J. Oper. Res. Soc. China (2014) 2:57–74
DOI 10.1007/s40305-014-0037-z

Consensus Proximal Support Vector Machine for Classification Problems with Sparse Solutions

Yan-Qin Bai · Yan-Jun Shen · Kai-Ji Shen

Received: 3 November 2013 / Revised: 23 December 2013 / Accepted: 10 January 2014 /
Published online: 21 March 2014

© Operations Research Society of China, Periodicals Agency of Shanghai University, and Springer-Verlag Berlin Heidelberg 2014

Abstract Classification problem is the central problem in machine learning. Support vector machines (SVMs) are supervised learning models with associated learning algorithms and are used for classification in machine learning. In this paper, we establish two consensus proximal support vector machines (PSVMs) models, based on methods for binary classification. The first one is to separate the objective functions into individual convex functions by using the number of the sample points of the training set. The constraints contain two types of the equations with global variables and local variables corresponding to the consensus points and sample points, respectively. To get more sparse solutions, the second one is l_1–l_2 consensus PSVMs in which the objective function contains an ℓ_1-norm term and an ℓ_2-norm term which is responsible for the good classification performance while ℓ_1-norm term plays an important role in finding the sparse solutions. Two consensus PSVMs are solved by the alternating direction method of multipliers. Furthermore, they are implemented by the real-world data taken from the University of California, Irvine Machine Learning Repository (UCI Repository) and are compared with the existed models such as ℓ_1-PSVM, ℓ_p-PSVM, GEPSVM, PSVM, and SVM-light. Numerical results show that our models outperform others with the classification accuracy and the sparse solutions.

Keywords Classification problems · Support vector machine · Proximal support vector machine · Consensus · Alternating direction method of multipliers

This work is supported by the National Natural Science Foundation of China (Grant No. 11371242) and the “085 Project” in Shanghai University.

Y.-Q. Bai (✉) · Y.-J. Shen · K.-J. Shen
Department of Mathematics, Shanghai University, Shanghai 200444, China
e-mail: yqbai@shu.edu.cn

Mathematics Subject Classification (2000) 90C10 · 90C20 · 49M20 · 65K05

1 Introduction

In machine learning and statistics, classification is the problem of identifying which of a set of categories (sub-populations) a new observation belongs to on the basis of a training set of data containing observations (or instances) whose category membership is known. The classification problems have practical applications in many areas of life, such as pattern recognition, regression forecasting, data processing, protein classification problem, meteorology, etc., [5, 6, 8, 26]. There are many methods for solving the classification problems, such as decision trees, neural networks, clustering algorithm, expectation-maximization (EM), support vector machine (SVM), etc., [3, 12]. Recently, a lot of attention has been paid to establishing the models and the algorithms which give a good trade-off of the better classification correctness versus the less computational efforts. Moreover, the more attention focuses on the simpler structure of models and the more sparse the solution for the classification problems is. The reader may be referred to the papers [17], Proximal SVM (PSVM) [11], etc.

SVM plays a key role in the classification problems and is used to transform the classification problems into an optimization problem. SVM has been widely used in many real-world problems such as pattern recognition, regression forecasting, data processing, etc., [1, 15, 16, 19, 20]. It is also an important achievement of machine learning theory in recent years. In 1995, Cortes and Vapnik formally proposed SVM based on statistical learning theory [9]. Vapnik first proposed the C-SVM and then gave the C-SVM with secondary relaxation variable [9, 27], where C is a regularization parameter to control the balance between the size of margin and the misclassification error. Then Schölkopf put forward the ν-SVM [23] to simplify the parameters adjustment of SVM, where ν is the upper bound of misclassification errors of the training samples, and simultaneously the lower bound of support vectors. From then on, there were a lot of extended SVMs, including one-class SVM [22, 25], Reduced SVM (RSVM) [18], Weighted SVM (WSVM) [8], LS-SVM (least squares SVM) [24], TSVM (Twin SVM) [17], PSVM [11], etc.

There are several standard methods for solving SVM such as modified gradient projection algorithm and interior-point methods, at least for small and medium size problems [4]. The alternating direction method of multipliers (ADMM) is mainly used in optimization problems with high dimension, and its initial point does not need to be feasible [2, 13, 14]. It is applicable in many cases, such as statistical learning, image processing, sensor networks, etc [4, 10, 13].

The original consensus problem is to deal with the optimization problem min $f(x) = \sum_{i=1}^{N} f_i(x)$ with the additive objective function such as $\sum_{i=1}^{N} f_i(x)$. Here $x \in \mathbb{R}^n$ is called the global variable. Consensus can be viewed as a simple technique for turning the additive objective into separable objective which splits easily by introducing the local variables x_i, $i = 1, \cdots, N$, that is, min $f(x) = \sum_{i=1}^{N} f_i(x_i)$ subject to $x_i - z = 0$, $i = 1, \cdots, N$. Boyd et al. pointed out that consensus problems have a long history, especially in conjunction with ADMM, (see, Boyd et al. [4]). Consensus models are usually used in the parallel optimization problems and require agreement among a number of processes for the same value. For a useful recent discussion of consensus algorithms, readers may refer to [23] and the references therein.

Motivated by recent attention in the context of consensus problems and algorithms, we note that the objective function in PSVM is $\|w\|_2^2 = \sum w_i^2$, and it can be handled by its own coordinate w_i corresponding to the element x_i in the train set. In this paper, we convert the original PSVM to the unconstrained form and also add $2l$ equality constraints to the model, where l is the number of the sample points. Then, we establish two consensus proximal support vector machines (PSVMs) models. The first one is to separate the objective functions into individual convex functions by using the number of the sample points of the training set. The constraints contain two types of the equations with global variables and local variables corresponding to the consensus points and sample points, respectively. To get more sparse solutions, the second one is consensus $l_1 - l_2$ PSVM in which a sum of absolute values of global variables are added to the objective function. Consensus least squares SVM was proposed before [19]. The difference between our consensus PSVM and consensus LSSVM is that our formulation leads to the strong convex objective functions, and thus we add more l equality constraints to the model, where l means the number of sample points. While consensus $l_1 - l_2$ PSVM aims to find a trade-off between classification performance and sparse solutions. These two consensus PSVMs are solved by the ADMM. As we mentioned above, consensus problem is to solve the problem in which a single global variable can be split into l parts. Therefore, ADMM can be derived either directly from the augmented Lagrangian or simply as a special case of the constrained optimization problem. Furthermore, they are implemented by the real-world data taken from the University of California, Irvine machine learning repository (UCI repository) and are compared with the existed models such as ℓ_1-PSVM, ℓ_p-PSVM, GEPSVM, PSVM, and SVM-light. Numerical results show that our models outperform to others with the classification accuracy and the sparse solutions.

The paper is organized as follows. In Sect. 2, we briefly recall the basic concepts of SVM and PSVM for binary classification problems. In Sect. 3, we present two consensus PSVMs that are consensus PSVM and consensus $\ell_1 - \ell_2$ PSVM. The second model contains an ℓ_1-norm term and an ℓ_2-norm term. The ℓ_2-norm term is responsible for the good classification performance, while ℓ_1-norm term plays an important role on finding sparse solutions. Section 4 investigates the performance of the two models via ADMM and compares them with other five SVMs by the numerical examples of the real-world data taking from the UCI Repository. Finally, we conclude the paper and briefly give the goal of the future research in Sect. 5.

