IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 22, 2018, accepted March 27, 2018, date of publication April 3, 2018, date of current version April 25, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2822546

Robust Nonparallel Proximal Support Vector Machine With Lp-Norm Regularization

XIAO-QUAN SUN[1], YI-JIAN CHEN[1], YUAN-HAI SHAO[2], CHUN-NA LI[1], AND CHANG-HUI WANG[3]

[1]Zhijiang College, Zhejiang University of Technology, Hangzhou 310024, China
[2]School of Economics and Management, Hainan University, Haikou 570228, China
[3]School of Electromechanical and Automotive Engineering, Yantai University, Yantai 264005, China

Corresponding author: Chun-Na Li (na1013na@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61703370 and Grant 61603338, in part by the Natural Science Foundation of Zhejiang Province under Grant LQ17F030003, Grant LY15F030013, and Grant LY18G010018, in part by the Natural Science Foundation of Hainan Province under Grant 118QN181, and in part by the Scientific Research Fund of Zhejiang Provincial Education Department under Grant Y201534889.

ABSTRACT As a useful classification method, generalized eigenvalue proximal support vector machine (GEPSVM) is recently studied extensively. However, it may suffer from the sensitivity to outliers, since the L2-norm is used as a measure distance. In this paper, based on the robustness of the L1-norm, we propose an improved robust L1-norm nonparallel proximal SVM with an arbitrary Lp-norm regularization (LpNPSVM), where $p > 0$. Compared with GEPSVM, the LpNPSVM is more robust to outliers and noise. A simple but effective iterative technique is introduced to solve the LpNPSVM, and its convergence guarantee is also given when $0 < p \leq 2$. Experimental results on different types of contaminated data sets show the effectiveness of LpNPSVM. At last, we investigate our LpNPSVM on a real spare parts inspection problem. Computational results demonstrate the effectiveness of the proposed method over the GEPSVM on all the noise data.

INDEX TERMS Classification algorithms, Lp-norm, regularization, robustness, support vector machines.

I. INTRODUCTION

Support vector machines (SVMs) [1], [2] are known as powerful tools for data classification. The standard support vector machine achieves classification by searching parallel hyperplanes with maximal margin between them. By dropping the parallelism condition, some nonparallel hyperplane classifiers were proposed. Comparing to the parallel SVMs, the nonparallel SVMs can cope with the XOR problem well, such as GEPSVM [3], [4], the twin support vector machine (TWSVM) [5], the multi-weight vector projection support vector machine (MVSVM) [9], the recursive projection twin support vector machine via within-class variance minimization (PTSVM) [10], the recursive least squares recursive projection twin support vector machine for classification (LSPTSVM) [11], [12], and the nonparallel hyperplane support vector machine (NHSVM) [6]–[8], etc.

Among them, GEPSVM is a successful representation, who aims to find two nonparallel hyperplanes such that each hyperplane is closest to one of the two classes and as far as possible from the other data set. In fact, GEPSVM is still one of the ongoing topics and is studied extensively from different aspects [13]–[22]. For example, Jayadeva *et al*. [24] extended binary GEPSVM to multiple category classification and further modified the MGEPSVM classifier to the fuzzy problem. Shao *et al*. [25] improved GEPSVM by replacing its generalized eigenvalue decomposition with a standard eigenvalue decomposition that resulted in simpler optimization problems with less computation cost and stronger generalization ability. Ye and Ye [26] raised a new algorithm via singular value decomposition to overcome the ill-conditioned problem that may happen in GEPSVM. By using the differential search algorithm, Marghny *et al.* [27] proposed an improved GEPSVM of fuzzy values named DSA-GEPSVM, and was further proposed a mixed kernel GEPSVM [28]. Manifold regularized proximal support vector machine via generalized eigenvalue (MRGEPSVM) was proposed in [29] by incorporating local geometry information within each class into GEPSVM through the regularization technique. GEPSVM was also recently extended to the the multi-view area [30].

However, all the above GEPSVMs are constructed based on the L2-norm. It is known that L2-norm is sensitive to outliers, since it will exaggerate the effect of the data point with large norm. As an effective way, L1-norm is usually used as a replacement of L2-norm to achieve robustness [31]–[36]. In particular, L1-norm proximal support vector machine (L1-NPSVM) was proposed in [37] by replacing the L2-norm terms with the corresponding L1-terms, as a robust improvement of GEPSVM. Further, Yan *et al*. [38] considered a regularized L1-norm based GEPSVM (L1-GEPSVM). L1-GEPSVM is demonstrated effective in binary classification, especially when noise problems are encountered. However, the regularization term is the L2-norm one, which many restrict the effectiveness of L1-GEPSVM.

In this paper, we propose an L1-norm based GEPSVM with arbitrary Lp-norm regularization (LpNPSVM), where $p > 0$, with the purpose to choose an appropriate model and improve the robustness of L1-GEPSVM further. In specific, LpNPSVM finds two nonparallel separating hyperplane such that each hyperplane is closest to the data points of its own class and meanwhile the furthest from the data points of other classes. Due to the employment of the Lp-norm for arbitrary p and its ratio formulation, LpNPSVM is not easily to be solved by traditional method. Here we construct a simple iterative algorithm to solve it. Moreover, we know when $p = 2$, LpNPSVM is L1-GEPSVM, and the convergence of L1-GEPSVM was proved. We will also investigate the convergence results when p lies in a certain range. In summary, the proposed LpNPSVM has the following characteristics:

(i) An Lp-norm regularization is considered in LpNPSVM instead of an L2-norm regularization term used in L1-GEPSVM, which makes LpNPSVM be more robust and can achieve better performance.

(ii) Though LpNPSVM contains an Lp-norm regularization and is of the ratio form, it can be solved through an effective iteration method, with each each iteration solving a strongly convex programming problem.

(iii) The theoretical convergence of the proposed algorithm is guaranteed when $0 < p \leq 2$.

(iv) The effectiveness and robustness of LpNPSVM are demonstrated by experiments on two artificial data sets, some UCI data sets and a real world problem.

The paper is organized as follows. Section II gives a brief review of the related work. Section III proposes our L1-norm based GEPSVM with arbitrary Lp-norm regularization (LpNPSVM) in the primal space and gives the corresponding theoretical analysis. Section IV makes experimental comparisons of our LpNPSVM with the related methods. Concluding remarks are given in Section V.

Notations are given as follows. We consider a binary classification problem in the n-dimensional real space $\mathbb{R}^n$. Given a training set $T = \{(\mathrm{x}_1, y_1), \ldots, (\mathrm{x}_m, y_m)\}$, where $\mathrm{x}_i \in \mathbb{R}^n$ is the input and $y_i \in \{-1, 1\}$ is the output, $i = 1, \ldots, m$. We further organize the m_1 inputs of Class 1 by matrix $X_1 \in \mathbb{R}^{n\times m_1}$ and the m_2 inputs of Class 2 by matrix $X_2 \in \mathbb{R}^{n\times m_2}$, where $m_1 + m_2 = m$. Our goal is to find a mapping from $\mathbb{R}^n \xrightarrow{d} \{-1, 1\}$ so that for each x, we can deduce its output by $d(\mathrm{x})$.

II. RELATED WORK

A. GEPSVM

The goal of GEPSVM [3] is to find two nonparallel hyperplanes

$$d_1(\mathrm{x}) = w_1^T \mathrm{x} + b_1 = 0 \text{ and } d_2(\mathrm{x}) = w_2^T \mathrm{x} + b_2 = 0 \quad (1)$$

to approximate data points of different classes, respectively. Each of the hyperplanes is generated such that it is close to the data points in one class while is far away from the data points in the other class. The optimization problems of GEPSVM are formulated as

$$\min_{(w_1,b_1)\neq 0} \frac{||w_1^T X_1 + e_1 b_1||_2^2}{||w_1^T X_2 + e_2 b_1||_2^2} \quad (2)$$

and

$$\min_{(w_2,b_2)\neq 0} \frac{||w_2^T X_2 + e_2 b_2||_2^2}{||w_2^T X_1 + e_1 b_2||_2^2}. \quad (3)$$

The above two problems can be solved as generalized eigenvalue problems. However, they may encounter the singularity problem when the involved matrices are semi-definite. Therefore, the Tikhonov regularization terms $\delta||(w_1, b_1)||_2^2$ and $\delta||(w_2, b_2)||_2^2$ are usually introduced, where $\delta > 0$ is a trade-off parameter.

Therefore, by defining $\overline{w}_1 = (w_1, b_1)^T$, $\overline{w}_2 = (w_2, b_2)^T$ and $\overline{X}_1 = (X_1, e_1)^T \in R^{(n+1)\times m_1}$, $\overline{X}_2 = (X_2, e_2)^T \in R^{(n+1)\times m_2}$, GEPSVM usually solves

$$\min_{\overline{w}_1\neq 0} \frac{||\overline{w}_1^T \overline{X}_1||_2^2 + \delta||\overline{w}_1||_2^2}{||\overline{w}_1^T \overline{X}_2||_2^2} \quad (4)$$

and

$$\min_{\overline{w}_2\neq 0} \frac{||\overline{w}_2^T \overline{X}_2||_2^2 + \delta||\overline{w}_2||_2^2}{||\overline{w}_2^T \overline{X}_1||_2^2}. \quad (5)$$

Then their solutions are the smallest eigenvalues of the generalized eigenvalue problems respectively:

$$\begin{aligned} (\overline{X}_1\overline{X}_1^T + \delta I)\overline{w}_1 &= \lambda \overline{X}_2\overline{X}_2^T \overline{w}_1, \quad \overline{w}_1 \neq 0, \\ (\overline{X}_2\overline{X}_2^T + \delta I)\overline{w}_2 &= \lambda \overline{X}_1\overline{X}_1^T \overline{w}_2, \quad \overline{w}_2 \neq 0, \end{aligned} \quad (6)$$

where I is the identity matrix of appropriate dimension.

