Polynomial Smooth Twin Support Vector Machines Based on Invasive Weed Optimization Algorithm

Shifei Ding[1, 2], Huajuan Huang[1, 2], Junzhao Yu[1, 2], Fulin Wu[1, 2]
School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China, 221116
Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, 100190
Email: dingsf@cumt.edu.cn

***Abstract*—Smoothing functions can transform the unsmooth twin support vector machines (TWSVM) into smooth ones, and thus better classification results can be obtained. It has been one of the key problems to seek a better smoothing function in this field for a long time. In this paper, a novel version for smooth TWSVM, termed polynomial smooth twin support vector machines (PSTWSVM), is proposed. In PSTWSVM, using the series expansion, a new class of polynomial smoothing is proposed, and then their important properties are discussed. It is shown that the approximation accuracy and smoothness rank of polynomial functions can be as high as required. Subsequently, the polynomial functions are used to convert the original constrained quadratic programming problems of TWSVM into unconstrained minimization problems, and then are solved by the well-known Newton-Armijo algorithm. Meanwhile, in order to find the suitable parameters of PSTWSVM, Invasive Weed Optimization (IWO) algorithm is used to optimize the proposed algorithm. Then we propose an algorithm called polynomial smooth twin support vector machines based on invasive weed optimization algorithm (PSTWSVM-IWO). Finally, the effectiveness of the proposed method is demonstrated via experiments on synthetic and UCI benchmark datasets.**

***Index Terms*—Polynomial function, Newton-Armijo, Invasive weed optimization algorithm, Parameter optimization, Twin support vector machines**

I. INTRODUCTION

Support vector machine (SVM) proposed by Vapnik and co-worker [1] is a computationally powerful kernel-based tool for binary data classification and regression. Because the theory of SVM is based on the idea of structural risk minimization principle, SVM has successfully solved the high dimensionality and local minimum problems. Therefore, compared with other machine learning methods, such as artificial neural network [2-3], SVM owns better generalization ability. Within a few years after its introduction SVM has played excellent performance in many real-world predictive data mining applications such as text categorization [4], time series prediction [5], pattern recognition [6] and image processing [7], etc.

Although SVM owns better generalization ability compared with many other machine learning methods, however, its computational complexity in training stage is too expensive. To address this problem, so far, many improved algorithms have been presented, such as chunking algorithm [8], decomposition algorithm [9] and sequential minimal optimization (SMO) [10], etc. However, these algorithms are too complex. On the other hand, many researchers have proposed some deformation algorithms based on standard SVM. For example, in 2006, Mangasarian et al. [11] proposed a nonparallel plane classifier for binary data classification, named generalized eigenvalue proximal support vector machine (GEPSVM). The essence of GEPSVM is to look for two nonparallel planes, so that data points of each class are proximal to one of them. GEPSVM has good learning speed, but its classification accuracy is low. In 2007, Jayadeva et al. [12] proposed a new machine learning method called twin support vector machine (TWSVM) for the binary classification in the spirit of GEPSVM. TWSVM would generate two non-parallel planes, such that each plane is closer to one of the two classes and is as far as possible from the other. In TWSVM, a pair of smaller sized quadratic programming problems (QPPs) are solved, instead of solving a single large one in SVM, makes the computational speed of TWSVM approximately 4 times faster than the traditional SVM. Because of its excellent performance, TWSVM has been applied to many areas such as speaker recognition [13], medical detection [14], etc.

Similar to SVM, TWSVM solves its QPPs in the dual space. However, this solving method will be affected by time and memory constraints when dealing with the large datasets, which would make the learning speed of TWSVM low. In order to address this problem, in 2008, M. Arun Kumar et al. [15] used the sigmoid function to approach the objective function of TWSVM and then proposed smooth twin support vector machines

This work is supported by the National Natural Science Foundation of China (Nos.61379101), the National Key Basic Research Program of China (No. 2013CB329502), and the Natural Science Foundation of Jiangsu Province (No.BK20130209).

Corresponding author, Shifei Ding, E-mail: dingsf@cumt.edu.cn.

doi:10.4304/jcp.9.5.1226-1233

(STWSVM). STWSVM directly solved QPPs in the original space instead of the dual space. Experimental results showed that STWSVM could make the classifier faster to compute in the classification phase than TWSVM. However, because of the low approximation ability of the sigmoid function, the classification accuracy of STWSVM was unsatisfactory. In order to further improve the classification performance of STWSVM, looking for a new smooth function with better approximation ability is the key problem.

In this paper, using the series expansion, a new class of polynomial smoothing is proposed. We have proved that the proposed smoothing functions have better smooth performance and their approximation accuracy can be as high as required. Subsequently, the polynomial functions are used to convert the original constrained quadratic programming problems of TWSVM into unconstrained minimization problems, and then are solved by the well-known Newton-Armijo algorithm. Based on the above idea, a novel version for smooth TWSVM, termed polynomial smooth twin support vector machines (PSTWSVM), is proposed in this paper. Besides, in order to overcome PSTWSVM parameters selection problem, we use Invasive Weed Optimization (IWO) algorithm [16] which has fast global searching ability to select PSTWSVM parameters, so that we would obtain the optimal parameters combination. Finally, the experimental results show the effectiveness and stability of the proposed method.

