Expert Systems with Applications 38 (2011) 5569–5576

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Stock trend prediction based on fractal feature selection and support vector machine

Li-Ping Ni *, Zhi-Wei Ni, Ya-Zhuo Gao

School of Management, HeFei University of Technology, HeFei 230009, China
Key Laboratory of Ministry of Education on Process Optimization & Intelligent Decision Making, HeFei 230009, China

ARTICLE INFO

Keywords:
Fractal feature selection
Support vector machine
Stock trend prediction

ABSTRACT

Stock trend prediction is regarded as a challenging task. Recently many researches have shown that a successful feature selection method can improve the prediction accuracy of stock market. This paper hybridizes fractal feature selection method and support vector machine to predict the direction of daily stock price index. Fractal feature selection method is suitable for solving the nonlinear problem and it can exactly spot how many important features we should choose. To evaluate the prediction accuracy of this method, this paper compares its performance with other five commonly used feature selection methods. The results show fractal feature selection method selects the relatively smaller number of features and it achieves the best average prediction accuracy.

1. Introduction

Stock prediction is a difficult task due to the nature of the stock data which is noisy and time varying. People are interested in predicting the stock trend, index or price. Therefore, many different methods and techniques have been presented. These methods include artificial neural network (ANN), linear and multi-linear regression (LR, MLR), genetic algorithm (GA), support vector machine (SVM) and other models. Atsalakis and Valavanis (2009) summarized the available articles that applied the soft computing methods to stock forecasting and pointed out that among them many utilized specific techniques to select important features.

In the area of stock prediction, feature selection refers to choosing a subset of original input variables which are usually technical or fundamental indicators. Because the selected feature subset can represent better the original character of dataset, prediction with them can improve the accuracy and efficiency. Cao, Chua, and Guan (2003) adopted Kernel Principal Component Analysis (KPCA) method as feature selection algorithm to predict stock price. Sui, Qi, Yu, Hu, and Zhao (2007) regarded the classification complexity of SVM as a feature selection criterion and used the selected feature subset to predict Shanghai Stock Exchange Composite Index (SSECI). Zhang, Sai, and Yuan (2008) integrated rough set (RS) and SVM to predict S&P 500 index and rough set is used for feature selection. Huang, Yang, and Chuang (2008) employed a wrapper approach to select the optimal feature subset and then used different classification algorithm to predict the trend in Taiwan and Korea stock markets. Huang and Tsai (2009) used the value of correlation coefficient of condition attribute and decision attribute to measure the importance of features and it selected seven relatively important stock market technical indicators to predict Taiwan index (FITX). Lee (2009) proposed a prediction model based on a hybrid feature selection method and SVM to predict the trend of stock market. The above mentioned papers showed that feature selection method they presented could improve the prediction accuracy to some extent. However these feature selection methods have their limitations, they cannot exactly spot the number of optimal features needed and some of them need to transform space. In this paper fractal feature selection algorithm is used to select relevant features. Fractal feature selection method has following advantages compared with other feature selection algorithms: First it can spot how many important features of the dataset are needed. Second it is suitable for solving nonlinear problem and can show the features that have nonlinear or even non-polynomial correlations. Finally it does not rotate the address space of the dataset, so the selection result is easy to be interpreted. Stock market is a nonlinear chaotic system and related researches have shown that it has apparent fractal character. We can use fractal feature selection algorithm to determine which technical indicator should be chosen.

The paper is organized as follows. Section 2 describes related work including fractal feature selection and its improved algorithm. Section 3 introduces support vector machine. Section 4 proposes the prediction method based on fractal feature selection and support vector machine. Section 5 presents the experimental results and analysis. Section 6 provides the conclusion.

* Corresponding author at: School of Management, HeFei University of Technology, HeFei 230009, China.
E-mail address: lilacnlp@126.com (L.-P. Ni).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.10.079

2. Related work

2.1. Fractal feature selection algorithm

Fractal feature selection algorithm was proposed by Traina, Jr. C., Traina, Wu, and Faloutsos (2000). The main idea of this algorithm is using fractal dimension to measure the importance of features. For a dataset, it has two dimensions. One is embedding dimension, the other is intrinsic dimension. Embedding dimension is the number of features of the dataset. Intrinsic dimension is the number of features that we should preserve regardless of the space where it is embedded. Fractal dimension can be seen as intrinsic dimension, so the upper bound of a dataset's fractal dimension is the number of features required to characterize a dataset. Based on this the greatest advantage of fractal feature selection algorithm is that it can determine the number of features in a feature subset.

Fractal dimension and embedding dimension have the following relations: If a dataset has all of its variables independent one from the others, its fractal dimension will be equal to its embedding dimension, otherwise its fractal dimension will be less than its embedding dimension. A redundant feature means the feature that has a correlation with other features. If we remove the redundant feature the fractal dimension will take a little change. That means the fractal dimension of a dataset is relatively not affected by redundant features. A feature is the least important one when it takes the smallest change of the dataset's fractal dimension after its removal. So fractal feature selection algorithm can be described as follows (Traina et al., 2000). Calculate the fractal dimension D of the whole dataset, and then sequentially drop the attributes which cause a minimum change of current dataset's fractal dimension until the number of remainder features is the upper bound of D.

