Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2013, Article ID 753251, 10 pages
http://dx.doi.org/10.1155/2013/753251

Research Article

A Rule-Based Model for Bankruptcy Prediction Based on an Improved Genetic Ant Colony Algorithm

Yudong Zhang,[1] Shuihua Wang,[1,2] and Genlin Ji[1]

[1] *School of Computer Science and Technology, Nanjing Normal University, Nanjing, Jiangsu 210023, China*
[2] *School of Electronic Science and Engineering, Nanjing University, Nanjing, Jiangsu 210046, China*

Correspondence should be addressed to Yudong Zhang; zhangyudongnuaa@gmail.com

Received 25 August 2013; Accepted 1 November 2013

Academic Editor: Xiaodong Lin

In this paper, we proposed a hybrid system to predict corporate bankruptcy. The whole procedure consists of the following four stages: first, sequential forward selection was used to extract the most important features; second, a rule-based model was chosen to fit the given dataset since it can present physical meaning; third, a genetic ant colony algorithm (GACA) was introduced; the fitness scaling strategy and the chaotic operator were incorporated with GACA, forming a new algorithm—fitness-scaling chaotic GACA (FSCGACA), which was used to seek the optimal parameters of the rule-based model; and finally, the stratified K-fold cross-validation technique was used to enhance the generalization of the model. Simulation experiments of 1000 corporations' data collected from 2006 to 2009 demonstrated that the proposed model was effective. It selected the 5 most important factors as "net income to stock broker's equality," "quick ratio," "retained earnings to total assets," "stockholders' equity to total assets," and "financial expenses to sales." The total misclassification error of the proposed FSCGACA was only 7.9%, exceeding the results of genetic algorithm (GA), ant colony algorithm (ACA), and GACA. The average computation time of the model is 2.02 s.

1. Introduction

Corporate bankruptcy is of great importance in economic phenomena. The health and success of the businesses are of widespread concern to policy makers, industry participants, investors, managers, and consumers [1]. It is a problem that affects the economy on a global scale. Accurately predicting the number and probability of failing firms serves as an index of the development and robustness of a country's economy [2, 3]. The high individual, economic, and social costs as a consequence of corporate failures or bankruptcies have spurred searches for better understanding and prediction capabilities [4].

Bankruptcy prediction is the technique of predicting bankruptcy and various measures of financial distress of public firms [5]. It is a vast area of finance and accounting research. The quantity of research is also a function of the availability of data; for public firms which went bankrupt or did not, numerous accounting ratios that might indicate danger can be calculated, and numerous other potential explanatory variables are also available.

1.1. Previous Works. Historically, numerous methods have been developed for predicting bankruptcy. Early research focused primarily on one-variable models such as financial ratios. The ratios were used individually and a cutoff score was established for each ratio based on minimizing misclassification. The one-variable methods were later criticized, in spite of their considerable results, because of the correlation among ratios and providing different signals for a form by ratios [1].

Later research turned to multivariable models that used statistical techniques such as multiple discriminant analysis (MDA) [6], logit [7], and quadratic interval logit [8]. Recently, research has shown that artificial intelligence such as feedforward neural networks (FNNs) can be an alternative methodology for classification problems to which traditional statistical methods have long been applied [9].

1.2. Our Contribution

1.2.1. Feature Selection. Large, high-dimensional data sets are common in the financial field. High-dimensional data presents many challenges for analysis; a fundamental challenge is the so-called *curse of dimensionality*. Observations in a high-dimensional space are more sparse and less representative than those in a low-dimensional space [10]. In this paper, we use the sequential feature selection method [11] to select only 5 features out of the original 20 features.

1.2.2. Rule-Based Model. Although studies and experiments demonstrate the usefulness of FNNs in different studies, there are some shortcomings in building and using the model. First, it is not easy for users to find an appropriate FNN model that can represent problem characteristics such as network architectures, learning methods, and parameters. Second, the FNNs should restudy as the data changes slightly. Third and most important, the user cannot readily comprehend the final rules that the NN models acquire. These unknown rules of FNNs are referred to as "black boxes."

As a solution, we have found that the "black box" problems can be solved successfully using a rule-based approach. It is capable of extracting classification rules that are easy for users to recognize [12]. Among these systems, a rule-based solution is widely used for classification problems, either through supervised or unsupervised learning.

1.2.3. Fitness-Scaling Chaotic GACA. Zhang and Wu proposed the genetic ant colony algorithm (GACA) that combines the genetic algorithm (GA) and the ant colony algorithm (ACA) [3]. The GACA performs well to find global optima, yet it is easy to be trapped into local minima at some extreme conditions. In order to improve the performance of GACA, two improvements are introduced. (1) Traditional selection function uses fitness values to select the individuals of the next generation. It assigns a higher probability of selection to individuals with higher fitness values. However, individuals with smaller fitness values will have little chance to be selected, forcing the population gathering near the best individual. The power-rank scaling method will adjust the fitness values in order to make the population diverse. (2) Chaos is introduced to improve the robustness of basic GACA, considering its outstanding performance of jumping out of stagnation. The improved algorithm is called fitness-scaling chaotic GACA (FSCGACA).

1.2.4. Cross-Validation. Constructing the best rule is a challenge due to the following two problems. The first is the overfitting problem, namely, the rules fit training data well but perform poorly out of sample. The other is the underfitting problem. The optimization algorithm may fail to determine the global minima because it can be misled by the local minima [13].

The solution to the first problem is to use cross-validation [14] which divides the dataset into training subset and validation subset. Then, the validation subset is used to monitor overfitting. The solution to the second problem is to develop a novel powerful global optimization method.

1.3. Structure. The structure of this paper is organized as follows. Section 2 introduced the basic concepts of ACA and GACA. Section 3 proposed a novel fitness-scaling chaotic genetic ant colony algorithm (FSCGACA) and gave its pseudocode. Section 4 discussed the sequential feature selection method, particularly, the sequential forward selection. Section 5 employed the rule-based model in the application of bankruptcy prediction. Section 6 presented the stratified k-fold cross-validation to avoid the overfitting. Experiments in Section 7 showed every step of the proposed bankruptcy prediction system and compared the proposed FSCGACA with GA, ACA, and GACA in terms of classification accuracy and computation time. Besides, we demonstrated the necessity of feature selection and compared the proposed rule-based model with the FNN model. Finally, Section 8 was devoted to conclusions.

2. Background

As GA is well known to the readers, we discussed the basic concepts of ACA and GACA in this section.

