International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 24 (2017) pp. 14421-14433

Adaptive Lion Fuzzy System to Generate the Classification Rules using Membership Functions based on Uniform Distribution

B. Lakshmi Ramani
Research Scholar, Department of Computer Science and Engineering,
Gandhi Institute of Technology and Management (GITAM), Visakhapatnam, India.

Orcid Id: 0000-0002-2969-6262

Dr. Padmaja Poosapati
Associate Professor, Department of Information Technology
Gandhi Institute of Technology and Management (GITAM), Visakhapatnam, India.

Abstract

There are many conventional classification approaches to classify large scale data. To deal with composite engineering problems the traditional classification methods become difficult in terms of complexity and efficiency. Thereby, to obtain efficient classifier model the attention of researchers is much towards the hybrid techniques like fuzzy with Neural Network, Genetic Algorithm and decision tree in the community of computational Intelligence. These techniques along with the optimization algorithm can be used to generate fuzzy rule based classifier.

In this work, an algorithm is proposed to design a fuzzy expert system is namely Adaptive Lion Fuzzy System (ALFS) that consists of two phases. In Phase I, the membership functions are developed based on Uniform Distribution. In Phase II, rules are generated using fitness function based on three factors such as length, matching and variance. Further, optimal rules are produced by adjusting the constants in the female update equation of Lion Algorithm (LA). Thus, the suggested approach in Phase II is named as Adaptive Lion Algorithm (ALA).

The Performance of the proposed system is evaluated with three metrics viz., sensitivity, specificity, and accuracy is exploited using three datasets, Statlog, heart disease, and lung cancer datasets. From the experimental results it is observed that the suggested ALFS has attained the maximum performance with sensitivity of 0.98, specificity of 0.959, and accuracy of 0.959, respectively as compared with some of the existing classification algorithms.

Keywords: Data classification, Fuzzy system, Rule generation, Membership function, optimal rules.

INTRODUCTION

Presently, managing large quantity of data has become a challenging process. Data mining is one of the promising areas that attract various industries developed to handle the huge volume of data. A challenging and effective approach in data mining is data classification [10]. Some of the popular classification algorithms available for data mining are decision tree classifiers [11], Bayesian classifiers [13], Rule-based classifiers [12], Artificial Neural Networks (ANNs) [15], Lazy Learners, ensemble methods [16], and Support Vector Machines (SVMs) [14]. Various techniques have been developed to mine the classification rules from the numeric data [17] [1].

The classifiers based on the fuzzy logic system [31] are categorized into two, namely pure fuzzy classifiers and Fuzzy Rule Based Classification System (FRBCS), also called fuzzy expert system [7]. Due to the lack of normalization and performance degradation, the former classifiers based on fuzzy clustering [19], fuzzy pattern matching [18] and fuzzy integral [20] are unsuitable to solve the classification problems [5]. The later classifier has a number of membership functions and if-then rules, which are generated by the domain experts. As a result, rule base is created for the fuzzy expert system based on the if-then rules, where qualitative reasoning is done to predict the results. The fuzzy rules are obtained from the training data space during the absence of domain experts. A fuzzy relation can be established depending on the if-then rules of the fuzzy logic system to show the relation between the input and the output values [21]. The process of extracting fuzzy rules from the data space is considered as a search problem in high dimensional data space, where every point specifies a set of rules, system behavior, and membership function [4].

The FRBCS involves a large number of widespread applications [22] that are effectively designed for supporting the requirements of intelligent support. Real-time issues in varied fields, like classification, classical network optimization, pattern recognition, pricing prediction system, travel choice behavior models, disease prediction, robotics based on behavior, identification of rice quality, etc., adopt FRBCS to attain optimized rule learning [23] [6]. The major advantage of the fuzzy rule-based classifier [24] is its capability of integrating human expert knowledge into the decision making process [5]. To generate and learn the classification rules of the fuzzy system, various methods, such as simple heuristic procedures, clustering methods [26], Genetic Algorithm (GA) [27], and neuro-fuzzy techniques [25], are available. Moreover, many heuristic and meta-heuristic algorithms, such as Particle Swarm Optimization (PSO) [28], Firefly [29], Simulated Annealing, Greywolf Optimizer [30], Artificial Bee Colony optimization (ABC), and so on, are developed based on the behavior of biological systems in nature [32] [37].

The rest of the paper is structured as follows: The survey providing various fuzzy logic systems that are used in the literature is described in section 2. Section 3 presents the suggested ALFS using the proposed ALA with an appropriate block diagram. The ALFS system employed for the classification is explained in section 4. The results of the suggested ALFS are discussed in section 5 with a comparative analysis, and the entire system is summarized in conclusion given in section 6.

RELATED WORK

Xiaodong Liu *et al.* [1] developed a function similar to the coherence membership function to explain fuzzy concepts built over the Axiomatic Fuzzy Set (AFS) theory. The membership function formed could hold two factors, such as fuzziness and randomness. Moreover, a method designed to create a classifier based on fuzzy rules using the membership functions. The AFS approach is applicable even for the datasets with mixed attributes. The major drawback of the approach is that it generates a large number of rules.

To optimize the membership functions and the rules, Binu Dennis *et al.* [2] presented Adaptive Genetic Fuzzy System (AGFS) for the classification of medical data. The technique projected to i) generate the rules from the data and the optimized rules selection based on GA and to describe the exploration problem in GA using systematic addition, ii) to develop a method for discretization and to scheme the membership function, and iii) to formulate a fitness function using the frequency of the rules occurred in the training data. AGFS is more applicable for the data having high dimension, but it does not provide precise rules.

D. Binu *et al.* [3] designed a classifier combining Bat algorithm with a fuzzy classifier for the effective generation of classification rules and membership functions. Here, the selection of optimized rules is based on Bat algorithm. The objective is to reduce the complexity in designing and discretizing the membership function. Even though the parameters are adjusted, the convergence rate was not affected and the rules produced are not optimized.

R. Chandrasekar *et al.* [4] presented a fuzzy system, named exponential brain storm fuzzy system, using the traditional fuzzy system with modifications in the rule definition. The authors developed Exponential Brain Storm Optimization (EBSO) algorithm using the exponential model in rule derivation. A technique based on uniform distribution is used to find the membership function. The system has high accuracy in classifying medical data, but requires further improvements in the classification.

Pugalendhi Ganesh Kumar *et al.* [5] developed a fuzzy expert system using microarray data classifier. To find the membership function, an algorithm, Ant Bee Algorithm (ABA), was presented, combining the benefits of two optimization algorithms, Ant Colony Optimization (ACO) and Ant Bee Colony (ABC). For the detection of informative genes, Mutual Information is utilized. The accuracy of the classification was improved, as the classifier had minimum false positive rate and large discrimination power. Although the accuracy is improved, a large number of genes can increase the error rate.

