Mathematical and Computer Modelling 57 (2013) 932–944

Contents lists available at SciVerse ScienceDirect

Mathematical and Computer Modelling

journal homepage: www.elsevier.com/locate/mcm

Forecasting the movement direction of exchange rate with polynomial smooth support vector machine☆

Yubo Yuan

Department of Computer Science and Technology, School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, China

ARTICLE INFO

Article history:
Received 4 September 2011
Received in revised form 22 March 2012
Accepted 6 October 2012

Keywords:
Data mining
Machine learning
Support vector machines
Neural networks
Financial time series
Forecasting

ABSTRACT

It is a very interesting topic to forecast the movement direction of financial time series by machine learning methods. Among these machine learning methods, support vector machine (SVM) is the most effective and intelligent one. A new learning model is presented in this paper, called the polynomial smooth support vector machine (PSSVM). After being solved by Broyden–Fletcher–Goldfarb–Shanno (BFGS) method, optimal forecasting parameters are obtained. The exchange rate movement direction of RMB (Chinese renminbi) vs USD (United States Dollars) is investigated. Six indexes of Dow Jones China Index Series are used as the input. 4 sections with 180 time experiments have been completed. Many results show that the proposed learning model is effective and powerful.

1. Introduction

Financial time-series forecasting is very challenging work. There are many methods and studies used in this area, such as the exponential smoothing method [1], autoregressive integrated moving average (ARIMA) models [2], various models for seasonality [3], state space and structural models and the Kalman filter [4,5], nonlinear models (including regime-switching models, functional-coefficient model, artificial neural network, deterministic versus stochastic dynamics and so on) [6], long memory models [7], autoregressive conditional heteroscedastic (ARCH) models (a more parsimonious model than ARCH is also called the generalized ARCH (GARCH) model) [8] and some combining forecasts, mixing, or pooling quantitative forecasts obtained from very different financial time series methods [9].

Among them, artificial neural networks (ANN) [10,11] is one of the most popular intelligent methods. The main idea with ANN is that inputs, or dependent variables, get filtered through one or more hidden layers each of which consist of hidden units, or nodes, before they reach the output variable. The intermediate output is related to the final output. But it too depends on the training data sets. With the progress of machine learning, one of the statistic learning methods called support vector machine (SVM [12–17]) has appeared rapidly and has already surpassed neural networks by applications in practice. Many traditional neural network models had implemented the empirical risk minimization principle, SVM implements the structural risk minimization principle. The former seeks to minimize the mis-classification error or deviation from a correct solution of the training data but the latter searches to minimize an upper bound of generalization error. In addition, the solution of SVM may be global optimum while other neural network models may tend to fall into a local optimal solution. Thus, over-fitting is unlikely to occur with SVM. Nowadays, it is becoming, more and more, an active learning method.

In 2002 [18], Francis and Cao presented a modified version of support vector machine, and called it the C-ascending support vector machine. They showed that the proposed one was very powerful with non-stationary financial time series.

☆ The research was supported by the National Natural Science Foundations of China (Nos, 61001200, 61101239) and the Natural Science Foundation of Zhejiang Province of China (No. Y6100010).

E-mail address: ybyuanvt@gmail.com.

0895-7177/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mcm.2012.10.004

They used three real futures collected from the Chicago Mercantile Market to test. Experimental results showed that the C-ascending support vector machine with the actually ordered sample data consistently forecast better than the standard support vector machine. Next, in 2003 [19], Cao also proposed an SVM expert with tree-structured architecture for forecasting financial time series. Simulations showed that the SVM expert had achieved significant improvement in the generalization performance in comparison with the single SVM model. Kyoungjae [20] also applied SVM to predict the stock price index. The experimental results showed that SVM provided a promising alternative to stock market prediction by comparing it with back-propagation neural networks.

In this paper, we propose a polynomial smooth support vector machine to forecast the movement direction of financial time series. In Section 2, a brief view of SVM theory and the basic mathematical model of the polynomial smooth support vector machine are presented. In Section 3, we present a performance comparison and theory analysis of the proposed model. The BFGS algorithm is employed to solve PSSVM in Section 4. In Section 5, an application for using the Dow Jones China Index Series to predict the RMB exchange rate movement direction is investigated. In Section 6, we conclude this work.

2. A brief overview of SVM for financial time series movement direction forecasting

In order to understand this paper easily, we introduce some notations used in the following sections.

- m: the number of days that the financial time series are investigated;
- n: the number of impactors of the movement direction;
- A: the database comes from financial time series which is represented by an $m \times n$ matrix;
- D: the movement direction matrix is a diagonal one whose diagonals are 1 or -1;
- ω: the decision vector that is used to determine the hyper-plane to forecast the movement direction;
- γ: a real number that is used as a forecasting parameter, mathematically, is the main control of the distance between the decision hyperplane and the origin;
- e: a vector whose elements are 1;
- x: the input vector that is the known-data used to forecast the movement direction.

If the financial time series database is given by an $m \times n$ matrix A, according to the membership of each point A_i in the classes 1 or -1 (example movement direction of a target financial time series) as specified by a given $m \times m$ diagonal matrix D with 1 or -1 diagonals, called the direction index matrix.

Let us define the linear separating hyper-surface as follows

$$P = \{x | x \in R^n, x^T\omega = \gamma\}. \tag{1}$$

It can be used to separate the R^n into two parts (left and right, or up and down, or front and back).

If we define the decision function as follows

$$f(x) = \begin{cases} +1 & \text{if } x^T\omega \geqslant \gamma, \\ -1 & \text{if } x^T\omega < \gamma. \end{cases} \tag{2}$$

It can be used to determine the location of the input point x. Example, $f(x) = 1$ denotes that the point is in the right subspace, $f(x) = -1$ denotes that the point is in the left one.

For a given database A and known locations, we want to look for a best hyper-surface to classify them. The standard support vector machine originated from this idea.

In order to understand this idea, Fig. 1 indicates the meaning of margin, support vector, hyper-surface and support hyper-surface.

The standard support vector machine (see [12–17]) for this problem is given by the following

$$\min_{(\omega,\gamma)\in R^{n+1}} \frac{1}{2}\omega^T\omega,$$
$$\text{s.t. } D(A\omega - e\gamma) \geq e.$$

In other words, the object of SVM is to find the optimal solution ω and γ from the above mathematical model.

