Applied Mathematics & Information Sciences
An International Journal

An Improved QPSO Algorithm for Parameters Optimization of LS-SVM

Yi Liu

Information College, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China

Received: 20 Aug. 2012, Revised: 5 Sep. 2012, Accepted: 1 Nov. 2012
Published online: 1 Feb. 2013

Abstract: Aiming at the parameter optimization of least square support vector machine (LS-SVM), an improved quantum-behaved particle swarm optimization (IQPSO) algorithm for LS-SVM parameter selection was proposed. Based on QPSO, the algorithm optimizes particle initializing positions and improves solving speed and precision by sampling and linearizing methods. IQPSO LS-SVM model was test by test functions and was compared with QPSO LS-SVM model. Furthermore, it was applied to thread's amount setting of database server in an agricultural producing system. The results show that the proposed model has greater solving speed and higher precision. It can meet the database's load requirement by thread's amount adjustment in agricultural producing system.

Keywords: LS-SVM, parameter optimization, Improved-QPSO, multi-thread

1 Introduction

In [1], support vector machine (SVM) is a learning method based on the structural risk minimization principle. Least squares support vector machine (LS-SVM) which was proposed in [2] converts the solving process of SVM from quadratic programming to linear equations, having specialties of fast learning and easy to use, it is widely applied to system identification, recognition and prediction in many fields. As the learning precision and generalization of LS-SVM model is determined by its parameters, algorithms for LS-SVM parameter choosing were put forward, such as genetic algorithm, particle swarm optimization (PSO) algorithm, quantum-behaved particle swarm optimization (QPSO) algorithm, see [3–6].QPSO is widely used as it can search for the global optimal solution with great convergence speed. In the paper, an improved QPSO algorithm (IQPSO) was proposed to optimize the particle initial positions in QPSO algorithm.

Test results indicate that IQPSO algorithm has better convergence rate and prediction than QPSO in applications of LS-SVM parameter optimization, giving greater generalization ability to LS-SVM models.

2 LS-SVM algorithm based on QPSO

$x_i : x_1, x_2, \cdots, x_l \in R^n$ and $y_j : y_1, y_2, \cdots, y_l \in R$ are assumed as input samples and output samples. Input samples are mapped to feature space R^m by nonlinear mapping $\Psi(x) = [\varphi(x_1), \varphi(x_2), \cdots, \varphi(x_l)]$, which converts estimation from nonlinear original space to linear feature space ($f(x) = w\varphi(x_i) + b$, w is weight vector of feature space). The regression of LS-SVM is:

$$\min_{w,\xi} J(w,\xi) = \frac{1}{2}w^T w + \frac{1}{2}\gamma\sum_{i=1}^{l}\xi_i^2 \quad (1)$$

where γ is normalization parameter, ξ is error, constraint condition is $y_i = w\varphi(x_i) + b + \xi_i$. The corresponding Lagrange function is:

$$L(w,\xi,a,b) = -\sum_{i=1}^{l} a_i[w^T\varphi(x_i) + b + \xi_i - y_i] + J(w,\xi) \quad (2)$$

Defines $K(x_i, y_j) = \varphi^T(x_i)\varphi(x_j)$. Linear matrix equation (3) is obtained.

* Corresponding author e-mail: baddielao@126.com

$$\begin{bmatrix} 0 & 1 & \cdots & 1 \\ 1 & K(x_1,x_1)+\frac{1}{\gamma} & \cdots & K(x_1,x_l) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K(x_l,x_1) & \cdots & K(x_l,x_l)+\frac{1}{\gamma} \end{bmatrix} \begin{bmatrix} b \\ a_1 \\ \vdots \\ a_l \end{bmatrix} = \begin{bmatrix} 0 \\ y_1 \\ \vdots \\ y_l \end{bmatrix} \quad (3)$$

The regressive result of LS-SVM algorithm is:

$$\hat{f}(x) = \sum_{i=1}^{l} a_i K(x_i, x) + b \quad (4)$$

Radial basis function(RBF) is the most widely used in practical applications, that is:

$$K(x_i, x_j) = \exp[-(x_i - x_j)^2/(2\sigma^2)] \quad (5)$$

γ and σ, the most important parameters of LS-SVM model, are the main factors for training precision and generalization ability.