To overcome the challenges, such as selecting the type, parameters, and number of membership function of automatic fuzzy system, Neelu Khare *et al.* [6] designed Brain Genetic Fuzzy System (BGFS) to classify the data. The classification using BGFS is based on the Exponential Genetic BSO (EGBSO) that integrated EBSO with GA. EGBSO is used to devise the membership functions, while the rules are found using EBSO algorithm. However, BGFS has to consider certain factors for the performance improvements.

P. Ganesh Kumar *et al.* [7] presented Genetic Swarm Algorithm (GSA) by uniting GA and PSO, to obtain the optimal set of rules and to adjust the membership function. For the improvement of the convergence of GSA and accuracy in the microarray classification, advanced genetic operators are created. In GSA, the rule set is represented using binary strings, whereas the membership function values are represented using floating point numbers. The learning ability of GSA is better, but the approach consumes a considerable CPU time for the classification.

For the performance improvement of fuzzy rule-based systems, Victoria López *et al.* [8] focused on the utilization of information granulation. A positive synergy was formed between the approaches used for data sampling for the modifications of the algorithm and thereby, creates a genetic programming model that utilize the linguistic variables in a hierarchical manner. Thus, the authors obtained a Hierarchical Fuzzy Rule Based Classification System (HFRBCS) to handle the imbalanced datasets. The limitation of the system is that the features are to be enhanced to attain better performance.

Challenges:

The challenges observed in the fuzzy systems studied in the literature during the classification of data are as follows,

- One of the main issues in developing the fuzzy expert systems is the generation of fuzzy if-then rules and the membership functions. In such system, the membership function is treated as a search problem, where every point represents the membership function, a rule set, and the behavior of the system [6].

- Managing the fuzzy rule based classifier itself is a complex process, as the search space of the fuzzy rules grows exponentially when the number of patterns is large. The exponential growth, in turn, makes the learning process complex, causing scalability and complexity issues [10].
- Although the fuzzy system contains the historic data, the designing process, such as the development of rule set and the membership function is difficult and requires the knowledge of domain experts. Hence, it is necessary to perform these two processes automatically without the requirement of the experts in the fuzzy classifier.
- Another challenge in the fuzzy classifier deals with the development of the rules from the historical data. Due to the huge data space, the process of generating the fuzzy rules is considered as a search problem. Moreover, the rules obtained must be precise, to be applicable to the fuzzy system, and it must be related to the format of classification rules.
- In [1], the rules are defined from the decision tree. But, the challenge is to optimize the rules in two locations, such as the length of the rule and the total number of rules. Since the consideration of multiple attributes in the rules affects the performance of both in terms of accuracy and time, the length of the rule is important for rule optimization. The number of rules is the total available rules that are to be placed in the rule base. Another challenge is regarding the weight of the rules, as each rule has to be assigned different weights based on the importance of the data.

METHODOLOGY

In this section, an Adaptive Lion Fuzzy System (ALFS) is proposed that overcomes the above challenges. As such, fuzzy expert system is a system that utilizes a set of membership functions and if-then rules to classify the data based on the fuzzy relation established between input-output. One of the important elements in the fuzzy expert system is knowledge acquisition, which is obtained in most of the systems from the domain experts.

Thus, the suggested ALFS method consists of two phases. In Phase I, the membership function of each feature based on Uniform Distribution are formed. In Phase II, the feasible rules are generated using the fitness function. Further, these rules are made optimal by adopting Lion Algorithm (LA).

The proposed ALFS, rather than depending on expert's knowledge, the system generates the rules from the training data automatically. The fuzzified input values are matched with the rules obtained in the testing phase. The block diagram of the proposed ALFS is pictured out in figure 1.

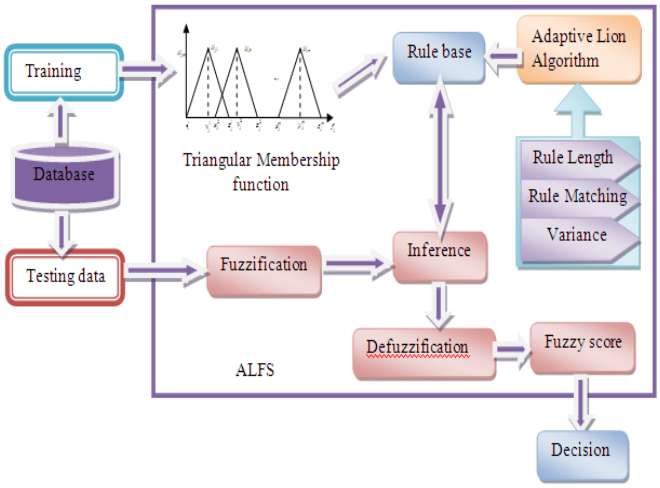

Figure 1: Block diagram of the proposed ALFS

Phase I:

Uniform distribution based membership function:

To design the membership function automatically, the proposed system follows the uniform distribution. The membership function is usually a curve, representing the data values mapped to the membership degree. Here, the number of membership functions for each feature is specified beforehand and the interval is defined based on the deviation factor.

Example 1

Let $X=\{p_1, p_2, p_3, p_4, p_5\}$ be a set of 5 members, $F=\{f_1, f_2, \ldots\ldots, f_8\}$ be a set of features where $f_1=$ Bengali, $f_2=$Marathi, $f_3=$Assame, $f_4=$male, $f_5=$female, $f_6=$Professor, $f_7=$Engineer, $f_8=$Doctor, i.e., each feature/attribute is a crisp concept [33] that are shown in table 1

Table1: Features description

	Bengali	Marathi	Assame	Male	Female	Prof.	Eng.	Doctor
p_1	Y	N	N	Y	N	N	Y	Y
p_2	N	Y	N	N	Y	Y	N	N
p_3	N	Y	N	Y	N	N	Y	N
p_4	Y	N	N	Y	N	Y	N	N
p_5	N	N	Y	Y	N	Y	Y	N

Consider a database 'A', having a number of data, as represented below,

$$A = \{b_{ij}\}; 1 < i \le p; 1 < j \le q \tag{1}$$

where, p is the number of data, and q is the number of features, which is expressed as,

$$D = \{f_j\}; 1 < j \le q \tag{2}$$

where, f_j represents the j^{th} feature in the data. The database is partitioned into training data, which is used for the generation of rules and membership function, and testing data, for the final classification. The training data is discretized in a specific interval by sorting the maximum and the minimum values of the features based on the deviation factor in ascending order. Hence, the maximum and the minimum values of each feature and the class are computed. Thus, the q^{th} feature has the following representation,

$$f_j = \left[f_{\min}^j \quad f_{\max}^j\right]; 1 < j \le q \tag{3}$$

where, $f_{\min}^j$ and $f_{\max}^j$ represent the minimum and the maximum values of j^{th} feature. Each feature is defined with a number of membership functions. The membership function is assumed to be the triangular membership function that is defined using three parameters, such as x, y, and z, in the interval specified. Figure 2 shows the triangular membership function of j^{th} feature.