2.2. Fractal dimension of a dataset

A dataset has fractal character if its partial distribution has the same property such as structure, statistic distribution etc. with itself (Bao, Yu, Sun, & Wang, 2004; Traina et al., 2000). The multifractal dimension D_q of a dataset can be measured by Eq. (1) (Yan, Li, & Yuan, 2006)

$$D_q = \begin{cases} \lim\limits_{r\to 0} \dfrac{\sum_i p_i \log p_i^q}{\log r} & q = 1 \\ \lim\limits_{r\to 0} \dfrac{1}{q-1} \dfrac{\log \sum_i p_i^q}{\log r} & q \neq 1 \end{cases} \qquad r \in [r_1, r_2] \tag{1}$$

where r is the grid size, p_i is the probability of the data points in the ith grid, q is an integer. Different q represents different meaning. For example when $q = 1$, D_1 is the information dimension, the change of D_1 means the change in trends. D_2 is the correlation dimension, the change of D_2 means the change of data distribution in the dataset (Barbara, 1999). Yan and Li (2007) pointed out multi-fractal dimension can reflect the different distribution of dataset. So multi-fractal dimension can describe the dataset better. In this paper we use multi-fractal dimension in fractal feature selection algorithm.

The box counting algorithm is often used to estimate fractal dimension. It places a standard grid of boxes upon the dataset and calculates the sum of the qth power of occupancies for the boxes with particular grid size, that is $s(r) = \sum_i C_{r,i}^q$. The fractal dimension equals $\frac{k_s}{q-1}$, k_s is the slope of the linear part of $\log s(r)$ versus $\log(r)$. Though least square method can estimate k_s, it fits all the points and thus noisy points will influence the fitting result. So a sectional line fitting method is used to fit the curve. The basic step of this method is as follows (Xie, 1992):

Suppose the data set to be fitted is S, $S = \{P_1, P_2, P_3, \ldots, P_{n-1}, P_n\}$. Set an angel α as a threshold.

Suppose a line L that is currently being fitted through $P_m, P_{m+1}, \ldots, P_k$, $(2 < k < n$, in the initial stage $m = 1$ and $k = 2)$ by least square method.

Step 1. Consider the point P_{k+1}. First find the corresponding point to P_m on line L, P'_m, then calculate the angle β between line L and line $P_{k+1}P'_m$. If $\beta \leqslant \alpha$, fit point $P_m, P_{m+1}, \ldots, P_k, P_{k+1}$ by least square method, update line L, $k = k + 1$, go to Step 1; If $\beta > \alpha$, go to Step 2.

Step 2. Record the line L as a sectional line, fit another new line $L1$ through P_k, P_{k+1}. Repeat the above processes for line $L1$.

Step 3. After every point in S has been fitted, suppose we get a group of sectional lines: $L, L1, L2, \ldots$ Choose the line which includes more than $n/2$ points to be fitted and has minimum fitting error.

2.3. Feature selection method based on fractal dimension and ant colony algorithm

The main disadvantage of fractal feature selection algorithm is its efficiency. Every time after a feature removal the fractal dimension of current dataset should be recomputed. In order to reduce the frequency of computing fractal dimension and accelerate the process of feature selecting we proposed fractal feature selection method based on fractal dimension and ant colony algorithm (herein after being referred to as "improved fractal feature selection method") (Ni, Ni, Wu, & Ye, 2009)

The improved fractal feature selection method can be constructed as follows.

Given the q_ith order fractal dimension D_{q_i} $(i = 1, 2, \ldots, t)$ (the value of q_i is determined in accordance with the dataset) of the dataset:

(1) Optimization goal by ant colony is to find the optimal feature subset which contains m $(m = \lceil D \rceil,\ D = \max(D_{q_i}))$ features and its fractal dimension is closest to the fractal dimension of original dataset.
(2) During once searching process, each ant visits m nodes; here one node represents a feature.
(3) Each ant releases pheromone on the nodes it travels according to fitness. The fitness function is defined as Eq. (2)

$$f(k) = \sqrt{\sum_{i=1}^{t} \left(frac_{q_i}(k) - D_{q_i} \right)^2} \tag{2}$$

where $frac_{q_i}(k)$ is the q_ith order fractal dimension of feature subset searched by ant k. The smaller the value of fitness function, the better the solution.

Through this positive feedback, pheromone on nodes which belong to better feature subsets gradually increase. Eventually ant colony converges to the optimal solution.

The overall process of the improved algorithm can be seen in Fig. 1. The process begins by generating a number of ants, n. Then each ant chooses a starting node according to the pheromone probability of nodes. From these initial positions, each ant starts to construct a candidate subset. They search successor nodes according to probabilistic transition rule until a search stopping criterion (the number of nodes passed by one ant reaches m) is satisfied. The candidate subsets are gathered and then evaluated. If the optimal solution does not change within a certain number of times N_s or the algorithm has executed a certain number of times $Nc_{\max}$, then the process halts and outputs the best feature subset. If neither condition holds, then the pheromone on each feature node is updated, a new set of ants are created and the process iterates once more.