QPSO algorithm bases on δ potential well, it can find out the global optimal solution in its solution space, see [7]. The particle renewing method is:

$$m_{best} = \frac{1}{M}\sum_{i=1}^{M} P_i \quad (6)$$

$$P_{C_{ij}} = \varphi P_{ij} + (1-\varphi)P_{gj} \quad (7)$$

$$x_{ij} = P_{C_{ij}} \pm \alpha |m_{best_j} - x_{ij}| \ln(\frac{1}{\mu}) \quad (8)$$

where m_{best} is the center of all particles' best positions; M is particle's amount; P_{ij} stands for the No. j dimension's optimal position of No. i particle; P_{gj} is the No. j dimension's optimal position in all particles; $P_{C_{ij}}$ is a random position; φ and μ are random number between 0 and 1. α is contraction-expansion coefficient; x_{ij} stands for the No. j dimension's position of No. i particle. QPSO optimizing processes are:

Step 1: initialize particle position x_{ij} randomly in possible range;

Step 2: calculate objective function value of every particle, renew local optimal position P_{ij} of every particle;

Step 3: renew global optimal position P_{gj} and x_{ij};

Step 4: repeat step 2 3 until accuracy requirement of objective function or iteration limit is met;

Step 5: output the QPSO optimization result P_{gj}.

3 Framework of IQPSO

In the first step of QPSO algorithm, particle initial positions x_{ij} are generated randomly, a relatively large amount of iterations is needed to meet the precision requirement of objective function. If there were one or more particle(s) closed to global optimal position P_{gj} after initialization, iterations will be decreased. Thus, the paper proposed an improved QPSO algorithm (IQPSO) to optimize x_{ij} by sampling and linearizing, and search for the ones which are close to P_{gj}.

The solution space of QPSO algorithm applied to LS-SVM is a two-dimensional γ-σ space, $\gamma \in [\gamma_L, \gamma_H]$, $\sigma \in [\sigma_L, \sigma_H]$. The space will be separated into $K*L$ subregions by IQPSO algorithm firstly, just like digitalizing an analog image into $K*L$ pixels. The central point of each subregion is sampling point, as shown in Fig. 1.

X_{ij} stands for the coordinate of each sampling point:

$$X_{ij} = [\gamma_L + \frac{\gamma_H - \gamma_L}{2K}(2i-1), \sigma_L + \frac{\sigma_H - \sigma_L}{2K}(2j-1)] \quad (9)$$

where $i = 1, 2, \cdots, K, j = 1, 2, \cdots, L$. F is defined as objective function. The more small $F(X_{ij})$ is, the more probability that P_{gj} is in X_{ij}'s subregion. All of X_{ij} are the candidate positions of x_{ij}. M is defined as the initial amount of particle swarm. The selection method of x_{ij} is:

$$x_{ij} = [X_{ij} | \min F(X_{ij}), i = 1, \cdots, K, j = 1, \cdots, L] \quad (10)$$

Select an initial particle by (10), delete its corresponding position in X_{ij}. Repeat M times until all particle positions are determined.

4 Discussion of some problems in IQPSO

Details between sampling points will be omitted while the global optimal position is estimated only by sampling point. Fig. 2 shows the situation of a one-dimensional discrete objective function $F(x)$. Sampling points $A-E$ are the candidate positions of particle initialization. M_1 and M_2 are the practical local optimal positions. According to (10), point B is the first choice as it has the minimum objective function, thus two problems appears:

Firstly, $F(B) < F(E)$, but actually E's adjacent local optimal position M_2 is smaller than B's adjacent local optimal position M_1;

Secondly, $F(B) < F(C)$, but actually C is closer to its adjacent local optimal position M_1 than B.

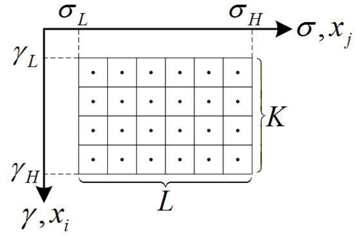


Fig. 1 Discretization of γ-σ space.

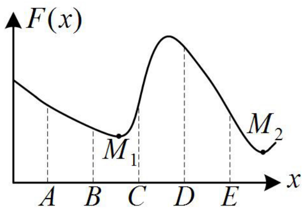


Fig. 2 Sampling of one-dimensional objective function.