2.1. Introduction of ACA. ACA is an algorithm developed recently to simulate the behavior of real ants to rapidly establish the shortest route from a food source to their nest and vice versa [12]. Ants begin randomly searching for food in the area surrounding their nest. When an individual ant encounters food along its path, it deposits a small quantity of pheromone at that location [15]. Other ants in the neighborhood detect this marked pheromone trail. As more ants follow the pheromone rich trail, the probability of the trial being followed by other ants is further enhanced by increased pheromone deposition [16]. This autocatalytic process reinforced by a positive feedback mechanism helps the ants to establish the shortest route. The flowchart of the algorithms is stated as follows.

Suppose an undirected graph $G = (O, A)$, where O is the set of nodes and A is the set of arcs connecting the nodes. The density of the nodes determines both the precision of a solution and the memory and computation time demands of the algorithm. All arcs O are initialized with a small amount of pheromone τ_0. The target is to find the shortest path from the source node O_1 to the destination node O_2.

In the second step, N ants are sequentially launched from O_1, where N is the number of ants in the colony. Each ant walks pseudorandomly from node to node via connecting arcs as far as the O_2 or dead end is reached. When deciding which node x to go to from a specific node y, the probability P_{xy} is assigned as follows:

$$P_{xy} = \frac{\tau_{xy}(k)^{\alpha}\eta_{xy}(k)^{\beta}}{\sum_{y \in \text{allowed}_y} \tau_{xy}(k)^{\alpha}\eta_{xy}(k)^{\beta}}. \tag{1}$$

Here, trail level $\tau_{xy}(k)$ is the amount of pheromone currently available at step k in the arc from node x to node y. It indicates how proficient the ant has been in the past to make the move from x to y. The attractiveness $\eta_{xy}(k)$ is the desirability of move from x to y. Parameters α and β control the

Table 1: Summaries of fitness scaling techniques.

Scaling types	Formula	Parameter explanation
Linear scaling	$f_{\text{linear}} = a + b \times f_{\text{raw}}$	a and b are constants defined by the users
Rank scaling	$f_{\text{rank}} = r$	r denotes the rank of the individual/ant
Power scaling	$f_{\text{power}} = f_{\text{raw}}^{p}$	p is a problem-dependent exponent that might change during a run to stretch or shrink the range as needed
Top scaling	$f_{\text{top}} = \begin{matrix} sf_{\text{raw}} \geq c \\ 0 f_{\text{raw}} < c \end{matrix}$	s is the user-defined constant; c is the threshold

relative importance of trail level and attractiveness, respectively. The trail levels of all arcs are updated according to moves that were part of "good" or "bad" solutions.

Consider that

$$\tau_{xy}(k+1) = \begin{cases} (1-\rho)\left[\tau_{xy}(k) + Q\right], & (x,y) \in \text{BestRoute} \\ (1-\rho)\,\tau_{xy}(k), & (x,y) \notin \text{BestRoute}. \end{cases} \tag{2}$$

Here, ρ denotes the pheromone evaporation coefficient and Q denotes a pheromone constant. Pheromone evaporation also has the advantage of avoiding the convergence to a locally optimal solution. If there was no evaporation at all, the paths chosen by the first ants would tend to be excessively attractive to the following ones.

At any iteration, the best route is calculated from N routes. The pheromones of the best route are enforced while others evaporate. It should be noted that local updates exist in some models; however, the models with local updates cannot guarantee convergence.

It should be noted that combinatory optimization problem can be solved directly by ACA; however, our mission is a continuous space optimization problem, so ACA cannot be directly used to solve our problem. A special coding strategy can be used to transform the continuous space into the routine search problem [17]. Figure 1 gives a simple example coding the value 4.85 as a routine.

2.2. Introduction of GACA. ACA converges relatively slowly due to the lack of pheromones during the initial stages. The ants search tediously at first, and once the pheromones have accumulated to some degree, the ants will converge to the optimal location relatively more quickly. Conversely, GA converges to the optimal solutions quickly at first, after which the population iteratively vibrates near the optimal solution.

Combining the advantages of the algorithms, the GACA was proposed by Zhang and Wu [3], and it can be divided into two stages. In the coarse-searching stage, GA approximates the neighborhood area of the optimal point. The processes are repeated until *MaxGAEpoch* iterations. In the fine-searching stage, ACA seeks the exact position of the optimal points. The concept and flowchart of GACA are depicted in Figure 2.

The parameter *MaxGAEpoch* is determined through trial and error. If it is too large, the GA will be excessively time-consuming. Conversely, if it is too small, GA does not approximate the optimal area and the algorithm may be misled by local minima. Setting *MaxGAEpoch* as 10 is appropriate to produce both efficient and inclusive results. The ACA will terminate according to any of the following criteria: the maximum epochs of ACA (*MaxACAEpoch*), the maximum stagnation (*MaxStag*), and the fitness tolerance (*FitTol*). These values are also determined by trial and error.

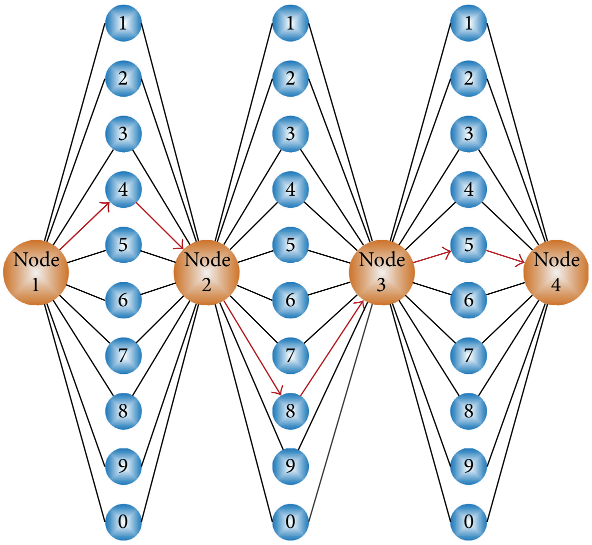


Figure 1: Coding strategy for ACA solving continuous optimization.

3. Fitness-Scaling Chaotic GACA

GACA has proven to perform better than GA and ACA in theories and simulations [18]. Furthermore, we can make improvements via following two aspects.

Fitness scaling converts the raw fitness scores that are returned by the fitness function to values in a range that is suitable for the selection function [19]. The selection function uses the scaled fitness values to select the bees of the next generation. The selection function assigns a higher probability of selection to bees with higher scaled values [20]. There exist numerous of fitness scaling methods, four popular ones of which are selected and shown in Table 1.

