Chapter 11
An Enhanced Support Vector Machines Model for Classification and Rule Generation

Ping-Feng Pai and Ming-Fu Hsu

Abstract. Based on statistical learning theory, support vector machines (SVM) model is an emerging machine learning technique solving classification problems with small sampling, non-linearity and high dimension. Data preprocessing, parameter selection, and rule generation influence performance of SVM models a lot. Thus, the main purpose of this chapter is to propose an enhanced support vector machines (ESVM) model which can integrate the abilities of data preprocessing, parameter selection and rule generation into a SVM model; and apply the ESVM model to solve real world problems. The structure of this chapter is organized as follows. Section 11.1 presents the purpose of classification and the basic concept of SVM models. Sections 11.2 and 11.3 introduce data preprocessing techniques, metaheuristics for selecting SVM models. Rule extraction of SVM models is addressed in Section 11.4. An enhanced SVM scheme and numerical results are illustrated in Section 11.5 and 11.6. Conclusions are made in Section 11.7.

Keywords: Support vector machines, Data preprocessing, Rule extraction, Classification.

11.1 Basic Concept of Classification and Support Vector Machines

The data mining technique observes enormous records comprising information about the target and input variables. Imagine that investors would like to classify the financial status based on characteristics of the firm, such as return on asset

Ping-Feng Pai
Department of Information Management, National Chi Nan University, Taiwan, ROC
e-mail: paipf@ncnu.edu.tw.

Ming-Fu Hsu
Department of International Business Studies, National Chi Nan University, Taiwan, ROC
e-mail: s97212903@ncnu.edu.tw

S. Koziel & X.-S. Yang (Eds.): Comput. Optimization, Methods and Algorithms, SCI 356, pp. 241–258.
springerlink.com

(ROA), quick ratio, and return on investment (ROI). This is a classification task and data mining techniques are suitable for this task. The goal of data mining is to build up a suitable model for a labeling process that approximates the original process as closely as possible. Thus, investors can adopt the well-developed model to learn the status of firm.

Support vector machines (SVM) were proposed by Vapnik [42, 43] originally for typical binary classification problems. The SVM implements the structural risk minimization (SRM) principle rather than the empirical risk minimization (ERM) principle employed by most traditional neural network models. The most important concept of SRM is the minimization of an upper bound to the generalization error instead of minimizing the training error. In addition, the SVM will be equivalent to solving a linear constrained quadratic programming (QP) problem, so that the solution for SVM is always unique and globally optimal [6, 12, 14, 41, 42, 43].

Given a training set of instance-base pairs (x_i, y_i), $i = 1,\ldots,m$, where $x_i \in R^n$ and $y_i \in \{\pm 1\}$, SVM determines an optimal separating hyperplane with the maximum margin by solving the following optimization problem:

$$\begin{aligned} &\min_{w,g} \quad \frac{1}{2} w^T w \\ &s.t. \quad y_i\left(w \cdot x_i + g\right) - 1 \geq 0 \end{aligned} \tag{11.1}$$

where w denotes the weight vector, and g denotes the bias term.

The Lagrange function's saddle point is the solution to the quadratic optimization problem:

$$L_h\left(w, g, \alpha\right) = \frac{1}{2} w^T \cdot w - \sum_{i=1}^{m} \left(\alpha_i y_i \left(w \cdot x_i + g\right) - 1\right) \tag{11.2}$$

where α_i is Lagrange multipliers and $\alpha_i \geq 0$.

To identify an optimal saddle point is necessary because the L_h must be minimized with respect to the primal variable w and $g$and maximized the non-negative dual variable α_i. By discriminating w and g, and proposing the Karush Kuhn-Tucker (KKT) condition for the optimum constrained function, L_h is transformed to the dual Lagrangian $L_E(\alpha)$:

$$\begin{aligned} &\max_{\alpha} \quad L_E(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \left\langle x_i, x_j \right\rangle \\ &s.t. \quad \alpha_i \geq 0, i = 1,\ldots,m \quad and \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \end{aligned} \tag{11.3}$$

Dual Lagrangian $L_E(\alpha)$ must be maximized with respect to non-negative α_i to identify the optimal hyperplane. The parameters w^* and g^* of the optimal hyperplane were determined by the solution α_i for the dual optimization problem. Therefore, the optimal hyperplane $f(x) = sign\left(\left\langle w^* \cdot x \right\rangle + g^*\right)$ can be illustrated as:

$$f(x) = sign\left(\sum_{i=1}^{m} y_i \alpha_i^* \langle x_i, x \rangle + g^*\right) \tag{11.4}$$

In a binary classification task, only a few subsets of the Lagrange multipliers α_i usually tend to be greater than zero. These vectors are the closest to the optimal hyperplane. The respective training vectors having non-zero α_i are called support vectors, as the optimal decision hyperplane $f(x,\alpha^*,g^*)$ depends on them exclusively. Figure 11.1 illustrates the basic structure of SVM.

Very few data sets in the real world are linearly separable. What makes SVM so remarkable is that the basic linear framework is easily extended to the case where the data set is not linearly separable. The fundamental concept behind this extension is to transform the input space where the data set is not linearly separable into a higher-dimensional space, where the data are linearly separable. Figure 11.2 illustrates the mapping concept of SVM.

