

Research Article

Credit Risk Prediction Using Fuzzy Immune Learning

Ehsan Kamaloo and Mohammad Saniee Abadeh

Faculty of Electrical and Computer Engineering, Tarbiat Modares University, Tehran 14115-143, Iran

Correspondence should be addressed to Mohammad Saniee Abadeh; saniee@modares.ac.ir

Received 5 February 2014; Revised 1 June 2014; Accepted 6 June 2014; Published 24 June 2014

Academic Editor: M. Onder Efe

Copyright © 2014 E. Kamaloo and M. Saniee Abadeh. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The use of credit has grown considerably in recent years. Banks and financial institutions confront credit risks to conduct their business. Good management of these risks is a key factor to increase profitability. Therefore, every bank needs to predict the credit risks of its customers. Credit risk prediction has been widely studied in the field of data mining as a classification problem. This paper proposes a new classifier using immune principles and fuzzy rules to predict quality factors of individuals in banks. The proposed model is combined with fuzzy pattern classification to extract accurate fuzzy if-then rules. In our proposed model, we have used immune memory to remember good B cells during the cloning process. We have designed two forms of memory: simple memory and k -layer memory. Two real world credit data sets in UCI machine learning repository are selected as experimental data to show the accuracy of the proposed classifier. We compare the performance of our immune-based learning system with results obtained by several well-known classifiers. Results indicate that the proposed immune-based classification system is accurate in detecting credit risks.

1. Introduction

Banks and financial agencies employ credit scoring models extensively to determine good and bad credits. Loans are usually the most significant cause of risk in banks. Using credit scoring will reduce the time of loan approval procedure [1] and save cost per loan and enhance credit decisions. This enhancement helps lenders to guarantee that they are applying the same criteria to same groups of borrowers [2]. In these situations banks can supervise the existing loans much easier than before [3]. Because of the fast growth of autofinancing in the last two decades, the use of data mining for credit risk prediction increases rapidly [4–7]. The first investigation into credit scoring was started by Olson and Wu in 2010 to classify credit applications as good or bad payers [8]. Fair and Isaac presented a credit scoring model in the early 60s [9]. Since then, various models have been developed using traditional statistical methods such as discriminant analysis method in [10, 11]. Ordinary linear regression has also been used as another traditional statistic method for credit scoring [12, 13]. Recent techniques of credit risk assessment [14–20] treat

lending decision problem as a binary classification problem [8].

The performance of bioinspired algorithms, like artificial neural networks and evolutionary computation, for various data mining problems has been demonstrated by many investigations previously [21–25]. Many bioinspired algorithms have been proposed for credit scoring [1, 21, 26]. Recently artificial immune systems (AIS) have been successfully employed in a wide variety of application areas. Artificial immune systems are computational systems inspired by the processes of the natural immune system. This metaheuristic emerged in the 90s as a new computational model in AI. Hunt and Cooke apply AIS to pattern recognition problems in 1996 [27]. Timmis and Knight define AIS as “adaptive systems inspired by theoretical immunology and observed immune functions, principles and models, which are applied to problem solving” [28]. There are various types of AIS, and researchers worked mostly on the theories of immune networks, clonal selection, and negative selection [29]. In this paper, we have proposed an AIS-based classification system with a new clonal selection

algorithm. Within the proposed AIS, fuzzy logic has been applied to extract interpretable fuzzy rules [30, 31].

The main reason that encouraged us to use the AIS metaheuristic for credit risk prediction problem is that AIS has a nature which we can use it for our problem effectively. This nature is that AIS tends to explore the search space of the problem very efficiently. This capability is associated with the hypermutation operator of AIS. We have selected AIS for credit scoring prediction problem because previous investigations show that the so-called classification problem has a very explorative search space. We have experienced this nature of credit scoring classification problem in our experiments vividly. The main observation which shows the explorative nature of this problem is that the fitness function outputs changes drastically for very similar inputs. Moreover, AIS has proved its high performance for two-class classification problems in previous investigations.

The new proposed classification system in this paper is an improved version of fuzzy artificial immune system (FAIS) [30] and comprehensible credit scoring- FAIS (CCS-FAIS) [31] classifiers as the two previous versions of AIS-based classification system for credit risk prediction. In our proposed model, we have employed immune memory to remember good B-cells during the cloning process. We have designed two forms of memory: simple memory and k -layer memory. Results demonstrate that our new definition of memory for AIS-based fuzzy rule extraction increases the final classification rate of credit scoring process considerably. The Weka Data Mining tool [32] has been used to compare our classifier with several well-known classifiers.

The rest of this paper is organized as follows. Section 2 discusses some algorithms presented for credit risk prediction problem. In Section 3, we describe immune systems and the concepts we have used in our proposed algorithm. Section 4 describes pattern classification with fuzzy logic. The proposed algorithm is presented in Section 5. Section 6 provides information of performed experiments and achieved results. Finally, Section 7 concludes the paper.

2. Literature Review

SVM is one of the popular learning methods presented for credit scoring classification problem. Choosing the optimal input feature subset and setting the best kernel parameters are the two problems that should be solved to propose an efficient SVM-based classifier [33]. Zhang et al. [3] and Huang et al. [33] used SVM for credit scoring. They show that SVM has a high and acceptable accuracy for this classification problem.

Hybrid data mining approaches also have been proposed for effective credit scoring. Yao [34] used neighborhood rough set and SVM as a hybrid classifier. In this classifier a neighborhood rough set has been employed for feature selection. Zhang et al. [35] proposed hybrid model based on genetic programming (GP) and SVM. This model used GP to extract if-then rules and for remaining instances of dataset it employed discriminator based on SVM. Yi [36] used a combination of decision tree and simulated annealing methods to build a model. In this hybridization, authors have combined

local search strategy of decision tree algorithms and global optimization of simulated annealing algorithm.

Exploring new techniques in credit scoring performance improvement can save too much money. In recent years, many bioinspired algorithms are presented for solving classification problems such as credit card fraud detection [37], credit scoring, security, and other applications [38]. Among these approaches, AIS is one of the newest methods that has been applied for the credit scoring purposes. Leung et al. [9] proposed a simple AIS (SAIS) algorithm that adopted few key concepts of AIS (affinity measure, cloning, and mutation). They found SAIS a very competitive classifier.

