8

Fuzzy Classification for Gene Expression Data Analysis

Gerald Schaefer[1], Tomoharu Nakashima[2] and Yasuyuki Yokota[2]

[1] School of Engineering and Applied Science, Aston University, U.K.
[2] Department of Computer Science and Intelligent Systems, Osaka Prefecture University, Japan

Summary. Microarray expression studies measure, through a hybridisation process, the levels of genes expressed in biological samples. Knowledge gained from these studies is deemed increasingly important due to its potential of contributing to the understanding of fundamental questions in biology and clinical medicine. One important aspect of microarray expression analysis is the classification of the recorded samples which poses many challenges due to the vast number of recorded expression levels compared to the relatively small numbers of analysed samples. In this chapter we show how fuzzy rule-based classification can be applied successfully to analyse gene expression data. The generated classifier consists of an ensemble of fuzzy if-then rules which together provide a reliable and accurate classification of the underlying data. Experimental results on several standard microarray datasets confirm the efficacy of the approach.

8.1 Introduction

Microarray expression studies measure, through a hybridisation process, the levels of genes expressed in biological samples. Knowledge gained from these studies is deemed increasingly important due to its potential of contributing to the understanding of fundamental questions in biology and clinical medicine. Microarray experiments can either monitor each gene several times under varying conditions or analyse the genes in a single environment but in different types of tissue. In this chapter we focus on the latter where one important aspect is the classification of the recorded samples. This can be used to either categorise different types of cancerous tissues as in [8] where different types of leukemia are identified, or to distinguish cancerous tissue from normal tissue as done in [2] where tumor and normal colon tissues are analysed.

One of the main challenges in classifying gene expression data is that the number of genes is typically much higher than the number of analysed samples. Also is it not clear which genes are important and which can be omitted without reducing the classification performance. Many pattern classification techniques have been employed to analyse microarray data. For example, Golub *et al.* [8] used a weighted voting scheme, Fort and Lambert-Lacroix [6] employed partial least squares and logistic

G. Schaefer et al.: *Fuzzy Classification for Gene Expression Data Analysis*, Studies in Computational Intelligence (SCI) **94**, 209–218 (2008)
www.springerlink.com

regression techniques, whereas Furey *et al.* [7] applied support vector machines. Dudoit *et al.* [5] investigated nearest neighbour classifiers, discriminant analysis, classification trees and boosting, while Statnikov *et al.* [16] explored several support vector machine techniques, nearest neighbour classifiers, neural networks and probabilistic neural networks. In several of these studies it has been found that no one classification algorithm is performing best on all datasets (although for several datasets SVMs seem to perform best) and that hence the exploration of several classifiers is useful. Similarly, no universally ideal gene selection method has yet been found as several studies [14, 16] have shown.

In this chapter we apply fuzzy rule based classification concepts to the classification of microarray expression data and show, based on a series of experiments, that it affords good classification performance for this type of problem. Several authors have used fuzzy logic to analyse gene expression data before. Woolf and Wang [19] used fuzzy rules to explore the relationships between several genes of a profile while Vinterbo *et al.* [18] used fuzzy rule bases to classify gene expression data. However, Vinterbo's method has the disadvantage that it allows only linear discrimination. Furthermore, they describe each gene by only 2 fuzzy partitions ('up' and 'down') while we also explore division into more intervals and show that by doing so increased classification performance is possible.

8.2 Methods

While in the past fuzzy rule-based systems have been mainly applied to control problems [17], more recently they have also been applied to pattern classification problems. Various methods have been proposed for the automatic generation of fuzzy if-then rules from numerical data for pattern classification [9–11] and have been shown to work well on a variety of problem domains.