Till now, the proposed methods are classified into two classes. One is the dual method by Lagrange multiplier. The other is to solve the optimal mathematical model directly. The first method is the basic and original one. Another one is to solve the model straightforwardly, there are a few published works. In this work, we focus on the latter.

Introducing a slack vector $y \in R^n$, we can reformulate the standard support vector machine as follows

$$\begin{aligned} &\min_{(\omega,\gamma,y)\in R^{n+1+m}} \nu e^T y + \frac{1}{2}\omega^T\omega, \\ &\text{s.t. } D(A\omega - e\gamma) + y \geq e, \\ &y \geq 0. \end{aligned} \tag{3}$$

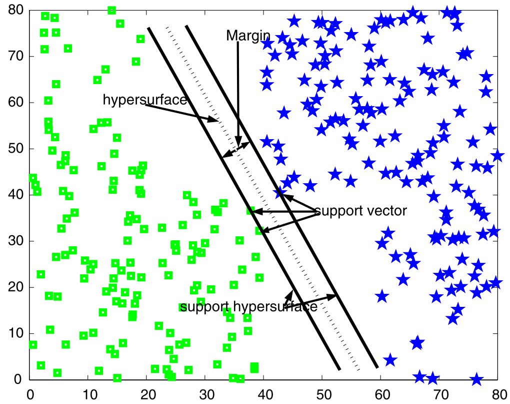

Fig. 1. The example graph of support vector machine classifier.

The first term in the objective function of (3) is the 1-norm of the slack variable y with positive weight $\nu \in R$. The second term $\omega^T\omega$ is the square of the 2-norm of the vector ω. Replacing the first term with the 2-norm vector y, the SVM problem can be modified into the following form

$$\begin{aligned} &\min_{(\omega,\gamma,y)\in R^{n+1+m}} \frac{\nu}{2}\|y\|_2^2 + \frac{1}{2}(\|\omega\|_2^2 + \gamma^2), \\ &\text{s.t. } D(A\omega - e\gamma) + y \geq e, \\ &y \geq 0. \end{aligned} \tag{4}$$

As a feasible solution of problem (4), y is given by

$$y = (e - (D(A\omega - e\gamma)))_+, \tag{5}$$

where the element of the vector $(a)_+$ is defined by

$$(a_i)_+ = \begin{cases} a_i, & \text{if } a_i > 0, \\ 0, & \text{if } a_i \leq 0. \end{cases} \tag{6}$$

Substituting y into the objective function of (4) converts problem (4) into an equivalent unconstrained optimization problem

$$\min_{(\omega,\gamma)\in R^{n+1}} \frac{\nu}{2}\|(e - (D(A\omega - e\gamma)))_+\|_2^2 + \frac{1}{2}(\|\omega\|_2^2 + \gamma^2). \tag{7}$$

This is a strongly convex minimization problem without any constraints and it has a unique solution.

Lee (see [21,22]) applied the smoothing techniques and reformulated (7) as a smooth SVM. They used the signal integral function as follows

$$p(x, k) = x + \frac{1}{k}\log(1 + \varepsilon^{kx}), \quad k > 0. \tag{8}$$

In this paper, we propose a polynomial function as follows

$$h(x, k) = \begin{cases} x, & \text{if } x > \dfrac{1}{k}, \\ -\dfrac{k^3}{16}\left(x + \dfrac{1}{k}\right)^3\left(x - \dfrac{3}{k}\right), & \text{if } -\dfrac{1}{k} \leq x \leq \dfrac{1}{k}, \\ 0, & \text{if } x < -\dfrac{1}{k}. \end{cases} \tag{9}$$

It is a fourth-order polynomial function. It is different with a plus function only on a small section $\left[-\frac{1}{k}, \frac{1}{k}\right]$. Moreover, it has a quadratic smooth performance on the interval $(-\infty, +\infty)$.

Remark 1. The motivation of the selection of polynomial function (7) is that, mathematically, the objective function in (7) is not differentiable at zero. It makes that many precious optimization algorithms cannot be used to solve the model (7). The basic condition of the algorithms is that the objection function is differentiable. In order to employ the algorithms to solve the model (7), we select the polynomial function (9) to approximate the plus function in (7).

If we replace the plus function in (7) by (9), a new smooth SVM model is obtained, called a polynomial smooth support vector machine (PSSVM). It is presented as follows

$$\min_{(\omega,\gamma)\in R^{n+1}} F(\omega,\gamma,k) = \frac{\nu}{2}\|h(e-(D(A\omega-e\gamma)),k)\|_2^2 + \frac{1}{2}(\|\omega\|_2^2+\gamma^2). \tag{10}$$

Its objective function is quadratic differential and convex. Many optimal algorithms can be used to solve it, such as the Newton method, the Quasi-Newton method and so on. In this paper, we propose the BFGS method.

3. Theory analysis of polynomial function

First, let us analyze the differential property of the polynomial. It is well known that the differential property of an objective function is very important because of the need of algorithm convergence.

Theorem 3.1. *If the approximation function has the formulation as (9), $h(x,k)$ is continuous and quadratical differentiable for any given $k \in Z^+$.*

Proof. Easily, since

$$\lim_{x\to+\frac{1}{k}} h(x,k) = \frac{1}{k}, \qquad \lim_{x\to-\frac{1}{k}} h(x,k) = 0,$$

and

$$\left(\frac{d(h(x,k))}{dx}\right)\Big|_{x=-\frac{1}{k}} = 0, \qquad \left(\frac{d(h(x,k))}{dx}\right)\Big|_{x=\frac{1}{k}} = 1.$$

Moreover, $h(x,k)$ satisfies

$$\left(\frac{d^2(h(x,k))}{dx^2}\right)\Big|_{x=-\frac{1}{k}} = 0, \qquad \left(\frac{d^2(h(x,k))}{dx^2}\right)\Big|_{x=\frac{1}{k}} = 0.$$

These computation results show that it is continuous and twice-order differentiable. This is the end of the proof. □

Theorem 3.2. *For $x \in R$, the forth polynomial function satisfies*

$$h(x,k)^2 - x_+^2 \le \frac{1}{19k^2}. \tag{11}$$

Proof. From the definition (8), the result $h(x,k) \geqslant x_+$ is obvious. In $\left(-\infty,-\frac{1}{k}\right)$ and $\left(\frac{1}{k},\infty\right)$, the left side of (11) is equal to zero, so the inequality (11) holds.

