Smoothing Newton Method for L_1 Soft Margin Data Classification Problem

Weibing Chen¹, Hongxia Yin², and Yingjie Tian¹

¹ Research Center on Fictitious Economy & Data Science,
Chinese Academy of Sciences, Beijing 100190, China

² Department of Mathematics and Statistics, Minnesota State University Mankato,
273 Wissink Hall, Mankato, MN 56001, USA

chenweibing06@mails.gucas.ac.cn,

hongxia.yin@mnsu.edu,

tianyingjie1213@163.com

Abstract. A smoothing Newton method is given for solving the dual of the l_1 soft margin data classification problem. A new merit function was given to handle the high-dimension variables caused by data mining problems. Preliminary numerical tests show that the algorithm is very promising.

1 Introduction

In the multiple kernel learning two-class nonlinear data classification problem, we suppose that n data $\{(x_i, y_i)\}$ are given, where $x_i \in \mathcal{X}$ for some input space $\mathcal{X} \subset \mathbb{R}^p$, and $y_i \in \{-1, 1\}$ indicating the class to which the point x_i belongs and are called labels. We assume that $\mathcal{F}$ is an embedding space (called feature space), φ is a map from $\mathcal{X}$ to $\mathcal{F}$. A kernel is a function k such that $k(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle$ for any $x_i, x_j \in \mathcal{X}$, where $\langle \cdot, \cdot \rangle$ is the inner production. A kernel matrix is a square matrix $K \in \mathbb{R}^{n \times n}$ such that $K_{ij} = k(x_i, x_j)$ for $x_1, \dots, x_n \in \mathcal{X}$. It is known that every kernel matrix is symmetric positive semidefinite. Kernel based method for two-class nonlinear classification is to find an affine function in the feature space, $f(x) = \langle w, \varphi(x) \rangle + b$ for some weight vector $w \in \mathcal{F}$ and $b \in \mathbb{R}$ to maximize the margin (or distance) between the parallel hyperplanes $\langle w, \varphi(x) \rangle + b = 1$ and $\langle w, \varphi(x) \rangle + b = -1$ in the high-dimensional feature space that are as far apart as possible while still separating the data. The hard margin problem is to find (w^*, b^*) that solves

$$\begin{aligned} \min_{w, b} \quad & \langle w, w \rangle \\ \text{s.t.} \quad & y_i(\langle w, \varphi(x_i) \rangle + b) \geq 1, \quad i = 1, \dots, n, \end{aligned} \tag{1}$$

which realizes the maximal margin classifier with geometric margin $\gamma = \frac{2}{\|w^*\|}$, assuming it exists. γ is actually the distance between the convex hulls of the two classes of data.

Problem (1) was widely investigated by many authors, see [1], [9], [10], [11], [13], and [14] for example. However, since the solution of (1) exists only when the labeled sample is linearly separable in the feature space ($f(x) = \langle w, \Phi(x) \rangle + b$), the following soft margin problem was defined by introducing slack variable $\xi = (\xi_1, \xi_2, \dots, \xi_n)^T$ with $\xi_i \geq 0$ for all $i = 1, \dots, n$ in order to relax the constraints in (1) to

$$y_i(\langle w, \Phi(x) \rangle + b) \geq 1 - \xi_i, \quad i = 1, \dots, n.$$

There are variety of ways to define the soft margin problem [4], [9], [10], [11]. In 2004, Ferris and Munson [5] produced a semismooth support vector machine (SVM) where the soft margin was defined by the least-square of the error (it is l_2 norm). In the paper, We consider the l_1 -norm soft margin problem

$$\begin{aligned} \min_{w, b, \xi} \quad & \langle w, w \rangle + C \sum_{i=1}^n \xi_i \\ \text{s.t. } & y_i(\langle w, \Phi(x) \rangle + b) \geq 1 - \xi_i, \quad i = 1, \dots, n, \\ & \xi_i \geq 0, \quad i = 1, \dots, n, \end{aligned} \quad (2)$$

where the parameter $C > 0$ determines the trade off between a large margin $\gamma = 2/\|w\|$ and a small error penalty. The dual problem of (2) is

$$\begin{aligned} \max \quad & 2\alpha^T e - \alpha^T G(K)\alpha \\ \text{s.t. } \quad & y^T \alpha = 0, \\ & 0 \leq \alpha \leq \mathbf{C}, \end{aligned} \quad (3)$$

