Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 35 (2008) 1817–1824
Expert Systems with Applications
www.elsevier.com/locate/eswa

Particle swarm optimization for parameter determination and feature selection of support vector machines

Shih-Wei Lin [a,*], Kuo-Ching Ying [b], Shih-Chieh Chen [c], Zne-Jung Lee [a]

[a] *Information Management, Huafan University, Taiwan*
[b] *Industrial Engineering and Management Information, Huafan University, Taiwan*
[c] *Industrial Management, National Taiwan University of Science and Technology, Taiwan*

Abstract

Support vector machine (SVM) is a popular pattern classification method with many diverse applications. Kernel parameter setting in the SVM training procedure, along with the feature selection, significantly influences the classification accuracy. This study simultaneously determines the parameter values while discovering a subset of features, without reducing SVM classification accuracy. A particle swarm optimization (PSO) based approach for parameter determination and feature selection of the SVM, termed PSO + SVM, is developed.

Several public datasets are employed to calculate the classification accuracy rate in order to evaluate the developed PSO + SVM approach. The developed approach was compared with grid search, which is a conventional method of searching parameter values, and other approaches. Experimental results demonstrate that the classification accuracy rates of the developed approach surpass those of grid search and many other approaches, and that the developed PSO + SVM approach has a similar result to GA + SVM. Therefore, the PSO + SVM approach is valuable for parameter determination and feature selection in an SVM.

Keywords: Particle swarm optimization; Support vector machine; Parameter determination; Feature selection

1. Introduction

Classification problems have been extensively studied. Numerous factors, such as incomplete data, and the choice of values for the parameters of a given model, may affect classification results. Classification problems have previously been solved with statistical methods such as logistic regression or discriminate analysis. Technological advances have led to the development of methods for solving classification problems, including decision trees, back-propagation neural networks, rough set theory and support vector machines (SVM). SVM which is an emerging data classification technique proposed by Vapnik (1995), and has been widely adopted in various fields of classification problems in recent years (Cao & Tay, 2003; Huang, Lai, Luo, & Yan, 2005; Liang, 2004; Ng & Gong, 2002; Shin, Lee, & Kim, 2005; Valentini, 2002).

In the SVM, the model for classification is generated from the training process with the training data. Later on, classification is executed based on the trained model. The largest problems encountered in setting up the SVM model are how to select the kernel function and its parameter values. Inappropriate parameter settings lead to poor classification results (Keerthi & Lin, 2003).

Classification problems generally involve a number of features. However, not all of these features are equally important for a specific task. Some of them may be redundant or even irrelevant. Better performance may be achieved by discarding some features. In other circumstances, the dimensionality of input space may be decreased to save some computation effort, although this may slightly lower classification accuracy. Therefore, the classification process must be fast and accurate, using the

* Corresponding author. Tel.: +886 2 26632102x4373; fax: +886 2 26632102.
E-mail address: swlin@cc.hfu.edu.tw (S.-W. Lin).

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.08.088

smallest number of features. This objective can be achieved using feature selection. Feature selection strategies are often implied to explore the effect of irrelevant attributes on the performance of classifier systems (Acir, Özdamar, & Guzelis, 2006; Valentini, Muselli, & Ruffino, 2004; Zhang, Guo, Du, & Li, 2005).

If the SVM is adopted without considering feature selection, then the dimension of the input space is large and non-clean, degrading the performance of the SVM. Likewise an efficient and robust feature selection method that eliminates noisy, irrelevant and redundant data, while maintaining the discriminating power of the data, is critical In such a system, features extracted from the original data are adopted as inputs to the classifiers in the SVM.

This study attempts to increase the classification accuracy rate by employing an approach based on particle swarm optimization (PSO) in SVM. This novel approach is termed PSO + SVM. The developed PSO + SVM approach not only tunes the parameter values of SVM, but also identifies a subset of features for specific problems, maximizing the classification accuracy rate of SVM. This makes the optimal separating hyper-plane obtainable in both linear and non-linear classification problems.

The remainder of this paper is organized as follows. Section 2 reviews pertinent literature on SVM and the feature selection. Section 3 then describes in detail the developed PSO + SVM approach for determining the parameter values for SVM with and without feature selection. Next, Section 4 compares the experimental results with those of existing approaches. Conclusions are finally drawn in Section 5, along with recommendations for future research.

2. Literature review

SVM technique is briefly described as follows (Burgers, 1998; Huang, Chen, & Wang, 2006; SchÖlkopf & Smola, 2002). Let (x_i, y_i), $1 \leqslant i \leqslant N$, denote a set of training data, where N represents the number of training data. Each datum must conform to the criteria $x_i \in R^d$ and $y_i \in \{-1,1\}$, where d denotes the number of dimensions of input data.