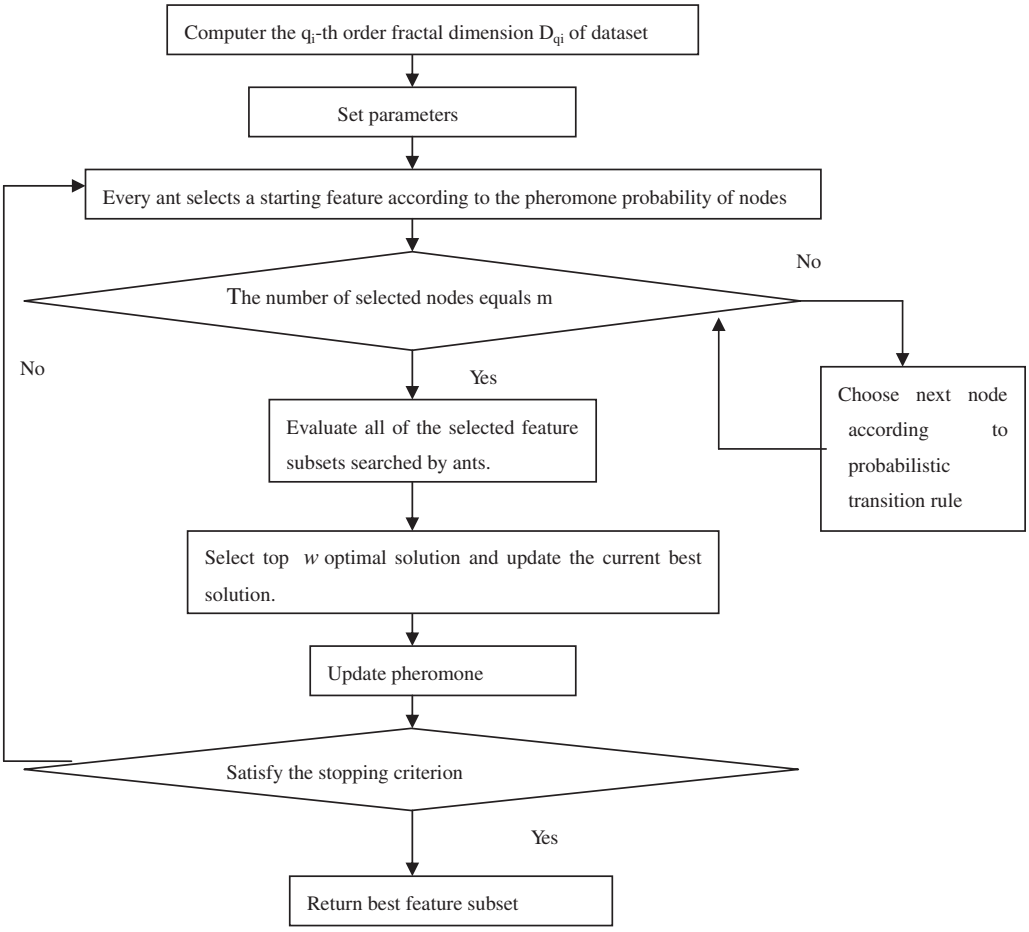

Fig. 1. The process of improved fractal feature selection method.

The following list is some key points in this algorithm.

(1) The pheromone probability of node i is defined according to the following equation:

$$p_i = \frac{(\tau_i(t) + L)^h}{\sum_{j=1}^{E}(\tau_j(t) + L)^h} \tag{3}$$

where $\tau_i(t)$ is the pheromone of node i at time t. E is the number of nodes. h and L are two parameters, $h \geqslant 2$. At the beginning $L \gg \tau_i(t)$, $(i = 1,2,3,\ldots,E)$, an ant tends to choose a starting node randomly. With the search when $\tau_i(t) > L$, an ant tends to choose the starting node with bigger τ.

(2) Probabilistic transition rule is defined as Eq. (4)

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_i(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{s \in J_k(i)} [\tau_s(t)]^\alpha \cdot [\eta_{is}(t)]^\beta}, & \text{if } j \in J_k(i) \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where k is the number of ants, η_{ij} is the heuristic factor of choosing feature j when at feature i, $J_k(i)$ is the set of nodes that have not yet been visited by ant k, $\alpha > 0$, $\beta > 0$ are two parameters that determine the relative importance of the pheromone value and heuristic factor. The heuristic factor $\eta_{ij}(t)$ is defined as follows:

$$\eta_{ij}(t) = 1 - |r_{ij}| \tag{5}$$

where r_{ij} is the correlation between node i and j (Hall, 2000). Eq. (6) is its definition (refer to Hall (2000) for details of this equation)

$$r_{ij} = \begin{cases} \frac{\mathrm{cov}(i,j)}{\sqrt{Var(i)Var(j)}} & \text{if } i,j \text{ are two continuous features} \\ \sum_{t=1}^{k} P(i = x_t) r_{i_{bx_t} j} & \text{if } i \text{ is numeric feature and } j \text{ is continuous feature} \\ \sum_{t=1}^{k}\sum_{m=1}^{l} P(i = x_t, j = y_m) r_{i,j} & i,j \text{ are two numeric features} \end{cases} \tag{6}$$

(3) Because the fractal dimensions of different feature subsets are relatively close (i.e. in a certain range of fractal dimension there are several feature subsets), a Rank-Based Version of Ant System (AS_{rank}) is introduced to prevent ants trapping into local minima. The basic idea of AS_{rank} is to sort the ants by its solution after all n ants have generated a candidate solution. Select top ω ants. The contribution to the feature node pheromone of an ant is weighted according to the rank μ of the ant. The pheromone of feature node is updated according to the following equation:

$$\tau_i(t+1) = \rho\tau_i(t) + \rho^0\Delta\tau_i^0 + \sum_{\mu=1}^{\omega}\rho^\mu\Delta\tau_i^\mu + \rho^*\Delta\tau_i^* \tag{7}$$

$$\Delta\tau_i^0 = \begin{cases} Q/f(k) & \text{if feature } i \text{ is visited by ant } k \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

$$\Delta\tau_i^{\mu} = \begin{cases} Q/f(k_{\mu}) & \text{if feature } i \text{ is visited by ant ranked } \mu(\mu \leqslant \omega) \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

$$\Delta\tau_i^{*} = \begin{cases} Q/f(k_{*}) & \text{if feature } i \text{ is a part of the best solution} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$f(k)$ and $f(k_{\mu})$ represents the fitness value of feature subset searched by ant k or the μth best ant respectively. $f(k^*)$ is the fitness value of the best solution. $\Delta\tau_i^{\mu}$ is the increase of pheromone on feature i caused by the μth best ant, $\Delta\tau_i^{*}$ is the increase of pheromone on feature i caused by elitist ants. ρ, ρ^0, ρ^{μ} $(1 \leqslant \mu \leqslant \omega)$ and ρ^* are pheromone incremental weight parameters. The values of these parameters meet the following criterion: $\rho^1 > \rho^2 > \cdots > \rho^{\omega} > \rho^0$, $\rho^{\omega} > \rho^*$.