B. L1-NORM PROXIMAL SUPPORT VECTOR MACHINE

Since the construction of GEPSVM is based on the L2-norm, it is sensitive to outliers. In order to alleviate the sensitivity, an L1-norm based proximal support vector machine was proposed in [37]:

$$\min_{\overline{w}_1\neq 0} \frac{||\overline{w}_1^T \overline{X}_1||_1}{||\overline{w}_1^T \overline{X}_2||_1} \quad (7)$$

and

$$\min_{\overline{w}_2\neq 0} \frac{||\overline{w}_2^T \overline{X}_2||_1}{||\overline{w}_2^T \overline{X}_1||_1}. \quad (8)$$

Since problems (7) and (8) involve the divisions of L1-norms, they are difficult to solve by traditional optimization technique. To solve the problems, [37] used a gradient ascending (GA) algorithm to a non-convex surrogate function, which is easy to be implemented, and is of the ability to decrease the objective of L1-NPSVM via each iteration. However, the non-convexity of the surrogate function may cause the miss of optimal solution. Besides, an appropriate step size is also needed.

C. L1-NORM GEPSVM

To give a more effective algorithm for problems (7) and (8), Yan *et al.* [38] proposed an L1-norm based GEPSVM by adding an L2-norm regularization term. The formulation is given as follows:

$$\min_{\overline{w}_1 \neq 0} \frac{||\overline{w}_1^T \overline{X}_1||_1 + \delta ||\overline{w}_1||_2^2}{||\overline{w}_1^T \overline{X}_2||_1} \tag{9}$$

and

$$\min_{\overline{w}_2 \neq 0} \frac{||\overline{w}_2^T \overline{X}_2||_1 + \delta ||\overline{w}_2||_2^2}{||\overline{w}_2^T \overline{X}_1||_1}, \tag{10}$$

where $\delta > 0$ is a tuning parameter.

Rather than using GA on the surrogate function as in L1-NPSVM, the above two problems are converted to strongly convex programming problems, and are solved through a simple iterative algorithm. The algorithm avoids the selection of appropriate step size, and is theoretically proved to be convergent to a local optimum.

III. L1-NORM NONPARALLEL PROXIMAL SUPPORT VECTOR MACHINE WITH LP-NORM REGULARIZATION

A. PROBLEM FORMULATION

For problems (4), (5) and (9), (10), the regularization terms in fact improve the stability and performance of GEPSVM and L1-NPSVM, respectively. However, regularization terms with different norms may affect the behaviors of GEPSVM and L1-GEPSVM. To make L1-GEPSVM can choose a more appropriate model with the purpose to improve the effectiveness, we here consider an L1-norm nonparallel proximal support vector machine with arbitrary Lp-norm regularization (LpNPSVM), $p > 0$, which is formulated as

$$\min_{\overline{w}_1 \neq 0} \frac{||\overline{w}_1^T \overline{X}_1||_1 + \delta ||\overline{w}_1||_p^p}{||\overline{w}_1^T \overline{X}_2||_1} \tag{11}$$

and

$$\min_{\overline{w}_2 \neq 0} \frac{||\overline{w}_2^T \overline{X}_2||_1 + \delta ||\overline{w}_2||_p^p}{||\overline{w}_2^T \overline{X}_1||_1}. \tag{12}$$

By varying p, LpNPSVM is adaptive to different data. After obtaining $\overline{w}_1$ and $\overline{w}_2$, a new data point $x \in \mathbb{R}^n$ is assigned to class i, $i = 1, 2$, according to which of the hyperplanes it is closer to, i.e.,

$$\text{Class x} = \min_{i=1,2} \frac{|w_i^T \mathrm{x} + b_i|}{||w_i||}. \tag{13}$$

B. THE SOLVING ALGORITHM OF LPNPSVM

We now give the solving algorithm of LpNPSVM (11) and (12). In the following, we will solve problem (11). Problem (12) can be solved similarly. We first write problem (11) as

$$\begin{aligned} &\min_{w} ||\overline{w}_1^T \overline{X}_1||_1 + \delta ||\overline{w}_1||_p^p \\ &\text{s.t. } ||\overline{w}_1^T \overline{X}_2||_1 = 1, \end{aligned} \tag{14}$$

which can be rewritten as

$$\begin{aligned} &\min_{w} \overline{w}_1^T \Big(\sum_{j=1}^{m_1} \frac{\overline{X}_{1j}^T \overline{X}_{1j}}{|\overline{w}_1^T \overline{X}_{1j}|} + \sum_{k=1}^{n} \frac{\delta I}{|\overline{w}_1|_k^{2-p}} \Big) \overline{w}_1 \\ &\text{s.t. } \sum_{l=1}^{m_2} N_i \text{sign}(\overline{w}_1^T \overline{X}_{2l}) \overline{X}_{2l} v = 1, \end{aligned} \tag{15}$$

where $\overline{X}_{1j}$ is the j-th column of $\overline{X}_1$, $j = 1, 2, \ldots, m_1$, and $\overline{X}_{2l}$ is the l-th column of $\overline{X}_2$, $l = 1, 2, \ldots, m_2$. Denote

$$H^{(t)} = \sum_{j=1}^{m_1} \frac{\overline{X}_{1j}^T \overline{X}_{1j}}{|(\overline{w}_1^{(t)})^T \overline{X}_{1j}|} + \sum_{k=1}^{d_1} \frac{\delta I}{|\overline{w}_1^{(t)}|_k^{2-p}}, \tag{16}$$

$$h^{(t)} = \sum_{l=1}^{m_2} N_i \text{sign}((\overline{w}_1^{(t)})^T \overline{X}_{2l}) \overline{X}_{2l}^T. \tag{17}$$

Then we solve the following problem to get $\overline{w}_1^{(t+1)}$:

$$\begin{aligned} &\min_{\overline{w}_1} \overline{w}_1^T H^{(t)} \overline{w}_1 \\ &\text{s.t. } (h^{(t)})^T \overline{w}_1 = 1. \end{aligned} \tag{18}$$

In specific, let the Lagrangian of (18) be

$$L(\overline{w}_1, \lambda) = \overline{w}_1^T H^{(t)} \overline{w}_1 - \lambda((h^{(t)})^T \overline{w}_1 - 1). \tag{19}$$

Then the KKT condition is

$$H^{(t)} \overline{w}_1 - \lambda h^{(t)} = 0, \tag{20}$$

$$(h^{(t)})^T \overline{w}_1 - 1 = 0. \tag{21}$$

From (20), we get $\overline{w}_1 = \lambda (H^{(t)})^{-1} h^{(t)}$. By combining (21), we have $\lambda (h^{(t)})^T (H^{(t)})^{-1} h^{(t)} = 1$, which gives $\lambda = ((h^{(t)})^T (H^{(t)})^{-1} h^{(t)})^{-1}$. Therefore,

$$\overline{w}_1 = \frac{(H^{(t)})^{-1} h^{(t)}}{(h^{(t)})^T (H^{(t)})^{-1} h^{(t)}}. \tag{22}$$

C. THE CONVERGENCE OF ALGORITHM 1

In this subsection, we discuss the convergence of Algorithm 1 when p takes certain values.

Define

$$\begin{aligned} f(\overline{w}_1) = &\frac{1}{2} \sum_{j=1}^{m_1} \frac{\overline{w}_1{}^T \overline{X}_{1j} \overline{X}_{1j}^T \overline{w}_1}{|(\overline{w}_1{}^{(t)})^T \overline{X}_{1j}|} + \frac{p}{2} \cdot \delta \cdot \overline{w}_1{}^T diag\Big(\frac{1}{|\overline{w}_1{}^{(t)}|_k^{2-p}}\Big) \overline{w}_1 \\ &+ \frac{1}{2} \sum_{j=1}^{m_1} |(\overline{w}_1{}^{(t)})^T \overline{X}_{1j}| + \Big(1 - \frac{p}{2}\Big) \delta ||\overline{w}_1{}^{(t)}||_p^p. \end{aligned} \tag{23}$$

Then the following results hold.

Algorithm 1 LpNPSVM algorithm

I. Input: The training input data matrix $\overline{X}_1$, $\overline{X}_2$, parameters
$\delta > 0$, test data sample x.
II. Process: The training input data matrix $\overline{X}_1$, $\overline{X}_2$, parameters
$\delta > 0$, test data sample x.
1. Initialization. Set the iteration number $t = 0$ and initialize
$\overline{w}_1^{(0)}$ and $\overline{w}_2^{(0)}$ as random vectors.
2. Repeat
(a) Compute $\overline{w}_1^{(t+1)} = (w_1^{(t+1)}, b_1^{(t+1)})^T$ according to (22), and compute $\overline{w}_2^{(t+1)} = (w_2^{(t+1)}, b_2^{(t+1)})^T$ similarly.
(b) set $t = t + 1$.
Until Convergence
III. Output: $\overline{w}_1^* = \overline{w}_1^{(t+1)}$, $\overline{w}_2^* = \overline{w}_2^{(t+1)}$.
IV. Predict: Assign x according to (13).

