

# A Hybrid Particle Swarm Optimization and Support Vector Regression Model for Financial Time Series Forecasting

Hong-I Hsieh

Graduate Institute of Business Administration, Fu-Jen Catholic University  
No.510, Zhongzheng Rd., Xinzhuang Dist., New Taipei City 24205, Taiwan  
Tel: +886-2-2905-2653 E-mail: timon.hsieh@gmail.com

Tsung-Pei Lee

Department of International Trade and Finance, Fu-Jen Catholic University  
No.510, Zhongzheng Rd., Xinzhuang Dist., New Taipei City 24205, Taiwan  
Tel: +886-2-2905- 3977 E-mail: 004167@mail.fju.edu.tw

Tian-Shyug Lee (Corresponding author)

Department of Business Administration, Fu Jen Catholic University  
No.510, Zhongzheng Rd., Xinzhuang Dist., New Taipei City 24205, Taiwan  
Tel: +886-2-2905-2653 E-mail: 036665@mail.fju.edu.tw

Received: March 2, 2011 Accepted: March 12, 2011 doi:10.5430/ijba.v2n2p48

*This research was partially supported by the National Science Council of the Republic of China under Grant Number NSC 97-2221-E-030-011-MY2.*

## Abstract

In this paper, a time series forecasting approach by integrating particle swarm optimization (PSO) and support vector regression (SVR) is proposed. SVR has been widely applied in time series predictions. However, no general guidelines are available to choose the free parameters of an SVR model. The proposed approach uses PSO to search the optimal parameters for model selections in the hope of improving the performance of SVR. In order to evaluate the performance of the proposed approach, the TAIEX (Taiwan Stock Exchange Capitalization Weighted Stock Index) closing cash index is used as the illustrative example. Experimental results show that the proposed model outperforms the traditional SVR model and provides an alternative in financial time series forecasting.

**Keywords:** Data mining, Particle swarm optimization, Support vector regression, Time series forecasting

## 1. Introduction

As financial time series are inherently noise and non-stationary, it is regarded as one of the most challenging applications of time series forecasting. Neural networks have been found to be useful techniques for modeling time series due to their ability to capture subtle functional relationships among the empirical data even though the underlying relationships are unknown or hard to describe (Zhang, Patuwo & Hu, 1998; Vellido, Lisboa & Vaughan, 1999; Lee & Chen, 2002; Lee & Chiu, 2002).

Support vector machine (SVM) is a novel neural network algorithm based on statistical learning theory (Vapnik, 2000). It can lead to great potential and superior performance in practical applications. This is largely due to the structure risk minimization principles in SVM. Therefore, the SVM has attracted the interest of researchers and has been applied in many applications such as texture classification, image recognition, data mining and bioinformatics (Norinder, 2003; Li, Kwok, Zhu & Wang, 2003; Shin, Lee & Kim, 2005). With introduction of Vapnik's  $\epsilon$ -insensitivity loss function, the regression model of SVMs, called support vector regression (SVR), has been receiving increasing attention to solve nonlinear regression problems. It has been successfully applied in different problems of time series prediction such as

production value forecast of machinery industry, engine reliability prediction and financial time series forecasting (Tay & Cao, 2001, 2003; Cao, 2003; Kim, 2003). Since there are many successful results of utilizing SVR in time series prediction, it motivates our research in using SVR for financial time series forecasting.

In the modeling of financial time series using SVR, one of the key problems is how to select model parameters correctly, which plays an important role in good generalization performance (Tay & Cao, 2001; Kim, 2003; Cherkassky & Yunqian, 2004; Park, Kim & Ahn, 2005). In the early stage, these parameters were determined arbitrarily or searched through trial-and-error method. Hsu, Chang, and Lin (2000) recommend a “grid-search” method to find the best combination of parameters. However, it might be a time-consuming task, especially when searching in a very large data set.

To overcome the drawback in determining the three parameters of SVR, this study proposes a particle swarm optimization (PSO)-SVM approach by integrating PSO in the model selection process of SVR. In order to evaluate the performance of the proposed approach, the TAIEX (Taiwan Stock Exchange Capitalization Weighted Stock Index) closing cash index is used as the illustrative example.

The rest of this paper is organized as follows. Section 2 gives a brief introduction about particle swarm optimization and support vector regression. The proposed PSO-SVR forecasting model is described in Section 3. Section 4 presents the experimental results from the TAIEX closing cash index dataset. The paper is concluded in Section 5.