For $x \in \left(-\frac{1}{k},0\right)$, since $-\frac{k^3}{16}\left(x+\frac{1}{k}\right)^3\left(x-\frac{3}{k}\right)$ are increasing functions, the forth polynomial function satisfies

$$h(x,k)^2 - x_+^2 \le a(0,k)^2 = \frac{9}{256k^2} \le \frac{1}{19k^2}.$$

For $x \in \left(0,\frac{1}{k}\right)$, we use the method of finding a maximized functional value and proof that the forth polynomial function satisfies

$$\max_{x\in\left(0,\frac{1}{k}\right)} (h(x,k)^2 - x_+^2) \le \frac{1}{19k^2}.$$

This is the end of the theorem. □

The result of Theorem 3.2 shows that the difference of the plus function and forth order polynomial function is very small. For example, if we select $k = 10$, the maximal difference between the two functions is $\frac{1}{1900} \approx 0.00052632$. They are very close. In other words, it is very accurate to use a fourth-order polynomial function to smooth it.

Let us compare it with the signal function integral. It is the same with the next paragraph, if we select $k = 10$, we draw a picture of them as follows (in Fig. 2). In fact, the sigmoid function integral [22, Lemma 2.1], satisfies

$$p(x,k)^2 - x_+^2 \le ((\log 2)^2 + 2\log 2)\frac{1}{k^2} \approx 0.6927\frac{1}{k^2};$$

The fourth-order polynomial function satisfies

$$(h(x,k)^2 - x_+^2) \le 0.0526\frac{1}{k^2}.$$

Obviously, it is more effective than the formal one. Now, we will design an algorithm to solve the smooth model (10) in the next section.

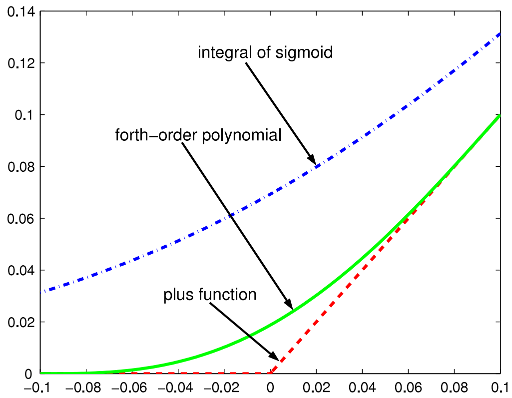

Fig. 2. The smooth performance comparison graph.

4. BFGS algorithm to solve PSSVM

BFGS methods are suitable for unconstrained optimization with function and gradient value evaluation available, and it is well known that the BFGS method is the most widely used one among various quasi-Newton methods.

BFGS algorithm

Step 1: Given the control factor of algorithm accuracy ϵ and the initial smooth parameter $k, H^0 = I, (\omega^0, \gamma^0) = p^0 \in R^{n+1}, \varepsilon = 10^{-8}$ and set $i := 0$;

Step 2: Evaluate $F^i = F(p^i, k)$ and $g^i = \nabla F(p^i, k)$;

Step 3: If $\|g^i\|_2^2 \le \varepsilon$, then stop, and accept $p^i = (\omega^i, \gamma^i)$ as the optimal solution of (9), else calculate $d^i = -H^i g^i$;

Step 4: Do a line search along direction d^i to get a step length $\alpha^i > 0$; let

$$p^{i+1} = p^i + \alpha^i d^i, \qquad s^i = p^{i+1} - p^i = -\alpha^i H^i g^i,$$

and evaluate $F^{i+1}(k_j) = F(p^{i+1}, k_j), g^{i+1} = \nabla F(p^{i+1}, k_j)$ and $y^i = g^{i+1} - g^i$;

Step 5: Update H^i to get H^{i+1}:

$$H^{i+1} = \left(I - \frac{s^i (y^i)^T}{(s^i)^T y^i}\right) H^i \left(I - \frac{y^i (s^i)^T}{(s^i)^T y^i}\right) + \frac{s^i (s^i)^T}{(s^i)^T y^i}; \tag{12}$$

Step 6: Set $i := i + 1$, go to step 3;

The proof of algorithm convergence and many detail knowledge can be seen in Refs. [24,23].

5. Application

In Section 4, we show an algorithm that can be used to get the forecasting parameters ω and γ. In this section, we give an application to show the effectiveness of the proposed model.

5.1. Source data

The first step is to prepare the database A. After we observed the exchange movement direction of RMB vs USD, with the Dow Jones China Index Series, we find that the exchange movement direction of RMB vs USD is strongly related with the Dow Jones China Index Series.

We decided that the Dow Jones China Index Series should be used to get the database A.

5.2. Brief introduction to Dow Jones China indexes

Dow Jones Indexes, a leading global index provider, announced changes in the composition of the Dow Jones China Index Series (Dow–China Indexes). Component changes in the Dow Jones China Offshore 50, Dow Jones China 88 (Dow–China 88), Dow Jones Shanghai (Dow–Shanghai), Dow Jones Shenzhen (Dow–Shenzhen), Dow Jones CBN China 600 and Dow Jones China Total Market (Dow–China Total Market) indexes. The Dow Jones China Offshore 50 Index (DJSC50) intends to represent the largest stocks of companies whose primary operations are in mainland China but that trade on the exchanges of Hong Kong and the US. The Dow-China 88 Index (DJSC88) is made up of the 88 largest and most liquid stocks in the Dow–China Total Market Index. The two primary criteria used in the component selection process are market capitalization and trading

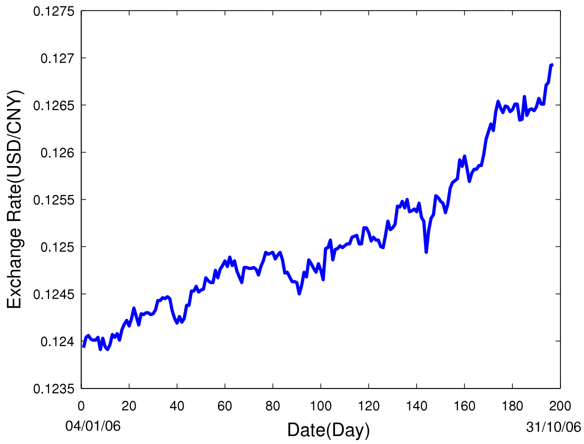

Fig. 3. The changing figure of the exchange rate of RMB vs USD from January 4, 2006.