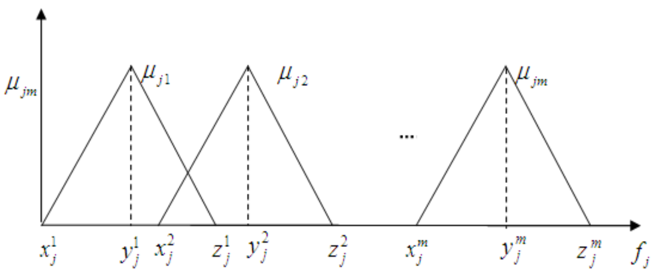

Figure 2: Triangular Membership Function

Hence, the membership function obtained from the feature vector is represented as,

$$f_j \Leftrightarrow \mu_{jl}; 1 < l \le m \tag{4}$$

where, μ_{jl} represents the membership function and m is the number of membership functions, which is assigned unequal value for each feature. The membership function of each feature depends on the feature and the three parameters, as defined below,

$$\mu_{jl} = F\left(f_j; x_j^l, y_j^l, z_j^l\right) \tag{5}$$

where, x_j^l, y_j^l, and z_j^l, are the parameters associated with l^{th} membership function in the j^{th} feature. Hence, for the feature f_j, the interval I to define the parameter of the membership function is specified within the range, $|I| = m+1$, based on a factor, known as deviation factor, and is given by,

$$I = \left\{f_{\min}^j, f_{\min}^j + k^j, f_{\min}^j + 2k^j, \ldots, f_{\max}^j\right\} \tag{6}$$

where, k^j is the deviation factor, which is computed based on the maximum and the minimum values of the j^{th} feature as,

$$k^j = \frac{f_{\max}^j - f_{\min}^j}{m} \tag{7}$$

where, m is the number of membership functions defined for f_j. Thus, the first membership function in the j^{th} feature can be computed as,

$$\mu_{jl} = \left(f_j; f_{\min}^j, \frac{f_{\min}^j + \left(f_{\min}^j + k^j\right)}{2}, f_{\min}^j + k^j\right) \tag{8}$$

Similarly, the value of μ_{j2} is given by,

$$\mu_{j2} = \left(f_j; f_{\min}^j + k^j, \frac{\left(f_{\min}^j + k^j\right) + \left(f_{\min}^j + 2k^j\right)}{2}, f_{\min}^j + 2k^j\right) \tag{9}$$

Therefore, the m^{th} membership function of the j^{th} feature is calculated as,

$$\mu_{jm} = \left(f_j; f_{\min}^j + (m-1)k^j, \frac{\left(f_{\min}^j + (m-1)k^j\right) + f_{\max}^j}{2}, f_{\max}^j\right) \tag{10}$$

where, $f_{\min}^j$, $f_{\max}^j$ are the minimum and the maximum values of j^{th} feature, and k is the deviation factor.

Phase II:

Optimal generation of fuzzy rules using proposed Adaptive Lion Algorithm:

The procedure of generating the rules is explained using the proposed ALA. To make LA adaptive, the constants in the female update process are modified so that the proposed fuzzy system is applicable for the classification of even imbalanced datasets. The optimal generation of rules in ALA depends on the length, matching, and the variance of the rules, which are utilized in the fitness function. Hence, it is possible to select the optimal fuzzy rules using the proposed ALA based on the fitness function that is adopted in the fuzzy system for the classification. Following sections describe the process of generating the rules using the proposed algorithm.

Solution formulation:

The objective of the proposed algorithm is to determine how the rules are to be generated. Hence, in ALA, the solution is a vector, where the first element indicates the number of rules, and the remaining elements define the rules. Let n be the number of rules, which is of dimension $q+2$, where q is the number of features and the other two elements, denoted as 2, correspond to the class of output and the weight of the rule. Hence, the size of the solution vector is $1+(q+2)*n$ and the solution is initially generated in random. From the random set of solutions of dimension $1+(q+2)*n$, the optimal one is selected using the proposed ALA based on the fitness computed. Thus, the fuzzy rules that are optimal for the classification of the data can be selected.

Multi-objective fitness function:

The fitness function of the proposed ALA that helps to determine the best solution, i.e. the optimal fuzzy rule, is developed using three factors, length of the rule, matching count of the rule, and variance of the rule. A solution is selected as the best solution if the fitness of the solution is in maximum. Hence, the best solution is suggested to have reduced length, high matching count, and minimum variance. The fitness of the proposed ALA is computed as,

$$N^{FIT} = \lambda_1[1-F_1]+\lambda_2 F_2+\lambda_3[1-F_3] \qquad (11)$$

where, λ_1, λ_2, and λ_3 are the weights, F_1 is the fitness factor defining the length, F_2 is the factor defining the matching, and F_3 is that of the variance of the rules.

Length: The first factor considered in the fitness evaluation is the length of the rule. Lesser the length of the rule greater is its effectiveness. Hence, the rules to be generated must have a minimum length. Including length as a factor in the fitness, the solution, i.e. the rules, with reduced length can be selected. The length of the rule is computed as,

$$F_1 = \frac{\sum_{j=1}^{n} L_j^N}{q*n} \qquad (12)$$

where, L^N is the length, q is the number of features, and n is the number of rules. The value of F_1 is between 0 and 1, where 0 is the best value and 1 is the worst.

Matching factor: The matching factor finds the matching between the rules obtained with the training data. The matching is found between the data and the rules, after the discretization of the data using the triangular membership function. Maximum matching is required for the rule to be selected as the best one, which indicates that the rule generated matches with the original data. Thus, the matching count is given by,

$$F_2 = \frac{\sum_{j=1}^{n} a(n_j, A)}{p*n} \qquad (13)$$

where, $a(n_j, A)$ is the function that computes the matching of n_j with A, and p is the number of data. F_2 ranges from 0 to 1, wherein 0 is the worst value and 1 is the best.

Variance: Variance is the third factor used to evaluate the feasibility of the solution. The variance of the rules obtained from different classes are computed and is divided by the total number of rules, for the normalization of the value in the interval [0,1]. The rule that has the minimum variance forms the best solution.

$$F_3 = \frac{v(n,C)}{n} \qquad (14)$$

where, $v(n,C)$ represents the function computing the variance of the rules in each class C. F_3 takes a value that ranges from 0 to 1 such that 1 is the best case and 0 is the worst value.

Proposed Adaptive Lion Algorithm:

This section explains the proposed ALA to make the algorithm adaptive for the extraction of fuzzy rules. LA [32] is an optimization algorithm developed based on the unique behavior of lion, namely territorial defense, territorial takeover, and so on. In LA, if the territorial lion is stronger, it attempts to defeat the nomadic lion in the territorial defense. ALA is made adaptive by adjusting the constants of the female lion update equation of LA. Following are the steps involved in the proposed algorithm.