## 2. Methodology

### 2.1 Particle Swarm Optimization

Particle swarm optimization (PSO) is a population-based heuristic method developed by Kennedy and Eberhart (1995). The PSO algorithm is inspired by the collective motion of biological organisms, such as bird flocking and fish schooling, to simulate the seeking behavior to a food source (Bratton & Kennedy, 2007). A PSO algorithm is initialized with a population of random particles treated as a point in  $D$ -dimensional search space. To find the optimum solution, each particle adjusts the direction through the best experience which it has found ( $pbest$ ) and the best experience been found by all other members ( $gbest$ ). Therefore, the particles fly around in a multidimensional space towards the better area over the search process.

The PSO system initially has a population of random solutions and then searches for optimum solution by updating process. Each particle consists of three vectors: the position for  $i$ th individual particle  $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ , the best previous position  $pbest$  that the  $i$ th particle has found  $\mathbf{p}_i = (p_{i1}, p_{i2}, \dots, p_{iD})$ , and its velocity  $\mathbf{v}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ . The performance of each particle is measured using a fitness function varying from problem in hand. During the iterative procedure, the particle's velocity and position are updated by:

$$\mathbf{v}_{id}^{new} = w \times \mathbf{v}_{id}^{old} + c_1 \times rand() \times (\mathbf{p}_{id} - \mathbf{x}_{id}^{old}) + c_2 \times Rnd() \times (\mathbf{p}_{gd} - \mathbf{x}_{id}^{old}) \quad (2.1)$$

$$\mathbf{x}_{id}^{new} = \mathbf{x}_{id}^{old} + \mathbf{v}_{id}^{new} \quad (2.2)$$

where  $c_1$  and  $c_2$  are two positive acceleration constants with the value of 2.0 suggested by Eberhart and Shi (2001b),  $w$  is an inertia weight,  $p_{id}$  is the  $pbest$  of  $i$ th particle and  $p_{gd}$  is the  $gbest$  of the group, and  $rand()$  and  $Rnd()$  are two random numbers uniformly generated from [0,1]. In a PSO system, particles change their positions at each time step until a relatively unchanging position has been encountered or a maximum number of iterations has been met. For further details regarding PSO, please refer to Eberhart and Shi (2001a), Eberhart and Shi (2001b), Kennedy, Eberhart, and Shi (2001), and Bratton and Kennedy (2007).

### 2.2 Support Vector Regression

The support vector regression is an adaptation of recently introduced statistical/ machine learning theory based classification paradigm namely, support vector machine. For illustrating the concept of SVR, a typical regression problem is formulated. Given the training data  $\{(\mathbf{x}_i, y_i), \dots, (\mathbf{x}_n, y_n)\}$ ,  $\mathbf{x} \in \mathbb{R}^n$ ,  $y \in \mathbb{R}$ , where  $\mathbf{x}_i$  is the  $i$ th input vector and  $y_i$  is the  $i$ th desired value. The objective is to find an unknown regression function,  $y = g(x)$ , and SVR approximates the function in the following form:

$$f(x) = \mathbf{w} \Phi(\mathbf{x}) + b \quad (2.3)$$

where  $\Phi(\mathbf{x})$  is the features, which is nonlinear mapped from the input space  $\mathbf{x}$ ;  $b$  is a constant, and  $\mathbf{w}$  denotes the weight vector estimated by minimizing the regularized risk function comprising the empirical risk  $R_{emp}[f]$  and a complexity term  $\|\mathbf{w}\|^2$ :

$$R_{reg}[f] = R_{emp}[f] + \lambda \|\mathbf{w}\|^2 = \frac{1}{l} \sum_{i=1}^l L(f(\mathbf{x}_i) - y_i) + \lambda \|\mathbf{w}\|^2 \quad (2.4)$$

where  $L(f(\mathbf{x}_i) - y_i)$  is the cost function. A number of cost functions such as the Laplace, Huber, and the Vapnik's  $\varepsilon$ -insensitive loss function can be used in the risk function (Gunn, 1998). The commonly used loss function proposed by Vapnik (2000) given as

$$L(f(\mathbf{x}) - y)_\varepsilon = \begin{cases} |f(\mathbf{x}) - y| - \varepsilon & \text{for } |f(\mathbf{x}) - y| \geq \varepsilon \\ 0 & \text{otherwise} \end{cases} \quad (2.5)$$