Notation $\mathbb{R}^n$ stands for the set of n-dimensional real vectors. $\|\cdot\|_1$ and $\|\cdot\|_2$ denote the ℓ_1-norm and ℓ_2-norm, respectively. The soft thresholding operator S is the proximity operator of ℓ_1-norm. The penalty parameter $\rho > 0$ is used as the step size, and in this paper, we set $\rho = 1$. $\varepsilon_i^{\text{primal}}$ and $\varepsilon_i^{\text{dual}}, i = 1, 2$ means feasibility tolerances for the primal conditions and the dual ones. In the numerical experiments, we set $\varepsilon_i^{\text{primal}} = 10^{-4}$ and $\varepsilon_i^{\text{dual}} = 10^{-3}, i = 1, 2$, respectively.

2 Standard SVM and PSVM

In this section, we briefly recall some preliminaries of SVM and PSVM for binary classification problems. We will not go into the details of the SVM and PSVM, and readers may refer to [9, 11, 27].

First, we recall the linear separated case of a binary classification. Given a training set $\{(x_1, y_1), \cdots, (x_l, y_l)\} \subseteq \mathbb{R}^n \times \{-1, +1\}$, it is linearly separable if there exists a hyperplane $w^{\mathrm{T}}x + b = 1$ such that

$$\begin{aligned} w^{\mathrm{T}}x_i + b \geqslant 1 \quad & \textit{if } y_i = +1 \quad i = 1, \cdots, l \\ w^{\mathrm{T}}x_i + b \leqslant -1 \quad & \textit{if } y_i = -1 \quad i = 1, \cdots, l \end{aligned}$$

Thus, the hard-margin SVM [9] can be formulated as follows:

$$\begin{aligned} &\min_{w,b} \frac{1}{2}\|w\|_2^2 \\ &\text{s.t.}\, y_i(w^{\mathrm{T}}x_i + b) \geqslant 1 \quad i = 1, \cdots, l \end{aligned} \tag{2.1}$$

where $x_i \in \mathbb{R}^n, y_i \in \{-1, 1\}, i = 1, \cdots, l, w \in \mathbb{R}^n, b \in \mathbb{R}$. Geometrically the hard-margin SVM is illustrated in Fig. 1. When there exist wild points, the case is the nonlinear separated case. Introducing a slack variables $\xi_i, i = 1, \cdots, l$ and a fixed penalty parameter $C > 0$ to the objective function of the hard-margin SVM, it can be converted to the soft-margin SVM [9] as follows:

$$\begin{aligned} &\min_{w,b,\xi} \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{l}\xi_i \\ &\text{s.t.}\, y_i(w^{\mathrm{T}}x_i + b) \geqslant 1 - \xi_i \quad i = 1, \cdots, l \\ &\quad \xi_i \geqslant 0 \quad i = 1, \cdots, l \end{aligned} \tag{2.2}$$

where $x_i \in \mathbb{R}^n, y_i \in \{-1, 1\}, \xi_i \in \mathbb{R}, \quad i = 1, \cdots, l, w \in \mathbb{R}^n$, and $b \in \mathbb{R}$. Figure 2 shows the geometric interpretation of the soft-margin SVM. The PSVM formulation is as good as the soft-margin SVM formulation with the advantage such as strong convexity of the objective function. It can give the explicit exact solution, whereas it is impossible to do that in the soft-margin SVM formulation. The formulation of PSVM [11] is as following.

$$\begin{aligned} &\min_{w,b,\xi} \frac{1}{2}\left(\|w\|_2^2 + b^2\right) + \frac{C}{2}\sum_{i=1}^{l}\xi_i^2 \\ &\text{s.t.}\, y_i\left(w^{\mathrm{T}}x_i + b\right) = 1 - \xi_i, \quad i = 1, \cdots, l \end{aligned} \tag{2.3}$$

where $x_i \in \mathbb{R}^n, y_i \in \{-1, 1\}, \eta_i \in \mathbb{R}, i = 1, \cdots, l, w \in \mathbb{R}^n$, and $b \in \mathbb{R}$. From Fig. 3, we can see that PSVM also aims to find the largest distance between two dashed lines, i.e., the bounding planes $w^{\mathrm{T}}x + b = 1$ and $w^{\mathrm{T}}x + b = \ -1$. Here minimizing $\frac{C}{2}\sum\limits_{i=1}^{l}\xi_i^2$ means that the bounding planes are located as far as possible in the middle of positive points and negative points, respectively.

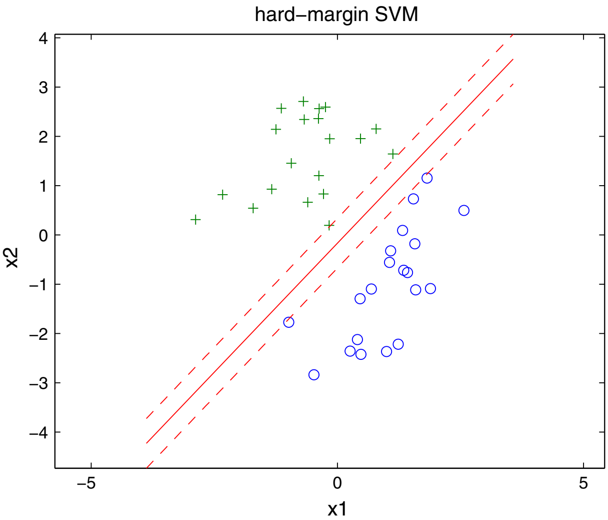

Fig. 1 The points of the *circle* class (the negative samples) and the *plus* symbol class (the positive samples) with a full line showing a separation hyperplane $w^{\mathrm{T}}x + b = 0$ between them

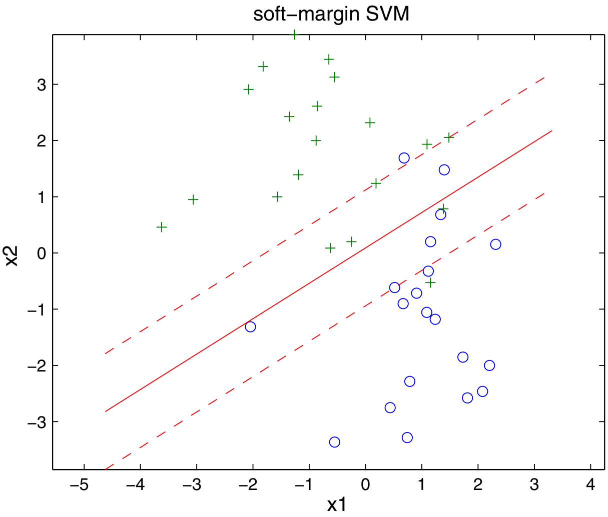

Fig. 2 The points of the *circle* class (the negative samples) and the *plus* symbol class (the positive samples) with a full line showing a separation hyperplane $w^{\mathrm{T}}x + b = 0$ between them