Here $\mathbf{C}$ is a constant vector with all components to be the same positive number C , $\alpha \in \mathbb{R}^n$ is the dual parameter with its components $\alpha_i (i = 1, 2, \dots, n)$ taking values in the interval $[0, C]$. For a fixed kernel K , (3) gives an upper bound on misclassification probability (see [1] Chapter 4 for details), solving problem (3) for a single kernel matrix is therefore a way to minimize the upper bound on error probability.

Smoothing Newton method was verified to be an efficient way to some nonsmooth equation system with global convergence and local superlinear convergence. Especially, it is successfully used to solve variation inequality, complementarity problems and KKT system of optimization problems. See [6], [3], and [8]. In the paper, we introduce a smoothing Newton method for solving the dual of the l_1 soft margin SVM problem (3). Moreover, in order to overcome the difficulty caused by the high-dimensional variables in data mining, we introduce a new merit function based on the Huber function [7].

The paper is organized as follows. In Section 2, we produce the smoothing equations for the KKT system of problem (3) by projection technique, then we introduce a new merit function and give a smoothing Newton method for solving (3). Numerical tests for illustrating the efficiency of the algorithm are given in Section 3. Conclusion and further remarks are in Section 4.

2 Smoothing Newton Method

In this section, we introduce a smoothing Newton method for solving (3). For this we rewrite the problem as the following minimization problem.

$$\begin{aligned} \min \quad & \frac{1}{2} \alpha^T G(K) \alpha - e^T \alpha \\ \text{s.t.} \quad & y^T \alpha = 0, \\ & 0 \leq \alpha \leq \mathbf{C}. \end{aligned} \quad (4)$$

The Lagrange function for problem (4) is

$$L(\alpha, \lambda) = \frac{1}{2} \alpha^T G(K) \alpha - e^T \alpha - \lambda y^T \alpha \quad (5)$$

and its derivative $L'_\alpha(\alpha, \lambda) = G(K)\alpha - e - \lambda y$, $L'_\lambda(\alpha, \lambda) = -y^T \alpha$. We denote $u = (\alpha, \lambda) \in \mathbb{R}^n \times \mathbb{R}$ and

$$F(u) = F(\alpha, \lambda) = \begin{bmatrix} G(K)\alpha - \lambda y - e \\ y^T \alpha \end{bmatrix}. \quad (6)$$

Then α^* is a solution of (4) if and only if there exists λ^* such that $u^* = (\alpha^*, \lambda^*)$ is a solution of the following variational inequality problem

$$(u - u^*)^T F(u^*) \geq 0, \forall u = (\alpha, \lambda) \in \Omega = [0, C]^n \times \mathbb{R}. \quad (7)$$

It is well known from [6] that solving (7) is equivalent to finding a root of the Robinson's Normal equations:

$$F(\Pi_\Omega(u)) + u - \Pi_\Omega(u) = 0 \quad (8)$$

where $\Pi_\Omega(u)$ is the projection of u onto Ω . From the definition of Ω we have that

$$\begin{bmatrix} G(K)\Pi_{[0,C]^n}\alpha - \lambda y - e \\ y^T \Pi_{[0,C]^n}\alpha \end{bmatrix} + \begin{bmatrix} \alpha \\ \lambda \end{bmatrix} - \begin{bmatrix} \Pi_{[0,C]^n}\alpha \\ \lambda \end{bmatrix} = 0, \quad (9)$$

that is,

$$H(u) := H(\alpha, \lambda) := \begin{bmatrix} G(K)\Pi_{[0,C]^n}\alpha + \alpha - \Pi_{[0,C]^n}\alpha - \lambda y - e \\ y^T \Pi_{[0,C]^n}\alpha \end{bmatrix} = 0. \quad (10)$$