3. Support vector machine

Support vector machine was developed by Vapnik (1995). Due to its outstanding performance in classification and regression many researches adopt it to forecast stock market. This paper uses its classification algorithm to predict the direction of stock market.

The theory of support vector machine for solving two-class classification problem can be described as follows.

Given a training dataset represented by (x_i, y_i), $i = 1, 2, \ldots, n$, $x_i \in R^d$, $y_i \in \{-1, 1\}$. Where n is the number of the training data, d is the original dimension of the training dataset and y_i is the label of the ith data point. To separate two different classes of data points SVM finds an optimal hyper plane by solving the following optimization problems:

$$\min_{w,b} \frac{1}{2}\|w\|^2 \tag{11}$$

With constraints

$$y_i(w_i + b) \geqslant 1, \quad \forall i \tag{12}$$

By introducing Lagarangian multipliers α, the optimization problem can be reformulated as Eq. (13)

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i [y_i(\omega \cdot x_i + b) - 1] \tag{13}$$

In order to get the minimum solution, partially differentiate w, b, α_i, and let them equal to zero

$$\begin{cases} \frac{\partial L}{\partial \omega} = 0 \Rightarrow \omega = \sum_{i=1}^{n} \alpha_i y_i x_i \\ \frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0 \\ \frac{\partial L}{\partial \alpha_i} = 0 \Rightarrow \alpha_i [y_i(w \cdot x_i + b) - 1] = 0 \end{cases} \tag{14}$$

According to Eqs. (14) and (12) the above mentioned problem can be considered as following dual problem:

$$\begin{cases} \max & \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \\ \text{s.t.} & \alpha_i \geqslant 0, \quad i = 1, \ldots, n \\ & \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases} \tag{15}$$

The optimal hyper plane decision function is

$$f(x) = \operatorname{sgn}\{(w^* \cdot x) + b^*\} = \operatorname{sgn}\left\{\sum_{i=1}^{n} \alpha_i^* y_i (x_i \cdot x) + b^*\right\} \tag{16}$$

For non-separation problem the SVM can be extended as follows:

$$\begin{cases} \max & \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \\ \text{s.t.} & 0 \leqslant \alpha_i \leqslant C, \quad i = 1, \ldots, n \\ & \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases} \tag{17}$$

where C is the penalty parameter.

Given a mapping $x \to \phi(x)$, which maps none separation data points into higher dimensional feature space and the dot product in the final space can be computed by Mercer kernel function K which means $\langle \phi(x_i) \cdot \phi(x_j) \rangle = K(x_i, x_j)$. So the decision function is:

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{n} \alpha_i^* y_i K(x_i, x) + b^*\right) \tag{18}$$

There are many different kernel functions such as polynomial kernel function, radius basis function (RBF) and Sigmoid function. RBF is used in this paper. RBF is defined as

$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$, γ is the kernel parameter

4. Prediction model based on fractal selection algorithm and SVM

Fig. 2 is the outline of the prediction model we proposed. First the improved fractal feature selection algorithm is used to get the optimal feature subset. Second the training set is trained by SVM. Because different parameters C and γ will influence the accuracy of SVM classifier, so a grid search method using k-cross validation (k is a parameter, in this paper k equals to 5) is used to find a best parameter combination. Third SVM with the best parameter combination is rerun on training set to form the training model and then testing data is predicted by the training model. Parameter C and γ grow exponentially, C is set from 2^{-5} to 2^{11}, γ is set from 2^{-11} to 2^{5}.

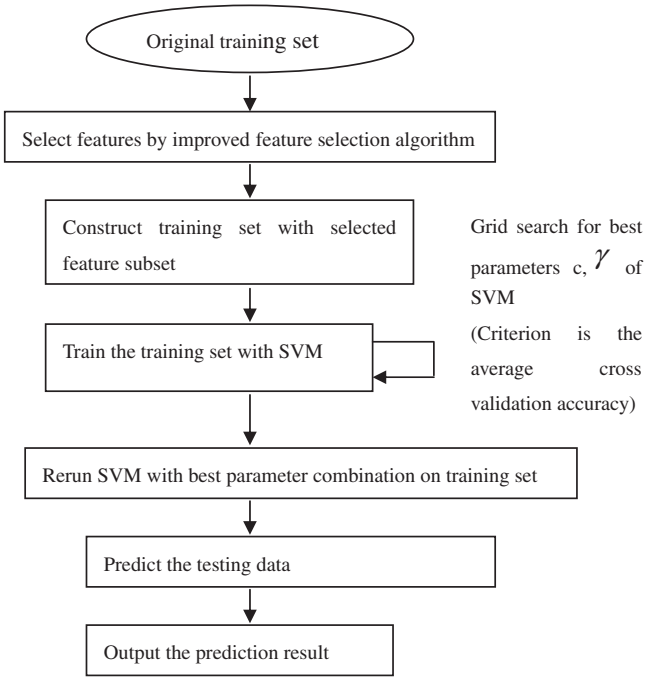

Fig. 2. The prediction process.

Generally the idea of integrating k-cross validation and grid method to select best parameters can be described as follows: set the range of parameters C and γ. Then do k-cross validation on training set with each pair of the parameters (C,γ). In k-cross validation method the training set is divided into k subsets. Each time, one of the k subsets is used as the test set and the other $k-1$ subsets are put together to form a training set. The average error across all k trials is computed. Finally select parameters C and γ with the minimum error as SVM parameters. That is to say the objective function of parameter selection is defined as Eq. (19)

$$\min_{c,\gamma} \frac{1}{k}\sum_{i=1}^{k} e_i \tag{19}$$

where k is the number of divided subsets, e_i is the error of ith trial.