The paper is organized as follows: In section 2, we propose the PSTWSVM model and prove its global convergence. In section 3, PSTWSVM-IWO is detailed introduced and analyzed. Computational comparisons on synthetic and UCI datasets are done in section 4 and section 5 gives concluding remarks.

II. Polynomial Smooth Twin Support Vector Machines

A. Twin Support Vector Machines

Consider a binary classification problem of classifying m_1 data points belonging to class +1 and m_2 data points belonging to class -1. Then let matrix A in $R^{m_1\times n}$ represent the data points of class +1 while matrix B in $R^{m_2\times n}$ represent the data points of class -1. Two nonparallel hyper-planes of the linear TSVMs can be expressed as follows.

$$x^T w_1 + b_1 = 0 \quad \text{and} \quad x^T w_2 + b_2 = 0 \tag{1}$$

The target of TSVMs is to generate the above two nonparallel hyper-planes in the n-dimensional real space R^n, such that each plane is closer to one of the two classes and is as far as possible from the other. A new sample point is assigned to class +1 or -1 depending upon its proximity to the two nonparallel hyper-planes. The linear classifiers are obtained by solving the following optimization problems.

$$\begin{aligned}&\min_{w^{(1)},b^{(1)},\xi^{(2)}} \frac{1}{2}\left\|Aw^{(1)}+e_1b^{(1)}\right\|^2+c_1e_2^T\xi^{(2)}\\ &s.t.\quad -(Bw^{(1)}+e_2b^{(1)})\geq e_2-\xi^{(2)},\\ &\qquad\quad \xi^{(2)}\geq 0.\end{aligned} \tag{2}$$

$$\begin{aligned}&\min_{w^{(2)},b^{(2)},\xi^{(1)}} \frac{1}{2}\left\|Bw^{(2)}+e_2b^{(2)}\right\|^2+c_2e_1^T\xi^{(1)}\\ &s.t.\quad (Aw^{(2)}+e_1b^{(2)})\geq e_1-\xi^{(1)},\\ &\qquad\quad \xi^{(1)}\geq 0.\end{aligned} \tag{3}$$

where c_1 and c_2 are penalty parameters, $\xi^{(1)}$ and $\xi^{(2)}$ are slack vectors, e_1 and e_2 are vectors of ones of appropriate dimensions.

In TWSVM, generally, we solve the QPPs in the dual space. However, this solving method will be affected by time and memory constraints when dealing with the big datasets. In order to improve the computational speed, the TWSVM model represented by (2) and (3) would be transformed into two unconstrained non-smooth optimization problems by using the plus function.

According to the KKT theorem, we can get

$$\xi^{(2)}=\max\{0,e_2+(Bw^{(1)}+e_2b^{(1)})\} \tag{4}$$

$$\xi^{(1)}=\max\{0,e_1-(Aw^{(2)}+e_1b^{(2)})\} \tag{5}$$

The optimization problems (2) and (3) can be rewritten as

$$\min\frac{1}{2}\left\|Aw^{(1)}+e_1b^{(1)}\right\|^2+c_1e_2^T\max\{0,(e_2+Bw^{(1)}+e_2b^{(1)})\} \tag{6}$$

$$\min\frac{1}{2}\left\|Bw^{(2)}+e_2b^{(2)}\right\|^2+c_2e_1^T\max\{0,(e_1-Aw^{(2)}+e_1b^{(2)})\} \tag{7}$$

Let $(x_1)_+=\max\{0,(e_2+Bw^{(1)}+e_2b^{(1)})\}$,

$(x_2)_+=\max\{0,(e_1-Aw^{(2)}+e_1b^{(2)})\}$,

where $(x_1)_+$ and $(x_2)_+$ are the plus functions. Apparently, the objective functions of the unconstrained optimization problems (6) and (7) are convex and non-smooth.

Theorem 1 The unconstrained TWSVM model can be represented as (6) and (7) and the model is continuous but non-smooth.

Theorem 1 shows that (6) and (7) are non-smooth, so we can't use the gradient optimization method such as the Newton-Armijo method to solve (6) and (7). In order to address this problem, we will use the polynomial smooth function to approach (6) and (7).

B. The Polynomial Smooth Function

Weierstrass Theorem [17] Set arbitrary continuous function $f(x)$, $x\in[m,n]$, existing polynomial $P_n(x)$ makes $\lim\limits_{n\to\infty}\max\limits_{m\leq x\leq n}\left|f(x)-P_n(x)\right|=0$.

Weierstrass's theorem shows that any continuous real-valued function in closed interval can be arbitrarily approached by the polynomial function. From theorem 1 we can know that the plus function is a continuous function, so we can use the polynomial function to approach it. In this paper, we will give the common formula of the polynomial smooth function by transforming it to an equivalent infinite series.