where  $\varepsilon$  defines the radius of a tube around the regression function outlined as Figure 1. Introduction of slack variable  $\xi^+$ ,  $\xi^-$  for the data points outside the tube, then the task in (2.4) becomes

$$\begin{aligned} \text{Min: } & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^l (\xi_i^+ + \xi_i^-) \\ \text{st. } & y_i - w_i \Phi(x_i) + b \leq \varepsilon + \xi_i^+ \\ & w_i \Phi(x_i) + b - y_i \leq \varepsilon + \xi_i^- \\ & \xi_i^+ + \xi_i^- \geq 0 \end{aligned} \quad (2.6)$$

where  $C$  in (2.6) is the regularized constant, and is considered to specify the trade-off between the empirical error and the flatness of the function. The vector  $\mathbf{w}$  can be shown as  $\mathbf{w} = \sum_{i=1}^l (\alpha_i - \alpha_i^*) \Phi(x_i)$ . Hence, the general form of

the SVR-based regression function can be written as

$$f(x) = \sum_{i=1}^l (\alpha_i - \alpha_i^*) \Phi(x_i) \Phi(x) + b = \sum_{i=1}^l (\alpha_i - \alpha_i^*) K(x_i, x) + b \quad (2.7)$$

where  $K(x_i, x)$  is the so-called kernel function. Although several choices for the kernel functions are available, the most widely used is the radial basis function (RBF) (Vapnik, 2000; Hsu, Chang & Lin, 2000). Thus, the RBF is applied in this study as the kernel function. The coefficients  $\alpha_i$  and  $\alpha_i^*$  can be obtained by solving the convex quadratic programming problem

$$\text{Max: } -\frac{1}{2} \sum_{i,j=1}^l (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) K(x_i, x_j) - \varepsilon \sum_{i=1}^l (\alpha_i - \alpha_i^*) + \sum_{i=1}^l y_i (\alpha_i - \alpha_i^*) \quad (2.8)$$

with the constraints:  $\sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0$ , and  $0 \leq \alpha_i, \alpha_i^* \leq C$ . The parameter  $b$  in (2.7) can be calculated as

$$\begin{aligned} b &= y_i - w_i \Phi(x_i) - \varepsilon \quad \text{for } \alpha_i \in (0, C) \\ b &= y_i - w_i \Phi(x_i) + \varepsilon \quad \text{for } \alpha_i^* \in (0, C) \end{aligned} \quad (2.9)$$

For more detailed function demonstrated, please refer to Smola and Schölkopf, (1998), Müller, Smola, Rätsch, Schölkopf, Kohlmorgen, and Vapnik (1999) and Vapnik (2000).

<Figure 1 about here>

### 3. Proposed PSO-SVR Forecasting Model

Because the selection of three parameters,  $C$ ,  $\varepsilon$ , and  $\gamma$ , of an SVR model is an important but difficult task, the proposed approach uses a *gbest* PSO, a neighborhood which each particle is able to communicate to every other particle, to search the optimal parameter setup for increasing the forecasting accuracy. Each particle in the swarm serves as a set of candidate parameters of an SVR model. Then, the fitness function of particles assessing the performance for PSO using the root mean square error (RMSE) criterion can be obtained in SVR training stage. If the stop criteria are not satisfied, then the change of the velocity and position for each particle are adjusted, and the SVR training process continues until stop criteria are met. Finally, a PSO-SVR model with minimum test error is built for financial time series forecasting. The framework of PSO-SVR approach is showed as Figure 2.

<Figure 2 about here>

#### 4. Experimental Results

For evaluating the performance of the proposed PSO-SVR forecasting model, the daily TAIEX closing cash index is used in this study. In forecasting the TAIEX closing cash index, the technical indicators are used as forecasting variables since technical indicators are the most widely adopted features in financial time series prediction (Balachandher, Fauzias & Lai, 2002; Leigh, Hightower & Modani, 2005). The five technical indicators, namely the previous day's cash market high ( $x_1$ ), low ( $x_2$ ), volume ( $x_3$ ), today's opening cash index ( $x_4$ ), and 10-days total amount weight stock price index (TAPI 10) ( $x_5$ ), determined by the review of domain experts and literatures (Wood, 2002; Leigh *et al.*, 2005) are selected as the forecasting variables for predicting TAIEX closing cash index. Please refer to Wood (2002) and Leigh *et al.* (2005) for more details about technical indicators. The five financial time series datasets and the daily TAIEX closing cash prices in the TAIEX dataset are depicted in Figure 3 and 4, respectively. The daily data of technical indicators and cash prices from January 2, 2003 to February 27, 2006 of TAIEX cash index provided by Capital Futures Corporation, Taipei, are collected. There are totally 781 data points in the dataset. Among, the first 623 data points (79.77% of the total sample points) are used as the training sample while the remaining 158 data points (20.23% of the total sample points) are used as the testing sample.