Recall that for any three numbers $c \in \mathbb{R} \cup \{-\infty\}$, $d \in \mathbb{R} \cup \{\infty\}$ with $c \leq d$ and $v \in \mathbb{R}$, the median function

$$\text{mid}(c, d, v) = \Pi_{[c,d]}(v) = \begin{cases} c & \text{if } v < c \\ v & \text{if } c \leq v \leq d \\ d & \text{if } d < v \end{cases}$$

and the Chen-Harker-Kanzow-Smale smoothing function for $\text{mid}(c, d, v)$ is

$$\phi(t, c, d, v) = \frac{c + \sqrt{(c-v)^2 + 4t^2}}{2} + \frac{d - \sqrt{(d-v)^2 + 4t^2}}{2}, \quad (11)$$

where $(t, v) \in \mathbb{R}_{++} \times \mathbb{R}$. It can be seen that function $\phi(\cdot)$ is continuously differentiable at any $(t, v) \in \mathbb{R}_{++} \times \mathbb{R}$.

By defining $\phi : \mathbb{R}_{++} \times \mathbb{R}^n$ with its components

$$\phi_i(t, \alpha) = \phi(t, 0, C, \alpha_i) = \frac{\sqrt{(\alpha_i)^2 + 4t^2}}{2} + \frac{C - \sqrt{(C - \alpha_i)^2 + 4t^2}}{2}, \quad (12)$$

where $(t, \alpha_i) \in \mathbb{R}_{++} \times \mathbb{R}, i = 1, 2, \dots, n$, we have the smoothing equations associated to equations (10) as follows

$$\begin{aligned} \sum_{j=1}^n G(K)_{ij} \phi_j(t, \alpha) + \alpha_i - \phi_i(t, \alpha) - \lambda y_i - 1 &= 0, \quad i = 1, \dots, n, \\ \sum_{j=1}^n y_j \phi_j(t, \alpha) &= 0. \end{aligned} \quad (13)$$

Then we have

$$\Phi(t, \alpha, \lambda) := \begin{bmatrix} G(K)\phi(t, \alpha) + \alpha - \phi(t, \alpha) - \lambda y - e \\ y^T \phi(t, \alpha) \end{bmatrix} = 0, \quad (14)$$

where $\Phi : \mathbb{R}_{++} \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n+1}$.

Let $z = (t, \alpha, \lambda) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$ and define $\Theta : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n+2}$,

$$\Theta(z) = \Theta(t, \alpha, \lambda) = \begin{bmatrix} t \\ \Phi(z) \end{bmatrix}. \quad (15)$$

It can be seen that Θ is continuously differentiable at any $z \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$.

Proposition 1. *For any $z = (t, \alpha, \lambda) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$ with $t > 0$, the matrix $\Theta'(z)$ is nonsingular.*

Proof. Since $C > 0$, there exist relative interior points for problem (4).

$$\Theta'(t, \alpha, \lambda) = \begin{bmatrix} 1 & 0 & 0 \\ (G(K) + I)\phi'_t(t, \alpha) & M(z) & -y \\ y^T \phi'_t(t, \alpha) & y^T \phi'_\alpha(t, \alpha) & 0 \end{bmatrix} \quad (16)$$

where

$$M(z) = G(K)\phi'_\alpha(t, \alpha) + I - \phi'_\alpha(t, \alpha), \quad (17)$$

and $\phi'_t : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ is the gradient of ϕ with respect to t . $\phi'_\alpha(t, \alpha) = \text{diag}\{q_i\}$ is a $n \times n$ diagonal matrix with its elements

$$q_i(z) = \frac{1}{2} \left[\left(\frac{\alpha_i}{\sqrt{\alpha_i^2 + 4t^2}} \right) + \left(\frac{C - \alpha_i}{\sqrt{(C - \alpha_i)^2 + 4t^2}} \right) \right], \quad i = 1, 2, \dots, n. \quad (18)$$