liquidity, as measured by average daily turnover. Buffers are in place to ensure that the Dow–China 88's component turnover rate remains low. Dow Jones Shanghai Index (DJSSH) and Dow Jones Shenzhen Index (DJSSZ) components are drawn from the universe of all stocks traded on the Shanghai Stock Exchange and the Shenzhen Stock Exchange, respectively. Each index is designed to cover 95% of the overall market capitalization. Components are selected based on float-adjusted market capitalization and liquidity. In order to achieve a diversified representation across economic sectors and industry groups, sector allocation is also considered in the component selection process. The Dow Jones CBN China 600 Index (DJSC600) is a broad yet investable measure of securities available to domestic Chinese investors. The index represents the 600 largest A-shares traded on the Shanghai and Shenzhen stock exchanges as ranked by float-adjusted market capitalization. It was introduced in conjunction with China Business Network (CBN), a Shanghai-based media firm. Dow Jones China Total Market Index (DJSCHINA) reflects approximately 95% of the free-float market capitalization for both the Shanghai and Shenzhen markets.

5.3. Data processing and experiments

5.3.1. Procedure to get the database A

We selected the upper six indexes to set up the database A. The research historical data is downloaded from http://chinaindex.dowjones.com/. Their closed data can be seen in Figs. 4–9 from January 4, 2006 to October 31, 2006. The historical data of the exchange rate of RMB vs USD is obtained from the Pacific Exchange Rate Service provided by Professor Werner Antweiler, University of British Columbia, Vancouver, Canada, respectively. The whole data set covers the period from January 4, 2006 to October 31, 2006, a total of 192 pairs of observations (these can be seen in Fig. 3). Form the beginning of 2006, the exchange rate of RMB (Chinese renminbi) vs USD (United States Dollar) climbs very fast. We expect to have a good performance when the Dow Jones China Index Series are employed to predict the movement direction of the exchange rate of RMB vs USD. The change of exchange rate of RMB vs USD is shown in Fig. 3.

For computation effectiveness, six indexes need to be normalized as following

$$A_t^j = \frac{A_t^j}{\max\{A_1^j, A_2^j, \ldots, A_{191}^j\}}, \quad t = 1, 2, \ldots, 191, j = 1, 2, \ldots, 6$$

where A_t^j denote the t-th close day value with the j-th DJS index.

The data set is divided into two parts with four different experiment sections. One part is the training database denoted as *ATR*, the other is the testing database denoted as *ATE*.

Training Database *ATR*			Testing Database *ATE*
Section	Number of experiment	Range of days	Range of days
Section 1	10	50–140	141–191
Section 2	20	45–140	141–191
Section 3	50	42–140	141–191
Section 4	100	41–140	141–191

Remark 2. In the experiment section 1, we completed 10 experiments and the training data set number is changed from 50 to 140 every 10 days. Experiment section 2 is from 45 to 140 every 5 days and 20 experiments are completed. Experiment

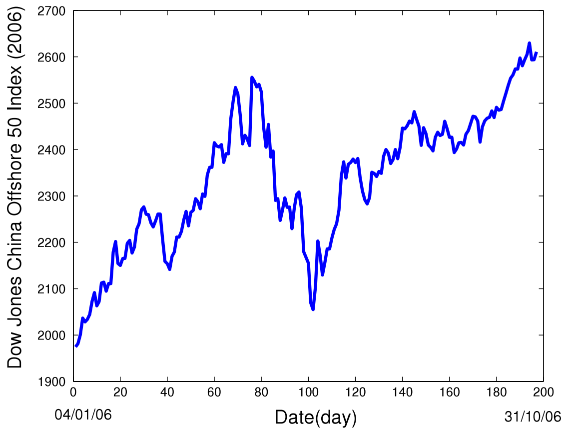

Fig. 4. The changing figure of DJSC50.

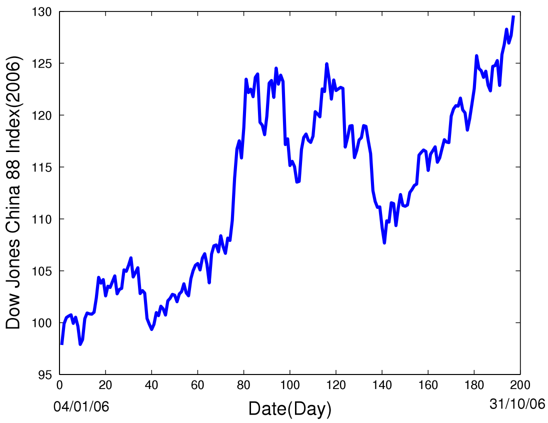

Fig. 5. The changing figure of DJSC88.

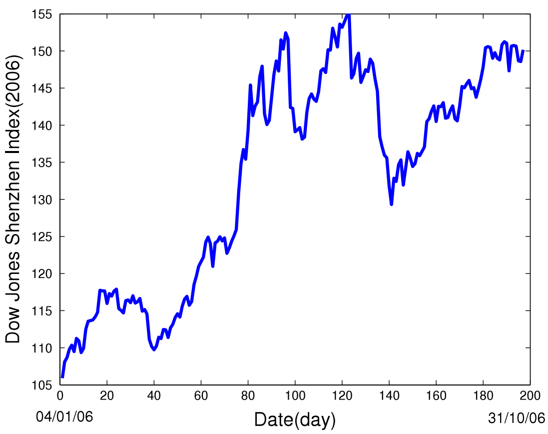

Fig. 6. The changing figure of DJSSZ.

section 3 is from 42 to 140 every 2 days and 50 experiments are completed. Experiment section 4 is from 41 to 140 every day and 100 experiments are completed. The test data set is from 141 to 191.

After doing this work, we get the database A and the corresponding training database. Next, we will show how to get the movement direction matrix D.