Among those fitness scaling methods, the power scaling finds a solution that is nearly the most quickly due to improvement of diversity, but it suffers from instability [21]. Meanwhile, the rank scaling shows stability on different types

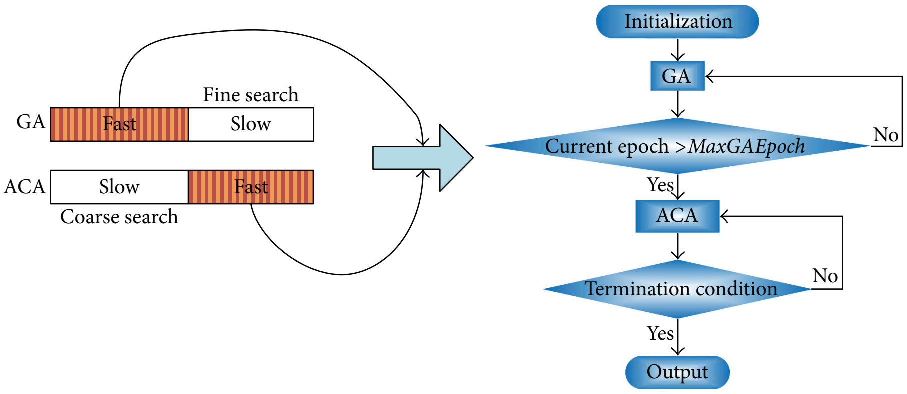


Figure 2: The concept and flowchart of GACA.

of tests [22]. Therefore, a new power-rank scaling method was proposed combining both power and rank strategies as follows:

$$f_i = \text{Scale}_{\text{power}}\left(\text{Scale}_{\text{rank}}\left(f_{\text{raw}}\right)\right) = \frac{r_i^p}{\sum_{i=1}^{N} r_i^p}, \tag{3}$$

where r_i is the rank of ith individual/ant and N is the number of population. Our strategy contains a three-step process. First, all individuals/ants are sorted to obtain the corresponding ranks. Second, powers are computed for exponential values k. Third, the scaled values are normalized by dividing the sum of the scaled values over the entire population.

Chaos theory is epitomized by the so-called butterfly effect established by Lorenz [23]. Attempting to simulate numerically a global weather system, Lorenz discovered that minute changes in initial conditions steered subsequent simulations towards radically different final results, rendering long-term prediction impossible in general [24]. Sensitive dependence on initial conditions is not only observed in complex systems but also in the simplest logistic equation. The well-known logistic equation shows that

$$x_{n+1} = 4 \times x_n \times \left(1 - x_n\right), \tag{4}$$

where $x_0 \in (0,1)$ and $x_0 \notin \{0.25, 0.5, 0.75\}$. The chaotic series can be used to generate the mutation operation. In all, the pseudocodes of FSCGACA are listed as follows:

Step 1 (parameter setting). Determine the population size N, crossover probability P_C, mutation probability P_M, elite selection probability P_E, trail level factor α, attractiveness factor β, pheromone evaporation coefficient ρ, pheromone constant Q, power scale factor p, initial logistic point x_0, and set iteration epoch $k = 0$,

Step 2 (initialization). Generate feasible solutions $x_i(k)$, $i \in [1, 2, \ldots, N]$ randomly. Their corresponding fitness values $f_{\text{raw}}[x_i(k)]$ are evaluated and scaled by formula (3) as $f[x_i(k)]$.

Step 3 (elitist selection). Select the best $P_E * N$ individuals to replace the same number worst individuals.

Step 4 (crossover). Choose $P_C * N$ individuals and do one-point crossover. That is, to choose the locus randomly, all data beyond the locus in either parent is swapped. The resulting individuals are the offspring.

Step 5 (mutation). Choose $P_E * N$ individuals and do uniform mutation. This operator replaces the value of the chosen individual with the chaotic number generated by formula (4) and maps it to the user-specified upper and lower bounds.

Step 6 (evaluation). The new individuals are supposed as $x_i'(k)$. Their corresponding fitness values $f_{\text{raw}}[x_i'(k)]$ are evaluated and scaled by formula (3) as $f[x_i'(k)]$.

Step 7 (update). If $f[x_i'(k)] < f[x_i(k)]$, then $x_i(k+1) = x_i'(k)$; otherwise, $x_i(k+1) = x_i(k)$.

Step 8. $k = k + 1$.

Step 9. If $k < MaxGAEpoch$, jump to Step 3; otherwise, jump to ACA Stage: Step 10.

Step 10 (transform). Generate $10 \times 10 \times \cdots \times 10$ graph according to problem precision. N Individuals are transformed to N ants. The paths corresponding to the values of individuals $x_i(k)$ (see Figure 1) are spread by pheromones τ with the amount of $f[x_i(k)]$. The heuristic function values η are set equal to the scaled fitness values $f[x_i(k)]$.

Step 11 (path selection). For each ant, select the new path by formula (1).

Step 12 (pheromone update). The trails are updated by formula (2).

Step 13. $k = k + 1$.

Step 14. If termination conditions are met, jump to Step 15; otherwise, jump to Step 11.

Step 15. Select and output the best route $x_{i^*}(k)$.

Table 2: Encoding mechanism.

Antecedent element 1		⋯	Antecedent element n		Consequent element
$\mathrm{CV}_{1\min}$	$\mathrm{CV}_{1\max}$	⋯	$\mathrm{CV}_{n\min}$	$\mathrm{CV}_{n\max}$	Bankrupt

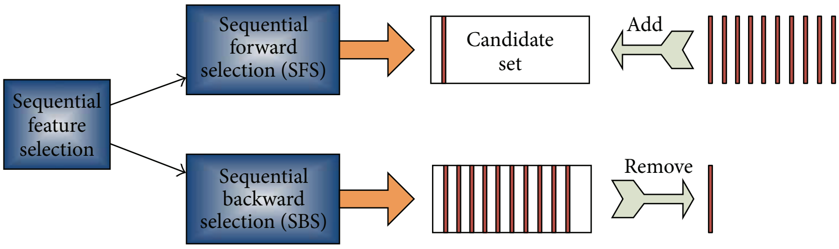

Sequential feature selection
Sequential forward selection (SFS)
Sequential backward selection (SBS)
Candidate set
Add
Remove

Figure 3: Two variants of sequential feature selection: the sequential forward selection and sequential backward selection.

4. Feature Selection

Classification methods often begin with some types of dimension reduction, by which high-dimensional data are approximated by points in a lower-dimensional space. In this paper, a sequential feature selection method [25] was applied. An objective function in feature selection is defined as *criteria*. Feature selection seeks to minimize the criteria over all feasible feature subsets. Common criteria include "mean squared error" and "misclassification rate" for regression models and classification models, respectively.

