Aberystwyth University

Subsethood-based fuzzy modelling and classification

Rasmani, Khairul; Shen, Qiang

Publication date:
2004

Citation for published version (APA):
Rasmani, K., & Shen, Q. (2004). *Subsethood-based fuzzy modelling and classification*. 181-188.

Subsethood-based Fuzzy Modelling and Classification

Khairul Anwar Rasmani
Centre for Intelligent Systems and their Applications, School of Informatics,
The University of Edinburgh
K.A.Rasmani@ sms.ed.ac.uk

Qiang Shen
Department of Computer Science,
University of Wales, Aberystwyth
qqs@aber.ac.uk

Abstract

Reasoning with fuzzy rule-based models has been widely applied to perform various real world classification tasks. The main advantage of this approach is that it supports inferences in the way people think and make judgements. However, in order to gain high classification accuracy, transparency and interpretability of such models has often been ignored. To counter against this limitation, this paper proposes a quantifier-based fuzzy modelling method based on fuzzy subsethood measurements. The resulting induced models are transparent, interpretable whilst still able to provide high classification accuracy. This is confirmed by experimental comparative studies between this work and previous subsethood-based modelling approaches, for classification problems using benchmark datasets.

1 Introduction

Various fuzzy rule-based systems (FRBS) have been developed to solve classification problems with the aims to produce high classification accuracy. However, in many cases transparency and interpretability of the systems are ignored. In certain application domains such as fault diagnosis and performance evaluation it is not only the classification results that are important but also how the results were reached. It is important that the designed fuzzy models are comprehensible by the user and their inference processes explainable to the user [18]. Otherwise the user may turn back to established non-fuzzy approaches which may be less difficult to implement and understood.

Although comprehensibility of fuzzy models has long been an issue in developing fuzzy models, many modelling approaches that were claimed to be 'comprehensive' may not be so when dealing with systems that have a very large number of conditional attributes. Thus, while comprehensibility is used to explain how the models work, applicability of the model for various types and sizes of dataset should also be explored. In particular, problems involving many conditional attributes will typically require a large number of rules. To reduce the complexity of rulesets, several approaches have been suggested such as the use of tolerance [10] and threshold values [3, 22]. Such work has a significant limitation as question will arise on what is the optimal tolerance or threshold value needed to obtain rulesets which will achieve the highest classification accuracy possible and what are the basis upon which to choose the tolerance/threshold values.

One of the techniques to create fuzzy models that are able to provide high classification accuracy is to use weights for modification of attribute values [13]. Crisp weights within the interval of [0,1] may be used for this. However, it is rather unnatural to modify fuzzy terms with non-fuzzy values. Their use may lead to confusion regarding the semantics of the fuzzy labels and the linguistic interpretation of a given fuzzy system [13]. Thus, as an alternative approach is worth investigating - using linguistic hedges or fuzzy quantifiers to modify fuzzy terms suggested in [1, 12]. This paper proposes such a method to generate fuzzy classification models which each have a fixed number of rules according to the number of classification outcomes without the use of any threshold values. Fuzzy quantifiers which are created from fuzzy subsethood values are used to improve the transparency and interpretability of the resulting systems.

The rest of the paper is organised as follows. Section 2 summarises the background theory of fuzzy subsethood values, fuzzy quantifiers and subsethood-based fuzzy rule induction algorithm. Section 3 describes the proposed modelling method, the *Fuzzy*QSBA. Section 4 presents experimental results on benchmark datasets. Finally, conclusions and future directions of this research are outlined in Section 5.

2 Background Theory

To be self-contained, this section gives a brief outline of the underlying theories employed in the present work.

2.1 Fuzzy Subsethood Values

Let A and E be two fuzzy sets defined on the universe U. The fuzzy subsethood value of A with regard to E, $S(E, A)$ represents the degree to which A is subset of E [3, 22]:

$$S(E,A) = \frac{M(E \wedge A)}{M(E)} = \frac{\sum_{x \in U} \nabla(\mu_E(x), \mu_A(x))}{\sum_{x \in U} \mu_E(x)} \tag{1}$$

where $S(E, A) \in [0,1]$ and ∇ is the t-norm operator.

Fuzzy subsethood values have been used to address different problems, including to measure the *degree of truth* of learned fuzzy rules [22], and to promote certain linguistic terms as part of the antecedent of an emerging fuzzy rule [3].

2.2 Fuzzy Quantifiers

In general, quantifier in logic can be expressed as $Q(x)A(x)$ where $Q(x)$ is a quantifier and $A(x)$ is a predicate for variable x [4]. In classical logic, both the quantifier and the predicate can be represented by crisp sets. In fuzzy logic the quantifier may be apply to crisp or fuzzy sets. A quantifier based on fuzzy sets seems to be more suitable for quantifier based fuzzy models which are described in natural language.