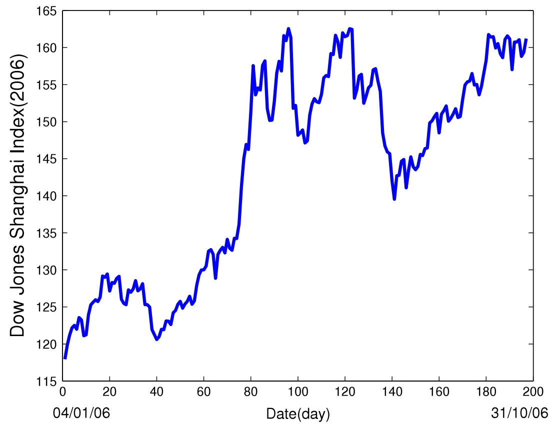

Fig. 7. The changing figure of DJSSH.

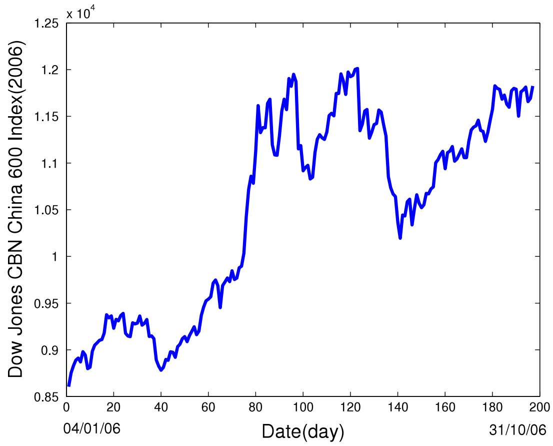

Fig. 8. The changing figure of DJSC600.

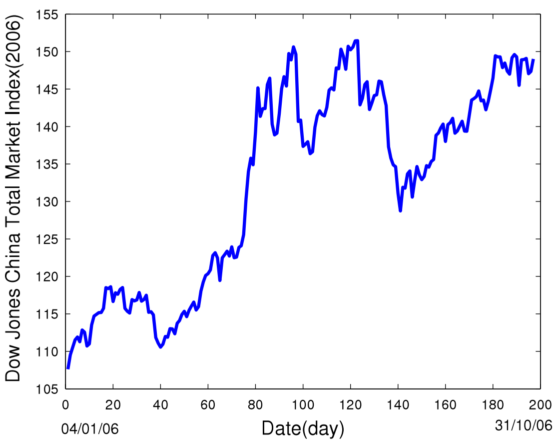

Fig. 9. The changing figure of DJSCHINA.

5.3.2. Procedure to get the movement direction matrix D

The exchange rate data of RMB vs USD was given by Werner Antweiler (Professor of University of British Columbia, Vancouver BC, Canada). The reader can download it from http://fx.sauder.ubc.ca/data.html. From January 4, 2006 to October 31, 2006, we can get the exchange rate vector $r = (r_1, r_2, \ldots, r_{192})^T$.

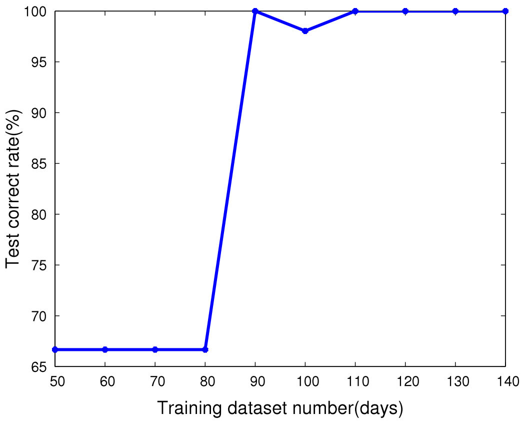

Fig. 10. The correct movement prediction rate of experiment section 1 (10 experiments).

The movement direction matrix is generated by the following procedure. The matrix $D = (d_{ij})_{i,j=1}^{191}$. The element d_{ij} is calculated with

$$d_{ij} = \begin{cases} +1, & \text{if } r_{i+1} - r_i \geqslant 0, i = j; \\ -1, & \text{if } r_{i+1} - r_i \geqslant 0, i = j; \\ 0, & i \neq j. \end{cases} \quad i, j = 1, 2, \ldots, 191.$$

Where r_i is the t-th day close value of exchange rate.

Also, the movement direction matrix is divided into two parts, the training part and testing one corresponding with Remark 2. We denote them as *DTR* and *DTE*.

5.3.3. *Basic procedure to forecast movement direction*

We denote the algorithm proposed in Section 4 as *BGP*.

The basic forecasting procedure of this paper is as follows.

Procedure for forecasting movement direction
S1: Initialize the Database A and D according to 5.3.1 and 5.3.2 S2: Get input: Training database *ATR* and Testing Database *ATE*; Training movement direction matrix *DTR* and Testing one *DTE*; S3: Employee the algorithm *BGP* in Section 4; S4: Get the optimal forecasting parameters ω and γ; S5: Employee the decision function (2) with Testing Database *ATE*; S6: Compare the results with the *DTE*;

5.4. *Results*

The correct test movement direction forecasting rate is computed by the following formulation

$$c = \frac{n}{tn} \times 100\%.$$

Where c denotes the correct test rate, n is the correct prediction time and tn is total number of test data points.

The four section experiment for correct movement prediction test rates are shown in Figs. 10–13.

Remark 3. An explanation about the four figures is listed.

(i) Fig. 10: The correct movement prediction rate of experiment section 1 (10 experiments).
(ii) Fig. 11: The correct movement prediction rate of experiment section 2 (20 experiments).
(iii) Fig. 12: The correct movement prediction rate of experiment section 3 (50 experiments).
(iv) Fig. 13: The correct movement prediction rate of experiment section 4 (100 experiments).

From these four sections of experiment results, if the number of training data is beyond 118 the correct movement prediction rate is 100%. It is a very nice and exciting result. This result is good and beyond the authors' predictions. The effective of PSSVM is proved.

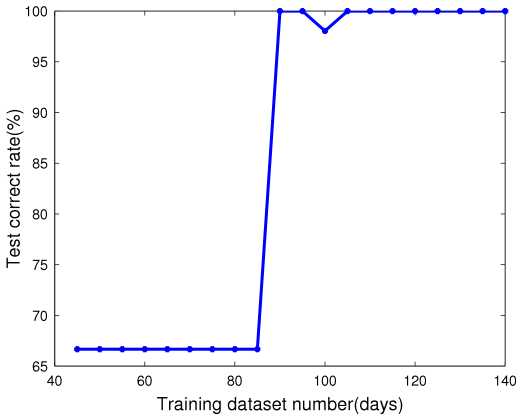

Fig. 11. The correct movement prediction rate of experiment section 2 (20 experiments).