<Figure 3 about here>

<Figure 4 about here>

The prediction performance is evaluated using the following statistical metrics, namely, the root mean square error (RMSE), mean absolute difference (MAD), mean absolute percentage error (MAPE), directional symmetry (DS), correct up trend (CU) and correct down trend (CD). The definitions of these criteria can be found in Table 1. RMSE, MAD and MAPE are measures of the deviation between actual and predicted values. The smaller values of RMSE, MAPE and MAD, the closer are the predicted time series values to that of the actual values. They can be used to evaluate the prediction error. DS provides the correctness of the predicted direction of the cash index in terms of percentage. CU and CD provide the correctness of the predicted up trend and predicted down trend of the cash index in terms of percentage. DS, CU and CD can be utilized to evaluate the prediction accuracy.

<Table 1 about here>

The forecasting results of the proposed PSO-SVR model is compared to the traditional SVR model using grid-search method to find the best parameter setup. For building the SVR forecasting model, the LIBSVM package proposed by Chang and Lin (2001) is adapted in this study. The original data are normalized within the range of [0, 1] when using LIBSVM package. The linear scaling ensures the large value input variables do not overwhelm smaller value inputs and then helps to reduce prediction errors.

As mentioned in Section 2.2, the RBF kernel function is adapted in this study. It is well known that the performance (estimation accuracy) of SVR depends on setting of parameters. Thus, the selection of three parameters, regularization constant  $C$ , loss function  $\varepsilon$  and  $\gamma$  (the width of the RBF) of an SVR model is important to the accuracy of forecasting. The searching space of the three parameters is set in  $[2^{-10}, 2^{10}]$ . In the selection of parameters for modeling traditional SVR,  $C = 2^{10}$ ,  $\varepsilon = 2^1$ , and  $\gamma = 2^{-3}$  can be obtained by the analytic approach mentioned above. The parameter setup is then used as the start point of grid search for searching the best parameters. The testing results of the traditional SVR model with combinations of different parameter setups are summarized in Table 2. From Table 2, it can be found that the parameter setup ( $C = 2^{10}$ ,  $\varepsilon = 2^1$ , and  $\gamma = 2^{-3}$ ) gives the best forecasting result (minimum testing MSE) and is considered as the best parameter setup for traditional SVR model in forecasting TAIEX closing cash index.

<Table 2 about here>

In the modeling of the proposed PSO-SVR model, the parameters setups to be tested of PSO should first be assigned. In this study, two positive acceleration constants  $c_1$  and  $c_2$  each equal to 2.0 (Eberhart & Shi, 2001b), and inertia weight  $w$  is a dynamic value starting with 1.0 and updated by  $w_{it} = a \cdot w_{it} - 1$ , where  $a = 0.99$ . Since there are no general rules for the choice of the population size (Bratton & Kennedy, 2007), the trial and error approach was used with the size 10, 20, and 40, and the maximum number of iterations to be tested was set from 15 to 800. Finally, the population size and maximum number of iterations are set to be 20 and 50, respectively. In order to compare the performance with traditional SVR model, the searching space of three SVR parameters is also set in  $[2^{-10}, 2^{10}]$ . Table 3 summaries the testing results of the proposed PSO-SVR model with combinations of different parameter setups. It can be observed that the parameter setup ( $C = 1024$ ,  $\varepsilon = 0.00098$ , and  $\gamma = 1.7257$ ) gives the best forecasting result and hence is the best parameter setup for the proposed model in forecasting the TAIEX closing cash index.