SVM attempts to identify a hyper-plane, which functions as a separating plane for classification of data, in a multi-dimensional space. The parameters w and b are given by

$$(\langle w \cdot x_i \rangle + b) = 0, \ i = 1, \ldots, N. \tag{1}$$

If a hyper-plane exists that satisfies Eq. (1), then linear separation is obtained. In this case, w and b can be rewritten as follows. Eq. (1) becomes

$$\min_{1 \leqslant i \leqslant N} y_i(\langle w \cdot x_i \rangle + b) \geqslant 1, \ i = 1, \ldots, N. \tag{2}$$

Let the distance from the data point to the hyper-plane be $1/\|w\|$. Among separating hyper-planes, there exists one optimal separating hyper-plane (OSH), and the distance between two support vector points on two sides of this hyper-plane is maximal. Because the distance between two support vector points is $1/\|w\|^2$, the minimal distance to OSH, $\|w\|^2$, may be derived from Eq. (2).

The margin of a separating hyper-plane, calculated as $2/\|w\|$, determines the hyper-plane's generalization ability. The OSH has the largest margin among separating hyper-planes. $\|w\|^2$ is minimized with Eq. (2) and Lagrange's polynomial. Let a denote $(a_1, \ldots, a_N)$. Combining Lagrange's polynomial (in the order of N) with Eq. (2) produces the following equations for maximization.

$$W(a) = \sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i,j=1}^{N} a_i a_j y_i y_j x_i x_j \tag{3}$$

where $a_i \geqslant 0$ and under constraint $\sum_{I=1}^{N} y_i a_i = 0$.

Quadratic programming method can be adopted to solve the above maximization problem. If a vector $a^0 = (a_1^0, \ldots, a_N^0)$ satisfies the Eq. (3) in maximization, then the OSH expressed in terms of (w_0,b_0) may be expressed as follows:

$$w_0 = \sum_{I=1}^{N} a_i^0 y_i x_i. \tag{4}$$

where the support vector points must comply with $a_i^0 \geqslant 0$ and Eq. (2). When considering expansion in constraint Eq. (4), the determinant function of hyper-plane is expressed as follows:

$$f(x) = \operatorname{sign}\left(\sum_{i=1}^{N} a_i^0 y_i x_i x + b_0 \right) = 0. \tag{5}$$

In most cases, the data are not linearly separable, and are consequently mapped to a higher-dimensional feature space. Therefore, if the data cannot be classified clearly in the current dimensional space, then the SVM will map them to a higher dimensional space for classification.

Input data are mapped to a higher dimensional feature space by plotting a nonlinear curve. The OSH is constructed in the feature space. By constructing the feature space, $\phi(x)$ can be adopted in constrained Eq. (3) as shown below:

$$W(a) = \sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i,j=1}^{N} a_i a_j y_i y_j \phi(x_i)\phi(x_j). \tag{6}$$

Given a symmetric and positive kernel function $K(x,y)$, the existence of Mercer's theorem can be deduced. Therefore, $K(x,y) = \phi(x)\phi(y)$. Provided that the kernel function K satisfies Mercer's theorem, the derived training algorithm is guaranteed for minimization

$$W(a) = \sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i,j=1}^{N} a_i a_j y_i y_j \phi(x_i)\phi(x_j). \tag{7}$$

The decision function is expressed as follows:

$$f(x) = \operatorname{sign}\left(\sum_{i=1}^{N} a_i y_i K(x_i \cdot x_j) + b \right). \tag{8}$$

Several kernel functions help the SVM in obtaining the optimal solution. The most frequently used such kernel functions are the polynomial, sigmoid and radial basis kernel function (RBF) (Liao, Fang, & Nuttle, 2004; Lin & Lin, 2003; Müller, Mike, Rätsch, Tsuda, & Schölkopf, 2001). The RBF is generally applied most frequently, because it can classify multi-dimensional data, unlike a linear kernel function. Additionally, the RBF has fewer parameters to set than a polynomial kernel. RBF and other kernel functions have similar overall performance. Consequently, RBF is an effective option for kernel function. Therefore, this study applies an RBF kernel function in the SVM to obtain optimal solution.

Two major RBF parameters applied in SVM, C and Υ, must be set appropriately. Parameter C represents the cost of the penalty. The choice of value for C influences on the classification outcome. If C is too large, then the classification accuracy rate is very high in the training phase, but very low in the testing phase. If C is too small, then the classification accuracy rate unsatisfactory, making the model useless. Parameter Υ has a much greater influence on classification outcomes than C, because its value affects the partitioning outcome in the feature space. An excessively large value for parameter Υ results in over-fitting, while a disproportionately small value leads to under-fitting (Pardo & Sberveglieri, 2005).