Lemma 1 [18] Two expansion of $m=\dfrac{1}{2}$ can be expressed as

$$\sqrt{1+x}=1+\frac{1}{2}x-\frac{1}{2\cdot 4}x^2+\frac{1\cdot 3}{2\cdot 4\cdot 6}x^3-\frac{1\cdot 3\cdot 5}{2\cdot 4\cdot 6\cdot 8}x^4+\cdots=$$
$$1+\frac{1}{2}x-\sum_{n=2}^{\infty}\frac{(2n-3)!!}{(2n)!!}(-x)^n,\ -1\le x\le 1 \tag{8}$$

Theorem 2 The plus function x_+ can be transformed to an equivalent infinite series in $[-\frac{1}{k},\frac{1}{k}]$ as follows.

$$x_+=\frac{1}{2k}\left(\frac{1+k^2x^2}{2}-\sum_{n=2}^{\infty}\frac{(2n-3)!!}{(2n)!!}(1-k^2x^2)^n\right)+\frac{x}{2} \tag{9}$$

Proof According to the definition of x_+, we can get

$$x_+=\max(0,x)=\frac{|x|+x}{2}=\frac{|kx|}{2k}+\frac{x}{2}=\frac{1}{2k}\sqrt{1+(k^2x^2)-1}+\frac{x}{2} \tag{10}$$

According to lemma 1 and (10), x_+ can be rewritten as

$$x_+=\frac{1}{2k}\left(\frac{1+k^2x^2}{2}-\sum_{n=2}^{\infty}\frac{(2n-3)!!}{(2n)!!}(1-k^2x^2)^n\right)+\frac{x}{2} \tag{11}$$

End.

Theorem 3 The polynomial approximation function for x_+ in $[-\frac{1}{k},\frac{1}{k}]$ is

$$P_n(x,k)=\begin{cases} x, & x\ge\frac{1}{k}\\ \frac{1}{2k}\left(\frac{1+k^2x^2}{2}-\sum_{l=2}^{n}\frac{(2l-3)!!}{(2l)!!}(1-k^2x^2)^l\right)+\frac{x}{2}, & |x|<\frac{1}{k},k>0\\ 0, & x\le-\frac{1}{k},\end{cases} \tag{12}$$

where n is a positive integer. The approximation image of the plus function by the polynomial function when $k=10$, $n=1,2$ is shown as figure 1. From Figure 1, we can see that the approximation accuracy of $P_n(x,k)$ will be higher with n larger.

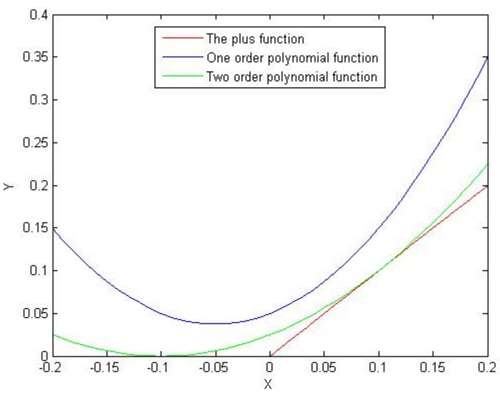


Figure 1. The approximation image of the plus function

Theorem 4 $P_n(x,k)$ is defined as (12), it has some characteristics as follows.

(1) $P_n(x,k)$ has n-order smoothness about x.

(2) $\lim_{n\to\infty}\max(P_n(x,k)-x_+)=0$.

Proof (1) If $P_n(x,k)$ has n-order smoothness about x, it must meet the following conditions.

$$P_n(\frac{1}{k},k)=\frac{1}{k},\quad P_n(-\frac{1}{k},k)=0\ ;$$
$$\nabla P_n(\frac{1}{k},k)=1,\quad \nabla P_n(-\frac{1}{k},k)=0\ ;$$
$$\nabla^n P_n(\frac{1}{k},k)=0,\quad \nabla^n P_n(-\frac{1}{k},k)=0\ ,\quad n\ge 2$$

According to (12), it can be got

$$P_n(\frac{1}{k},k)=\frac{1}{k},\quad P_n(-\frac{1}{k},k)=0$$

We find the partial derivative of x, it can have when $n\ge 1$,

$$\nabla P_n(x,k)=\begin{cases} 1, & x\ge\frac{1}{k}\\ \frac{kx}{2}\left(1+\sum_{l=2}^{n}\frac{(2l-3)!!}{(2l-2)!!}(1-k^2x^2)^{l-1}\right)+\frac{1}{2}, & |x|<\frac{1}{k},k>0\\ 0, & x\le-\frac{1}{k}\end{cases}$$

when $n\ge 2$,

$$\nabla^2 P_n(x,k)=\begin{cases} 0, & x\ge\frac{1}{k}\\ \frac{k}{2}\left(1+\sum_{l=2}^{n}\frac{(2l-3)!!}{(2l-2)!!}(1-k^2x^2)^{l-1}\right)-\\ \frac{k^3x^2}{2}\sum_{l=2}^{n}\frac{(2l-3)!!}{(2l-4)!!}(1-k^2x^2)^{l-2}, & |x|<\frac{1}{k},k>0\\ 0, & x\le-\frac{1}{k}\end{cases}$$

$\nabla P_n(x,k)$, $\nabla^2 P_n(x,k)$ and $\nabla^n P_n(x,k)$ ($n>2$) are existence and continuation in $x=\pm\frac{1}{k}$. So $P_n(x,k)$ has n-order smoothness about x.

(2) According to weierstrass's Theorem, it can be got easily

$\lim_{n\to\infty}\max(P_n(x,k)-x_+)=0$.

End.

Theorem 4 shows that the polynomial smooth function transformed to an equivalent infinite series can achieve arbitrary precision to approach the plus function when n is large enough.

C. The Optimal Smoothing Factor

There is a parameter k called smoothing factor in (12). We give the formula of optimal smoothing factor as follows.