Z(-1,-1)	Z(-1,0)	Z(-1,1)
Z(0,-1)	Z(0,0)	Z(0,1)
Z(1,-1)	Z(1,0)	Z(1,1)

Fig. 3 A 3*3 template.

The problems also exist in the two-dimensional objective function $F(X_{ij})$. Besides increasing sampling density, estimation by linearization is the simplest and quickest solving method. The specific methods are:

Method 1: linearize directly through X_{ij}, calculate approximate local optimal position (e.g. , the intersection point between line AB and line CD in Fig. 2 is the approximate position of M_1), and set the position as one of the candidate positions of particle initialization.

Method 2: regard Fig. 1 as a two-dimensional image, X_{ij} is pixel's position, $F(X_{ij})$ is the gray value of X_{ij} . Calculate the gradient of X_{ij} by gradient operator, and add it to $F(X_{ij})$ with certain weight, the result is the minimum objective function in the subregion of X_{ij}. X_{ij} remains as the candidate position of particle initialization.

Method 1 is relatively hard to calculate in two-dimension with multi-plane intersecting problem, on the contrary gradient operator is easy to perform in method 2; moreover, as the precision of global optimal position will be increased during iteration process, and linearization is an estimation method, it's enough to find out the subregion where the global optimal position is the most possibly inside, calculation of approximate local optimal position is unnecessary.

5 Gradient weighting of IQPSO

In [8], a 3*3 template is usually used for image filtering in digital image processing, as shown in Fig. 3.

Defines X_{ij} as pixel's position, $F(X_{ij})$ as the gray value of X_{ij} , the filtering equation of X_{ij} is:

-1	-2	-1
0	0	0
1	2	1

γ - axis template

-1	0	1
-2	0	2
-1	0	1

σ - axis template

Fig. 4 Sobel template.

$$g(X_{ij}) = \sum_{a=-1}^{1}\sum_{b=-1}^{1} Z(a,b)F(X_{i+a,j+b}) \qquad (11)$$

The gradient vector of function $F(X_{ij})$ is defined as:

$$\bigtriangledown F(X_{ij}) = \begin{bmatrix} G_{X_i} \\ G_{X_j} \end{bmatrix} \qquad (12)$$

Gradient vector of discrete function $F(X_{ij})$ could be figured out by gradient template. Prewitt and Sobel templates are typical in it. Sobel template is used in the paper, as shown in Fig. 4.

According to (11), the gradient of $F(X_{ij})$ is:

$$\begin{aligned} G_{X_i} &= F(X_{i+1,j-1}) + 2F(X_{i+1,j}) + F(X_{i+1,j+1}) \\ &\quad - F(X_{i-1,j-1}) - 2F(X_{i-1,j}) - F(X_{i-1,j+1}) \\ G_{X_j} &= F(X_{i-1,j+1}) + 2F(X_{i,j+1}) + F(X_{i+1,j+1}) \\ &\quad - F(X_{i-1,j-1}) - 2F(X_{i,j-1}) - F(X_{i+1,j-1}) \end{aligned} \qquad (13)$$

G_{X_i} and G_{X_j} in (13) are weighted mean differences, they represent the maximum change value of $F(X_{ij})$ when X_{ij} moves 8 times of unit distance in the direction of X_i (or X_j). Unit distance is the distance between two adjacent sampling points (pixels). On the other hand, IQPSO algorithm searches the minimum objective functions in each rectangle subregion which center is X_{ij}, so the maximum distance when X_{ij} moves in the direction of X_i (or X_j) is half of unit distance. The maximum change value of $F(X_{ij})$ when X_{ij} moves in its subregion is:

$$\max \triangle F(X_{ij}) = \frac{|G_{X_i}| + |G_{X_j}|}{16} \qquad (14)$$

As its based on linearization, when the maximum increase value of $F(X_{ij})$ is found, the maximum decrease value with the same size will be found in the opposite direction. The minimum value of objective function in X_{ij}'s subregion is defined as:

$$\bar{F}(X_{ij}) = F(X_{ij}) - \max \triangle F(X_{ij}) \qquad (15)$$

Finally, (10) should be modified into (16) to perform gradient weighting.