<Table 3 about here>

The TAIEX closing cash price index forecasting results of the testing sample by traditional SVR and the proposed PSO-SVR models are computed and listed in Table 4. From Table 4, it can be found that the RMSE, MAD and MAPE of the PSO-SVR model are, respectively, 49.10, 36.32 and 0.58%. It is evident that those values are smaller than those of the traditional SVR model. It therefore indicates that there is a smaller deviation between the actual and predicted values in the proposed PSO-SVR model. Moreover, compared to the traditional SVR models, the PSO-SVR model has the highest DS (directional symmetry) and CD (correct down trend) values which are 59.24% and 57.32%, respectively. DS, CU and CD provide good measures of the consistency in prediction of the price direction. Thus, it can be concluded that the proposed PSO-SVR model outperforms the traditional SVR model in both prediction error and prediction accuracy. Other than the forecasting capabilities, we can also observe that the training time of the proposed PSO-SVR model is much faster than that of the traditional SVR model. It therefore should be a better alternative as it will give us more time in designing proper strategies after obtaining the forecasting result on the top of its low prediction error and high prediction accuracy.

<Table 4 about here>

## 5. Conclusions

This paper proposed a financial time series forecasting model by integrating PSO and SVR. The TAIEX closing cash index is used in this study for evaluating the performance of the proposed method. This study compared the forecasting results of the proposed method with those of the traditional SVR method using prediction error and prediction accuracy as criteria. Compared to the traditional SVR model, experimental results showed that the proposed PSO-SVR model can produce lower prediction error and higher prediction accuracy in the datasets. Besides, the CPU time for the proposed model is about eight times faster than that when using the traditional SVR method. Therefore the PSO-SVR model should be indeed a better alternative than the traditional SVR model since it can save lots of modeling time on the top of its better forecasting capabilities.

## References

- Balachandher, K. G., Fauzias, M. N., & Lai, M. M. (2002). An Examination of the Random Walk Model and Technical Trading Rules in the Malaysian Stock Market. *Quarterly Journal of Business & Economics*, 41, 81-104.
- Bratton, D., & Kennedy, J. (2007). Defining a Standard for Particle Swarm Optimization. *Proceedings of the 2007 IEEE Swarm Intelligence Symposium*, 120-127.
- Cao, L. J. (2003). Support Vector Machines Experts for Time Series Forecasting. *Neurocomputing*, 51, 321-339, doi:10.1016/S0925-2312(02)00577-5, [http://dx.doi.org/10.1016/S0925-2312\(02\)00577-5](http://dx.doi.org/10.1016/S0925-2312(02)00577-5)
- Chang, C. C., & Lin, C. J. (2001). LIBSVM: A Library for Support Vector Machines, <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- Cherkassky, V., & Yunqian M. (2004). Practical Selection of SVM Parameters and Noise Estimation for SVM Regression. *Neural Networks*, 17, 113-126, doi: 10.1016/S0893-6080(03)00169-2, [http://dx.doi.org/10.1016/S0893-6080\(03\)00169-2](http://dx.doi.org/10.1016/S0893-6080(03)00169-2)
- Eberhart, R. C., & Shi, Y. (2001a). Tracking and Optimizing Dynamic Systems with Particle Swarms. *Proceedings of Congress on Evolutionary Computation*, 94-97, IEEE Service Center, New Jersey.
- Eberhart, R. C., & Shi, Y. (2001b). Particle Swarm Optimization: Developments, Applications and Resources. *Proceedings of Congress on Evolutionary Computation*, 81-86, IEEE Service Center, Seoul.
- Gunn, S. R. (1998). Support vector machines for classification and regression. *Technical report*, Faculty of Engineering, Science and Mathematics School of Electronics and Computer Science.
- Hsu, C. W., Chang, C. C., & Lin C. J. (2000). A Practical Guide to Support Vector Classification. *Technical report*, Taipei.
- Kennedy J., & Eberhart R. C. (1995). Particle Swarm Optimization. *Proceedings of IEEE International Conference on Neural Networks*, 1942-1948. IEEE Press, New Jersey.
- Kennedy, J., Eberhart, R. C., & Shi, Y. (2001). *Swarm intelligence*. San Francisco: Morgan Kaufmann.
- Kim, K. J. (2003). Financial Time Series Forecasting Using Support Vector Machines. *Neurocomputing*, 55, 307-319, doi:10.1016/S0925-2312(03)00372-2, [http://dx.doi.org/10.1016/S0925-2312\(03\)00372-2](http://dx.doi.org/10.1016/S0925-2312(03)00372-2)

