A Comparative Study of Fuzzy Classifiers on Breast Cancer Data

Ravi. Jain[1], and Ajith. Abraham[2]
[1]School of Information Technology,
James Cook University (Cairns Campus),
Smithfield, Australia 4878.
{ravi.jain@jcu.edu.au}

[2] Computer Science Department,
Oklahoma State University (Tulsa Campus),
700 N Greenwood Avenue, Tulsa, Oklahoma OK 74106, USA.
{aa@cs.okstate.edu}

Abstract. In this paper, we examine and compare the performance of four fuzzy rule generation methods on Wisconsin breast cancer data [2]. These methods were reported by Ishibuchi [1] et al. For the diagnosis of breast cancer, the determination of the presence of *benign/malignant* breast tumors represents a very complex problem (even for an experienced cytologist). The goal of this paper is to compare and contrast fuzzy rule generation methods on breast cancer data that involve no time-consuming tuning procedures. Since The performance of each approach for test patterns (i.e., the generalization of ability of each approach) is evaluated by cross validation techniques on breast cancer data sets.

1. Introduction

Breast cancer is the most common cancer in women in many countries. Most breast cancers are detected as a lump/mass on the breast, by self-examination/mammography, or by both [3]. Screening mammography is the best tool available for detecting cancerous lesions before clinical symptoms appear. Surgery - either a biopsy or a lumpectomy have been the most common method to remove them. Fine needle aspiration (FNA) of breast masses is a cost-effective, non-traumatic, and mostly non-invasive diagnostic test that obtains information needed to evaluate malignancy. Recently, a new minimally invasive technique, which uses super-cooled nitrogen to freeze and shrink a non-cancerous tumor and destroy the blood vessels feeding the growth of the tumour has been developed [4] in USA. Artificial Intelligence techniques (such as neural networks, fuzzy logic, genetic programming and combination of these methods) have attracted a lot of attention in the area of medical diagnosis. These techniques are successfully applied to a wide variety of decision-making problems. Several methods have been proposed for generating fuzzy if-then rules for pattern classification problems. Neural network architectures and learning mechanisms were used for tuning fuzzy if-then rules. Genetics-based machine learning frameworks were used for generating and selecting emphasized the direct generation of fuzzy if-then rules from training patterns. While many sophisticated approaches have been proposed, very simple fuzzy if-then rules were also used as fuzzy classifiers. The main aim of this paper is to examine the performance of some direct rule generation methods that involve no time-consuming tuning procedures on *breast cancer data.* The first one generates fuzzy if-then rules using the mean and the standard deviation of attribute values. The mean and the standard deviation were used as parameters of membership functions when they compared fuzzy classifiers with neural network classifiers. The second approach generates fuzzy if-then rules using the histogram of attributes values. In the first two approaches, a single fuzzy if-then rule is generated for each class. The third approach generates fuzzy if-then rules with certainty each attribute into homogeneous fuzzy sets. In the fourth approach, only overlapping areas are partitioned. This approach is a modified version of the third approach.

2. Rule Generation Procedure

In this section, we explain each of four approaches examined in this paper. The performance of each approach is evaluated by computer simulations on breast cancer data sets.

Assuming that we have an n-dimensional c-class pattern classification problem whose pattern space is an n-dimensional unit cube $[0,1]^n$ and m patterns $x_p = (x_{p1},\ldots,x_{pn})$, $p = 1,2,\ldots,m$, are given for generating fuzzy if-then rules where $x_p \in [0,1]$ for p =1,2,..., m, i =1,2,...,n. In computer simulations, all attribute values are normalized into the unit interval [0,1].