Fuzzy logic has been used for designing classification systems drastically [39, 40]. The important advantage of fuzzy logic is its influential capability in managing uncertainty and vagueness [41]. Most of fuzzy classifiers generate a list of fuzzy if-then rules. These rules are represented in linguistic forms that make them interpretable by users. Experts can validate and correct the rules. This increases the interaction with users. Lei and Ren-hou [42] proposed a classifier based on immune principles and fuzzy rules. They apply their algorithm on 15 well-known UCI machine learning repository [43] data sets and achieve high accuracy. The fuzzy AIS term in our proposed method is similar to Lei and Ren-hou method. The major difference between our proposed method and Lei and Ren-hou algorithm is in the definition of fitness function. They used a simple function as fitness function, but we have improved fitness function by extra terms. This function has been discussed in Section 5 in detail.

3. Immune Systems

Researchers have been inspired by biology in solving computational problems. There have been several techniques on biological metaphors such as evolutionary algorithms, swarm intelligence, and neural networks [29]. Artificial immune systems are bioinspired algorithms that have been active and prolific over the last decade [44, 45]. The basis of AIS is human immune system which exploits learning and memorizing capabilities of immune systems [46]. Timmis et al. proposed the relation of immunology and computation [29] at the 80s. Hunt and Cooke investigated the nature of learning in the immune system and proposed a learning algorithm [27]. Timmis and Knight [28], de Castro and von Zuben [47], and Dasgupta [48] developed basic models of artificial immune systems which are the sources of current AIS algorithms. Immune-inspired models have been applied on wide variety of research fields ranging from pattern recognition, such as classification and clustering, anomaly detection [49, 50], and optimization [51, 52], to robotics [53, 54] and image processing [55, 56]. The key characteristics of immune systems are learning, adaptability, memory mechanisms, and self-organization which are desirable to inspired algorithms. Clonal selection, immune networks, and negative selection are the three main immunological theories that are employed as strongly accepted perspectives in AIS.

Attribute 1	Attribute 2	Attribute 3	Attribute 4	Attribute 5	...	Attribute n
S	DC	L	DC	DC	...	ML

\Rightarrow Class is c

FIGURE 1: A sample fuzzy rule which is “if Attribute 1 is Small and Attribute 3 is Large...and Attribute n is Medium Large then class is c .”

3.1. Natural Immune System. The human immune system is a distributed pattern detection system with many functional components located in specific parts throughout the body. The immune system controls defense mechanism through innate and adaptive responses [57]. Innate responses are against any invaders that enter the body, but adaptive responses are directed against particular invaders and demonstrate learning, recognition, memory acquisition, and self-regulations of the body. These invaders that infect the body are called antigens. Antigens provoke the immune responses. The core of adaptive responses is lymphocytes which are provided with a sort of receptors to recognize antigens. Lymphocytes are divided into two types as B cells and T cells. In the case of invasion, appropriate B cells attempt to clone with producing sufficient proteins to remove antigens (called antibodies). A B cell holds antibodies on its shell which can identify the antigens invading the body. The matching between antigen and antibody is complementary and is similar to “lock and key” [58]. T cells do not interact with antigens directly. They circulate through the body and scan the surface of body cells for the presence of foreign antigens that have been combined with the cell. Then, T cells bind to these cells and become activated. Activated T cells secrete some chemicals as alert signals to others. B cells which take these signals from the T cells become stimulated with the detection of antigen by their antibodies.

3.2. Clonal Selection Theory. The clonal selection theory describes the basic response of the adaptive immune system to an antigenic stimulus. The idea is that only those cells that are capable of detecting the antigen will proliferate and others cannot clone. This theory applies for both T cells and B cells. Before the receptor of B cells binds to an antigen and B cells become stimulated and differentiate into memory cells, colonies of B cells are created. During the cloning process, B cells undergo somatic hypermutation which keeps the diversity of B cell population for future strange antigens. After cloning, activated B cells (or memory cells) produce huge amounts of antibodies which results in elimination of the antigen. Some of memory cells remain within the host to generate a rapid response upon a subsequent encounter with the same or similar antigen [29]. CLONALG [59] and B cell algorithm [60] are AIS algorithms which are based on clonal selection theory. These algorithms have cloning, mutation, and selection operators which makes them similar to genetic algorithms.

4. Fuzzy Rule-Based Pattern Classification

In this section, we briefly explain the fuzzy rule-based pattern classification method which was first proposed by Ishibuchi

et al. [61] and used in many investigations [30, 31, 42, 62–65]. This method consists of fuzzy rule generation and fuzzy reasoning procedures.

4.1. Fuzzy Rule Generation. Let us assume that the pattern space is n -dimension continuous space with c classes. For simplicity, each dimension must be in the unit interval $[0, 1]$. The training data set includes m labeled patterns, which is shown in

$$X_p = (x_{p1}, x_{p2}, \dots, x_{pn}), \quad \text{class is } c_K \quad (1)$$

$$p = 1, 2, \dots, m. \quad K = 1, 2, \dots, c.$$

The purpose is generating fuzzy if-then rules with the following form: Rule R_i : if x_{p1} is A_{i1} and...and x_{pn} is A_{in} then X_p belongs to Class C_i with $CF = CF_i$, where R_i is the label of the i th fuzzy if-then rule, A_{i1}, \dots, A_{in} are antecedent fuzzy sets in the unit interval $[0, 1]$, C_i is the resultant class, and CF_i is the certainty factor (or rule weight) of the fuzzy if-then rule R_i which is a real number in the unit interval $[0, 1]$ (Figure 1 demonstrates a sample fuzzy if-then rule). There might have been some *do not care* antecedents in the rules and these antecedents are usually omitted. Therefore, the number of antecedents of a rule is less than or equal to n . Some rules may have a few antecedent conditions which makes them more understandable to users.