Feature selection has two variants shown in Figure 3. In sequential forward selection (SFS), features are sequentially added to an empty candidate set until the addition of further features does not decrease the criterion [26]. In sequential backward selection (SBS), features are sequentially removed from a full candidate set until the removal of further features increases the criterion [27].

In this paper, since the original features are as high as 20, we use the SFS to determine the important features. We defined the criteria as the deviance of the fit using "*binomial*" [28]. We use *binomial* criterion because the prediction contains two outcomes (either bankrupt or nonbankrupt), and we use "deviance of fit" since it can represent the classification margin while the "misclassification rate" cannot [29].

5. Rule-Based Model

Assume that the classification problem contains c classes in the n dimensional pattern space and there are p vectors $X_i = [x_{i1}, x_{i2}, \ldots, x_{in}]$ $(i = 1, 2, \ldots, p)$. An if-then rule is presented as

Bankruptcy Rule

If $(\mathrm{CV}_{1\min} \le x_{i1} \le \mathrm{CV}_{1\max})$ and $(\mathrm{CV}_{2\min} \le x_{i2} \le \mathrm{CV}_{2\max})$ and $\cdots$ and $(\mathrm{CV}_{n\min} \le x_{in} \le \mathrm{CV}_{n\max})$,

then the firm will bankrupt,

where n is the number of attributes and $\mathrm{CV}_{k\min}$ and $\mathrm{CV}_{k\max}$ are the minimum and maximum bounds of the kth attribute x_{ik}, respectively. The rule is then encoded as Table 2.

We choose the misclassification error (ME) as the fitness evaluation which is defined as below

$$\mathrm{ME} = \frac{\text{Number of misclassified firms}}{\text{Number of all firms}}. \tag{5}$$

The goal is to find the optimal parameters in Table 2 to make the ME as small as possible. Classification Accuracy (CA) is defined as the ratio of number of classified firms to the number of all firms. The sum of ME and CA is 1

$$\mathrm{CA} = \frac{\text{Number of classified firms}}{\text{Number of all firms}}. \tag{6}$$

6. Stratified K-fold Cross-Validation

Typically, a statistical model that deals with the inherent data variability is inferred from the database (i.e., the training set) and employed by statistical learning machines for the automatic construction of classifiers. The model has a set of adjustable parameters that are estimated in the learning phase using a set of examples. Nevertheless, the learning machine must ensure a reliable estimation of the parameters and consequently good generalization, that is, correct responses to new examples. Hence, the learning device must efficiently find a trade-off between its complexity, which is measured by several variables, such as the effective number of free parameters of the classifier and the feature input space dimension, and the information on the problem given by the training set (e.g., measured by the number of samples).

Cross-validation methods are usually employed to assess the statistical relevance of the classifiers. It consists of four types: random subsampling, K-fold cross-validation, leave-one-out validation, and Monte Carlo Cross-Validation [30]. The K-fold cross-validation is applied due to its simple and easy properties while using all data for training and validation. The mechanism is to create a K-fold partition of the whole dataset, repeat K times to use $K - 1$ folds for training and a left fold for validation, and finally average the error rates of K experiments. The schematic diagram of 5-fold cross-validation is shown in Figure 4.

The K folds can be purely random partitioned; however, some folds may have quite different distributions from other

Table 3: Five selected variables by SFS method.

Variables	Description
x_1	Net income to stock broker's equality
x_2	Quick ratio
x_4	Retained earnings to total assets
x_5	Stockholders' equity to total assets
x_{12}	Financial expenses to sales

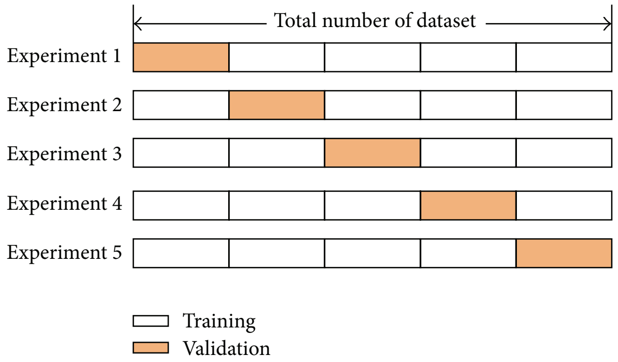


Figure 4: A 5-fold cross-validation.

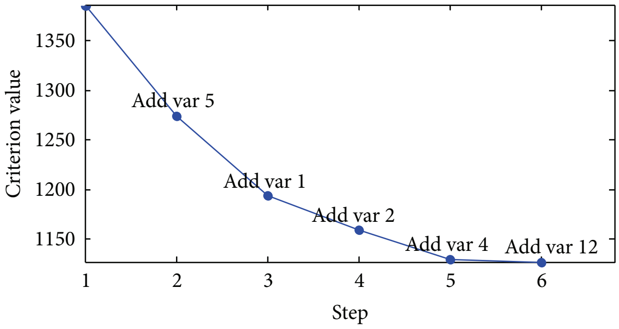


Figure 5: Six steps of SFS add variables 5, 1, 2, 4, and 12 sequentially.

folds. Therefore, the stratified K-fold cross-validation was employed, in which every fold has nearly the same class distributions [31]. The folds are selected so that the mean response value is approximately equal in all the folds. In case of a dichotomous classification, this means that each fold contains roughly the same proportions of the two types of class labels.

Another challenge was to determine the number of folds. If K is set too large, the bias of the true error rate estimator will be small, but the variance of the estimator will be large and the computation will be time-consuming. Alternatively, if K is set too small, the computation time will decrease and the variance of the estimator will be small, but the bias of the estimator will be large [32]. In this study, we empirically determined K to be 5 through the trial-and-error method.

7. Experiments and Discussions

The program was in-house developed by MATLAB 2013a and run on IBM P4 machine with 2 GHz processor and 1 G ram. The data set contains 1000 externally audited midsized manufacturing firms, 500 of which filed for bankruptcy and the other 500 for nonbankruptcy during the period 2006–2009. Each observation contains 21 financial indicators, of which 20 variables include the financial statistical measurements of the corporation and the last variable indicates bankruptcy status.

7.1. Feature Selection. In the feature selection stage, we normalized all the 20 variables to the range $[0, 1]$ and then selected 5 financial indicators using SFS method. The algorithm only consists of 6 steps, in which variable 5, 1, 2, 4, and 12 are sequentially added into the model as shown in Figure 5. The criterion value decreases gradually as the SFS progresses until any addition of a new variable will increase the criterion value. The physical meanings of the 5 selected variables are listed in Table 3. The other literature discussing bankruptcy also include those 5 features [33, 34], which demonstrate the effectiveness of SFS and the importance of those selected 5 features.

