNewton	**30.7**	**0.0767**	89.6169	**87.1825**	**9.6**	1e-6
351× 34							
BUPA Liver	NLPSVM	100	0.1062	70.1791	**67.916**	5.9	4e-4
BUPA Liver	CPLEX		0.2278	**70.4994**	67.2941	6	
BUPA Liver	LPNewton	**63.3**	**0.0623**	69.1814	67.563	**5.2**	1e-6
345× 6							
Pima Indians	NLPSVM	93.2	0.2169	73.5809	72.6692	6.8	4e-4
Pima Indians	CPLEX		1.1707	**76.8086**	**75.2683**	5.8	
Pima Indians	LPNewton	**40.6**	**0.0904**	76.0563	75.0051	**4.6**	1e-6
768× 8							
Cleveland	NLPSVM	42.2	0.0515	85.6742	84.1609	7.5	4e-4
Cleveland	CPLEX		0.1409	**85.9348**	84.1609	8.4	
Cleveland	LPNewton	**25.3**	**0.028**	85.7478	**84.5287**	**7.1**	1e-6
297× 13							
Housing	NLPSVM	66.6	0.0891	83.9049	83.8078	9.1	4e-4
Housing	CPLEX		0.363	**86.8035**	**84.3882**	10.5	
Housing	LPNewton	**57.4**	**0.0781**	85.6626	83.2078	**7.7**	1e-6
506× 13							
Galaxy Dim	NLPSVM	97.5	1.097	94.4392	94.4415	5.9	4e-4
Galaxy Dim	CPLEX		12.5357	**95.5153**	**95.5153**	11.5	
Galaxy Dim	LPNewton	**39.2**	**0.4297**	94.4948	94.5131	**4.8**	1e-6
4192× 14							

Table 1: **Comparison of the Linear Classifier (13) obtained by NLPSVM (Fung and Mangasarian, 2004), CPLEX (ILO, 2003) and our Generalized LP-Newton Algorithm 3 for (16) on six public data sets. Time is for one fold in seconds, Train and Test corectness is the average over ten folds and Features (Feat) denote the average number over ten folds of input space features utilized by the linear classifier. Epsilon (Eps) is the finite parameter defined in (16). Best result is in bold. Note that LPNewton is fastest and has least features.**

Data Set/Size	Algorithm	Iters	Time	Train %	Test %	Card(v)	Eps
Ionosphere	NLPSVM	81.7	0.181	92.0242	89.4683	18.5	4e-4
Ionosphere	CPLEX		0.1555	**94.7773**	**91.4683**	15.5	
Ionosphere	LPNewton	**36.5**	**0.103**	92.5297	91.1587	**11.2**	1e-6
351× 34							
BUPA	NLPSVM	88.3	0.1706	68.8514	65.2521	**15.5**	4e-4
BUPA	CPLEX		0.2552	**74.1061**	69.2521	17.3	
BUPA	LPNewton	**88.2**	**0.1345**	73.6572	**70.6975**	25.5	1e+0
345× 6							
Cleveland	NLPSVM	84.6	0.1128	83.168	80.4368	9.1	4e-4
Cleveland	CPLEX		0.1097	**85.0383**	81.8161	11.8	
Cleveland	LPNewton	**80.2**	**0.1061**	83.0151	**82.8621**	**5.6**	1e-9
297× 13							

Table 2: **Comparison of the Nonlinear Classifier (22) obtained by NLPSVM (Fung and Mangasarian, 2004), CPLEX (ILO, 2003) and our Generalized LPNewton Algorithm 3 for (27) on three public data sets. Time for one fold is in seconds, Train and Test corectness is on ten folds. Card(v) denotes the average number of nonzero components of $v = r - s$ as defined in (25) and (27) and hence that is the number of kernel functions utilized by the nonlinear classifier (22). Epsilon (Eps) is the finite parameter defined in (16) with the replacement (23) of A by $K(A, A')D$. Features (Feat) denotes the average number of features over ten folds. Reduced SVM (RSVM) (Lee and Mangasarian, 2001) was used to speed all computations by using the reduced kernel $K(A, \bar{A}')$ where $\frac{m}{10}$ randomly chosen rows of A constitute the rows of rows of $\bar{A}$. Best result is in bold. Note that LPNewton is fastest.**

Acknowledgments

I am indebted to my PhD student Michael Thompson for the computational results given in this work. Research described in this Data Mining Institute Report 05-03, August 2005, was supported by National Science Foundation Grants CCR-0138308 and IIS-0511905, the Microsoft Corporation and ExxonMobil. Revised January 2006.