Grid search (Hsu, Chang, & Lin, 2003; Wang, Wu, & Zhang, 2005) is the most common method to determine appropriate values for C and Υ. Values for parameters C and Υ that lead to the highest classification accuracy rate in this interval can be found by setting appropriate values for the upper and lower bounds (the search interval) and the jumping interval in the search. Nevertheless, this approach is a local search method, and vulnerable to local optima. Additionally, setting the search interval is a problem. Too large a search interval wastes computational resource, while too small a search interval might render a satisfactory outcome impossible.

In addition to the commonly used, grid search, other techniques are employed in SVM to improve the possibility of a correct choice of parameter values. The F-score adopts statistical type I and II errors, and random forest (RF) (Wei & Lin, 2005). Pai and Hong (2005) proposed an SA-based approach to obtain parameter values for SVM, and applied it in real data; however, this approach does not address feature selection, and therefore may exclude the optimal result.

As well as the two parameters C and Υ, other factors, such as the quality of the feature's dataset, may influence the classification accuracy rate. For instance, the correlations between features influence the classification result. Accidental removal of important features might lower the classification accuracy rate. Additionally, some dataset features may have no influence at all, or may contain a high level of noise. Removing such features can improve the searching speed and accuracy rate.

Approaches for feature selection can be categorized into two models, namely a filter model and a wrapper model (Liu & Motoda, 1998). Statistical techniques, such as principal component analysis, factor analysis, independent component analysis and discriminate analysis can be adopted in filter-based feature selection approaches to investigate other indirect performance measures, most of which are based on distance and information. Chen and Hsieh (2006) presented latent semantic analysis and web page feature selection, which are combined with the SVM technique to extract features. Gold, Holub, and Sollich (2005) presented a Bayesian viewpoint of SVM classifiers to tune hyper-parameter values in order to determine useful criteria for pruning irrelevant features. Even though the filter model is fast, the resulting feature subset may not be optimal (Liu & Motoda, 1998).

The wrapper model (Kohavi & John, 1997) applies the classifier accuracy rate as the performance measure. Some researchers have concluded that if the purpose of the model is to minimize the classifier error rate, and the measurement cost for all the features is equal, then the classifier's predictive accuracy is the most important factor. Restated, the classifier should be constructed to achieve the highest classification accuracy. The features adopted by the classifier are then chosen as the optimal features. In the wrapper model, meta-heuristic approaches are commonly employed to help in looking for the best feature subset. Although meta-heuristic approaches are slow, they obtain the (near) best feature subset.

Jack and Nandi (2002) and Shon, Kim, Lee, and Moon (2005), employed GA to screen the features of a dataset. The selected subset of features is then fed into the SVM for classification testing. Zhang, Jack, and Nandi (2005) developed a GA-based approach to discover a beneficial subset of features for SVM in machine condition monitoring. Samanta, Al-Balushi, and Al-Araimi (2003) proposed a GA approach to modify the RBF width parameter of SVM with feature selection. Nevertheless, since these approaches only consider the RBF width parameter for the SVM, they may miss the optimal parameter setting. Huang and Wang (2006) presented a GA-based feature selection and parameters optimization for SVM. Moreover, Huang et al. (2006) utilized the GA-based feature selection and parameter optimization for credit scoring.

3. The developed PSO + SVM approach

3.1. Particle swarm optimization

Particle swarm optimization (PSO) (Kennedy & Eberhart, 1995) is an emerging population-based meta-heuristic that simulates social behavior such as birds flocking to a promising position to achieve precise objectives in a multi-dimensional space. Like evolutionary algorithms, PSO performs searches using a population (called swarm) of individuals (called particles) that are updated from iteration to iteration. To discover the optimal solution, each particle changes its searching direction according to two factors, its own best previous experience (pbest) and the

best experience of all other members (gbest). Shi and Eberhart (1998) called pbest the cognition part, and gbest the social part.

Each particle represents a candidate position (i.e., solution). A particle is considered as a point in a D-dimension space, and its status is characterized according to its position and velocity. The D-dimensional position for the particle i at iteration t can be represented as $x_i^t = \{x_{i1}^t, x_{i2}^t, \ldots x_{iD}^t\}$. Likewise, the velocity (i.e., distance change), which is also an D-dimension vector, for particle i at iteration t can be described as $v_i^t = \{v_{i1}^t, v_{i2}^t, \ldots, v_{iD}^t\}$. Fig. 1 illustrates the above concept of modulation of searching points.

Let $p_i^t = \{p_{i1}^t, p_{i2}^t, \ldots, p_{iD}^t\}$ represent the best solution that particle i has obtained until iteration t, and $p_g^t = \{p_{g1}^t, p_{g2}^t, \ldots, p_{gD}^t\}$ denote the best solution obtained from p_i^t in the population at iteration t. To search for the optimal solution, each particle changes its velocity according to the cognition and social parts as follows:

$$V_{id}^t = V_{id}^{t-1} + c_1 r_1 (P_{id}^t - x_{id}^t) + c_2 r_2 (P_{gd}^t - x_{id}^t), \quad d = 1, 2, \ldots, D \tag{9}$$

where c_1 indicates the cognition learning factor; c_2 indicates the social learning factor, and r_1 and r_2 are random numbers uniformly distributed in U(0,1). Each particle then moves to a new potential solution based on the following equation:

$$X_{id}^{t+1} = X_{id}^t + V_{id}^t, \ d = 1, 2, \ldots, D \tag{10}$$

The basic process of the PSO algorithm is given as follows.

