

Optimized Hybridization of Ant Colony Optimization and Genetic Algorithm (HACOGA) Based IC-FNN Classifier for Abalone

Pramoda Patro^{1,*}, Krishna Kumar², and G. Suresh Kumar³

¹Research Scholar, Department of Mathematics, Koneru Lakshmaiah Education Foundation, Vaddeswaram 522502, Andhra Pradesh, India

²Department of Mathematics, MIT School of Engineering, MIT Art, Design and Technology University, Loni Kalbhori 412201, Pune, India

³Department of Mathematics, Koneru Lakshmaiah Education Foundation, Vaddeswaram 522502, Andhra Pradesh, India

Classification generally assigns objects to enormous predefined categories and it is pervasive crisis that covers various application. Preparing the data for Classification and Prediction is the major problem in classification. In order to rectify this issue, an approximate function is proposed using IC-FNN. For acquiring correlated fuzzy rules and non-separable rules that comes under proper optimization problem. The extracted fuzzy rule's parameter was fine-tuned sourced on hierarchical Levenberg Marquardt (LM) learning method for enhancing performance. But here parameters of fuzzy rules aren't tuned perfectly. Hybridization of Ant Colony Optimization Genetic Algorithm (HACOGA) is proposed here to rectify these issues. It tunes the parameters of the extracted fuzzy rules. Hybridization is enforced to certain factors and ACO and GA variables that share same characteristics in the computation. Experimental results shows that proposed HACOGA assist in enhancing the performance of FNN with recall, precision, accuracy and *F*-measure for the Abalone age prediction dataset.

Keywords: Classification, Novel FNN, Correlated Contours, Fuzzy Rules, Hybrid Optimization, Correlated, Nonseparable, HACOGA.

1. INTRODUCTION

To spotlight on classes or categories, classification generally assigns items in a collection and targets at precisely identifying every case target values in data. Fuzzy inference systems (FIS) is one amongst popular applications of fuzzy-logic and fuzzy-set-theory, which acquires the diagnosis and off line method simulation, classification tasks, various online decision making modern tools and control methods. FIS specifically based on two fold identities. Else, it possesses an ability to manage the linguistic ideas. It is universal approximates, because it performs the non-linear mappings among outputs and inputs [1].

These two features assist in modelling two types of FIS design. 1st FIS targets on competency of fuzzy concept to design language processing [2]. FIS comprises of fuzzy concepts which are constructed from gaining expertise and we name it as fuzzy controllers or fuzzy expert systems, based on its last utilization. Expert knowledge was cast off to design knowledge engineering for simulation. Knowledge engineering works according to the classical

Boolean logic and it doesn't suit to control the progressiveness in the primary process phenomena [3, 4]. Here, gradual rules were brought-in in expert knowledge based simulators and it points human knowledge restriction, predominantly complexity in modelling complex process interactions.

Appropriate generalization capability and high semantic level were offered by FIS. Unfortunately, the tediousness of huge systems tends to inadequate accuracy in simulation outcomes. Expert knowledge works according to FIS, which gives poor performances. Parameter evaluation and structure identification were considered as two significant problems in building an FIS. The former one works on describing numerous fuzzy rules and it works on estimating deriving appropriate parameter values to build effectual system [5].

For acquiring correlated fuzzy and non-separable rules we bring-in proper optimization problem. But here parameters of fuzzy rules aren't adjusted correctly. Hybridization of Ant Colony Optimization Genetic Algorithm (HACOGA) is proposed here to rectify these issues. It tunes extracted fuzzy rules parameters. Hybridization

*Author to whom correspondence should be addressed.

enforces certain factors and GA and ACO variables, which in turn, share similar computation characteristics. The anticipated hybridization process is predominantly enhances FNN performance.

2. LITERATURE REVIEW

A FNN having novelty model with parallel fuzzy concepts (CFNN) which works on Levenberg–Marquardt (LM) optimization approach was anticipated by Ebadzadeh and Salimi-Badr [7]. This method gives an approximate non linear function, particularly the functions with greater correlation among the input variables with least amount of fuzzy rules. Multivariable Gaussian fuzzy membership function were established, which assumes the correlation among the input variables and it will design non-separable relations variables interaction. LM optimization approach helps to get the knowledge factors of both premise and fuzzy rule set. This algorithm helps us to enforce the seven tested samples along with time-series prediction, static function approximation, real-world complex regression problem and non-linear dynamic system identification.

A fresh fuzzy rule transfer approach for self-construction fuzzy neural inference networks was anticipated by Prasad et al. [8]. A novel FNN framework is basically merging an Auto Regressive with exogenous input (ARX) with nonlinear Tanh function in Takagi-Sugeno (T-S) kind fuzzy part was proposed by Zhang and Tao [9]. For optimizing parameters and structure of FNN under unknown plant dynamics simultaneously, an improved genetic algorithm is designed. ARX plus input structure the nonlinear function sub-model, number of fuzzy rules, and membership function parameters was enhanced by neighbourhood search operator, hybrid encoding/decoding and preserve operator. According to experimental output it is concluded that anticipated approach is better in structure simplification, generalization capability and modelling precision. Various types of classification on machine methods are defined with challenges, drawbacks with advantages are defined [26].

An efficient optimization technique was brought-in by Han et al. [10]. In learning process, this SASOA-FNN helps to systematize structure and manage parameters concurrently. Third, SASOA-FNN convergence is confirmed with updated and fixed structure and guidelines for parameters choosing. At last, experiments were executed on various nonlinear systems for checking the effectiveness. The comparison result states that proposed SASOA-FNN gives better performance.