Let $dz := (dt, d\alpha, d\lambda)^T \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$ satisfies $\Theta'(z)dz = 0$. Then from the above equations $dt = 0$ and

$$M(z)d\alpha - yd\lambda = 0, \quad (19)$$

$$y^T \phi'_\alpha(t, \alpha) d\alpha = 0. \quad (20)$$

Since $\phi'_\alpha(t, \alpha)$ is positive definite for $t > 0$ then we have from (20) that $y^T d\alpha = 0$. Substitute it into (19) and left multiply (19) by $(d\alpha)^T$ we have

$$(d\alpha)^T M(z)(d\alpha) + (d\alpha)^T y d\lambda = 0,$$

which means that $d\alpha = 0$. Since $G(K)$ is positive semidefinite and y_i takes value at $\{-1, 1\}$, from (19) we have that $d\lambda = 0$. Therefore, $\Theta'(z)$ is nonsingular for any z with $t > 0$.

In smoothing Newton method, people usually take $\|\Theta(z)\|^2$ as the merit function. However, since the data mining problems usually produce large-scale optimization problems, in order to overcome this difficulty in numerical computation, we introduce the following merit function $\Upsilon(z)$:

$$\Upsilon(z) = \sum_{j=1}^n \rho_{h_j}(\Theta_j(z))$$

where

$$\rho_{h_j}(\xi) = \begin{cases} \xi^2/2, & \text{if } |\xi| \leq h_j, \\ h_j|\xi| - h_j^2/2, & \text{otherwise} \end{cases}$$

is the well-known Huber function [7]. Since Huber function is linear when ξ is not small enough, the computation on the merit function will be very simple when the iteration is far from the optimal solution. However, we can not simply apply Newton's method on $\Upsilon(z)$ since Huber function is smooth but not second order differentiable. Before we give the smoothing Newton method, we let $\gamma \in (0, 1)$ is a real number and define $\beta : \mathbb{R}^{n+2} \rightarrow \mathbb{R}_+$,

$$\beta(z) := \gamma \min\{1, \|\Theta(z)\|\},$$

and

$$\Omega := \{z = (t, \alpha, \lambda) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R} \mid u \geq \beta(z)\bar{t}\},$$

where $\bar{t}$ is a given positive number. Now we are ready to give the smoothing Newton method.

Algorithm 1

Step 0. Choose constants $\bar{t} > 0, \delta \in (0, 1)$ and $\sigma \in (0, 1/2)$. Let $\bar{z} = (\bar{t}, 0, 0) \in \mathbb{R}_+ \times \mathbb{R}^n \times \mathbb{R}$, $t^0 = \bar{t}$ and $(\alpha^0, \lambda^0) \in \mathbb{R}^{n+1}$ be an arbitrary point. Set $l := 0$.

Step 1. If $\Theta(z^l) = 0$ then stop. Otherwise, let $\beta_l := \beta(z^l)$.

Step 2. Compute $\Delta z^l := (\Delta t^l, \Delta \alpha^l, \Delta \lambda) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$ by

$$\Theta(z^l) + \Theta'(z^l) \Delta z^l = \beta_l \bar{z}. \quad (21)$$

Step 3. Let m_l be the smallest nonnegative integer m satisfying

$$\Upsilon(z^l + \delta^m \Delta z^l) \leq [1 - 2\sigma(1 - \gamma\|\bar{t}\|)\delta^m] \Upsilon(z^l). \quad (22)$$

Define $z^{l+1} := z^l + \delta^{m_l} \Delta z^l$.

Step 4. Set $l := l + 1$ and go to Step 1.

From Proposition 1 and by using nonsmooth analysis, we can prove the following convergence theorem of the above algorithm. We omit its proof here because of the page limitation.

Theorem 1. *Suppose that z^* is an accumulation point of the infinite sequence $\{z^k\}$ generated by Algorithm 1. If all $V \in \partial\Theta(z^*)$ are nonsingular, then for any initial point $z^0 = (t^0, \alpha^0, \lambda^0)$ with $t^0 > 0$, the sequence $\{z^k\}$ converges to z^* quadratically. i.e., for k large enough,*

$$\|z^{k+1} - z^*\| = O(\|z^k - z^*\|^2)$$

and

$$t^{k+1} = O(t^k)^2.$$

3 Numerical Tests

In this section we report our numerical results of smoothing Newton methods for data classification of the heart disease, ionosphere and breast-cancer-Wisconsin data obtained from the UCI repository. Our numerical experiments were carried out in personal computer with 1060GHz AMD Sempron (tm) Processor and 512MB memory and 80G hard disk. The program is written in MATLAB 7.0.1.