Lemma 1: For any $p > 0$,

$$f(\overline{w}_1{}^{(t+1)}) \leq f(\overline{w}_1{}^{(t)}). \tag{24}$$

Proof: Since $\overline{w}_1{}^{(t+1)}$ minimizes (15), we have

$$\begin{aligned}
&\frac{1}{2}\sum_{j=1}^{m_1}\frac{(\overline{w}_1{}^{(t+1)})^T\overline{X}_{1j}\overline{X}_{1j}^T\overline{w}_1{}^{(t+1)}}{|(\overline{w}_1{}^{(t)})^T\overline{X}_{1j}|}\\
&\quad+\frac{p}{2}\delta(\overline{w}_1{}^{(t+1)})^T diag(\frac{1}{|\overline{w}_1{}^{(t)}|_k^{2-p}})\overline{w}_1{}^{(t+1)}\\
&\leq \frac{1}{2}\sum_{j=1}^{m_1}\frac{(\overline{w}_1{}^{(t)})^T\overline{X}_{1j}\overline{X}_{1j}^T\overline{w}_1{}^{(t)}}{|(\overline{w}_1{}^{(t)})^T\overline{X}_{1j}|}\\
&\quad+\frac{p}{2}\delta(\overline{w}_1{}^{(t)})^T diag(\frac{1}{|\overline{w}_1{}^{(t)}|_k^{2-p}})\overline{w}_1{}^{(t)}.
\end{aligned} \tag{25}$$

By adding the term

$$\frac{1}{2}\sum_{j=1}^{m_1}|(\overline{w}_1{}^{(t)})^T\overline{X}_{1j}| + (1-\frac{p}{2})\delta||\overline{w}_1{}^{(t)}||_p^p \tag{26}$$

to both sides of (25), and from the definition of $f(\overline{w}_1)$, we get (24). □

Lemma 2: For $0 < p \leq 2$,

$$||(\overline{w}_1{}^{(t+1)})^T\overline{X}_1||_1+\delta||\overline{w}_1{}^{(t+1)}||_p^p\leq||(\overline{w}_1{}^{(t)})^T\overline{X}_1||_1+\delta||\overline{w}_1{}^{(t)}||_p^p. \tag{27}$$

Proof: We first need the following equality in [39]: for any vector $v = (v_1, \ldots, v_{n+1})^T \in \mathbb{R}^{n+1}$,

$$||v||_1 = \min_{u\in\mathbb{R}_+^{n+1}} \frac{1}{2}\sum_{j=1}^{m_1}\frac{v_j^2}{u_j} + \frac{1}{2}||u||_1. \tag{28}$$

The minimum is uniquely achieved at $u_j = |v_j|$ for $j = 1, \ldots, m_1$, where $u = (u_1, \ldots, u_{n+1})^T \in \mathbb{R}^{n+1}$.

Therefore, we have

$$\begin{aligned}
||(\overline{w}_1{}^{(t+1)})^T\overline{X}_1||_1 &\leq \frac{1}{2}\sum_{j=1}^{m_1}\frac{(\overline{w}_1{}^{(t+1)})^T\overline{X}_{1j}\overline{X}_{1j}^T(\overline{w}_1{}^{(t+1)})^T}{|(\overline{w}_1{}^{(t)})^T\overline{X}_{1j}|}\\
&\quad+\frac{1}{2}\sum_{j=1}^{m_1}|(\overline{w}_1{}^{(t)})^T\overline{X}_{1j}|.
\end{aligned} \tag{29}$$

We also note that when $0 < p \leq 2$, for arbitrary vectors α and $\beta \neq 0$,

$$||\alpha||_p^p \leq \frac{p}{2}\alpha^T \mathrm{diag}(|\beta|^{p-2})\alpha + (1-\frac{p}{2})||\beta||_p^p, \tag{30}$$

where $|\beta|^{p-2}$ is the element-wise power of the absolute value of β [40]. By taking $\alpha = \overline{w}_1{}^{(t+1)}$ and $\beta = \overline{w}_1{}^{(t)}$, we have

$$\begin{aligned}
\delta||\overline{w}_1{}^{(t+1)}||_p^p &\leq \frac{p}{2}\cdot\delta(\overline{w}_1{}^{(t+1)})^T diag(\frac{1}{|\overline{w}_1{}^{(t)}|_k^{2-p}})\overline{w}_1{}^{(t+1)}\\
&\quad+(1-\frac{p}{2})\delta||\overline{w}_1{}^{(t)}||_p^p.
\end{aligned} \tag{31}$$

By combining (29) and (31), and considering the definition of $f(\overline{w}_1{}^{(t+1)})$, we get

$$||(\overline{w}_1{}^{(t+1)})^T\overline{X}_1||_1 + \delta||\overline{w}_1{}^{(t+1)}||_p^p \leq f(\overline{w}_1{}^{(t+1)}). \tag{32}$$

From Lemma 1, we know

$$||(\overline{w}_1{}^{(t+1)})^T\overline{X}_1||_1 + \delta||\overline{w}_1{}^{(t+1)}||_p^p \leq f(\overline{w}_1{}^{(t)}). \tag{33}$$

Since

$$f(\overline{w}_1{}^{(t)}) = ||(\overline{w}_1{}^{(t)})^T\overline{X}_1||_1 + \delta||\overline{w}_1{}^{(t)}||_p^p, \tag{34}$$

it follows (27) holds. □

Now we have the convergence of Algorithm 1 when $0 < p \leq 2$.

Proposition 3: When $0 < p \leq 2$, Algorithm 1 monotonically decreases the objective of problem (11) in each iteration, and hence converges.

Proof: We need only to prove

$$\sum_{i=1}^{c}N_i\|(w^{(t+1)})^T V_i\|_1 \geq \sum_{i=1}^{c}N_i\|(w^{(t)})^T V_i\|_1. \tag{35}$$

Since $w^{(t+1)}$ satisfies the condition of problem (15), we have

$$\sum_{i=1}^{c}\sum_{k=1}^{d_1}N_i\mathrm{sign}((w^{(t)})^T V_{ik})V_{ik}^T w^{(t+1)} = 1. \tag{36}$$

Since $w^{(t)}$ is in the feasible region, it follows

$$\sum_{i=1}^{c}\sum_{k=1}^{d_1}N_i\mathrm{sign}((w^{(t)})^T V_{ik})V_{ik}^T w^{(t)} = \sum_{i=1}^{c}N_i\|(w^{(t)})^T V_i\|_1 = 1. \tag{37}$$

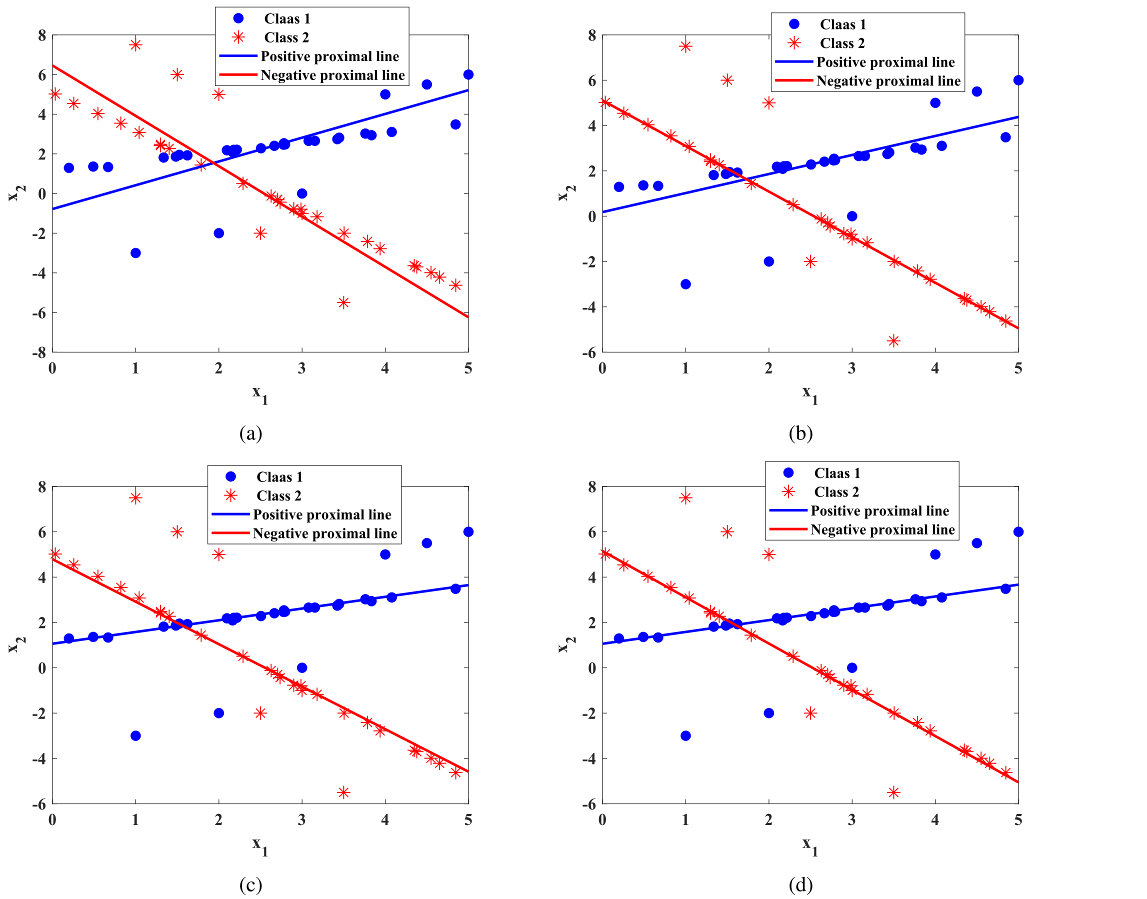


FIGURE 1. Artificial data set 1 with outliers learned by GEPSVM, L1-NPSVM, L1GEPSVM and LpNPSVM, respectively. (a) GEPSVM. (b) L1-NPSVM. (c) L1GEPSVM. (d) LpNPSVM ($p = 1$).

TABLE 1. Classification results for GEPSVM, L1-NPSVM, L1-GEPSVM and LpNPSVM on artificial data set 1.

Method	GEPSVM	L1-NPSVM	LpNPSVM ($p = 0.5$)	LpNPSVM ($p = 1$)	LpNPSVM ($p = 1.5$)	LpNPSVM ($p = 2$)	LpNPSVM ($p = 5$)
Accuracy	88.33 10^{-2}	89.71 0.01	95.00 10^{-2}	95.00 10^{-2}	93.33 10^{-1}	93.33 10^{-3}	93.33 10^{-6}

TABLE 2. Classification results for GEPSVM, L1-NPSVM, L1-GEPSVM and LpNPSVM on artificial data set 2.