An enhanced self-construction fuzzy neural inference network version, termed as soft-boosted SONFIN (SB-SONFIN) was proposed by Prasad et al. [11]. For minimizing the rate of error and for improving quick learning, design softly boosts learning process of SONFIN. By considering fuzzy rules and primary weights

SB-SONFIN improves SONFIN learning process, which has two significant SONFIN parameters, SB-SONFIN progresses learning procedure using: 1) weights initializing with fuzzy sets width indeed of random digits and 2) to enhance the rate of parameter with amount of fuzzy rules. On numerous real world and benchmark datasets, recommended soft boosting strategy is validated. According to experimental output it is concluded that SB-SONFIN possesses ability to surpass other known approaches on diverse datasets. In image processing to balance the energy accuracy for the quality improvement of images are defined [25].

An proficient learning method for constructing self-organizing FNNs was established by Han et al. [12], where we adapt algorithm having second order with malleable learning rate, parameters and it can be determined the network size with equal time in learning procedure. Initially, entire SOFNN parameters were adapted with help of SOA strategy for accomplishing the fast convergence through influential search scheme. Next, SOFNN structure may be self regulated with help of relative importance index of every rules. Later, a new layer is introduced to find the edge detection of the image processing [24].

Self-regulating FNN with adaptive gradient algorithm for nonlinear systems modelling was suggested by Han et al. [13]. Initially, fuzzy rules (PFR) potentially help to get normalized layer output and to reduce error ratio in the process of training. Structure learning concept is established to define the size of network sourced on PFR. Next, an adaptive gradient having malleable nature with flexible learning rate is created to correct SOFNN-AGA parameter. Furthermore, in self-organizing structure and fixed structure cases, theoretical analysis on SOFNN-AGA convergence is provided to demonstrate efficiency. Fuzzy method with HASO used to minimize the space and complexity of the FNN [27].

3. PROPOSED METHODOLOGY

Fuzzy Neural Network (FNN) was explained in this section for function approximation using 3 processes: initially, interpretable and intuitive fuzzy rules. Subsequently, according to fuzzy rules extraction, an appropriate framework is built. At last, initial values are fine-tuned according to Hybridization of Ant Colony Optimization Genetic Algorithm [HACOGA].

3.1. Extracting Intuitive Fuzzy Rules and Initial Interpretable

If the destination function surface (landscape in nature) is considered as landscape of the mountain, then every hill in landscape will be appropriate interpretable and intuitive fuzzy rule for approximate the function. So, the demanded model rules these should be non-separable and correlated.

These inseparably and correlated fuzzy concepts will be determined as below

$$\mu(Y) = e^{-(Y-M)^T} Q^{-1}(Y-M) \quad (1)$$

where the M specifies hill centre and Q specifies weight matrix to generate inseparably fuzzy rule correlated contours. Certainly, Q defines outline and control of fuzzy rule contours. Usually, hills centres in function landscape points superior output values in neighbourhoods. Hence, local maximums were chosen from training samples, is appropriate candidates for rule centres. Accordingly, rules centre, (M_j for j th rule), gets training samples [14]. Subsequently, matrix Q_j for j th fuzzy rule ($j = 1, 2, 3, \dots, r$), defines fuzzy rule contour to contours which is of covered hill in the landscape function. For haul out fuzzy concepts with contours same as closed region of destination function, matrix Q_j for j th fuzzy concept is determined on rectifying subsequent optimization crisis:

$$\rho: \begin{cases} \text{Max} & \zeta = \sum_{j=1}^r \sum_{k=1}^N \mu_j(X_k) \cdot y_k^* \\ \text{Subject to:} & |Q_j| = \rho_j \end{cases} \quad (2)$$

where $\mu_j(X_k) = e^{-(X_k - M_j)^T} Q_j^{-1}(Q_j^{-1}(x_j - M_j))$ is the fuzzy membership value of k th input sample X_k to j th fuzzy rule, y_k^* specifies appropriate result value of k th input sample (X_k), and ρ_j specifies scalar value with constant digit.

As the result training samples digits were treated as membership values weights in object function and the conditions could be stated for every fuzzy rule (FR_j):

$$\forall X_k, X_i \in FR_j: \begin{cases} \text{if } y_k^* = y_i^* \text{ then } \mu_i(X_k) \approx \mu_j(x_i) \\ \text{if } y_k^* > y_i^* \text{ then } \mu_i(X_k) > \mu_j(x_i) \end{cases} \quad (3)$$

To discard undesired optimum converging of τ ($Q_j = 0$), constraint ($|Q_j| = \rho_j$) is included. In order to rectify the optimization problem ρ determined in (2) by assuming constraints, novel augmented objective function according to Lagrange approach is demonstrated as trail:

$$\tau = \sum_{i=1}^R \sum_{k=1}^N \mu_i(x_k) \cdot y_k^* - \sum_{i=1}^R \eta_i (|Q_i| - \rho_i) \quad (4)$$

Which η_i is Lagrange multiplier it is essential for rectify subsequent equations

$$\frac{\partial \tau}{\partial \sigma_i} = 0, \quad \forall i = 1, 2, \dots, R \quad (5)$$

Sourced on (1), (5) and considering Q_i as symmetrical form of matrix, subsequent equations will be determined for i th fuzzy rule.

$$\frac{\partial \tau}{\partial Q_i} = \sum_{k=1}^N \mu_i(X_k) \cdot y_k^* Q_i^{-1} (X_k - M_i)(X_k - M_i)^T Q_i^{-1} \quad (6)$$

Now from (5) and (6) the following equation is obtained:

$$\sum_{k=1}^N \mu_i(X_k) \cdot y_k^* Q_i^{-1} (X_k - M_i)(X_k - M_i)^T Q_i^{-1} = \eta_i Q_i^{-1} \quad (7)$$

