IJST, Transactions of Electrical Engineering, Vol. 37, No. E1, pp 67-77
Printed in The Islamic Republic of Iran, 2013
© Shiraz University

A NEW CLUSTERING-BASED APPROACH FOR MODELING FUZZY RULE-BASED CLASSIFICATION SYSTEMS*

F. FARAHBOD AND M. EFTEKHARI**

Dept. of Computer Engineering, Shahid Bahonar University, Kerman, I. R. of Iran
Email: m.eftekhari@mail.uk.ac.ir

Abstract– In the present study, we propose a novel clustering-based method for modeling accurate fuzzy rule-based classification systems. The new method is a combination of a data mapping method, subtractive clustering method and an efficient gradient descent algorithm. A data mapping method considers the intricate geometric relationships that may exist among the data and computes a new representation of data that optimally preserves local neighbourhood information in a certain sense. The approach uses subtractive clustering method to extract the fuzzy classification rules from data; the rule parameters are then optimized by using an efficient gradient descent algorithm. Twenty datasets taken from UCI repository are employed to compare the performance of the proposed approach with the other similar existing classifiers. Some non-parametric statistical tests are utilized to compare the results obtained in experiments. The statistical comparisons confirm the superiority of the proposed method compared to other similar classifiers, both in terms of classification accuracy and computational effort.

Keywords– Pattern classification, Fuzzy rule extraction, Subtractive clustering

1. INTRODUCTION

In recent years, fuzzy models have been used widely because they are able to work with imprecise data, handle the complex nonlinear problems and acquired knowledge with these models is more interpretable than the black-box models. Fuzzy Rule-Based Classification System (FRBCS) is a special case of fuzzy modeling and focuses on finding a compact set of fuzzy if-then classification rules to model the input output behaviour of the system. In the design of FRBCS, construction of rule-base is the most challenging problem. It is desirable that the rule-base covers all the states of the system and at the same time, the number of rules should be kept low to increase the generalizing ability of the model, and to ensure a compact and transparent model.

Many approaches have been proposed to construct the rule-base from numerical data. These include heuristic approaches [1, 2], neuro-fuzzy techniques [3-6], genetic algorithms [7-11], clustering methods [12-22] and data mining techniques [23, 24]. Clustering-based rule extraction methods help avoid combinatorial explosion of rules with increasing dimension of the input space, because the resultant rules are scattered in the input space rather than placed according to grid-like partitions in the input space. Each cluster can be considered as a fuzzy rule and essentially identifies a region in the data space that contains a sufficient mass of data to support the existence of a fuzzy input output relationship.

Many approaches have been proposed that use clustering methods for learning fuzzy if-then classification rules from numerical data. Below, some of them are mentioned.

Yager and Filev developed a simple method based on the mountain clustering for generation of fuzzy rules [25]. Han et al. proposed an improved fuzzy neural network based on Takagi-Sugeno (T-S) model.

*Received by the editors August 4, 2012; Accepted April 8, 2013.
**Corresponding author

Subtractive clustering is used for extracting rules from data [13]. Zhao et al. proposed a two-stage approach to extract compact Takagi-Sugeno (TS) fuzzy models using subtractive clustering and particle swarm optimization (PSO) from numeric data [15]. Subtractive clustering was employed to extract a fuzzy rules base and the PSO algorithm is used to search the optimal rule parameters simultaneously. Eftekhari and Katebi proposed a hybrid approach for building optimal fuzzy model from data for nonlinear dynamical systems [16]. This approach was a combination of a genetic algorithm, subtractive clustering and unscented filter. Subtractive clustering has been used to construct rule base and unscented filter has been employed for optimization of model parameters. Demirli et al. used the subtractive clustering method as a system identification tool to model job sequencing problems from an existing sequence (output data) and job attribute data (input data) [26]. Torun and Tohumoglu developed a new systematic way in order to obtain optimized fuzzy inference system for classification [12]. Subtractive clustering has been used to construct rule base and simulated annealing is employed for optimization of classifier parameters. Elmzabi et al. proposed a method in order to generate the Mamdani fuzzy inference systems [27]. This method used the results of the subtractive clustering in order to generate the Mamdani fuzzy rules and the genetic algorithms for the parameters optimization of these rules. Zhang and Lu proposed a method for creating ensembles of classifiers via Fuzzy C-Mean (FCM) clustering [28]. Radha and Rajagopalan in [18] and Chen in [29] proposed methods that used FCM to generate fuzzy rules from data to deal with data classification problem. Hossen et al. proposed a novel modified adaptive fuzzy inference system, which automatically generates fuzzy membership functions via the FCM clustering and fuzzy rules from the modified Apriori algorithm based on input-output data sets [30]. Zhang et al in [19] and Zhang and Liu in [20] proposed an enhanced clustering algorithm, IFCM, which originates from traditional FCM algorithm and can process with interval sets. They showed that the IFCM algorithm can be used to extract fuzzy rules for an interval type-2 fuzzy logic system.

In this paper, first, a data mapping method is employed. This method uses laplacian eigenmaps for data representation and builds a graph incorporating neighbourhood information, using the notation of laplacian of graph for computing new representation. Then, new representation of data is used as input data for building FRBCS based on a clustering approach. We use subtractive clustering method and an efficient gradient descent method for constructing a compact and optimal rule-base.

The remainder of this paper is organised as follows. In Section 2, the proposed new clustering-based method for modeling FRBCSs is presented. Experiment results are reported in Section 3. Finally, the article is concluded in Section 4.

2. PROPOSED METHOD

In this section, basic stages of the proposed method are explained. The flowchart of the proposed algorithm is shown in Fig. 1. Part 2 is performed after part 1. Therefore in part 2, new representation of data is utilized.

a) Computing new representation of the data

Part 1 shows the steps of the proposed algorithm used to compute new representation of the data. In this stage, we compute a new representation of the data. First, each attribute of data points is rescaled to unit interval [0, 1] by use of a linear transformation. Then, the adjacency matrix $A \in R^{n\times n}$ is formed. Each element of A is defined by $A_{ij} = exp(-\left\|x_i - x_j\right\|^2)$ if $i \neq j$, and $A_{ii} = 0$. In the next steps, the diagonal matrix D whose *(i, i)*-element is the sum of *i-th* row of A ($D_{ii} = \sum_j A_{ij}$) and the scaled adjacency matrix ($A' = D^{-1/2} A D^{-1/2}$) are constructed. We find $y_1, y_2, ..., y_k$, the k largest eigenvectors of A' (chosen to be orthogonal to each other in the case of repeated eigenvalues), and construct the matrix $Y = [y_1, y_2, ..., y_k] \in R^{n\times k}$ by stacking the eigenvectors in columns. Then, each row of Y is rescaled to unit

interval [0, 1] and is treated as a point in R^k. New data representation is used as input data for modeling the FRBCS in other stages. Usually, algorithms that are used for classification problems don't consider the intricate geometric relationships that may exist among the data. In this work, we use a data mapping method that considers the intricate geometric relationships that may exist among the data and computes a new representation of data that optimally preserves local neighbourhood information in a certain sense.

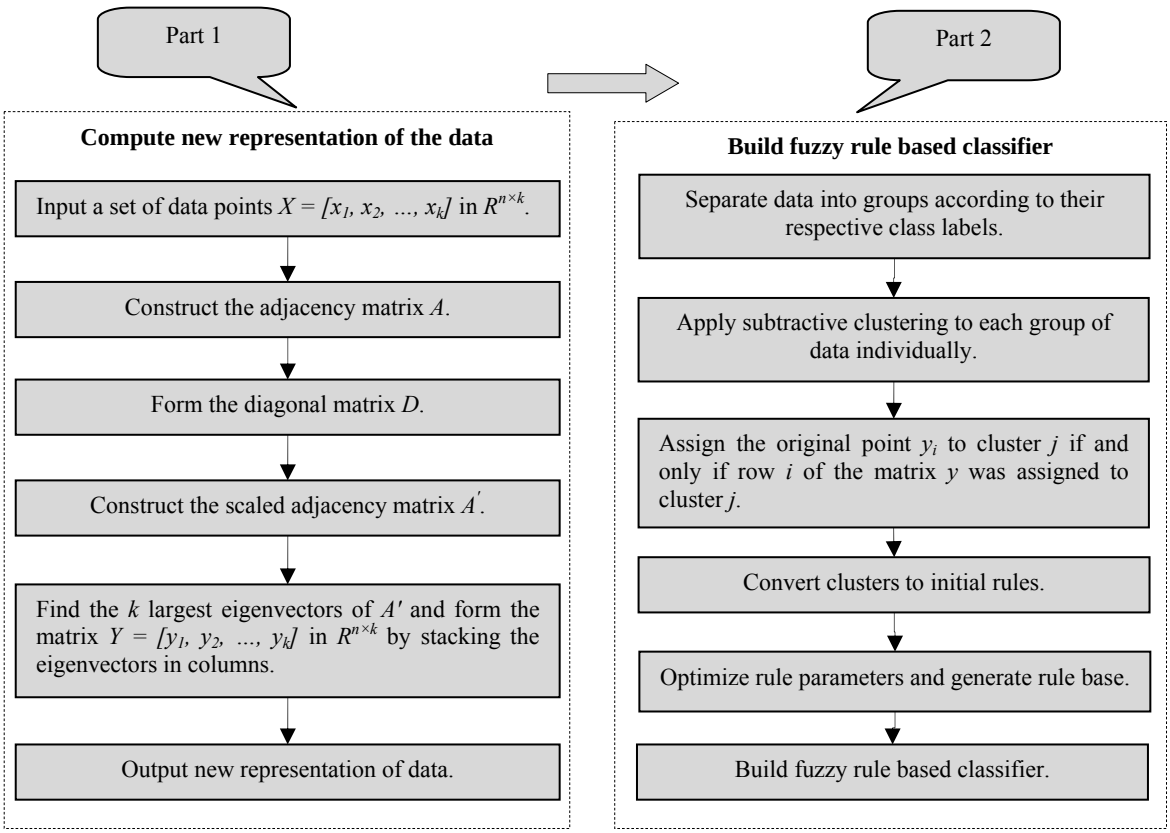

Fig. 1. Proposed algorithm flowchart

b) Building fuzzy rule based classification system

Part 2 consists of the most general steps of the proposed method in modeling FRBCS. In the first step in this part, we separate the data into groups according to their respective class labels. Then we apply subtractive clustering to each group of data individually to extract the rules for identifying each class of data. The cluster centers found in each group of a specific class are points in the feature space that their neighborhood will map into the given class. In the next step, we convert each cluster center into a fuzzy rule. After the initial rules have been obtained, we use an efficient gradient descent algorithm to tune the parameters of membership functions to minimize the classification error. The individual sets of rules are then combined to form the rule base of the classifier. When performing classification, the consequent of the rule with the highest degree of fulfillment is selected to be the output class of the classifier.

1. The subtractive clustering method: The subtractive clustering method, proposed by Chiu [32], is an extension of the grid-based mountain clustering method introduced by Yager and Filev [33, 34]. In subtractive clustering method, data points are considered as the candidates for cluster centers. Consider a collection of n data points $\{x_1, x_2, ..., x_n\}$, where x_i is a vector in the feature space. For this method, data points have to be rescaled into [0, 1] in each dimension. We consider each data point as a possible cluster center and define a measure of the potential of data point x_i as:

$$P_i = \sum_{j=1}^{n} exp(-\alpha \left\| x_i - x_j \right\|^2), \alpha = 4/r_a^2 , \quad (1)$$

where $\|.\|$ denotes the Euclidean distance, and r_a is a positive constant. The constant r_a is effectively a normalized radius defining a neighbourhood; data points outside this radius have little influence on the potential. Thus, the measure of the potential of a data point is a function of its distances to all other data points.

After the potential of each data point has been computed, the data point with the highest potential is selected as the first cluster center. Let x_1^* be the location of the first cluster center and P_1^* be its potential value. We then revise the potential of each data point x_i by the formula as follows:

$$P_i = P_i - P_1^* exp(-\beta \left\| x_i - x_1^* \right\|^2), \beta = 4/r_b^2 , r_b = 1.25\ r_a . \quad (2)$$

We subtract an amount of potential from each data point as a function of its distance from the first cluster center. The data points near the first cluster center will have greatly reduced potential, and therefore would be unlikely to be selected as the next cluster center. The constant r_b is a radius defining the neighbourhood which will have measurable reductions in potential.

When the potentials of all data points have been revised, the data point with the highest potential value is selected as the second cluster center. The process is then continued further. In general, after the kth cluster center has been obtained, the potential of each data point is revised by the formula as:

$$P_i = P_i - P_k^* exp(-\beta \left\| x_i - x_j^* \right\|^2), \quad (3)$$

where x_k^* is the location of the k-th cluster center and P_k^* is its potential value. The process of acquiring new cluster center and revising potentials repeats until the remaining potential of all data points falls below some fraction of the potential of the first cluster center. In addition to this criterion for ending the clustering process are criteria for accepting and rejecting cluster centers that help avoid marginal cluster centers [35].

2. Converting clusters to initial rules: Each cluster center may be translated into a fuzzy rule. Suppose cluster center x_i^* was found in the group of data for class c_m, this cluster center provides the rule:

$Rule\ i: if\ X_1\ is\ A_{i1}\ and\ X_2\ is\ A_{i2}\ and \ldots then\ class\ c_m.$

Where X_j is the j-th input feature and A_{ij} is the membership function in the i-th rule associated with the j-th input feature. The membership function A_{ij} is given by the formula as:

$$A_{ij}(X_j) = exp\{-1/2 (x_j - x_{ij}^*/\sigma_{ij})^2\}, \ \sigma_{ij}^2 = 1/2\alpha , \alpha = 4/r_a^2 , \quad (4)$$

where x_{ij}^* is the j-th element of x_i^*, and r_a is a positive constant. The degree of fulfillment of each rule is defined as:

$$\mu_i = exp(-\alpha \left\| x - x_i^* \right\|^2). \quad (5)$$

When performing classification, the consequent of the rule with the highest degree of fulfillment is selected to be the output class of the classifier. In this stage, we obtain a set of rules for identifying each class of data. The obtained rules are then combined to form the rule base of the classifier.

3. Optimizing membership functions: We use the following classification error measure for a data point that belongs to some class c, proposed in [31]:

$$E = 1/2 (1 - \mu_{c,\max} + \mu_{\neg c,\max})^2 . \quad (6)$$

Where $\mu_{c,\max}$ is the highest degree of fulfilment among all rules that infer class c and $\mu_{\neg c,\max}$ is the highest degree of fulfilment among all rules that do not infer class c. Note that this error measure is zero only if a

rule that would correctly classify the data point has degree of fulfilment of 1 and all rules that would misclassify the data point have degree of fulfilment of 0 [31]. The membership functions parameters are updated according to the following gradient descent formulae:

$$x_{ij}^{*} = x_{ij}^{*} - \lambda\, \partial E \big/ \partial x_{ij}^{*}, \quad \partial E \big/ \partial x_{ij}^{*} = \pm \mu_i ((1 - \mu_{c,\max} + \mu_{\neg c,\max})(X_j - x_{ij}^{*}) \big/ \sigma_{ij}^{2}) \tag{7}$$

$$\sigma_{ij} = \sigma_{ij} - \lambda\, \partial E \big/ \partial \sigma_{ij}, \quad \partial E \big/ \partial \sigma_{ij} = \pm \mu_i ((1 - \mu_{c,\max} + \mu_{\neg c,\max})(X_j - x_{ij}^{*})^2 \big/ \sigma_{ij}^{3}). \tag{8}$$

In (7) and (8), λ is a positive learning rate. An important feature of this optimization process is that only two rules are updated: the rule that provided $\mu_{c,\max}$ and the rule that provided $\mu_{\neg c,\max}$; the '+' sign is used for the rule that provided $\mu_{c,\max}$ and the '-' sign is used for the rule that provided $\mu_{\neg c,\max}$. This gradient descent algorithm can be viewed as a type of competitive learning algorithm: a winner in the 'good rule' category is reinforced and a winner in the 'bad rule' category is punished. Because only two rules are updated each time, the algorithm is highly efficient [31].

3. EXPERIMENTAL RESULTS

In this section, the performance of new proposed method is examined. We have used 20 data sets with numerical attributes from the University of California, Irvine machine learning repository (UCI) [36], all of them are valid for classification tasks. Table 1 shows specification of these data sets.

Table 1. Statistics of data sets used for proposed method evaluation

Data set	Number of attributes	Number of examples	Number of classes
Wisconsin	9	699	2
Pima	8	768	2
Haberman	3	306	2
Bupa	6	345	2
Heart	13	270	2
Monk-2	6	432	2
Appendicitis	7	106	2
Saheart	9	462	2
Tic-tac-toe	9	958	2
Wine	13	178	3
Newthyroid	5	215	3
Iris	4	150	3
Balance	4	625	3
Post	8	90	3
Tae	5	151	3
Hayes-roth	4	160	3
Car	6	1728	4
Vehicle	18	846	4
Glass	9	214	7
Ecoli	7	336	8

We employ ten-fold cross validation (*10-CV*) testing method as a validation scheme to perform experiments and analyze results. The algorithm is run five times and the average of accuracies is calculated for each data set. In ten-fold cross validation method, each data set is randomly divided into ten disjoint sets of equal size (the size of each set is *m / 10*, where *m* is the total number of patterns in data set). The FRBCS is trained ten times, each time one of ten sets holds out as a test set for evaluating FRBCS and the nine remainder sets are used for training. The classification accuracy is computed in each

time and estimated classifier performance is the average of these 50 classification accuracies (estimated classifier accuracy is the average over 50 runs).

Experimental analysis for performance evaluation of the proposed method is a necessary task in an investigation. For the sake of comparison, Table 3 gives the accuracies obtained by the proposed method as well as those of previously developed methods over different data sets. In this table, we have used 20 data sets with numerical attributes from the University of California, Irvine machine learning repository (UCI) [36], all of them are valid for classification tasks. We measured performance of each classifier by means of its accuracy over test data by using 5 repetitions of *10-CV* cross validation. The best results in each row (for each data set) are highlighted by boldface.

In the Table 3, the first column shows names of data sets. The average classification accuracy for each data set by the proposed method and the algorithms are introduced in Table 2, are shown in the 2nd to 15th columns, respectively. Experiment results in Table 3 show that the proposed method achieves a higher average classification accuracy rate in the vast majority of experiment cases.

Table 2. The algorithms compared with proposed method in experiments

References	Method
Chen and Chen[37]	M1
Carvalho and Freitas[38]	M2
Gray and Fan [39]	M3
Sánchez and Otero [40]	M4
Mansoori et al. [41]	M5
Ishibuchi et al. [10]	M6
Ishibuchi and Yamamoto [42]	M7
Gao and Wang [43]	M8
Wang et al. [44]	M9
Paredes and Vidal [45]	M10
Nakashima et al. [46]	M11
Sánchez and Couso [47]	M12
Gonzalez and Perez [48]	M13

Table 3. Comparing the classification accuracy of proposed method with the other classification approaches (*10-CV* test method)

Method / Data set	Proposed method	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13
Wisconsin	**98.71**	95.30	93.85	94.98	95.84	93.69	95.27	91.11	94.99	96.86	65.52	96.85	95.56	95.42
Pima	**81.57**	65.10	73.31	70.71	75.01	73.05	74.09	73.06	69.94	72.14	72.01	73.71	72.93	73.32
Haberman	**80.00**	51.67	73.48	71.22	73.81	73.52	72.19	73.19	68.64	68.94	73.52	73.80	71.56	71.90
Bupa	**79.41**	42.10	65.88	58.74	64.63	57.34	64.95	57.87	63.16	63.17	58.63	59.37	57.37	56.06
Heart	**96.29**	80.37	79.62	71.85	75.92	75.56	73.70	51.85	75.18	77.40	76.29	77.30	78.14	78.51
Monk-2	95.34	47.33	97.46	95.19	96.77	80.64	95.88	42.89	74.90	70.70	86.53	69.31	94.70	**97.26**
Appendicitis	**93.75**	80.18	84.09	85.81	85.00	86.90	83.00	85.81	74.45	87.63	85.72	83.72	84.09	82.18
Saheart	**76.08**	65.36	68.82	65.36	70.96	67.96	67.07	72.48	64.05	70.10	65.54	69.87	70.95	65.12
Tic-tac-toe	82.10	65.34	77.97	69.93	**87.16**	69.93	72.45	50.14	73.17	81.84	34.66	49.16	76.47	65.34
Wine	**97.53**	93.20	94.90	78.76	94.96	90.42	89.80	93.82	96.63	96.63	94.37	94.01	89.24	92.12
Newthyroid	96.76	69.80	91.64	81.77	91.70	86.53	89.35	84.24	88.44	98.61	97.20	**97.25**	92.57	90.75
Iris	**96.66**	83.33	96.00	86.00	93.33	94.00	93.33	92.66	92.66	94.66	93.33	96.00	93.33	95.33
Balance	88.78	46.08	75.49	71.34	79.34	69.12	86.39	90.39	74.56	**89.26**	76.34	85.61	74.56	74.08
Post	**87.50**	71.38	69.16	67.91	63.89	67.91	65.69	42.91	49.58	60.83	46.52	40.56	68.06	71.38
Tae	**56.16**	32.46	41.12	41.16	49.79	49.12	55.83	55.12	41.79	46.46	41.83	54.46	55.75	52.41
Hayes-roth	73.12	40.62	74.37	50.62	74.37	44.99	61.87	58.75	46.25	47.50	37.50	51.87	65.62	**77.50**
Car	83.13	70.02	81.36	76.26	83.21	67.18	72.22	77.83	87.67	**89.69**	78.02	86.22	80.26	70.02
Vehicle	**75.00**	32.52	71.50	47.87	48.57	52.48	50.24	60.77	74.23	69.39	69.26	65.20	60.50	60.00
Glass	**90.47**	32.89	61.16	44.53	62.77	61.02	56.60	60.04	70.94	72.66	70.87	58.30	55.15	58.14
Ecoli	**87.87**	42.56	76.47	62.79	70.26	74.44	73.83	72.02	67.88	82.47	82.47	80.40	70.22	84.53

However, this observation-based evaluation does not reflect whether or not the differences among the methods are significant. We use statistical tests to make sure that the difference is significant, that is, very unlikely to be caused by chance - the so-called *p*-value of the test [49]. To evaluate performance of the proposed method, we use Friedman test [49-52], which is a non-parametric statistical analysis based on multiple comparison procedures. In order to perform a multiple comparison, it is necessary to check whether all results obtained by the algorithms present any inequality. Friedman test ranks the algorithms for each data set separately, the best performing algorithm obtaining the rank of 1, the second best rank 2, and so on. In case of ties, average ranks are assigned. Under the null-hypothesis, it is stated that all the algorithms are equivalent, so a rejection of this hypothesis implies the existence of differences among the performance of all the algorithms studied [51, 52]. Friedman test's way of working is described as follows.

Let r_i^j be the rank of the *j-th* of *k* algorithms on the *i-th* of *N* data sets. The Friedman test compares the average ranks of algorithms, $R_j = (1/N).\sum_i r_i^j$. Under the null-hypothesis, which states that all the algorithms are equivalent and so their ranks R_j should be equal, the Friedman statistic is distributed according to χ_F^2 with $k - 1$ degrees of freedom and is as [52]:

$$\chi_F^2 = \frac{12N}{k(k+1)}.[\sum_j R_j^2 - \frac{k(k+1)^2}{4}]. \tag{9}$$

Average ranks obtained by each method in the Friedman test are shown in Table 4. In this table, value of Friedman statistic (distributed according to chi-square with 13 degrees of freedom) is 92.602875 and *p*-value computed by this test is 0. These ranking values will be useful to calculate the *p*-values and to detect significant differences between the methods. Evidently, the rank assigned to the proposed method is less than other methods. Hence, new hybrid method is the best method. Statistical test are done using average accuracies in test data.

Then, Holm's test and Finner test are used as post-hoc methods. Holm's test [53] is a multiple comparison procedure that can work with a control algorithm (which is usually the best according to Friedman rankings computation) and is compared with the remaining methods. The tests statistics for comparing the *i-th* and *j-th* method using this procedure are:

$$z = (R_i - R_j)\Big/\sqrt{k(k+1)/6N}\,. \tag{10}$$

The *z* value is used to find the corresponding probability from the table of normal distribution, which is then compared with an appropriate level of confidence α. Holm's test adjusts the value for α in order to compensate for multiple comparisons.

Holm's test adjusts value of α in a step-down manner. Let $p_1, p_2, ..., p_{k-1}$ be the ordered *p*-values (smallest to largest), so that $p_1 \le p_1 \le ... \le p_{k-1}$, $H_1, H_2, ..., H_{k-1}$ be the corresponding hypotheses. The Holm procedure rejects H_1 to H_{i-1} if i is the smallest integer such that $p_i > \alpha / (k - i)$. Holm's step-down procedure starts with the most significant *p*-value. If p_1 is below $\alpha / (k - 1)$, the corresponding hypothesis is rejected and we are allowed to compare p_2 with $\alpha / (k - 2)$. If the second hypothesis is rejected, the test proceeds with the third, and so on. As soon as a certain null hypothesis cannot be rejected, all the remaining hypotheses are retained as well [51]. The Finner procedure [54] adjusts the value of α in a step-down manner, as Holm's method does. It rejects H_1 to H_{i-1} if i is the smallest integer so that $p_i > 1 - (1 - \alpha)^{(k-1)/i}$ [51].

The *p*-values obtained by applying Holm method and Finner method over the results of Friedman procedure are shown in Table 5. Holm's and Finner procedures reject those hypotheses that have a *p*-value

≤ 0.05. As Table 5 shows, Holm's and Finner procedures verify that the proposed method performs better than all other approaches, because all approaches have a *p*-value ≤ 0.05.

Table 4. Average rankings of algorithms by Friedman procedure

Algorithm	Ranking
Proposed method	**1.85**
M1	11.75
M2	5.575
M3	10.7
M4	5.45
M5	9.075
M6	7.725
M7	8.7
M8	9.075
M9	5.3
M10	8.325
M11	6.575
M12	7.625
M13	7.275

Table 5. Post Hoc comparison table for $\alpha = 0.05$ (Friedman)

i	Algorithm	*z*	*p*	Holm	Finner
13	M1	7.483697	0	0.003846	0.003938
12	M3	6.689971	0	0.004167	0.00786
11	M5	5.461587	0	0.004545	0.011767
10	M8	5.461587	0	0.005	0.015659
9	M7	5.178113	0	0.005556	0.019535
8	M10	4.89464	0.000001	0.00625	0.023396
7	M6	4.441083	0.000009	0.007143	0.027242
6	M12	4.36549	0.000013	0.008333	0.031072
5	M13	4.100915	0.000041	0.01	0.034888
4	M11	3.571764	0.000355	0.0125	0.038688
3	M2	2.815835	0.004865	0.016667	0.042474
2	M4	2.721344	0.006502	0.025	0.046244
1	M9	2.607955	0.009108	0.05	0.05

4. CONCLUSION

We proposed a new clustering-based method for modeling accurate fuzzy rule based classification systems. At first, a data mapping method was utilized to compute a new representation of data. Using this new representation caused better classification rates and helped increase the speed of the classification process (this is particularly relevant in high dimensional problems). This clustering-based method extracts fuzzy classification rules from new data by subtractive clustering and then optimizes the rule parameters by using an efficient gradient descent algorithm. Also, to compare the proposed method against other well known previously developed methods, several statistical tests were done. Simulation results show that the new approach significantly improves the classification performance.

REFERENCES

1. Zolghadri, M. J. & Taheri, M. (2008). A proposed method for learning rule weights in fuzzy rule-based classification systems. *Fuzzy Sets and Systems*, Vol. 159, No. 4, pp. 449-459.
2. Mansoori, E. G., Zolghadri, M. J. & Katebi, S. D. (2007). A weighting function for improving fuzzy classification systems performance. *Fuzzy Sets and Systems*, Vol. 158, No. 5, pp. 583-591.

3. Lin, Ch. J., Lee, Ch. Y. & Chen, Ch. H. (2007). A novel neuro-fuzzy inference system with multi-level membership function for classification applications. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol. 11, No. 4, pp. 365-372.
4. Muslimi, B., Capretz, M. A. M. & Samarabandu, J. (2006). An efficient technique for extracting fuzzy rules from neural networks. *International Journal of Intelligent Technology*, Vol. 1, No. 4, pp. 299-305.
5. Chakraborty, D. & Pal, N. R. (2004). A neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification.*IEEE Transactions on Neural Networks*, Vol. 15, No. 1, pp. 110-112.
6. Mitra, S. & Hayashi, Y. (2000). Neuro-fuzzy rule generation: survey in soft computing framework. *IEEE Transactions on Neural Networks*, Vol. 11, No. 3, pp. 748-768.
7. Ishibuchi, H. & Nojima, Y. (2007). Analysis of interpretability-accuracy trade of fuzzy systems by multiobjective fuzzy genetics-based machine learning. *International Journal of Approximate Reasoning*, Vol. 44, pp. 4-31.
8. Özyer, T., Alhajj, R. & Barker, K. (2007). Intrusion detection by integrating boosting genetic fuzzy classifier and data mining criteria for rule pre-screening. *Journal of Network and Computer Applications*, Vol. 30, pp. 99-113.
9. Hu, Y. Ch. (2005). Finding useful fuzzy concepts for pattern classification using genetic algorithms. Information Sciences, Vol. 175, pp. 1-19.
10. Ishibuchi, H., Yamamoto, T. & Nakashima, T. (2005). Hybridization of fuzzy GBML approaches for pattern classification problems. *IEEE Transactions on Systems, Man, and Cybernetics*, Part B, Vol. 35, No. 2, pp. 359-365.
11. Ho, Sh. Y., Chen, H. M., Ho, Sh. J. & Chen, T. K. (2004). Design of accurate classifiers with a compact fuzzy-rule base using an evolutionary scatter partition of feature space. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, Vol. 34, pp. 1031-1044.
12. Torun, Y. & Tohumoğlu, G. (2011). Designing simulated annealing and subtractive clustering based fuzzy classifier. *Applied Soft Computing*, Vol. 11, pp. 2193-2201.
13. Han, M., Sun, Y. & Fan, Y. An improved fuzzy neural network based on T–S model. *Journal of Expert Systems with Applications*, Vol. 34, No. 4, pp. 2905-2920.
14. Zhao, L., Yang, Y. & Zeng, Y. (2009). Eliciting compact T–S fuzzy models using subtractive clustering and coevolutionary particle swarm optimization. *Journal of Neurocomputing*, Vol. 72, No. 10, pp. 2569-2575.
15. Zhao, L., Yang, Y. & Zeng, Y. (2008). Extracting compact T-S fuzzy models using subtractive clustering and particle swarm optimization. *Proceedings of the 27th Chinese Control Conference*, pp. 362-366.
16. Eftekhari, M. & Katebi, S. D. (2008). Extracting compact fuzzy rules for nonlinear system modeling using subtractive clustering, GA and unscented filter. *Journal of Applied Mathematical Modeling*, Vol. 32, No. 12, pp. 2634-2651.
17. Chen, Ch., Li, Sh., Liu, Ta., Chen, K. & Wong, Ch. (2011). A Clustering-based algorithm to extracting fuzzy rules for system modeling. *International Journal of Advancements in Computing Technology(IJACT)*, Vol. 3, No. 11, pp. 394-401.
18. Radha, R. & Rajagopalan, S. P. (2010). Generating membership values and fuzzy association rules from numerical data. *International Journal on Computer Science and Engineering (IJCSE)*, Vol. 2, No. 8, pp. 2705-2715.
19. Zhang, W., Hu, H. & Liu, W. (2007) Rules extraction of interval type-2 fuzzy logic system based on fuzzy c-means clustering. *4th International Conference on Fuzzy Systems and Knowledge Discovery*, Vol. 2, pp. 256-260.
20. Zhang, W. & Liu, W. (2007). IFCM: Fuzzy clustering for rule extraction of interval type-2 fuzzy logic system. *Proceedings of the 46th IEEE Conference on Decision and Control*, pp. 5318-5322.

21. Mansoori, E. G. (2011). FRBC:A fuzzy rule-based clustering algorithm. *IEEE Transactions on Fuzzy Systems*, Vol. 19, No. 5, pp. 960-971.
22. Chen, Sh., Chen, Y., Zhang, W. & Yan, W. (2010). A method of extracting fuzzy control rules of structural vibration using FCM algorithm. *Proceedings of the 5th IEEE Conference on Industrial Electronics and Applications*, pp. 1550-1555.
23. Shirazi, H. & Sammut, C. A. (2008). Acquiring control knowledge from examples using ripple-down rules and machine learning. *Iranian Journal of Science & Technology, Transaction B, Engineering*, Vol. 32, No. B3, pp. 295-304.
24. Deypir, M. & Sadreddini, M. H. (2009). Distributed association rules mining using non-derivable frequent patterns. *Iranian Journal of Science & Technology, Transaction B, Engineering*, Vol. 33, No. B6, pp. 511-526.
25. Yager, R. R. & Filev, D. P. (1994). Generation of fuzzy rules by mountain clustering. *Journal of Intelligent and Fuzzy Systems*, pp. 209-219.
26. Demirli, K., Cheng, S. X. & Muthukumaran, P. (2003). Subtractive clustering based modeling of job sequencing with parametric search. *Fuzzy Sets and Systems*, Vol. 137, pp. 235–270.
27. Elmzabi, A., Bellafkih, M., Ramdani, M. & Zeitouni, K. (2004). A conditional fuzzy clustering with adaptive method. *10th Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU 2004*, Perugia – Italy, pp. 85-90.
28. Zhang, H. & Lu, J. (2010). *Creating ensembles of classifiers via fuzzy clustering and deflection*. Fuzzy Sets and Systems, pp. 1790-1802.
29. Chen, M. Y. (2002). Establishing interpretable fuzzy models from numeric data. *Proceedings of the 4th world congress on Intelligent Control and Automation*, June 10-14, Shanghai, P. R. China.
30. Hossen, J., Rahman, A., Samsudin, K., Rokhani, F., Sayeed, S. & Hasan, R. (2010). A novel modified adaptive fuzzy inference engine and its application to pattern classification, *International Journal of Computer and Information Engineering*, Vol. 4, No. 3, pp. 219-225.
31. Chiu, S. (1996). Method and software for extracting fuzzy classification rules by subtractive clustering. *IEEE Conference on Fuzzy Information Processing Society*, pp. 461-465.
32. Chiu, S. L. (1994). *Fuzzy model identification based on cluster estimation*. *Journal of Intelligent and Fuzzy Systems*, Vol. 2, No. 3, pp. 267-278.
33. Yager, R. R. & Filev, D. P. (1994). Approximate clustering via the mountain method. *IEEE Transactions on Fuzzy systems, Man, and Cybernetics*, Vol. 24, pp. 1279-1284.
34. Yager, R. R. & Filev, D. P. (1994). *Essentials of fuzzy modeling and control*. John Wiley & Sons, Inc, New York.
35. Chiu, S. L. (1997). *Extracting fuzzy rules from data for function approximation and pattern recognition, Chapter 9 in fuzzy information engineering: a guided tour of applications*. ed. D. Dubois, H. Prade, and R. Yager, John Wiley & Sons.
36. Frank, A. & Asuncion, A. (2010). UCI machine learning repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California', School of Information and Computer Science.
37. Chen, Z. & Chen, G. (2008). Building an associative classifier based on fuzzy association rules. *International Journal of Computational Intelligence Systems*, Vol. 1, No. 3, pp. 262-273.
38. Carvalho, D. R. & Freitas, A. A. (2004). A hybrid decision tree/genetic algorithms method for data mining. *Information Sciences*, Vol. 163, pp. 13-35.
39. Gray, J. B. & Fan, G. (2008). Classification tree analysis using TARGET. *Computational Statistics & Data Analysis*, Vol. 52, pp. 136-1372.
40. Sánchez, L. & Otero, J. (2007). Boosting fuzzy rules in classification problems under single-winner inference. *International Journal of Intelligent Systems*, Vol. 22, pp. 1021-1034.

41. Mansoori, E. G., Zolghadri, M. J. & Katebi, S. D. (2008). SGERD: A steady-state genetic algorithm for extracting fuzzy classification rules from data. *IEEE Transactions on Fuzzy Systems*, Vol. 16, No. 4, pp. 1061-1071.
42. Ishibuchi, H. & Yamamoto, T. (2005). Rule weight specification in fuzzy rule-based classification systems. *IEEE Transactions on Fuzzy Systems*, Vol. 13, No. 4, pp. 428-435.
43. Gao, Q. B. & Wang, Z. Z. (2007). Center-based nearest neighbor classifier. *Pattern Recognition*, Vol. 40, pp. 346-349.
44. Wang, J., Neskovic, P. Cooper, L. N. (2007). Improving nearest neighbor rule with a simple adaptive distance measure. *Pattern Recognition Letters*, Vol. 28, pp. 207-213.
45. Parades, R. & Vidal, E. (2006). Learning weighted metrics to minimize nearest-neighbor classification error. *IEEE Transactions on Pattern analysis and Machine Intelligence*, Vol. 28, No. 7, pp. 1100-1110.
46. Nakashima, T., Schaefer, G., Yokota, Y. & Ishibuchi, H. (2007). A weighted fuzzy classifier and its application to image processing tasks. *Fuzzy Sets and Systems*, Vol. 158, pp. 284-294.
47. Sánchez, L., Couso, I. & Corrales, J. A. (2001). Combining GP operators with SA search to evolve fuzzy rule based classifier. *Information Sciences*, Vol. 136, pp. 175-191.
48. González, A. & Pêrez, R. (2001). Selection of relevant features in a fuzzy genetic learning algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol. 31, pp. 417-425.
49. Yílíz, O. T., Aslan, Ö. & Alpaydin, E. (2011). Multivariate statistical tests for comparing classification algorithms. in LION'05 2011, *Proceedings of the 5th international conference on Learning and Intelligent Optimization*, Springer-Verlag Berlin, Heidelberg, pp. 1-15.
50. Sheskin, D. (2007). *Handbook of parametric and nonparametric statistical procedures*. 4nd ed., Chapman and Hall/CRC.
51. García, S., Fernández, A., Luengo, J. & Herrera, F. (2010). Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Journal of Information Sciences: an International Journal*, Vol. 180, No. 10, pp. 2044-2064.
52. Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. Journal of machine learning research, Vol. 7, pp. 1-30.
53. Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, Vol. 6, No. 2, pp. 65-70.
54. Finner, H. (1993). On a monotonicity problem in step-down multiple test procedures. *Journal of the American Statistical Association*, Vol. 88, No. 423, pp. 920-923.