Method	GEPSVM	L1-NPSVM	LpNPSVM ($p = 0.5$)	LpNPSVM ($p = 1$)	LpNPSVM ($p = 1.5$)	LpNPSVM ($p = 2$)	LpNPSVM ($p = 5$)
Accuracy	97.03 10^{-1}	99.17 0.005	99.17 10^{-2}	99.38 10^{-3}	99.38 10^{1}	98.96 10^{-1}	98.54 10^{-6}

Therefore,

$$\sum_{i=1}^{c} N_i \|(w^{(t+1)})^T V_i\|_1 \geq \sum_{i=1}^{c} \sum_{k=1}^{d_1} \text{sign}((w^{(t)})^T V_{ik}) V_{ik}^T w^{(t+1)} = 1 = \sum_{i=1}^{c} N_i \|(w^{(t)})^T V_i\|_1 \tag{38}$$

by (37).

Consequently,

$$\begin{aligned} J_0(w^{(t+1)}) &= \frac{\sum_{i=1}^{c} \sum_{j=1}^{N_i} ||(w^{(t+1)})^T \overline{X}_1||_1 + \delta ||w^{(t+1)}||_p^p}{\sum_{i=1}^{c} N_i \|(w^{(t+1)})^T V_i\|_1} \\ &\leq \frac{\sum_{i=1}^{c} \sum_{j=1}^{N_i} ||(w^{(t)})^T \overline{X}_1||_1 + \delta ||w^{(t)}||_p^p}{\sum_{i=1}^{c} N_i \|(w^{(t)})^T V_i\|_1} \\ &= J_0(w^{(t)}) \end{aligned} \tag{39}$$

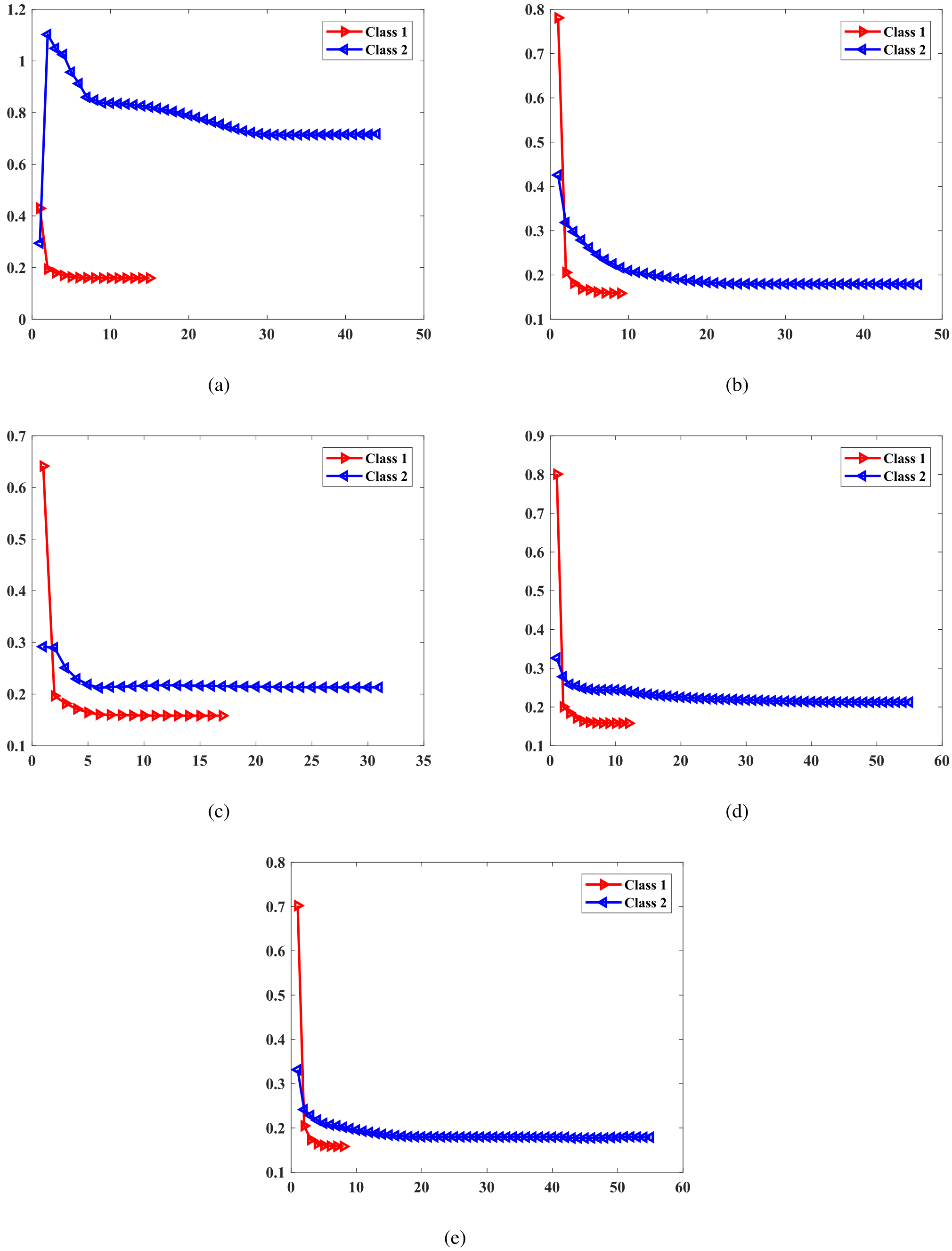


FIGURE 2. Objective variation along the iteration numbers for LpNPSVM on artificial data set 1. (a) p=0.5. (b) p=1. (c) p=1.5. (d) p=2. (e) p=5.

can be established by combining Lemma 2 and (35), and Algorithm 1 monotonically decreases the objective of problem (11) in each iteration when $0 < p \leq 2$. Since problem (11) is obviously lower bounded, Algorithm 1 converges when $0 < p \leq 2$. □

IV. EXPERIMENTAL RESULTS

In this section, we experimentally compare LpNPSVM with GEPSVM [3], L1-NPSVM [37] and L1GEPSVM [38]. For our LpNPSVM, p is taken as 0.5, 1, 1.5, 2 and 5. Note that when $p = 2$, our LpNPSVM is just L1GEPSVM. Experiments are conducted on two artificial data sets with outliers, 8 UCI benchmark data sets [41] and a real world problem. All these methods are implemented in MATLAB R2017b environment on a PC with Intel i5 processor (3.30 GHz) with 4 GB RAM, and the involved generalized eigenvalue problems are implemented by using a simple MATLAB function like "eig". The parameter in GEPSVM and the parameter

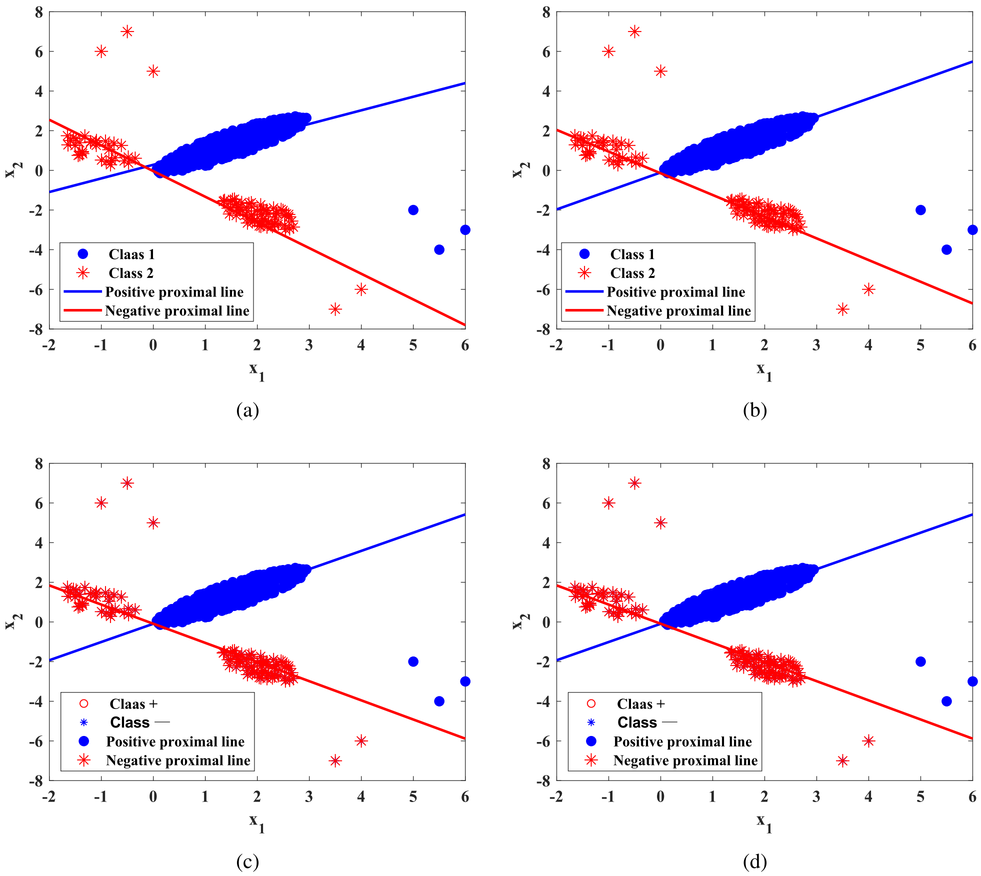


FIGURE 3. Artificial data set 2 with outliers learned by GEPSVM, L1-NPSVM, L1GEPSVM and LpNPSVM respectively. (a) GEPSVM. (b) L1-NPSVM. (c) L1GEPSVM. (d) LpNPSVM ($p = 1.5$).

in our LpNPSVM are selected from the set $\{10^{-6}, \ldots, 10^{3}\}$, and the learning rate in L1-NPSVM is chosen from the set $\{0.0005, 0.001, 0.005, 0.01, 0.05\}$.