3 Two Consensus PSVM and Their Training Approaches

In this section, we start by reformulating the first consensus PSVM model. Then we consider the sparse solutions of the consensus PSVM and present the second one

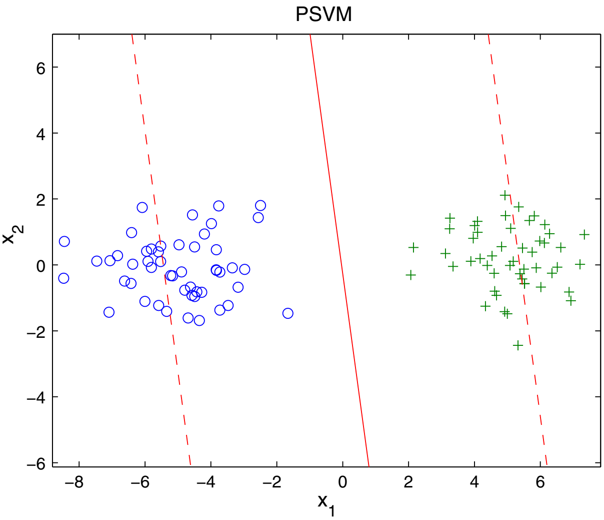

Fig. 3 The points of the *circle* class (the negative samples) and the *plus* symbol class (the positive samples) with a full line showing a separation hyperplane $w^{\mathrm{T}}x + b = 0$ between them

called the $l_1{-}l_2$ consensus PSVMs in which the objective function contains an ℓ_1-norm term and an ℓ_2-norm term. The ℓ_2-norm term is responsible for the good classification performance, while ℓ_1-norm term plays an important role on finding sparse solutions. The idea of consensus is based on parallel computing in which computational efforts can be shorted for solving large or complex problems. The consensus models are usually used in the parallel optimization.

3.1 Consensus PSVM

By introducing two types of the variables, the local variables $w_i \in \mathbb{R}^n, b_i \in \mathbb{R}, i = 1, \cdots, l$ and global variables $z \in \mathbb{R}^n, d \in \mathbb{R}$. The original PSVM (2.3) can be converted to the consensus PSVM as follows:

$$\begin{aligned} \min_{w,b,z,d} \; & \lambda\|z\|_2^2 + \lambda d^2 + \frac{1}{l}\sum_{i=1}^{l}\left(1 - y_i\left(w_i^{\mathrm{T}}x_i + b_i\right)\right)^2 \\ \text{s.t.}\; & w_i = z, \quad i = 1, \cdots, l, \\ & b_i = d, \quad i = 1, \cdots, l, \end{aligned} \tag{3.1}$$

where $\lambda > 0, x_i \in \mathbb{R}^n, y_i \in \{-1, 1\}, w_i \in \mathbb{R}^n, b_i \in \mathbb{R}, \quad i = 1, \cdots, l, z \in \mathbb{R}^n, d \in \mathbb{R}$.

Obviously, the objective function is separable in (3.1). The splitting technique can be used to convert the implicit expression into the explicit one, and thus, the model contains more information. Moreover, (3.1) is called the global consensus model since all the local variables should be equal. We expect that the model would produce higher classification accuracy and computational efficiency than single model strategies.

The abbreviated form of consensus PSVM can be expressed as

$$\begin{aligned} &\min_{w,b,z,d} \sum_{i=1}^{l} f_i(w_i, b_i) + g(z, d) \\ &\text{s.t.}\, w_i - z = 0, \quad i = 1, \cdots, l, \\ &\quad b_i - d = 0, \quad i = 1, \cdots, l, \end{aligned} \tag{3.2}$$

where $\sum_{i=1}^{l} f_i(w_i, b_i) = \frac{1}{l}\sum_{i=1}^{l}\left(1 - y_i\left(w_i^{\mathrm{T}} x_i + b_i\right)\right)^2, g(z, d) = \lambda\|z\|_2^2 + \lambda d^2.$

Let α and β be the Lagrangian multipliers for the equality constraints in (5). The augmented Lagrangian function can be expressed as follows:

$$\begin{aligned} L(w, b, z, d, \alpha, \beta) = &\sum_{i=1}^{l} f_i(w_i, b_i) + g(z, d) + \sum_{i=1}^{l} \alpha_i^{\mathrm{T}}(w_i - z) + \frac{\rho}{2}\sum_{i=1}^{l} \|w_i - z\|_2^2 \\ &+ \sum_{i=1}^{l} \beta_i^{\mathrm{T}}(b_i - d) + \frac{\rho}{2}\sum_{i=1}^{l} \|b_i - d\|_2^2 \end{aligned}$$

Here, we primarily give the optimal conditions of consensus PSVM,

$$\begin{cases} \nabla_{w_i} f_i\left(w_i^*, b_i^*\right) + \alpha_i^* + \rho\left(w_i^* - z^*\right) = 0,\, i = 1, \cdots, l, \\ \nabla_{b_i} f_i\left(w_i^*, b_i^*\right) + \beta_i^* + \rho\left(b_i^* - d^*\right) = 0,\, i = 1, \cdots, l, \\ \nabla_z g(z^*, d^*) + \sum_{i=1}^{l}\left(-\alpha_i^* - \rho\left(w_i^* - z^*\right)\right) = 0, \\ \nabla_d g(z^*, d^*) + \sum_{i=1}^{l}\left(-\beta_i^* - \rho\left(b_i^* - d^*\right)\right) = 0, \\ w_i^* - z^* = 0,\, i = 1, \cdots, l, \\ b_i^* - d^* = 0,\, i = 1, \cdots, l, \end{cases}$$

Subsequently, we give the stopping criteria due to the optimal conditions,

$$\begin{aligned} 0 &= \nabla_z g(z^{k+1}, d^k) - \sum_{i=1}^{l} \alpha_i^k - \sum_{i=1}^{l} \rho\left(w_i^{k+1} - z^{k+1}\right) \\ &= \nabla_z g(z^{k+1}, d^k) - \sum_{i=1}^{l} \alpha_i^k - \sum_{i=1}^{l} \rho \hat{r}_i^{k+1} \\ &= \nabla_z g(z^{k+1}, d^k) - \sum_{i=1}^{l} \alpha_i^{k+1} \end{aligned}$$

and

$$\begin{aligned} 0 &= \nabla_d g(z^{k+1}, d^{k+1}) - \sum_{i=1}^{l} \beta_i^k - \sum_{i=1}^{l} \rho\left(b_i^{k+1} - d^{k+1}\right) \\ &= \nabla_d g(z^{k+1}, d^{k+1}) - \sum_{i=1}^{l} \beta_i^k - \sum_{i=1}^{l} \rho \hat{s}_i^{k+1} \\ &= \nabla_d g(z^{k+1}, d^{k+1}) - \sum_{i=1}^{l} \beta_i^{k+1} \end{aligned}$$

Obviously, the third and the fourth conditions are always satisfied.