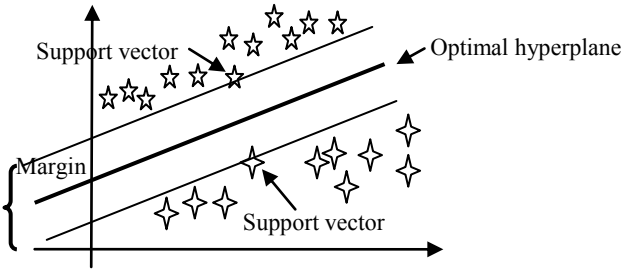

Fig. 11.1 The basic structure of the SVM [12]

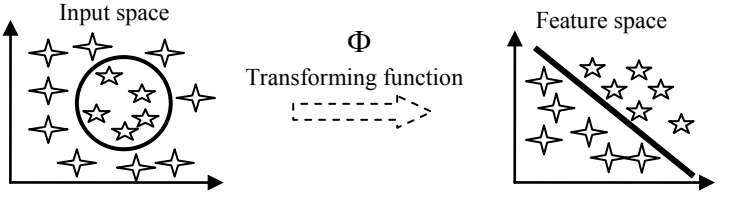

Fig. 11.2 Mapping a non-linear data set into a feature space [6]

In terms of the introduced slack variables, the problem of discovering the hyperplane with minimizing the training errors is illustrated as follows:

$$\begin{aligned} &\min_{w,g,\xi} \ \frac{1}{2} w^T \cdot w + C\sum_{i=1}^{m} \xi_i \\ &s.t. \ \ y_i\left(\langle w.x_i \rangle + g\right) + \xi - 1 \geq 0 \\ &\quad \xi_i \geq 0 \end{aligned} \tag{11.5}$$

where C is a penalty parameter on the training error, and ξ_i is the non-negative slack variable. The constant C used to determine the trade-off between margin size

and error. Observe that C is positive and cannot be zero; that is, we cannot simply ignore the slack variables by setting $C = 0$. With a large value for C, the optimization will try to discover a solution with a small number of non-zero slack variables because errors are costly [14]. Above all, it can be concluded that a large C implies a small margin, and a small C implies a large margin.

The Lagrangian method can be used to solve the optimization model, which is almost equivalent to the method for dealing with the optimization problem in the separable case. One has to maximize the dual variables Lagrangian:

$$\begin{aligned} &\max_{\alpha} \quad L_E(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \langle x_i \cdot x_j \rangle \\ &s.t. \quad 0 \le \alpha_i \le C, i = 1, \ldots, m \; and \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \end{aligned} \tag{11.6}$$

A dual Largrangian $L_E(\alpha)$ has to be maximized with respect to non-negative α_i under the constraints $\sum_{i=1}^{m} \alpha_i y_i = 0$ and $0 \le \alpha_i \le C$ to determine the optimal hyperplane. The penalty parameter C is an upper bound on α_i, and determined by the user.

The mapping function Φ is used to map the training samples from the input space into a higher-dimensional feature space. In Eq.11.6, the inner products are substituted by the kernel function $(\Phi(x_i) \cdot \Phi(y_i)) = K(x_i, x_j)$, and the nonlinear SVM dual Lagrangian $L_E(\alpha)$ shown in Eq.(11.7) is similar to that in the linear generalized case:

$$\begin{aligned} &L_E(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j K(x_i \cdot x_j) \\ &s.t. \quad 0 \le \alpha_i \le C, i = 1, \ldots, m \; and \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \end{aligned} \tag{11.7}$$

Hence, followed the steps illustrated in the linear generalized case, we derive the decision function of the following form:

$$f(x) = sign\left(\sum_{i=1}^{m} y_i \alpha_i^* \langle \Phi(x), \Phi(x_i) \rangle + g^* \right) = sign\left(\sum_{i=1}^{m} y_i \alpha_i^* \langle K(x, x_i) \rangle + g^* \right) \tag{11.8}$$

The function K is defined as the kernel function for generating the inner products to construct machines with different types of nonlinear decision hyperplane in the input space. There are several kernel functions, depicted as follows. The determination of kernel function type depends on the problem's complexity [12].

Radial Basis Function (RBF): $K(x, x_i) = \exp\left\{ - \|x - x_i\|^2 / 2\sigma^2 \right\}$

Polynomial kernel of degree d: $K(x, x_i) = (x, x_i)^d$

Sigmoid kernel: $K(x, x_i) = \tanh(K(x, x_i) + r)$

11.2 Data Preprocessing

Data sometimes are missing, noisy and inconsistent; and irrelevant or redundant attributes of data increase the computational complexity and decrease performance of data mining models. To be useful for data mining purposes, the original data need to be preprocessed in the form of cleaning, transformation, and reduction. The data without the preprocessing procedures would cause confusion for the data mining procedure and result in unreliable output.

11.2.1 Data Cleaning

The purpose of data cleaning is to fill in missing value, eliminate the noise (outliers), and correct the inconsistencies in the data. Let us look at the following approaches for missing value [9, 21, 35, 37]:

- Ignore the missing value.
- Fill in the missing value manually.
- Apply a global constant to replace the missing value.
- Apply the mean attribute to replace the missing value.
- Apply the most probable value to fill in the missing value.

Noise data (e.g., outlier) is a random error or variance in the measured data. Even a small number of extreme values can lead to different results and impair the conclusion. There are some smoothing methods (e.g., binning, regression and clustering) to offset the effect caused by a small number of extreme values [3, 28, 37, 44]. Human error in data entry, deliberate errors and data decay are some of the reasons for inconsistent data. Missing values, noise, and inconsistent data lead to inaccurate results. Data cleaning is the first step to analyzing the original data which would lead to reliable mining result. Figure 11.3 illustrates the original data processed by the procedure of data cleaning [9, 36].

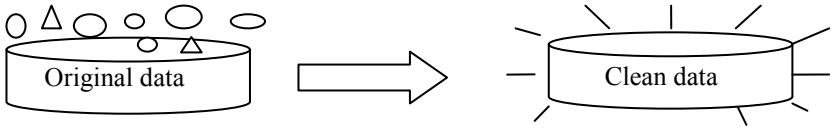

Fig. 11.3 Data cleaning [12]

11.2.2 Data Transformation

Data transformation is used to transform or consolidate data into forms suitable for the data mining process. Data transformation consists of the following processes [15, 17, 36, 38, 39]:

- Smoothing is employed to remove the noise from the data is illustrated in Fig. 11.4.
- Aggregation aggregates the data to construct the data cube for analysis.