By multiplying Q_i twice to left and right sides of (7), Q_i will be computed as trail:

$$Q_i = \frac{1}{\eta_i} \cdot \sum_{k=1}^N \mu_i(X_k) \cdot y_k^* (X_k - M_i)(X_k - M_i)^T \quad (8)$$

By defining Q_i^{+i} as shown below:

$$Q_i^{+} = \frac{\sum_{k=1}^N \mu_i(X_k) \cdot y_k^* (X_k - M_i)(X_k - M_i)^T}{\sum_{k=1}^N \mu_i(X_k) \cdot y_k^*} \quad (9)$$

Equation (8) is rewritten as follows:

$$Q_i = Q_i^{+} \times \frac{\sum_{k=1}^N \mu_i(X_k) \cdot y_k^*}{\eta_i} \quad (10)$$

By substituting right-hand Eq. (10) in indicated constraint in Eq. (2), subsequent equations are attained:

$$\left(\frac{\sum_{k=1}^N \mu_i(X_k) \cdot y_k^*}{\eta_i} \right)^n |Q_i^{+}| = \rho_i \implies \frac{\sum_{k=1}^N \mu_i(X_k) \cdot y_k^*}{\eta_i} = \left(\frac{\rho_i}{|Q_i^{+}|} \right)^{1/n} \quad (11)$$

Sourced on Eqs. (9) and (10), appropriate value of Q_i is determined as trails:

$$Q_i = Q_i^{+} \times \left(\frac{\rho_i}{|Q_i^{+}|} \right)^{1/n} \quad (12)$$

Equations (9), (12) and $\mu_i(X_k)$ for input samples and fuzzy rule are iteratively computed, till objective function ζ convergence.

3.2. Designing FNN Structure Sourced on Extracted Interpretable and Intuitive Rules

Next acquiring matrix Q_i for every fuzzy rule, matrix Q_i is decomposed with respect to Eigen values and Eigen vectors as trails:

$$\begin{aligned} Q_i &= \Phi_i \Lambda_i \Phi_i^T \\ \implies Q_i^{-1} &= \Phi_i \Lambda_i^{-1} \Phi_i^T Q_i^{-1} = \Phi_i \Lambda_i^{-1/2} \Lambda_i^{-1/2} \Phi_i^T \\ \implies Q_i^{-1} &= (\Lambda_i^{-1/2} \Phi_i^T)^T (\Lambda_i^{-1/2} \Phi_i^T) \\ \implies Q_i^{-1} &= \Gamma_i^T \Gamma_i \end{aligned} \quad (13)$$

Where Φ_i specifies eigenvector matrix where every column is equal to Q_i eigenvector, Λ_i specifies matrix with eigen values, which is diagonal matrix that has Q_i eigen values, on diagonal, and Γ_i specifies matrix as trails:

$$\begin{cases} \Gamma_i = \Lambda_i^{-1/2} \Phi_i^T \\ Q_i^{-1} = \Gamma_i^T \Gamma_i \end{cases} \quad (14)$$

Similar to the PCA, it is feasible to get rid of the Eigen values lower when compared with a pretend threshold and respective eigenvectors from Λ_i and Φ_i , correspondingly, for providing the dimension reduction ability. Dimension reduction for every fuzzy rule causes rise in generalization by avoiding small eigen values.

Based on Eq. (14), inseparably rule determined in Eq. (1) is rewritten sourced on Eq. (14) in i th rule as trails:

$$\begin{aligned}\mu_i(X) &= e^{-(X-M_i)^T Q_i^{-1} (X-M_i)} \\ &= e^{-(X-M_i)^T \Gamma_i^T \Gamma_i (X-M_i)} \\ &= e^{-\|\Gamma_i(X-M_i)\|^2} \\ &= e^{-\|Z_i - \psi_i\|^2} \\ &= \prod_{j=1}^{n_i} e^{-(z_{ij} - \psi_{ij})^2} \\ &= \prod_{j=1}^i \mu_{ij}\end{aligned}\quad (15)$$

where Γ_i specifies transform matrix to indicate i th non-separable fuzzy rule as separable one, $Z_i = \Gamma_i^T X$ specifies extracted high level feature vector for i th fuzzy rule, ψ_i specifies transferred center of i th fuzzy rule and z_{ij} and ψ_{ij} specifies j th dimension of Z_i and ψ_i [14]. With Eq. (15), it confirms inseparably fuzzy rule can be specified as separable fuzzy rule based on high-level extracted features Z_i .

The anticipated process acquires linear transform matrices (Γ_i) which is same as if PCA, however we have numerous significant variations among them. Decomposed matrix in PCA is covariance matrix with illustration of complete training input samples (devoid of determining output training samples), however decomposed matrix in anticipated approach is weight matrix Q_i that specifies outline and control of i th fuzzy rule contours. Sourced on output and input training samples Matrix Q_i is measured, which belongs to i th fuzzy rule and not entire sample trainings. Hence, basic PCA idea differs from suggested approach. Significant differences among anticipated approach and PCA is indicated as trails:

PCA decomposed matrix is computed according to taking in training samples, however decomposed matrix Q_i is attained on output and taking in training samples;

PCA decomposed matrix is computed on entire taking in training samples, however matrix Q_i is attained sourced on taking in and taking out training samples which comes under i th fuzzy rule;

PCA decomposed matrix is sample covariance matrix, however matrix Q_i specifies matrix that specifies outline and control of i th fuzzy rule contours.

Result enforces PCA and anticipated approach with 'm' extracted fuzzy rules contours are performed by every approach. outline and control of destination function covered region contours couldn't be modelled, by hauling out linear transform for every fuzzy rule sourced on PCA.