The first and the second conditions involve the dual feasibility, so we have

$$\begin{aligned}0 &= \nabla_{w_i} f_i\left(w_i^{k+1}, b_i^k\right) + \alpha_i^k + \rho\left(w_i^{k+1} - z^k\right)\\ &= \nabla_{w_i} f_i\left(w_i^{k+1}, b_i^k\right) + \alpha_i^k + \rho\left(w_i^{k+1} - z^{k+1}\right) + \rho\left(z^{k+1} - z^k\right)\\ &= \nabla_{w_i} f_i\left(w_i^{k+1}, b_i^k\right) + \alpha_i^{k+1} + \rho\left(z^{k+1} - z^k\right), \quad i = 1, \cdots, l\end{aligned}$$

and

$$\begin{aligned}0 &= \nabla_{b_i} f_i\left(w_i^{k+1}, b_i^{k+1}\right) + \beta_i^k + \rho\left(b_i^{k+1} - d^k\right)\\ &= \nabla_{b_i} f_i\left(w_i^{k+1}, b_i^{k+1}\right) + \beta_i^k + \rho\left(b_i^{k+1} - d^{k+1}\right) + \rho\left(d^{k+1} - d^k\right)\\ &= \nabla_{b_i} f_i\left(w_i^{k+1}, b_i^{k+1}\right) + \beta_i^{k+1} + \rho\left(d^{k+1} - d^k\right), \quad i = 1, \cdots, l,\end{aligned}$$

where $r = \rho\left(z^{k+1} - z^k\right)$ and $s = \rho\left(d^{k+1} - d^k\right)$ are called the dual residuals.

From the last two conditions, we can get the primal residuals as follows:

$$\hat{r}_i = w_i - z, \hat{s}_i = b_i - d, \quad i = 1, \cdots, l.$$

Accordingly, the primal residuals can be written as

$$\|\hat{r}\|_2^2 = \sum_{i=1}^{l} \|w_i - z\|_2^2, \|\hat{s}\|_2^2 = \sum_{i=1}^{l} \|b_i - d\|_2^2.$$

And therefore, the stopping criteria can be expressed as,

$$\|\hat{r}\|_2^2 \leqslant \varepsilon_1^{\text{primal}}, \|\hat{s}\|_2^2 \leqslant \varepsilon_2^{\text{primal}}, \|r\|_2^2 \leqslant \varepsilon_1^{\text{dual}}, \|s\|_2^2 \leqslant \varepsilon_2^{\text{dual}}.$$

Now we can establish the consensus PSVM algorithm for classification problems,

Algorithm for Consensus PSVM

Given a training set $T = \{(x_1, y_1), \cdots, (x_l, y_l)\} \subseteq \mathbb{R}^n \times \{-1, 1\}$ and select parameters λ and ρ. With the given iterate t^k, we can get the new iterate t^{k+1} as follows:

Step 1. Update $\tilde{t}^k = \{\tilde{w}_1^k, \cdots, \tilde{w}_l^k, \tilde{b}_1^k, \cdots, \tilde{b}_l^k, \tilde{z}^k, \tilde{d}^k, \tilde{\alpha}_1^k, \cdots, \tilde{\alpha}_l^k, \tilde{\beta}_1^k, \cdots, \tilde{\beta}_l^k\}$ in the alternating order by ADMM iterative scheme.

$$\begin{aligned}\tilde{w}_i^{k+1} &:= \arg\min_{w_i} \left(\frac{1}{l}\left(1 - y_i\left(w_i^{\mathrm{T}} x_i + b_i^k\right)\right)^2 + \alpha_i^{k\mathrm{T}}\left(w_i - z^k\right) + \frac{\rho}{2}\left\|w_i - z^k\right\|_2^2\right), \quad i = 1, \cdots, l,\\ \tilde{b}_i^{k+1} &:= \arg\min_{b_i} \left(\frac{1}{l}\left(1 - y_i\left(\left(\tilde{w}_i^{k+1}\right)^{\mathrm{T}} x_i + b_i\right)\right)^2 + \beta_i^{k\mathrm{T}}\left(b_i - d^k\right) + \frac{\rho}{2}\left\|b_i - d^k\right\|_2^2\right), \quad i = 1, \cdots, l,\\ \tilde{z}^{k+1} &:= \arg\min_{z} \left(\lambda\|z\|_2^2 + \sum_{i=1}^{l}\left(-\alpha_i^{k\mathrm{T}} z + \frac{\rho}{2}\left\|\tilde{w}_i^{k+1} - z\right\|_2^2\right)\right),\\ \tilde{d}^{k+1} &:= \arg\min_{d} \left(\lambda d^2 + \sum_{i=1}^{l}\left(-\beta_i^{k\mathrm{T}} d + \frac{\rho}{2}\left\|\tilde{b}_i^{k+1} - d\right\|_2^2\right)\right),\\ \tilde{\alpha}_i^{k+1} &:= \alpha_i^k + \rho\left(\tilde{w}_i^{k+1} - \tilde{z}^{k+1}\right), \quad i = 1, \cdots, l,\\ \tilde{\beta}_i^{k+1} &:= \beta_i^k + \rho\left(\tilde{b}_i^{k+1} - \tilde{d}^{k+1}\right), \quad i = 1, \cdots, l.\end{aligned}$$

Step 2. Stopping criteria. Quit if the following conditions are satisfied.

$$\|\hat{r}\|_2^2 \leqslant \varepsilon_1^{\text{primal}}, \|\hat{s}\|_2^2 \leqslant \varepsilon_2^{\text{primal}}, \|r\|_2^2 \leqslant \varepsilon_1^{\text{dual}}, \|s\|_2^2 \leqslant \varepsilon_2^{\text{dual}}.$$

Then, we get the solution $t^* = \{w_1^*, \cdots, w_l^*, b_1^*, \cdots, b_l^*, z^*, d^*, \alpha_1^*, \cdots, \alpha_l^*, \beta_1^*, \cdots, \beta_l^*\}$.

Step 3. Construct the decision function as $f(x) = \text{sgn}\Big((z^*)^{\text{T}} x + d^*\Big)$.

According to the above algorithm for Consensus PSVM, it is evident that after each iteration t^k, we can compute the consensus point $(\tilde{z}^{k+1}, \tilde{d}^{k+1})$, in particular when the iteration reaches the stopping criteria, an optimal or near optimal consensus point (z^*, d^*) can be obtained.

What's more, we briefly analyze the computational complexity of our methods. The complexity mainly relies on ADMM iterative scheme. For each iteration, $w_1, \cdots, w_l$ are solved at the same time. After solving $w_1, \cdots, w_l, b_1, \cdots, b_l$ are also solved at the same time. Then z, d can be solved. At last, $\alpha_1, \cdots, \alpha_l, \beta_1, \cdots, \beta_l$ can be also solved simultaneously. Thus, each iteration contains only 4 flops. Compared with PSVM algorithm [11], the total cost of our methods is also small.