- Generalization replaces the lower-level data with higher-level data.
- Normalization scales the attribute data to fall within a small specified range.

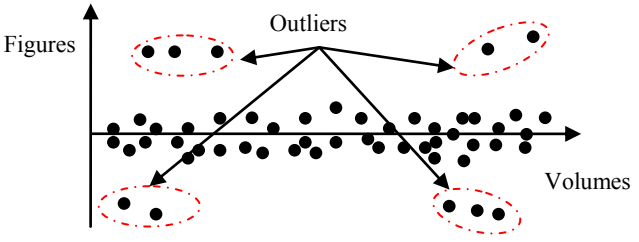

Fig. 11.4 The process of smoothing

11.2.3 Data Reduction

The purpose of the data reduction is to create a reduced representation of the dataset which is much smaller in volume yet closely sustains the integrity of the raw data. Dealing with the reduced data set enhances efficiency while producing the same analytical results. Data reduction consists of the following process [1, 2, 4, 5, 7, 18, 19, 24, 40, 45]:

- The aggregation of the data cube is employed to construct a data cube which is illustrated in Fig. 11.5.
- Attribute selection is used to remove the irrelevant, redundant or weak attributes, as shown in Fig. 11.6.
- Dimension reduction is used to reduce or compress the representation of the raw dataset. Raw data which can be reconstructed from the compressed data without losing any information is called lossless. In contrast, the approximation of the reconstructed raw data is called lossy.

Year 2008	
Quarter	Sales
Q 1	300
Q 2	400
Q 3	450
Q 4	550

Year 2009	
Quarter	Sales
Q 1	440
Q 2	410
Q 3	550
Q 4	600

Aggregation	
Years	Sales
2008	1700
2009	2000

Fig. 11.5 Aggregation of the data cube [12]

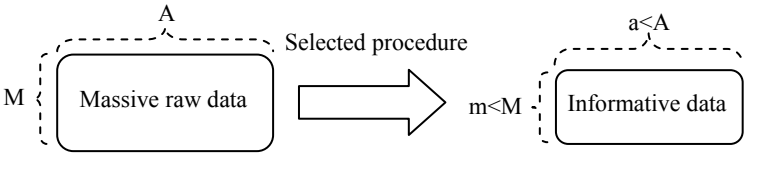

Fig. 11.6 Attribute selection [12]

11.3 Parameter Determination of Support Vector Machines by Meta-heuristics

Appropriate parameter setting can improve the performance of SVM models. Two parameters (C and σ) have to be determined in the SVM model with RBF kernel. The parameter C is the cost of penalty which influences the classification performance. If C is too large, the classification accuracy is very high in training data set, but very low in testing data set. If C is too small, the classification accuracy is inferior. The parameter σ has more influence than parameter C on classification outcome, because the value affects the partitioning outcome in the feature space. A large value for parameter σ leads to over-fitting, while a small value results in under-fitting [22]. The Grid search [24] is the most common approach to determine parameters of SVM models. Nevertheless, this approach is a local search technique, and tends to reach the local optima [20]. Furthermore, setting appropriate search intervals is an essential problem. A large search interval increases the computational complexity, while a small search interval would cause an inferior outcome. Some metaheuristics were proposed to select satisfactory parameters of SVM models [29, 30, 31, 32, 33, 34, 35]. The basic concept is to transfer the fitness functions of meta-heuristics into the forms of classification performance criteria (classification accuracy or error) of the SVM models. The fitness function of proposed metahuristics is used to measure the classification accuracy of the SVM model. Making the classification performance criteria acceptable for the metaheuristic algorithms is the most critical part of this procedure.

11.3.1 Genetic Algorithm

Holland [13] proposed the genetic algorithm (GA) to understand the adaptive processes of natural systems. Subsequently, they were employed for optimization and machine learning in the 1980's. Originally, GA was associated with the use of binary representation, but currently we can find it used with other types of representations and applied in many research domains. The basic principle is the principle of survival of the fittest. It tries to keep genetic information from generation to generation. The major merits of GA are their ability to find optimal or near optimal solutions with relatively modest computational requirements. The concept is briefly illustrated as follows and illustrated in Fig. 11.7. :

- Initialization: The initial population of chromosomes is established randomly.
- Evaluating fitness: Evaluate the fitness of each chromosome. The classification accuracy is used as the fitness function.
- Selection: Select a mating pair for reproduction.
- Crossover and mutation: Create new offspring by performing crossover and mutation operations.
- Next generation: Create a population for the next generation.
- Stop condition: If the number of generations equals a threshold, then the best chromosomes are presented as a solution; otherwise go back to step (b) [29, 31].