Pattern classification typically is a supervised process where, based on set of training samples with known classifications, a classifier is derived that performs automatic assignment to classes based on unseen data. Let us assume that our pattern classification problem is an n-dimensional problem with C classes (in microarray analysis C is often 2) and m given training patterns $\mathbf{x}_p = (x_{p1}, x_{p2}, \ldots, x_{pn})$, $p = 1, 2, \ldots, m$. Without loss of generality, we assume each attribute of the given training patterns to be normalised into the unit interval $[0, 1]$; that is, the pattern space is an n-dimensional unit hypercube $[0, 1]^n$. In this study we use fuzzy if-then rules of the following type as a base of our fuzzy rule-based classification systems:

$$\begin{aligned} &\text{Rule } R_j\text{: If } x_1 \text{ is } A_{j1} \text{ and } \ldots \text{ and } x_n \text{ is } A_{jn} \\ &\text{then Class } C_j \text{ with } CF_j, \quad j = 1, 2, \ldots, N, \end{aligned} \tag{8.1}$$

where R_j is the label of the j-th fuzzy if-then rule, $A_{j1}, \ldots, A_{jn}$ are antecedent fuzzy sets on the unit interval $[0, 1]$, C_j is the consequent class (i.e. one of the C given classes), and CF_j is the grade of certainty of the fuzzy if-then rule R_j. As antecedent fuzzy sets we use triangular fuzzy sets as in Figure 8.1 where we show a partition of the unit interval into a number of fuzzy sets.

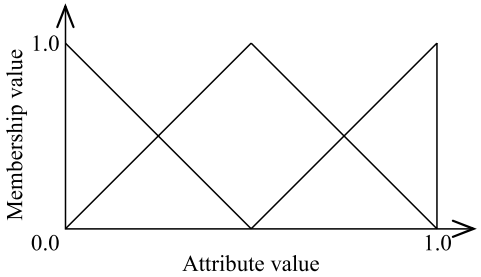

Fig. 8.1. Example triangular membership function ($L = 3$)

Our fuzzy rule-based classification system consists of N fuzzy if-then rules each of which has a form as in Equation (8.1). There are two steps in the generation of fuzzy if-then rules: specification of antecedent part and determination of consequent class C_j and the grade of certainty CF_j. The antecedent part of fuzzy if-then rules is specified manually. Then the consequent part (i.e. consequent class and the grade of certainty) is determined from the given training patterns [13]. In [12] it is shown that the use of the grade of certainty in fuzzy if-then rules allows us to generate comprehensible fuzzy rule-based classification systems with high classification performance.

8.2.1 Fuzzy Rule Generation

Let us assume that m training patterns $\mathbf{x}_p = (x_{p1}, \ldots, x_{pn})$, $p = 1, \ldots, m$, are given for an n-dimensional C-class pattern classification problem. The consequent class C_j and the grade of certainty CF_j of the if-then rule are determined in the following two steps:

1. Calculate $\beta_{\text{Class } h}(j)$ for Class h as

$$\beta_{\text{Class } h}(j) = \sum_{\mathbf{x}_p \in \text{Class } h} \mu_j(\mathbf{x}_p), \tag{8.2}$$

   where

$$\mu_j(\mathbf{x}_p) = \mu_{j1}(x_{p1}) \cdot \ldots \cdot \mu_{jn}(x_{pn}), \tag{8.3}$$

   and $\mu_{jn}(\cdot)$ is the membership function of the fuzzy set A_{jn}. In this chapter we use triangular fuzzy sets as in Figure 8.1.
2. Find Class $\hat{h}$ that has the maximum value of $\beta_{\text{Class } h}(j)$:

$$\beta_{\text{Class } \hat{h}}(j) = \max_{1 \le k \le C} \{\beta_{\text{Class } k}(j)\}. \tag{8.4}$$