A. ARTIFICIAL DATA SETS

1) ARTIFICIAL DATA SET 1

In order to see the effectiveness of our proposed LpNPSVM, we first apply it to a simple two-dimensional artificial "Cross Planes" data set, which was usually used in nonparallel SVMs to indicate their classification abilities [3], [42]. The "Cross Planes" data set contains 2 classes, with Class 1 and Class 2 having 30 data points, respectively. Class 1 is generated from the line $y = 0.5x$ with variance 0.2, and Class 2 is generated from the line $y = -2x$ with variance 0.2. To investigate the robustness of the proposed method, 6 and 5 outliers are added in these two-classes, respectively. Class 1 is represented by the red "*", and Class 2 is represented by the blue solid "o". Figure 1 (a)-(d) show the above data set and the proximal lines generated by GEPSVM, L1-NPSVM, L1GEPSVM and our LpNPSVM, respectively.

Obviously, when no outliers are added, the slopes of the ideal proximal lines for Class 1 and Class 2 should be 0.5 and -2, respectively. When outliers are considered, a robust method should have little influence to the above directions. Therefore, we apply each method to this artificial data with outliers, and compute the difference between their proximal hyperplanes and the ideal one. In specific, for GEPSVM, we get the slope difference between the hyperplane of Class 1 and the ideal hyperlane is $\Delta_{GEPSVM+} = 0.6987$, and the one between the hyperplane of Class 2 and the ideal hyperlane is $\Delta_{GEPSVM-} = -0.5392$. Similarly, we get $\Delta_{L1-NPSVM+} = 0.3139$, $\Delta_{L1-NPSVM-} = -0.0142$, $\Delta_{L1GEPSVM+} = 0.0171$, $\Delta_{L1GEPSVM-} = 0.1247$, $\Delta_{LpNPSVM+} = 0.0215$, $\Delta_{LpNPSVM+} = -0.0392$. Here $p = 1$ is taken for our LpNPSVM.

TABLE 3. The UCI data sets information

Data set	Sample size	Number of features
Australian	690	14
CMC	1473	9
Haberman	306	3
Ionosphere	351	34
Sonar	208	60
TIC	5822	85
Spanbas	4601	57
TicTacToe	958	9

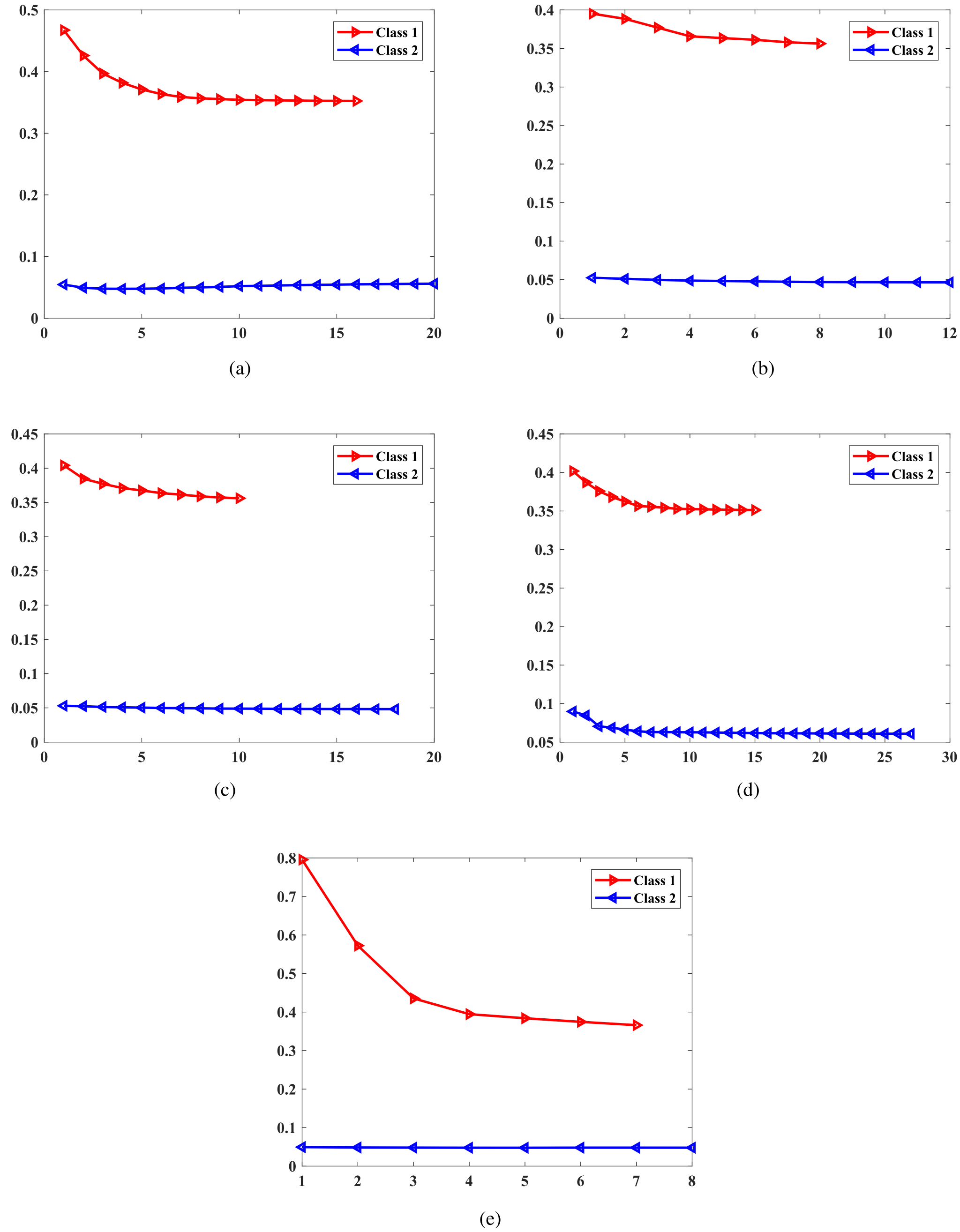


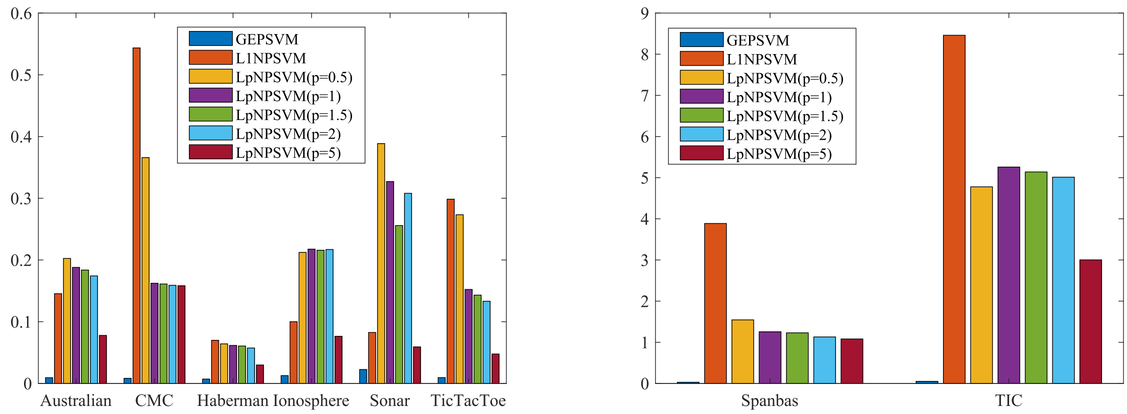
FIGURE 4. Objective variation along the iteration numbers for LpNPSVM on artificial data set 2. (a) p=0.5. (b) p=1. (c) p=1.5. (d) p=2. (e) p=5.

From the above results, and by observing Figure 1, we see GEPSVM drifts away the most from the ideal directions for both two classes. L1-NPSVM fits well for Class 2, but does not work well for Class 1. L1GEPSVM and LpNPSVM have the similar performance and both are better than GEPSVM and L1-NPSVM, but comparing to LpNPSVM, L1GEPSVM deviates a little for the negative class. To see the comparison more clearly, we also list the classification results in Table 1. The obtained proximal hyperplanes directions and the classification results suggest that LpNPSVM is more robust to outliers comparing to other methods.

To see the convergence of our LpNPSVM, we also give the two objectives value of our LpNPSVM for each iteration. The maximum iteration number is set to 100, and the tolerance is set to 10^{-5}. The corresponding results for different p are depicted in Figure 2. By observing the figure, we see though we only prove the convergent results for $0 < p \leq 2$, the experimental results show LpNPSVM is convergent for all these p on this data set.

TABLE 4. Classification results for GEPSVM, L1-NPSVM, L1-GEPSVM and LpNPSVM on the UCI data sets.