Step 1: (Initialization) Randomly generate initial particles.
Step 2: (Fitness) Measure the fitness of each particle in the population.
Step 3: (Update) Compute the velocity of each particle with Eq. (9).
Step 4: (Construction) For each particle, move to the next position according to Eq. (10).
Step 5: (Termination) Stop the algorithm if termination criterion is satisfied; return to Step 2 otherwise.

The iteration is terminated if the number of iteration reaches the pre-determined maximum number of iteration.

3.2. Apply PSO to SVM

This study developed a PSO approach, termed PSO + SVM, for parameter determination and feature selection in the SVM. Without feature selection, two decision variables, designated C and Υ, are required. For the feature selection, if n features are required to decide which features are chosen, then $2 + n$ decision variables must be adopted. The value of n variables ranges between 0 and 1. If the value of a variable is less than or equal to 0.5, then its corresponding feature is not chosen. Conversely, if the value of a variable is greater than 0.5, then its corresponding feature is chosen. Fig. 2 illustrates the solution representation.

Fig. 3 shows the flowchart for PSO + SVM. First, the population of particles is initialized, each particle having a random position within the D-dimensional space and a random velocity for each dimension. Second, each particle's fitness for the SVM is evaluated. The each particle's fitness in this study is the classification accuracy. If the fitness is better than the particle's best fitness, then the posi-

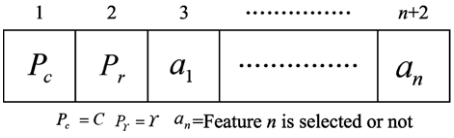

Fig. 2. Solution representation.

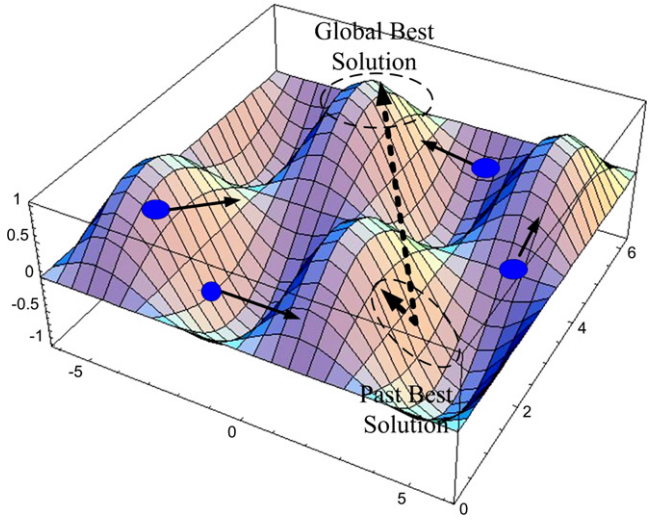

Fig. 1. Search concept of particle swarm optimization.

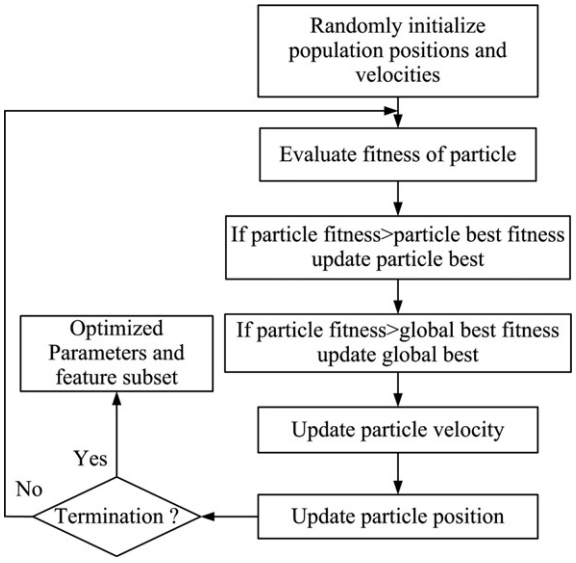

Fig. 3. The flowchart of PSO algorithm.

tion vector is saved for the particle. If the particle's fitness is better than the global best fitness, then the position vector is saved for the global best. Finally the particle's velocity and position are updated until the termination condition is satisfied.