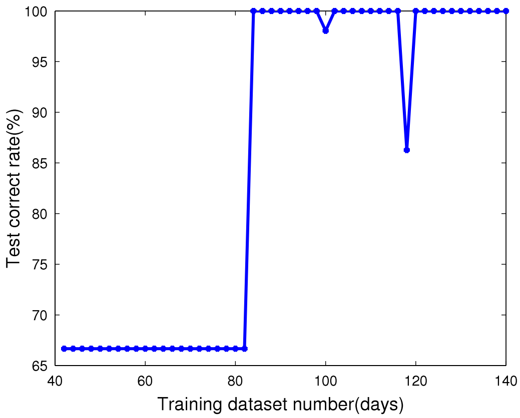

Fig. 12. The correct movement prediction rate of experiment section 3 (50 experiments).

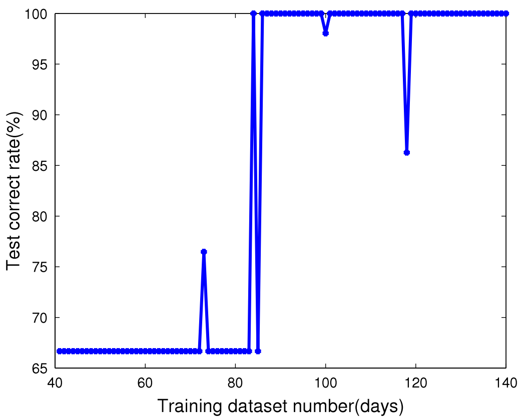

Fig. 13. The correct movement prediction rate of experiment section 4 (100 experiments).

6. Conclusions

In this paper, we have presented a new machine learning method to forecast the movement direction of financial time series, called the polynomial smooth support vector machine. Experimental results show that the proposed method is

Table 1
PACIFIC exchange rate service.

Jul.Day	YYYY/MM/DD	Wdy	USD/CNY	Jul.Day	YYYY/MM/DD	Wdy	USD/CNY
2453739	2006/01/03	Tue	0.12393	2453740	2006/01/04	Wed	0.12393
2453741	2006/01/05	Thu	0.12404	2453742	2006/01/06	Fri	0.12406
2453745	2006/01/09	Mon	0.12402	2453746	2006/01/10	Tue	0.12401
2453747	2006/01/11	Wed	0.12401	2453748	2006/01/12	Thu	0.12404
2453749	2006/01/13	Fri	0.12391	2453752	2006/01/16	Mon	0.12403
2453753	2006/01/17	Tue	0.12394	2453754	2006/01/18	Wed	0.12391
2453755	2006/01/19	Thu	0.12396	2453756	2006/01/20	Fri	0.12407
2453759	2006/01/23	Mon	0.12404	2453760	2006/01/24	Tue	0.12408
2453761	2006/01/25	Wed	0.12401	2453762	2006/01/26	Thu	0.12405
2453763	2006/01/27	Fri	0.12402	2453766	2006/01/30	Mon	0.12401
2453767	2006/01/31	Tue	0.12405	2453768	2006/02/01	Wed	0.12401
2453769	2006/02/02	Thu	0.12404	2453770	2006/02/03	Fri	0.12405
2453773	2006/02/06	Mon	0.12412	2453774	2006/02/07	Tue	0.12418
2453775	2006/02/08	Wed	0.12422	2453776	2006/02/09	Thu	0.12416
2453777	2006/02/10	Fri	0.12424	2453780	2006/02/13	Mon	0.12435
2453781	2006/02/14	Tue	0.12426	2453782	2006/02/15	Wed	0.12417
2453783	2006/02/16	Thu	0.12429	2453784	2006/02/17	Fri	0.12428
2453787	2006/02/20	Mon	0.12430	2453788	2006/02/21	Tue	0.12430
2453789	2006/02/22	Wed	0.12428	2453790	2006/02/23	Thu	0.12429
2453791	2006/02/24	Fri	0.12433	2453794	2006/02/27	Mon	0.12443
2453795	2006/02/28	Tue	0.12443	2453796	2006/03/01	Wed	0.12446
2453797	2006/03/02	Thu	0.12445	2453798	2006/03/03	Fri	0.12447
2453801	2006/03/06	Mon	0.12445	2453802	2006/03/07	Tue	0.12432
2453803	2006/03/08	Wed	0.12424	2453804	2006/03/09	Thu	0.12419
2453805	2006/03/10	Fri	0.12426	2453808	2006/03/13	Mon	0.12420
2453809	2006/03/14	Tue	0.12424	2453810	2006/03/15	Wed	0.12438

Table 2
PACIFIC exchange rate service.

2453811	2006/03/16	Thu	0.12438	2453812	2006/03/17	Fri	0.12453
2453815	2006/03/20	Mon	0.12453	2453816	2006/03/21	Tue	0.12458
2453817	2006/03/22	Wed	0.12452	2453818	2006/03/23	Thu	0.12454
2453819	2006/03/24	Fri	0.12455	2453822	2006/03/27	Mon	0.12467
2453823	2006/03/28	Tue	0.12464	2453824	2006/03/29	Wed	0.12462
2453825	2006/03/30	Thu	0.12462	2453826	2006/03/31	Fri	0.12475
2453829	2006/04/03	Mon	0.12467	2453830	2006/04/04	Tue	0.12476
2453831	2006/04/05	Wed	0.12480	2453832	2006/04/06	Thu	0.12485
2453833	2006/04/07	Fri	0.12479	2453836	2006/04/10	Mon	0.12489
2453837	2006/04/11	Tue	0.12480	2453838	2006/04/12	Wed	0.12485
2453839	2006/04/13	Thu	0.12474	2453843	2006/04/17	Mon	0.12468
2453844	2006/04/18	Tue	0.12462	2453845	2006/04/19	Wed	0.12478
2453846	2006/04/20	Thu	0.12478	2453847	2006/04/21	Fri	0.12477
2453850	2006/04/24	Mon	0.12477	2453851	2006/04/25	Tue	0.12478
2453852	2006/04/26	Wed	0.12476	2453853	2006/04/27	Thu	0.12470
2453854	2006/04/28	Fri	0.12479	2453857	2006/05/01	Mon	0.12476
2453858	2006/05/02	Tue	0.12476	2453859	2006/05/03	Wed	0.12482
2453860	2006/05/04	Thu	0.12477	2453861	2006/05/05	Fri	0.12482
2453864	2006/05/08	Mon	0.12485	2453865	2006/05/09	Tue	0.12494
2453866	2006/05/10	Wed	0.12492	2453867	2006/05/11	Thu	0.12493
2453868	2006/05/12	Fri	0.12494	2453871	2006/05/15	Mon	0.12487
2453872	2006/05/16	Tue	0.12491	2453873	2006/05/17	Wed	0.12494
2453874	2006/05/18	Thu	0.12486	2453875	2006/05/19	Fri	0.12472
2453879	2006/05/23	Tue	0.12473	2453880	2006/05/24	Wed	0.12468
2453881	2006/05/25	Thu	0.12463	2453882	2006/05/26	Fri	0.12463
2453885	2006/05/29	Mon	0.12462	2453886	2006/05/30	Tue	0.12450
2453887	2006/05/31	Wed	0.12459	2453888	2006/06/01	Thu	0.12473