Method	GEPSVM	L1-NPSVM	LpNPSVM ($p=0.5$)	LpNPSVM ($p=1$)	LpNPSVM ($p=1.5$)	LpNPSVM ($p=2$)	LpNPSVM ($p=5$)
Items	Acc±std(%) Parameter	Acc±std(%) Parameter	Acc±std(%) Parameter	Acc±std(%) Parameter	Acc±std(%) Parameter	Acc±std(%) Parameter	Acc±std(%) Parameter
Australian	64.61±0.05 10^{-1}	78.26±0.04 0.05	71.30±0.09 10^{-3}	71.10±0.07 10^{1}	72.61±0.08 10^{1}	73.92 ±0.07 10^{3}	**85.56±0.05** 10^{-3}
CMC	50.66±0.06 10^{1}	71.58±0.05 0.05	77.26±0.18 10^{-5}	**77.40±0.03** 10^{-5}	77.39± 0.05 10^{-5}	77.39±0.05 10^{2}	77.39± 0.04 10^{0}
Haberman	72.08±0.11 10^{2}	73.23±0.21 0.01	75.21±0.07 10^{-2}	**77.03±0.06** 10^{2}	75.77±0.06 10^{-3}	74.80±0.06 10^{-6}	74.76±0.07 10^{-4}
Ionosphere	67.35±0.14 10^{-2}	68.57±0.16 0.005	74.22± 0.14 10^{-1}	73.89± 0.08 10^{-1}	**75.29±0.08** 10^{2}	74.88± 0.10 10^{-6}	75.09±0.15 10^{-2}
Sonar	61.55±0.12 10^{1}	65.37±0.12 0.005	62.61±0.13 10^{-5}	**67.73±0.11** 10^{0}	63.63±0.10 10^{1}	65.61±0.10 10^{2}	67.55±0.06 10^{-3}
TicTacToe	76.87±0.28 10^{-3}	52.61± 0.06 0.01	92.18±0.18 10^{3}	**98.33±0.09** 10^{3}	**98.33±0.05** 10^{2}	**98.33±0.02** 10^{2}	98.01±0.01 10^{-4}
Spanbas	65.84±0.05 10^{-4}	60.09± 0.11 0.001	65.04±0.14 10^{2}	63.26± 0.11 10^{0}	70.66±0.11 10^{2}	69.03±0.08 10^{1}	**72.33±0.08** 10^{0}
TIC	84.29±0.36 10^{1}	89.40±0.22 0.0005	93.65± 0.38 10^{-2}	90.88±0.36 10^{-5}	85.02 ±0.36 10^{-3}	93.92±0.36 10^{0}	**94.02±0.09** 10^{2}



(a) (b)

FIGURE 5. The time expenses for GEPSVM, L1-GEPSVM and LpNPSVM on the UCI data sets. (a) The first 6 data sets. (b) The last 2 data sets.

2) ARTIFICIAL DATA SET 2

We now apply LpNPSVM on the second artificial data set. The data set contains 480 data sets from 2 classes. Class 1 is generated by an ellipse of 376 data points, with its center, major axis and minor axis being (1.5, 1.3), 2 and 0.5, respectively. The direction between its major axis direction and the x-axis is $\pi/4$. Class 2 is generated by two ellipses of 104 data points, with major axis and minor axis being 1 and 0.5, and centers being (2, -2) and (-1, 1), respectively. The direction between its major axis direction and the x-axis is $3\pi/4$. The above two classes both contain outliers, as shown in Figure 3.

Figure 3 also shows the proximal hyperplane directions obtained on this data set by each method. As in artificial data set 1, we also compute the direction differences between their proximal hyperplanes and the ideal one, and get the following results: $\Delta_{GEPSVM+} = -0.3135$, $\Delta_{GEPSVM-} = -0.2942$, $\Delta_{L1-NPSVM+} = -0.0681$, $\Delta_{L1-NPSVM-} = -0.0941$, $\Delta_{L1GEPSVM+} = -0.1347$, $\Delta_{L1GEPSVM-} = 0.0539$, $\Delta_{LpNPSVM+} = -0.0811$, $\Delta_{LpNPSVM+} = 0.0341$. Here $p = 0.5$ is taken for our LpNPSVM. We also give the classification result for each method in Table 2. The above results clearly reveal that L1-NPSVM, L1GEPSVM and our LpNPSVM all possess robustness, while our LpNPSVM performs better than L1GEPSVM. Note the fact that when $p = 2$, LpNPSVM is just L1GEPSVM. This shows that the Lp-norm is necessary for our LpNPSVM.

Similar to the artificial data set 1, we also study the convergence of LpNPSVM on this artificial data set 2, and the corresponding results for different p are given in Figure 4. The results reveal that LpNPSVM is also convergent for all these p on the data set.

B. UCI BENCHMARK DATA SETS

We further investigate the behavior of our LpNPSVM on 8 benchmark data sets by conducting 10-fold cross validation. The data sets information is listed in Table 3. To test the

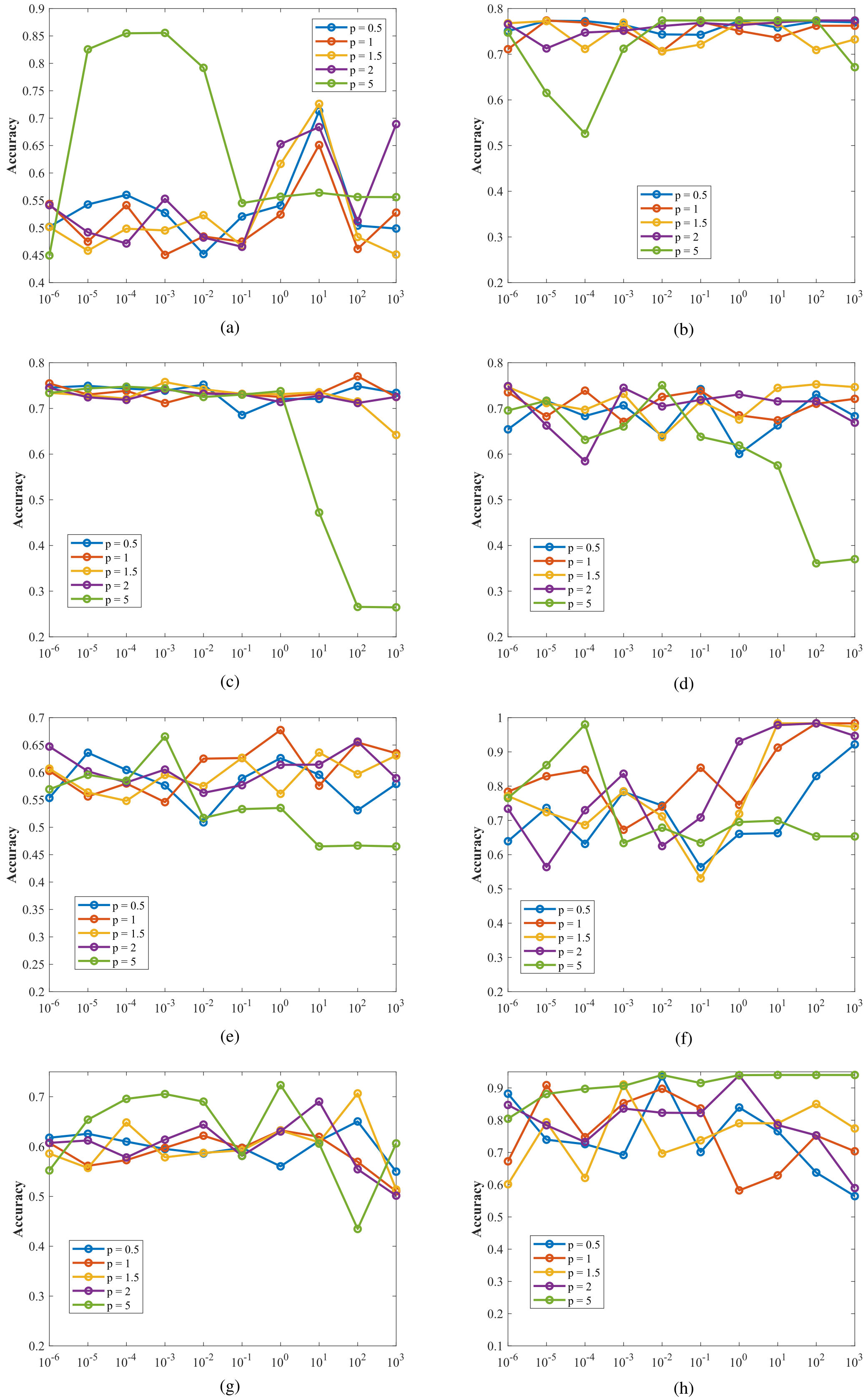


FIGURE 6. 10-fold classification accuracy (%) depending on the parameter δ for LpNPSVM on the UCI data sets. (a) Australian. (b) CMC. (c) Haberman. (d) Ionosphere. (e) Sonar. (f) TicTacToe. (g) Spanbas. (h) TIC.