We have used a typical set of linguistic values as antecedent fuzzy sets. The membership function of each linguistic value is obtained by homogeneously partitioning the domain of each attribute into symmetric triangular fuzzy sets ($f_{\text{membership}}$ in (2)). We use such simple specification in experiments to demonstrate the high performance of our fuzzy classifier system, even if the membership function of each antecedent fuzzy set is not tailored. However, we can use any tailored membership function in our fuzzy classifier system for a particular pattern classification problem. Consider

$$f_{\text{membership}}(x) = \begin{cases} -4x + 1 & 0 \leq x < 0.125 \implies S \\ 4x & 0.125 \leq x < 0.25 \implies MS \\ -4x + 2 & 0.25 \leq x < 0.375 \implies MS \\ 4x - 1 & 0.375 \leq x < 0.5 \implies M \\ -4x + 3 & 0.5 \leq x < 0.625 \implies M \\ 4x - 2 & 0.625 \leq x < 0.75 \implies ML \\ -4x + 4 & 0.75 \leq x < 0.875 \implies ML \\ 4x - 3 & 0.875 \leq x \leq 1 \implies L \end{cases} \quad (2)$$

$$f_{\text{membership}}(\text{DC}) = 1, \quad \forall x \in [0, 1],$$

where S, MS, M, ML, L, and DC, respectively, stand for small, medium small, medium, medium large, large, and do

Step 1. **for each** pattern X_p **do**
 (1.1) Compatibility: $\mu_i(X_p) = \prod_{j=1}^n f_{\text{membership}}(x_{pj})$
 Step 2. **for each class** h **do**
 (2.1) Calculate relative sum of compatibility grades: $\beta_h(R_i) = (\sum_{X_p \in h} \mu_i(X_p))/N_h$
 Step 3. Find class \hat{h} which has maximum $\beta_h(R_i)$.
 Step 4. $CF_i = (\beta_{\hat{h}}(R_i) - \bar{\beta})/(\sum_{h=1}^c \beta_h(R_i))$ where $\bar{\beta} = (\sum_{h=1, h \neq \hat{h}}^c \beta_h(R_i))/(c-1)$

PSEUDOCODE 1: Pseudocode for calculating grade of certainty CF for R_i .

not care. The grade of certainty (CF) for each fuzzy rule is determined by (Pseudocode 1). Compatibility (μ) of a pattern to a rule is the product of membership amount of pattern at each dimension. Zero compatibility of a pattern to a rule means that the rule has not covered the pattern. After calculating μ for each pattern, the relative sum of compatibilities (β) is calculated per class and finally the certainty factor is achieved by relative difference of maximum value of β and sum of other β s.

4.2. Fuzzy Reasoning. When the antecedent fuzzy sets of each rule are given, we can determine consequent class and the grade of certainty of each rule by fuzzy rule generation method which has been described in previous section in detail. The proposed classifier in this paper generates a set of fuzzy if-rules. The achieved rule set is then employed to predict unknown instances. The fuzzy reasoning procedure ensures us which rules can vote for class of the test instance. We use single winner rule in our algorithm. Let us assume that we have a set of fuzzy rules S extracted from training data set. The input pattern $Y_p = (y_{p1}, y_{p2}, \dots, y_{pn})$ is classified by a single winner rule R_M in S which is determined as follows:

$$\mu_M(y_p) \cdot CF_M = \max \{ \mu_i(y_p) \cdot CF_i \mid R_i \in S \}. \quad (3)$$

Product of compatibility grade of input test instance and grade of certainty for the winner rule has the most value in the rule set.

5. Research Procedure

This section presents the proposed algorithm and discusses about each of its steps in detail. Comprehensible credit scoring-FAIS (CCS-FAIS) [31] and fuzzy artificial immune system (FAIS) [30] are two fuzzy classifiers that we have proposed earlier using immune principles. These classifiers were based on the clonal selection theory. The clonal selection principle is used to describe the main features of an adaptive immune response to an antigenic stimulus. The main idea is that only those B cells that identify the antigens are selected to proliferate. The selected cells are exposed to an affinity maturation process, which develops their affinity to the selective antigens. In this paper, no distinction is made between a B cell and its antibody; therefore, each individual in our immune model will be called B cell.

Our previous FAIS and CCS-FAIS classification systems used population of B cells. In these classifiers, each B cell had

primary age to live in the population. Age of B cells should be increased if their fitness had been improved during the maturation process; otherwise, those B cells that their current ages reach to their corresponding maximum age thresholds would die.

In this paper, we have improved the performance of FAIS and CCS-FAIS classifiers. The differences of IFAIS (current paper method) with CCS-FAIS and FAIS are as follows.

- (1) In our proposed model, we have employed immune memory to remember good B cells during the cloning process.
- (2) We have designed two forms of memory to remember good B cells during the cloning process: simple memory and k -layer memory.
- (3) The IFAIS benefits from using several diverse selection procedures to develop an efficient clonal selection algorithm.

The goal of the immune model is to obtain a set of rules with high accuracy. Each B cell represents a rule. As we mentioned in Section 3, each rule is coded according to Figure 1.

5.1. Affinity Functions. Equation (4) demonstrates the used affinity functions which have been previously presented in CCS-FAIS and FAIS [30, 31]. Consider

$$f_P(R_i) = \frac{\sum_{p=1|c^p=c_i}^K w^p \cdot \mu_i(x^p)}{\sum_{p=1|c^p=c_i}^K w^p},$$

$$f_N(R_i) = \frac{\sum_{p=1|c^p \neq c_i}^K w^p \cdot \mu_i(x^p)}{\sum_{p=1|c^p \neq c_i}^K w^p},$$

$$\text{fitness}^1(R_j) = w_P \cdot f_P(R_j) - w_N \cdot f_N(R_j).$$

$$\begin{aligned} \text{fitness}^2(R_j) &= w_P \cdot f_P(R_j) + w_{\text{NCP}} \cdot \text{NCP}_{\text{normal}}(R_j) \\ &\quad - w_N \cdot f_N(R_j) - w_{\text{NMP}} \cdot \text{NMP}_{\text{normal}}(R_j), \end{aligned}$$

$$\text{fitness}^3(R_j) = w_{\text{BF}} \cdot \text{fitness}^2(R_j) - w_{\text{LEN}} \cdot \text{length}(R_j). \quad (4)$$

5.2. Immune Memory. The memory cells in natural immune system are used for eliminating similar foreign substances. In this paper, we have employed immune memory during the cloning process for selected B cells. According to the cloning

```

procedure IFAIS
begin
  Initialization(); // a population of B cells is generated
  Rule Generation(); // a population of B cells searches for optimized rule iteratively
  Rule Learning(); // From the final population, the best B cell based on fitness is selected.
  Termination Test // If a stopping condition is satisfied, the learning of current class is finished and
  the algorithm is going to learn the next class.
End

```

PSEUDOCODE 2: Pseudocode of IFAIS.

```

(1) procedure Proposed Classifier
(2) do
(3)   Set current learning class as  $c$ ;
(4)   While Termination Test
(5)     Generate initial B-cell repertoire from class  $c$  antigens;
(6)     While cycle < Max.Iterations {
(7)       Perform Clonal Selection Procedure;
(8)       // Use three selection procedures as:
(9)       //(1) Roulette Wheel Selection,
(10)      //(2) Tournament Selection,
(11)      //(3) Uniform Selection.
(12)      Use memory (Simple and  $k$ -layered to clone selected B-Cells;
(13)      Perform Hyper-mutation;
(14)     }
(15)     Perform Rule Learning Procedure;
(16)     //(1) Select the best B cell
(17)     //(2) Add rule of the best B cell to the current rule set
(18)     If classification rate is not increased, then the current loop exits.
(19)   }
(20) Until All classes have been learned