I. Initialization:

The fundamental step of the algorithm is the initialization of pride constituted by female lion and male lion, as represented below,

$$Y^F = \{Y_1^F, Y_2^F, \ldots, Y_U^F\} \tag{15}$$

$$Y^M = \{Y_1^M, Y_2^M, \ldots, Y_V^M\} \tag{16}$$

where, U and V are the total number of female and male lions in the pride. In this work, the solutions in U and V are assumed to be 10. Moreover, two nomadic lions, denoted as Y_1^N and Y_2^N, exist in the population, among which Y_1^N is initialized in the first step and Y_2^N will be initialized at the territorial defense.

II. Fitness Computation:

After the initialization of Y^F, Y^M, and Y_1^N, the fitness of each is computed using equation (11). The solution among Y^M that has the maximum fitness, represented as $F(Y^M)$, and is selected as the best solution, i.e. $F^r = F(Y^M)$.

III. Fertility Computation:

To avoid the problem of convergence in local optima, the algorithm updates the female lion in the current population. This includes the utilization of few parameters, such as laggardness rate, female update count, sterility rate, and female generation count. Thus, the female update in LA is given as,

$$Y_r^F(t+1) = \left[Y_r^F(t) + (0.1b_2 - 0.05)\left(Y_r^M(t) - b_1 Y_r^F(t)\right)\right] \tag{17}$$

where, $Y_r^F(t)$ and $Y_r^M(t)$ are the r^{th} female and male lions at the current iteration t, b_1 and b_2 are two integers chosen randomly between 0 and 1. The performance of LA can be improved in solving optimization problems by making it adaptive in solving such problems. To make the algorithm adaptive, the update equation is modified by replacing the random integers with a term γ_{adap} as given below,

$$Y_r^F(t+1) = \left[Y_r^F(t) + (2\gamma_{adap} b_2 - \gamma_{adap})\left(Y_r^M(t) - b_1 Y_r^F(t)\right)\right] \tag{18}$$

where, γ_{adap} is the adjusting parameter defined based on the number of iteration and the fitness values of $Y^F(t)$ and $Y^M(t)$ as,

$$\gamma_{adap} = \left[\frac{t}{t_{\max}} + \frac{F(Y^F(t))}{F(Y^M(t))}\right] \tag{19}$$

where, t is the current iteration, $t_{\max}$ is the maximum iteration, $F(Y^F(t))$ is the fitness of $Y^F(t)$, and $F(Y^M(t))$ is the fitness of $Y^M(t)$. Thus, equation (19) forms the female lion update equation of the proposed ALA, which makes it suitable to apply for large scale problems.

IV. Mating:

Mating involves two processes, namely crossover, and mutation, wherein four cubs are formed based on the crossover probability. The result of crossover is given as,

$$Y^c = M_s \circ Y^M + \overline{M}_s \circ Y^F ; s = 1,2,3,4 \tag{20}$$

where, M is the crossover mask, $\circ$ is the Hadamard product, and Y^c is the cub obtained. Among the four cubs obtained, the one with the best fitness is the male cub, Y^{m-c} and the second best is the female cub, Y^{f-c}. Both the solutions undergo mutation to determine the best Y^{m-c} and Y^{f-c}.

V. Territorial defense:

In territorial defense, the pride and the nomad lions are updated based on the survival fight. Here, Y_2^N is initialized via mutation. Finally, the lion that wins in territorial defense is represented as, Y^{Nomad}. If Y^M is defeated, it will be replaced by Y^{Nomad} and a nomad Y^N is selected when its e^N is greater than or equal to an exponential function given as follows,

$$e_1^N = \exp\left(\frac{u_1}{\max(u_1, u_2)}\right)\frac{\max\left(F(Y_1^N), F(Y_2^N)\right)}{F(Y_1^N)} \tag{21}$$

where, u_1 denotes the Euclidean distance from Y_1^N to Y^M and u_2 denotes the Euclidean distance from Y_2^N to Y^M. Hence, Y^N is selected based on the following criterion,

$$Y^N = \begin{cases} Y_1^N \ ; \ if \ e_1^N \geq E \\ Y_2^N \ ; \ otherwise \end{cases} \tag{22}$$

where, E is the exponential of unity. In territorial takeover, the territory is owned by Y^{m-c} and Y^{f-c} when they mature.

VI. Termination:

The steps from II to V are repeated until the condition to terminate the algorithm is met. Finally, the solution having the maximum fitness will be selected as the best solution.

Algorithm of Adaptive Lion Algorithm:

1 Input: Y^M, Y^F, R^L, R^S

2 Output: Y^M

3 Parameters: Laggardness rate R^L, sterility rate R^S, nomad lions $\left(Y_1^N, Y_2^N\right)$, winner Y^{Nomad}, iteration t.

4 Begin

5 Initialize Y^M, Y^F and Y_1^N

6 if $t < t_{\max}$

7 Compute the fitness using equation (11)

8 Fertility Computations: if $F^r \geq F\left(Y^M\right)$

9 $R^L = R^L + 1$

10 else

11 $F^r = F\left(Y^M\right)$

12 End if

13 Evaluate $Y^F(t+1)$ using equation (18)

14 if $F\left(Y^F(t+1)\right) > F\left(Y^F(t)\right)$

15 $Y^F = Y^F(t+1)$

16 End if

17 Territorial defense: if $F\left(Y^{Nomad}\right) > F\left(Y^M\right) \& F\left(Y^{Nomad}\right) > F\left(Y^{f-c}\right) \& F\left(Y^{Nomad}\right) > F\left(Y^{m-c}\right)$

18 $Y^M = Y^{Nomad}$

19 else

20 Update nomadic coalition by selecting Y^N as Y^M using equation (22)

21 End if

22 Territorial Takeover: if $F\left(Y^M\right) > F\left(Y^{m-c}\right)$

23 $Y^M = Y^{m-c}$

24 End if

25 if $F\left(Y^f\right) > F\left(Y^{f-c}\right)$

26 $Y^f = Y^{f-c}$

27 End if

28 $t = t + 1$

29 Return Y^M

30 Terminate

Rule weight Formulation:

Adjusting the weight of the fuzzy rules obtained is an important process, for which most of the FRBCS utilize several heuristic and learning techniques. Rule weighting is done in the fuzzy systems to utilize the information available in the training data for the performance enhancement of the classifier. The optimal rules that are generated using ALA from the 'n' rules are different and hence, they have a varying degree of suitability. Therefore, it is a necessity to consider the most appropriate mechanism for rule weighting. In the proposed ALFS, the rules are weighted based on two factors, such as matching and length of the rules. Hence, the rule weight for the ALFS is computed as,

$$R^W = \frac{1}{2} * \left[\left(1 - F_1\right) + F_2\right] \tag{23}$$

where, F_1 is the length of the rule and F_2 represents the matching count of the rule with the database.

PROPOSED SYSTEM

Adaptive Lion Fuzzy System (ALFS):

In this section, the ALFS designed by adopting the proposed ALA in the rule generation, for the classification of data.

Algorithm of Adaptive lion Fuzzy System:

1 Input: Measured crisp inputs.

2 Output: crisp outputs.

3 begin

4 Find the membership function using equation (4)

5 Generate the rules using fitness equation (11)

6 Select the optimal rules from equation (18) and (22)

7 Fuzzy score is generated by using equation (23)

8 end

After the formation of fuzzy membership function and rule base, the classification is performed in the proposed ALFS. As depicted in figure 1, the three major components of ALFS are fuzzification, fuzzy inference, and defuzzification. The input testing data value is converted into a fuzzy value in fuzzification, while the inference infers the rule by matching the rule base with the input. The set of rules generated using ALA, based on the training data given as input, can be represented as $Z^R = \{Z_h^R\}; 1 < h \le n$, where n is the total number of rules. The if-then rules are generated optimally according to the rule weights defined. Finally, in defuzzification, the fuzzy score is generated based on three linguistic terms, such as ‘small’, ‘medium’ and ‘large’, which describes the relation of the input with the class. Based on the fuzzy score, the input data is classified as per the class labels found in test data.

RESULTS & DISCUSSIONS

In this section, we present the experimental results of the proposed ALFS-based fuzzy rule extraction with the performance comparison by taking 3 datasets. Three datasets are Statlog [34] (Australian credit approval), heart disease, and lung cancer datasets, are utilized for the experimentation. The Statlog dataset deals with credit card applications and has 690 instances and 14 attributes. It consists of eight categorical and six numerical attributes. For the protection of confidential data, the attribute names and the values are converted into meaningless symbols. The dataset has a mixture of continuous and nominal valued attributes, with few missing values. The second dataset employed for the experiment is the heart disease dataset [35], which consists of 303 instances and 76 attributes. This involves four databases, namely Cleveland, Hungary, Switzerland, and VA Long Beach. Among the 76 attributes available, 14 of them, including age, sex, cp, trestbps, and so on, are used commonly for the experiments. The sources of the dataset include Hungarian Institute of Cardiology in Budapest, University Hospital in Switzerland, and so on. The donor of the dataset 2 is David W. Aha. The attributes in the dataset 2 are characterized as categorical, integer, and real. The third dataset considered is the lung cancer dataset [36], which is donated by Stefan Aeberhard. The number of instances and attributes in the dataset are 32 and 56, and the dataset includes three kinds of pathological lung cancers. Initially, this dataset was used to demonstrate the power of the optimal discriminant plane in an experiment performed by Hong and Young.

Experimental Study:

A comparative analysis is done and the methods employed for the comparison are AFSDT [1], AGFS [2], BGFS [6], and BSFS [4]. In [1], a fuzzy system, called AFS-DT is developed to extract the fuzzy rules from the decision tree, whereas in [2], AGFS is designed to classify the medical data by making the GFS adaptive. Modifying BSO algorithm, EBSO is developed and is utilized in the fuzzy system, to design BSFS in [4]. Meanwhile, in [6], another fuzzy system, BGFS is presented for data classification using EGBSO that integrates GA in EBSO. The performance of these methods is compared with that of the proposed ALFS for the evaluation.

The Performance of the compartive methods is evaluated using three metrics, such as sensitivity, specificity, and accuracy with the datasets, Statlog (Dataset 1), heart disease (Dataset 2), and lung cancer (Dataset 3) and the metrics are defined as follows,

i) *Sensitivity:* It refers to the proportion of positives correctly identified in the classification result.

ii) *Specificity:* Specificity gives the measure of the proportion of negatives identified exactly from the classification.

iii) *Accuracy:* Accuracy is the degree of trueness that gives the proportion of true positives or true negatives.

$$Sensitivity = \frac{TP}{TP + FN} \tag{24}$$

$$Specificity = \frac{TN}{TN + FP} \tag{25}$$

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{26}$$

where, TN is True Negative, TP is True Positive, FN is False Negative, and FP is False Positive.

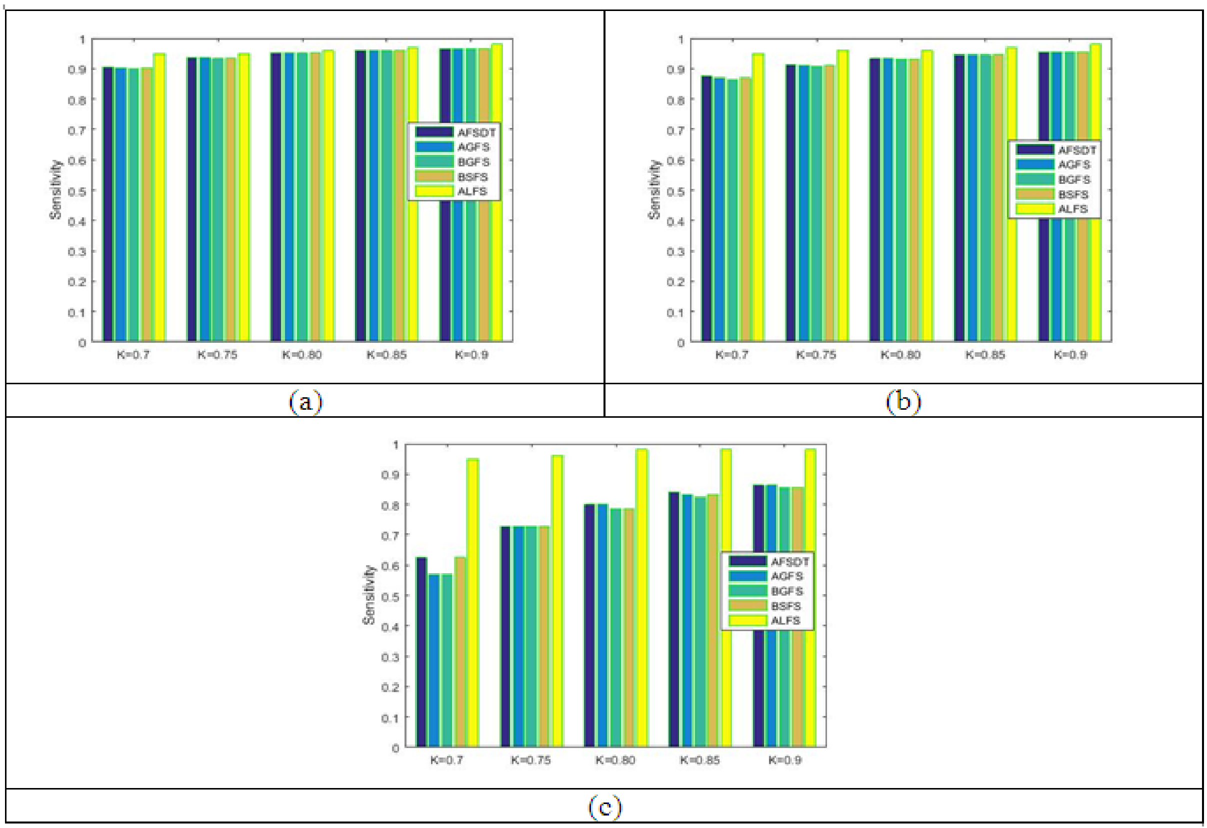

Figure 4: Analysis based on sensitivity using (a) dataset 1, (b) dataset 2, (c) dataset 3

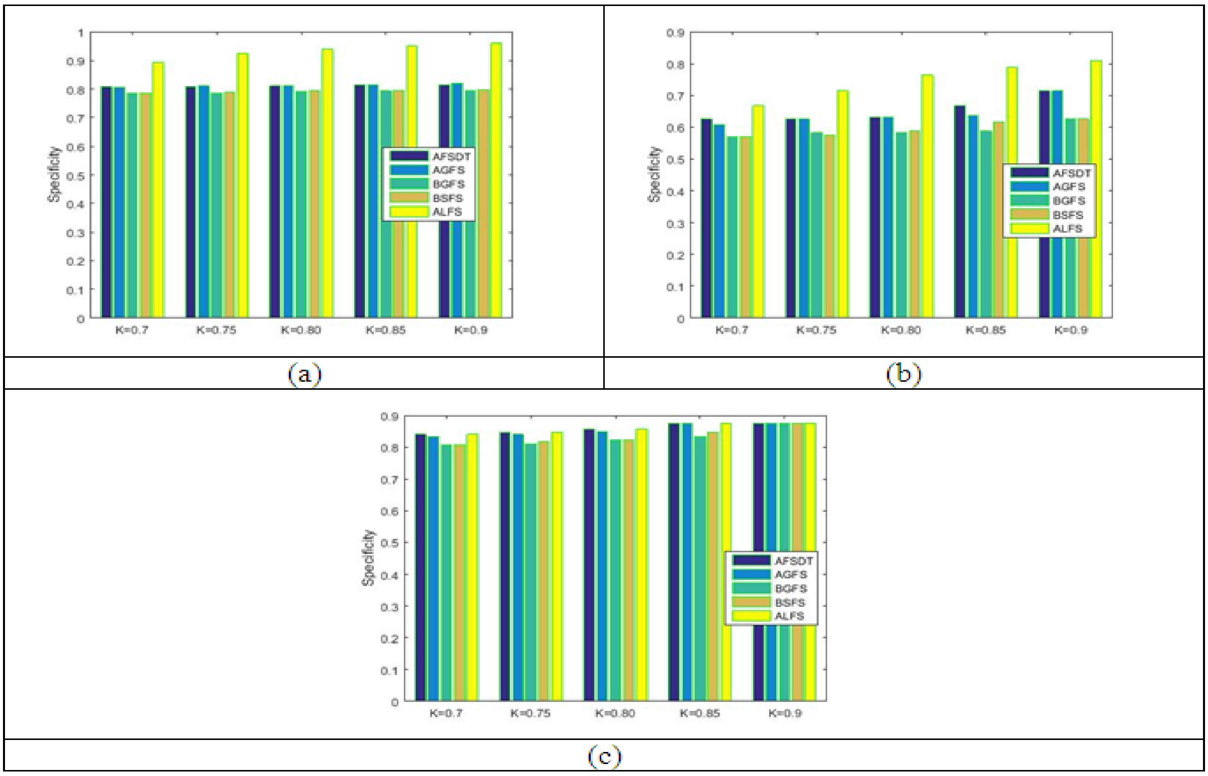

Figure 5: Analysis based on specificity using (a) dataset 1, (b) dataset 2, (c) dataset 3

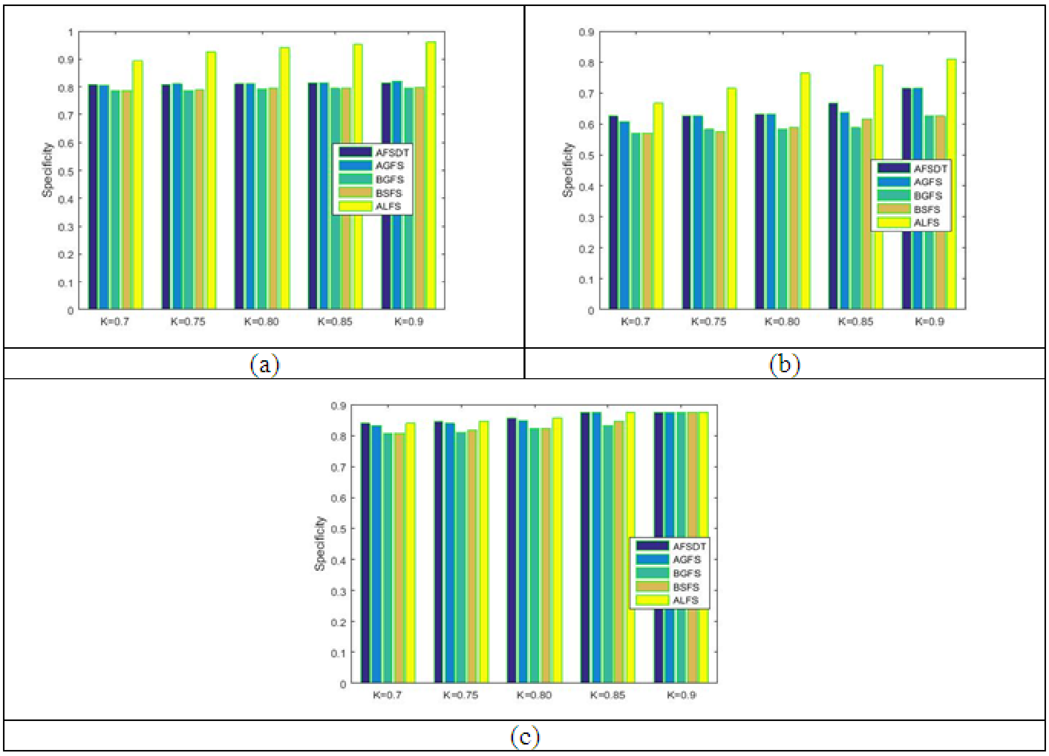

Figure 6: Analysis based on accuracy using (a) dataset 1, (b) dataset 2, (c) dataset 3

Comparative Analysis based on sensitivity:

The analysis based on sensitivity in the comparative methods is illustrated using figure 4, for the three datasets by varying the percentage of training data, denoted as K, as 70, 75, 80, 85, and 90. Figure 4.a shows the results analysis based on sensitivity using dataset 1. It is observed that when the percentage of data is increased, the sensitivity is maximized in all the comparative methods. For 70% data, the sensitivity attained using AFSDT is 0.906, which increases to 0.968 for the maximum percentage of data. The sensitivity produced by AGFS and BGFS is 0.903 and 0.9, when 70% is the training data. Meanwhile, the proposed system has a sensitivity of 0.95 and 0.98, for the training data taken as 70% and 90%, respectively. In figure 4.b, the sensitivity analysis in all the methods compared using dataset 2 is depicted, which clearly shows that the proposed system has the maximum sensitivity than the other existing methods for all the percentages of training data considered. When 0.956 is the maximum sensitivity obtained using AFSDT, the proposed ALFS has the sensitivity 0.98. The analysis based on sensitivity using dataset 3 is sketched out in figure 4.c. The sensitivity obtained for 80% training data is 0.8 using AFSDT and AGFS and it is increased to 0.864 for the maximum data considered in training.

Comparative Analysis based on specificity:

Figure 5 pictures out the analysis carried out based on the metric, specificity, in all the five methods using the three datasets. The specificity analysis using dataset 1 is represented in figure 5.a, where the maximum specificity is observed at 100% training data in all the methods. The maximum specificity obtained in the existing AFSDT, AGFS, BGFS, and BSFS, is 0.816, 0.821, 0.795, and 0.797, while that in ALFS is 0.959, respectively. Figure 5.b shows the analysis based on specificity using dataset 2 in all the five methods. The specificity of AFSDT, AGFS, BGFS, BSFS, and ALFS, is in maximum for 100% training data with values 0.714, 0.714, 0.625, 0.625, and 0.809, respectively. In figure 5.c, the specificity analysis in the comparative methods using dataset 3 is shown. Initially, as the training data considered is 70%, the specificity attained is 0.84, 0.833, 0.808, 0.808, and 0.84, by AFSDT, AGFS, BGFS, BSFS, and ALFS, but have the specificity of 0.875 for 100% training data. However, the proposed system seems to have the maximum specificity than the existing methods considered.

Comparative Analysis based on accuracy:

The comparative analysis based on accuracy is illustrated using figure 6, for the three datasets. Figure 6.a presents the accuracy analysis plot using dataset 1, showing the maximum accuracy for 100% training data. AFSDT & BSFS attained the accuracies as 0.816 and 0.797 with 100% training data, the proposed ALFS has an accuracy of 0.959. The analysis based on accuracy using dataset 2 is presented in figure 6.b.

Table 2: Performance Comparison

Comparative methods	*Dataset 1*			*Dataset 2*			*Dataset 3*		
	Sensitivity	*Specificity*	*Accuracy*	*Sensitivity*	*Specificity*	*Accuracy*	*Sensitivity*	*Specificity*	*Accuracy*
AFSDT	0.968	0.816	0.816	0.956	0.714	0.714	0.864	0.875	0.851
AGFS	0.968	0.821	0.821	0.955	0.714	0.714	0.864	0.875	0.851
BGFS	0.967	0.795	0.795	0.955	0.625	0.625	0.857	0.875	0.829
BSFS	0.967	0.797	0.797	0.955	0.625	0.625	0.857	0.875	0.829
ALFS	**0.98**	**0.959**	**0.959**	**0.98**	**0.809**	**0.809**	**0.98**	**0.875**	**0.915**

The accuracy measured by AFSDT for 70% data is 0.625, which increases to 0.714 at the maximum data considered. Meanwhile, the proposed ALFS has the maximum accuracy of 0.809. In figure 6.c, the chart showing the accuracy analysis plotted against varying percentages of data is pictured out. For 70% data, the accuracy obtained by AFSDT, AGFS, BGFS, BSFS, and ALFS, is 0.75, 0.733, 0.733, 0.75, and 0.75, which rise to 0.851, 0.851, 0.829, 0.829, 0.915, for the maximum data percentage. Thus, it is clear that the proposed ALFS has maximum accuracy than the existing methods, for all the percentages of data considered.

SUMMARY

The performance of the proposed ALFS is compared with other existing methods like AFSDT, AGFS, BGFS, BSFS, and ALFS are shown in table 2.

The performance evaluation metrics, namely sensitivity, specificity, and accuracy, measured by the five comparative methods with the maximum percentage of training data it is observed that the maximum performance attained by the proposed ALFS. For dataset 2, when 0.956 is the maximum sensitivity of AFSDT, ALFS has a value 0.98. The accuracy measured for the third dataset is 0.851 using AFSDT and AGFS. Meanwhile, in BGFS and ASFS, it is reduced to 0.829, respectively. From the table2, it is observed that the three metrics measured by the comparative methods is maximum for the dataset 1, where the proposed ALFS has an accuracy of 0.959.

CONCLUSION

In this paper, an adaptive fuzzy system is designed for data classification using the proposed ALFS that modifies the traditional fuzzy logic system. In ALFS, the triangular membership function is designed for every feature based on a uniform distribution. For the training data given, the fuzzy rules are generated by proposing ALA, which makes LA adaptive by adjusting the constants in the female update process. Matching the rules with the fuzzified values obtained using the triangular membership function, the score is determined, and thereby, the proposed ALFS classifies the data. The performance of the proposed system is evaluated using three metrics, sensitivity, specificity, and accuracy, and is compared with existing AFSDT, AGFS, BGFS, BSFS, and ALFS. The experiment is carried out in MATLAB using Statlog, heart disease, and lung cancer datasets, where the maximum performance is observed for Statlog dataset. From the performance comparison, it can be concluded that the proposed ALFS has attained the maximum performance in terms of the three metrics than the existing methods, with sensitivity of 0.98, specificity of 0.959, and accuracy of 0.959, respectively.

REFERENCES

[1] Xiaodong Liu, Xinghua Feng, and Witold Pedrycz, "Extraction of fuzzy rules from fuzzy decision trees: An axiomatic fuzzy sets (AFS) approach," Data & Knowledge Engineering, vol. 84, pp. 1-25, March 2013.

[2] Binu Dennis and Selvi Muthukrishnan, "AGFS: Adaptive Genetic Fuzzy System for medical data classification", Applied Soft Computing, vol. 25, pp. 242-252, December 2014.

[3] D. Binu and M. Selvi, "BFC: Bat Algorithm Based Fuzzy Classifier for Medical Data Classification," Journal of Medical Imaging and Health Informatics, vol. 5, no. 3, pp. 599-606, June 2015.

[4] R. Chandrasekar and Neelu Khare, "BSFS: Design and Development of Exponential Brain Storm Fuzzy System for Data Classification," International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 25, no. 2, pp. 267−284, 2017.

[5] Pugalendhi GaneshKumar, Chellasamy Rani, Durairaj Devaraj, and T. Aruldoss Albert Victoire, "Hybrid Ant Bee Algorithm for Fuzzy Expert System Based Sample Classification," IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 11, no. 2, pp. 347-360, March/April 2014.

[6] Chandrasekar Ravi and Neelu Khare, "BGFS: Design and Development of Brain Genetic Fuzzy System for Data Classification," Journal of Intelligent Systems, 2016.

[7] P. Ganesh Kumar, T. Aruldoss Albert Victoire, P. Renukadevi, and D. Devaraj, "Design of fuzzy expert system for microarray data classification using a novel Genetic Swarm Algorithm," Expert Systems with Applications, vol. 39, no. 2, pp. 1811-1821, February 2012.

[8] Victoria López, Alberto Fernández, María José del Jesus, and Francisco Herrera, "A hierarchical genetic fuzzy system based on genetic programming for addressing classification with highly imbalanced and borderline data-sets," Knowledge-Based Systems, vol. 38, pp. 85-104, January 2013.

[9] Biao Qin, Yuni Xia, Sunil Prabhakar, and Yicheng Tu, "A Rule-Based Classification Algorithm for Uncertain Data," In Proceedings of the EEE International Conference on Data Engineering, pp. 1633-1640, 2009.

[10] Sanket S. Ranaware and Dr. G.P. Potdar, "A Survey on Different Techniques of Classification in Data mining," International Research Journal of Engineering and Technology (IRJET), vol. 3, no. 5, pp.899-902, 2016.

[11] J. R. Quinlan, “Programs for Machine Learning,” Morgan Kaufman Publishers, 1993.

[12] W. W. Cohen, “Fast effective rule induction,” In Proceedings of the 12th International Conference on Machine Learning, pp. 115–123, 1995.

[13] P. Langley, W. Iba, and K. Thompson, “An analysis of Bayesian classifiers,” In Proceedings of the National Conference on Artigicial Intelligence, pp. 223– 228, 1992.

[14] V. Vapnik, “The Nature of Statistical Learning Theory,” Springer Verlag,1995.

[15] R. Andrews, J. Diederich, and A. Tickle, “A survey and critique of techniques for extracting rules from trained artificial neural networks,” Knowledge Based Systems, vol. 8, no. 6, pp. 373–389, 1995.

[16] T. G. Dietterich, “Ensemble methods in machine learning,” Lecture Notes in Computer Science, vol. 1857, pp. 1–15, 2000.

[17] H. Ichihashi, T. Shirai, K. Nagasaka, and T. Miyoshi, “Neuro fuzzy ID3: a method of inducing fuzzy decision trees with linear programming for maximizing entropy and algebraic methods,” Fuzzy Sets and Systems, vol. 81, no. 1, pp. 157–167, 1996.

[18] G. Schaefer, “Thermography Based Breast Cancer Analysis Using Statistical Features and Fuzzy Classification,” Pattern Recognition, vol. 42, no. 6, pp. 1133-1137, 2009.

[19] X. Zong, Z. Yong, J. Li-min, and H. Wei-li, “Construct Interpretable Fuzzy Classification System Based on Fuzzy Clustering Initialization,” Information Technology, vol. 11, no. 6, pp. 91- 107, 2005.

[20] Y. Hu, “Fuzzy Integral-Based Perceptron for Two-Class Pattern Classification Problems,” Information Sciences, vol. 177, no. 7, pp. 1673-1686, 2007.

[21] P. G. Kumar, T. A. A. Victoire, C. Renukadevi and D. Devaraj, “Design of fuzzy expert system for microarray data classification using a novel Genetic Swarm Algorithm,” Expert Systems with Applications, vol. 39, pp.1811–1821, 2012.

[22] W. Combs and J. Andrews, “Combinatorial rule explosion eliminated by a fuzzy rule configuration,” IEEE Transaction on Fuzzy Systems, vol.6, pp.1–11, 1998.

[23] C. A. P. Reyes, “Evolutionary fuzzy modeling,” Coevolutionary Fuzzy Modeling, Lecture Notes in Computer Science, vol. 3204, pp. 27–50, 2004.

[24] Z. Wang and V. Palade, “A Comprehensive Fuzzy-Based Framework for Cancer Microarray Data Gene Expression Analysis,” In Proceedings of the IEEE International Conference on Bioinformatics and Bioengineering, pp. 1003-1010, 2007.

[25] P. P. Angelov and R. A. Buswell, “Automatic generation of fuzzy rule-based models from data by genetic algorithms,” Information Sciences, vol.150, n0. 1-2, pp. 17-31 2003.

[26] I. Jagielska, C. Matthews and T. Whitfort, “An investigation into the application of neural networks, fuzzy logic, genetic algorithms, and rough sets to automated knowledge acquisition for classification problems,” Neuro Computing, vol. 24, no. 1-3, pp. 37-541999.

[27] M. Russo, “FuGeNeSys — A fuzzy genetic neural system for fuzzy modeling,” IEEE Transaction on Fuzzy Systems, vol.6, no.3, pp. 373-388, 1998.

[28] X. S. Yang, “Nature-Inspired Metaheuristic Algorithms,” Luniver Press, 2008.

[29] M. Mitchell, “An Introduction to Genetic Algorithms,” MIT Press, 1998.

[30] S. Mirjalili, S. M. Mirjalili and A. Lewis, “Grey wolf optimizer,” Advances in Engineering Software, vol. 69, pp. 46–61, 2014.

[31] J. Casillas, B. Carse and L. Bull, “Fuzzy-XCS: A michigan genetic fuzzy system,” IEEE Transaction on Fuzzy Systems, vol. 15, no.4, pp. 536-550, 2007.

[32] B R Rajakumar, "Lion Algorithm for Standard and Large Scale Bilinear System Identification: A Global Optimization based on Lion’s Social Behavior", In proceedings of IEEE Congress on Evolutionary Computation (CEC), Beijing, China,l July 6-11, 2014.

[33] Lakshmi Ramani B and Padmaja Poosapati "A Study of Notations and Illustrations of Axiomatic Fuzzy Set Theory", International Journal of Computer Applications, Volume 134 – No.11, January 2016

[34] Statlog (Australian Credit Approval) Data Set from "http://archive.ics.uci.edu/ml/datasets/statlog + (australian+credit+approval)",accessed on July 2017.

[35] Heart Disease Data Set from "http://archive.ics.uci.edu/ml/datasets/heart+Disease", accessed on July 2017.

[36] Lung cancer Data Set from "https://archive.ics.uci.edu/ml/datasets/lung+cancer", accessed on July 2017.

[37] B. Lakshmi Ramani, Padmaja Poosapati "A literature Study of Fuzzy Rule Based Classifiers", International Journal of Scientific Research and Review ISSN NO: 2279-543X, Volume 6, Issue 10, 2017.