Theorem 5 Give arbitrary precision E, if the smooth function $P_n(x,k)$ meets the condition $|P_n(x,k)-x|\le E$ when it approaches to x_+, the smoothing factor k is called the optimal smoothing factor and is denoted as $k_{opt}(n,E)$.

Because the error of $P_n(x,k)$ approaching to x_+ is maximum in $x=0$, we can get $k_{opt}(n,E)$ when it meets the condition $P_n(x,k)-x_+ \le E$ in $x=0$.

Therefore, if $x=0$, calculate (12), we can get

$$k_{opt}(n,E) \ge \frac{\frac{1}{2}-\sum_{l=2}^{n}\frac{(2l-3)!!}{(2l)!!}}{2E} \quad (13)$$

D. PSTWSVM Algorithm

Because $P_n(x,k)$ has n-order smoothness when $n \ge 2$, Newton-Armijo optimization algorithm can be used to solve the following unconstrained optimization problems.

$$\min \frac{1}{2}\left\|Aw^{(1)}+e_1b^{(1)}\right\|^2 + c_1e_2^T P((e_2+Bw^{(1)}+e_2b^{(1)}),k) \quad (14)$$

$$\min \frac{1}{2}\left\|Bw^{(2)}+e_2b^{(2)}\right\|^2 + c_2e_1^T P((e_1+Aw^{(2)}+e_1b^{(2)}),k) \quad (15)$$

Algorithm 1 PSTWSVM based on the Newton-Armijo method

Input: Give the initial value $(w^0,b^0)\in R^{n+1},\eta$, let the iteration number $i=0$, the order of polynomial function n, the arbitrary precision E;

Output: The optimal value of the objective function.

Step1: calculate $P_n(x,k)$ and $g^i=\nabla P_n(x,k)$.

Step2: If $\left\|g^i\right\| \le \eta$, select $(w^*,b^*)=(w^i,b^i)$, then terminate programs. Otherwise according to $\nabla^2 P_n(x,k)d^i=-g^i$, calculate the down direction d^i.

Step3: take $\delta\in(0,\frac{1}{2})$, $\lambda_i=\max\{1,\frac{1}{2},\frac{1}{4},\cdots\}$, let $P_n(x,k)-P_n((w^i,b^i)+\lambda_i d^i,k)\ge -\delta\lambda_i g^i d^i$, then let $(w^{i+1},b^{i+1})=(w^i,b^i)+\lambda_i d^i$.

Step4: Let $i\leftarrow i+1$, turn to Step2.

E. The Nonlinear PSTWSM

If the previous conclusions are extended to nonlinear smooth PSTWSVM, it can be used to deal with the nonlinear problem.

In order to obtain the nonlinear classifiers we consider the following kernel generated surfaces

$$K(x^T,C^T)u_1+b_1=0,\ K(x^T,C^T)u_2+b_2=0, \quad (16)$$

where $C^T=[A \quad B]^T$, $(u_{(i)},b_{(i)})\in(R^m\times R)$ $(i=1,2)$ and K is an chosen kernel. The nonlinear TWSVM are obtained by solving the following optimization problems.

$$\begin{aligned}&\min_{w^{(1)},b^{(1)},\xi^{(2)}} \frac{1}{2}\left\|K(A,C^T)w^{(1)}+e_1b^{(1)}\right\|^2+c_1e_2^T\xi^{(2)}\\ &s.t.\quad -(K(B,C^T)w^{(1)}+e_2b^{(1)})\ge e_2-\xi^{(2)},\\ &\qquad\qquad \xi^{(2)}\ge 0.\end{aligned} \quad (17)$$

$$\begin{aligned}&\min_{w^{(1)},b^{(1)},\xi^{(2)}} \frac{1}{2}\left\|K(B,C^T)w^{(2)}+e_2b^{(1)}\right\|^2+c_2e_1^T\xi^{(1)}\\ &s.t.\quad (K(A,C^T)w^{(2)}+e_1b^{(2)})\ge e_1-\xi^{(1)},\\ &\qquad\qquad \xi^{(1)}\ge 0.\end{aligned} \quad (18)$$

Introducing the plus function, (17) and (18) can be transformed into the following optimization problems without constraint.

$$\min\frac{1}{2}\left\|K(A,C^T)w^{(1)}+e_1b^{(1)}\right\|^2+c_1e_2^T(e_2+K(B,C^T)w^{(1)}+e_2b^{(1)})_+ \quad (19)$$

$$\min\frac{1}{2}\left\|K(B,C^T)w^{(2)}+e_2b^{(2)}\right\|^2+c_2e_1^T(e_1-K(A,C^T)w^{(2)}+e_1b^{(2)})_+ \quad (20)$$

We can get the nonlinear PSTSVMs-NA model using the polynomial smooth function.

$$\min\frac{1}{2}\left\|K(A,C^T)w^{(1)}+e_1b^{(1)}\right\|^2+c_1e_2^TP((e_2+K(B,C^T)w^{(1)}+e_2b^{(1)}),k) \quad (21)$$

$$\min\frac{1}{2}\left\|K(B,C^T)w^{(2)}+e_2b^{(2)}\right\|^2+c_2e_1^TP((e_1-K(A,C^T)w^{(2)}+e_1b^{(2)}),k) \quad (22)$$

The previous conclusions and theorems are also applicable to the nonlinear PSTWSVM model.