3.2 Consensus ℓ_1–ℓ_2 PSVM

To get more sparse solutions, we add a term of absolute values of global variables z to the objective function of consensus PSVM 5, then ℓ_1–ℓ_2 PSVM can be reformulated as following:

$$\begin{aligned} \min_{w,b,z,d} \; & \lambda\|z\|_1 + (1-\lambda)\|z\|_2^2 + (1-\lambda)d^2 + \frac{1}{l}\sum_{i=1}^{l}\big(1 - y_i\big(w_i^{\text{T}} x_i + b_i\big)\big)^2 \\ \text{s.t.}\; & w_i = z, \quad i = 1, \cdots, l, \\ & b_i = d, \quad i = 1, \cdots, l, \end{aligned} \tag{3.3}$$

where $\lambda \in [0, 1], x_i \in \mathbb{R}^n, y_i \in \{-1, 1\}, w_i \in \mathbb{R}^n, b_i \in \mathbb{R}, i = 1, \cdots, l, z \in \mathbb{R}^n, d \in \mathbb{R}$.

The augmented Lagrangian function of (6) can be expressed as

$$\begin{aligned} L(w, b, z, d, \alpha, \beta) = & \lambda\|z\|_1 + (1-\lambda)\|z\|_2^2 + (1-\lambda)d^2 + \frac{1}{l}\sum_{i=1}^{l}\big(1 - y_i\big(w_i^{\text{T}} x_i + b_i\big)\big)^2 \\ & + \sum_{i=1}^{l}\alpha_i^{\text{T}}(w_i - z) + \frac{\rho}{2}\sum_{i=1}^{l}\|w_i - z\|_2^2 + \sum_{i=1}^{l}\beta_i^{\text{T}}(b_i - d) + \frac{\rho}{2}\sum_{i=1}^{l}\|b_i - d\|_2^2 \end{aligned}$$

Thus, the iterative scheme of ADMM for solving (6) is as follows:

$$
\begin{aligned}
w_i^{k+1} &:= \arg\min_{w_i}\left(\frac{1}{l}\left(1-y_i\left(w_i^{\mathrm{T}}x_i+b_i^k\right)\right)^2+\alpha_i^{k\mathrm{T}}\left(w_i-z^k\right)+\frac{\rho}{2}\left\|w_i-z^k\right\|_2^2\right),\ i=1,\cdots,l,\\
b_i^{k+1} &:= \arg\min_{b_i}\left(\frac{1}{l}\left(1-y_i\left(\left(w_i^{k+1}\right)^{\mathrm{T}}x_i+b_i\right)\right)^2+\beta_i^{k\mathrm{T}}\left(b_i-d^k\right)+\frac{\rho}{2}\left\|b_i-d^k\right\|_2^2\right),\ i=1,\cdots,l,\\
z^{k+1} &:= S_{\frac{\lambda}{2(1-\lambda)+\rho l}}\left(\frac{\sum\limits_{i=1}^{l}\rho w_i^{k+1}+\sum\limits_{i=1}^{l}\alpha_i^k}{2(1-\lambda)+\rho l}\right),\\
d^{k+1} &:= \arg\min_{d}\left((1-\lambda)d^2+\sum_{i=1}^{l}\left(-\beta_i^{k\mathrm{T}}d+\frac{\rho}{2}\left\|b_i^{k+1}-d\right\|_2^2\right)\right),\\
\alpha_i^{k+1} &:= \alpha_i^k+\rho\left(w_i^{k+1}-z^{k+1}\right),\ i=1,\cdots,l,\\
\beta_i^{k+1} &:= \beta_i^k+\rho\left(b_i^{k+1}-d^{k+1}\right),\ i=1,\cdots,l.
\end{aligned}
$$

Compared with (3.1), the ADMM iterative scheme is different in z-iteration,

1° $z_j>0$,

$$
\begin{aligned}
0 &= 2(1-\lambda)z_j+\lambda-\sum_{i=1}^{l}\alpha_{ij}-\sum_{i=1}^{l}\rho\left(w_{ij}-z_j\right),\\
z_j &= \frac{\sum\limits_{i=1}^{l}\rho w_{ij}+\sum\limits_{i=1}^{l}\alpha_{ij}-\lambda}{2(1-\lambda)+\rho l};
\end{aligned}
$$

2° $z_j<0$,

$$
\begin{aligned}
0 &= 2(1-\lambda)z_j-\lambda-\sum_{i=1}^{l}\alpha_{ij}-\sum_{i=1}^{l}\rho\left(w_{ij}-z_j\right),\\
z_j &= \frac{\sum\limits_{i=1}^{l}\rho w_{ij}+\sum\limits_{i=1}^{l}\alpha_{ij}+\lambda}{2(1-\lambda)+\rho l};
\end{aligned}
$$

3° $z_j=0$,

$$
\begin{aligned}
0 &= 2(1-\lambda)z_j+\lambda h_j-\sum_{i=1}^{l}\alpha_{ij}-\sum_{i=1}^{l}\rho\left(w_{ij}-z_j\right),\\
0 &= \lambda h_j-\sum_{i=1}^{l}\alpha_{ij}-\sum_{i=1}^{l}\rho w_{ij},\\
&\left|\sum_{i=1}^{l}\rho w_{ij}+\sum_{i=1}^{l}\alpha_{ij}\right|\leqslant|\lambda|,
\end{aligned}
$$

where $j=1,\cdots,n$, $\left\|h_j\right\|_2^2\leqslant 1$.

At the last iteration, we can get z^*,

$$
z_j^*=\max\left\{\frac{1}{2(1-\lambda)+\rho l}\left(\left|\sum_{i=1}^{l}\rho w_{ij}^*+\sum_{i=1}^{l}\alpha_{ij}^*\right|-\lambda\right),0\right\}\operatorname{sgn}\left(\frac{\sum\limits_{i=1}^{l}\rho w_{ij}^*+\sum\limits_{i=1}^{l}\alpha_{ij}^*}{2(1-\lambda)+\rho l}\right)
$$

where $j=1,\cdots,n$.

From the definition of the soft thresholding operator S, z^* is given by,

$$z_j^* = \begin{cases} \frac{1}{2(1-\lambda)+\rho l}\left(\sum\limits_{i=1}^{l} \rho w_{ij}^* + \sum\limits_{i=1}^{l} \alpha_{ij}^* - \lambda\right), & \sum\limits_{i=1}^{l} \rho w_i^* + \sum\limits_{i=1}^{l} \alpha_i^* > \lambda \\ 0, & \left|\sum\limits_{i=1}^{l} \rho w_i^* + \sum\limits_{i=1}^{l} \alpha_i^*\right| \le \lambda \\ \frac{1}{2(1-\lambda)+\rho l}\left(\sum\limits_{i=1}^{l} \rho w_{ij}^* + \sum\limits_{i=1}^{l} \alpha_{ij}^* + \lambda\right), & \sum\limits_{i=1}^{l} \rho w_i^* + \sum\limits_{i=1}^{l} \alpha_i^* < -\lambda \end{cases}$$

where $j = 1, \cdots, n$.

Equivalently, $z^* = S_{\frac{\lambda}{2(1-\lambda)+\rho l}}\left(\frac{\sum\limits_{i=1}^{l} \rho w_i^* + \sum\limits_{i=1}^{l} \alpha_i^*}{2(1-\lambda)+\rho l}\right)$.