```

ALGORITHM 1: An overview of the proposed classifier. At initialization, a population of B-cells is generated from instances of class c . then some B-cells are selected to proliferate in rule generation phase. The life cycle of B-cells is controlled by age. The best B-cell is added to rule set if the classification rate increases more than a threshold. At last, if the termination test satisfies the classifier learns rules for class c .

method, a B cell is changed randomly. Randomness of the modification is a way of exploring in the search space. The balance of exploration and exploitation is a major problem in heuristic search algorithms. In order to exploit the previous knowledge of cloning, the memory records the changes of B cells, which enables the algorithm to produce higher quality B cells. The cloning method with this kind of memory increases the probability of modifications, which have been recorded in memory in former iterations of algorithm. We called this type of memory *simple memory*. In each iteration, the contents of memory degrade slightly. The effectiveness of memory decreases gradually using the proliferation procedure. When the generation of high quality B cells using the memory is stopped, the number of biased memory-based changes decreases accordingly.

During the cloning process, it might be more effective if we consider more than one modification for the selected B cell. In a simple memory, all changes are recorded independently; therefore, we define a new type of memory which is named *k-layer memory*. In this memory type, k is the maximum number of simultaneous changes on a B cell. For

example a 3-layer memory contains 3 kinds of memories. The first memory records just 1 modification, the second memory records 2 simultaneous modifications, and the last memory records 3 simultaneous modifications. A k -layer memory needs a large amount of physical memory to run efficiently; therefore, it is not a useful method for huge data sets. The detailed implementation of these memories has been explained in the next section.

5.3. Proposed Classifier. An overview of the proposed classifier is presented in Pseudocode 2 and Algorithm 1. The main loop of the algorithm applies the learning procedure for each class separately. This loop consists of 4 steps: initialization, rule generation, rule learning, and termination test. Rule generation phase employs an AIS-based algorithm to find a single rule, based on the initiated population. In the rule learning stage, when a rule is added to the final learned rule set, the learning mechanism reduces the weight of those training instances that are covered by the new learned rule. Therefore, in the next rule generation round, the AIS-based

rule induction procedure focuses on those instances that are currently uncovered or misclassified. At the beginning of the learning process, the weights of the whole training instances are set to 1. Each step of the new proposed AIS-based algorithm is described briefly in Pseudocode 2. The details of our algorithm is presented in Algorithm 1 (IFAIS stands for improved FAIS).

(1) *Initialization.* In this stage, a population of B cells is generated. The number of initial population is constant. This number is a parameter which is named *initial Population Size*. To generate a B cell, an instance of current class from data set is selected randomly and fuzzy terms for antecedent part of the rule (B cell) are computed according to each attribute value of the selected training instance. Consequent part of the generated rule becomes the class of selected instance. Initial age of B cell is another parameter denoted by *default Age*. After generation of initial population, fitness is computed for each B cell independently.

(2) *Rule Generation.* In this step, a population of B cells searches for optimized rule iteratively. At the first step of IFAIS, some B cells are selected to be cloned. This selection is based on roulette-wheel selection algorithm. B cells with higher fitness have more chance to be selected. The number of selected B cells is constant (*selection Size*). Now it is time to proliferate the selected B cells. Hypermutation occurred during the cloning process. A B cell contains a rule and the rule has antecedents. Hypermutation considers a change to these antecedents which causes a change to the corresponding B cell. Maximum number of simultaneous changes in antecedents of a B cell would be determined by a parameter which is named *max Term Changes Number*. We need to restrict the number of changes because increasing the number of modified antecedents of a rule increases the probability of corruption of that rule significantly. Now a random number is generated to determine the number of changes to the selected B cell (*max value is max Term Changes Number*), after that the algorithm determines which antecedents must be changed using immune memory. In this algorithm, simple memory is presented by a matrix. Rows are fuzzy terms, columns are attributes, and entry (i, j) is the value of changing j th attribute to i th fuzzy term, so the probability of choosing j th attribute is sum of j th column entries divided by sum of all entries and the probability of changing to a fuzzy term is proportional to the value of each fuzzy term. The determination of correct factor which would be able to reveal the progress of a change is critical. In this algorithm, we use relative affinity (difference of new affinity and old affinity) as value of a change to a fuzzy term. If the algorithm uses k -layer memory, we should note that k is the same as *max Term Changes Number*. The k -layer memory contains k matrices. Dimension of 1-matrix is like simple memory. p -matrix is used when p simultaneous changes occurred; therefore, the number of columns would be equal to the number of attributes or C (attributes, p) and the number of rows would be the same as num of fuzzy terms ^{p} . If we do not use memory, the probability of changing an attribute value to do not care is a parameter that is called dont Care

Replacement Rate. The effect of memory controls by weight means of default probability and memory probability. This weight is a parameter which is called *memory Weight*. Number of clones produced for each B cell is another parameter which is called *clone Number*. The age of generated B cells is calculated using (5). This equation controls the population size. Consider

$$\text{Age}^{\text{new}} = \text{Age}^{\text{old}} + \text{Age}^{\text{default}} \times \text{affinity} \quad (5)$$

$$\text{if } \text{affinity}^{\text{new}} > \text{affinity}^{\text{old}}.$$

After cloning, B cells of previous generations became older. Some B cells would be deleted from the main population because their age reaches 0.

(3) *Rule Learning.* When AIS algorithm is finished, the fittest B cell is selected. The rule which is represented by this B cell is added to the final resulted rule set. Then, the classification rate of current rule set is compared to the old rule set which does not contain the new rule. Classification rate is calculated using (6). If the difference is higher than a threshold (*accuracy Threshold*), the addition is accepted. Consider

$$\text{classification rate} = \frac{\text{NCP}}{\text{number of patterns}}. \quad (6)$$

(4) *Termination Test.* If a stopping condition is satisfied, the learning of current class is finished, and the algorithm is going to learn the next class. If the condition is not satisfied, the algorithm tries to learn another rule by initializing a new population for the next execution of AIS. We can use any stopping condition for terminating the loop. We limit the number of learned rules for each class. This is done by a parameter which is called *max Rule Set Size*.