- Lee, T. S., & Chen, N. J. (2002). Investigating the Information Content of Non-Cash-Trading Index Futures Using Neural Networks. *Expert Systems with Applications*, 22, 225-234, doi:10.1016/S0957-4174(01)00056-2, [http://dx.doi.org/10.1016/S0957-4174\(01\)00056-2](http://dx.doi.org/10.1016/S0957-4174(01)00056-2)
- Lee, T. S., & Chiu, C. C. (2002). Neural Network Forecasting of an Opening Cash Price Index. *International Journal of Systems Science*, 33, 229-237, doi:10.1080/00207720110092216, <http://dx.doi.org/10.1080/00207720110092216>
- Leigh, W., Hightower, R., & Modani, N. (2005). Forecasting the New York Stock Exchange Composite Index with Past Price and Interest Rate on Condition of Volume Spike. *Expert Systems with Applications*, 28, 1-8, doi:10.1016/j.eswa.2004.08.001, <http://dx.doi.org/10.1016/j.eswa.2004.08.001>
- Li, S., Kwok, J. T., Zhu, H., & Wang, Y. (2003). Texture Classification Using the Support Vector Machines. *Pattern Recognition*, 36, 2883-2893, doi:10.1016/S0031-3203(03)00219-X, [http://dx.doi.org/10.1016/S0031-3203\(03\)00219-X](http://dx.doi.org/10.1016/S0031-3203(03)00219-X)
- Müller, K. R., Smola, A. J., Rätsch, G., Schölkopf, B., Kohlmorgen, J., & Vapnik, V. (1999). Predicting Time Series with Support Vector Machines. In B. Schölkopf, C. J. C. Burges, & A. J. Smola (Eds.). *Advances in Kernel Methods—Support Vector Learning* (pp. 243-254). Massachusetts: MIT Press.
- Norinder, U. (2003). Support Vector Machine Models in Drug Design: Applications to Transport Processes and QSAR Using Simplex Optimisations and Variable Selection. *Neurocomputing*, 55, 337-346, doi:10.1016/S0925-2312(03)00374-6, [http://dx.doi.org/10.1016/S0925-2312\(03\)00374-6](http://dx.doi.org/10.1016/S0925-2312(03)00374-6)
- Park, Y., Kim, S. W., & Ahn, H. S. (2005). Support Vector Machine Parameter Tuning Using Dynamic Encoding Algorithm for Handwritten Digit Recognition. *The fifth International Conference on information, Communications and Signal Processing*, 361-366.
- 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, doi:10.1016/j.eswa.2004.08.009, <http://dx.doi.org/10.1016/j.eswa.2004.08.009>
- Smola, A. J., & Schölkopf, B. (1998). A Tutorial on Support Vector Regression. *Technical report*, Royal Holloway College, University of London, UK.
- Tay, F. E. H., & Cao, L. J. (2001). Application of Support Vector Machines in Financial Time Series Forecasting. *Omega*, 29, 309-317, doi:10.1016/S0305-0483(01)00026-3, [http://dx.doi.org/10.1016/S0305-0483\(01\)00026-3](http://dx.doi.org/10.1016/S0305-0483(01)00026-3)
- Tay, F. E. H., & Cao, L. J. (2003). Support Vector Machine with Adaptive Parameters in Financial Time Series Forecasting. *IEEE Transactions on Neural Networks*, 14, 1506-1518, doi:10.1109/TNN.2003.820556, <http://dx.doi.org/10.1109/TNN.2003.820556>
- Vapnik, V. N. (2000). *The Nature of Statistical Learning Theory*. (2nd ed.). New York: Springer.
- Vellido, A., Lisboa, P. J. G., & Vaughan, J. (1999). Neural Networks in Business: A Survey of Applications (1992-1998). *Expert Systems with Applications*, 17, 51-70, doi:10.1016/S0957-4174(99)00016-0, [http://dx.doi.org/10.1016/S0957-4174\(99\)00016-0](http://dx.doi.org/10.1016/S0957-4174(99)00016-0)
- Wood, S. (2002). *Float Analysis: Powerful Technical Indicators Using Price and Volume*, New York: John Wiley & Sons.
- Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with Artificial Neural Networks: The State of the Art. *International Journal of Forecasting*, 14, 35-62, doi:10.1016/S0169-2070(97)00044-7, [http://dx.doi.org/10.1016/S0169-2070\(97\)00044-7](http://dx.doi.org/10.1016/S0169-2070(97)00044-7)