By this method sometimes prediction set is earlier than training set. It does not conform to the character of time series. In this paper a modified k-cross validation method (Lin et al., 2008) is adopted to overcome that problem. This method only uses earlier subset to predict later one. More specifically, the process is as follows. First divide training set into k subsets. Second each time first i $(1 \leqslant i \leqslant k-1)$ subsets are put together to form the training set and then predict the next subset. Finally computer the average error across all $k-1$ trials.

5. Experiments and analysis

5.1. Data preparation and preprocessing

To evaluate the prediction model, this paper uses Shanghai Stock Exchange Composite Index (SSECI) as dataset and 19 technical indicators which are shown in Appendix A as the whole features.

Because the fluctuation range of each indicator has apparent difference so for each indicator its mean $\bar{x}_i$ and variance σ_i^2 are computed and then a set of rescaled variables is defined by Eq. (20)

$$x_i = \frac{x_i - \bar{x}_i}{\sigma_i} \tag{20}$$

In this paper the experiment is to predict the direction of daily stock price index. '1' and '−1' denote the next day's index is higher or lower than today's index respectively. The data set include 2171 data points which span from January 2000 to December 2008 and it is split into two subsets by slicing the time period, as shown in Table 1. Prediction model is built for each of the two subsets. Then the performance is used to evaluate the efficiency of the proposed method based on fractal feature selection and SVM.

5.2. Other related feature selection methods

In order to compare the performance, 5 other feature selection methods which have been implemented by WEKA are also applied in this experiment. The following is the illustration of them.

(1) Information Gain: this method measures the importance of a feature by measuring the Information Gain with the respect to the class. Information Gain is given by:

$$\text{InfoGain} = H(Y) - H(Y|X)$$

where X and Y are features and

$$H(Y) = -\sum_{y\in Y} p(y)\log_2(p(y)), \quad H(Y|X) = -\sum_{x\in X} p(x)\sum_{y\in Y} p(y|x)\log_2(p(y|x))$$

(2) Symmetrical uncertainty: this method measures the importance of a feature by measuring the symmetrical uncertainty with respect to the class, and the value of symmetrical uncertainty is:

$$SU = 2.0 * \frac{InfoGain}{H(Y)+H(X)}$$

(3) ReliefF: evaluates the worth of a feature by repeatedly sampling an instance and considering the value of the given feature for the nearest instance of the same and different class. The feature values of these instances are used to update the scores for each feature.

(4) Correlation-based Feature Selection (Cfs): Cfs method ranks the worth of subsets of features. As the feature subset space is usually huge, Cfs uses a best-first-search heuristic. This heuristic algorithm takes into account the usefulness of individual features for predicting the class along with the level of inter-correlation among them with the belief that good feature subsets contain features highly correlated with the class, yet uncorrelated with each other. The heuristic merit of a feature subset is:

$$\text{Cfs}_s = \frac{k\bar{r}_{cf}}{\sqrt{k+k(k-1)\bar{r}_{ff}}}$$

where k is the number of features in feature subset S, $\bar{r}_{ff}$ is the average feature to feature correlation. $\bar{r}_{cf}$ is the average feature to class correlation.

(5) OneR: OneR evaluates the importance of a feature according to the classification accuracy with this feature.

Table 1
Dividing the SSECI data set into two subsets.

Dataset	Training period	Testing period
1	2000/01/04–2006/12/29	2007/01/04–2007/12/28
2	2001/01/02–2007/12/23	2008/01/02–2008/12/23

5.3. Experimental result of feature selection

The experimental process can be described as follows: First calculate the multi-fractal dimension of two training sets respectively and then set parameters of the improved fractal feature selection algorithm. Finally run the algorithm to select features on each dataset and give the comparison with other five feature selection methods.

5.3.1. Multi-fractal dimension of two training sets

Tables 2 and 3 give the qth (q from 0 to 14) multi-fractal dimension of two training sets. Where $|R|$ represents the correlation coefficient of the linear fitting.

In theory the value of fractal dimension decreases as order q increases. However in practice the above-mentioned relationship is not always satisfied due to the fitting error. For example in training set 1 the computation result shows $D_0 < D_1$ and $D_1 < D_2$. In order to get more accurate result these dimensions which do not obey the fractal dimension change law are neglected before doing feature selection. Finally the q–D_q plots of training set 1 and training set 2 can be drawn as Figs. 3 and 4.

In the improved fractal feature selection algorithm choosing how many and which several order fractal dimensions should be decided first. This paper set $t = 4$ (i.e. four different order fractal dimensions are chosen). Figs. 3 and 4 show when q increases to a certain value, the fractal dimension changes little. So it is not necessary to select the higher order fractal dimension. The certain value of the two training sets is 6 and 14, respectively. So in training set 1 we set $q_0 = 2,\ q_1 = 3,\ q_2 = 4,\ q_3 = 14,\ D_{q_0} = D_2,\ D_{q_1} = D_3,\ D_{q_2} = D_4,\ D_{q_3} = D_{14}$.

Table 2
The multi-fractal dimension D_q of training set 1.

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8
1.265	3.834	3.846	3.762	3.699	3.644	3.593	3.549	3.51
$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$
0.9	0.974	0.965	0.965	0.964	0.963	0.962	0.962	0.962
D_9	D_{10}	D_{11}	D_{12}	D_{13}	D_{14}			
3.478	3.451	3.428	3.409	3.392	3.377			
$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$			
0.961	0.961	0.961	0.961	0.961	0.961			

Table 3
The multi-fractal dimension D_q of training set 2.