very effective. Results in this paper are only based on a theoretic macroeconomic analysis. In practice, the exchange rate is impacted by various facts, such as wars, earthquakes, influence disease and many other political reasons. They will cause a great change in the exchange rate. Results shown in this work are correct only under a pure hypothesis. But, the result can indicate that the machine learning methods are very important for forecasting research and the polynomial smooth support vector machine is a very powerful model. See Tables 1–6.

Table 3
PACIFIC exchange rate service.

Jul.Day	YYYY/MM/DD	Wdy	USD/CNY	Jul.Day	YYYY/MM/DD	Wdy	USD/CNY
2453889	2006/06/02	Fri	0.12468	2453892	2006/06/05	Mon	0.12486
2453893	2006/06/06	Tue	0.12482	2453894	2006/06/07	Wed	0.12477
2453895	2006/06/08	Thu	0.12473	2453896	2006/06/09	Fri	0.12482
2453899	2006/06/12	Mon	0.12475	2453900	2006/06/13	Tue	0.12465
2453901	2006/06/14	Wed	0.12498	2453902	2006/06/15	Thu	0.12499
2453903	2006/06/16	Fri	0.12507	2453906	2006/06/19	Mon	0.12486
2453907	2006/06/20	Tue	0.12497	2453908	2006/06/21	Wed	0.12498
2453909	2006/06/22	Thu	0.12501	2453910	2006/06/23	Fri	0.12499
2453913	2006/06/26	Mon	0.12501	2453914	2006/06/27	Tue	0.12503
2453915	2006/06/28	Wed	0.12503	2453916	2006/06/29	Thu	0.12510
2453917	2006/06/30	Fri	0.12511	2453921	2006/07/04	Tue	0.12512
2453922	2006/07/05	Wed	0.12503	2453923	2006/07/06	Thu	0.12503
2453924	2006/07/07	Fri	0.12520	2453927	2006/07/10	Mon	0.12520
2453928	2006/07/11	Tue	0.12515	2453929	2006/07/12	Wed	0.12506
2453930	2006/07/13	Thu	0.12510	2453931	2006/07/14	Fri	0.12507

Table 4
PACIFIC exchange rate service.

Jul.Day	YYYY/MM/DD	Wdy	USD/CNY	Jul.Day	YYYY/MM/DD	Wdy	USD/CNY
2453934	2006/07/17	Mon	0.12507	2453935	2006/07/18	Tue	0.12500
2453936	2006/07/19	Wed	0.12499	2453937	2006/07/20	Thu	0.12512
2453938	2006/07/21	Fri	0.12527	2453941	2006/07/24	Mon	0.12518
2453942	2006/07/25	Tue	0.12520	2453943	2006/07/26	Wed	0.12524
2453944	2006/07/27	Thu	0.12543	2453945	2006/07/28	Fri	0.12542
2453948	2006/07/31	Mon	0.12548	2453949	2006/08/01	Tue	0.12541
2453950	2006/08/02	Wed	0.12550	2453951	2006/08/03	Thu	0.12537
2453952	2006/08/04	Fri	0.12538	2453956	2006/08/08	Tue	0.12540
2453957	2006/08/09	Wed	0.12537	2453958	2006/08/10	Thu	0.12546
2453959	2006/08/11	Fri	0.12531	2453962	2006/08/14	Mon	0.12527
2453963	2006/08/15	Tue	0.12494	2453964	2006/08/16	Wed	0.12517
2453965	2006/08/17	Thu	0.12530	2453966	2006/08/18	Fri	0.12534
2453969	2006/08/21	Mon	0.12554	2453970	2006/08/22	Tue	0.12552
2453971	2006/08/23	Wed	0.12548	2453972	2006/08/24	Thu	0.12546
2453973	2006/08/25	Fri	0.12536	2453976	2006/08/28	Mon	0.12545
2453977	2006/08/29	Tue	0.12562	2453978	2006/08/30	Wed	0.12568
2453979	2006/08/31	Thu	0.12570	2453980	2006/09/01	Fri	0.12572
2453984	2006/09/05	Tue	0.12592	2453985	2006/09/06	Wed	0.12585
2453986	2006/09/07	Thu	0.12596	2453987	2006/09/08	Fri	0.12583

Table 5
PACIFIC exchange rate service.

Jul.Day	YYYY/MM/DD	Wdy	USD/CNY	Jul.Day	YYYY/MM/DD	Wdy	USD/CNY
2453990	2006/09/11	Mon	0.12569	2453991	2006/09/12	Tue	0.12578
2453992	2006/09/13	Wed	0.12582	2453993	2006/09/14	Thu	0.12582
2453994	2006/09/15	Fri	0.12586	2453997	2006/09/18	Mon	0.12586
2453998	2006/09/19	Tue	0.12598	2453999	2006/09/20	Wed	0.12614
2454000	2006/09/21	Thu	0.12622	2454001	2006/09/22	Fri	0.12630
2454004	2006/09/25	Mon	0.12623	2454005	2006/09/26	Tue	0.12643
2454006	2006/09/27	Wed	0.12654	2454007	2006/09/28	Thu	0.12647
2454008	2006/09/29	Fri	0.12642	2454011	2006/10/02	Mon	0.12649
2454012	2006/10/03	Tue	0.12648	2454013	2006/10/04	Wed	0.12643
2454014	2006/10/05	Thu	0.12645	2454015	2006/10/06	Fri	0.12651
2454019	2006/10/10	Tue	0.12651	2454020	2006/10/11	Wed	0.12634
2454021	2006/10/12	Thu	0.12635	2454022	2006/10/13	Fri	0.12659
2454025	2006/10/16	Mon	0.12639	2454026	2006/10/17	Tue	0.12645
2454027	2006/10/18	Wed	0.12646	2454028	2006/10/19	Thu	0.12644
2454029	2006/10/20	Fri	0.12648	2454032	2006/10/23	Mon	0.12657
2454033	2006/10/24	Tue	0.12651	2454034	2006/10/25	Wed	0.12651

Table 6
PACIFIC exchange rate service.