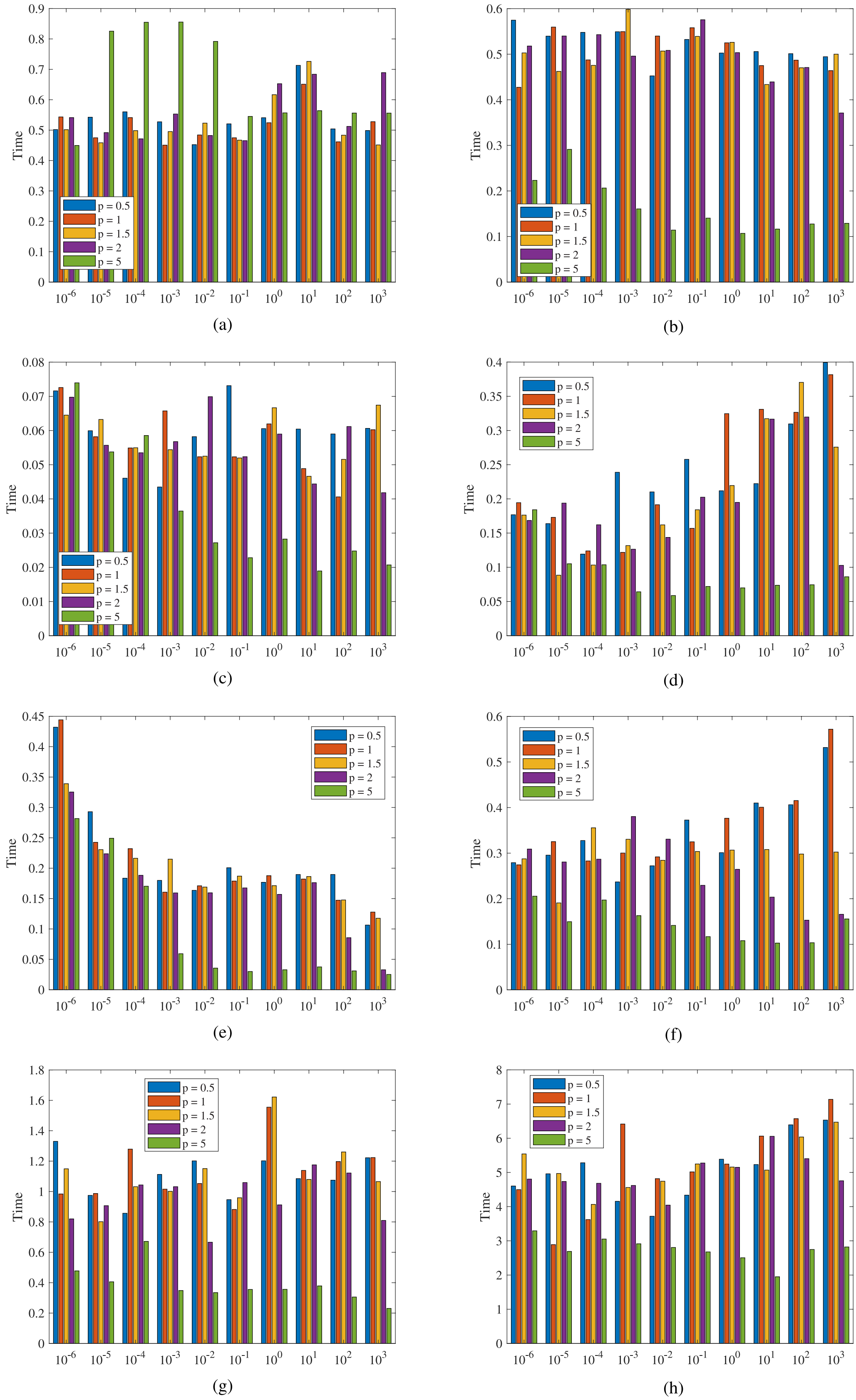


FIGURE 7. 10-fold classification computation time (s) depending on learning rate δ for LpNPSVM on the UCI data sets. (a) Australian. (b) CMC. (c) Haberman. (d) Ionosphere. (e) Sonar. (f) TicTacToe. (g) Spanbas. (h) TIC.

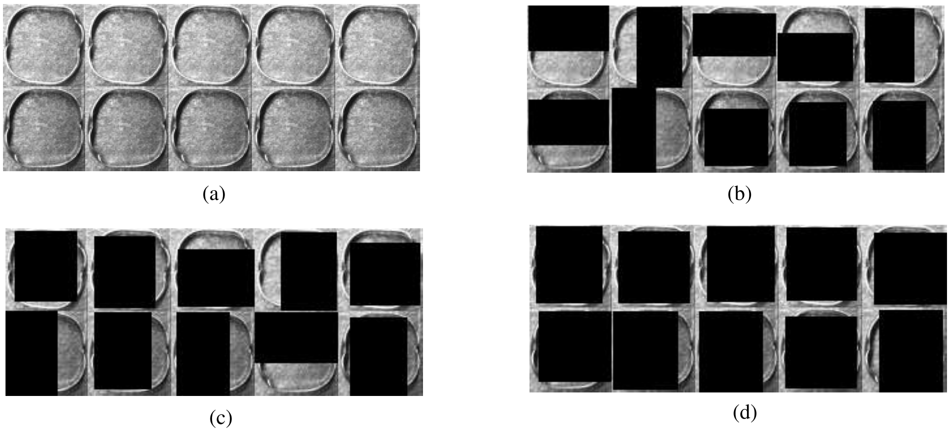


FIGURE 8. Sample images of the spare parts. (a) Original images. (b) 50% occluded images. (c) 60% occluded images. (d) 70% occluded images.

TABLE 5. Classification results for GEPSVM, L1-NPSVM, L1-GEPSVM and LpNPSVM on the spare parts inspection data set.

Method	GEPSVM	L1-NPSVM	LpNPSVM ($p = 0.5$)	LpNPSVM ($p = 1$)	LpNPSVM ($p = 1.5$)	LpNPSVM ($p = 2$)	LpNPSVM ($p = 5$)
50% block	47.27	54.55	97.27	**100**	**100**	54.55	77.27
	10^1	0.05	10^{-2}	10^{-3}	10^1	10^2	10^1
60% block	91.28	58.18	96.36	**98.18**	96.36	54.55	83.64
	10^2	0.05	10^1	10^2	10^{-4}	10^{-3}	10^2
70% block	45.45	50.91	**96.36**	88.18	83.64	45.45	74.55
	10^1	0.005	10^3	10^1	10^1	10^2	10^{-1}

robustness of various methods, we randomly choose 20% training data and add noise on it. In specific, random 20% features of the selected data are added with Gaussian noise of mean 0 and 0.05.

Firstly, we study the influence of parameter $\delta > 0$ to the performance and computation time of our LpNPSVM for different p. For each p, we first draw the accuracy variation along different δ, and the corresponding results are depicted in Figure 6. From the figure, we see as δ getting larger, $p = 5$ fluctuates the most. The highest accuracy is possessed by $p = 1$ on four data sets, while $p = 5$ has the best performance on three data sets. For different p, their optimal δ are normally different. The result shows that selecting optimal δ is necessary to achieve good accuracy.

We also investigate the influence of δ to the computation time of our LpNPSVM for different p, as shown in Figure 7. By observing the figure, we see for different data sets, their patterns may be different. For example, for the Ionosphere data set, the computation time first descends and then ascends, and for the Sonar data set, its computation time is basically descending. This may be because for some data sets, too small or too large δ may result in slow convergence speed, and hence increase computation time.

Now we compare the behavior of our LpNPSVM with GEPSVM and L1-NPSVM, and the corresponding classification results are listed in Table 4. In Table 4, the 10-fold mean accuracy and its standard derivation for each method are listed, and the best accuracy of each method is shown by bold figure. It is obvious that on all the data sets, our LpNPSVM under optimal p always owns the highest accuracy, and on almost all the data sets, LpNPSVM for all the p performs better than GEPSVM and L1-NPSVM. Moreover, on different data sets, their optimal p are different. For example, on the CMC data set, our LpNPSVM ($p = 5$) performs the best, while on the Haberman data set, LpNPSVM ($p = 1$) performs the best. The phenomenon shows the necessity of selecting proper p, and also demonstrates the improvement of our method to L1GEPSVM. Moreover, when p is small, for example, when $p = 0.5$, the standard derivation of LpNPSVM is larger than the ones when p is greater. This may be caused by the fact that problem (14) is nonconvex when $p < 1$. We also depict the time expense for all the methods. Since for the data sets Spanbas and TIC, their computation times are relatively higher comparing to other data sets, we draw their time expense separately as in Figure 5 (a) and (b). From the figures, we see as p getting larger, the time expense becomes lower for our LpNPSVM. Generally, L1-NPSVM has more computation time than our LpNPSVM, and GEPSVM runs the fast.

C. SPARE PARTS INSPECTION DATA SET

In this subsection, we consider a real world problem. In engineering area, the directions of placement of the spare parts are usually important. Figure 8(a) shows such a situation. The figure gives the images of a kind of spare parts. The first row of Figure 8(a) shows the upside direction of the spare parts, and Figure 8(a) shows the downside direction of the spare parts. Therefore, in this case, it is a binary classification problem. This real data set contains 220 data images, including 120 upside direction ones and 100 downside direction ones. Each image is of the size 32×32. We randomly choose 50% data images from each class to form the training set, while the rest form the test data set. In order to test the robustness of LpNPSVM, we artificially pollute the training images with random black rectangular at random position. The rectangular takes 50% of each image, as shown in Figure 8(b).

We now apply our LpNPSVM and other methods to the above data set, and list their corresponding accuracies in Table 5. From the table, we see GEPSVM is obviously affected seriously by the noise, and our LpNPSVM performs the best when $p = 0.5, 1, 1.5$. For our LpNPSVM. $p = 2$ behaves inferior to the other p. Note that when $p = 2$, our LpNPSVM is just L1GEPSVM. Therefore, we see LpNPSVM performs better than GEPSVM, L1NPSVM and L1GEPSVM. Moreover, the result also supports the fact that for our LpNPSVM, the selection of p is necessary.

V. CONCLUSION

In this paper, we have proposed an improved robust L1-norm nonparallel proximal support vector machine with Lp-norm regularization (LpNPSVM), which aims at giving a more robust performance to outliers for binary classification in contrast to L1-GEPSVM. LpNPSVM considered an arbitrary Lp-norm regularization term rather than an L1-norm term in L1-GEPSVM. The convergence of LpNPSVM when $0 < p \leq 2$ is also ensured. The robustness and effectiveness of the proposed method are supported by a series of numerical experiments on two artificial data sets, 8 UCI data sets and a real world data set. Our Matlab code and slide can be downloaded from http://www.optimal-group.org/Resources/Code/LpNPSVM.html. Since when $0 < q < 1$, the Lq-norm is more robust than the L1-norm, in the future, we will consider the Lq-norm GEPSVM with regularization term. Also, study the convergence of LpNPSVM for $p > 2$ is also interesting.