4. Experiment results

The platform adopted to develop the PSO + SVM approach is a PC with the following features: Intel Pentium IV 3.0 GHz CPU, 512 MB RAM, a Windows XP operating system and the Visual C++ 6.0 development environment. To measure the performance of the developed PSO + SVM approach, the following datasets were used: Australian, Boston housing, Breast cancer, Bupa live, German, Ionosphere, Pima, Sonar, Car Evaluation Database, Glass, Teaching Assistant Evaluation, Vehicle, Vowel and Wine, from the UCI machine learning repository (Hettich, Blake, & Merz, 1998), and Bioinformatics, taken from Hsu et al. (2003). The predicted data of the Boston housing dataset was transformed from continuous into binary class (Fung & Mangasarian, 2003). Table 1 presents the properties of these datasets.

Scaling was applied to prevent feature values in greater numeric ranges from dominating those in smaller numeric ranges, and to prevent numerical difficulties in the calculation. Experimental results obtained in this study demonstrate that scaling the feature value improves the classification accuracy of SVM. In general, the range of each feature value can be linearly scaled to the range $[-1, +1]$.

The k-fold method presented by Salzberg (1997) was employed in the experiments. In this study, the value of k is set to 10. Thus, the dataset was split into 10 parts, with each part of the data sharing the same proportion of each class of data. Nine data parts were applied in the training process, while the remaining one was utilized in the testing process (Han & Kamber, 2003). The program was run 10 times to enable each slice of data to take a turn as the testing data. The rate of accuracy in classification of this experiment was computed by summing the individual accuracy rate for each run of testing, and then dividing the total by 10. Since the number of data in each class is not a multiple of 10, the dataset cannot be partitioned fairly. However, the ratio of the number of data in the training set to the number of data in the validation set was maintained as closely as possible to 9:1. Fig. 4 shows the architecture of the developed PSO-based parameter determination and feature selection approach for SVM.

Through initial experiment, the parameter values of the developed PSO + SVM approach were set as follows. Both the cognition learning factor c_1 and the social learning factor c_2 were set to 2. When not considering feature selection, the number of particles and generations were found to be 6 and 50; thus the total number of solutions evaluated was 300. With feature selection, the number of features selected for use can be obtained by the PSO + SVM approach. Since the PSO + SVM approach has a larger solution space, in terms of number of features, the number of solution evaluated is also larger. The number of solution evaluated was raised to 2000 by setting the number of particles and generations to 8 and 250, respectively. The searching range of parameter C of SVM was between 0.01 and 35,000, while the searching range of parameter Υ of SVM was between 0.0001 and 32 (Lin & Lin, 2003).

Table 1
Dataset from the UCI repository

No	Dataset	No. of classes	No. of instances	No. of features
1	Australian	2	653	15
2	Bioinformatics	3	391	20
3	Boston housing	2	1012	13
4	Breast cancer	2	683	10
5	Bupa live	2	345	6
6	Car evaluation	4	1728	6
7	Cleveland heart	2	296	13
8	German	2	1000	30
9	Glass	6	214	9
10	Ionosphere	2	351	34
11	Iris	3	150	4
12	Pima	2	768	8
13	Sonar	2	208	60
14	Teaching assistant evaluation	3	151	5
15	Vehicle	4	846	18
16	Vowel	11	528	10
17	Wine	3	175	13

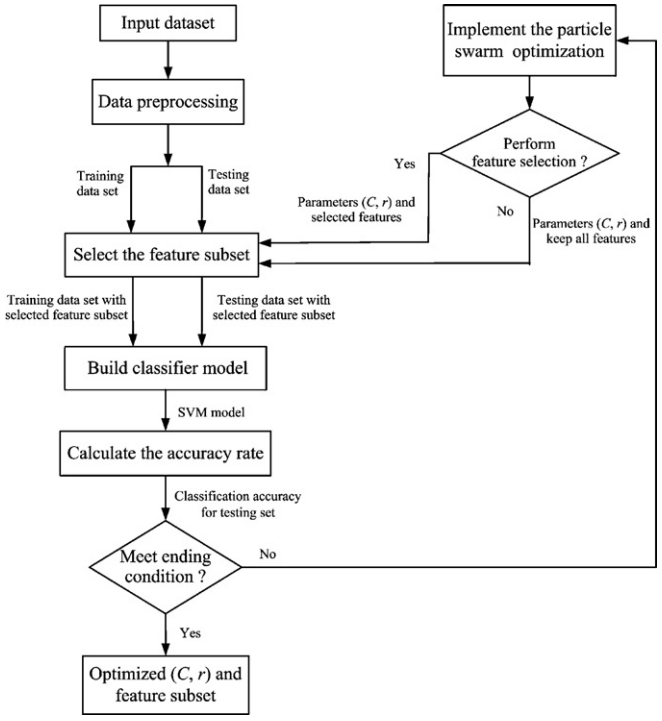

Fig. 4. The architecture of the proposed PSO-based parameters determination and feature selection approach for SVM.