Table 1. Performance metrics and their calculations

Metrics	Calculation	Metrics	Calculation
RMSE	$RMSE = \sqrt{\frac{\sum_{i=1}^N (T_i - A_i)^2}{N}}$	DS	$DS = \frac{100}{N} \times \sum_{i=1}^N d_i$ , where $d_i = \begin{cases} 1 & \text{if } (A_i - A_{i-1})(T_i - T_{i-1}) \geq 0 \\ 0 & \text{otherwise} \end{cases}$
MAD	$MAD = \frac{\sum_{i=1}^N  T_i - A_i }{N}$	CU	$CU = \frac{100}{N_1} \times \sum_{i=1}^N d_i$ , where $d_i = \begin{cases} 1 & \text{if } (T_i - T_{i-1}) > 0 \text{ and } (A_i - A_{i-1})(T_i - T_{i-1}) \geq 0 \\ 0 & \text{otherwise} \end{cases}$
MAPE	$MAPE = \frac{\sum_{i=1}^N \frac{ T_i - A_i }{T_i}}{N}$	CD	$CD = \frac{100}{N_2} \times \sum_{i=1}^N d_i$ , where $d_i = \begin{cases} 1 & \text{if } (T_i - T_{i-1}) < 0 \text{ and } (A_i - A_{i-1})(T_i - T_{i-1}) \geq 0 \\ 0 & \text{otherwise} \end{cases}$

Note that T and A represent the actual and predicted value, respectively, N is total number of data points, N<sub>1</sub> is number of data points belong to up trend and N<sub>2</sub> is number of data points belong to down trend.

Table 2. Model selection results of traditional SVR model (partial)

C	ε	γ	Testing RMSE
2 <sup>10</sup>	2 <sup>-1</sup>	2 <sup>-3</sup>	49.3396
2 <sup>10</sup>	2 <sup>1</sup>	2 <sup>-3</sup>	49.2717
2 <sup>10</sup>	2 <sup>3</sup>	2 <sup>-3</sup>	49.4277

Table 3. Model selection results of the proposed PSO-SVR model (partial)

Iteration	C	ε	γ	RMSE
20	1024	0.00098	1.7043	49.1070
27	1024	0.00098	1.7082	49.1040
28	1024	0.00098	1.7257	49.0968
29	1024	0.00098	1.7249	49.0968
49	1024	0.00098	1.7249	49.0968

Table 4. Forecasting results using traditional SVR and PSO-SVR models

Models	Metrics					
	RMSE	MAD	MAPE	DS	CU	CD
PSO-SVR model (164 seconds)	49.10	36.32	0.58%	59.24%	61.33%	57.32%
Traditional SVR model (1252 seconds)	49.27	36.67	0.59%	58.60%	62.67%	54.88%

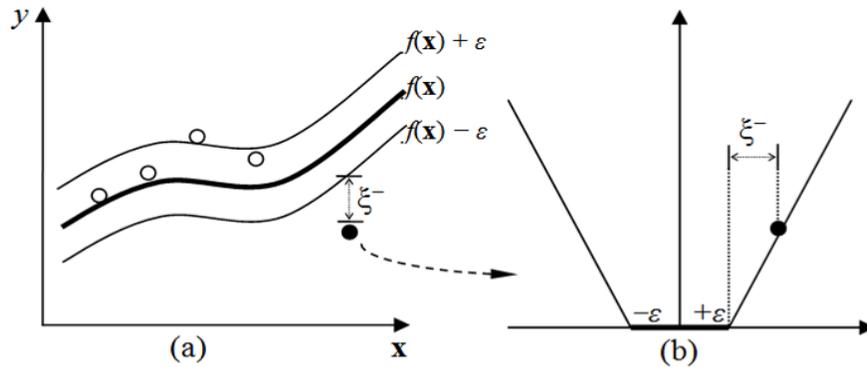


Figure 1. An SVR function using  $\epsilon$ -insensitive loss function

- (a) The tube of  $\epsilon$  accuracy is fitted to the data and points lying outside the tube are measured using slack variable  $\xi^+$ ,  $\xi^-$ .
- (b) The  $\epsilon$ -insensitive loss function in which the slope is determined by the parameter  $C$ .