7.2. Algorithm Comparison. In this section, the rule-based model was established based on the 5 selected variables. The proposed FSCGACA was employed to optimize the parameters of the rule-based model. Besides, we run GA [34], ACA [3], and GACA [3] for comparison. The parameters and termination criteria of all algorithms were obtained through trial-and-error and listed in Table 4.

Each algorithm ran 100 times to eliminate the randomness. The average of the results of 100 runs is listed in Table 5. It shows the rule model established by all algorithms and their corresponding classification accuracy. The proposed FSCGACA performs best and achieves the highest CA as 92.1%, followed by the GACA with the CA as 91.8%. The ACA achieves 91.4% CA. The GA is the worst algorithm with the CA as 91.1%. The rule-based model established by the proposed FSCGACA can be translated as

> **if** "net income to stock broker's equality" is between 0.1324 and 0.6573 and
>
> "quick ratio" is between 0.0257 and 0.8038 and
>
> "retained earnings to total assets" falls between 0.0138 and 0.8957 and
>
> "stockholders' equity to total assets" falls between 0.0226 and 0.8168 and
>
> "financial expenses to sales" falls between 0.0522 and 0.5805,
>
> **then** the firm will bankrupt.

7.3. Convergence Performance. A typical run is shown in Figure 6. The convergence curve of the proposed FSCGACA is distinct from the others. In the beginning (from 1st to 25th epoch), the curve exhibits slowest the decline, followed by the one fastest (from 25th to 75th epoch) till finding the global minimal point (after 75th epoch). This kind of declining way adheres to our expectation. FSCGACA spends

Table 4: Parameters setting of algorithms.

Algorithm	Parameters setting
GA	$N = 20, P_C = 0.8, P_M = 0.1, P_E = 0.1$
ACA	$N = 20, \alpha = 1, \beta = 3, \rho = 0.1, Q = 500$
GACA	$N = 20, \alpha = 1, \beta = 3, \rho = 0.1, Q = 500, P_C = 0.8, P_M = 0.1, P_E = 0.1$
FSCGACA	$N = 20, \alpha = 1, \beta = 3, \rho = 0.1, Q = 500, P_C = 0.8, P_M = 0.1, P_E = 0.1, p = 2, x_0 = 0.15$
Algorithm	Termination criteria
GA	$MaxStag = 100, FitTol = 10^{-4}, MaxEpoch = 100$
ACA	$MaxStag = 100, FitTol = 10^{-4}, MaxEpoch = 100$
GACA	$MaxStag = 100, FitTol = 10^{-4}, MaxGAEpoch = 10, MaxACAEpoch = 90$
FSCGACA	$MaxStag = 100, FitTol = 10^{-4}, MaxGAEpoch = 10, MaxACAEpoch = 90$

Table 5: Rules extracted by algorithms (average of 100 runs).

Algorithm	Antecedent element 1		Antecedent element 2		Antecedent element 3		Antecedent element 4		Antecedent element 5		CA*	ME**
	$CV_{1\min}$	$CV_{1\max}$	$CV_{2\min}$	$CV_{2\max}$	$CV_{4\min}$	$CV_{4\max}$	$CV_{5\min}$	$CV_{5\max}$	$CV_{12\min}$	$CV_{12\max}$		
GA	0.1106	0.6708	0.0353	0.8380	0.1030	0.8964	0.0885	0.7925	0.0585	0.5787	91.1%	8.9%
ACA	0.1350	0.6649	0.0330	0.8055	0.0814	0.8866	0.0329	0.8997	0.0487	0.7981	91.4%	8.6%
GACA	0.1361	0.6637	0.0341	0.8149	0.1012	0.8960	0.0728	0.7964	0.0611	0.6842	91.8%	8.2%
FSCGACA	0.1324	0.6573	0.0257	0.8038	0.0138	0.8957	0.0226	0.8168	0.0522	0.5805	92.1%	7.9%

*CA: classification accuracy, **ME: misclassification error.

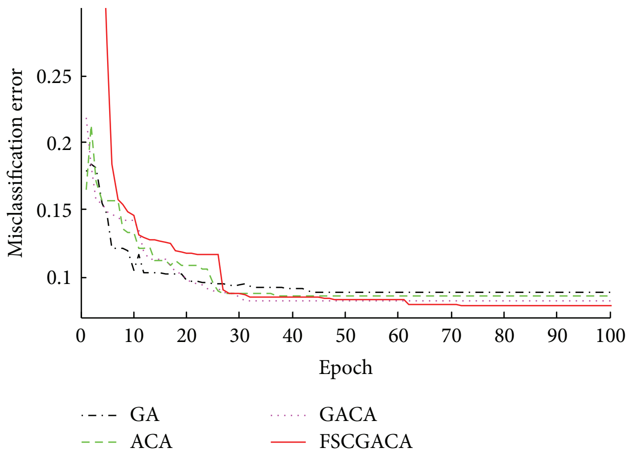


Figure 6: Curves of misclassification error versus epoch.

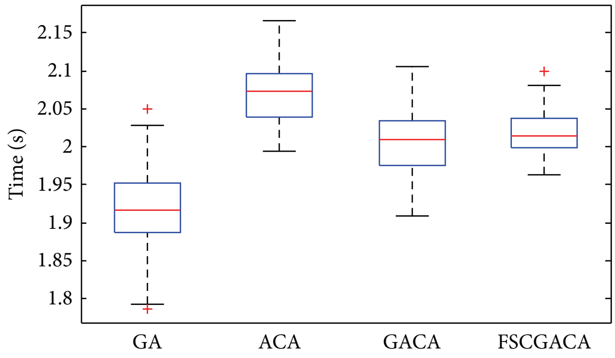


Figure 7: Boxplot of computation time comparison (100 runs).

more individuals in exploring areas approximated to global minimal points during the coarse-searching stage, so, it does not perform as well as we expected in terms of best fitness function, but it has located more potential areas. Afterwards, the fitness curve exhibits the sharpest decline when exploiting those areas in the fine-searching stage. Subsequently, the FSCGACA becomes dominant among all algorithms from 65th epoch. In all, Figure 6 indicates that the FSCGACA regulates the tradeoff between exploration and exploitation in a remarkably efficient way, so it exceeds other algorithms including GA, ACA, and GACA.

7.4. Computation Time Comparison. The distribution of computation time of 100 runs is shown in Figure 7. The central mark denotes the median, the edges of the box denote the 25th and 75th percentile, the whiskers extend to the most extreme data points, and the outliers are plotted as the plus symbol. The average computation times of the GA, ACA, GACA, and FSCGACA are 1.91, 2.07, 2.01, and 2.02 s, respectively.