III. PSTWSVM BASED ON INVASIVE WEED OPTIMIZATION ALGORITHM

A. Analysis the Penalty Parameters of PSTWSVM

The role of penalty parameters c_1 and c_2 is to adjust the ratio between the confidence range with the experience risk in the defining feature, so that the generalization ability of PSTWSVM can achieve the best state. The values of c_1 and c_2 smaller expresses the punishment on empirical error smaller. Do it this way, the complexity of PSTWSVM is smaller, but its fault tolerant ability is worse. The values of c_1 and c_2 are greater, the data fitting degree is higher, but its generalization capacity will be reduced. From the above analysis, we can know that the parameters selection is very important for PSTWSVM.

After the above analysis, in this paper, Invasive Weed Optimization (IWO) algorithm which has fast global searching ability is used to select the PSTWSVM parameters and the mixed kernel parameters.

B. Invasive Weed Optimization

In 2006, a novel stochastic optimization model, invasive weed optimization (IWO) algorithm [16], was proposed by Mehrabian and Lucas, which is inspired from a common phenomenon in agriculture: colonization of invasive weeds. Not only it has the robustness, but also it is easy to understand and program. So far, it has been applied in many engineering fields [19-20].

In the classical IWO, weeds represent the feasible solutions of problems and population is the set of all weeds. A finite number of weeds are being dispread over the search area. Every weed produces new weeds depending on its fitness. The generated weeds are randomly distributed over the search space by normally distributed random numbers with a mean equal to zero. This process continues until maximum number of weeds is reached. Only the weeds with better fitness can survive and produce seed, others are being eliminated. The process continues until maximum iterations are reached or hopefully the weed with best fitness is closest to optimal solution.

The process is addressed in details as follows:

Step 1: *Initialize a population*

A population of initial solutions is being dispread over the D dimensional search space with random positions.

Step 2: *Reproduction*

The higher the weed's fitness is, the more seeds it produces. The formula of weeds producing seeds is

$$weed_n = \frac{f - f_{\min}}{f_{\max} - f_{\min}}(s_{\max} - s_{\min}) + s_{\min} \qquad (23)$$

where, f is the current weed's fitness. f_{max} and f_{min} respectively represent the maximum and the least fitness of the current population. s_{max} and s_{min} respectively represent the maximum and the least value of a weed.

Step 3: *Spatial dispersal*

The generated seeds are randomly distributed over the D dimensional search space by normally distributed random numbers with a mean equal to zero, but with a varying variance. This ensures that seeds will be randomly distributed so that they abide near to the parent plant. However, standard deviation (σ) of the random function will be reduced from a previously defined initial value (σ_{init}) to a final value (σ_{final}) in every generation. In simulations, a nonlinear alteration has shown satisfactory performance, given as follows

$$\sigma_{cur} = \frac{(iter_{\max} - iter)^n}{(iter_{\max})^n}(\sigma_{init} - \sigma_{final}) + \sigma_{final} \qquad (24)$$

Where, $iter_{max}$ is the maximum number of iterations, σ_{cur} is the standard deviation at the present time step and n is the nonlinear modulation index. Generally, n is set to 3.

Step 4: *Competitive exclusion*

After passing some iteration, the number of weeds in a colony will reach its maximum (P_MAX) by fast reproduction. At this time, each weed is allowed to produce seeds. The produced seeds are then allowed to spread over the search area. When all seeds have found their position in the search area, they are ranked together with their parents (as a colony of weeds). Next, weeds with lower fitness are eliminated to reach the maximum allowable population in a colony. In this way, weeds and seeds are ranked together and the ones with better fitness survive and are allowed to replicate. The population control mechanism also is applied to their offspring to the end of a given run, realizing competitive exclusion.

C. The Algorithm steps of PSTWSVM-IWO

The accuracy in the sense of CV is used for the fitness of IWO. So the algorithm steps of PSTWSVM-IWO are as follows:

Step1: Select the training dataset and the testing dataset.

Step2: Preprocessing the dataset.

Step3: Constructe the PSTWSVM model.

Step4: Select the optimal parameters using IWO algorithm.

Step5: Train the PSTWSVM model using the optimal parameters.

Step6: Predict the testing dataset.

Step7: Output the classification accuracy.

IV. EXPERIMENT RESULTS AND ANALYSIS

In order to verify the efficiency of PSTWSVM and PSTWSVM-IWO, we conduct two experiments. In the first experiment, in order to show the advantage of PSTWSVM, we make experiments on several benchmark datasets using four algorithms, that is, GEPSVM, TWSVM, STWSVM and PSTWSVM. In the second experiment, we make experiment on NDC dataset to compare PSTWSVM with PSTWSVM-IWO. The dual QPPs arising in TWSVM are solved using mosek optimization toolbox for MATLAB [21] which implements fast interior point based algorithms. Classification accuracy of each algorithm is measured by standard tenfold cross-validation methodology.