Jul.Day	YYYY/MM/DD	Wdy	USD/CNY	Jul.Day	YYYY/MM/DD	Wdy	USD/CNY
2454035	2006/10/26	Thu	0.12671	2454036	2006/10/27	Fri	0.12674
2454039	2006/10/30	Mon	0.12692	2454040	2006/10/31	Tue	0.12693
2454041	2006/11/01	Wed	0.12700	2454042	2006/11/02	Thu	0.12700
2454043	2006/11/03	Fri	0.12702	2454046	2006/11/06	Mon	0.12698
2454047	2006/11/07	Tue	0.12698	2454048	2006/11/08	Wed	0.12707
2454049	2006/11/09	Thu	0.12708	2454050	2006/11/10	Fri	0.12718
2454054	2006/11/14	Tue	0.12716	2454055	2006/11/15	Wed	0.12713
2454056	2006/11/16	Thu	0.12706	2454057	2006/11/17	Fri	0.12706
2454060	2006/11/20	Mon	0.12699	2454061	2006/11/21	Tue	0.12707
2454062	2006/11/22	Wed	0.12714	2454063	2006/11/23	Thu	0.12717
2454064	2006/11/24	Fri	0.12727	2454067	2006/11/27	Mon	0.12754
2454068	2006/11/28	Tue	0.12749	2454069	2006/11/29	Wed	0.12764
2454070	2006/11/30	Thu	0.12764				

References

[1] R. Lawton, How should additive Holt–Winters estimates be corrected? International Journal of Forecasting 14 (1998) 393–403.
[2] G. Chevillon, D.F. Hendry, Non-parametric direct multistep estimation for forecasting economic processes, International Journal of Forecasting 21 (2005) 201–218.
[3] P.H. Franses, D. Van Dijk, The forecasting performance of various models for seasonality and nonlinearity for quarterly industrial production, International Journal of Forecasting 21 (2005) 87–102.
[4] K.D. Patterson, Forecasting the final vintage of real personal disposable income: a state space approach, International Journal of Forecasting 11 (1995) 395–405.
[5] J. Durbin, S.J. Koopman, Time Series Analysis by State Space Methods, Oxford University Press, Oxford, 2001.
[6] M.P. Clements, P.H. Franses, N.R. Swanson, Forecasting economic and financial time-series with non-linear models, International Journal of Forecasting 20 (2004) 169–183.
[7] C.M. Hurvich, Multistep forecasting of long memory series using fractional exponential models, International Journal of Forecasting 18 (2002) 167–179.
[8] J.W. Galbraith, T. Kisinbay, Content horizons for conditional variance forecasts, International Journal of Forecasting 21 (2005) 249–260.
[9] M. Hibon, T. Evgeniou, To combine or not to combine: selecting among forecasts and their combinations, International Journal of Forecasting 21 (2005) 15–24.
[10] H.S. Hippert, C.E. Pedreira, R.C. Souza, Neural networks for short-term load forecasting: a review and evaluation, IEEE Transactions on Power Systems 16 (2001) 215–226.
[11] H.S. Hippert, D.W. Bunn, R.C. Souza, Large neural networks for electricity load forecasting: are they overfitted? International Journal of Forecasting 21 (2005) 425–434.
[12] V. Vapnik, S.E. Golowich, A. Smola, Support vector method for function approximation, regression estimation, and signal processing, in: M.C. Mozer, M.I. Jordan, T. Petsche (Eds.), Advances in Neural Information Processing Systems, Vol. 9, MIT Press, Cambridge, MA, 1997, pp. 1–420.
[13] C.J.C. Burges, A tutorial on support vector machines for pattern recognition, Data Mining and Knowledge Discovery 2 (1998) 121–167.
[14] P. Navneet, Y.C. Edward, KDX: an indexer for support vector machines, IEEE Transaction on Knowledge and Data Engineering 18 (2006) 748–763.
[15] N. Julia, S. Christoph, S. Gabriele, Combined SVM-based feature selection and classification, Machine Learning 61 (2005) 129–150.
[16] Y. Yuan, T. Huang, A Polynomial Smooth Support Vector Machine for Classification, in: Lecture Notes in Artificial Intelligence, vol. 3584, Springer-Verlag, 2005, pp. 370–377.
[17] Y. Yuan, C. Li, A New Smooth Support Vector Machine, in: Lecture Notes in Artificial Intelligence, vol. 3801, Springer-Verlag, 2005, pp. 392–397.
[18] Francis E.H. Tay, L.J. Cao, Modified support vector machines in financial time series forecasting, Neurocomputing 48 (2002) 847–861.
[19] L.J. Cao, Support vector machines experts for time series forecasting, Neurocomputing 51 (2003) 321–339.
[20] K. Kyoungjae, Financial time series forecasting using support vector machines, Neurocomputing 55 (2003) 307–319.
[21] Y.J. Lee, W.F. Hsieh, C.M. Huang, ϵ-SSVR: a smooth support vector machine for ϵ-insensitive regression, IEEE Transactions on Knowledge and Data Engineering 17 (2005) 678–685.
[22] Y.J. Lee, O.L. Mangarasian, SSVM: a smooth support vector machine for classification, Computational Optimization and Applications 22 (2001) 5–21.
[23] J.Z. Zhang, C.X. Xu, Properties and numerical performance of quasi-Newton methods with modified quasi-Newton equations, Journal of Computational and Applied Mathematics 137 (2001) 269–278.
[24] C.X. Xu, J.Z. Zhang, A survey of quasi-Newton equations and quasi-Newton methods for optimization, Annals of Operations Research 103 (2001) 213–234.