The GA costs the least time, while the ACA costs the most time, because GA generates new individual using either crossover (1 operation generate 2 offspring) or mutation (1 operation generates 1 offspring), but the ACA generates new path by combining arcs by formula (1). Since the precision of our model is 10^{-4}, a new path should combine 4 arcs, that is, to repeat formula (1) four times as shown in Figure 8. Therefore, ACA will cost more time than GA. The GACA and FSCGACA run GA first and followed by ACA, so their computation time is between GA and ACA.

Table 6: Encoding mechanism without feature selection.

Indication element (logical)				Antecedent element (real valued)					Consequent element
t_1	t_2	$\cdots$	t_{20}	$\text{CV}_{1\text{min}}$	$\text{CV}_{1\text{max}}$	$\cdots$	$\text{CV}_{n\text{min}}$	$\text{CV}_{n\text{max}}$	Bankrupt

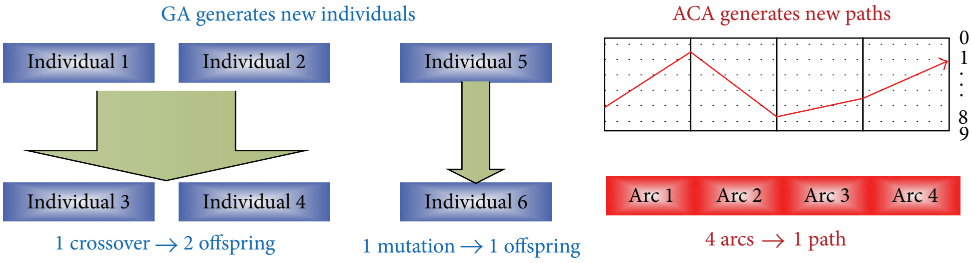


Figure 8: GA versus ACA in terms of generating new offspring/paths.

7.5. Effect of Feature Selection. In this section that we discuss the advantages of feature selection. If we omit feature selection in the rule-based model, the coding strategy will include 20 logical indication elements $\mathbf{t} = [t_1, t_2, \ldots, t_{20}]$, of which 1 indicates that the corresponding feature is included and 0 denotes it is neglected. Table 6 gives the encoding mechanism without using feature selection.

Bankruptcy Rule with Logical Indicators

If $(\text{CV}_{1\,\text{min}} \le x_{i1} \le \text{CV}_{1\,\text{max}})$ or $(!t_1)$ and $(\text{CV}_{2\,\text{min}} \le x_{i2} \le \text{CV}_{2\,\text{max}})$ or $(!t_2)$ and $\cdots$ and

$(\text{CV}_{n\,\text{min}} \le x_{in} \le \text{CV}_{n\,\text{max}})$ or $(!t_n)$,

then the firm will bankrupt.

The omission of feature selection increases the parameter space by adding 20 logical variables $\mathbf{t}$ and increasing the number of antecedent element CV from 5 to 20, so the optimization algorithms become unstable, time-consuming, and easy to fall stagnancy. We run the proposed FSCGACA with and without feature selection for 100 times, respectively. The averaged classification accuracy of FSCGACA without feature selection is only 19.2%, compared to the result with feature selection as 92.1%. Therefore, feature selection is essential and effective.

7.6. Comparison with Neural Network. In this section, we use the FNN with structure {5-4-1} to predict bankruptcy. The number of input neurons is set as 5 because only 5 features are selected (Table 3). The number of hidden neurons is chosen by the Bayesian probability method [35]. The number of output neurons is set as 1 because this is a binary classification problem, of which only one output neuron is able to indicate bankruptcy or not. The Levenberg-Marquardt method [36] was used to train the neural network. The convergence plot of the neural network is shown in Figure 9. We found that at the 5th epoch, the validation error begin to increase, which is an indication of overfitting.

After training, all data are submitted to the FNN. The total misclassification error is 7.4%, better than our method as 7.9%. A question "why do not use FNN?" is raised. The answer is stated in the introduction as the FNN does not have explicit physical meanings due to the nonlinear interaction between input layer and hidden layer. Although the performance of the proposed rule-based method is a bit lower, it can provide detailed physical model for economists to analyze (Table 5).

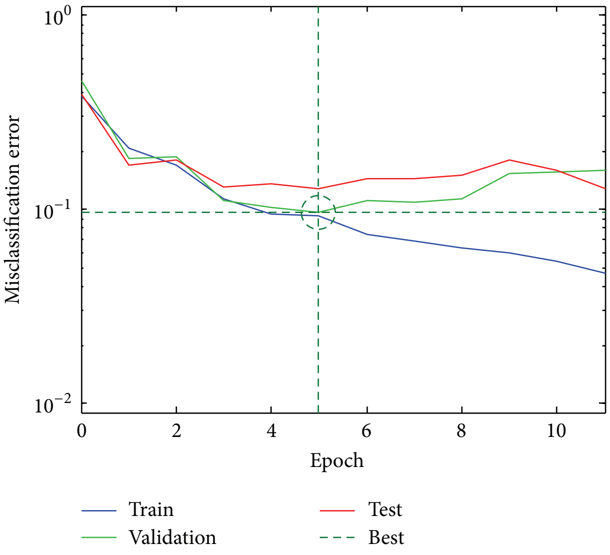


Figure 9: Convergence plot of FNN.

8. Conclusion

In this paper, we proposed a novel system to predict corporate bankruptcy. The procedure consists of four stages: first, sequential forward selection was used to extract the 5 most important features out of 20; second, we use a rule-based model to approximate the given dataset; third, the proposed FSCGACA algorithm was used to find optimal parameters of the model; and fourth, the cross-validation technique was employed to prevent overfitting.

The importance of feature selection was demonstrated in the experiments and discussions Section. If we omit feature selection, the classification accuracy of final system will decline dramatically from 92.1% to 19.2%. The proposed FSCGACA algorithm shows superiority to GA, ACA, and GACA, because it regulates the tradeoff between exploration and exploitation in a remarkably efficient way (Figure 6). Experiment results also demonstrate that the rule-based model is effective and the total misclassification can reach as low as only 7.9%. Although the FNN can reach a lower misclassification error of 7.4%, our proposed method has clearer physical meaning (Table 5).

The future work will focus on the following aspects. (1) Employing SFS method as feature selection method. We will compare SFS with other excellent feature selection methods. (2) Comparing FSCGACA with other swarm intelligence algorithms, including firefly algorithm [37], particle swarm optimization [38], and genetic pattern research [39]. (3) Researching the fundamental and basic principles of FSCGACA with the help of mathematicians. (4) Seeking economic explanation of our method with the help of economists. (5) Applying the proposed FSCGACA to other industrial and academic fields.