A. The First Experiment

In this experiment, we make experiments on several benchmark datasets using four algorithms, that is, GEPSVM, TWSVM, STWSVM and PSTWSVM. The optimal parameters of these algorithms are searched from $\left\{2^i \mid i = -6, -4, -2, 0, 1, 2, 4, 6\right\}$ using the grid search algorithm. In PSTWSVM, the parameter of Newton-Armijo method is set $\varepsilon_1 = 1.0E-3$, the approximation accuracy of smooth function is set $\varepsilon_2 = 1.0E-3$. For the nonlinear case, we only consider the Gaussian kernel function. The optimal value of Gaussian kernel parameter is selected over the range $\left\{2^i \mid i = -6, -4, -2, 0, 1, 2, 4, 6\right\}$. The order of polynomial is set $n = 5$.

TABLE1.
COMPARISON FOR LINEAR KERNEL

Dataset	PSTWSVM	STWSVM	TWSVM	GEPSVM
Hepatitis	78.05 ± 4.31	77.39 ± 2.15	78.08 ± 2.16	77.28 ± 2.78
Housing	86.21 ± 2.39	84.42 ± 3.87	85.42 ± 4.53	74.81 ± 2.85
Wdbc	96.10 ± 6.32	94.89 ± 4.31	96.22 ± 6.67	92.81 ± 2.54
Glass6	96.52 ± 4.56	95.70 ± 6.05	96.55 ± 2.40	96.21 ± 2.72
Votes	95.50 ± 1.23	94.96 ± 4.24	95.85 ± 2.24	95.63 ± 2.74

TABLE 2.
COMPARISON FOR GAUSSIAN KERNEL

Dataset	PSTWSVM	STWSVM	TWSVM	GEPSVM
Australian	88.23 ± 3.14	86.83 ± 3.24	85.56 ± 2.17	85.24 ± 2.09
Breast-cancer	71.16 ± 1.28	69.38 ± 1.12	69.42 ± 2.32	67.78 ± 1.56
Heart	84.54 ± 4.32	82.89 ± 4.56	82.24 ± 3.59	80.13 ± 3.42
Pima	82.12 ± 3.07	79.76 ± 3.05	78.52 ± 2.48	76.56 ± 2.16
Votes	97.26 ± 2.35	95.14 ± 3.27	95.09 ± 2.56	93.45 ± 2.12
Sonar	91.47 ± 2.24	89.67 ± 3.12	88.99 ± 4.67	85.96 ± 2.23
CMC	77.56 ± 3.25	75.58 ± 3.85	69.54 ± 2.09	66.52 ± 2.16

TABLE 3.
DESCRIPTION OF NDC DATASETS

Dataset	# Training data	# Test data	# Feature
NDC-500	500	50	32
NDC-700	700	70	32
NDC-900	900	90	32
NDC-1k	1000	100	32
NDC-2k	2000	200	32
NDC-3k	3000	300	32
NDC-4k	4000	400	32
NDC-5k	5000	500	32
NDC-10k	10,000	1000	32
NDC-1l	100,000	10,000	32
NDC-3l	300,000	30,000	32
NDC-5l	500,000	50,000	32
NDC-1m	1,000,000	100,000	32

TABLE 4.
COMPARION FOR LINEAR KERNEL

Dataset	PSTWSVM-IWO Train (%) Test (%) Time (s)	PSTWSVM Train (%) Test (%) Time (s)	TWSVM Train (%) Test (%) Time (s)
	81.07	80.05	79.93
NDC-3k	79.69	77.64	77.66
	3.2412	9.1545	27.08
	81.08	79.89	79.80
NDC-4k	74.92	73.78	73.75
	3.5605	10.0665	60.94
	79.87	78.33	79.15
NDC-5k	82.24	80.26	80.23
	4.0734	11.0761	114.24
	87.23	86.48	86.45
NDC-10k	88.56	87.32	87.38
	4.1178	15.1239	1092.07
	85.59	84.35	-
NDC-1l	87.75	86.28	-
	4.996	16.014	-
	81.43	78.75	-
NDC-3l	79.34	75.78	-
	5.899	18.103	-
	79.54	78.26	-
NDC-5l	82.01	79.17	-
	6.1312	18.3505	-

"-" We stop experiment as computing time was very high

Table 1 shows the comparison of classification accuracy for PSTWSVM with GEPSVM, TWSVM and STWSVM for linear kernel on five UCI datasets. Table 2 shows the comparison of classification performance for nonlinear extensions of PSTWSVM with GEPSVM, TWSVM and STWSVM. Table 1 and table 2 show that the accuracy performance of PSTWSVM is better than STWSVM. Therefore, we can know the approximation ability of polynomial function is better than the sigmoid function.

TABLE5.
COMPARION FOR GAUSSIAN KERNEL

Dataset	PSTWSVM-IWO Train (%) Test (%) Time (s)	PSTWSVM Train (%) Test (%) Time (s)	TWSVM Train (%) Test (%) Time (s)
	100.00	100.00	100.00
NDC-500	80.25	80.15	79.18
	0.4654	0.5632	0.789
	99.88	99.54	99.27
NDC-700	85.23	83.17	84.29
	1.3009	4.3123	1.7322
	99.76	99.56	99.56
NDC-900	82.85	81.36	80.58
	1.5017	2.5354	3.4675
	98.83	98.47	98.85
NDC-1k	85.32	84.12	83.85
	1.5029	3.5198	4.1176
	100.00	100.00	99.68
NDC-2k	88.45	88.24	88.27
	11.09	21.25	25.8958
	100.00	100.00	99.53
NDC-3k	91.67	90.12	90.45
	65.299	78.609	85.445