5.4. *Classification Reasoning Technique.* After rule extraction procedure, the classifier must employ these learned rules to predict the class of a test record. The usual reasoning method of fuzzy classifiers is based on (3) which is explained in Section 4 in detail. We use (3) to predict the class of an input test instance whenever all of the rules are not applicable for the input test instance. The algorithm finds the most similar rule to this instance. In this method, a rule is created from the instance like the initialization phase primary rules are generated from instances. The most similar rule is the rule which has the highest length of longest common subsequence (LCS) with the newly generated rule. The length of common subsequence of the selected rule must be greater than minimum length. This value is another parameter which is named *min Rule Similarity Length*. This method decreases the number of unclassified instances of the algorithm.

6. Experimental Results

In this section, two credit data sets were used to evaluate the predictive accuracy of the proposed classifier. Australian credit approval and German credit approval data sets are available from UCI Machine Learning Repository. In Australian credit approval data set, all names and values have

TABLE 1: UCI datasets used in our experiments. Instances of the Australian are almost equally distributed between classes, but in the German they are more unbalanced.

Dataset	Number of classes	Number of attributes	Number of instances	Classes
Australian	2	14	690	307 negative 383 positive
German	2	24	1000	700 negative 300 positive

TABLE 2: Parameter specification of IFAIS in our experiments.

Parameter	Value in Australian	Value in German
initial Population Size	100	300
max Iteration	50	50
default Age	5	5
selection Size	100	100
clone Number	10	10
max Term Changes Number	3	2
dontCare Replacement Rate	0.5	0.2
max Rule Set Size	5	10
accuracy Threshold	0.03	0.03
max Rule Similarity Length	10	17
min Covered Percent	0	30
memory Weight	0.5	0.5
w_p	0.01	0.35
w_{NCP}	0.69	0.1
w_N	0.01	0.45
w_{NMP}	0.29	0.1
w_{BF}	0.8	0.999
w_{LEN}	0.2	0.001

been changed to meaningless to protect confidentiality of the data. Table 1 illustrates the information of these data sets. In Australian credit data there are 383 instances where assigning credit to them has high risk and 307 instances are creditworthy applicants. The German credit data is more unbalanced, and it consists of 300 instances where credit should not be assigned and 700 instances are creditworthy applicants.

Each value in the Australian and German data sets is normalized between 0.0 and 1.0 using the min-max transformation method. Table 2 represents the parameter settings that have been used in IFAIS. Simulations have been performed by Weka data mining tool. Table 2 shows an interesting fact about the used datasets: the German credit scoring dataset is more complicated than Australian credit scoring dataset. This is because of the need of IFAIS to work with a greater initial population size and maximum rule set size when it is applied on the German credit scoring dataset.

In Figures 2 and 3 the progress of classification rate per rule of the proposed classifier for Australian and German credit data sets has been illustrated, respectively. These figures illustrate the role of each evolved fuzzy if-then rule for the two used datasets. Our algorithm uses iterative rule learning and it employs AIS per iteration to find a rule (rule generation

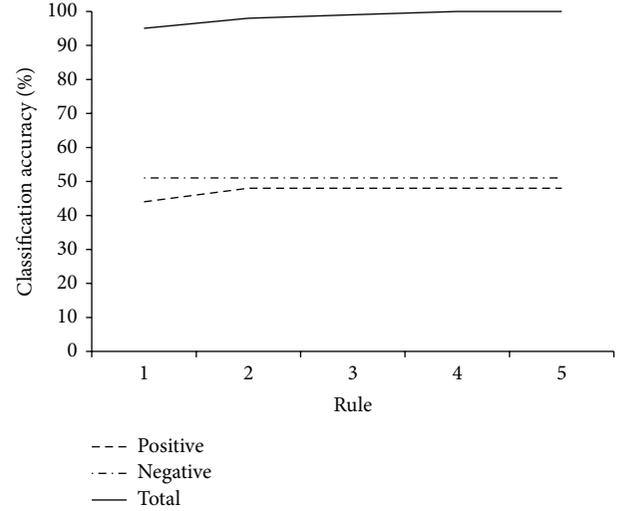


FIGURE 2: Progress of classification rate per rule of IFAIS for Australian credit data set.

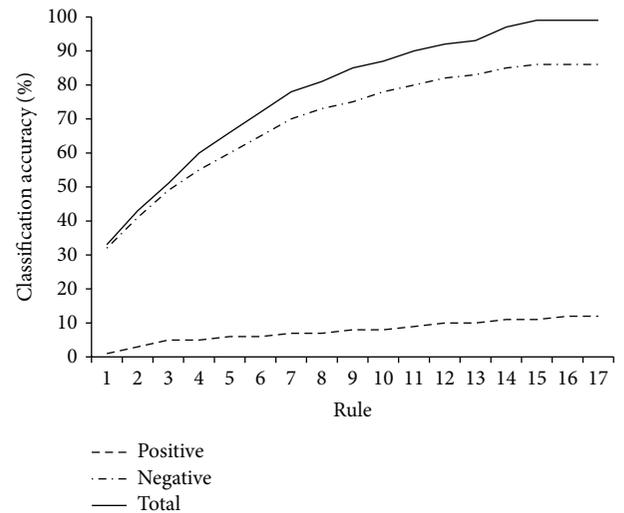


FIGURE 3: Progress of classification rate per rule of IFAIS for German credit data set.

phase). After the extraction of each rule the weights of instances that have been covered by the rule are decreased. In IFAIS, the instances are removed from data set which means the weights are set to zero. According to Figures 2 and 3, the first extracted rules are more general and shorter than later rules. All of the extracted rules participate in the decision-making process and the rules classify almost the whole test data.

According to Figures 2 and 3, we can also compare the complexity of data sets. To accomplish this, we have extracted 5 rules for each class of Australian dataset. For each class of the German dataset, we had 12 and 17 rules for negative and positive classes, respectively.

The difference in number of extracted rules shows that German data set patterns are more complex than Australian

TABLE 3: Predictive accuracy of FAIS and different classifiers.

Classifier	Australian	German
AIRS1: Weka version of [66]	83.2	68.71
AIRS2: Weka version of [66]	84.41	69.64
AIRS as conventional AIS [9]	85.2	71.3
CCS-FAIS [55]	80.7	71.1
CLONALG: Weka version of [59]	82.22	66.42
CSCA*	84.78	70.17
DMNBtext	82.58	69.99
DTNB	85.41	71.52
FAIS [30]	85.51	72
Immunos-1*	76.12	61.38
Immunos-99*	76.72	63.67
LibSVM	85.51	70.98
LWL	85.51	70
Kstar	78.88	69.89
PART	83.32	70.11
SAIS [9]	85.2	75.4
SMO with RBFKernel	85.51	70
IFAIS using simple memory	86.52	72.3
IFAIS using 3-layer memory	87.83	74.9

The results have been obtained using Weka machine learning tool.