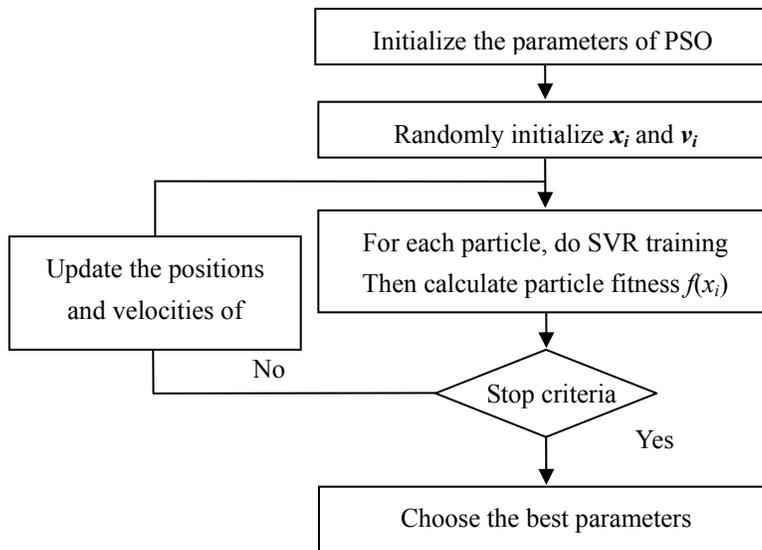


Figure 2. Flow diagram of PSO-SVR forecasting model

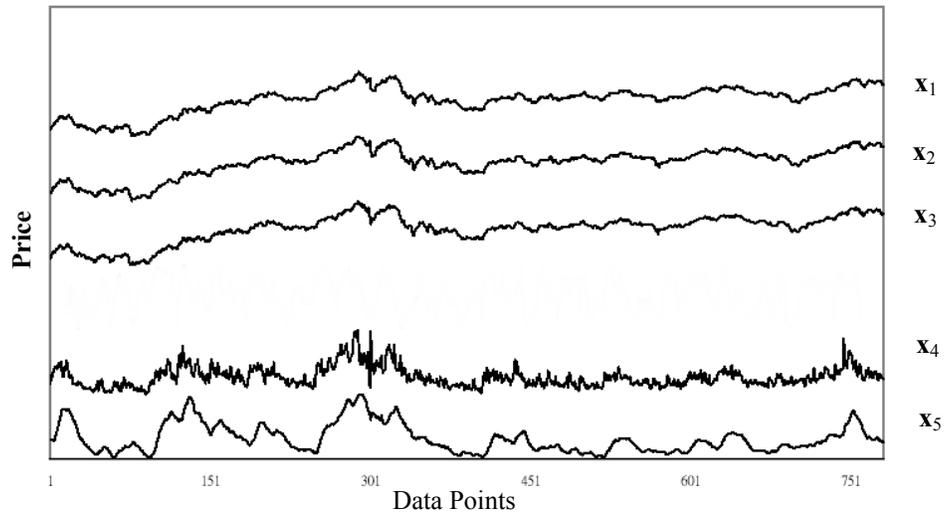


Figure 3. Five financial time series data

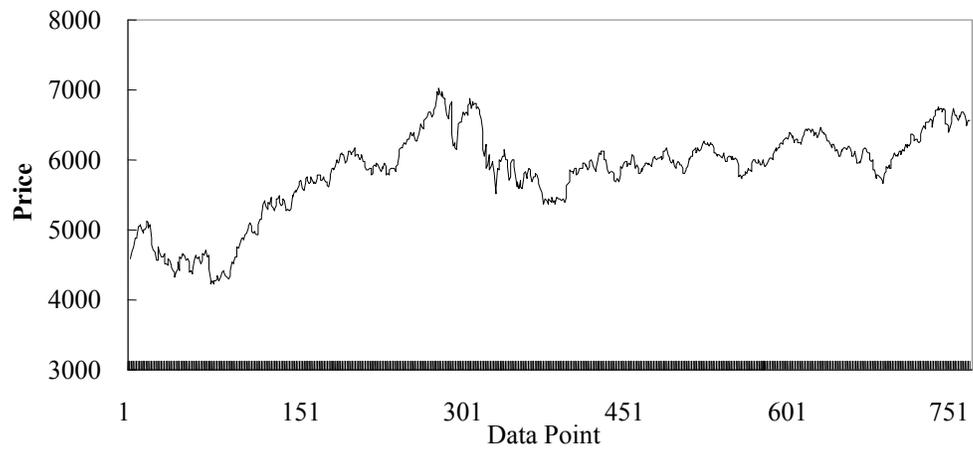


Figure 4. The daily TAIEX closing cash prices from January 2, 2003 to February 27, 2006