B. The Second Experiment

Similar to SVM and TWSVM, the learning performance and generalization ability of PSTWSVM is very dependent on its parameters selection. In the above experiment, we used the grid search algorithm to find the parameters values, which is a commonly used method. However, it will lead to low efficiency when dealing with big data. In this paper, we try to use invasive weed optimization (IWO) algorithm to optimize PSTWSVM and propose an algorithm called PSTWSVM-IWO. In this section, we will conduct experiment on NDC datasets which are generated by David Musicant' NDC Data Generator [22] to test the ability of our algorithm for dealing with big data. The parameters of IWO are as follows: $D = 5$, $P\_MAX = 30$, $s_{\max} = 5$, $s_{\min} = 1$, $n = 3$, $\sigma_{init} = [1, 0.1, 1, 1, 1]$, $\sigma_{final} = [0.1, 0.1, 0.1, 0.1, 0.1]$. In IWO algorithm, the accuracy in the sense of CV is used for the fitness of IWO. Therefore, the fitness value is closer to 100, the obtained parameters is closer to the optimal value. Table 3 gives a description of NDC datasets. Table 4 shows the comparison of computing time and accuracy for three algorithms with linear kernel. On the other hand, table 5 shows the comparison of classification performance for these algorithms with Gaussian kernel. Figure 2~3 are the fitness curves of IWO searching the optimal parameters for dealing with NDC-500, NDC-700, respectively.

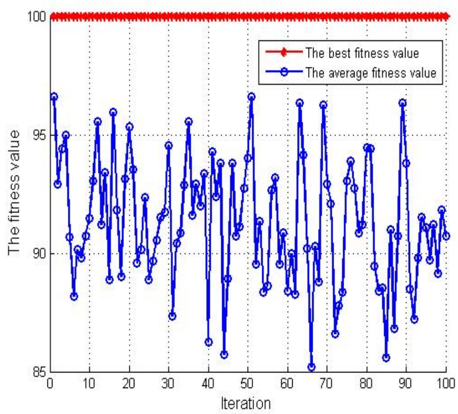


Figure 2. The fitness curves of IWO searching the optimal parameters for dealing with NDC-500

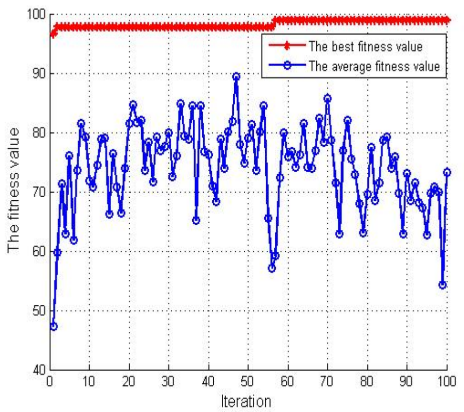


Figure 3. The fitness curves of IWO searching the optimal parameters for dealing with NDC-700

From table 4, we can see that in view of the high computing time, TWSVM can't work when the training samples reach 100000. However, PSTWSM-IWO and PSTWSVM can get reasonable accuracy in the relatively short time when the training samples reach 500000, which indicates that PSTWSM-IWO and PSTWSVM have the advantage on dealing with big data comparing with TWSVM. Furthermore, from table 4 and 5, we also see that the classification accuracy of PSTWSVM-IWO and the computing time are better than PSTWSVM. Therefore, PSTWSVM-IWO is suitable for dealing with big data. Figure 2 and figure 3 shows that the optimization ability of IWO is very strong.

V. CONCLUSION AND FURTURE WORK

In order to improve the performance of STWSVM, seeking a better smoothing function is the key problem. In this paper, a novel version for smooth TWSVM, called polynomial smooth twin support vector machines (PSTWSVM), is proposed. Firstly, using the series expansion, a new class of polynomial smoothing is proposed, and then we prove their important properties. Subsequently, the polynomial functions are adopted to convert the original constrained QPPs of TWSVM into unconstrained minimization problems, and then are solved by the well-known Newton-Armijo algorithm. The experiments show that the proposed algorithm can obtain better classification than STWSVM. In view of the good optimization ability of Invasive Weed Optimization (IWO) algorithm, it is used to optimize PSTWSVM in this paper. And then we propose an algorithm called polynomial smooth twin support vector machines based on invasive weed optimization algorithm (PSTWSVM-IWO). We enhance our algorithm to deal with big data, the results indicate that PSTWSVM-IWO is a good method to deal with large datasets.

REFERENCES

[1] L. Ding and F. Yu. "A classification algorithm for network traffic based on improved support vector machine", Journal of Computers, 2013, vol. 8, no.4, pp.1090-1096.

[2] Y.N. Zhang and S.F. Ding. "An algorithm research for prediction of extreme learning machines based on rough sets", Journal of Computers, 2013, vol. 8, no.5, pp.1335-1342.

[3] L.S. Yin and Y.G. He. "Adaptive chaotic prediction algorithm of RBF neural network filtering model based on phase space reconstruction", Journal of Computers, 2013, vol. 8, no.6, pp.1449-1455.

[4] D. Morariu and R. Cretulescu. "Improving a SVM Meta-classifier for Text Documents by using Naïve Bayes", International Journal of Computers Communications & Control, 2010, vol. 5, no.3, pp.351-361.