REFERENCES

[1] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[2] N. Y. Deng, Y. J. Tian, and C. H. Zhang, *Support Vector Machines: Optimization Based Theory, Algorithms, and Extensions*. Boca Raton, FL, USA: CRC Press, 2012.
[3] O. Mangasarian and E. Wild, "Multisurface proximal support vector machine classification via generalized eigenvalues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 69–74, Jan. 2006.
[4] Jayadeva, R. Khemchandani and S. Chandra, "Generalized eigenvalue proximal support vector machines," in *Twin Support Vector Machines*. Cham, Switzerland: Springer, 2017, pp. 25–42.
[5] Jayadeva, R. Khemchandani, and S. Chandra, "Twin support vector machines for pattern classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 905–910, May 2007.
[6] Y.-H. Shao, W.-J. Chen, and N.-Y. Deng, "Nonparallel hyperplane support vector machine for binary classification problems," *Inf. Sci.*, vol. 263, pp. 22–35, Apr. 2014.
[7] Y. Tian, Z. Qi, X. Ju, Y. Shi, and X. Liu, "Nonparallel support vector machines for pattern classification," *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1067–1079, Jul. 2014.
[8] S. Ding, X. Hua, and J. Yu, "An overview on nonparallel hyperplane support vector machine algorithms," *Neural Comput. Appl.*, vol. 25, no. 5, pp. 975–982, 2014.
[9] Q. Ye, C. Zhao, N. Ye, and Y. Chen, "Multi-weight vector projection support vector machines," *Pattern Recognit. Lett.*, vol. 31, no. 13, pp. 2006–2011, 2010.
[10] X. Chen, J. Yang, Q. Ye, and J. Liang, "Recursive projection twin support vector machine via within-class variance minimization," *Pattern Recognit.*, vol. 44, nos. 10–11, pp. 2643–2655, 2011.
[11] Y. Shao, N. Deng, and Z. Yang, "Least squares recursive projection twin support vector machine for classification," *Pattern Recognit.*, vol. 45, no. 6, pp. 2299–2307, 2012.
[12] S. Ding and X. Hua, "Recursive least squares projection twin support vector machines for nonlinear classification," *Neurocomputing*, vol. 130, pp. 3–9, Apr. 2014.
[13] M. R. Guarracino, C. Cifarelli, O. Seref, and P. M. Pardalos, "A classification method based on generalized eigenvalue problems," *Optim. Methods Softw.*, vol. 22, no. 1, pp. 73–81, 2007.
[14] J. Wan, Y. Ming, and G. Ji, "A new semi-supervised multi-surface proximal support vector machine model," in *Proc. IEEE Int. Conf. Artif. Intell. Comput. Intell. (AICI)*, Oct. 2010, pp. 510–514.
[15] P. Xanthopoulos, M. R. Guarracino, and P. M. Pardalos, "Robust generalized eigenvalue classifier with ellipsoidal uncertainty," *Ann. Oper. Res.*, vol. 216, no. 1, pp. 327–342, 2014.
[16] F. Dufrenois and J. C. Noyer, "Generalized eigenvalue proximal support vector machines for outlier description," in *Proc. IEEE Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2015, pp. 1–9.
[17] Z. Dong, G. Ji, J. Yang, Y. Zhang, and S. Wang, "Preclinical diagnosis of magnetic resonance (MR) brain images via discrete wavelet packet transform with Tsallis entropy and generalized eigenvalue proximal support vector machine (GEPSVM)," *Entropy*, vol. 17, no. 4, pp. 1795–1813, 2015.
[18] L. Zhang and P. N. Suganthan, "Oblique decision tree ensemble via multisurface proximal support vector machine," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2165–2176, Oct. 2015.
[19] Y. Chen, M. Yang, X. Chen, B. Liu, H. Wang, and S. Wang, "Sensorineural hearing loss detection via discrete wavelet transform and principal component analysis combined with generalized eigenvalue proximal support vector machine and Tikhonov regularization," *Multimedia Tools Appl.*, vol. 77, no. 3, pp. 3775–3793, 2017.
[20] M. R. Guarracino, M. Sangiovanni, G. Severino, G. Toraldo, and M. Viola, "On the regularization of generalized eigenvalues classifiers," in *Proc. AIP Conf.*, 2016, vol. 1776. no. 1, p. 040005.
[21] R. Khemchandani, K. Goyal, and S. Chandra, "Generalized eigenvalue proximal support vector regressor for the simultaneous learning of a function and its derivatives," *Int. J. Mach. Learn. Cybern.*, pp. 1–12, May 2017.
[22] M. Viola, M. Sangiovanni, G. Toraldo, and M. R. Guarracino, "Semi-supervised generalized eigenvalues classification," *Ann. Oper. Res.*, pp. 1–18, Oct. 2017.
[23] Z.-M. Yang, H.-J. Wu, C.-N. Li, and Y.-H. Shao, "Least squares recursive projection twin support vector machine for multi-class classification," *Int. J. Mach. Learn. Cybern.*, vol. 7, no. 3, pp. 411–426, 2016.
[24] Jayadeva, R. Khemchandai, and S. Chandra, "Fuzzy multi-category proximal support vector classification via generalized eigenvalues," *Soft Comput.*, vol. 11, no. 7, pp. 679–685, 2007.
[25] Y.-H. Shao, N.-Y. Deng, W.-J. Chen, and Z. Wang, "Improved generalized eigenvalue proximal support vector machine," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 213–216, Mar. 2013.
[26] Q. Ye and N. Ye, "Improved proximal support vector machine via generalized eigenvalues," in *Proc. IEEE Int. Joint Conf. Comput. Sci. Optim.*, Apr. 2009, pp. 705–709.
[27] M. H. Marghny and R. M. A. El-Aziz, "Differential search algorithm-based parametric optimization of fuzzy generalized eigenvalue proximal support vector machine," *Int. J. Comput. Appl.*, vol. 108, no. 19, pp. 38–46, 2015.
[28] M. H. Marghny and R. M. A. El-Aziz, "Data classification based on GEPSVM using backtracking search algorithm," *Data Mining Knowl. Eng.*, vol. 7, no. 2, pp. 89–94, 2015.
[29] J. Liang, F.-Y. Zhang, X.-X. Xiong, X.-B. Chen, L. Chen, and G.-H. Lan, "Manifold regularized proximal support vector machine via generalized eigenvalue," *Int. J. Comput. Intell. Syst.*, vol. 9, no. 6, pp. 1041–1054, 2016.
[30] S. Sun, X. Xie, and C. Dong, "Multiview learning with generalized eigenvalue proximal support vector machines," *IEEE Trans. Cybern.*, to be published, doi: 10.1109/TCYB.2017.2786719.

[31] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 9, pp. 1672–1680, Sep. 2008.

[32] H. Wang, Q. Tang, and W. Zheng, "L1-norm-based common spatial patterns," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 3, pp. 653–662, Mar. 2012.

[33] F. Zhong and J. Zhang, "Linear discriminant analysis based on L1-norm maximization," *IEEE Trans. Image Process.*, vol. 22, no. 8, pp. 3018–3027, Aug. 2013.

[34] C.-N. Li, Y.-H. Shao, and N.-Y. Deng, "Robust L1-norm two-dimensional linear discriminant analysis," *Neural Netw.*, vol. 65, pp. 92–104, May 2015.

[35] Q. Ye, J. Yang, F. Liu, C. Zhao, N. Ye, and T. Yin, "L1-norm distance linear discriminant analysis based on an effective iterative algorithm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 1, pp. 114–129, Jan. 2016.

[36] H. Yan *et al.*, "Least squares twin bounded support vector machines based on L1-norm distance metric for classification," *Pattern Recognit.*, vol. 74, pp. 434–447, Feb. 2017.

[37] C.-N. Li, Y.-H. Shao, and N.-Y. Deng, "Robust L1-norm non-parallel proximal support vector machine," *Optimization*, vol. 65, no. 1, pp. 169–183, 2016.

[38] H. Yan, Q. Ye, T. Zhang, D.-J. Yu, and Y. Xu, "L1-norm GEPSVM classifier based on an effective iterative algorithm for classification," *Neural Process. Lett.*, pp. 1–26, Sep. 2017.

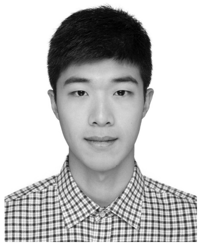
[39] R. Jenatton, G. Obozinski, and F. Bach, "Structured sparse principal component analysis," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 366–373.

[40] J. Wang, "Generalized 2-D principal component analysis by Lp-norm for image analysis," *IEEE Trans. Cybern.*, vol. 46, no. 3, pp. 792–803, Mar. 2016.

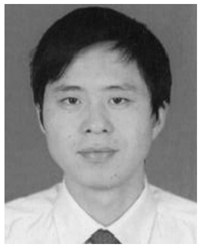
[41] C. L. Blake and C. J. Merz. (1998). *UCI Repository of Machine Learning Databases*. [Online]. Available: http://archive.ics.uci.edu/ml/index.php

[42] Y.-H. Shao, C.-H. Zhang, X.-B. Wang, and N.-Y. Deng, "Improvements on twin support vector machines," *IEEE Trans. Neural Netw.*, vol. 22, no. 6, pp. 962–968, Jun. 2011.

XIAO-QUAN SUN received the master's degree from the School of Software, Fudan University, China, in 2005. He is currently an Associate Professor with the Zhijiang College, Zhejiang University of Technology. His research interests include pattern recognition and signal detection.



YI-JIAN CHEN is currently pursuing the degree with the Zhijiang College, Zhejiang University of Technology. His research interests include data mining and machine learning.



YUAN-HAI SHAO received the master's degree in information and computing science from the College of Mathematics, Jilin University, the master's degree in applied mathematics, and the Ph.D. degree in operations research and management from the College of Science, China Agricultural University, China, in 2006, 2008, and 2011, respectively. He is currently a Professor with the School of Economics and Management, Hainan University. His research interests include data mining, machine learning, and optimization methods.

CHUN-NA LI received the master's and Ph.D. degrees from the Department of Mathematics, Harbin Institute of Technology, China, in 2009 and 2012, respectively. She is currently an Associate Professor with the Zhijiang College, Zhejiang University of Technology. Her research interests include data mining, machine learning, and optimization methods.

CHANG-HUI WANG received the master's degree from the Department of Mathematics, Harbin Institute of Technology, China, in 2009, and the Ph.D. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2016. He is currently a Lecturer with the School of Electromechanical and Automotive Engineering, Yantai University. His current research interests include automotive engine, system identification, adaptive control, and intelligent systems and control.