

# Learning Fuzzy Classification Rules from Data

Hans Roubos<sup>1</sup>, Magne Setnes<sup>2</sup>, and Janos Abonyi<sup>3</sup>

<sup>1</sup> Delft University of Technology, ITS, Control Laboratory,  
P.O. Box 5031, 2600 GA Delft, The Netherlands, hans@ieee.org

<sup>2</sup> Heineken Technical Services, R&D, Burgemeester Smeetsweg 1,  
3282 PH Zoeterwoude, The Netherlands, magne@ieee.org

<sup>3</sup> University of Veszprem, Department of Process Engineering,  
P.O. Box 158, H-8201 Veszprem, Hungary, abonyij@fmt.vein.hu

**Abstract.** Automatic design of fuzzy rule-based classification systems based on labeled data is considered. It is recognized that both classification performance and interpretability are of major importance and effort is made to keep the resulting rule bases small and comprehensible. An iterative approach for developing fuzzy classifiers is proposed. The initial model is derived from the data and subsequently, feature selection and rule base simplification are applied to reduce the model, and a GA is used for model tuning. An application to the Wine data classification problem is shown.

## 1 Introduction

Rule-based expert systems are often applied to classification problems in fault detection, biology, medicine etc. Fuzzy logic improves classification and decision support systems by allowing the use of overlapping class definitions and improves the interpretability of the results by providing more insight into the classifier structure and decision making process [13]. The automatic determination of fuzzy classification rules from data has been approached by several different techniques: neuro-fuzzy methods [6], genetic-algorithm based rule selection [5] and fuzzy clustering in combination with GA-optimization [12]. Traditionally, algorithms to obtain classifiers have focused either on accuracy or interpretability. Recently some approaches to combining these properties have been reported; fuzzy clustering is proposed to derive transparent models in [9], linguistic constraints are applied to fuzzy modeling in [13] and rule extraction from neural networks is described in [8].

In this paper we describe an approach that addresses both issues. Compact, accurate and linguistically interpretable fuzzy rule-based classifiers are obtained from labeled observation data in an iterative fashion. An initial model is derived from the observation data and subsequently, feature selection and rule base simplification methods [10] are applied to reduce the model. After the model reduction, a real-coded GA is applied to improve the classification accuracy [7,11]. To maintain the interpretability of the rule base, the GA search-space is restricted to the neighborhood of the initial rule base.

## 2 Fuzzy Models for Classification

### 2.1 The Model Structure

We apply fuzzy classification rules that each describe one of the  $N_c$  classes in the data set. The rule antecedent is a fuzzy description in the  $n$ -dimensional feature space and the rule consequent is a crisp (non-fuzzy) class label from the set  $\{1, 2, \dots, N_c\}$ :

$$R_i: \text{ If } x_1 \text{ is } A_{i1} \text{ and } \dots x_n \text{ is } A_{in} \text{ then } g_i = p_i, \quad i = 1, \dots, M. \quad (1)$$

Here  $n$  denotes the number of features,  $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$  is the input vector,  $g_i$  is the output of the  $i$ th rule and  $A_{i1}, \dots, A_{in}$  are the antecedent fuzzy sets. The **and** connective is modeled by the product operator, allowing for interaction between the propositions in the antecedent. The degree of activation of the  $i$ th rule is calculated as:

$$\beta_i(\mathbf{x}) = \prod_{j=1}^n A_{ij}(x_j), \quad i = 1, 2, \dots, M. \quad (2)$$

The output of the classifier is determined by the rule that has the highest degree of activation:

$$y = g_{i^*}, \quad i^* = \arg \max_{1 \leq i \leq M} \beta_i. \quad (3)$$

In the following we assume that the number of rules corresponds to the number of classes, i.e.,  $M = N_c$ . The certainty degree of the decision is given by the normalized degree of firing of the rule:

$$CF = \beta_{i^*} / \sum_i \beta_i. \quad (4)$$

### 2.2 Data Driven Initialization

From the  $K$  available input-output data pairs  $\{\mathbf{x}_k, y_k\}$  we construct the  $n$ -dimensional pattern matrix  $\mathbf{X}^T = [\mathbf{x}_1, \dots, \mathbf{x}_K]$  and the corresponding label vector  $\mathbf{y}^T = [y_1, \dots, y_K]$ . The fuzzy antecedents  $A_{ij}(x_j)$  in the initial rule base are now determined in by a three-step algorithm. In the first step,  $M$  multivariable membership functions are defined in the product space of the features. Each describes a region where the system can be approximated by a single fuzzy rule. This partitioning is often realized by iterative methods such as clustering [7]. Here, given the labeled data, a one-step approach is proposed. This assumes that each class is described by a single, compact construct in the feature space. If this is not the case, other methods such as, e.g. relational classification [9], can be applied. Similar to the Gustafson and Kessel's clustering algorithm [4], the approach proposed here assumes that