[5] Z.Y. Chen and Z.P. Zhi. "Distributed customer behavior prediction using multiplex data: A collaborative MK-SVM approach", Knowledge-Based Systems, 2012, vol.35, pp.111-119.

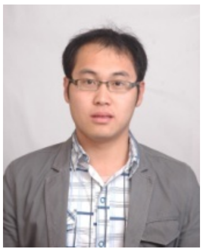
[6] F.Q. Shi and J. Xu. "Emotional cellular-based multi-class fuzzy support vector machines on product's KANSEI extraction", Applied Mathematics & Information Sciences, 2012, vol.6, no.1, pp. 41-49.

[7] C.S. Lo and C. M. Wang. “Support vector machine for breast MR image classification”, Computers & Mathematics with Applications, 2012, vol.64, no.5, pp.1153-1162.
[8] C. Cortes and V.N. Vapnik, “Support vector networks”, Machine Learning, 1995, vol.20, pp.273-297.
[9] E. Osuna and R. F. Girosi. “An improved training algorithm for support vector machines”, Proceedings of the 1997 IEEE Workshop on Neural Networks for Signal Processing. New York: IEEE Press, 1997, pp.276-285.
[10] J.C. Platt. “Using analytic QP and sparseness to speed training of support vector machines”, In M. Kearns, S. Solla and D.Cohn, Advances in Neural Information Processing Systems 11. Cambridge, MA: MIT Press,1999, pp. 557-563.
[11] O. L. Mangasarian. “Multisurface proximal support vector machine classification via generalized eigenvalues”, IEEE Transactions on Pattern Aalysis and Machine Intelligence, 2006, vol.28, no.1, pp. 69-74.
[12] Jayadeva, K. Reshma and S. Chandra, “Twin support vector machines for pattern classification”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, vol. 29, no.5, pp. 905-910.
[13] H. H. Cong, C. F. Yang and X. R. Pu. “Efficient Speaker Recognition based on Multi-class Twin Support Vector Machines and GMMs”, 2008 IEEE Conference on Robotics, Automation and Mechatronics, 2008, pp.348-352.
[14] X. S. Zhang and X. B. Gao and Y. Wang. “Twin Support Vector Machine For MCs Detection”, Journal of Electronics (China), 2009, vol.26, no.3, pp.318-325.
[15] M. Arun Kumar and M. Gopal. “Application of smoothing technique on twin support vector machines”, Pattern Recognition Letters, 2008, vol.28, pp.1842-1848.
[16] A.R. Mehrabian, C. Lucas, “A novel numerical optimization algorithm inspired from weed colonization”, Ecological Informatics, 2006, vol. 1, no.4, pp:355-366.
[17] Y. Q. Liu, S. Y. Liu and M. T. Gu. “Self-training Polynomial Support Smooth Semi-supervised Support Vetor Machines”, Journal of System Simulation, 2009, vol.21, no.18, pp.5740-5743.
[18] R. Bin and L. L. Cheng, “Polynomial smoothing support vector regression”, Control Theory & Applications, 2011, vol.28, no.2, pp.261-265.
[19] G.R. Gourab, D. Swagatam, C. Prithwish and N.S. Ponnuthurai, “Design of Non-Uniform Circular Antenna Arrays Using a Modified Invasive Weed Optimization Algorithm”, IEEE Transactions on antennas and propagation, vol.59, no.1, pp.110-118.
[20] F.M. Monavar and N. Komjani, “Bandwidth enhancement of microstrip patch antenna using jerusalem cross-shaped frequency selective surfaces by invasive weed optimization approach”, Progress In Electromagnetics Research, vol.121,pp.103-120.
[21] Mosek, http://www.mosek.com, 2007.
[22] O. L. Mangasarian and D. R, “Musicant, Lagrangian support vector machines”, J. Machine Learn. Res. 2001, vol.1, pp.161-177.

Shifei Ding received his bachelor's degree and master's degree from Qufu Normal University in 1987 and 1998 respectively. He received his Ph.D degree from Shandong University of Science and Technology in 2004. He received postdoctoral degree from Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences in 2006. And now, he works in China University of Mining and Technology as a professor and Ph.D supervisor. His research interests include intelligent information processing, pattern recognition, machine learning, data mining, and granular computing et al. He has published 3 books, and more than 150 research papers in journals and international conferences.

Huajuan Huang, born in 1984. Received her B.Sc.degree and M.Sc. degree in applied computer Technology from Guangxi University for Nationalities, Guangxi, China, in 2006 and 2009 respectively. Since 2011, she has been a Ph.D. degree candidate in applied computer Technology from the China University of Mining and Technology, Xuzhou, China. Her current research interests include data mining, parrern recognition and computational intelligence.

Junzhao Yu is currently a graduate student now studying at School of Computer Science and Technology, China University of Mining and Technology. He received his B.Sc. degree in computer science from China University of Mining and Technology in 2011. His research interests include pattern recognition, machine learning, and twin support vector machines et al.



Fulin Wu is currently a graduate student now studying in School of Computer Science and Technology, China University of Mining and Technology, and his supervisor is Prof. Shifei Ding. He received his B.Sc. degree in computer science from China University of Mining and Technology in 2012. His research interests include cloud computing, feature selection, pattern recognition, machine learning et al.