If two or more classes take the maximum value, the consequent class C_j of the rule R_j can not be determined uniquely. In this case, specify C_j as $C_j = \phi$. If a single

class $\hat{h}$ takes the maximum value, let C_j be Class $\hat{h}$. The grade of certainty CF_j is determined as

$$CF_j = \frac{\beta_{\text{Class } \hat{h}}(j) - \bar{\beta}}{\sum_h \beta_{\text{Class } h}(j)} \tag{8.5}$$

with

$$\bar{\beta} = \frac{\sum_{h \neq \hat{h}} \beta_{\text{Class } h}(j)}{C-1}. \tag{8.6}$$

8.2.2 Fuzzy Reasoning

Using the rule generation procedure outlined above we can generate N fuzzy if-then rules as in Equation (8.1). After both the consequent class C_j and the grade of certainty CF_j are determined for all N rules, a new pattern $\mathbf{x} = (x_1, \ldots, x_n)$ can be classified by the following procedure:

1. Calculate $\alpha_{\text{Class } h}(\mathbf{x})$ for Class h, $j = 1, \ldots, C$, as

$$\alpha_{\text{Class } h}(\mathbf{x}) = \max\{\mu_j(\mathbf{x}) \cdot CF_j | C_j = h\}, \tag{8.7}$$

2. Find Class h' that has the maximum value of $\alpha_{\text{Class } h}(\mathbf{x})$:

$$\alpha_{\text{Class } h'}(\mathbf{x}) = \max_{1 \leq k \leq C} \{\alpha_{\text{Class } k}(\mathbf{x})\}. \tag{8.8}$$

If two or more classes take the maximum value, then the classification of $\mathbf{x}$ is rejected (i.e. $\mathbf{x}$ is left as an unclassifiable pattern), otherwise we assign $\mathbf{x}$ to Class h'.

8.2.3 Rule splitting

It is generally known that any type of rule-based system suffers from the curse of dimensionality. That is, the number of generated rules increases exponentially with the number of attributes involved. Our fuzzy rule-based classifier is no exception, in particular considering that for successful classification of microarray data typically at least a few dozens genes are selected. For example, based on the selection of 50 genes, the classifier would generate $2^{50} = 1.1259 * 10^{15}$ rules even if we only partition each axis into two which is clearly prohibitive both in terms of storage requirements and computational complexity. We therefore apply a rule splitting step and limit the number of attributes in a fuzzy if-then rule to 2. As the number of combinations of attribute pairs is $\binom{50}{2} = 1225$ for 50 genes and as for two fuzzy sets for each attribute $2^2 = 4$ rules are necessary in total we need only $4 * 1225 = 4900$ rules, a significantly lower number than 2^{50}. Of course, techniques can be employed to further decrease this number; although we refrained from it in our experiments a rule pruning step similar to the one outlined in [18] can be applied to arrive at a smaller and more compact classifier rule base.

8.3 Results and Discussion

Before we report on the experimental results we obtained from our classification method we wish to point out a few important differences of our work compared to the fuzzy classifier employed by Vinterbo *et al.* in [18]. The algorithm in [18] represents a fairly simple fuzzy classification approach and provides only linear separation of classes. That is, separate classes can be divided by a hyperplane in feature space. In contrast, with our classifier it is also possible to perform non-linear separation. While at the moment this might be of little effect (due to the limited size of data samples) as has been shown in [3] with increasing sizes of datasets this could prove useful in the near future. Furthermore, our classifier employs the concept of grade of certainty which not only provides improved classification performance but can also provide an additional feedback and/or a means for pattern rejection (due to too low classification confidence). Finally, the classifier in [18] only employed 2 fuzzy partitions per gene to model up and down regulation. While this might seem intuitive it does not necessarily afford best classification performance. In our work we experimented with up to five partitions per attribute.