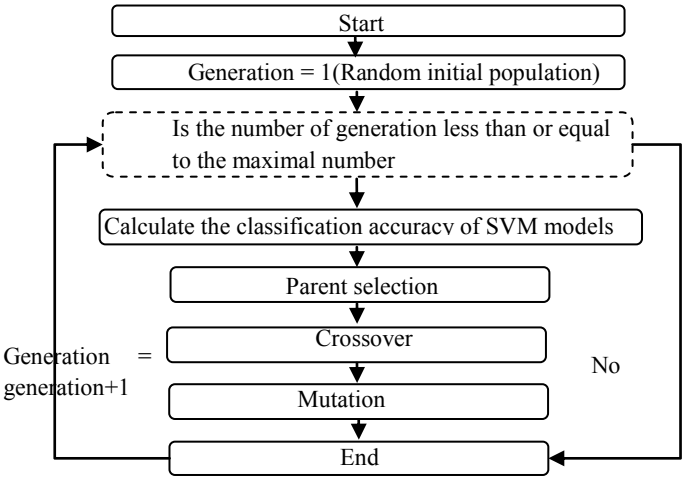

Fig. 11.7 The architecture of GA to determine parameters of SVM

11.3.2 Immune Algorithm

The immune algorithm (IA) [10] was based on the natural immune systems which efficiently distinguish all cells within the body and classify those cells as self or non-self cells. Non-self cells trigger a defense procedure which defends against foreign invaders. The antibodies are expressed by two SVM parameters. The classification error of SVM is contained in the denominator part of the affinity formula. Therefore, the reason for maximizing the affinity of IA is to minimize classification errors of the SVM model. IA search algorithm applied to determine the parameters of SVM is described as follows and illustrates in Fig. 11.8. :

- Initialization: Both the initialized antibody population and the population of the initial antibody were created randomly.
- Evaluation fitness: The classification error (CE) was treated as the fitness of IA.
- Affinity and similarity: When affinity values are high, the affinity and the similarity antibodies having higher activation levels of antigens are identified. To maintain the diversity of the antibodies stored in the memory cells, antibodies with a higher affinity value and a lower similarity value have a good likelihood of entering the memory cells. Eq. (11.9) is used to depict the affinity between the antibody and antigen:

$$Antigen = 1/1 + CE \tag{11.9}$$

A smaller CE indicates a higher affinity value. Eq. (11.10) is applied to illustrate the similarity between antibodies:

$$Antibodies = 1/1 + G_{ij} \tag{11.10}$$

where G_{ij} is the difference between the two classification errors calculated by the antibodies inside and outside the memory cells.

- Selection: Select the antibodies in the memory cells. Antibodies with higher values of *Antigen* are treated as candidates to enter the memory cell. However, the antibody candidates with $Antibodies_{ij}$ values exceeding the threshold are not qualified to enter the memory cell.
- Crossover and mutation: The antibody population is undergoing crossover and mutation. Crossover and mutation are used to generate new antibodies. When conducting the crossover operation, strings representing antibodies are paired randomly. Segments of paired strings between two predetermined break-points are swapped.
- Perform tabu search [11] on each antibody: Evaluate neighbor antibodies and adjust the tabu list. The antibody with the better classification error and not recorded on the tabu list is placed on the tabu list. If the best neighbor antibody is the same as one of the antibodies on the tabu list, then the next set of neighbor antibodies is generated and the classification error of the antibody calculated. The next set of neighbor antibodies is generated from the best neighbor antibodies in the current iteration.
- Current antibody selection by tabu search: If the best neighbor antibody is better than the current antibody, then the current antibody is replaced by the best neighbor antibody. Otherwise, the current antibody is retained.
- Next generation: From a population for the next generation.
- Stop criterion: If the number of epochs is equal to a given scale, then the best antibodies are presented as a solution; otherwise go to Step (b) [32, 33].

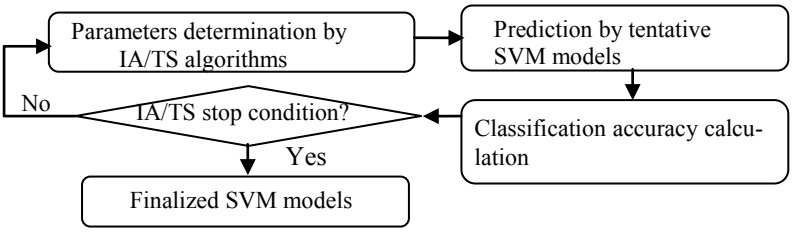

Fig. 11.8 The architecture of IA/TS to determine parameters of SVM

11.3.3 Particle Swarm Optimization

The particle swarm optimization (PSO) algorithm [16] is another population-based meta-heuristic inspired by swarm intelligence. It simulates the behavior of birds flocking to a promising position with sufficient food. A particle is considered as a point in a *G*-dimensional space and its status is characterized according to its position y_{ig} and velocity s_{ig}. The *G*-dimensional position for the particle *i* at iteration *t* is expressed as $y_i^t = \{y_{i1}^t, \ldots, y_{iG}^t\}$.

The velocity, which is also a *G*-dimensional vector, for particle *i* at iteration *t* is illustrated as $s_i^t = \{s_{i1}^t, \ldots, s_{iG}^t\}$. Let $b_i^t = \{b_{i1}^t, \ldots, b_{iG}^t\}$ be the best solution that particle *i* has obtained until iteration *t*, and $b_m^t = \{b_{m1}^t, \ldots, b_{mG}^t\}$ represents the best

solution from b_i^t in the population at iteration t. To search for an optimal solution, each particle changes its velocity according to cognition and sociality. Each particle then moves to a new potential solution. The use of PSO algorithm to select SVM parameters is described as follows. First, initialize a random population of particles and velocities. Second, define the fitness of each particle. The fitness function of PSO is represented as the classification accuracy of SVM models. Each particle's velocity is expressed by Eq. (11.11). For each particle, the procedure then moves to the next position according to Eq. (11.12).

$$S_{ig}^t = S_{ig}^{t-1} + c_1 j_1 \left(B_{ig}^t - y_{ig}^t\right) + c_2 j_2 \left(B_{mg}^t - y_{mg}^t\right), g = 1, \ldots, G \tag{11.11}$$

where c_1 is the cognitive learning factor, c_2 is the social learning factor, and j_1 and j_2 are the random numbers uniformly distributed in $U(0,1)$.

$$Y_{ig}^{t+1} = Y_{ig}^t + S_{ig}^t, g = 1, \ldots, G \tag{11.12}$$

Finally, if the termination criterion is reached, the algorithm stops; otherwise return to the step of fitness measurement [34]. The architecture of PSO is illustrated in Fig. 11.9.