Although there exist different types of quantifier, this paper will mainly refer to the fuzzy relative quantifier Q where $\mu_Q(q) \in [0,1]$ with q defined on real interval [0,1]. In particular, Q possesses the *non-decreasing* behavior: $\forall\ q_1, q_2 \in Q,\ q_1 < q_2 \rightarrow \mu_Q(q_1) \le \mu_Q(q_2)$. Such a quantifier is based on the quantified statement *"Q Es are As"* where Q is the linguistic quantifier and A and E are fuzzy values defined on $X = \{x_1, x_2, \ldots, x_k\}$. An example of a quantified statement is *"Most students who get a high score are young"*, where *"most"* is the quantifier, *"high"* and *"young"* are the fuzzy values E and A respectively.

In general, the membership function $\mu_Q(q)$ of a quantifier Q has no direct meaning. Thus in evaluating a fuzzy quantified proposition, a quantification mechanism is needed to map the membership value $\mu_Q(q)$ such that:

$$F : (\mu_Q(q)) \rightarrow \mathrm{I} \in [0,1] \tag{2}$$

In this paper, the result of evaluating the fuzzy relative quantifier is referred to as the truth-value of the quantifier, and is presented using notation T_Q.

Fuzzy quantification technique can be based on generalization of first order logic quantifiers, where the quantification mechanism involves the definition of the existential quantifier, $\exists$ (*exists at least one)* and of the universal quantifier, $\forall$ (*for all)*. However, the two-valued quantification technique seems too strict as it will return two extreme values thus ignoring the existence of other quantifications that are readily available in fuzzy terms and natural language such as "almost half", "nearly all", "few", "most", etc. Extending this representation language to fuzzy sets, the truth value of the existential relative quantifier and the universal relative quantifier can be defined [1, 4] as:

$$\mathrm{T}_{\exists, A/E} = \mathop{\Delta}_{k=1}^{N} [\mu(e_k) \nabla \mu(a_k)] \tag{3}$$

$$\mathrm{T}_{\forall, A/E} = \mathop{\nabla}_{k=1}^{N} [\mu(e_k) \Rightarrow \mu(a_k)] \tag{4}$$

where $\mu(a_k)$ and $\mu(e_k)$ are the membership functions of fuzzy sets A and E respectively, and $\Rightarrow$ denotes fuzzy implication.

It is obvious that this definition covers as its specific cases classical existential and universal quantifiers. The multi-valued fuzzy quantification can be defined using any available functions such as *non-decreasing*, *non-increasing* or *unimodal* whithin the above definition. Several different quantifiers can be defined between the existential quantifier and the universal quantifier, for example "almost all of them", "almost three-quarter of them", "almost half of them ", "almost a quarter of them" and "a few of them". Several existing methods proposed and discussed in [1, 4] can be used in evaluating the quantified proposition.

The multi-valued quantifiers can be expanded further because the number of quantifiers that can exist may not be limited to only a few specific ones. However, the problems in expanding this kind of quantifier lie in the need to pre-define each of the quantifiers. Limited pre-defined quantifiers are difficult to be adapted to suit fuzzy models which generate rules based on training data. This is because small changes in the dataset might cause the change of the entire ruleset. Thus, a continuous fuzzy quantification method may be more appropriate.

Vila et al. [20] proposed a continuous quantifier which uses linear interpolation between the two extreme cases of the existential quantifier $\exists$ and the universal quantifier $\forall$. In particular, the quantifier was defined as a linear interpolation:

$$Q(E,A)=(1-\lambda_Q).\mathrm{T}_{\forall,A/E}+\lambda_Q.\mathrm{T}_{\exists,A/E} \qquad (5)$$

where Q is the quantifier for fuzzy set A relative to fuzzy set E and λ_Q is the *degree of orness* of the two extreme quantifiers. The truth-values of the existential quantifier $\mathrm{T}_{\exists,A/E}$ and the universal quantifier $\mathrm{T}_{\forall,A/E}$ were defined as:

$$\mathrm{T}_{\exists,A/E}=\mathop{\Delta}_{k=1}^{N}\mu(a_k)\nabla\mu(e_k) \qquad (6)$$

$$\mathrm{T}_{\forall,A/E}=\mathop{\nabla}_{k=1}^{N}(1-\mu(e_k))\Delta\mu(a_k) \qquad (7)$$

where a_k and e_k are the membership functions of fuzzy sets A and E respectively, ∇ represents the *t-norm* and Δ represents the *t-conorm*. This definition will enable the creation of all possible quantifiers that exist between the existential and universal quantifiers.

2.3 Subsethood-based Fuzzy Rule Algorithm (SBA)

The Subsethood-based Fuzzy Rule Algorithm (SBA) represents learned knowledge in fuzzy decision trees for handling classification tasks [3]. This approach involves three main steps: a) classifying training data into subgroups according to the underlying classification results, b) calculating fuzzy subsethood values for every linguistic term, and c) creating rules based on fuzzy subsethood values.

The generation of fuzzy rules is therefore, dependent on the fuzzy subsethood values between the decision to be made and the possible linguistic terms of the conditional attributes. Fuzzy rules are created subject to a pre-specified threshold value $\alpha \in [0, 1]$. Any linguistic terms that have a subsethood value that is greater than or equal to α will automatically be chosen as an antecedent for the resulting fuzzy rules.