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8
0.7549	2.976	2.858	2.735	2.647	2.571	2.506	2.481	2.462
$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$
0.840	0.918	0.930	0.923	0.917	0.915	0.914	0.948	0.949
D_9	D_{10}	D_{11}	D_{12}	D_{13}	D_{14}			
2.445	2.429	2.414	2.399	2.386	2.373			
$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$	$\|R\|$			
0.949	0.950	0.950	0.950	0.950	0.950			

In training set 2, set $q_0 = 1$, $q_1 = 2$, $q_2 = 3$, $q_3 = 6$, $D_{q_0} = D_1$, $D_{q_1} = D_2$, $D_{q_2} = D_3$, $D_{q_3} = D_6$.

5.3.2. Parameter settings

Parameter settings of improved fractal feature selection algorithm are given as follows:

$$\alpha = 2, \quad \beta = 1, \quad \rho = 0.9, \quad \rho^0 = 0.1, \quad \rho^1 = 0.5,$$
$$\rho^2 = 0.45, \quad \rho^3 = 0.4, \quad \rho^* = 0.1, \quad \omega = 3, \quad L = 30,$$
$$h = 2, \quad N_s = 10, \quad Nc_{\max} = 300$$

The number of ants equals to $\lceil \frac{E}{m} \rceil$, where E is the dimension of original dataset and m is the upper bound of the maximum D_{q_i} of the dataset. According to the fractal dimension the value m of two training sets is 4 and 3, respectively.

In order to give the performance comparison, the parameters of other five feature selection methods are also set. Since Information Gain, Symmetrical uncertainty, ReliefF and OneR methods just rank the features, the number of features to be chosen should be determined. In the experiment when using Information Gain algorithm and Symmetrical uncertainty algorithm there are many features with zero Information Gain or symmetrical uncertainty. That is to say these features have no contribution abilities to classification, so in these two methods such features are discarded. For the other two algorithms in order to give the fair performance comparison

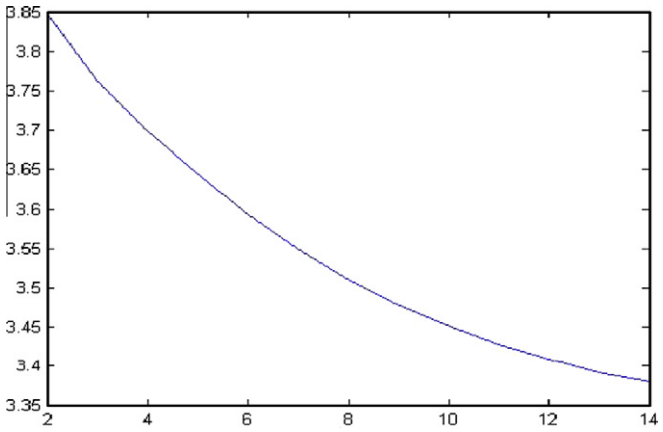

Fig. 3. The q–D_q plot of training set 1.

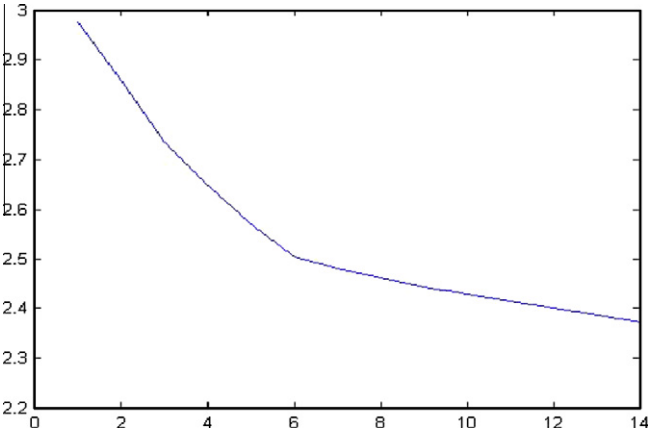

Fig. 4. The q–D_q plot of training set 2.

the number of selected features are set to the same as fractal feature selection method's.In addition Cfs algorithm adopts the default settings of WEKA.

5.3.3. Performance comparison

Tables 4 and 5 exhibit the performance of the six different feature selection methods on two datasets. The last row of the table represents prediction without feature selection.

The experimental results show most of these feature selection methods can improve the average prediction accuracy and reduce the dimension of original dataset greatly. With Tables 4 and 5, the average prediction accuracy of two datasets is 52.87% when using Information Gain method, 52.87% when using Symmetrical uncertain method, 46.32% when using ReliefF method, 56.60% when using Cfs method, 57.44% when using OneR method, and 58.86%when using the improved fractal feature selection method. Compared with the other feature selection methods the improved fractal feature selection method can give higher prediction accuracy with relatively small number of features.

Table 4
The performance of six different feature selection methods on dataset 1.

Feature selection methods	Selected feature subset	Prediction accuracy (%)
SVM + Information Gain	MTM, MFI	52.479
SVM + Symmetrical uncertain	MTM, MFI	52.479
SVM + ReliefF	*K*, TRIX, VR, PSY	46.694
SVM + Cfs	MFI	61.984
SVM + OneR	VR, PDI, RSI, ROC	64.463
SVM + improved fractal feature selection	AR, RSI, VRSI, MFI	64.050
SVM	All the features	62.810

Table 5
The performance of six different feature selection methods on dataset 2.