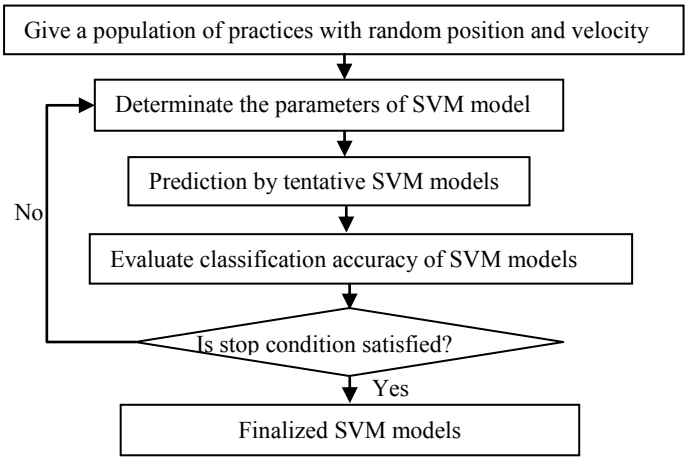

Fig. 11.9 The architecture of PSO to determine parameters of SVM

11.4 Rule Extraction Form Support Vector Machines

Support vector machines are state-of-the art data mining techniques which have proven their performance in many research domains. Unfortunately, while the models may provide a high accuracy compared to other data mining techniques, their comprehensibility is limited. In some areas, such as credit scoring, the lack of comprehensibility of a model is a main drawback causing reluctance of users to use the model [8]. Furthermore, when credit has been denied to a customer, the Equal Credit Opportunity Act of the US requires that the financial institution

provide specific reasons why the application was rejected; and indefinite and vague reasons for denial are illegal [23]. Comprehensibility can be added to SVM by extracting symbolic rules from the trained model. Rule extraction techniques would be used to open up the black box of SVM and generate comprehensible decision rules with approximately the same detective power as the model itself. There are two ways to open up the black box of SVM, as shown in Fig. 11.10.

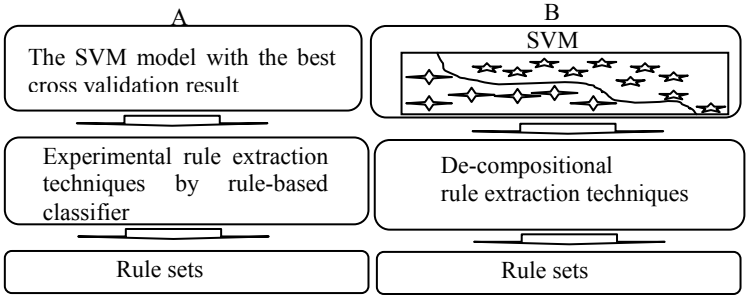

Fig. 11.10 Experimental (A) and de-compositional (B) rule extraction techniques [23]

The SVM with the best cross validation (CV) result is then fed into rule-based classifier (i.e., decision tree, rough set and so on) to derive the comprehensive decision rules for humans to understand (experimental rule extraction technique). The concept behind this procedure is the assumption that the trained model can more appropriately represent the data than can the original dataset. This is to say that the data of the best CV result is cleaner and free of curial conflicts. The CV is a re-sampling technique which adopts multiple random training and test subsamples to overcome the overfitting problem. Overfitting would lead to SVM losing its applicability, as shown in Fig. 11.11. The CV analysis would yield useful insights on the reliability of the SVM model with respect to sampling variation.

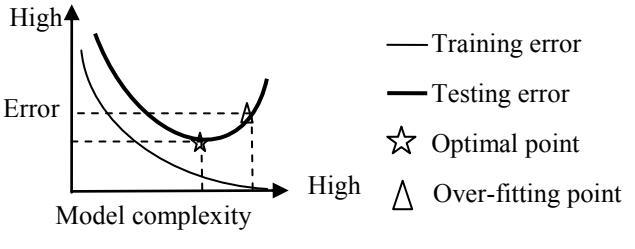

Fig. 11.11 Classification errors vs. model complexity of SVM models [12]

Decompositional rule extraction was proposed by Nunez et al. [25, 26] and proposes rule-defining regions based on the prototype and support vectors [23]. The representative of the obtained clusters is prototype vectors. The clustering task is overcome by vector quantization. There are two kinds of rules which can be

proposed: equation rules and interval rules, respectively corresponding to an ellipsoid and interval region, which can be built in the following manner [18]. Applying the prototype vector as center, an ellipsoid is constructed where the axes are determined by the support vector within the partition lying the furthest from the center. The long axes of the ellipsoid are defined by the straight line connecting these two vectors. The interval regions are defined from ellipsoids parallel to the coordinate axes [23]. Figure 11.12 is used to illustrate the basic structure of SVM + Prototype approach.