the shape of the fuzzy sets can be approximated by ellipsoids. Hence, each class prototype is represented by a center  $\mathbf{v}$  and its covariance matrix  $\mathbf{Q}$ :

$$\mathbf{v}_i = \frac{1}{K_i} \sum_{k|y_k=i} \mathbf{x}_k, \quad (5)$$

$$\mathbf{Q}_i = \frac{1}{K_i} \sum_{k|y_k=i} (\mathbf{x}_k - \mathbf{v}_i)(\mathbf{x}_k - \mathbf{v}_i)^T. \quad (6)$$

where  $i$  denotes the index of the classes,  $i = 1, \dots, N_c$ , and  $K_i$  represents the number of samples that belong to the  $i$ th class. In the second step, the algorithm computes the fuzzy partition matrix  $\mathbf{U}$  whose  $ik$ th element  $u_{ik} \in [0, 1]$  is the membership degree of the data object  $\mathbf{x}_k$  in class  $i$ . This membership is based on the distance between the observation and the class center:

$$D_{ik}^2 = (\mathbf{x}_k - \mathbf{v}_i)\mathbf{Q}_i^{-1}(\mathbf{x}_k - \mathbf{v}_i)^T. \quad (7)$$

Using this distance, the membership becomes:

$$u_{ik} = 1 / \sum_{j=1}^K \left( \frac{D_{ik}}{D_{jk}} \right)^{2/(m-1)}, \quad (8)$$

where  $m$  denotes a weighting exponent that determines the fuzziness of the obtained partition ( $m = 1.8$  is applied in the example).

The rows of  $\mathbf{U}$  now contain pointwise representations of the multidimensional fuzzy sets describing the classes in the feature space. In the third step, the univariate fuzzy sets  $A_{ij}$  in the classification rules (1) are obtained by projecting the rows of  $\mathbf{U}$  onto the input variables  $x_j$  and subsequently approximate the projections by parametric functions [1]. In the example we apply triangular fuzzy sets.

### 2.3 Ensuring Transparency and Accuracy

Fixed membership functions are often used to partition the feature space [5]. Membership functions derived from the data, however, explain the data-patterns in a better way. Typically less sets and fewer rules result than in a fixed partition approach. Hence, the initial rule base constructed by the proposed method fulfills many criteria for transparency and good semantic properties [11,13]: moderate number of rules, distinguishability, normality and coverage. The transparency and compactness of the rule base can be further improved by model reduction methods. Two methods are presented here. The first method is an open-loop feature selection algorithm that is based on Fisher's interclass separability criterion [2] calculated from the covariances of the clusters. The other method is the similarity-driven simplification proposed by Setnes et al. [10].

## 2.4 Feature Selection Based on Interclass Separability

Using too many features results in difficulties in the prediction and interpretability capabilities of the model due to redundancy, non-informative features and noise. Hence, feature selection is usually necessary. We apply the *Fischer interclass separability method* which is based on statistical properties of the labeled data. This criterion is based on the *between-class* (12) and *within-class* (13) scatter or covariance matrices that sum up to the *total scatter matrix* (11) which is the covariance of the whole training data:

$$\mathbf{Q}_t = \frac{1}{K} \sum_{i=1}^K (\mathbf{x}_k - \mathbf{v})(\mathbf{x}_k - \mathbf{v})^T, \quad (9)$$

$$\mathbf{v} = \frac{1}{K} \sum_{i=1}^K \mathbf{x}_k = \frac{1}{K} \sum_{i=1}^{N_c} K_i \mathbf{v}_i \quad (10)$$

The total scatter matrix can be decomposed as:

$$\mathbf{Q}_t = \mathbf{Q}_b + \mathbf{Q}_w, \quad (11)$$

$$\mathbf{Q}_b = \sum_{i=1}^{N_c} K_i (\mathbf{v}_i - \mathbf{v})(\mathbf{v}_i - \mathbf{v})^T, \quad (12)$$

$$\mathbf{Q}_w = \sum_{i=1}^{N_c} \mathbf{Q}_i. \quad (13)$$

The feature interclass separability selection criterion is a trade-off between  $\mathbf{Q}_b$  and  $\mathbf{Q}_w$ . A feature ranking is made iteratively by leaving out the worst feature in each step and is exploited for the open-loop feature selection:

$$J_j = \det(\mathbf{Q}_b) / \det(\mathbf{Q}_w), \quad (14)$$

where  $\det$  is the determinant and  $J_j$  is the criterion value including  $j$  features.