The order of classifiers is alphabetical. The most accurate is bold and the second most is italic.

Comparison of predictive accuracies illustrated our proposed algorithm is competitive with other classifiers.

*means classifiers which have been manually added to Weka and are available at <http://weka.classalgos.sourceforge.net>.

data set (In Figure 3 the increasing rate of graph for negative class is higher than the same class in Figure 2).

The negative class in German data set has more complicated signature because in comparison to Australian data set, the later learned rules have more effects (the steep of the graph is very slow). This fact shows that achieving accurate knowledge for Australian data set is more difficult than German data set. In Australian data set the first rules are very important because final classification accuracy is nearly equal to the classification accuracy at those points.

In Table 1, we have demonstrated the distribution of instances over the two classes of Australian and German credit scoring datasets. In Australian data set this distribution is approximately equal. Figure 3 shows this fact too because the accuracy of classes are nearly the same. In German data set the number of negative class instances is more than instances of positive class. In Figure 3 we have seen the important role of negative class in final classification accuracy. The extracted rules of positive class have covered very few test records. We followed 10-fold cross-validation procedure to evaluate the accuracy of our classifier. The classification rate is measured with Weka machine learning software and compared with well-known classifiers in Weka including LibSVM, PART, DTNB, Kstar, LWL, DMNBtext, SMO with RBFKernel, and J48. In these classifiers, DTNB and PART extract rules, J48 uses decision tree, LWL and Kstar are lazy classifiers, LibSVM and SMO are two implementations of SVM, and DMNBtext

TABLE 4: Confusion matrix.

Actual	Predicted	
	Negative	Positive
Negative	TN	FP
Positive	FN	TP

TABLE 5: Comparing precision, recall, and F -measure of IFAIS and selected classifiers.

Classifier	Class	Australian		German	
		pos.	neg.	pos.	neg.
DMNBtext	Precision	0.82	0.83	0.5	0.71
	Recall	0.87	0.77	0.09	0.96
	F -measure	0.85	0.8	0.16	0.82
DTNB	Precision	<i>0.87</i>	<i>0.84</i>	<i>0.77</i>	0.53
	Recall	<i>0.87</i>	0.84	0.41	0.84
	F -measure	0.87	<i>0.84</i>	0.47	0.81
IFAIS	Precision	0.93	0.79	0.64	0.80
	Recall	0.8	0.93	<i>0.47</i>	0.89
	F -measure	<i>0.86</i>	0.85	0.54	0.84
LibSVM	Precision	0.93	0.79	0.78	0.71
	Recall	0.8	0.93	0.05	0.99
	F -measure	<i>0.86</i>	0.85	0.09	<i>0.83</i>
LWL	Precision	0.93	0.79	∞	0.7
	Recall	0.8	0.93	0.0	1.0
	F -measure	<i>0.86</i>	0.85	∞	0.82
Kstar	Precision	0.76	0.85	0.5	0.76
	Recall	0.91	0.64	0.39	0.83
	F -measure	0.83	0.73	0.44	0.79
PART	Precision	0.85	0.82	0.5	<i>0.78</i>
	Recall	0.85	0.81	0.49	0.79
	F -measure	0.85	0.81	0.5	0.79
SMO with RBFKernel	Precision	0.93	0.79	∞	0.7
	Recall	0.8	0.93	0.0	1.0
	F -measure	<i>0.86</i>	0.85	∞	0.82

The results are measured by Weka machine learning software. The order of classifiers is alphabetical. The best result is bold and the second most is italic.

uses Bayesian decision theory. Table 3 summarizes the prediction accuracies of the proposed algorithms and other classifiers. In this table, other AIS-based algorithms are also demonstrated. Other performance measures for comparing our classifier with mentioned classifiers are precision, recall, and F -measure. These measures can be obtained using (7) and according to Table 4. Consider

$$\begin{aligned} \text{recall}^{\text{NEG}} &= \frac{\text{TN}}{\text{TN} + \text{FP}} \text{recall}^{\text{POS}} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ \text{precision}^{\text{NEG}} &= \frac{\text{TN}}{\text{TN} + \text{FN}} \text{precision}^{\text{POS}} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (7) \\ F\text{-measure} &= \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \end{aligned}$$

According to Table 5 our proposed algorithm has the most value in F -measure in both positive and negative classes

(except in positive class for Australian data set that the value is the second best). F -measure is the harmonic mean of precision and recall. The significance of precision and recall is dependent to the domain. In some application areas precision is more interested than recall and in some others recall is more important. But with F -measure we consider both precision and recall measures concurrently. Therefore, IFAIS is a reliable learning algorithm for classification problems.

7. Conclusion

In this paper, we proposed a fuzzy classification system for credit scoring named IFAIS. The proposed classifier combines fuzzy logic and AIS concepts. The new proposed classification system was an enhanced version of FAIS and CCS-FAIS classifiers as the two earlier versions of AIS-based classification system for credit scoring. In our proposed IFAIS, we used immune memory to remember good B cells during the cloning process. We designed two forms of memory: simple memory and k -layer memory. Results indicated that our new definition of memory for immune-based fuzzy rule extraction increases the final classification rate of credit risk prediction significantly.

According to the promising results that we have obtained, using the immune principles is very effective for credit risk prediction; therefore, we will consider other concepts in artificial immune systems like negative selection or immune network as our future work.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] H. A. Abdou, "Genetic programming for credit scoring: the case of Egyptian public sector banks," *Expert Systems with Applications*, vol. 36, no. 9, pp. 11402–11417, 2009.
- [2] L. J. Mester, "What's the point of credit scoring?" *Business Review*, vol. 3, pp. 3–16, 1997.
- [3] L. Zhang, X. Hui, and L. Wang, "Application of adaptive support vector machines method in credit scoring," in *Proceedings of the 16th International Conference on Management Science and Engineering (ICMSE '09)*, pp. 1410–1415, September 2009.
- [4] S. Vukovic, B. Delibasic, A. Uzelac, and M. Suknovic, "A case-based reasoning model that uses preference theory functions for credit scoring," *Expert Systems with Applications*, vol. 39, no. 9, pp. 8389–8395, 2012.
- [5] B. W. Yap, S. H. Ong, and N. H. M. Husain, "Using data mining to improve assessment of credit worthiness via credit scoring models," *Expert Systems with Applications*, vol. 38, no. 10, pp. 13274–13283, 2011.
- [6] W. Gang and M. Jian, "A hybrid ensemble approach for enterprise credit risk assessment based on Support Vector Machine," *Expert Systems with Applications*, vol. 39, no. 5, pp. 5325–5331, 2012.
- [7] X. Zhou, W. Jiang, Y. Shi, and Y. Tian, "Credit risk evaluation with kernel-based affine subspace nearest points learning method," *Expert Systems with Applications*, vol. 38, no. 4, pp. 4272–4279, 2011.
- [8] D. L. Olson and D. D. Wu, "Review of innovative CSR: from risk management to value creation," *Journal of Cleaner Production*, vol. 18, no. 16, pp. 1767–1768, 2010.
- [9] K. Leung, F. Cheong, and C. Cheong, "Consumer credit scoring using an artificial immune system algorithm," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '07)*, pp. 3377–3384, Singapore, September 2007.
- [10] E. Altman, "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy," *Journal of Finance*, vol. 23, no. 4, pp. 589–609, 1968.
- [11] P. A. Lachenbruch, *Discriminant Analysis*, Hafner, New York, NY, USA, 1975.
- [12] W. E. Henley and D. J. Hand, "A k -nearest-neighbour classifier for assessing consumer credit risk," *Journal of the Royal Statistical Society Series D: The Statistician*, vol. 45, no. 1, pp. 77–95, 1996.
- [13] Y. E. Orgler, "A credit scoring model for commercial loans," *Journal of Money Credit Bank*, pp. 435–445, 1970.
- [14] S. Finlay, "Credit scoring for profitability objectives," *European Journal of Operational Research*, vol. 202, no. 2, pp. 528–537, 2010.
- [15] D. Wu and D. L. Olson, "Enterprise risk management: coping with model risk in a large bank," *Journal of the Operational Research Society*, vol. 61, no. 2, pp. 179–190, 2010.
- [16] D. Wu and D. L. Olson, "Enterprise risk management: small business scorecard analysis," *Production Planning & Control*, vol. 20, no. 4, pp. 362–369, 2009.
- [17] D. D. Wu and D. L. Olson, "Introduction to the special section on 'optimizing risk management: Methods and tools,'" *Human and Ecological Risk Assessment*, vol. 15, no. 2, pp. 220–226, 2009.
- [18] D. D. Wu, X. Kefan, L. Hua, Z. Shi, and D. L. Olson, "Modeling technological innovation risks of an entrepreneurial team using system dynamics: an agent-based perspective," *Technological Forecasting and Social Change*, vol. 77, no. 6, pp. 857–869, 2010.
- [19] D. D. Wu and D. Olson, "Enterprise risk management: a DEA VaR approach in vendor selection," *International Journal of Production Research*, vol. 48, no. 16, pp. 4919–4932, 2010.
- [20] D. D. Wu and D. L. Olson, "Introduction to special section on 'Risk and Technology,'" *Technological Forecasting and Social Change*, vol. 77, no. 6, pp. 837–839, 2010.
- [21] L. Wang and C. Fang, "An effective shuffled frog-leaping algorithm for multi-mode resource-constrained project scheduling problem," *Information Sciences*, vol. 181, no. 20, pp. 4804–4822, 2011.
- [22] Y. Wen, H. Xu, and J. Yang, "A heuristic-based hybrid genetic-variable neighborhood search algorithm for task scheduling in heterogeneous multiprocessor system," *Information Sciences*, vol. 181, no. 3, pp. 567–581, 2011.
- [23] J. Yang, H. Xu, and P. Jia, "Effective search for Pittsburgh learning classifier systems via estimation of distribution algorithms," *Information Sciences*, vol. 198, pp. 100–117, 2012.
- [24] L. Yu, "An evolutionary programming based asymmetric weighted least squares support vector machine ensemble learning methodology for software repository mining," *Information Sciences*, vol. 191, pp. 31–46, 2012.

- [25] R. Zhang and C. Wu, "Bottleneck machine identification method based on constraint transformation for job shop scheduling with genetic algorithm," *Information Sciences*, vol. 188, pp. 236–252, 2012.
- [26] V. S. Desai, J. N. Crook, and G. A. Overstreet Jr., "A comparison of neural networks and linear scoring models in the credit union environment," *European Journal of Operational Research*, vol. 95, no. 1, pp. 24–37, 1996.
- [27] J. E. Hunt and D. E. Cooke, "Learning using an artificial immune system," *Journal of Network and Computer Applications*, vol. 19, no. 2, pp. 189–212, 1996.
- [28] J. Timmis and T. Knight, "Artificial immune systems: using the immune system as inspiration for data mining," in *Data Mining: A Heuristic Approach*, pp. 209–230, Group Idea Publishing, 2001.
- [29] J. Timmis, A. Hone, T. Stibor, and E. Clark, "Theoretical advances in artificial immune systems," *Theoretical Computer Science*, vol. 403, no. 1, pp. 11–32, 2008.
- [30] E. Kamaloo and M. S. Abadeh, "An artificial immune system for extracting fuzzy rules in credit scoring," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '10)*, pp. 1–8, Barcelona, Spain, July 2010.
- [31] E. Kamaloo and M. S. Abadeh, "Comprehensible credit scoring with fuzzy artificial immune system," in *Proceedings of the 18th Iranian Conference on Electrical Engineering (ICEE '10)*, pp. 542–547, Isfahan, Iran, May 2010.
- [32] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.
- [33] C. Huang, M. Chen, and C. Wang, "Credit scoring with a data mining approach based on support vector machines," *Expert Systems with Applications*, vol. 33, no. 4, pp. 847–856, 2007.
- [34] P. Yao, "Hybrid classifier using neighborhood rough set and SVM for credit scoring," in *Proceedings of the International Conference on Business Intelligence and Financial Engineering (BIFE '09)*, pp. 138–142, Beijing, China, July 2009.
- [35] D. Zhang, M. Hifi, Q. Chen, and W. Ye, "A hybrid credit scoring model based on genetic programming and support vector machines," in *Proceedings of the 4th International Conference on Natural Computation (ICNC '08)*, pp. 8–12, October 2008.
- [36] J. Yi, "Credit scoring model based on the decision tree and the simulated annealing algorithm," in *Proceedings of the WRI World Congress on Computer Science and Information Engineering (CSIE '09)*, pp. 18–22, Los Angeles, Calif, USA, April 2009.
- [37] M. F. A. Gadi, X. Wang, and A. P. Do Lago, "Credit card fraud detection with artificial immune system," *Lecture Notes in Computer Science*, vol. 5132, pp. 119–131, 2008.
- [38] W.-C. Yeh, "Novel swarm optimization for mining classification rules on thyroid gland data," *Information Sciences*, vol. 197, pp. 65–76, 2012.
- [39] Z. Pei, G. Resconi, A. J. van der Wal, K. Qin, and Y. Xu, "Interpreting and extracting fuzzy decision rules from fuzzy information systems and their inference," *Information Sciences*, vol. 176, no. 13, pp. 1869–1897, 2006.
- [40] M. J. Zolghadri and E. G. Mansoori, "Weighting fuzzy classification rules using receiver operating characteristics (ROC) analysis," *Information Sciences*, vol. 177, no. 11, pp. 2296–2307, 2007.
- [41] X. Chang and J. H. Lilly, "Evolutionary design of a fuzzy classifier from data," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 34, no. 4, pp. 1894–1906, 2004.
- [42] Z. Lei and L. Ren-Hou, "Designing of classifiers based on immune principles and fuzzy rules," *Information Sciences*, vol. 178, no. 7, pp. 1836–1847, 2008.
- [43] A. Frank and A. Asuncion, *UCI Machine Learning Repository*, 2010, <http://archive.ics.uci.edu/ml>.
- [44] D. Yang, L. Jiao, M. Gong, and F. Liu, "Artificial immune multi-objective SAR image segmentation with fused complementary features," *Information Sciences*, vol. 181, no. 13, pp. 2797–2812, 2011.
- [45] J. Zhao, Q. Liu, W. Wang, Z. Wei, and P. Shi, "A parallel immune algorithm for traveling salesman problem and its application on cold rolling scheduling," *Information Sciences*, vol. 181, no. 7, pp. 1212–1223, 2011.
- [46] Y. Zhong, L. Zhang, B. Huang, and P. Li, "An unsupervised artificial immune classifier for multi/hyperspectral remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 2, pp. 420–431, 2006.
- [47] L. N. de Castro and F. J. von Zuben, "Learning and optimization using the clonal selection principle," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 3, pp. 239–251, 2002.
- [48] D. Dasgupta, "Advances in artificial immune systems," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 40–43, 2006.
- [49] J. Greensmith, U. Aickelin, and G. Tedesco, "Information fusion for anomaly detection with the dendritic cell algorithm," *Information Fusion*, vol. 11, no. 1, pp. 21–34, 2010.
- [50] Z. Jinquan, L. Xiaojie, L. Tao, L. Caiming, P. Lingxi, and S. Feixian, "A self-adaptive negative selection algorithm used for anomaly detection," *Progress in Natural Science*, vol. 19, no. 2, pp. 261–266, 2009.
- [51] R. R. Sumar, A. A. Rodrigues Coelho, and L. D. Santos Coelho, "Use of an artificial immune network optimization approach to tune the parameters of a discrete variable structure controller," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5009–5015, 2009.
- [52] K. Tan, C. Goh, A. Mamun, and E. Ei, "An evolutionary artificial immune system for multi-objective optimization," *European Journal of Operational Research*, vol. 187, no. 2, pp. 371–392, 2008.
- [53] H. Y. K. Lau, V. W. K. Wong, and I. S. K. Lee, "Immunity-based autonomous guided vehicles control," *Applied Soft Computing Journal*, vol. 7, no. 1, pp. 41–57, 2007.
- [54] G. Luh and W. Liu, "An immunological approach to mobile robot reactive navigation," *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 30–45, 2008.
- [55] E. Hart and J. Timmis, "Application areas of AIS: the past, the present and the future," *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 191–201, 2008.
- [56] W. Wang, S. Gao, and Z. Tang, "A complex artificial immune system," in *Proceedings of the 4th International Conference on Natural Computation (ICNC '08)*, pp. 597–601, Jinan, China, October 2008.
- [57] K. Polat and S. Güneş, "A hybrid medical decision making system based on principles component analysis, k -NN based weighted pre-processing and adaptive neuro-fuzzy inference system," *Digital Signal Processing*, vol. 16, no. 6, pp. 913–921, 2006.
- [58] X. Shen, X. Z. Gao, R. Bie, and X. Jin, "Artificial immune networks: models and applications," in *Proceedings of the International Conference on Computational Intelligence and Security (ICCIAS '06)*, pp. 394–397, October 2006.

- [59] L. de Castro and F. von Zuben, "aiNet: an artificial immune network for data analysis," in *Data Mining: A Heuristic Approach*, pp. 231–259, Group Idea Publishing, 2001.
- [60] J. Kelsey and J. Timmis, "Immune inspired somatic contiguous hypermutation for function optimisation," in *Genetic and Evolutionary Computation (GECCO '03)*, 2003.
- [61] H. Ishibuchi, K. Nozaki, N. Yamamoto, and H. Tanaka, "Selecting fuzzy if-then rules for classification problems using genetic algorithms," *IEEE Transactions on Fuzzy Systems*, vol. 3, no. 3, pp. 260–270, 1995.
- [62] O. Cordón and F. Herrera, "A three-stage evolutionary process for learning descriptive and approximate fuzzy-logic-controller knowledge bases from examples," *International Journal of Approximate Reasoning*, vol. 17, no. 4, pp. 369–407, 1997.
- [63] F. Hoffmann, "Combining boosting and evolutionary algorithms for learning of fuzzy classification rules," *Fuzzy Sets and Systems*, vol. 141, no. 1, pp. 47–58, 2004.
- [64] K. Nozaki, H. Ishibuchi, and H. Tanaka, "Adaptive fuzzy rule-based classification systems," *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 3, pp. 238–250, 1996.
- [65] M. S. Abadeh, J. Habibi, M. Daneshi, M. Jalali, and M. Khezzadeh, "Intrusion detection using a hybridization of evolutionary fuzzy systems and artificial immune systems," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '07)*, pp. 3547–3553, September 2007.
- [66] A. Watkins, J. Timmis, and L. Boggess, "Artificial immune recognition system (AIRS): An immune-inspired supervised learning algorithm," *Genetic Programming and Evolvable Machines*, vol. 5, no. 3, pp. 291–317, 2004.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