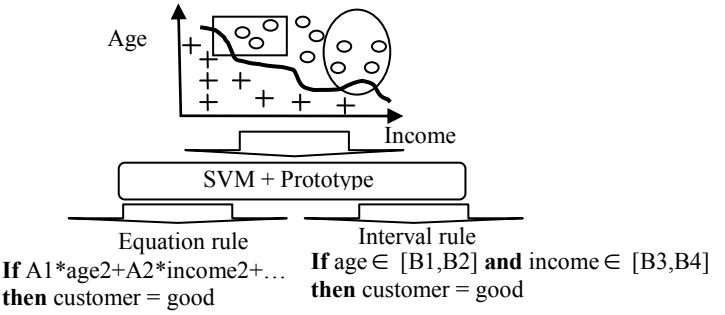

Fig. 11.12 SVM + Prototype model [25, 26]

11.5 The Proposed Enhanced SVM Model

In this section, the scheme of a proposed ESVM model is illustrated. Figure 11.13 shows the flowchart of the ESVM model, including functions of data preprocessing, parameter determination and rule generation. First, the raw data is processed by data-preprocessing techniques containing data cleaning, data transformation, feature selection, and dimension reduction. Second, the preprocessed data are divided into two sets: training and testing data sets. The training data set is used to select a data set used for rule generation. To prevent overfitting, a cross-validation (CV) procedure is performed at this stage. The testing data set is employed to examine the classification performance of a well-trained SVM model. Sequentially, metaheuristics are used to determine the SVM parameters. The training errors of SVM models are formulated as forms of fitness function of metaheuristics. Thus, each succeeding iteration produces a smaller classification error. The parameter search procedure is performed until the stop criterion of the metaheuristic is reached. The two parameters resulting in the smallest training error are then employed to undertake testing procedures and therefore testing accuracy is obtained. Finally, the CV training data set with the smallest testing error is utilized to derive decision rules by rule extraction mechanisms. Accordingly, the proposed ESVM model can provide decision rules as well as classification accuracy for decision makers.

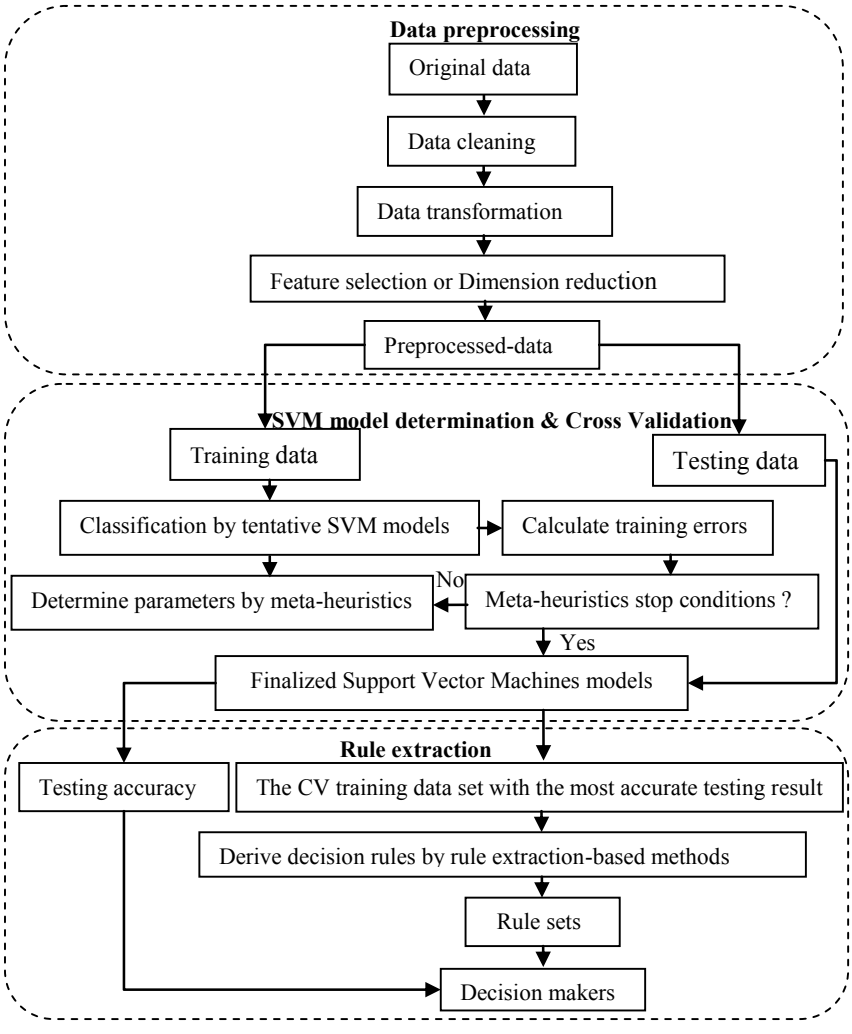

Fig. 11.13 The flowchart of the ESVM model

11.6 A Numerical Example and Empirical Results

A numerical example borrowed from Pai et al. [34] was used here to illustrate the classification and rule generation of SVM models. The original data used in this example contain 75 listed firms in Taiwan's stock market. These firms were divided into 25 fraudulent financial statement (FFS) firms and 50 non-fraudulent financial statement (non-FFS) firms. Published indication or proof of involvement in issuing FFS was found for the 25 FFS firms. The classification of a financial

statement as fraudulent is based on the Security of Futures Investor Protection Center in Taiwan (SFI) and the Financial Supervisory Commission of Taiwan (FSC) during the 1999-2005 reporting period. All the condition variables were used in the sample were generated from formal financial statements, such as balance sheets and income statements. The 18 features consist of 16 financial variables and two corporate governance variables were adopt in this study. The features selected by sequential forward selection (SFS) were illustrated in Table. 11.1. In addition, the grid search (GS) approach, genetic algorithms (GA), simulated annealing algorithms (SA) and particle swam optimization (PSO) were used to deal with the same data in selecting SVM parameters. The classification performances of four approaches in determining SVM parameters were summarized in Table 11.2. It can be concluded that the PSO algorithm was superior to the other three approaches in terms of average testing accuracy in this study. To demonstrate the generalization ability of SVM, three other classifiers, C4.5 decision tree (C4.5), multi-layer perception (MLP) neural networks, and RBF networks were examined. Table 11.3 indicates that the SVM model outperformed the other three classifiers in terms of testing accuracy. Moreover, the CART approach was used to derive "if-then" rules from the CV training data set with the best testing result. Thus, this procedure can help auditors to allocate limited audit resources. The decision rules derived from CART are listed in Table 11.4. It can be observed that the feature of "Pledged Share of Directors"is the first split point. This implies that shares pledged by directors are essential in detecting FFS by top management. Clearly, auditors have to concentrate on this critical signal in audit procedures.