Table 2
Comparison between the PSO + SVM, NSVM, SVM, LSVM and approaches proposed by Fung & Mangasarian and Liao et al. (%)

Dataset	PSO + SVM	NSVM	SVM	LSVM	Gaussian kernel	Polynomial kernel	Sigmoid kernel
Boston housing	99.90[a]	86.60	85.80	86.60	–	–	–
Bupa liver	80.52[a]	70.20	69.30	70.20	71.35	72.85	73.17
Cleveland heart	87.83[a]	86.30	85.90	86.30	85.11	84.67	85.17
Ionosphere	97.20[a]	89.80	88.30	89.80	93.12	92.15	94.37
Pima	80.21[a]	77.00	77.10	77.00	–	–	–
Breast cancer	97.95[a]	–	–	–	96.37	96.37	96.23

– Approach did not use the dataset for test.
[a] The highest classification accuracy rate among approaches.

The results of the developed PSO + SVM approach without feature selection were compared with those of Fung and Mangasarian (2003) and Liao et al. (2004). Fung and Mangasarian tested several UCI datasets using Newton SVM (NSVM), SVM and Lagrangina SVM (LSVM) without feature selection. Liao et al. employed three kernel functions, a Gaussian kernel, a polynominal kernel and a sigmoid kernel in the SVM to test several datasets from UCI. Table 2 shows the results of these approaches. All of the accuracy rates of the developed PSO + SVM approach are better than those obtained with Fung and Mangasarian. The developed PSO + SVM approach generated the best C and Υ values, yielding a higher classification accuracy rate across different datasets.

Table 3
Comparison between the PSO + SVM and GA + SVM approach proposed by Huang et al. (%)

Dataset	Without feature selection		With feature selection	
	PSO + SVM	GA + SVM	PSO + SVM	GA + SVM
Australian	88.09	88.09[b]	91.03[a]	88.10[b]
Breast cancer	97.95[a]	94.23	99.18[a]	96.19
Cleveland heart	88.17	94.58[a]	92.83	94.80[a]
German	79.00	84.24[a,b]	81.62	85.60[a,b]
Ionosphere	97.50[a]	96.61	99.01[a]	98.56
Iris	98.00[a]	97.56	99.20	100.00[a]
Pima	80.19	82.98[a]	82.68[a]	81.50
Sonar	88.32	95.22[a]	96.26	98.00[a]
Vehicle	88.71[a]	85.87	89.83[a]	84.06
Vowel	99.27[a]	95.13[b]	100.00[a]	99.30[b]

[a] The higher classification accuracy rate between two approaches.
[b] Inconsistency is due to the different version of dataset used.

The results obtained by the developed PSO + SVM approach with/without feature selection were compared with those of GA + SVM developed by Huang et al. (2006) The classification accuracy rates are cited from their original papers. Table 3 shows a comparison of the results. Without feature selection, the PSO + SVM approach yielded a higher classification accuracy rate in five datasets, while the GA + SVM approach did so in four. With feature selection, the PSO + SVM approach yielded the higher classification accuracy rate in six datasets, while the GA + SVM approach did so in four. Thus, the developed PSO + SVM approach yielded more appropriate parameters and subset, giving higher classification accuracy rate across different datasets.

Table 4
Experimental results of the developed PSO + SVM approach with and without feature selection and grid search (%)

Dataset	(1) PSO + SVM without feature selection	(2) PSO + SVM with feature selection	(3) grid search	Pair t test (1) v.s. (3) P-Value	Pair t test (2) v.s. (3) P-Value
Australian	88.09	91.03	84.54	<0.001	<0.001
Bioinformatics	89.09	90.63	83.92	<0.001	<0.001
Boston housing	99.90	100.00	99.80	0.099	0.099
Breast cancer	97.95	99.18	96.64	<0.001	<0.001
Bupa live	80.81	82.05	71.83	<0.001	<0.001
Car evaluation	99.89	99.68	99.89	0.161	0.067
Cleveland heart	88.17	92.83	81.37	<0.001	<0.001
German	79.00	81.62	75.30	<0.001	<0.001
Glass	78.04	85.26	70.61	<0.001	<0.001
Ionosphere	97.50	99.01	93.08	<0.001	<0.001
Iris	98.00	99.20	96.00	<0.001	<0.001
Pima	80.19	82.68	76.69	<0.001	<0.001
Sonar	88.32	96.26	87.90	0.007	<0.001
Teaching assistant evaluation	77.09	82.63	64.26	<0.001	<0.001
Vehicle	88.71	89.83	84.28	<0.001	<0.001
Vowel	99.27	100.00	98.91	0.011	<0.001
Wine	99.56	100.00	96.60	<0.001	<0.001

Confidence level $\alpha = 0.05$.