References

D. P. Bertsekas. *Nonlinear Programming.* Athena Scientific, Belmont, MA, second edition, 1999.

P. S. Bradley and O. L. Mangasarian. Feature selection via concave minimization and support vector machines. In J. Shavlik, editor, *Machine Learning Proceedings of the Fifteenth International Conference(ICML '98)*, pages 82–90, San Francisco, California, 1998. Morgan Kaufmann. ftp://ftp.cs.wisc.edu/math-prog/tech-reports/98-03.ps.

G. B. Dantzig. *Linear Programming and Extensions.* Princeton University Press, Princeton, New Jersey, 1963.

F. Facchinei. Minimization of SC^1 functions and the Maratos effect. *Operations Research Letters*, 17:131–137, 1995.

A. V. Fiacco and G. P. McCormick. *Nonlinear Programming: Sequential Unconstrained Minimization Techniques.* John Wiley & Sons, New York, NY, 1968.

M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.

G. Fung and O. L. Mangasarian. A feature selection Newton method for support vector machine classification. *Computational Optimization and Applications*, 28(2):185–202, July 2004. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/02-03.ps.

G. Fung, O. L. Mangasarian, and J. Shavlik. Knowledge-based support vector machine classifiers. In Suzanna Becker, Sebastian Thrun, and Klaus Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 521–528. MIT Press, Cambridge, MA, October 2003. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/01-09.ps.

J.-B. Hiriart-Urruty, J. J. Strodiot, and V. H. Nguyen. Generalized hessian matrix and second-order optimality conditions for problems with C^{L1} data. *Applied Mathematics and Optimization*, 11:43–56, 1984.

ILOG CPLEX 9.0 User's Manual. ILOG, Incline Village, Nevada, 2003. http://www.ilog.com/products/cplex/.

Y.-J. Lee and O. L. Mangasarian. RSVM: Reduced support vector machines. In *Proceedings of the First SIAM International Conference on Data Mining, Chicago, April 5-7, 2001, CD-ROM*, 2001. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/00-07.ps.

S. Lucidi. A new result in the theory and computation of the least-norm solution of a linear program. *Journal of Optimization Theory and Applications*, 55:103–117, 1987.

O. L. Mangasarian. *Nonlinear Programming.* SIAM, Philadelphia, PA, 1994.

O. L. Mangasarian. Parallel gradient distribution in unconstrained optimization. *SIAM Journal on Control and Optimization*, 33(6):1916–1925, 1995. ftp://ftp.cs.wisc.edu/tech-reports/reports/1993/tr1145.ps.

O. L. Mangasarian. A finite Newton method for classification problems. Technical Report 01-11, Data Mining Institute, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin, December 2001. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/01-11.ps.*Optimization Methods and Software* 17, 2002, 913-929.

O. L. Mangasarian. A Newton method for linear programming. *Journal of Optimization Theory and Applications*, 121:1–18, 2004. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/02-02.ps.

O. L. Mangasarian. Knowledge-based linear programming. *SIAM Journal on Optimization*, 15:375–382, 2005. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/03-04.ps.

O. L. Mangasarian. Arbitrary-norm separating plane. *Operations Research Letters*, 24: 15–23, 1999. ftp://ftp.cs.wisc.edu/math-prog/tech-reports/97-07r.ps.

O. L. Mangasarian. Generalized support vector machines. In A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 135–146, Cambridge, MA, 2000. MIT Press. ftp://ftp.cs.wisc.edu/math-prog/tech-reports/98-14.ps.

O. L. Mangasarian and R. R. Meyer. Nonlinear perturbation of linear programs. *SIAM Journal on Control and Optimization*, 17(6):745–752, November 1979.

O. L. Mangasarian, J. W. Shavlik, and E. W. Wild. Knowledge-based kernel approximation. *Journal of Machine Learning Research*, 5:1127–1141, 2004. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/03-05.ps.

MATLAB. *User's Guide.* The MathWorks, Inc., Natick, MA 01760, 1994-2001. http://www.mathworks.com.

P. M. Murphy and D. W. Aha. UCI machine learning repository, 1992. www.ics.uci.edu/∼mlearn/MLRepository.html.

S. Odewahn, E. Stockwell, R. Pennington, R. Humphreys, and W. Zumach. Automated star/galaxy discrimination with neural networks. *Astronomical Journal*, 103(1):318–331, 1992.

B. Schölkopf and A. Smola. *Learning with Kernels.* MIT Press, Cambridge, MA, 2002.

V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer, New York, second edition, 2000.

J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani. 1-Norm support vector machines. In Sebastian Thrun, Lawrence K. Saul, and Bernhard Schölkopf, editors, *Advances in Neural Information Processing Systems 16–NIPS2003*. MIT Press, 2004.