Feature selection methods	Selected feature subset	Prediction accuracy (%)
SVM + Information Gain	MFI, CCI, *D*, VR, CR, TRIX, AR, ROC, MTM, PSY	53.252
SVM + Symmetrical uncertain	MFI, *D*, VR, TRIX, CCI, CR, AR, ROC, MTM, PSY	53.252
SVM + ReliefF	BOLL, MFI, VR	45.935
SVM + Cfs	*D*, VR, AR, TRIX, CCI, MFI	51.220
SVM + OneR	MFI, RSI, MTM	50.407
SVM + improved fractal feature selection	MDI, VRSI, MFI	53.659
SVM	All the features	44.716

6. Conclusion

This paper hybridizes the fractal feature selection method and SVM to predict the daily trend of Shanghai Stock Exchange Index and compares with other five commonly used feature selection methods. The experimental results show fractal feature selection method can get higher prediction accuracy than prediction without feature selection and it is better than other five feature selection methods.

This paper only takes the technical indicators into consideration. In fact there are many factors affecting the stock price. So in the future we can try testing more influencing factors with this prediction model, including macro and micro factors in stock field or other financial fields.

Acknowledgement

This work was supported by the National Nature Science Foundation of China Under Grant No. 70871033, the National High Technology Research and Development Program of China under Grant No. 2007AA04Z116 and the Science Research and Development Foundation of HeFei University of Technology under Grant No. 2009HGXJ0040.

Appendix A

Appendix A.The table of definitions of 19 technical indices.

Feature	Description	Formula
MACD9	Moving average convergence and divergence	$(\mathrm{DIF}-\mathrm{DEA})*2$, $\mathrm{DIF}=\mathrm{EMA}_{12}-\mathrm{EMA}_{26}$, $\mathrm{DEA}=\mathrm{EMA}(\mathrm{DIF},9)$
BOLL20	Bollinger Bands	$\mathrm{MA}(20)$ $\mathrm{MA}(20)=\frac{1}{20}\sum_{i=1}^{20}C_{t-i+1}$ (C_t is the closing time at t)
K9	Stochastic index K	$K(9)_t=\frac{2}{3}*K(9)_{t-1}+\frac{1}{3}*\mathrm{RSV}(9)_t$
D9	Stochastic index D	$D(9)_t=\frac{2}{3}*D(9)_{t-1}+\frac{1}{3}*K(9)_t$
MTM10	Momentum	$\mathrm{MTM}(10)=C_t/C_{t-10}*100$
WR10	William's Overbought/Oversold Index	$\mathrm{WR}(10)_t=\left(\frac{\mathrm{HP}_{10}-C_t}{\mathrm{HP}_{10}-\mathrm{LP}_{10}}\right)*100$, where HP_{10} *is* the highest price in previous 10 days,and LP_{10} is the lowest price in previous 10 days
PSY12	Psychological Line	$\mathrm{PSY}(12)_t=\left(\frac{\mathrm{UD}_{12}}{12}\right)*100$, where UD_{12} *is* the number of upward days during previous 12 days
VR26	Volume ratio	$\mathrm{VR}(26)_t=\left(\mathrm{UV}_{26}+\frac{\mathrm{TV}_{26}}{2}\right)/(\mathrm{DV}_{26}+\frac{\mathrm{TV}_{26}}{2})$ where UV_{26} is the volume summation at upward days, DV_{26} is the volume summation at downward days and TV_{26} is the volume summation at previous 26 days
CR26	C ratio	$\mathrm{CR}(26)_t=\mathrm{BS}(26)_t/\mathrm{SS}(26)_t*100$ $\mathrm{BS}(26)_t=\sum_{i=1}^{n}\max\{0,H_{t-i+1}-\mathrm{TP}_{t-i}\}$ $\mathrm{SS}(26)_t=\sum_{i=1}^{n}\max\{0,\mathrm{TP}_{t-i}-L_{t-i+1}\}$, where H_i is the highest price at time i, TP_i is the middle price at time i, L_i is the lowest price at time i
MFI26	Money Flow Index	$\mathrm{MFI}(26)_t=\frac{100}{1+\mathrm{MR}(26)_t}$ $\mathrm{MR}(26)_t=\frac{\mathrm{PMF}(26)_n}{\mathrm{NMF}(26)_n}$, where PMF is the positive summation of cash flow in previous 26 days, NMF is the negative summation of cash flow in previous 26 days
AR26	A Ratio	$\mathrm{AR}(26)_t=\frac{\sum_{i=1}^{26}(H_{t-i+1}-O_{t-i+1})}{\sum_{i=1}^{26}(O_{t-i+1}-L_{t-i+1})}*100$, where O_i is open price at time i
BR26	B Ratio	$\mathrm{BR}(26)_t=\frac{\sum_{i=1}^{26}\max\{0,H_{t-i+1}-C_{t-i}\}}{\sum_{i=1}^{26}\max\{0,C_{t-i}-L_{t-i+1}\}}*100$
+DI14/PDI14	Directional indicator up	$+\mathrm{DI}(14)_t=\frac{\left(\sum_{i=t-13}^{t}(+DM_i)/14\right)}{\left(\sum_{i=t+13}^{t}TR_i/14\right)}*100$, where $+\mathrm{DM}_i=H_i-H_{i-1}$, $\mathrm{TR}_i=\max\{H_i-L_i,H_i-C_{i-1},L_i-C_{i-1}\}$
−DI14/MDI14	Directional indicator down	$-\mathrm{DI}(14)_t=\frac{\left(\sum_{i=t-13}^{t}(-DM_i)/14\right)}{\left(\sum_{i=t+13}^{t}TR_i/14\right)}*100$, where $-\mathrm{DM}_i=L_i-L_{i-1}$
RSI6	Relative Strength Index	$\mathrm{RSI}(6)_t=\frac{\mathrm{UC}(6)_t}{\mathrm{UC}(6)_t+\mathrm{DC}(6)_t}*100$, where $\mathrm{UC}(6)_t=\sum_{i=1}^{6}\max\{0,C_{t-i+1}-C_{t-i}\}$ $\mathrm{DC}(6)_t=\sum_{i=1}^{6}\max\{0,C_{t-i}-C_{t-i+1}\}$
TRIX9	Triple Exponentially Smoothed Average	$\mathrm{TRIX}(9)_t=\frac{\mathrm{TEMA}_t-\mathrm{TEMA}_{t-9}}{\mathrm{TEMA}_{t-9}}*100$, where $\mathrm{TEMA}_t=\frac{8}{10}\mathrm{TEMA}(9)_t+\frac{2}{10}\mathrm{DEMA}_t$ $\mathrm{DEMA}_t=\frac{8}{10}\mathrm{DEMA}(9)_t+\frac{2}{10}\mathrm{EMA}_t$
CCI14	Commodity Channel Index	$\mathrm{CCI}(14)_t=\mathrm{TP}_t-\mathrm{MATP}(14)_t/\alpha D(14)_t$, where $\mathrm{TP}_t=\frac{H_t+L_t+C_t}{3}$ $\alpha=0.015,\mathrm{MATP}(14)_t=\frac{1}{14}\sum_{i=1}^{14}\mathrm{TP}_{t-i+1}$ $D(14)_t=\frac{1}{14}\sum_{i=1}^{14}\lvert\mathrm{TP}_{t-i+1}-\mathrm{MATP}(14)_t\rvert$
ROC12	Rate of Change	$(C_t-C_{t-12})/C_{t-12}*100$
VRSI6	Volume Relative Strength Index	$\mathrm{VRSI6}=\mathrm{UV}(6)_t/V(6)_t*100$ $\mathrm{UV}(6)_t=\sum_{i=1}^{5}\max\{0,\mathrm{sign}(v_{t-i+1}-v_{t-i})\}v_{t-i+1}$, $v(6)_t=\sum_{i=1}^{5}v_{t-i+1}$