Table 11.1 The selected features by feature selection [34]

Method	Features
SFS	A1: Net income to Fixed asset; A2: Net profit to Total asset; A3: Earnings before Interest and Tax; A4: Inventory to Sales; A5: Total debt to Total Asset; A6: Pledged shares of Directors

Table 11.2 Classification performance of four methods in determining SVM parameters [34]

Methods	Cross-validation					Accuracy (%)
	CV-1	CV-2	CV-3	CV-4	CV-5	
Grid	86.67	80	73.33	80	80	80
GA	80	86.67	80	86.67	86.67	84
SA	80	86.67	86.67	93.33	96.67	86.67
PSO	93.33	80	93.33	93.33	93.33	92

Table 11.3 Testing accuracy of six classifiers [34]

Classifier	Cross-validation					Accuracy (%)
	CV-1	CV-2	CV-3	CV-4	CV-5	
C4.5	73.33	80	86.67	93.33	86.67	84
MLP	73.33	86.67	80	86.67	86.67	82.67
RBFNN	86.67	80	80	86.67	80	82.67
SVM	93.33	86.67	93.33	93.33	93.33	92

Table 11.4 Decision rules derived from CART [34]

(1) If "pledged shares of directors" $\geq$ 44.405 , then "FFS"
(2) If "pledged shares of directors" $<$ 44.405 and "net profit to total assets" $<$ -0.3229 , then "FFS"
(3) If "pledged shares of directors" $<$ 44.405 , "net profit to total assets" $\geq$ -0.3229 and "net income to fixed assets" $\geq$ 0.0497 , then "non-FFS"
(4) If "pledged shares of directors" $<$ 44.405 , "net profit to total assets" $\geq$ -0.3229, "net income to fixed assets" $<$ 0.0497 and "earnings before interest and tax" $<$ -42220, then "non-FFS"
(5) If "pledged shares of directors" $<$ 44.405, "net profit to total assets" $\geq$ -0.3229, "net income to fixed assets" $<$ 0.0497, "earnings before interest and tax" $\geq$ -42220, and "total debt to total assets " $\geq$ 1.48 then, "FFS"
(6) If "pledged shares of directors" $<$ 44.405, "net profit to total assets" $\geq$ -0.3229, "net income to fixed assets" $<$ 0.0497, "earnings before interest and tax" $\geq$ -42220, and "total debt to total assets" $<$ 1.48 then, "non-FFS"

11.7 Conclusion

In this chapter, the three essential issues influencing the performance of SVM models were pointed out. The three issues are: data preprocessing, parameter determination and rule extraction. Some investigations have been conducted into each issue respectively. However, this chapter is the first study proposing an enhanced SVM model which deals with three issues at the same time. Thanks to data preprocessing procedure, the computation cost decreases and the classification accuracy increases. Furthermore, the ESVM model provides rules for decision makers. Rather than the expression of complicated mathematical functions, it is easy for decision makers to realize the relation and strength between condition attributes and outcome intuitively form a set of rules. These rules can be reasoned in both forward and backward ways. For the example in Section 11.6, the forward reasoning can provide a good direction for managers to improve the current financial status; and the backward reasoning can protect the wealth of investors and sustain the stability of financial market.

Acknowledgments. The authors would like to thank the National Science Council of the Republic of China, Taiwan for financially supporting this research under Contract No. 96-2628-E-260-001-MY3 & 99-2221-E-260-006.

References

1. Agarwal, S., Agrawal, R., Deshpande, P.M., Gupta, A., Naughton, J.F., Ramakrishnan, R., Sarawagi, S.: On the computation of multidimensional aggregates. In: Proc. Int. Conf. Very Large Data Bases, pp. 506–521 (1996)
2. Barbar'a, D., DuMouchel, W., Faloutos, C., Haas, P.J., Hellerstein, J.H., Ioannidis, Y., Jagadish, H.V., Johnson, T., Ng, R., Poosala, V., Ross, K.A., Servcik, K.C.: The New Jersey data reduction report. Bull. Technical Committee on Data Engineering 20, 3–45 (1997)
3. Ballou, D.P., Tayi, G.K.: Enhancing data quality in data warehouse environments. Comm. ACM 78, 42–73 (1999)
4. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classifcation and Regression Trees, Wadsworth International Group (1984)
5. Chakrabart, S., Cox, E., Frank, E., Guiting, R.H., Han, J., Jiang, X., Kamber, M., Lightstone, S.S., Nadeau, T.P., Neapolitan, R.E., Pyle, D., Refaat, M., Schneider, M., Teorey, T.J.I., Witten, H.: Data Mining: Know It All. Morgan Kaufmann, San Francisco (2008)
6. Taylor, J.S., Cristianini, N.: Support Vector Machines and other kernel-based learning methods. Cambridge University Press, Cambridge (2000)
7. Dash, M., Liu, H.: Feature selection methods for classification. Intell. Data Anal. (1), 131–156 (1997)
8. Dwyer, D.W., Kocagil, A.E., Stein, R.M.: Moody's kmv riskcalc v3.1 model (2004)
9. English, L.: Improving Data Warehouse and Business Information Quality: Methods for Reducing Costs and Increasing. John Wiley & Sons, Chichester (1999)
10. Farmer, J.D., Packard, N.H., Perelson, A.: The immune system, adaptation, and machine learning. Physica. D 22(1–3), 187–204 (1986)
11. Glover, F., Kelly, J.P., Laguna, M.: Genetic algorithms and tabu search: hybrids for optimization. Comput. Oper. Res. 22, 111–134 (1995)
12. Hamel, L.H.: Knowledge Discovery with Support Vector Machines. Wiley, Chichester (2009)
13. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
14. Huang, C.L., Chen, M.C., Wang, C.J.: Credit scoring with a data mining approach based on support vector machines. Expert Systems with Applications 33(4), 847–856 (2007)
15. Kennedy, R.L., Lee, Y., Van Roy, B., Reed, C.D., Lippman, R.P.: Solving Data Mining Problems Through Pattern Recognition. Prentice-Hall, Englewood Cliffs (1998)
16. Kennedy, J., Eberhart, R.: Particle swarm optimization, In Proceedings of IEEE conference on neural network, vol. 4, pp. 1942–1948 (1995)
17. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artif. Intell. 97, 273–324 (1997)
18. Langley, P., Simon, H.A., Bradshaw, G.L., Zytkow, J.M.: Scientific Discovery: Computational Explorations of the Creative Processes. MIT Press, Cambridge (1987)