## 2.5 Similarity-driven rule base simplification

The similarity-driven rule base simplification method [10] uses a similarity measure to quantify the redundancy among the fuzzy sets in the rule base. A similarity measure based on the set-theoretic operations of intersection and union is applied:

$$S(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (15)$$

where  $|\cdot|$  denotes the cardinality of a set, and the  $\cap$  and  $\cup$  operators represent the intersection and union, respectively. If  $S(A, B) = 1$ , then the two membership functions  $A$  and  $B$  are equal.  $S(A, B)$  becomes 0 when the membership functions are non-overlapping.

Similar fuzzy sets are merged when their similarity exceeds a user defined threshold  $\theta \in [0, 1]$  ( $\theta=0.5$  is applied). Merging reduces the number of different fuzzy sets (linguistic terms) used in the model and thereby increases the transparency. If all the fuzzy sets for a feature are similar to the universal set, or if merging led to only one membership function for a feature, then this feature is eliminated from the model. The method is illustrated in Fig. 1

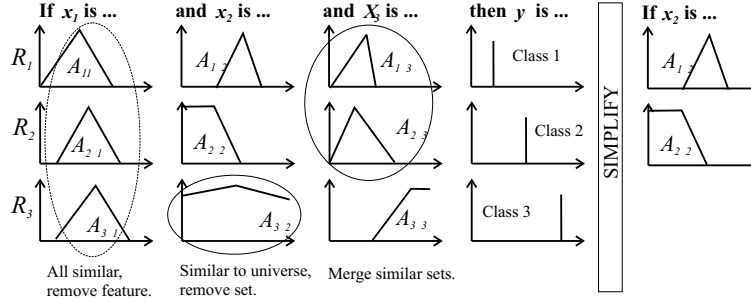


Fig. 1. Similarity-driven simplification.

### 2.6 Genetic Multi-Objective Optimization

To improve the classification capability of the rule base, we apply a genetic algorithm (GA) optimization method [11]. Also other model properties can be optimized by applying multi-objective functions, like, e.g., search for redundancy [7]. When an initial fuzzy model has been obtained from data, it is simplified and optimized in an iterative fashion. Combinations of the GA with the model reduction tools described above can lead to various modeling schemes. Three different approaches are shown in Fig. 2.

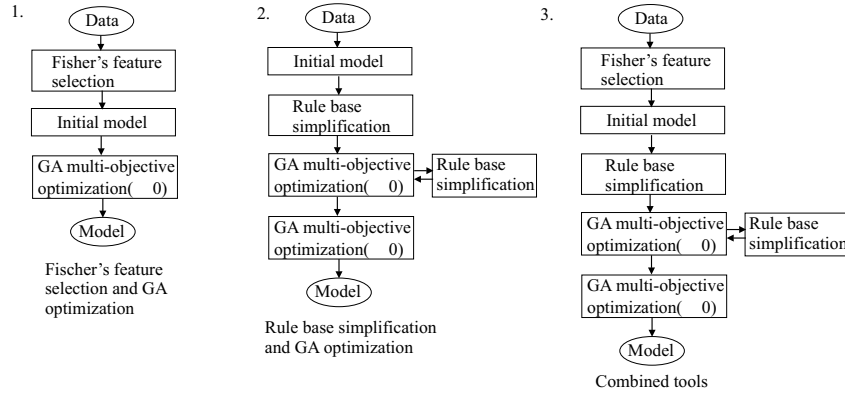


Fig. 2. Modeling schemes resulting from a combination of tools.

The model accuracy is measured in terms of the number of misclassifications. To further reduce the model complexity, the misclassification rate is combined with a similarity measure in the GA objective function. Similarity

is rewarded during the iterative process, that is, the GA tries to emphasize the redundancy in the model. This redundancy is then used to remove unnecessary fuzzy sets in the next iteration. In the final step, fine tuning is combined with a penalized similarity among fuzzy sets to obtain a distinguishable term set for linguistic interpretation.

The GAs is subject to minimize the following multi-objective function:

$$J = (1 + \lambda S^*) \cdot MSE, \quad (16)$$

where  $S^* \in [0, 1]$  is the average of the maximum pairwise similarity that is present in each input, i.e.,  $S^*$  is an aggregated similarity measure for the total model. The weighting function  $\lambda \in [-1, 1]$  determines whether similarity is rewarded ( $\lambda < 0$ ) or penalized ( $\lambda > 0$ ).