4 Numerical Experiments

In this section, we firstly report results on five synthetic datasets in Table 1 in terms of the classification accuracy. Then eight real-world datasets are taken from the UCI Repository, including Heart disease, Australian credit approval, Sonar, Pima Indians diabetes, Spambase, Mushroom, BUPA liver, and Ionosphere (Table 2). They are used to evaluate the classification performance and the execution time of our approaches in Tables 3, 4, and 5, respectively. Compared with real-world datasets, the dimension of synthetic datasets is lower, and the number of sample points is smaller. Two classes of sample points are selected better in synthetic datasets, and thus, they are easier to separate. While sample points of real-world datasets may be crowded together, and they are hard to be controlled by the artificial means (Table 6).

The consensus PSVM training approaches are implemented on a PC with an Intel Pentium IV 1.73 GHz CPU, 1,024 MB RAM, the Windows XP operating system, and the Matlab 2011a development environment. In the following experiments, we set the penalty parameter $\rho = 1$ and set the termination tolerances $\varepsilon_i^{\text{primal}} = 10^{-4}$ and $\varepsilon_i^{\text{dual}} = 10^{-3}, i = 1, 2$, respectively. The variables $\tilde{w}_i^0, \tilde{b}_i^0, \tilde{z}^0, \tilde{d}^0, \tilde{\alpha}_i^0$ and $\tilde{\beta}_i^0$, $i = 1, \cdots, l$ are initialized to zero.

Figure 4 shows the iterative process of the problem with the synthetic data, including 50 positive instances and 50 negative instances. The examples split into ten groups, in the worst case, each group contains only the examples of the same kind. The circles and crosses stand for two different classes.

Firstly, we give the classification accuracy of our approaches for the synthetic data in Table 1.

To demonstrate the performance of our methods better, we report results on the following datasets from the UCI Repository.

We summarize the classification performances of consensus PSVM and consensus ℓ_1–ℓ_2 PSVM for the real-world data in Tables 3 and 4, respectively.

Table 1 Performance of classification on five problems with synthetic data

+train./−train./dimension	Consensus PSVM	Consensus ℓ_1−ℓ_2 PSVM
50/50/2	95.00 % ($\lambda = 1$)	97.00 % ($\lambda = 1$)
100/80/2	96.11 % ($\lambda = 1$)	94.44 % ($\lambda = 1$)
500/500/2	96.50 % ($\lambda = 1$)	96.50 % ($\lambda = 1$)
50/50/3	98.00 % ($\lambda = 1$)	98.00 % ($\lambda = 1$)
100/100/6	97.50 % ($\lambda = 1$)	96.50 % ($\lambda = 1$)

Table 2 Datasets from the UCI Repository

Dataset	Classes	Instances	Features
Heart	2	270	13
Australian	2	690	14
Sonar	2	208	60
Pima	2	768	8
Spambase	2	4 601	57
Mushroom	2	8 124	22
BUPA	2	345	6
Ionosphere	2	351	34

Table 3 Classification accuracy rates obtained by consensus PSVM, consensus ℓ_1−ℓ_2 PSVM, ℓ_p-PSVM, and ℓ_1-PSVM [7]

Dataset	Consensus PSVM	Consensus ℓ_1−ℓ_2 PSVM	ℓ_p-PSVM	ℓ_1-PSVM
Heart	**84.81 %** ($\lambda = 1$)	84.07 % ($\lambda = 0.5$)	79.63 %	79.63 %
Australian	**86.09 %** ($\lambda = 1$)	**86.09 %** ($\lambda = 0.5$)	85.80 %	85.94 %
Sonar	**89.90 %** ($\lambda = 1$)	85.10 % ($\lambda = 0.5$)	77.51 %	75.62 %

Table 4 Classification accuracy rates obtained by consensus PSVM, consensus ℓ_1−ℓ_2 PSVM, GEPSVM, PSVM, and SVM-light [21]

Dataset	Consensus PSVM	Consensus ℓ_1−ℓ_2 PSVM	GEPSVM	PSVM	SVM-light
Pima	**78.39 %** ($\lambda = 1$)	77.86 % ($\lambda = 0.5$)	73.6 %	75.9 %	75.7 %
Spambase	**88.81 %** ($\lambda = 1$)	87.37 % ($\lambda = 0.5$)	76.8 %	77.1 %	77.1 %
Mushroom	**94.37 %** ($\lambda = 1$)	92.15 % ($\lambda = 0.5$)	81.1 %	80.9 %	81.5 %

And we also compare them with the numerical results of ℓ_1-PSVM, ℓ_p-PSVM, GEPSVM, PSVM, and SVM-light in terms of the classification accuracy, respectively.

Table 5 Average time on the four datasets obtained by consensus PSVM, consensus ℓ_1–ℓ_2 PSVM, PSVM [11]

Dataset	Consensus PSVM (s)	consensus ℓ_1–ℓ_2 PSVM (s)	PSVM (s)
Pima	**0.003 7**	0.015 8	0.02
Mushroom	**0.008 3**	0.095 9	1.15
BUPA	**0.012 3**	0.152 8	0.02
Ionosphere	**0.006 6**	0.081 1	0.17

Table 6 Sparse solutions and classification accuracy obtained by consensus ℓ_1–ℓ_2 PSVM and ℓ_1-PSVM [7]

Dataset	consensus ℓ_1–ℓ_2 PSVM	ℓ_1-PSVM
Heart	**84.07 %** ($\lambda = 0.5, \sharp = \mathbf{4}$)	79.63 % ($\sharp = 3$)
Australian	**86.09 %** ($\lambda = 0.5, \sharp = \mathbf{7}$)	85.94 % ($\sharp = 2$)
Sonar	**83.65 %** ($\lambda = 1, \sharp = \mathbf{37}$)	75.62 % ($\sharp = 36$)

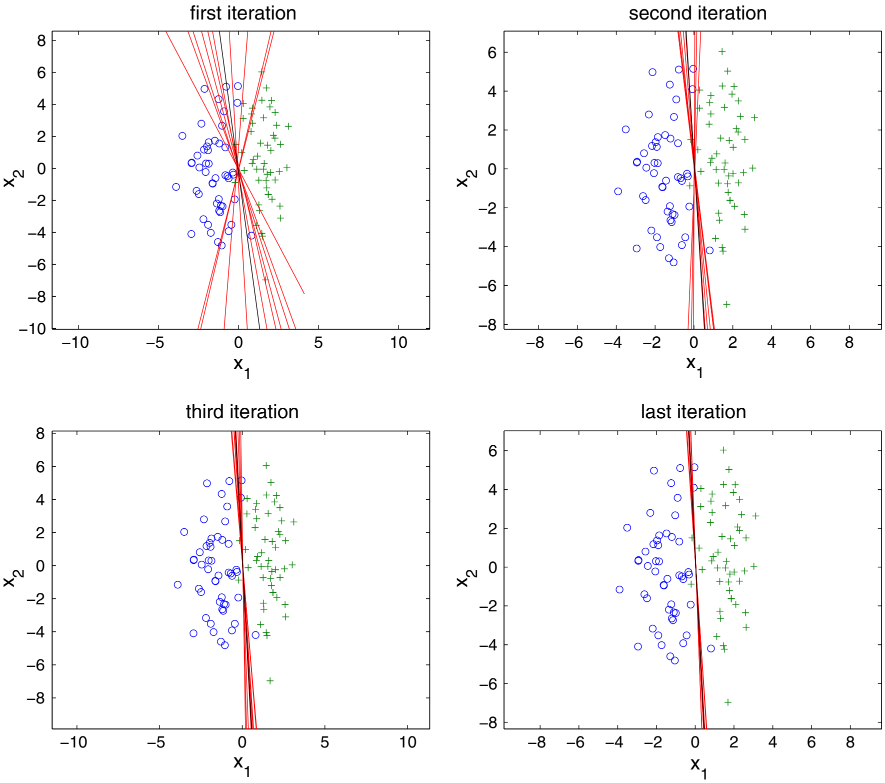

Fig. 4 The iterative process of the problem

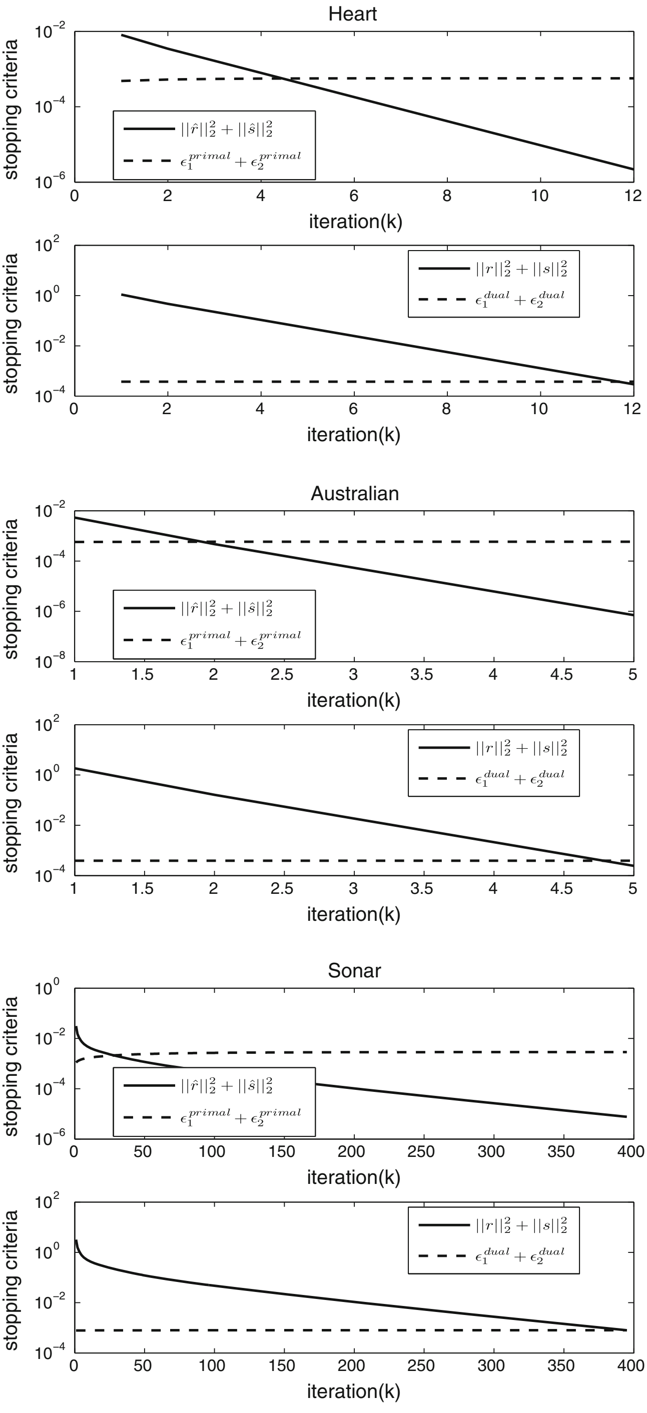

Fig. 5 The iterations of the three problems, including Heart, Australian and Sonar

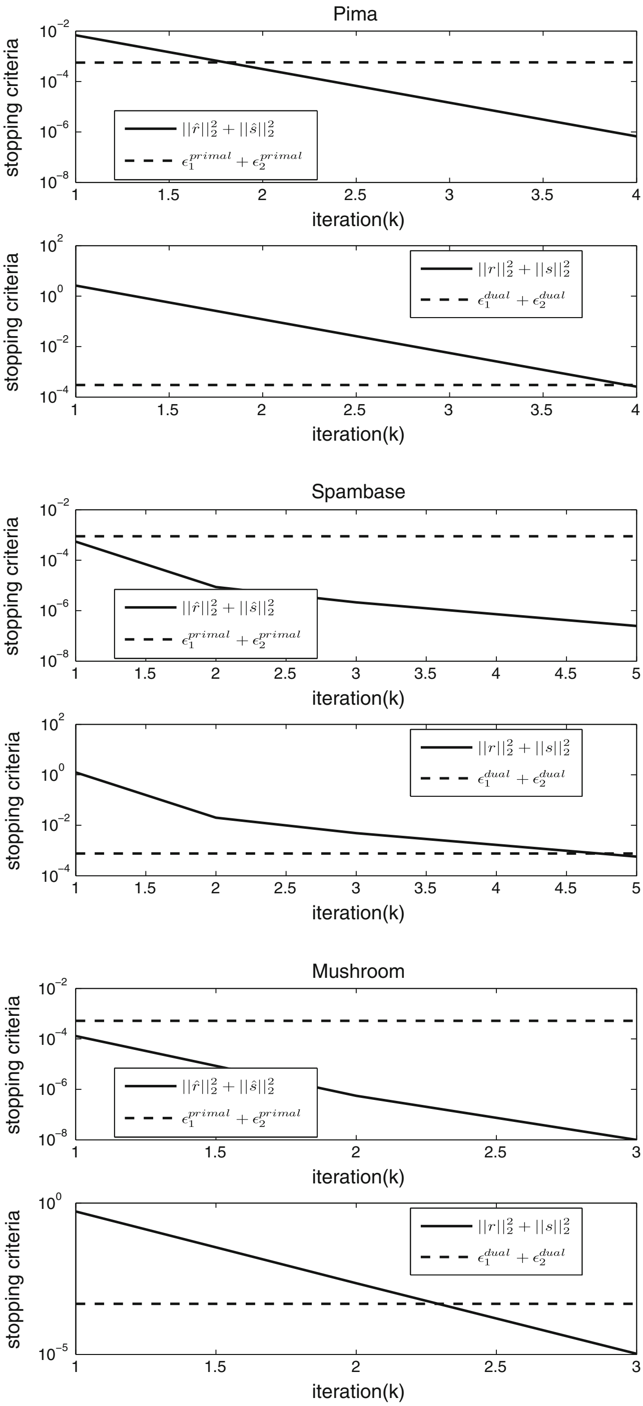

Fig. 6 The iterations of the three problems, including Pima, Spambase, and Mushroom

The numerical results ℓ_1-PSVM and ℓ_p-PSVM are taken from [7], those of PSVM are from [11] and those of GEPSVM and SVM-Light are from [21], respectively.

The first column of the following tables lists the names of datasets. The second column lists the results obtained by our consensus PSVM and they are labeled in bold which are tried to emphasize the effectiveness.

From Table 3, we can see that our approaches succeed in getting the highest correctness among the approaches.

In addition, we give the iterations of our first approach in Fig. 5.

It is obvious that our methods outperform GEPSVM, PSVM, and SVM-light in Table 4. Likewise, we give the iterations of our first approach in Fig. 6.

Table 5 contains the execution time of our methods. From the average time, it is evident that our first approach is faster than PSVM.

Finally, we compare our second approach, i.e., consensus ℓ_1–ℓ_2 PSVM with ℓ_1-PSVM [7] in terms of more sparse solutions and higher classification accuracy. For each dataset, including Heart disease, Australian credit approval, and Sonar, we choose the appropriate parameter λ for consensus ℓ_1–ℓ_2 PSVM. Table 6 shows the numerical results, where ♯ means the number of zero variables in w^* of ℓ_1-PSVM and z^* of consensus ℓ_1–ℓ_2 PSVM.

From the experiments, we find that if $\lambda \to 0$ in consensus ℓ_1–ℓ_2 PSVM, the classification performance of the model is better. While $\lambda \to 1$ in consensus ℓ_1–ℓ_2 PSVM, more sparse solutions can be obtained. We also find that the assumption that the larger λ is, the better classification performance of consensus PSVM is not valid.

5 Conclusions and Future Research

We have proposed two consensus PSVMs for the classification problems, and the two consensus PSVMs have been solved by the ADMM. Furthermore, they have been implemented by the real-world data taken from the University of California, Irvine Machine Learning Repository (UCI Repository) and are compared with the existed models such as ℓ_1-PSVM, ℓ_p-PSVM, GEPSVM, PSVM, and SVM-light. Numerical results show that our models outperform others with the classification accuracy and the sparse solutions. Moreover, we can see that consensus ℓ_1–ℓ_2 PSVM succeeds in finding more sparse solutions with higher accuracy than ℓ_1-PSVM.

We considered the binary linear classification problems and investigated the numerical behaviors of two consensus PVSMs. Our future research will derive the analysis of computation complexity of ADMM for two models thoroughly. Furthermore, we will consider the multi-class classification and nonlinear classification. We presume that the classification performance of consensus PSVM is related to the characteristic structure of the sample points. Thus, our next research will also analyze the relationship among the selection of λ, the characteristic structure of datasets, and the classification performance of consensus PSVM.

The size of datasets used in our numerical test is not large-scale problems in which the dimension of the problems is about hundred millions or more. ADMM has a great advantage in large-scale problems, and it has been applied in [4]. We plan to do such a large-scale experiment under the parallel environment with a distributed system (including several computers) or a clustering in the next step. Thus, our methods can be verified and extended better.

References

1. Bagarinao, E., Kurita, T., Higashikubo, M., Inayoshi, H.: Adapting SVM image classifiers to changes in imaging conditions using incremental SVM: an application to car detection. Comput. Vis. **5996**, 363–372 (2010)
2. Bertsekas, D.P., Tsitsiklis, J.N.: Parallel and distributed computation: numerical methods. Prentice Hall, Englewood Cliffs (1989)
3. Bishop, C.M.: Pattern recognition and machine learning. Springer, Heidelberg (2007)
4. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends Mach. Learn. **3**, 1–122 (2010)
5. Chakrabarty, B., Li, B.G., Nguyen, V., Van Ness, R.A.: Trade classification algorithms for electronic communications network trades. J. Bank. Financ. **31**, 3806–3821 (2007)
6. Chandrasekar, V., Keränen, R., Lim, S., Moisseev, D.: Recent advances in classification of observations from dual polarization weather radars. Atmos. Res. **119**, 97–111 (2013)
7. Chen, W.J., Tian, Y.J.: l_p-norm proximal support vector machine and its applications. Proc. Comput. Sci. **1**, 2417–2423 (2012)
8. Chew, H.G., Crisp, D., Bogner, R.E. et al.: Target detection in radar imagery using support vector machines with training size biasing. In: Proceedings of the sixth international conference on control, Automation, Robotics and Vision, Singapore (2000)
9. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**, 273–297 (1995)
10. Forero, P.A., Cano, A., Giannakis, G.B.: Consensus-based distributed support vector machines. J. Mach. Learn. Res. **11**, 1663–1707 (2010)
11. Fung, G.M., Mangasarian, O.L.: Proximal support vector machine classifiers. Proc. Knowledge Discovery and Data Mining, Asscociation for Computing Machinery, NewYork:77–86 (2001)
12. Han, J.W., Kamber, M., Pei, J.: Data mining: Concepts and techniques, third edition. Morgan Kaufmann, Burlington (2011)
13. He, B.S., Tao, M., Yuan, X.M.: Alternating direction method with gaussian back substitution for separable convex programming. SIAM J. Optim. **22**, 313–340 (2011)
14. He, B.S., Zhou, J.: A modified alternating direction method for convex minimization problems. Appl. Math. Lett. **13**, 123–130 (2000)
15. Hernandez, J.C.H., Duval, B., Hao, J.K.: SVM-based local search for gene selection and classification of microarray data. Bioinform. Res. Dev. **13**, 499–508 (2008)
16. Imam, T., Ting, K.M., Kamruzzaman, J.: z-SVM: an SVM for improved classification of imbalanced data. Adv. Artif. Intell. **4304**, 264–273 (2006)
17. Jayadeva, Khemchandani, R., Chandra, S.: Twin support vector machines for pattern classification. IEEE Trans. Pattern Anal. Mach. Intell. **29**, 905–910 (2007)
18. Lee, Y.J., Mangasarian, O.L.: RSVM: reduced support vector machines. In Proceedings of the First SIAM International Conference on Data Mining, CD-ROM Proceedings, Chicago (2001)
19. Li, Y.K., Shao, X.G., Cai, W.S.: A consensus least squares support vector regression (LS-SVR) for analysis of near-infrared spectra of plant samples. Tanlan **72**, 217–222 (2007)
20. Lu, C.J., Shao, Y.E., Chang, C.L.: Applying ICA and SVM to mixture control chart patterns recognition in a process. Adv. Neural Netw. **6676**, 278–287 (2011)
21. Mangasarian, O.L., Wild, E.W.: Multisurface proximal support vector machine classification via generalized eigenvalues. IEEE Trans. Pattern Anal. Mach. Intell. **28**, 69–74 (2006)
22. Schölkopf, B., Platt, J., Shawe, T.J. et al.: Estimating the support of a high-dimensional distribution. Neural Comput. **13**, 1443–1471 (2001)
23. Schölkopf, B., Smola, A.J., Bartlett, P.: New support vector algorithms. Neural Comput. **12**, 1207–1245 (2000)
24. Suykens, J.A.K., Vandewalle, J.: Least squares support vector machine classifiers. Neural Process. Lett. **9**, 293–300 (1999)

25. Tax, D., Duin, R.: Support vector domain description. Pattern Recognit. Lett. **20**, 1191–1199 (1999)
26. Üstün, T.B.: International classification systems for health. International Encyclopedia of Public Health:660–668 (2008)
27. Vapnik, V.: Statistical learning theory. Wiley, New York (1998)