Conflict of Interests

The authors declare that they do not have any commercial or associative interests that represents a conflict of interest in connection with the work submitted.

References

[1] A. Cielen, L. Peeters, and K. Vanhoof, "Bankruptcy prediction using a data envelopment analysis," *European Journal of Operational Research*, vol. 154, no. 2, pp. 526–532, 2004.

[2] J. H. Min and C. Jeong, "A binary classification method for bankruptcy prediction," *Expert Systems with Applications*, vol. 36, no. 3, part 1, pp. 5256–5263, 2009.

[3] Y. Zhang and L. Wu, "Bankruptcy prediction by genetic ant colony algorithm," *Advanced Materials Research*, vol. 186, pp. 459–463, 2011.

[4] L. Nanni and A. Lumini, "An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring," *Expert Systems with Applications*, vol. 36, no. 2, part 2, pp. 3028–3033, 2009.

[5] V. Agarwal and R. Taffler, "Comparing the performance of market-based and accounting-based bankruptcy prediction models," *Journal of Banking and Finance*, vol. 32, no. 8, pp. 1541–1551, 2008.

[6] H. Etemadi, A. A. A. Rostamy, and H. F. Dehkordi, "A genetic programming model for bankruptcy prediction: empirical evidence from Iran," *Expert Systems with Applications*, vol. 36, no. 2, part 2, pp. 3199–3207, 2009.

[7] F. M. Tseng and Y. C. Hu, "Comparing four bankruptcy prediction models: logit, quadratic interval logit, neural and fuzzy neural networks," *Expert Systems with Applications*, vol. 37, no. 3, pp. 1846–1853, 2010.

[8] C. H. Wu, G. Tzeng, Y. Goo, and W. Fang, "A real-valued genetic algorithm to optimize the parameters of support vector machine for predicting bankruptcy," *Expert Systems with Applications*, vol. 32, no. 2, pp. 397–408, 2007.

[9] N. Chauhan, V. Ravi, and D. K. Chandra, "Differential evolution trained wavelet neural networks: application to bankruptcy prediction in banks," *Expert Systems with Applications*, vol. 36, no. 4, pp. 7659–7665, 2009.

[10] M. H. Nguyen and F. de la Torre, "Optimal feature selection for support vector machines," *Pattern Recognition*, vol. 43, no. 3, pp. 584–591, 2010.

[11] I. O. Oduntan, M. Toulouse, R. Baumgartner, C. Bowman, R. Somorjai, and T. G. Crainic, "A multilevel tabu search algorithm for the feature selection problem in biomedical data," *Computers & Mathematics with Applications*, vol. 55, no. 5, pp. 1019–1033, 2008.

[12] S. M. Vieira, J. M. C. Sousa, and T. A. Runkler, "Two cooperative ant colonies for feature selection using fuzzy models," *Expert Systems with Applications*, vol. 37, no. 4, pp. 2714–2723, 2010.

[13] Y. Zhang, Y. Jun, G. Wei, and L. Wu, "Find multi-objective paths in stochastic networks via chaotic immune PSO," *Expert Systems with Applications*, vol. 37, no. 3, pp. 1911–1919, 2010.

[14] Y. Zhang and L. Wu, "Classification of fruits using computer vision and a multiclass support vector machine," *Sensors*, vol. 12, no. 9, pp. 12489–12505, 2012.

[15] C. J. Kratochvil, T. E. Wilens, L. L. Greenhill et al., "Effects of long-term atomoxetine treatment for young children with attention-deficit/hyperactivity disorder," *Journal of the American Academy of Child and Adolescent Psychiatry*, vol. 45, no. 8, pp. 919–927, 2006.

[16] S. Satchell and W. Xia, "Analytic models of the ROC curve: applications to credit rating model validation," in *The Analytics of Risk Model Validation*, C. George and S. Stephen, Eds., pp. 113–133, Academic Press, Burlington, Mass, USA, 2008.

[17] J. Dréo and P. Siarry, "An ant colony algorithm aimed at dynamic continuous optimization," *Applied Mathematics and Computation*, vol. 181, no. 1, pp. 457–467, 2006.

[18] Y. D. Zhang and L. N. Wu, "A genetic ant colony classifier," in *Proceedings of the WRI World Congress on Computer Science and Information Engineering (CSIE '09)*, pp. 744–748, Los Angeles, Calif, USA, April 2009.

[19] E. R. Tkaczyk, K. Mauring, A. H. Tkaczyk et al., "Control of the blue fluorescent protein with advanced evolutionary pulse shaping," *Biochemical and Biophysical Research Communications*, vol. 376, no. 4, pp. 733–737, 2008.

[20] Y. Zhang, L. Wu, and S. Wang, "UCAV path planning based on FSCABC," *Information*, vol. 14, no. 3, pp. 687–692, 2011.

[21] A. M. Korsunsky and A. Constantinescu, "Work of indentation approach to the analysis of hardness and modulus of thin coatings," *Materials Science and Engineering A*, vol. 423, no. 1-2, pp. 28–35, 2006.

[22] Y. Wang, B. Li, and T. Weise, "Estimation of distribution and differential evolution cooperation for large scale economic load dispatch optimization of power systems," *Information Sciences*, vol. 180, no. 12, pp. 2405–2420, 2010.

[23] B. Peng, B. Liu, F. Zhang, and L. Wang, "Differential evolution algorithm-based parameter estimation for chaotic systems," *Chaos, Solitons and Fractals*, vol. 39, no. 5, pp. 2110–2118, 2009.

[24] Y. Zhang, L. Wu, and S. Wang, "UCAV path planning by fitness-scaling adaptive chaotic particle swarm optimization," *Mathematical Problems in Engineering*, vol. 2013, Article ID 705238, 9 pages, 2013.

[25] S. Nakariyakul and D. P. Casasent, "An improvement on floating search algorithms for feature subset selection," *Pattern Recognition*, vol. 42, no. 9, pp. 1932–1940, 2009.

[26] Y. Peng, Z. Wu, and J. Jiang, "A novel feature selection approach for biomedical data classification," *Journal of Biomedical Informatics*, vol. 43, no. 1, pp. 15–23, 2010.

[27] Ö. Uncu and I. B. Türkşen, "A novel feature selection approach: combining feature wrappers and filters," *Information Sciences*, vol. 177, no. 2, pp. 449–466, 2007.

[28] S. C. Yusta, "Different metaheuristic strategies to solve the feature selection problem," *Pattern Recognition Letters*, vol. 30, no. 5, pp. 525–534, 2009.

[29] E. Avci, "Selecting of the optimal feature subset and kernel parameters in digital modulation classification by using hybrid genetic algorithm-support vector machines: HGASVM," *Expert Systems with Applications*, vol. 36, no. 2, part 1, pp. 1391–1402, 2009.