### 3 Example: Wine Data

The Wine data contains the chemical analysis of 178 wines produced in the same region in Italy but derived from three different cultivars. The problem is to distinguish the three different types based on 13 continuous attributes derived from chemical analysis. (Fig. 3). Corcoran and Sen [3] applied all

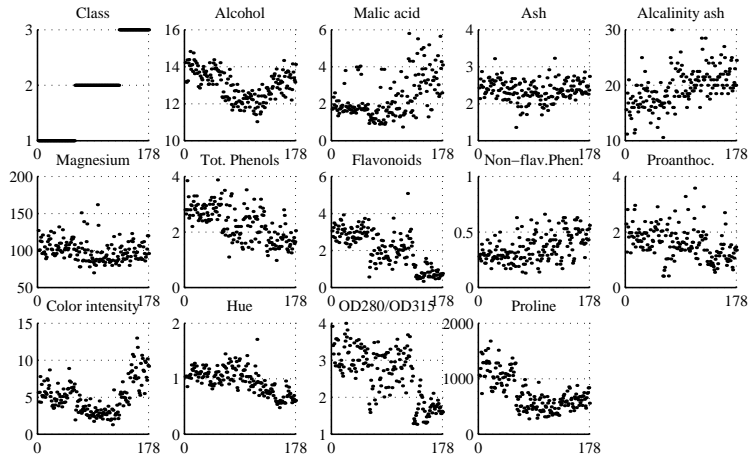
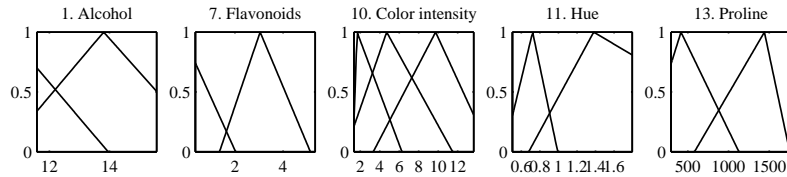


Fig. 3. Wine data: 3 classes and 13 attributes.

the data for learning 60 non-fuzzy if-then rules in a real-coded genetic based machine learning approach and Ishibuchi et al. [5] applied all the data for designing a fuzzy classifier with 60 fuzzy rules by means of an integer-coded genetic algorithm and grid partitioning (Table 2).

An initial classifier with three rules was constructed with the proposed covariance-based model initialization by using all samples resulting in 90.5% correct, 1.7% undecided and 7.9% misclassifications with the following average certainty factors (CF) [82.0, 99.6, 80.5] for the three wine classes. Improved classifiers are developed based on the three schemes given in Fig. 2:



**Fig. 4.** The fuzzy sets of the optimized three rule classifier for the Wine data.

**Scheme 1:** The Fisher interclass separability criterion gives the following feature ranking  $\{13, 12, 1, 4, 7, 6, 10, 9, 3, 2, 11, 5, 8\}$ . Classifiers were made by adding features *one by one* and 400 iterations with the GA-optimization. The two best classifiers were obtained by using the first 5 or 7 features (15 or 21 fuzzy sets). This gave 98.9% and 99.4% correct classification with  $CF$  for the three classes  $[0.95, 0.94, 0.84]$  and  $[0.94, 0.99, 0.97]$ , respectively.

**Scheme 2:** The similarity-driven simplification removed the following eight inputs in 3 steps: (i)  $\{3, 5\}$ , (ii)  $\{2, 4, 8, 9\}$ , (iii)  $\{6, 12\}$ . After each reduction, 200 GA-iterations were done and 400 after the last reduction. The final three-rule classifier (Table 1) contains only 11 fuzzy sets (Fig. 4). The classification result was 99.4% correct and  $CF$  for the three wine classes was  $[0.96, 0.94, 0.94]$ .

**Scheme 3:** Five features were selected based on the feature ranking initially resulting in 5% misclassification. Successively, 3 fuzzy sets and 1 feature were removed by iterative similarity-driven simplification and GA optimization (200 iterations). After the final GA tuning (400 iterations) the classification rate was 98.3% with  $CF$ s  $[0.93, 0.91, 0.91]$ . The final model contains features  $\{1, 7, 12, 13\}$ . The fuzzy sets obtained for  $\{1, 7, 13\}$  are similar to those obtained in Scheme 2 (Fig. 4).