3. Rule Generation based on the Mean and the Standard Deviation of Attribute Values

In this approach, a single fuzzy if-then rule is generated for each class. The fuzzy if-then rule for the k^{th} class can be written as

If x_1 is A_1^k and ... and x_n is A_1^k then Class k, (1)

where A_i^k is an antecedent fuzzy set for the i^{th} attribute. The membership function of A_i^k is specified as of

$$A_i^k(x_i) = \exp\left(-\frac{\left(x_i - \mu_i^k\right)^2}{2\left(\sigma_i^k\right)^2}\right) \tag{2}$$

where μ_i^k is the mean of the i^{th} attribute values x_{pi} of Class k patterns, and σ_i^k is the standard deviation. Fuzzy if-then rules for the two-dimensional two class pattern classification problem are written as follows:

IF x_3 is A_3^1 and x_4 is A_4^1 THEN Class 2 (3)

IF x_3 is A_3^2 and x_4 is A_4^2 THEN Class 3 (4)

The membership function of each antecedent fuzzy set is specified by the mean and the standard deviation of attribute values. For a new pattern $x_p = (x_{p3},x_{p4})$, the winner rule is determined as follows:

$$A_3^*(x_{p3}).A_2^*(x_{p4}) = \max\left\{A_1^k(x_{p3}).A_2^k(x_{p4}) \middle| k = 1,2\right\} \tag{5}$$

For each attribute 20 membership functions $f_h()$, h=1,2,...,20 were used. The fuzzy partition used only for calculating the histogram.

4. Rule Generation based on the Histogram of Attribute Values

In this method the use of histogram an antecedent membership function and each attribute is partitioned into several fuzzy sets. We used 20 membership functions $f_h(.)$, h=1,2,...,20 for each attribute in computer simulations.

The smoothed histogram $m_i^k(x_i)$ of Class k patterns for the i[th] attribute is calculated using the 20 membership functions f_h (.) as follows:

$$m_i^k(x_i) = \frac{1}{m^k} \sum_{x_p \in Class\ \mathrm{k}} f_h\left(x_{pi}\right) \tag{6}$$

$$for\ \beta_{\mathrm{h-1}} \leq x_i \leq \beta_h,\ \mathrm{h=1,2,...,20}$$

where m_k is the number of Class k patterns, $\left[\beta_{h-1,}\beta_h\right]$ is the h[th] crisp interval corresponding to the 0.5-level set of the membership function f_h (.):

$$\beta_1 = 0,\ \ \beta_{20} = 1, \tag{7}$$

$$\beta_h = \frac{1}{20-1}\left(h - \frac{1}{2}\right)\ \text{for h=1,2,...,19} \tag{8}$$

The smoothed histogram in (6) is normalized so that its maximum value is 1. As in the first approach based on the mean and the standard deviation, a single fuzzy if-then rule in (2) is generated for each class in the second approach.

5. Rule Generation of based on Simple Fuzzy Grid

In the first two approaches, a single fuzzy if-then rule was generated for each class using the information about training patterns. On the contrary, many fuzzy if-then rules are generated in the third approach by partitioning each attribute into homogeneous fuzzy sets.
One disadvantage of this approach is that the number of possible fuzzy if-then rules exponentially increases with the dimensionality of the pattern space. For coping with this difficulty, some GA-based rule selection approaches have been proposed to find a compact rule set [7,10,12]. The number of fuzzy if-then rules can be also decreased by feature selection [14].

Because the specification of each membership function does not depend on any information about training patterns, this approach uses fuzzy if-then rules with certainty grades. The local information about training patterns in the corresponding fuzzy subspace is used for determining the consequent class and the grade of certainty.

In this approach, fuzzy if-then rules of the following type are used:

IF x_1 is A_{j1} and ... and x_n is A_{jn}

THEN

Class C_j, with CF = CF_j, j =1,2,..., N (9)

where j indexes the number of rules, N is the total number of rules, A_{ji} is the antecedent fuzzy set of the i-th rule for the i-th attribute, C_j; is the consequent class, and CFj is the grade of certainty. The consequent class and the grade of certainty of each rule are determined by the following simple heuristic procedure [13]:

Step 1: Calculate the compatibility of each training pattern $x_p = (x_{p1}, x_{p2}, \ldots, x_{pn})$ with the j-th fuzzy if-then rule by the following product operation:

$$\pi_j\left(x_p\right) = A_{j1}\left(x_{p1}\right) \times \ldots \times A_{jn}\left(x_{pn}\right), \ p = 1, 2, \ldots, m. \tag{10}$$

Step 2: For each class, calculate the sum of the compatibility grades of the training patterns with the j-th fuzzy if-then rule R_j:

$$\beta_{class\ k}\left(R_j\right) = \sum_{x_p \in class\ k}^{n} \pi\left(x_p\right), \ \mathrm{k=1,2,\ldots,c} \tag{11}$$

where $\beta_{class\ k}\left(R_j\right)$ the sum of the compatibility grades of the training patterns in Class k with the j-th fuzzy if-then rule R_j.

Step 3: Find Class A_j^* that has the maximum value $\beta_{class\ k}\left(R_j\right)$:

$$\beta_{class\ \mathrm{k}_\mathrm{j}^*} = Max\{\beta_{class\ 1}(R_j), \ldots, \beta_{class\ \mathrm{c}}(R_j)\} \tag{12}$$

If two or more classes take the maximum value or no training pattern compatible with the j-th fuzzy if-then rule (i. e., if $\beta_{\mathrm{Class}\ k}(R_j)=0$ for k =1,2,..., c), the consequent class C_i can not be determined uniquely. In this case, let C_i be ϕ. If a single class takes the maximum value, the consequent class C_j *is* determined by (7).

Step 4: If the consequent class C_i is 0, let the grade of certainty CF_j be $CF_j = 0$. Otherwise the grade of certainty CF_j *is* determined as follows:

$$CF_j = \frac{(\beta_{class\ \mathrm{k}_\mathrm{j}^*} - \overline{\beta})}{\sum_{k=1}^{c} \beta_{class\ \mathrm{k}}\left(\mathrm{R_j}\right)} \tag{13}$$

$$CF_j = \frac{(\beta_{class\ k_j^*} - \bar{\beta})}{\sum_{k=1}^{c} \beta_{class\ k}\left(R_j\right)} \tag{14}$$

6. Rule Generation based on Fuzzy Partition of Overlapping Areas

In the third approach, the shape of each membership function was specified without utilizing any information about training patterns. A simple modification of the third approach is to partition only overlapping areas.

This approach generates fuzzy if-then rules in the same manner as the simple fuzzy grid approach except for the specification of each membership function. Because this approach utilizes the information about training patterns for specifying each membership function as in the first and second approaches, the performance of generated fuzzy if then rules is good even when we do not use the certainty grade of each rule in the classification phase. In this approach, the effect of introducing the certainty grade to each rule is not large if compared with the third approach. In computer simulations of the next section, we used fuzzy if-then rules with certainty grades in this approach as in the third approach.

7. Simulation Results

The Wisconsin *breast cancer* dataset was obtained from repository of machine learning database University of California, Irvine. This data set has 32 attributes (30 real-valued input features) and 569 instances of which 357 benign and 212 malignant class. There are several studies based on this database. Bennet and Mangasarian [19] used linear programming techniques, obtaining a 99.6% classification rate on 487 cases (the reduced database available at the time). However, diagnostic decisions are essentially black boxes, with no explanation as to how they were attained.

In the first approach, a single fuzzy if-then rule was generated for each class using the mean and the standard deviation of attribute values. A single fuzzy if-then rule for each class was not sufficient for the breast cancer data. In the second approach, a single fuzzy if-then rule was generated for each class using the histogram of attribute values. The third approach generated fuzzy if-then rules by homogeneously partitioning each attribute. Thus a pattern space was partitioned into a simple fuzzy grid. The information about attribute values was not used for specifying the membership function of each antecedent fuzzy set. The local information of training patterns was utilized when the consequent class and the certainty grade were specified. The 10-fold cross-validation procedure was iterated 5 times using different partitions of data sets into ten subsets. Simulation results are summarized in Table 1. In this table, the 85.96% classification rate by the fuzzy classifier systems is the best result among simulation results. From this table, we can see that the performance of the fuzzy rule-based classification systems except for the histogram approach is comparable to the best result by the fuzzy classifier systems.

Table 1. Simulation Results

Methods	With CF	Single winner rule (without CF)
Mean & Deviation	85.94%	85.94
Histogram	62.74%	84.36
Simple Grid	62.39	62.39
Modified Grid	62.57%	85.96

8. Conclusion

In this paper, we examined the performance of four fuzzy rule generation methods that could generate fuzzy if-then rules directly from training patterns.

Grid-based approaches, had high generalization ability but the number of fuzzy if-then rules is exponentially increased with the dimensionality of the pattern space. Thus a large number of fuzzy if-then rules are usually generated for real-world pattern classification problems. This leads to several drawbacks: overfitting to training patterns, large memory storage requirement, and slow inference speed. The last approach was a modified version of the simple fuzzy grid approach. Only overlapping areas of different classes were partitioned into fuzzy subspaces.

On the contrary, the number of fuzzy if-then rules in the first two approaches is the same as the number of classes. As shown in this paper, a single fuzzy if-then rule for each class is not always sufficient for real-world pattern classification problems. While each approach is very simple and has the above-mentioned drawbacks, generated fuzzy rule-based systems have high classification ability. The performance of fuzzy rule based systems can be improved by feature selection and rule selection.

References

1. Ishibuchi, H., and Nakashima, T., A Study on Generating Classification Rules Using Histogram Edited by Jain L.C., and Jain, R.K., KES'98, pp. 132-140
2. Merz J., and Murphy, P.M., UCI repository of machine learning databases. http://www.ics.uci.edu/˜learn/MLRepository.html, (1996)
3. DeSilva, C.J.S. et al. Artificial Neural networks and Breast Cancer Prognosis The Australian Computer Journal, 26, pp. 78-81,(1994)
4. The Weekend Australian, Health Section, p 7. July, 13-14, (2002)
5. Cios, K.J., et.al. Using Fuzzy Sets to Diagnose Coronary Artery Stenois, IEEE Computer, pp. 57-63, (1991)
6. Ishibuchi, H. et.al Pattern and Feature Selection By Genetic Algorithms in Nearest Neighbour Classification unpublished report.
7. Ishibuchi, H. et.al, A fuzzy classifier system that generates fuzzy if-then rules for pattern classification problems, Proc. Int. Conf. Evolutionary Computat. Perth, Australia, vol. 2, pp. 759-764, (1995)
8. Pal, S. K. and Mitra, S. Multilayer perceptron, fuzzy sets, and classification, IEEE Trans. on Neural Networks, 3, no. 5, pp. 683-697, (1992)
9. Nauck, D., and Kruse, R., A neuro-fuzzy method to learn fuzzy classification rules from data, Fuzzy Sets and Systems, vol. 89, pp. 277-288, (1997)
10. Ishibuchi, H., Nozaki, K.,Yamamoto, N., and Tanaka, H., Selecting fuzzy if-then rules for classification problems using genetic algorithms, IEEE Trans. on Fuzzy

Systems, vol. 3, no. 3, pp. 260-270, (1995)
11. H. Ishibuchi, H., T. Nakashima, T., and Murata, T., A fuzzy classifier system that generates fuzzy if then rules for pattern classification problems, Proc. 2nd IEEE International Conference on Evolutionary Computation, pp.759-764, Perth, Australia, (1995)
12. Yuan,Y., and Zhuang,H., A genetic algorithm for generating fuzzy classification rules, Fuzzy Sets and Systems, vol. 84, no. 1, pp. 1-19, (1996)
13. Ishibuchi, H., Nozaki, K., and Tanaka, H., Distributed representation of fuzzy rules and its application to pattern classification, Fuzzy Sets and Systems, vol. 52, no. 1, pp. 21-32, (1992)
14. H. Ishibuchi, H., Nakashima,T., and Morisawa, T.,Simple fuzzy rule-based classification systems performed well on commonly used real-world data sets, Proc. of North American Fuzzy Information Processing Society Meeting, Syracuse, pp 21-24, (1997)
15. Weiss, S.,M., and Kulikowski, C.,K., Computer Systems That Learn, Morgan Kaufmann Publishers, San Mateo, CA, (1991)
16. Goldberg, D.E., *Genetic algorithms*. Addison-Wesley Publishing Company, (1989)
17. Goldberg, D.E., Genetic and evolutionary algorithms come of age. Communications of the ACM, vol. 37, No. 3, pp. 113-119, (1994)
18. Zadeh, L.A., Fuzzy Logic IEEE Computer, pp. 83-93 (1988)
19. Bennett, K.,P., and Mangasarian, O.L., Neural network training via linear programming. In P. M. Pardalos, editor, Advances in Optimization and Parallel Computing, pages 56-57. Elsevier Science, 1992