Table 1 Parameter optimization results of LS-SVM.

test function	σ_0	optimal algorithm	iteration times	elapsed time (ms)
f_1	0.05	QPSO	114	3.842
		IQPSO	71	2.576
	0.1	QPSO	148	5.415
		IQPSO	98	3.691
f_2	0.05	QPSO	121	4.058
		IQPSO	84	2.983
	0.1	QPSO	163	5.984
		IQPSO	107	4.119

$$x_{ij} = [X_{ij} | \min \bar{F}(X_{ij}), i = 1, \cdots, K, j = 1, \cdots, L] \quad (16)$$

IQPSO LS-SVM algorithm is based on sampling. When it can't meet the sampling theorem, the high-frequency components of objective function will be disappeared, and the initial particles will concentrate on the low-frequency components of objective function, so the algorithm performances will be decreased. Thus, in particle initialization, half of M particles are selected by (16), and the rest of particles are generated by random method.

6 Simulation test

The test functions for IQPSO LS-SVM algorithm are set as the following two-dimension functions:

$$\begin{aligned} f_1 &= \sin x_1 \cos x_2 + \sigma_0, x_1, x_2 \in [-\pi, \pi] \\ f_2 &= (2x_1^2 + 2x_2^2 - 1)e^{-(x_1^2+x_2^2)} + \sigma_0, x_1, x_2 \in [-2, 2] \end{aligned} \quad (17)$$

where σ_0 is a Gaussian noise which mean value is 0 and deviation is 0.05, 0.1. 200 groups of data were randomly selected, 100 for training and 100 for testing. IQPSO and QPSO algorithms were all applied to parameter optimization of LS-SVM model. MSE was used as objective function; precision requirement was less than 10^{-3}. Particle amount M was set as 30, initialization ranges of LS-SVM parameters were: γ=[0,1000] and σ=[0,10]. Parameter K and L in IQPSO algorithm were both set as 30, that is, 900 sampling points were set.

IQPSO algorithm will spend more time than QPSO algorithm while initializing particles. But as shown in Table 1, compared with QPSO algorithm, IQPSO algorithm greatly reduces iteration times and shortens elapsed time to meet precision requirement. Based on IQPSO algorithm, LS-SVM model has higher precision and tracking effect.

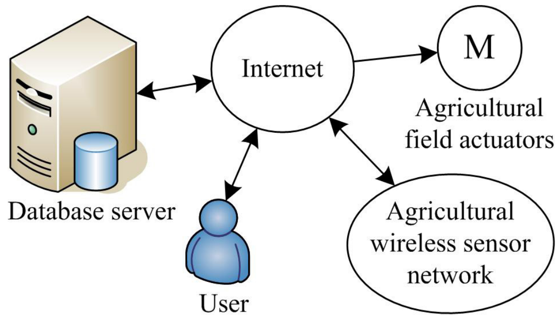


Fig. 5 Structure of agricultural producing system.

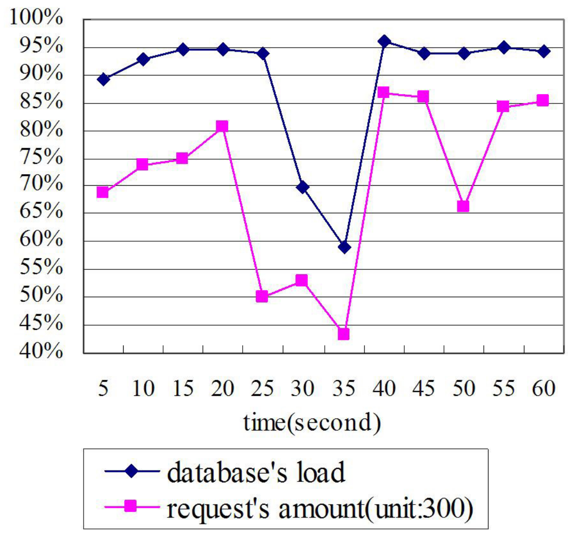


Fig. 6 Short-term load rate of database.

7 Application on agricultural producing system

As shown in Fig. 5, data of agricultural producing field are collected by wireless sensor network, transmitted by router and internet into database server. To realize agricultural automation, the agricultural field actuators are controlled based on the data in database. User can inquire data from server. In order to handle various kinds of concurrent requests, multithreading is necessary to the database server. Thread's amount will affect the performance of agricultural producing system straightly, it shall be adjusted dynamically. When thread's amount is too large, database will be overloaded; on the contrary, database's operation will be overtime, control of agricultural field actuators will be influenced. By LS-SVM model, the mathematic relation between thread's amount and database's load was built up to maximize thread's amount without database's overload.

Database's load rate was set as the output of LS-SVM model, thread's amount and other measurable properties were set as the input vector. In training stage, thread's amount changed every 5 seconds, it was random and with a range from 0 to 200. 100 groups of data were randomly selected for training. Iteration times of IQPSO algorithm was set as 100. The other settings were the same with simulation test. After optimized by IQPSO, the parameters of LS-SVM were γ=297.24, σ=1.657.

In normal stage, thread's amount was figured out and renewed every 5 seconds through the LS-SVM model obtained in training stage. Database's load rate limit was set as 95%. Figure 6 shows a one-minute short-term load rate of database, and corresponding concurrent request's amount (unit: 300) for contrast. When request's amount was about 75% of unit value, that is about 225 concurrent requests, database's load rate met the limit. With the regulating action of thread's amount, the maximum database's load rate was 96.1%, relative error was 1.16%. The model met the error request. When request's amount decreased, database's load rate still kept around the limit for a while to handle the earlier untreated requests.

8 Conclusion

The proposed IQPSO algorithm estimates particle initial positions of QPSO algorithm by sampling and linearizing, it not only has the advantages of QPSO algorithm, but also reduces iterations and shortens searching time. Application indicates that thread's amount was estimated effectively in database server system with IQPSO LS-SVM algorithm, and database could work in optimum load status.

Acknowledgement

The author is financially supported by the Science and Technology Planning Project from Science and Technology Department of Guangdong Province, project No. 2012A080104018.

References

[1] V. N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, New York (1998).

[2] J. A. K. Suykens and J. Vandewalle, Least squares support vector machine classifiers, Neural Processing Letters, vol. **9**, no. 3, pp. 293-300 (1999).

[3] S. Osowski, R. Siroic, T. Markiewicz, Application of support vector machine and genetic algorithm for improved blood cell recognition, Instrumentation and Measurement, vol. **58**, no. 7, pp. 2159-2168 (2009).

[4] B. T. Thanh and M. Parnichkun, Balancing Control of Bicyrobo by Particle Swarm Optimization-Based Structure-Specified Mixed H2/H∞ Control, International Journal of Advanced Robotic Systems, vol. **5**, no. 4, pp. 395-402 (2008).

[5] L. C. Zhou, H. Z. Yang, C. B. Liu, QPSO-based hyper-parameters selection for LS-SVM regression, Proceedings of Fourth International Conference on Natural Computation, IEEE, Jinan, pp. 130-133 (2008).

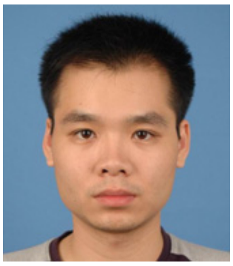
[6] B. Chen, X. Ye, L. An and Y. Wang, A Novel Quantum-Behaved Particle Swarm Algorithm and Its Application, Information Technology Journal, vol. **11**, pp. 536-539 (2012)

[7] J. Sun, B. Feng, W. B. Xu, Particle swarm optimization with particles having quantum behavior, Proceedings of Congress on Evolutionary Computation, IEEE, Portland, pp. 325-331 (2004).

[8] R. C. Gonzalez, R. E. Woods, Digital image processing (second edition), In: Q. Q. Ruan, Y. Z. Ruan, editors, Publishing House of Electronics Industry, pp. 467-469 (2004).

[9] H. Q. Zhu, C. H. Yang, J. Wang, W. H. Gui, Hybrid-QPSO-based parameters optimization of LS-SVM and its application, Journal of Central South University: Science and Technology, vol. **42**, no. 4, pp. 1000-1004 (2011).

[10] Y. Liu, Simulation research on path planning of mobile robot, Computer Simulation, vol. **28**, no. 6, pp. 227-230 (2011).



Yi Liu is presently employed as a Lecturer at Zhongkai University of Agriculture and Engineering, China. He obtained his master degree from South China University of Technology(China). His main research fields now are intelligent and predictive control systems.