It is feasible to fuzzy sets shapes for every high-level obtained lineaments of every fuzzy concept by including outline control parameter β as trails:

$$\mu_i^+(X) = \prod_{j=1}^{n_i} \mu_{ij}^{\beta_{ij}} \quad (16)$$

Where μ_{ij} is derived as follows:

$$\mu_{ij} = e^{-(z_{ij} - \psi_{ij})^2} \quad (17)$$

Fuzzy sets shapes are flexible and considered as usual Gaussian form. Hence, it is feasible to outline the fuzzy sets among semi-triangular and semi-trapezoidal mode. Moreover, it is understandable if β is equal to 1, membership function is Gaussian. If parameter is selected to be superior than 1 and it is considered to be nearer to trapezoidal and selected as lesser than 1, function shaper will be nearer to triangular. Hence, it is parameter (β) the regulator. Determining the membership function to diverse forms in membership function can be proficient to offer fuzzy rules of contours with complex outlines. Figure 1 depicts diverse fuzzy sets shapes defined according to anticipated membership function for diverse Regulator parameter (β) values. Moreover, figure depicts samples of different feasible contours shapes of fuzzy rules sourced on expansion principle for 2 high-level lineaments and non-interactive.

This representation can haul out suitable and high-level independent lineaments to create each fuzzy rule, assuming association with input variables with every other and output variable. Therefore, i th intuitive fuzzy rule is demonstrated as trails:

FR_{*i*}: if X belongs to the i th hill then y is ω_i

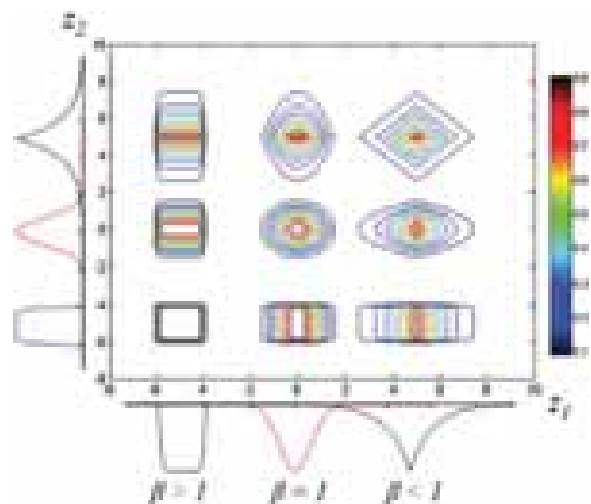


Fig. 1. Diverse probable outlines produced fuzzy sets and diverse fuzzy rules contours with various values of β for two high-level features z_1 and z_2 [14].

Where i th hill in function landscape is $\mu_i^+(X)$. Thus, it is feasible to re-formulate Rule _{i} as trails:

$$\text{FR}_i: \text{if } X \in \mu_i^+(X) \Rightarrow y \text{ is } \omega_i \quad (18)$$

All above Figures 1–3 are considered for the purpose of clearing the concepts of FNN. Input layer, Transformation layer, Fuzzy sets layer, Fuzzy rules layer, and Output layer (see Fig. 3) [14] are 5 diverse layers of anticipated model.

Output layer acts as an essential role in determining subsequent fuzzy rules part, and transformation (conversion) layer, fuzzy sets and fuzzy rules layers works as premise parts. Demonstration of every layer is elaborated below trails:

I-Input layer: Taking in variables of network are layers result values and specified as depicted below:

$$X = [x_1, x_2, \dots, x_n]^T \quad (19)$$

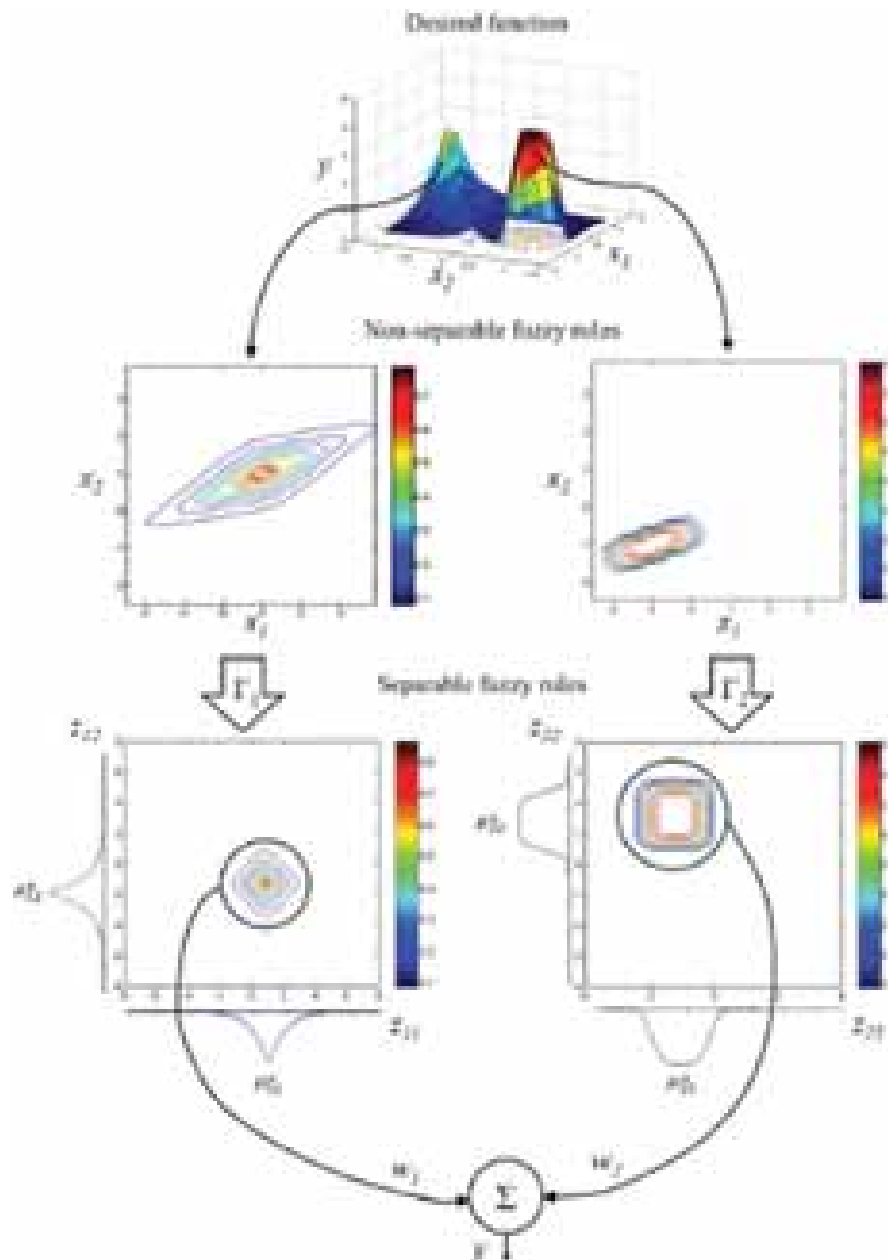


Fig. 2. Inseparably fuzzy rules extraction for interactive variables using transformation to diverse new spaces with novel high-level non-interactive lineaments, and demonstrating separably fuzzy concepts in novel spaces [14].

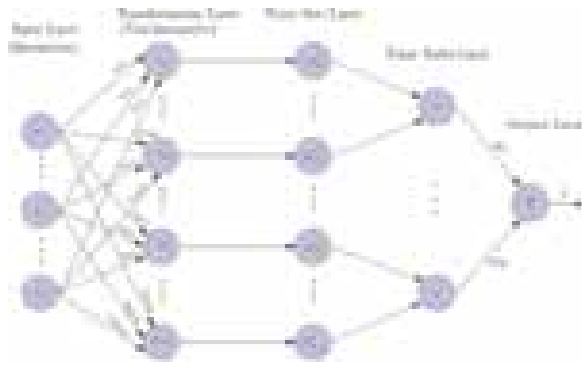


Fig. 3. The proposed fuzzy neural network architecture [14].

II-Transformation (conversion layer) layer: Every neuron of the layer enforces linear transformation or conversion on the intake variables to map the initial interactive input variables. For i th fuzzy rule, the transformation is

$$\begin{cases} Z_i = \Gamma^T X \\ \psi_i = \Gamma_i^T M_i \end{cases} \quad (20)$$

III-Fuzzy sets layer: Here, fuzzy sets with various shapes were determined for every hauling out high-level outline in fuzzy laws. The value of membership function for j th dimension of i th high-level outline vector (i th fuzzy law) is depicted as trails:

$$\mu_{ij}^+ = A(z_{ij}, \psi_{ij}, \beta_{ij}) = \mu_{ij}^{\beta_{ij}} = e^{-((Z_{ij} - \psi_{ij})^2) \beta_{ij}} \quad (21)$$

IV-Rules of Fuzzy or Fuzzy rules layer: Here every neuron indicates a fuzzy rule. It calculates a membership value according to the T-Norm operator which is enforced on previous layer output. As features extracted using transformation (conversion) layer are reciprocal, it is satisfactory to make use of T-Norm operator for constructing fuzzy laws in extracted high-level outline spaces.

$$\mu_i^+(X) = \prod_{j=1}^{n_i} \mu_{ij}^+ \quad (22)$$

V-Output (resultant layer) layer: It is having one linear neuron in final layer that offers network output and functions as an essential role of parameters of subsequent part. Last network output is evaluated as trails:

$$y = \sum_{i=1}^R \mu_i^{+\omega_i} \quad (23)$$

To initialize network factors, primarily local maximum points are hauled out from training samples as fuzzy rules centres [12]. For fuzzy rule, like structure has part parameter (W), n_i parameters of fuzzy rule centre (ω), n_i regulator parameters (β), and $n \times n_i$ transformation matrix (T) parameter. Subsequently, the parameters, p for anticipated structure is determined as trails:

$$p = R + (n + 2) \sum_{i=1}^R n_i \quad (24)$$

3.3. Fine-Tuning Method Using Hybrid Optimization Method

ACO and Hybrid GA is novel algorithm to rectifying the parameter tuning in establishing new fuzzy neural network.

3.3.1. Genetic Algorithm

For simulating the evolution procedure and organism natural selection [15], Genetic Algorithm (GA) is computational technique which is modelled to pursue series as producing initial evaluation, population, crossover, selection, mutation, and regeneration [16]. Initial population turn into significant solution and it is usually produced randomly. Mutation and Crossover will be performed to acquire novel and superior ones. GA idea is novel solution generation should be superior when contrast to previous one. This procedure is iterated till attaining certain stopping criteria. GA is significantly suitable in diverse optimization crisis as this approach doesn't need superior initial knowledge of problem which is rectified in nature; which is treated as a benefit of use of GA. This algorithm is possible to know the global optimum and utilized to problem [17, 18].

3.3.2. Ant Colony Optimization (ACO)

In ACO, Ants uses some pheromone during walking in line and broadcast with other ant in searching. Those who couldn't smell pheromone moves at random route. When huge ants are pleasingly tracking shortest one, pheromones of certain path improves, and ant will choose subsequent city in accordance to particular probability, which is distance matrix, pheromone matrix function and parameters. This process iterates till every ant visits each city one time. This is initial cycle in algorithm, which proceed till it reaches end criteria. Complete route is modified as pheromone matrix is updated in each cycle. To solve parameter tuning [8], ACO is dedicated. However, this algorithm has different weaknesses, as like its performance works according to the previous cycle, and it is easy to convergent and stagnant, then requires a huge processing time. And this creates great dispute for searching space and ACO computation time.