[30] A. C. Pereira, M. S. Reis, P. M. Saraiva, and J. C. Marques, "Madeira wine ageing prediction based on different analytical techniques: UV-vis, GC-MS, HPLC-DAD," *Chemometrics and Intelligent Laboratory Systems*, vol. 105, no. 1, pp. 43–55, 2011.

[31] R. J. May, H. R. Maier, and G. C. Dandy, "Data splitting for artificial neural networks using SOM-based stratified sampling," *Neural Networks*, vol. 23, no. 2, pp. 283–294, 2010.

[32] S. Armand, E. Watelain, E. Roux, M. Mercier, and F. Lepoutre, "Linking clinical measurements and kinematic gait patterns of toe-walking using fuzzy decision trees," *Gait and Posture*, vol. 25, no. 3, pp. 475–484, 2007.

[33] J. H. Min and Y. C. Lee, "Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters," *Expert Systems with Applications*, vol. 28, no. 4, pp. 603–614, 2005.

[34] H. Ahn and K. J. Kim, "Bankruptcy prediction modeling with hybrid case-based reasoning and genetic algorithms approach," *Applied Soft Computing Journal*, vol. 9, no. 2, pp. 599–607, 2009.

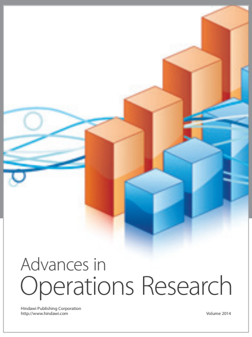
[35] K. V. Yuen and H. F. Lam, "On the complexity of artificial neural networks for smart structures monitoring," *Engineering Structures*, vol. 28, no. 7, pp. 977–984, 2006.

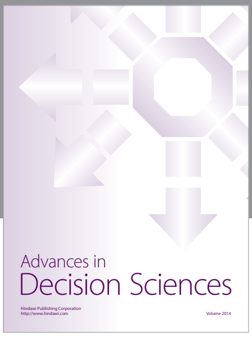
[36] H. Temurtas, N. Yumusak, and F. Temurtas, "A comparative study on diabetes disease diagnosis using neural networks," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8610–8615, 2009.

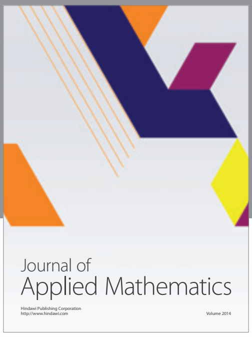
[37] Y. Zhang, L. Wu, and S. Wang, "Solving two-dimensional HP model by firefly algorithm and simplified energy function," *Mathematical Problems in Engineering*, vol. 2013, Article ID 398141, 9 pages, 2013.

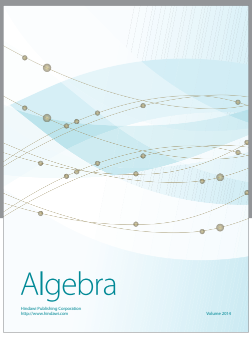
[38] Y. Zhang, S. Wang, G. Ji, and Z. Dong, "An MR brain images classifier system via particle swarm optimization and Kernel support vector machine," *The Scientific World Journal*, vol. 2013, Article ID 130134, 9 pages, 2013.

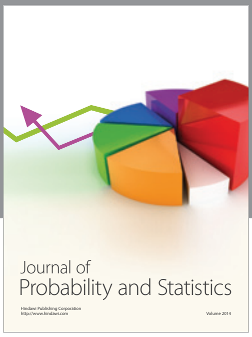
[39] Y. Zhang, S. Wang, G. Ji, and Z. Dong, "Genetic pattern search and its application to brain image classification," *Mathematical Problems in Engineering*, vol. 2013, Article ID 580876, 8 pages, 2013.


Advances in
Operations Research

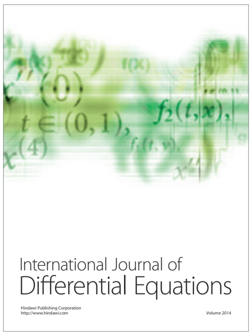

Advances in
Decision Sciences


Journal of
Applied Mathematics

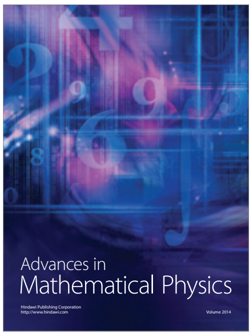

Algebra

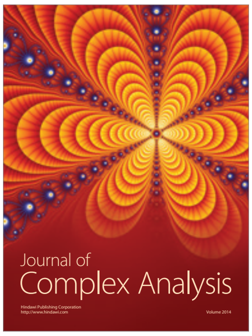

Journal of
Probability and Statistics

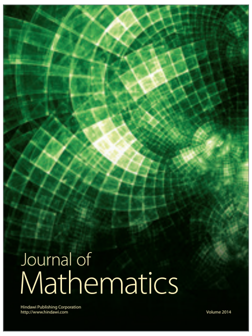
The Scientific
World Journal

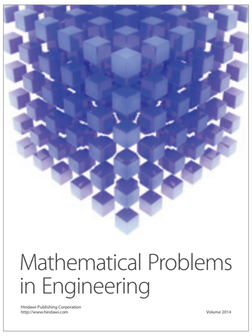

International Journal of
Differential Equations

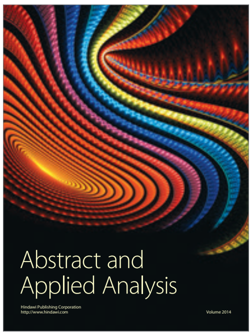
International Journal of
Combinatorics

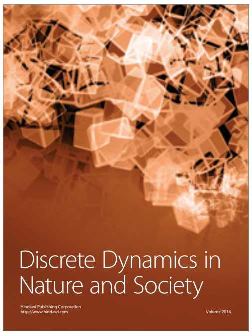

Advances in
Mathematical Physics

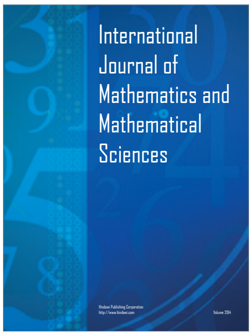

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

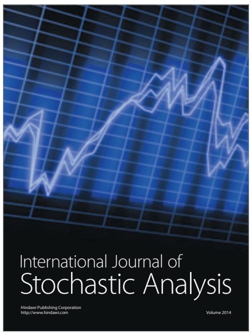

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization