Artif Life Robotics (2009) 14:418–421
DOI 10.1007/s10015-009-0700-3

ORIGINAL ARTICLE

Yusuke Nojima · Hisao Ishibuchi

Incorporation of user preference into multi-objective genetic fuzzy rule selection for pattern classification problems

Received and accepted: May 19, 2009

Abstract In the design of fuzzy-rule-based systems, we have two conflicting objectives: accuracy maximization and interpretability maximization. As a measure of interpretability, a number of criteria have been proposed in the literature. Most of those criteria have been incorporated into fitness functions in order to automatically find accurate and interpretable fuzzy systems by genetic algorithms. However, interpretability is very subjective and is rarely defined for any users beforehand. In this article, we propose the incorporation of user preference into multi-objective genetic fuzzy rule selection for pattern classification problems. User preference is represented by a preference function which is changeable according to the user's direct manipulation during evolution. The preference function is used as one of the objective functions in multi-objective genetic fuzzy rule selection. The effectiveness of the proposed method is examined through some case studies for the design of fuzzy-rule-based classifiers.

Key words Multi-objective genetic fuzzy systems · Fuzzy-rule-based systems · User preference · Interactive genetic algorithms · Pattern classification problems

1 Introduction

Fuzzy-rule-based systems have been widely used for pattern classification, function approximation, modeling, forecasting, and control. One advantage of fuzzy-rule-based systems over other nonlinear systems such as neural networks is their linguistic interpretability. That is, each fuzzy rule is linguistically interpretable when fuzzy-rule-based systems are designed using the linguistic knowledge of human experts. However, linguistic knowledge is not always available, especially for high-dimensional data. Thus, since the early 1990s, various approaches have been proposed in the literature for extracting fuzzy rules from numerical data. Evolutionary algorithms can be used not only for parameter tuning, but also for discrete optimization such as input selection, rule generation, and rule selection.[1] Most fitness functions were based on the maximization of the accuracy of fuzzy-rule-based systems only. Since the late 1990s, the importance of maintaining interpretability in the design of fuzzy-rule-based systems has been pointed out in many studies. Interpretability maximization as well as accuracy maximization was taken into account in order to design accurate and interpretable fuzzy-rule-based systems.[2] The number of fuzzy rules in a system has generally been used as one of the complexity measures. In the literature, other measures are the total number of condition parts, transparency, compactness, and so on. However, interpretability is very subjective and is rarely specified beforehand without actual users.

For the design of simple and accurate fuzzy-rule-based classifiers, we have already proposed multi-objective genetic fuzzy rule selection.[3] We have used two objective functions: to maximize the number of correctly classified training patterns, and to minimize the number of fuzzy rules in a fuzzy-rule-based classifier. Considering user preference on the interpretability of fuzzy-rule-based classifiers, we propose the incorporation of user preference, represented by a preference function, into multi-objective genetic fuzzy rule selection for pattern classification problems. During evolution, the preference function can be changed interactively, and is used as one of the objective functions. That is, our method can find nondominated solutions (fuzzy-rule-based classifiers) in terms of three objectives: accuracy maximization, complexity minimization, and preference maximization. Through some case studies, we examine the effectiveness of the proposed idea.

Y. Nojima (✉) · H. Ishibuchi
Department of Computer Science and Intelligent Systems, Graduate School of Engineering, Osaka Prefecture University, 1-1 Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531, Japan
e-mail: nojima@cs.osakafu-u.ac.jp

This work was presented in part at the 14th International Symposium on Artificial Life and Robotics, Oita, Japan, February 5–7, 2009

2 Genetic fuzzy rule selection with user preference

In this section, we explain fuzzy-rule-based classifiers and multi-objective genetic fuzzy rule selection. We also explain user preference, and the preference function proposed in this paper.

2.1 Fuzzy-rule-based classifiers

Let us assume that we have m training (i.e., labeled) patterns $\mathbf{x}_p = (x_{p1}, \ldots, x_{pn})$, $p = 1, 2, \ldots, m$ from M classes in an n-dimensional pattern space, where x_{pi} is the attribute value of the p-th pattern for the i-th attribute ($i = 1, 2, \ldots, n$). For simplicity, we assume that all the attribute values have already been normalized into real numbers in the unit interval [0, 1]. Thus the pattern space of our classification problem is an n-dimensional unit-hypercube $[0, 1]^n$.

For our n-dimensional pattern classification problem, we use fuzzy rules of the following type:

$$\text{Rule } R_q\text{: If } x_1 \text{ is } A_{q1} \text{ and} \ldots \text{and } x_n \text{ is } A_{qn} \text{ then Class } C_q \text{ with } CF_q \tag{1}$$

where R_q is the label of the q-th fuzzy rule, $\mathbf{x} = (x_1, \ldots, x_n)$ is an n-dimensional pattern vector, A_{qi} is an antecedent fuzzy set ($i = 1, 2, \ldots, n$), C_q is a class label, and CF_q is a rule weight. We denote the antecedent fuzzy sets of R_q as a fuzzy vector $\mathbf{A}_q = (A_{q1}, A_{q2}, \ldots, A_{qn})$.

We use 14 fuzzy sets in four fuzzy partitions with different granularities in Fig. 1. In addition to those 14 fuzzy sets, we also use the domain interval [0, 1] itself as an antecedent fuzzy set in order to represent a *don't care* condition.

The consequent class C_q and the rule weight CF_q of each fuzzy rule R_q are specified from training patterns compatible with its antecedent part $\mathbf{A}_q = (A_{q1}, A_{q2}, \ldots, A_{qn})$ in the following heuristic manner. First we calculate the confidence of each class for the antecedent part $\mathbf{A}_q$ as

$$c(\mathbf{A}_q \Rightarrow \text{Class } h) = \frac{\sum_{\mathbf{x}_p \in \text{Class } h} \mu_{\mathbf{A}_q}(\mathbf{x}_p)}{\sum_{p=1}^{m} \mu_{\mathbf{A}_q}(\mathbf{x}_p)}, \quad h = 1, 2, \ldots, M \tag{2}$$

Then the consequent class C_q is specified by identifying the class with the maximum confidence:

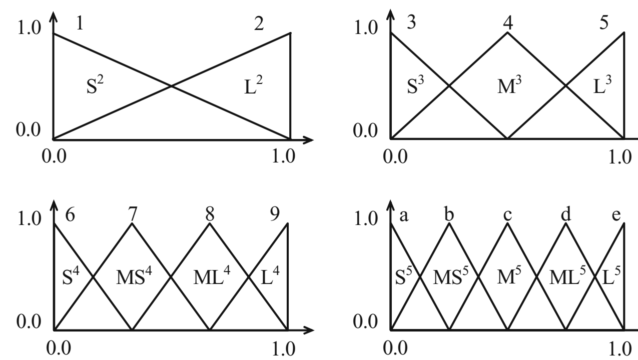


Fig. 1. Membership functions used

$$c(\mathbf{A}_q \Rightarrow \text{Class } C_q) = \max_{h=1,2,\ldots,M} \{c(\mathbf{A}_q \Rightarrow \text{Class } h)\} \tag{3}$$

In this manner, we generate the fuzzy rule R_q with the antecedent part $\mathbf{A}_q$ and the consequent class C_q.

The rule weight CF_q of each fuzzy rule R_q is specified by the confidence values

$$CF_q = c(\mathbf{A}_q \Rightarrow \text{Class } C_q) - \sum_{h=1, h \neq C_q}^{M} c(\mathbf{A}_q \Rightarrow \text{Class } h) \tag{4}$$

We do not use the fuzzy rule R_q as a candidate rule if the rule weight CF_q is not positive (i.e., if its confidence is not larger than 0.5).

As with confidence, support is also often used for evaluating the interestingness of individual rules. Support can be calculated as

$$s(R_q) = s(\mathbf{A}_q \Rightarrow \text{Class } C_q) = \frac{\sum_{\mathbf{x}_p \in \text{Class } C_q} \mu_{\mathbf{A}_q}(\mathbf{x}_p)}{m} \tag{5}$$

Let S be a set of fuzzy rules of the form in Eq. 1. When an input pattern $\mathbf{x}_p$ is to be classified by S, we first calculate the compatibility grade of $\mathbf{x}_p$ with the antecedent part $\mathbf{A}_q$ of each fuzzy rule R_q in S using the product operation. Then a single winner rule is identified using the compatibility grade and the rule weight of each fuzzy rule. The input pattern $\mathbf{x}_p$ is classified as the consequent class of the winner rule.

2.2 Multi-objective genetic fuzzy rule selection

Multi-objective genetic fuzzy rule selection is a two-step method. In the first step, a prespecified number of promising fuzzy rules are generated from training patterns as candidate rules. In the second step, an EMO algorithm is used to search for nondominated fuzzy-rule-based classifiers (i.e., nondominated subsets of the candidate rules generated in the first step).

Since we use the 14 antecedent fuzzy sets in Fig. 1 and a *don't care* for each attribute of our n-dimensional classification problem, the total number of possible fuzzy rules is 15^n. Among these possible rules, we examine only short fuzzy rules with a small number of antecedent conditions (i.e., short fuzzy rules with many *don't care* conditions) to generate candidate rules. Here, we examine fuzzy rules with three antecedent conditions or fewer. For prescreening candidate rules, we use the product of the support $s(R_q)$ and the confidence $c(R_q)$. That is, we choose a prespecified number of the best candidate rules for each class with respect to $s(R_q)$ $c(R_q)$.

Let us assume that we have N candidate rules (i.e., N/M candidate rules for each of M classes). Any subset S of the N candidate rules can be represented by a binary string of length N: $S = s_1 s_2 \ldots s_N$, where $s_j = 1$ and $s_j = 0$ mean the inclusion and exclusion of the j-th candidate rule R_j in the subset S, respectively ($j = 1, 2, \ldots, N$). Such a binary string S is used as an individual (i.e., a fuzzy classifier) in an EMO algorithm for multi-objective genetic fuzzy rule selection.

Each fuzzy rule-based classifier S is evaluated by the following three objectives:

– $f_1(S)$: the number of correctly classified training patterns;
– $f_2(S)$: the number of selected fuzzy rules;
– $f_3(S)$: user preference.

That is, our multi-objective genetic fuzzy rule selection is written as

$$\text{Maximize } f_1(S) \text{ and } f_3(S)\text{, and minimize } f_2(S) \tag{6}$$

We use NSGA-II of Deb et al.[4] to search for nondominated fuzzy-rule-based classifiers with respect to these three objectives. Here, uniform crossover and bit-flip mutation were used in NSGA-II. In order to decrease the number of fuzzy rules in S efficiently, a larger mutation probability is assigned to the mutation from 1 to 0 than that from 0 to 1. In addition, the unnecessary fuzzy rules which were not selected as a winner rule were removed from S after calculating the first objective.

2.3 User preference on interpretability

Interpretability is very subjective, and is rarely specified without actual users. One approach may be to use various interpretability measures as objective functions. However, current evolutionary multi-objective optimization algorithms are not appropriate for problems with more than four objectives.[5] For these reasons, we combine multiple interpretability criteria into a single preference function. Then users change the priority of criteria in the preference function during the evolution of multi-objective genetic fuzzy rule selection.

We specify an interval for internal evaluations. During this interval, the preference function is not changed. After the interval, the user checks some of the nondominated classifiers and changes the priority of criteria in the preference function. Then another internal evaluation process starts. By repeating this interactive process, the user can modify the preference function and find the classifier with a high user preference value.

In this article, we use three criteria for representing user preference: average confidence, average support, and the number of attributes used. Confidence and support have often been used to examine the interestingness of individual rules.[6] Of course, we can use other criteria in the preference function.

3 User interface

We developed a user interface for presenting a fuzzy-rule-based classifier to the user and incorporating their preference (Fig. 2). The antecedent part of each fuzzy rule is shown together with its consequent class, confidence, and support. Solid triangles and open rectangles mean membership functions and *don't care* conditions, respectively. The accuracy of the classifier is shown at the bottom right-hand side of the classifier. The gray zone at the bottom of the interface is a user manipulation area.

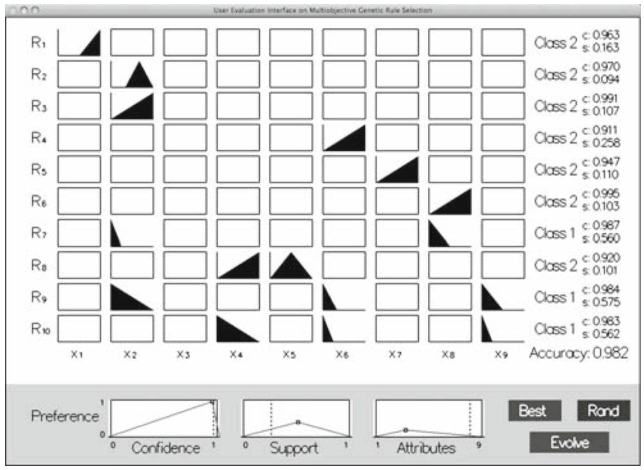


Fig. 2. A user interface for the proposed method

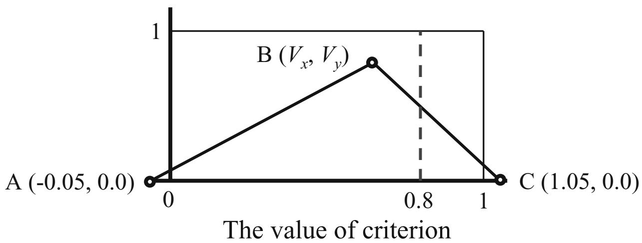


Fig. 3. Satisfaction level functions for interpretability criteria

Individual preference and its priority for each criterion are represented by a satisfaction level function with two segments, A–B and B–C, in Fig. 3. The three points A, B, and C are (–0.05, 0.0), (V_x, V_y), and (1.05, 0.0), respectively. Users can change the preference and the priority of each criterion by moving point B (V_x, V_y) in $0 \le V_x \le 1$ and $0 \le V_y \le 1$. If the value of some criterion is 0.8 in Fig. 3, the output value on the criterion is 0.5.

A preference function is composed of the three satisfaction-level functions, as in Fig. 3. In this article, the simple sum of the output values is used as the satisfaction degree of user preference on the interpretability of fuzzy classifiers.

Each vertical dashed line of satisfaction-level functions represents the actual values of three criteria for the displayed classifier. Thus, users can refer to this information and change the position of the vertices of the triangles. That is, users can modify the preference function (i.e., satisfaction-level functions) according to their impression from some displayed classifiers.

There are three buttons in the bottom right-hand corner. The button "Best" is to show the best classifier in terms of user preference. The button "Rand" is to show three classifiers randomly selected among nondominated ones. The button "Evolve" is to start another internal evaluation process with a prespecified number of generations.

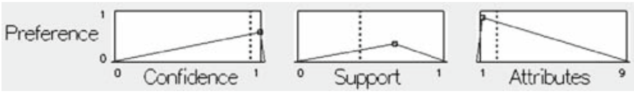


Fig. 4. Satisfaction level functions in Case 1

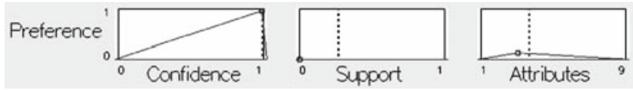


Fig. 6. Satisfaction level functions in Case 2

Fig. 5. Nondominated classifier with the highest user preference value in Case 1

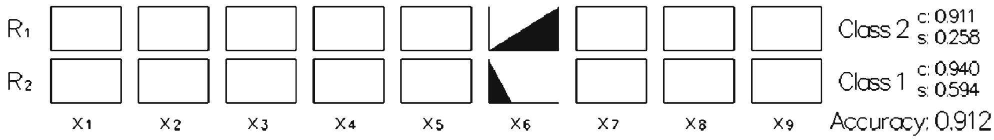
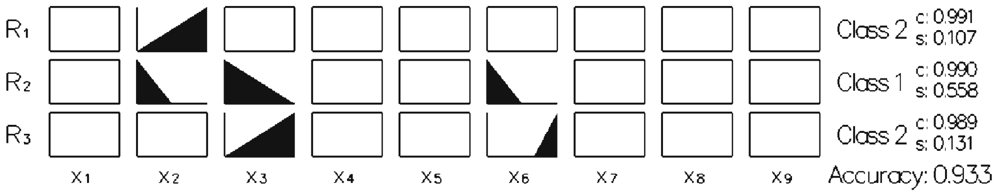


Fig. 7. Nondominated classifier with the highest user preference value in Case 2



4 Case studies

In this section, we show two case studies in which two users have different preferences on interpretability. We used the Wisconsin breast cancer data (683 patterns, 9 attributes, 2 classes), which is available from the UCI machine learning repository. The parameter settings were as follows:

– Number of extracted rules per class: 300;
– Population size: 200;
– Number of generations: 500;
– Interval for internal evaluations: 50 generations.

Case 1. We assumed that the user prefers a very simple rule set. At the 250th generation, the user specified the satisfaction level functions in Fig. 4. The classifier obtained with the highest user preference value is shown in Fig. 5. Each rule has fairly high confidence and support. The total number of attributes used is only one. This is a very simple rule set which means "if the value of *Bare Nuclei* is high, the sample is malignant" and "if the value of *Bare Nuclei* is small, the sample is benign."

Case 2. We assumed that the user prefers very accurate rules. As in Case 1, at the 250th generation, the user specified the satisfaction level functions in Fig. 6. The classifier obtained with the highest user preference value is shown in Fig. 7. We can see that each rule has a very high confidence value comparing with the rules in Case 1.

5 Conclusion

We have proposed the incorporation of user preference into multi-objective genetic fuzzy rule selection. We used a preference function to represent user preference as an additional objective in the multi-objective problem. Through some case studies, we demonstrated that our method can obtain nondominated fuzzy-rule-based classifiers in terms of accuracy and interpretability considering user preference. In future work, we will further examine the effect of changing the preference function on the search performance of our method.

Acknowledgment This work was partially supported by Grand-in-Aid for Young Scientists (B): KAKENHI (18700228).

References

1. Cordon O, Herrera F, Hoffmann F, et al (2001) Genetic fuzzy systems. World Scientific, Singapore
2. Ishibuchi H, Nozaki K, Yamamoto N, et al (1995) Selecting fuzzy if–then rules for classification problems using genetic algorithms. IEEE Trans Fuzzy Syst 3:260–270
3. Ishibuchi H, Murata T, Turksen IB (1997) Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems. Fuzzy Sets Syst 89:135–150
4. Deb K, Pratap A, Agarwal S, et al (2002) A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans Evolut Comput 6:182–197
5. Ishibuchi H, Tsukamoto N, Nojima Y (2008) Evolutionary many-objective optimization: a short review. Proceedings of the 2008 IEEE Congress on Evolutionary Computation, IEEE, Hong Kong, pp 2424–2431
6. Bayardo RJ Jr, Agrawal R (1999) Mining the most interesting rules. Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, San Diego, USA, pp 145–153