3.3.3. Hybridization of Ant Colony Optimization and Genetic Algorithm [HACOGA]

In Hybridization, GA gains merits to begin pheromone in ACO matrix and rejoin ACO route [19]. Here, GA chromosome helps to optimize amount of subsequent sample for classification from every ant and for discarding dependency on previous cycle. The diversity of the ant's tour can be preserved [20] from these chromosomes. Hybridization is shortening in choosing changeable ACO parameter which requires human knowledge and depends on coincidence. ACO helps in temporary solution generator which improves the implementation by executing GA operators

iteratively until we reach end criteria. Anticipated GA and ACO hybridization is somewhat effective by expanding the crossover operator in GA (EXO) to generate superior output produced by ACO. Here we get two steps for this solution. Initially, we make use of the ACO to produce solution that acquires local maxima. Solution generated in this step is iterated and independent to one another. ACO local optima solution is considered as GA population, for attaining superior global optima. Tweak probability to route outcomes in ACO to high, through this proposed work. Subsequent section offers details of anticipated work.

3.3.4. Hybridization Technique

Idea of this work is to determine a proper way to hybridize ACO and GA for recognizing FNN solution to build the combining specific process of Genetic Algorithm and ACO to carry out GACO. Hybridization is enforced to few variables and parameters of GA or ACO to share some computation characteristics, i.e., GA the size of population and ants numbers in ACO, number of cycles in ACO, number of generations in GA and GA chromosome. Modification is represented in as shaded drawing [21, 22].

3.3.5. HACOGE

Demonstrating process as below:

1. The cities which are not visited and their indices are determined as set termed allowed.
2. The city which is primarily visited is illustrated with no. of gene 1 in initial gene groups openly.

If no. of gene 1 is c , then initially visited city is c th index in permitted. If there is no gene in primary group, then initially visited city is last index in facilitates.

3. Subsequently, quantity of gene(s) 1 is utilized in subsequent genes and Eq. (26) to demonstrate cities that is visited one after another.
4. As city will be defined as visited, then city index is eliminated from allowed.
5. Terminal index enduring in allowed is final visited city. In essential GA, chromosome specifies fuzzy neural network solution indirectly.

To resolve 'n' FNN parameter tuning, the representation of chromosome is modelled as trails:

1. Every chromosome comprises of $(n - 1)$ gene groups.

$$N_{\text{gen}} = \sum_{i=1}^{n-1} i \quad (25)$$

2. The quantity of gene present in i th group is similar to $(n - i)$ bit, therefore the sum of amount group of genes (N_{gen}) in chromosome (usage of Eq. (24)).

Number of bit 1 is utilized in i th gene group to demonstrate cities that is probable to be visited using ants.

Process to decode chromosome is provided to fuzzy neural network solution:

1. Nonvisited city indices are depicted as a set termed allowed.
2. i th city (visited) is demonstrated using quantity of genes 1 in i th gene groups. If number of gene 1 is k , then i th city (visited) is k th index in permitted. In case of i th, group, if no gene is there, then i th city (visited) is final index in permitted.
3. If we will arrange the city as a set visited, city index is removed from allowed.

$$\rho_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{k \in \text{allowed}_k} [\tau_{ik}(t)]^\alpha \cdot [\eta_{ik}]^\beta} & \text{if } j \in \text{allowed}_k \\ 0 & \text{otherwise} \end{cases} \quad (26)$$

Here α and β specifies parameters that manage relative significance of trail versus visibility, k defines the ant index, permitted k is set of nonvisited city that possibly visited using k th ant, i specifies current terminal visited city, j specifies the indices of subsequent visited city that visited from i th city, t specifies time index, τ specifies trail intensity, η and is visibility.

4. Last city is final index remaining in permitted [23].

4. RESULTS AND DISCUSSION

The current section examines experimental output of proposed method. The model is executed with the help of JAVA. When distinguished with the current IC-FNN algorithm with the proposed HACOGE algorithm are compared to recall, precision, accuracy and F -measure for abalone age prediction dataset. Abalone age is defined through cutting shell of cone, staining, and counting number of rings via microscope-time-consuming and boring task. Other factors, which are simpler to acquire, are utilized to identify age. In Ref. [27], the attributes information of abalone with data type, measurement explanations of individuals terms are done very nicely. From the above information, it is easy to calculate and predict the age of abalone.

5. PERFORMANCE METRICS

Precision points to percentage of results, which are appropriate and is depicted as,

$$\text{Precision} = \frac{\text{True positive}}{\text{true positive} + \text{false positive}} \quad (27)$$

Recall points to percentage of entire relevant outcomes accurately categorized with anticipated algorithm which are determined as

$$\text{Recall} = \frac{\text{True positive}}{\text{true positive} + \text{False Negative}} \quad (28)$$

To describe harmonic precision mean and recall value F -measure is used and is defined by

$$F\text{-measure} = \frac{\text{True positive}}{\text{true positive} + \text{False Negative}} \quad (29)$$

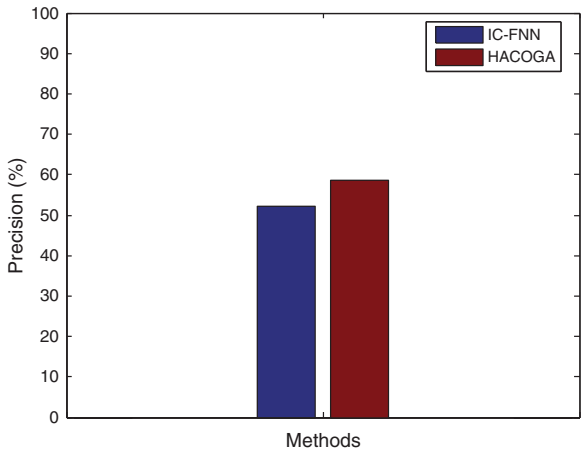


Fig. 4. Precision results versus classification methods.