Table 5
Experimental results of the developed PSO + SVM approach with and without feature selection

Dataset	PSO + SVM with feature selection			PSO + SVM without feature selection	Pair t test P-value
	No. of original features	No. of selected features	Average accuracy rate (%)	Average accuracy rate (%)	
Australian	15	9.00 ± 2.01	91.03	88.09	<0.001
Bioinformatics	20	15.00 ± 2.32	90.63	89.09	<0.001
Boston housing	13	7.00 ± 1.00	100.00	99.90	0.500
Breast cancer	10	6.00 ± 1.29	99.18	97.95	<0.001
Bupa live	6	4.00 ± 0.88	82.05	80.81	0.165
Car evaluation	6	5.00 ± 0.44	99.68	99.89	0.067
Cleveland heart	13	8.00 ± 1.69	92.83	88.17	<0.001
German	30	18.00 ± 3.49	81.62	79.00	<0.001
Glass	9	5.00 ± 1.00	85.26	78.04	<0.001
Ionosphere	34	21.00 ± 3.23	99.01	97.50	<0.001
Iris	4	2.00 ± 0.64	99.20	98.00	<0.001
Pima	8	5.00 ± 1.26	82.68	80.19	<0.001
Sonar	60	37.00 ± 4.75	96.26	88.32	<0.001
Teaching assistant evaluation	5	3.00 ± 0.85	82.63	77.09	<0.001
Vehicle	18	13.00 ± 1.85	89.83	88.71	<0.001
Vowel	10	7.00 ± 0.95	100.00	99.27	<0.001
Wine	13	8.00 ± 1.56	100.00	99.56	0.022

Confidence level $\alpha = 0.05$.

An experiment using seventeen datasets from UCI was performed to further compare the results of the developed PSO + SVM approach with and without feature selection. The results obtained were compared with those of the grid search, as shown in Table 4. Results obtained using the developed PSO + SVM approach with and without feature selection were better than those of grid search in all cases examined. The use of a feature selection was found to improve the classification accuracy rate for each dataset except the Car Evaluation dataset. Only the classification accuracy rate of the Boston housing dataset and Car Evaluation dataset did not exhibit a significant improvement between the developed PSO + SVM approach and the grid search. This result indicates that good results are also obtainable from models with few features, clearly revealing that certain features are redundant or insignificant relative to particular classification problems. Undoubtedly, the PSO + SVM approach can simultaneously determine the parameter values and find a subset of features without lowering SVM classification accuracy.

Finally, to identify any differences in the classification accuracy rates of PSO + SVM with and without feature selection, the results of PSO + SVM with and without feature selection were compared, and are presented in Table 5. As shown in the table, only Boston housing, Bupa live, and Car Evaluation did not exhibit a statistical significant difference. Therefore, the PSO + SVM with feature selection has better performance than that of PSO + SVM without feature selection.

5. Conclusions and future research

This study presents a particle swarm optimization-based approach, capable of searching for the optimal parameter values for SVM to obtain a subset of beneficial features. This optimal subset of features is then adopted in both training and testing to obtain the optimal outcomes in classification. Comparison of the obtained results with those of other approaches demonstrates that the developed PSO + SVM approach has a better classification accuracy than others tested. After using feature selection in the experiment, the PSO + SVM approach is applied to eliminate unnecessary or insignificant features, and effectively determine the parameter values, in turn improving the overall classification results.

Results of this study were obtained with an RBF kernel function. However, other kernel parameters can also be optimized using the same approach. Experimental results obtained from UCI datasets, other public datasets and real-world problems can be tested in the future to verify and extend this approach.

Acknowledgment

The authors would like to thank the National Science Council of the Republic of China, Taiwan for financially supporting this research under Contract No. NSC96-2416-H-211-002.

References

Acir, N., Özdamar, Ö., & Guzelis, C. (2006). Automatic classification of auditory brainstem responses using SVM-based feature selection algorithm for threshold detection. *Engineering Applications of Artificial Intelligence, 19*, 209–218.

Burgers, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery, 2*, 121–167.

Cao, L. J., & Tay, F. E. H. (2003). Support vector machine with adaptive parameters in financial time series forecasting. *IEEE Transactions on Neural Network, 14*(6), 1506–1518.

Chen, R.-C., & Hsieh, C.-H. (2006). Web page classification based on a support vector machine using a weighed vote schema. *Expert Systems with Applications, 31*, 427–435.

Fung, G., & Mangasarian, O. L. (2003). Finite newton method for lagrangian support vector machine classification. *Neurocomputing, 55*, 39–55.

Gold, C., Holub, A., & Sollich, P. (2005). Bayesian approach to feature selection and parameter tuning for support vector machine classifiers. *Neural Networks, 18*, 693–701.

Han, J., & Kamber, M. (2003). *Data mining: concepts and techniques*. San Francisoc: Morgan Kaufmann.