3.1 The Pre-process of the Data

In data mining, the raw data we got may include very big and/or very small numbers. As we known, in numerical computation by computer, the big number may ‘eat’ the small number. In order to overcome this difficulty and obtain good computation results from Algorithm 1, we need to do the data pre-process first, which includes normalization and standardization for the data.

The data normalization and standardization we took are as follows,

$$v' = v - \min(v) \quad (23)$$

$$v'' = v' / \max(v') \quad (24)$$

where the vector v is a $n \times 1$ vector, and it presents an attribute of the data, (23) presents the standardization of the data and (24) presents the normalization of the data. When there is an attribute which has the same number in every sample, that is v is a vector of the same number, v' will be zero vector, and v'' will be not feasible. Thus we must delete the attributes which have the same numbers in every sample, so that the revised data can be feasible.

3.2 Numerical Results

Now we present the numerical results of smoothing Newton method for the soft margin data classification problem on the heart disease, ionosphere, breast-cancer-Wisconsin, and the data of Credit Cards. For the convenience of expression, we denote the linear kernel function $k_1(x_1, x_2) = x_1 \cdot x_2'$ as K_1 , polynomial

kernel function $k_2(x_1, x_2) = (x_1 \cdot x_2' + 1)^d$ as K_2 , and the Gaussian kernel function $k_3(x_1, x_2) = \exp(-(x_1 - x_2)^T(x_1 - x_2)/\sigma^2)$ as K_3 .

We choose the values of all parameters in Algorithm 1 as

$$\gamma = 0.01, \quad \sigma = 0.25, \quad \delta = 0.5, \quad \epsilon = 0.0001, \quad C = 1.$$

For each data-set, we randomly take 60% of the data as the training set and the left 40% as the test set. Numerical results on standard benchmark datasets are summarized in Table 1, Table 2 and Table 4. In the process of computation, numerical differentiation can be used to take the place of accurate derivative. We take two-point differentiation formula in our numerical tests:

$$f'(x) \approx \frac{f(x + \epsilon) - f(x)}{\epsilon}, \quad \epsilon > 0. \tag{25}$$

We first apply Algorithm 1 on the training set and obtain the value α , then we use it to classify the test data to check the accuracy of our methods. In Table 1 below, we listed the processing time (in seconds) and the number of iteration for working out the value of α . For example, in the heart disease data-set, the total number of the data is 270, the computation time is 4.969 seconds, and the number of Newton iterations in Algorithm 1 is 18 for using the linear kernel function K_1 .

Table 1. Time and number of iterations in computation

	total	K_1	K_2	K_3	K_3
			$d = 2$	$\sigma^2 = 1$	$\sigma^2 = 0.2$
heart disease	270	4.969 18	6.203 22	4.156 15	5.047 17
ionosphere	351	12.328 25	56.188 111	7.75 15	7.094 14
breast-cancer-W	699	100.64 47	59.000 29	39.734 19	41.719 19

After we obtained the value of α from Algorithm 1, we can easily check the accuracy of our algorithm by applying it on the test set, and it only takes little time. Table 2 below listed the accuracy of our algorithm with different kernel functions.

Table 2. The accuracy of Algorithm 1

	K_1	K_2	K_3	K_3
		$d = 2$	$\sigma^2 = 1$	$\sigma^2 = 0.2$
heart disease	86.36%	77.27%	77.27%	78.18%
ionosphere	92.20%	96.45%	97.87%	78.01%
breast-cancer-W	98.57%	98.92%	98.57%	98.21%

In Table 3 below we listed the accuracy of data classification by using semi-definite programming (SDP) on the l_1 soft margin classification problem with $C = 1$ in [12]. It can be seen from Table 2 and Table 3 that Algorithm 1 can provide better classification for the problems except the polynomial kernel classification K_2 with $d = 2$ for heart disease data.