Table I. Performance comparison result values.

Methods	Precision (%)	Recall (%)	F-measure (%)	Accuracy (%)
IC-FNN	52.2411	74.8435	61.5323	83.3213
HACOGA	58.7630	84.7639	69.4083	89.4052

Accuracy is a metric for evaluating classification models. Generally, accuracy is predictions fractions for this model. Officially, accuracy has subsequent definition:

Accuracy = (True positive + True Negative) / Total (30)

Figure 4 explains the comparative outcomes of anticipated HACOGA method and present IC-FNN method with respect to precision. From outcome, it confirmed that anticipated HACOGA model generated superior precision

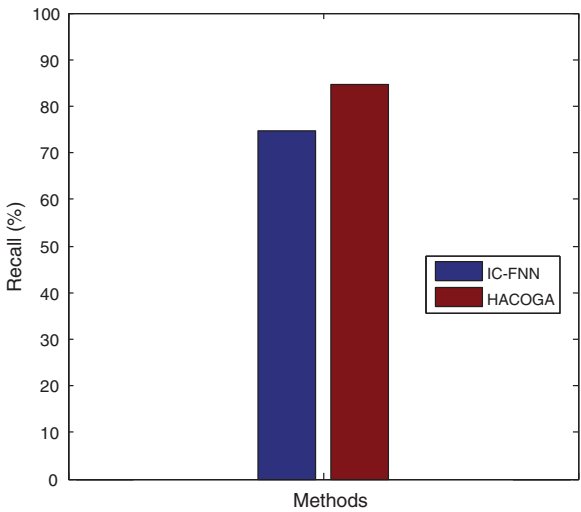


Fig. 5. Recall results versus classification methods.

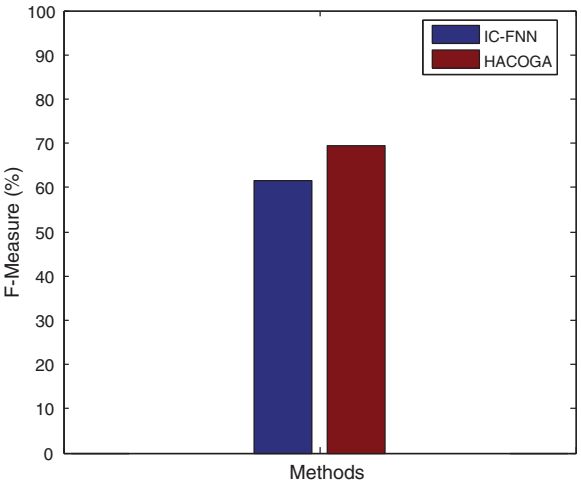


Fig. 6. F-measure results versus classification methods.

results of 58.7630% whereas current IC-FNN method gives only 52.2411% (see Table I) correspondingly.

Figure 5 provides performance comparison outcomes of anticipated HACOGA method and current IC-FNN method with respect to Recall. From outcomes, it confirmed that anticipated HACOGA model offers superior recall outcomes of 84.7639% where current IC-FNN method generates only 74.8435%, correspondingly.

Figure 6 gives performance comparison outcomes of anticipated HACOGA method and current IC-FNN method with respect to F-measure. Based on the output, it confirms that the proposed HACOGA model generates higher F-measure results of 69.4083% whereas current IC-FNN method produces only 61.5323% correspondingly.

Figure 7 provides performance comparison outcomes of anticipated HACOGA method and current IC-FNN method with respect to accuracy. Based on the output, it confirms that the proposed HACOGA model generates higher

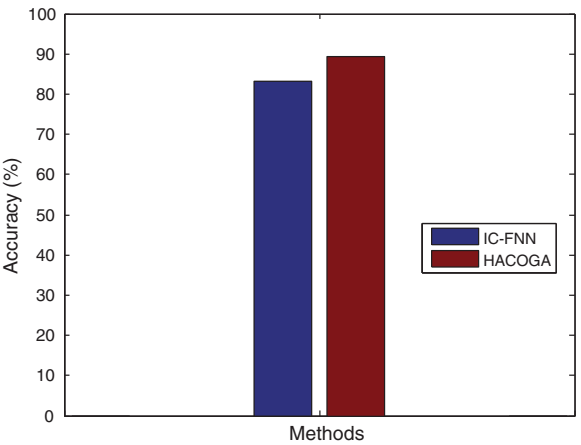


Fig. 7. Accuracy results versus classification methods.

accuracy results of 89.4052% whereas current IC-FNN method produces only 83.3213% correspondingly.

6. CONCLUSION AND FUTURE WORK

Here, an appropriate hybrid optimization problem is presented with the help of parallel and inseparably fuzzy rules. To outline contours with various shapes, novel shape able membership function with adaptive shape is brought-in to determine fuzzy sets. Subsequently, according to the hybrid optimization approach, extracted fuzzy rules parameters are fine-tuned. Anticipated method performance is computed in time-series prediction and real-world regression problems and distinguished with subsequent current approaches. Based on experiments, the anticipated approach could build parsimonious structures with superior accuracy, in comparison to the current approaches. In future work shall examine the Fuzzy Neural network to beat the demerits of proposed model and also makes use of the subset method based Fuzzy Neural Network model in the field of image compression and control.

References

- Chaka, K., **2017**. Introduction to fuzzy system interpretability. in *Design of Interpretable Fuzzy Systems*, Cham, Switzerland, Springer. pp.27–36.
- Alonso, J.M., Castiello, C. and Mencar, C., **2015**. Interpretability of fuzzy systems: Current research trends and prospects. in *Springer Handbook of Computational Intelligence*. Berlin, Heidelberg, Springer. pp.219–237.
- Azar, A.T. and Hassaniien, A.E., **2015**. Dimensionality reduction of medical big data using neural-fuzzy classifier. *Soft Computing*, 19(4), pp.1115–1127.
- Mendel, J.M., **2017**. Uncertain rule-based fuzzy systems. in *Introduction and New Directions*, Cham, Switzerland, Springer International Publishing. p.684.
- Cpalka, K., **2017**. *Design of Interpretable Fuzzy Systems*. Heidelberg, Springer. Vol. 684.
- Wai, R.J., Chen, M.W. and Liu, Y.K., **2015**. Design of adaptive control and fuzzy neural network control for single-stage boost inverter. *IEEE Transactions on Industrial Electronics*, 62(9), pp.5434–5445.
- Ebadzadeh, M.M. and Salimi-Badr, A., **2015**. CFNN: Correlated fuzzy neural network. *Neurocomputing*, 148, pp.430–444.
- Prasad, M., Lin, Y.Y., Lin, C.T., Er, M.J. and Prasad, O.K., **2015**. A new data-driven neural fuzzy system with collaborative fuzzy clustering mechanism. *Neurocomputing*, 167, pp.558–568.
- Zhang, R. and Tao, J., **2018**. A nonlinear fuzzy neural network modeling approach using an improved genetic algorithm. *IEEE Transactions on Industrial Electronics*, 65(7), pp.5882–5892.
- Han, H., Wu, X., Liu, H. and Qiao, J., **2018**. An Efficient optimization method for improving generalization performance of fuzzy neural networks. *IEEE Transactions on Fuzzy Systems*, 27(7), pp.1347–1361.
- Prasad, M., Lin, C.T., Li, D.L., Hong, C.T., Ding, W.P. and Chang, J.Y., **2017**. Soft-boosted self-constructing neural fuzzy inference network. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(3), pp.584–588.
- Han, H., Zhang, L., Wu, X. and Qiao, J., **2017**. An efficient second-order algorithm for self-organizing fuzzy neural networks. *IEEE Transactions on Cybernetics*, (99), pp.1–13.
- Han, H.G., Lin, Z.L. and Qiao, J.F., **2017**. Modeling of nonlinear systems using the self-organizing fuzzy neural network with adaptive gradient algorithm. *Neurocomputing*, 266, pp.566–578.
- Ebadzadeh, M.M. and Salimi-Badr, A., **2018**. IC-FNN: A novel fuzzy neural network with interpretable, intuitive, and correlated-contours fuzzy rules for function approximation. *IEEE Transactions on Fuzzy Systems*, 26(3), pp.1288–1302.
- Ghamisi, P. and Benediktsson, J.A., **2015**. Feature selection based on hybridization of genetic algorithm and particle swarm optimization. *IEEE Geosciences and Remote Sensing Letters*, 12(2), pp.309–313.
- Bi, W., Dandy, G.C. and Maier, H.R., **2015**. Improved genetic algorithm optimization of water distribution system design by incorporating domain knowledge. *Environmental Modelling & Software*, 69, pp.370–381.
- Cerrada, M., Zurita, G., Cabrera, D., Sánchez, R.V., Artés, M. and Li, C., **2016**. Fault diagnosis in spur gears based on genetic algorithm and random forest. *Mechanical Systems and Signal Processing*, 70, pp.87–103.
- Yuan, X., Elhoseny, M., El-Minir, H.K. and Riad, A.M., **2017**. A genetic algorithm-based, dynamic clustering method towards improved WSN longevity. *Journal of Network and Systems Management*, 25(1), pp.21–46.
- Sama, M., Pellegrini, P., D'Ariano, A., Rodriguez, J. and Pacciarelli, D., **2016**. Ant colony optimization for the real-time train routing selection problem. *Transportation Research Part B: Methodological*, 85, pp.89–108.
- Fetanat, A. and Khorasaninejad, E., **2015**. Size optimization for hybrid photovoltaic-wind energy system using ant colony optimization for continuous domains based integer programming. *Applied Soft Computing*, 31, pp.196–209.
- Serpell, M., Smith, J.E., **2010**. Self-adaptation of mutation operator and probability for permutation representations in genetic algorithms. *Evolutionary Computation*, 18(3), pp.491–514.
- Khanbary, L.M.O. and Vidyarthi, D.P., **2009**. Modified genetic algorithm with threshold selection. *International Journal of Artificial Intelligence*, 2(S09), pp.14–26.
- Hung, K.S., Su, S.S. and Lee, Z.J., **2007**. Improving ant colony optimization algorithms for solving traveling salesman problems. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 11(4), pp.433–442.
- Pramoda Patro, Krishna Kumar and Suresh Kumar, G., **2018**. Applications of three layer CNN in image processing. *Journal of Advanced Research in Dynamical and Control Systems*, 1, pp.510–512.
- Pramoda Patro, Krishna Kumar and Suresh Kumar, G., **2017**. Cellular neural network, fuzzy cellular neural network and its applications. *International Journal of Control Theory and Applications*, 10, pp.161–168.
- Pramoda Patro, Krishna Kumar and Suresh Kumar, G., **2018**. Various classifiers performance based machine learning methods. *International Journal of Recent Technology and Engineering*, 8(3S3), pp.305–310.
- Pramoda Patro, Krishna Kumar and Suresh Kumar, G., **2020**. Neuro fuzzy system with hybrid ant colony particle swarm optimization (HASO) and robust activation. *Journal of Advanced Research in Dynamical & Control Systems*, 12, pp.741–750.

Received: 20 March 2020. Accepted: 2 April 2020.