Hettich, S., Blake, C., & Merz, C. (1998). UCI repository of machine information and computer sciences, available at <http://www.ics.uci.edu/~mlearn/MLRepository.html>.

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2003). A practical guide to support vector classification. Technical report, University of National Taiwan, Department of Computer Science and Information Engineering, July, pp. 1–12.

Huang, C.-L., Chen, M.-C., & Wang, C.-J. (2006). Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications*. doi:10.1016/j.eswa.2006.07.007.

Huang, C.-J., Lai, W.-K., Luo, R.-L., & Yan, Y.-L. (2005). Application of support vector machine to bandwidth reservation in sectored cellular communications. *Engineering Applications of Artificial Intelligence, 18*, 585–594.

Huang, C.-L., & Wang, C.-J. (2006). A GA-based feature selection and parameters optimization for support vector machines. *Expert Systems with Applications, 31*, 231–240.

Jack, L. B., & Nandi, A. K. (2002). Fault detection using support vector machines and artificial neural networks, augmented by genetic algorithms. *Mechanical Systems and Signal Processing, 16*, 373–390.

Keerthi, S. S., & Lin, C.-J. (2003). Asymptotic behaviors of support vector machines with gaussian kernel. *Neural Computation, 15*, 1667–1689.

Kennedy, J., & Eberhart, R. C. (1995). Particle swarm optimization. *Proceedings of IEEE Conference on Neural Network, 4*, 1942–1948.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence, 97*, 273–324.

Liao, Y., Fang, S.-C., & Nuttle, H. L. W. (2004). A neural network model with bounded-weights for pattern classification. *Computers and Operations Research, 31*, 1411–1426.

Liang, J.-Z. (2004). SVM multi-classifier and web document classification. *Proceedings of the Third International Conference on Machine Learning and Cybernetics, 3*, 1347–1351.

Lin, H.-T., & Lin, C.-J. (2003). A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods. Technical report, University of National Taiwan, Department of Computer Science and Information Engineering. March, pp. 1–32.

Liu, H., & Motoda, H. (1998). *Feature Selection for knowledge discovery and data mining*. Boston: Kluwer Academic.

Müller, K. R., Mike, S., Rätsch, G., Tsuda, K., & Schölkopf, B. (2001). An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks, 12*(2), 181–201.

Ng, J., & Gong, S. (2002). Composite support vector machines for detection of faces across views and pose estimation. *Image and Vision Computing, 20*, 359–368.

Pai, P.-F., & Hong, W.-C. (2005). Support vector machines with simulated annealing algorithms in electricity load forecasting. *Energy Conversion and Management, 46*, 2669–2688.

Pardo, M., & Sberveglieri, G. (2005). Classification of electronic nose data with support vector machines. *Sensors and Actuators B Chemical, 107*, 730–737.

Salzberg, S. L. (1997). On comparing classifiers: pitfalls to avoid and a recommended approach. *Data Mining and Knowledge Discovery, 1*, 317–327.

Samanta, B., Al-Balushi, K. R., & Al-Araimi, S. A. (2003). Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection. *Engineering Applications of Artificial Intelligence, 16*, 657–665.

SchÖlkopf, B., & Smola, A. J. (2002). *Learning with kernels*. London: MIT.

Shi, Y., & Eberhart, R. C. (1998). A Modified particle swarm optimizer. *Proceeding of the IEEE congress on Evolutionary Computation*, 69–73.

Shin, K.-S., Lee, T.-S., & Kim, H.-J. (2005). An application of support vector machines in bankruptcy prediction Model. *Expert Systems with Applications, 28*, 127–135.

Shon, T., Kim, Y., Lee, C., & Moon, J. (2005). A machine learning framework for network anomaly detection using SVM and GA. In *Proceedings of IEEE Workshop on Information Assurance and Security* 2, pp. 176–183.

Valentini, G. (2002). Gene expression data analysis of human lymphoma using support vector machines and output coding ensembles. *Artificial Intelligence in Medicine, 26*, 281–304.

Valentini, G., Muselli, M., & Ruffino, F. (2004). Cancer recognition with bagged ensembles of support vector machines. *Neurocomputing, 56*, 461–466.

Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York: Springer.

Wang, J., Wu, X., & Zhang, C. (2005). Support vector machines based on *k*-means clustering for real-time business intelligence systems. *International Journal of Business Intelligence and Data Mining, 1*(1), 54–64.

Wei, Y., & Lin, C.-J. (2005). *Feature Extraction, Foundations and Applications*. Springer.

Zhang, Y. L., Guo, N., Du, H., & Li, W. H. (2005). Automated defect recognition of C-SAM image in IC packaging using support vector machines. *International Journal of Advanced Manufacturing Technology, 25*, 1191–1196.

Zhang, L., Jack, L. B., & Nandi, A. K. (2005). Fault detection using genetic programming. *Mechanical Systems and Signal Processing, 19*, 271–289.