Table 3. Accuracy of SDP in [12]

	K_1	K_2	K_3	K_3
		$d = 2$	$\sigma^2 = 1$	$\sigma^2 = 0.2$
heart disease	84.3%	79.3%	59.5%	—
ionosphere	83.1%	94.5%	92.1%	—
breast-cancer-W	87.7%	96.4%	89.0%	—

Table 4. Test results for credit cards data

Kernal	Parametes	Time (min)	Iterations	Accuracy
K_1		5.4083	57	79.39%
K_2	$d = 2$	3.8000	31	83.26%
K_3	$\sigma^2 = 2$	1.5516	16	84.19%
K_3	$\sigma^2 = 1$	1.5576	16	84.15%

Finally, we apply Algorithm on the data of Credit Cards. There are 6000 samples in total, 5040 of which are good credit, 960 of which are bad credit. Each sample involves 65 attributes, which were processed into numbers. Because the number of the samples is large, the data was randomly partitioned into 10% training and 90% test sets, that is the training data-set has 600 samples(500 good, 100 bad samples). The Table 4 below shows the detail of Algorithm 1 on the problem.

4 Conclusions

In the paper, we provide a smoothing Newton method for support vector machine model for l_1 soft magian data classification problem. The algorithm is global and local quadratic convergent. Numerical tests on some well-know data classification problem shows that the method is fast and can provider better accuracy that the method in literature. Further research on variety of kernel matrix classification and smoothing Newton methods for ν -support vector machines [2] and multi-class classification [15] are under going.

Acknowledgments. This research has been partially supported by a grant from National Natural Science Foundation of China (#10671203, #70621001, #70531040, #70501030, #10601064, #70472074) and Faculty Research Grant of Minnesota State University Mankato.

References

1. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines. Cambridge University Press, Cambridge (2000)
2. Chen, P., Lin, C., Schölkopf, B.: A tutorial on ν -support vector machines. Appl. Stoch. Models Bus. Ind. 21, 111–136 (2005)

3. Chen, X., Qi, L., Sun, D.: Global and superlinear convergence of smoothing Newton method and its application to general box constrained variational inequalities. *Math. Comp.* 67, 519–540 (1998)
4. Cortes, C., Vapnik, V.: Support vector networks. *Machine Learning* 20, 1–25 (1995)
5. Ferris, M.C., Munson, T.S.: Semismooth support vector machines. *Math. Program. Ser.B* 101, 185–204 (2004)
6. Harker, P.T., Pang, J.-S.: Finite-dimensional variational inequality and nonlinear complementarity problem: A survey of theory, algorithm and applications. *Math. Program.* 48, 161–220 (1990)
7. Huber, P.J.: Robust regression: Asymptotics, conjectures, and Monte Carlo: *Ann. Statist.* 1, 799–821 (1973)
8. Qi, L., Sun, D., Zhou, G.: A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities. *Math. program.* 87, 1–37 (2000)
9. Mangasarian, O.L.: Mathematical programming in data mining. *Data Mining and Knowledge Discovery* 1, 183–201 (1997)
10. Mangasarian, O.L., Musicant, D.R.: Lagrangian support vector machines. *Journal of Machine Learning Research* 1, 161–177 (2001)
11. Mangasarian, O.L.: A finite Newton method for classification. *Optim. Methods Softw.* 17, 913–929 (2002)
12. Lanckriet, G.R.G., Cristianini, N., Ghaoui, L.E., Bartlett, P., Jordan, M.I.: Learning the kernel matrix with semidefinite programming. *J. Machine Learning Research* 5, 27–72 (2004)
13. Schölkopf, B., Smola, A.J.: *Learning with Kernels— Support Vector Machines, Regularization, Optimization, and Beyond.* The MIT Press, Cambridge (2002)
14. Vapnik, V.N.: *The Nature of Statistical Learning Theory*, 2nd edn. Springer, Heidelberg (2000)
15. Zhong, P., Fukushima, M.: Regularized nonsmooth Newton method for multi-class support vector machines. *Optim. Methods Softw.* 22, 225–236 (2007)