References

Atsalakis, G. S., & Valavanis, K. P. (2009). Surveying stock market forecasting techniques – Part II: Softcomputing methods. *Expert Systems with Applications, 36*(3), 5932–5941.

Xie, Y. B. (1992). Sectional line fitting by means of least sequence method. *Journal of Nan Chang Institute of Aeronautical Technology*(1), 19–25 (in Chinese).

Bao, Y. B., Yu, G., Sun, H. L., & Wang, D. L. (2004). Performance optimization of fractal dimension based feature selection algorithm. In *International conference on advances in web-age information management, Dalian, China* (pp. 739–744).

Barbara, D. (1999). *Chaotic mining: Knowledge discovery using the fractal dimension (extended abstract).* <http://www.almaden.ibm.com/cs/dmkd/papers/barbara.ps> Accessed 10.06.09.

Cao, L. J., Chua, K. S., & Guan, L. K. (2003). Combining KPCA with support vector machine for time series forecasting. In *Proceedings of IEEE international conference on computational intelligence for financial engineering* (pp. 325–329).

Hall, M. A. (2000).Correlation-based feature selection for discrete and numeric class machine learning. In *Proceedings of the 17th international conference on machine learning, Stanford, USA* (pp. 359–366).

Huang, C.-L., & Tsai, C.-Y. (2009). A hybrid SOFM-SVR with a filter-based feature selection for stock market forecasting. *Expert Systems with Applications, 36*(2), 1529–1539.

Huang, C.-J., Yang, D.-X., & Chuang, Y.-T. (2008). Application of wrapper approach and composite classifier to the stock trend prediction. *Expert Systems with Applications, 34*(4), 2870–2878.

Lee, M.-C. (2009). Using support vector machine with a hybrid feature selection method to the stock trend prediction. *Expert Systems with Applications, 36*(8), 10896–10904.

Lin, S.-K., Xu, C.-F., Qiao, J.-Z., Zhang, S.-M., Zhi, L.-J., & Yu, G. (2008). *Journal of Software, 19*(Suppl.), 121–130 (in Chinese).

Ni, L.-P., Ni, Z.-W., Wu, H., & Ye, H.-Y. (2009). Feature selection method based on fractal dimension and ant colony optimization algorithm. *Pattern Recognition and Artificial Intelligence, 22*(2), 293–298.

Sui, X.-s., Qi, Z.-y., Yu, D.-r., Hu, Q.-h., & Zhao, H. (2007). A novel feature selection approach using classification complexity for SVM of stock market trend prediction. In International conference on management science and engineering, Harbin, China (pp.1654–1659).

Traina, C., Jr., Traina, A., Wu, L., & Faloutsos, C. (2000). Fast feature selection using fractal dimension. In *Proceedings of SBBD (Brazilian symposium on databases), Paraiba, Brazil, 2000* (pp.158–171).

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

WEKA. *Software*. <http://www.cs.waikato.ac.nz/ml/weka/> Accessed 10.06.09.

Yan, G. H., & Li, Zh. H. (2007). Using cluster similarity to detect natural cluster hierarchies. In *Proceedings of the 4th international conference on fuzzy systems and knowledge discovery, Haikou, China* (pp. 291–295).

Yan, G. H., Li, Zh. H., & Yuan, L. (2006). On combining fractal dimension with GA for feature subset selecting. In *MICAI (5th Mexican international conference on artificial intelligence), Apizaco, Mexico* (pp. 543–553).

Zhang, T., Sai, Y., & Yuan, Z. (2008). Research of stock index futures prediction model based on rough set and support vector machine. In *Proceedings of the IEEE international conference on granular computing, Hangzhou, China* (pp. 797–800).