To demonstrate the usefulness and efficacy of our proposed approach we evaluated our proposed method on several gene expression data sets that are commonly used in the literature. In the following we characterise each dataset briefly:-

- Colon dataset [2]: This dataset is derived from colon biopsy samples. Expression levels for 40 tumor and 22 normal colon tissues were measured for 6500 genes using Affymetrix oligonucleotide arrays. The 2000 genes with the highest minimal intensity across the tissues were selected. We pre-process the data following [5], i.e. perform a thresholding [floor of 100 and ceil of 16000] followed by filtering [exclusion of genes with max/min < 5 and (max-min) < 500] and $\log_{10}$ transformation.
- Leukemia dataset [8]: Bone marrow or peripheral blood samples were taken from 47 patients with acute lymphoblastic leukemia (ALL) and 25 patients with acute myeloid leukemia (AML). The ALL cases can be further divided into 38 B-cell ALL and 9 T-cell ALL samples and it is this 3-class division that we are basing our experiments on rather than the simpler 2-class version which is more commonly referred to in the literature. Each sample is characterised by 7129 genes whose expression levels where measured using Affymetrix oligonucleotide arrays. The same preprocessing steps as for the Colon dataset are applied.
- Lymphoma dataset [1]: This dataset contains gene expression data of diffuse large B-cell lymphoma (DLBCL) which is the most common subtype of non-Hodgink's lymphome. In total there are 47 samples of which 24 are of germinal centre B-like and the remaining 23 of activated B-like subtype. Each sample is described by 4026 genes, however there are many missing values. For simplicity we removed genes with missing values from all samples.
- Ovarian dataset [15]: This data stems from experiments designed to identify proteomic patterns in serum that distingiush ovarian cancer from non-cancer. The proteomic patterns were obtained through mass spectroscopy and there are 91

non-cancer and 162 ovarian cancer samples. While this is not a gene expression dataset it shares many commonalities with such which is the reason why we have included it in our study. The relative amplitude of the intensity at each of the 15154 molecular mass/charge (M/Z) identities was normalised against the most and least intense values according to: $NV = (V - Min)/(Max - Min)$ where NV is the normalised and V is the original value while Min and Max are the minimum and maximum intensities in the data stream [14].

Although all datasets except for the Leukemia set represent 2-class problems due to the large number of genes involved any rule based classification system would consist of a very large number of rules and hence represent a fairly complex process. Also, not all genes are equally important for the classification task at hand. We therefore sort the significance of genes according to the BSS/WSS (the ratio of between group to within group sum of squares) criterion used in [5] and consider only the top 50 respectively 100 genes as input for our classification problem.

In a first step we train our classifiers on all samples available and perform the resulting classification performance. This of course provides only a partial indication as the training data and test data are identical. We therefore perform standard leave-one-out cross-validation where classifier training is performed on all available data except for the sample to be classified and this process is performed for all samples [1]. Fuzzy rule based classifiers with partition sizes L between 2 and 5 partitions for each gene were constructed following the process described in Section 8.2. To evaluate the achieved results we also implemented nearest neighbour and CART classifiers. The nearest neighbour classifier we employ searches through the complete training data to identify the sample which is closest to a given test input and assigns the identified sample's class. CART [4] is a classical rule based classifier which builds a recursive binary decision tree based on misclassification error of subtrees.

The results on the four datasets are given in Tables 8.1 to 8.4 where detailed performance on training and unseen (leave-one-out) test data is shown. Given are the number of correctly classified samples (CR), the number of incorrectly classified or unclassified samples (FR), and the classification accuracy (Acc.), i.e. the percentage of correctly classified samples.

Looking at the results for the Colon dataset which are given in Table 8.1, on training data the fuzzy classifier with $L = 5$ and the nearest neighbour classifier both achieve 100% classification accuracy based on 50 genes while for the case of 100 genes also the fuzzy classifier with $L = 4$ achieves perfect classification. More interesting of course is the performance on test data, i.e. the results of the leave-one-out cross validation we performed. Here for the case of 50 selected features the fuzzy classifier with 3 partitions performs best with a classification accuracy of 85.48% which corresponds to 9 incorrectly classified cases while nearest neighbour classification and CART produce 13 and 14 errors respectively. However when selecting the 100 top genes the nearest neighbour classifier performs slightly better than the fuzzy system. It is interesting to compare the performance of the fuzzy rule-based classifier

[1] It should be noted that the top 50 respectively 100 genes were selected solely based on the training set

Table 8.1. Classification performance on Colon dataset given in terms of number of correctly classified samples (CR), falsely classified or unclassified samples (FR), and classification accuracy (Acc.). Results are given both for training data and for leave-one-out cross validation. Experiments were performed with 50 and 100 selected genes respectively and with a varying number L of partitions per gene. For comparison results obtained using a nearest neighbour classifier and a rule-based CART classifier are also listed

		training data			test data		
n	classifier	CR	FR	Acc.	CR	FR	Acc.
50	fuzzy $L=2$	55	7	88.71	50	12	80.65
	fuzzy $L=3$	56	6	90.32	**53**	**9**	**85.48**
	fuzzy $L=4$	59	3	95.16	52	10	83.87
	fuzzy $L=5$	**62**	**0**	**100**	48	14	77.42
	nearest neighbour	**62**	**0**	**100**	49	13	79.03
	CART	59	3	95.16	48	14	77.42
100	fuzzy $L=2$	53	9	85.48	44	18	70.97
	fuzzy $L=3$	59	3	95.16	51	11	82.26
	fuzzy $L=4$	**62**	**0**	**100**	50	12	80.65
	fuzzy $L=5$	**62**	**0**	**100**	46	16	74.19
	nearest neighbour	**62**	**0**	**100**	**52**	**10**	**83.87**
	CART	60	2	96.77	45	17	72.58

Table 8.2. Classification performance on Leukemia dataset, laid out in the same fashion as Table 8.1

		training data			test data		
n	classifier	CR	FR	Acc.	CR	FR	Acc.
50	fuzzy $L=2$	68	4	94.44	66	6	91.67
	fuzzy $L=3$	71	1	98.61	68	4	94.44
	fuzzy $L=4$	**72**	**0**	**100**	67	5	93.06
	fuzzy $L=5$	71	1	98.61	66	6	91.67
	nearest neighbour	**72**	**0**	**100**	**70**	**2**	**97.22**
	CART	72	0	100	47	25	65.28
100	fuzzy $L=2$	67	5	93.06	63	8	87.50
	fuzzy $L=3$	71	1	98.61	**71**	**1**	**98.61**
	fuzzy $L=4$	**72**	**0**	**100**	69	3	95.83
	fuzzy $L=5$	**72**	**0**	**100**	67	5	93.06
	nearest neighbour	**72**	**0**	**100**	70	2	97.22
	CART	**72**	**0**	**100**	45	27	62.50

when using different numbers of partitions for each attribute. It can be seen that on this dataset the best performance is achieved when using 3 partitions (although on training data alone more partitions afford better performance). In particular it can be observed that the case with $L=2$ as used in the work of Vinterbo *et al.* [18] produces the worst results and hence confirms that increasing the number of fuzzy intervals as we suggest leads to improved classification performance. However, it can also be seen that applying too many partitions can decrease classification performance as is apparent in the case of $L=5$ on test data.

Table 8.3. Classification performance on Lymphoma dataset, laid out in the same fashion as Table 8.1

n	classifier	training data			test data		
		CR	FR	Acc.	CR	FR	Acc.
50	fuzzy $L = 2$	**47**	**0**	**100**	45	2	95.74
	fuzzy $L = 3$	**47**	**0**	**100**	46	1	97.87
	fuzzy $L = 4$	**47**	**0**	**100**	**47**	**0**	**100**
	fuzzy $L = 5$	**47**	**0**	**100**	44	3	93.62
	nearest neighbour	**47**	**0**	**100**	45	2	95.74
	CART	45	2	95.74	36	11	76.60
100	fuzzy $L = 2$	**47**	**0**	**100**	44	3	93.62
	fuzzy $L = 3$	**47**	**0**	**100**	44	3	93.62
	fuzzy $L = 4$	**47**	**0**	**100**	44	3	93.62
	fuzzy $L = 5$	**47**	**0**	**100**	39	8	82.98
	nearest neighbour	**47**	**0**	**100**	**47**	**0**	**100**
	CART	43	4	91.49	38	9	80.85

Table 8.4. Classification performance on Ovarian cancer dataset, laid out in the same fashion as Table 8.1

n	classifier	training data			test data		
		CR	FR	Acc.	CR	FR	Acc.
50	fuzzy $L = 2$	224	29	88.54	224	29	88.54
	fuzzy $L = 3$	249	4	98.42	249	4	98.42
	fuzzy $L = 4$	251	2	99.21	249	4	98.42
	fuzzy $L = 5$	248	5	98.02	247	6	97.63
	nearest neighbour	253	0	99.60	252	1	99.60
	CART	243	10	96.05	228	25	90.12
100	fuzzy $L = 2$	223	30	88.14	221	32	87.35
	fuzzy $L = 3$	248	5	98.02	248	5	98.02
	fuzzy $L = 4$	250	3	98.81	249	4	98.42
	fuzzy $L = 5$	250	3	98.81	249	4	98.42
	nearest neighbour	253	0	99.60	252	1	99.60
	CART	251	2	99.21	239	14	94.47

Turning our attention to the results on the Leukemia dataset which are given in Table 8.2 we see a similar picture. Again the worst performing fuzzy classifier is that which uses only two partitions per gene while the best performing one as assessed by leave-one-out cross validation is the case of $L = 3$. CART performs fairly poorly on this dataset with classification accuracies on the test data reaching only about 65% (despite perfect classification on training data) while nearest neighbour classification performs well again confirming previous observations that despite its simplicity nearest neighbour classifiers are well suited for gene expression classification [5]. The best classification results are achieved by the fuzzy classifier with $L = 3$ for the case of 100 selected genes with a classification accuracy of 98.61% and the nearest neighbour classifier with 97.22% for 50 selected genes.

Table 8.3 lists the results obtained from the Lymphoma dataset. Here all classifiers except CART achieve perfect classification on the training data. Perfect

classification on test data is provided by the fuzzy classifier with $L = 4$ for 50 selected genes and by nearest neighbour classification based on 100 genes.

Finally, we examine the results obtained from the Ovarian dataset which are given in Table 8.4. Here we can see that once again CART provides the poorest classification while nearest neighbour classification achieves the best performance, misclassifying only 1 sample for both 50 and 100 selected genes. In contrast for the best fuzzy classifier 4 samples are misclassified or rejected which confirms previous observations that different classifiers are better suited for different datasets. Again, the case with $L = 2$ achieves significantly worse results for the fuzzy classifier compared to other partitions.

In summary we see that our fuzzy rule-based classifier provides good classification performance on all four datasets clearly outperforming classical rule-based classification and performing fairly similar to a nearest neighbour classifier. However, it should be noted that in our experiments the nearest neighbour classifier always provided a prediction while for our fuzzy classifier we rejected samples which could not uniquely classified (the false rate FR comprises both incorrectly classified and rejected cases). By randomly classifying rejected patterns we could have achieved improved classification accuracy, however this is not in our interest as a random classification hardly provides any insight in the actual expression level data. We also wish to again point out that restriction to 'up' and 'down' regulated partitions for fuzzy classification as in [18] has a negative impact on the classification performance. Our experiments suggest that selecting 3 or 4 fuzzy partitions for each gene can provide much improved classification accuracy. On the other hand using too many partitions as in the cae of $L = 5$ can also have negative effects on the classification performance.

8.4 Conclusions

In this chapter we proposed the application of fuzzy rule based classification for the analysis of gene expression data. The generated classifier consists of an ensemble of fuzzy if-then rules which together provide a reliable and accurate classification of the underlying data. In addition the structure of our classifier has the potential to contribute to the understanding of the underlying data as it is based on a combination of simple, human-understandable rules. Furthermore, for each classification the grade of certainty is provided, which represents the level of confidence the system has in the prediction of a specific sample, and which could hence be utilised in further stages of analysis.

References

1. A.A. Alizadeh, M.B. Eisen, E.E. Davis, C. Ma, I.S. Lossos, A. Rosenwald, J.C. Boldrick, H. Sabet, T. Tran, X. Yu, J.I. Powell, L. Yang, G.E. Marti, T. Moore, J. Hudson, L. Lu, D.B. Lewis, R. Tibshirani, G. Sherlock, W.C. Chan, T.C. Greiner, D.D. Weisenburger,

J.O. Armitage, R. Warnke, R. Levy, W. Wilson, M.R. Grever, J.C. Byrd, D. Botstein, P.O. Brown, and L.M. Staudt. Different types of diffuse large B-cell lymphoma identified by gene expression profiles. *Nature*, 403:503–511, 2000.

2. U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, and A.J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. In *Proc. Natnl. Acad. Sci. USA*, volume 96, pages 6745–6750, 1999.
3. A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini. Tissue classification with gene expression profiles. *Journal of Computational Biology*, 7:559–583, 2000.
4. L. Breiman, J.H. Friedman, R. Olshen, and R. Stone. *Classification and Regression Trees*. Wadsworth, 1984.
5. S. Dudoit, J. Fridlyand, and T.P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97(457):77–87, 2002.
6. G. Fort and S. Lambert-Lacroix. Classification using partial least squares with penalized logistic regression. *Bioinformatics*, 21(7):1104–1111, 2005.
7. T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.
8. T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, and E.S. Lander. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.
9. M. Grabisch and F. Dispot. A comparison of some methods of fuzzy classification on real data. In *2nd Int. Conference on Fuzzy Logic and Neural Networks*, pages 659–662, 1992.
10. H. Ishibuchi and T. Nakashima. Improving the performance of fuzzy classifier systems for pattern classification problems with continuous attributes. *IEEE Trans. on Industrial Electronics*, 46(6):1057–1068, 1999.
11. H. Ishibuchi and T. Nakashima. Performance evaluation of fuzzy classifier systems for multi-dimensional pattern classification problems. *IEEE Trans. Systems, Man and Cybernetics - Part B: Cybernetics*, 29:601–618, 1999.
12. H. Ishibuchi and T. Nakashima. Effect of rule weights in fuzzy rule-based classification systems. *IEEE Trans. Fuzzy Systems*, 9(4):506–515, 2001.
13. H. Ishibuchi, K. Nozaki, and H. Tanaka. Distributed representation of fuzzy rules and its application to pattern classification. *Fuzzy Sets and Systems*, 52(1):21–32, 1992.
14. H. Liu, J. Li, and L. Wong. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. *Gene Informatics*, 13:51–60, 2002.
15. E.F. Petricon et al. Use of proteomic patterns in serum to identify ovarian cancer. *The Lancet*, 359:572–577, 2002.
16. A. Statnikov, C. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy. A comprehensive evaluation of multicategory classification methods for microarray expression cancer diagnosis. *Bioinformatics*, 21(5):631–643, 2005.
17. M. Sugeno. An introductory survey of fuzzy control. *Information Science*, 30(1/2):59–83, 1985.
18. S.A. Vinterbo, E-Y. Kim, and L. Ohno-Machado. Small, fuzzy and interpretable gene expression based classifiers. *Bioinformatics*, 21(9):1964–1970, 2005.
19. P.J. Woolf and Y. Wang. A fuzzy logic approach to analyzing gene expression data. *Physiological Genomics*, 3:9–15, 2000.