19. Liu, H., Motoda, H.: Feature Extraction, Construction, and Selection: A Data Mining Perspective. Kluwer Academic Publishers, Dordrecht (1998)
20. Lin, S.W., Shiue, Y.R., Chen, S.C., Cheng, H.M.: Applying enhanced data mining approaches in predicting bank performance: A case of Taiwanese commercial banks. Expert Syst. Appl. (36), 11543–11551 (2009)
21. Loshin, D.: Enterprise Knowledge Management: The Data Quality Approach. Morgan Kaufmann, San Francisco (2001)
22. Lopez, F.G., Torres, G.M., Batista, B.M.: Solving feature subset selection problem by parallel scatter search. Eur. J. Oper. Res. (169), 477–489 (2006)
23. Martens, D., Baesens, B., Gestel, T.V., Vanthienen, J.: Comprehensible credit scoring models using rule extraction from support vector machines. Eur. J. Oper. Res. 183(3), 1466–1476 (2007)
24. Martin, D.: Early warning of bank failure a logit regression approach. J. Bank. Financ. (1), 249–276 (1977)
25. Nunez, H., Angulo, C., Catala, A.: Rule extraction from support vector machines. In: European Symposium on Artificial Neural Networks Proceedings, pp. 107–112 (2002)
26. Nunez, H., Angulo, C., Catala, A.: Rule based learning systems from SVM and RBFNN. Tendencias de la mineria de datos en espana, Red Espaola de Minera de Datos (2004)
27. Neter, J., Kutner, M.H., Nachtsheim, C.J., Wasserman, L.: Applied Linear Statistical Models. Irwin (1996)
28. Olson, J.E.: Data Quality: The Accuracy Dimension. Morgan Kaufmann, San Francisco (2003)
29. Pai, P.F., Hong, W.C.: Forecasting regional electricity load based on recurrent support vector machines with genetic algorithms. Electr. Pow. Syst. Res. 74(3), 417–425 (2005)
30. Pai, P.F., Lin, C.S.: A hybrid ARIMA and support vector machines model in stock price forecasting. Omega 33(6), 497–505 (2005)
31. Pai, P.F.: System reliability forecasting by support vector machines with genetic algorithms. Math. Comput. Model. 433(3-4), 262–274 (2006)
32. Pai, P.F., Chen, S.Y., Huang, C.W., Chang, Y.H.: Analyzing foreign exchange rates by rough set theory and directed acyclic graph support vector machines. Expert Syst. Appl. 37(8), 5993–5998 (2010)
33. Pai, P.F., Chang, Y.H., Hsu, M.F., Fu, J.C., Chen, H.H.: A hybrid kernel principal component analysis and support vector machines model for analyzing sonographic parotid gland in Sjogren's Syndrome. International Journal of Mathematical Modelling and Numerical Optimisation (2010) (in press)
34. Pai, P.F., Hsu, M.F., Wang, M.C.: A support vector machine-based model for detecting top management fraud. Knowl.-Based Syst. 24(2), 314–321 (2011)
35. Pyle, D.: Data Preparation for Data Mining. Morgan Kaufmann, San Francisco (1999)
36. Quinlan, J.R.: Unknown attribute values in induction. In: Proc. 1989 Int. Conf. Machine Learning (ICML 1989), Ithaca, NY, pp. 164–168 (1989)
37. Redman, T.: Data Quality: Management and Technology. Bantam Books (1992)
38. Ross, K., Srivastava, D.: Fast computation of sparse datacubes. In: Proc Int. Conf. Very Large Data Bases, pp. 116–125 (1997)
39. Sarawagi, S., Stonebraker, M.: Efficient organization of large multidimensional arrays. In: Proc. Int. Conf. Data Engineering, ICDE 1994 (1994)
40. Siedlecki, W., Sklansky, J.: On automatic feature selection. Int. J. Pattern Recognition and Artificial Intelligence (2), 197–220 (1988)

41. Scholkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge (2001)
42. Vapnik, V.: Statistical learning theory. John Wiley and Sons, New York (1998)
43. Vapnik, V., Golowich, S., Smola, A.: Support vector machine for function approximation, regression estimation, and signal processing. Advances in Neural Information processing System (9), 281–287 (1996)
44. Wang, R., Storey, V., Firth, C.: A framework for analysis of data quality research. IEEE Trans. Knowledge and Data Engineering (7), 623–640 (1995)
45. Zhao, Y., Deshpande, P.M., Naughton, J.F.: An array-based algorithm for simultaneous multi-dimensional aggregates. In: Proc. 1997 ACM-SIGMOD Int. Conf. Management of Data, pp. 159–170 (1997)