In this example, feature reduction is obtained by all three schemes. Differences in the reduction methods are: (i) Similarity analysis results in a closed-loop feature selection because it depends on the actual model while the applied open-loop feature selection can be used beforehand as it is independent from the model. (ii) In similarity analysis, a feature can be removed from individual rules. In the interclass separability method the feature is omitted in all the rules.

The obtained result is comparable to those in [3] and [5], but our classifiers use far less rules (3 compared to 60) and less features. Comparing the fuzzy sets in Fig. 4 with the data in Fig. 3 shows that the obtained rules are highly interpretable.

**Table 1.** Three rule fuzzy classifier (L=low, M=medium, H=high).

|       | 1   | 2   | 3   | 4    | 5   | 6   | 7   | 8     | 9   | 10  | 11  | 12  | 13  |       |
|-------|-----|-----|-----|------|-----|-----|-----|-------|-----|-----|-----|-----|-----|-------|
|       | Alc | Mal | Ash | aAsh | Mag | Tot | Fla | nFlav | Pro | Col | Hue | OD2 | Pro | Class |
| $R_1$ | H   | -   | -   | -    | -   | -   | H   | -     | -   | M   | H   | -   | H   | 1     |
| $R_2$ | L   | -   | -   | -    | -   | -   | -   | -     | -   | L   | H   | -   | L   | 2     |
| $R_3$ | H   | -   | -   | -    | -   | -   | L   | -     | -   | H   | L   | -   | L   | 3     |

**Table 2.** Classification rates on the Wine data for ten independent runs.

| Method               | Best result | Aver result     | Worst result | Rules | Model eval |
|----------------------|-------------|-----------------|--------------|-------|------------|
| Corcoran and Sen [3] | 100%        | 99.5%           | 98.3%        | 60    | 150000     |
| Ishibuchi et al. [5] | 99.4%       | 98.5%           | 97.8%        | 60    | 6000       |
| This paper           | 99.4 %      | various schemes | 98.3%        | 3     | 4000-8000  |

## 4 Conclusion

The design of fuzzy rule-based classifiers is approached by combining separate tools for feature selection, model initialization, model reduction and model tuning. It is shown that these can be applied in an iterative way. A covariance-based model initialization method is applied to obtain an initial fuzzy classifier. Successive application of feature selection, rule base simplification and GA-based tuning resulted in compact and accurate classifiers. The proposed approach was successfully applied to the Wine data.

## References

1. Babuška R. (1998) *Fuzzy Modeling for Control*. Kluwer Academic Publishers, Boston.
2. Cios K.J., Pedrycz W., Swiniarski R.W. (1998) *Data Mining Methods for Knowledge Discovery*. Kluwer Academic Press, Boston.
3. Corcoran A.L., Sen S. (1994) Using real-valued genetic algorithms to evolve rule sets for classification. In *IEEE-CEC*, June 27-29, 120–124, Orlando, USA.
4. Gustafson, D.E., Kessel, W.C. (1979) Fuzzy clustering with a fuzzy covariance matrix, In *Proc. IEEE CDC*, 761-766, San Diego, USA.
5. Ishibuchi H., Nakashima T., Murata T. (1999) Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems. *IEEE Trans. SMC-B* **29**, 601–618.
6. Nauck D., Kruse R. (1999) Obtaining interpretable fuzzy classification rules from medical data. *Artificial Intelligence in Medicine* **16**, 149–169.
7. Roubos J.A., Setnes M. (2000) Compact fuzzy models through complexity reduction and evolutionary optimization. In *FUZZ-IEEE*, 762-767, May 7-10, San Antonio, USA.
8. Setiono R. (2000) Generating concise and accurate classification rules for breast cancer diagnosis. *Artificial Intelligence in Medicine* **18**, 205-219.
9. Setnes M., Babuška R. (1999) Fuzzy relational classifier trained by fuzzy clustering, *IEEE Trans. SMC-B* **29**, 619–625
10. Setnes M., Babuška R., Kaymak U., van Nauta Lemke H.R. (1998) Similarity measures in fuzzy rule base simplification. *IEEE Trans. SMC-B* **28**, 376–386.
11. Setnes M., Roubos J.A. (1999) Transparent fuzzy modeling using fuzzy clustering and GA's. In *NAFIPS*, June 10-12, 198–202, New York, USA.
12. Setnes M., Roubos J.A. (in press, 2000) GA-fuzzy modeling and classification: complexity and performance. *IEEE Trans. FS*.
13. Valente de Oliveira J. (1999) Semantic constraints for membership function optimization. *IEEE Trans. FS* **19**, 128–138.