These include those terms whose negation is of subsethood value that is greater than or equal to α. If there are training cases that the generated rules do not cover because none of the subsethood values is greater than α, additional rules will be created based on the membership function values of those rules generated earlier, with regard to another preset threshold value $\beta \in [0, 1]$. This is needed to provide full coverage of the generated rule set.

This technique has been tested using the Saturday Morning Problem (SMP) data and was shown to produce better results [3] as compared to the earlier subsethood-based learning algorithm [22]. However, both of these approaches assume that all pieces of information gathered from the training data are equally important. This may not be the case in modelling many real problems [5]. A modified approach that takes a certain weighting strategy to represent the degrees of "importance" is therefore necessary.

3 FuzzyQSBA

FuzzyQSBA is developed on the basis of Weighted Subsethood based Algorithm WSBA [17], which induces fuzzy rules that weight the contributions of conditional attributes to the conclusion (classification). The weighting is in crisp values created from fuzzy subsethood values. The work here is to modify the WSBA to enable the use of fuzzy linguistic quantifiers to replace crisps weights used in the models induced by WSBA. This section will first explain in brief the background of WSBA and then describes improved version of WSBA, namely FuzzyQSBA.

3.1 Weighted Subsethood-based Algorithm (WSBA)

WSBA makes use of subsethood values as relative weights over the significance of different conditional attributes which they may have upon the conclusion, in conjunction with the use of default fuzzy general rules. As with many existing techniques for representing weights, in WSBA, measures of weighting are limited to the range of 0 to 1, with 0 representing the lowest weight (or of least importance) and 1 the highest (or of most importance). Such weights can be calculated from fuzzy subsethood values as follows. Note that the meaning of *subsethood* is herein extended to allow fuzzy sets associated with different linguistic variables to be related.

Suppose that the subsethood value for a certain linguistic term A_j of linguistic variable A with regard to classification X is $S(E, A)$, and that the linguistic variable A has the following possible linguistic terms: $A_1, A_2, \ldots, A_l$. Then, the relative weight for linguistic term A_i, with regard to classification E is:

$$\mathrm{w}(E,A_i)=\frac{S(E,A_i)}{\max\limits_{j=1..l}S(E,A_j)} \qquad (8)$$

Clearly, $w(E, A_i) \in [0,1]$ and $i = 1, 2, \ldots, l$. This allows the creation of a weight for each linguistic term per condition attribute. Intuitively, the linguistic term with the highest subsethood value will be the most important and that with the lowest will be the least important.

By multiplying each possible linguistic value by its respective weight, the proposed WSBA fuzzy rules will be in the form:

Rule 1 **IF** A is $w(E_1,A_1)A_1$ *OR* $w(E_1,A_2)A_2$ *OR* …*OR* $w(E_1,A_i)A_i$ **AND** B is $w(E_1,B_1)B_1$ *OR* $w(E_1,B_2)B_2$ *OR*… *OR* $w(E_1,B_j)B_j$ **AND** … **AND** H is $w(E_1,H_1)H_1$ *OR* $w(E_1,H_2)H_2$ *OR* … *OR* $w(E_1,H_k)H_k$ **THEN** the output is E_1

Rule 2 **IF** A is $w(E_2,A_1)A_1$ *OR* $w(E_2,A_2)A_2$ *OR* …*OR* $w(E_2,A_i)A_i$ **AND** B is $w(E_2,B_1)B_1$ *OR* $w(E_2,B_2)B_2$ *OR*… *OR* $w(E_2,B_j)B_j$ **AND** … **AND** H is $w(E_2,H_1)H_1$ *OR* $w(E_2,H_2)H_2$ *OR* … *OR* $w(E_2,H_k)H_k$ **THEN** the output is E_2

.

.

.

Rule n **IF** A is $w(E_n,A_1)A_1$ *OR* $w(E_n,A_2)A_2$ *OR* …*OR* $w(E_n,A_i)A_i$ **AND** B is $w(E_n,B_1)B_1$ *OR* $w(E_n,B_2)B_2$ *OR*… *OR* $w(E_n,B_j)B_j$ **AND** … **AND** H is $w(E_n,H_1)H_1$ *OR* $w(E_n,H_2)H_2$ *OR* … *OR* $w(E_n,H_k)H_k$ **THEN** the output is E_n

(9)

In above definition, "OR" is interpreted by the t-conorm operator and "AND" by the t-norm operator.

Initially, all linguistic terms of each attribute are used to describe the antecedent of each rule. This may look tedious, but the reason for keeping this complete form is that every linguistic term may contain important information that should be taken into account. Otherwise, there is no need for adopting the given fuzzy partitions of the underlying domains in the first place.

Computationally, the ruleset can be simply represented by

$$Y_k = \mathop{\Delta}_{i=1..m} (\mathop{\nabla}_{j=1..n} (w_{A_{ij},E_k} \times \mu_{A_{ij}}(x)), k = 1,2,...,n. \quad (10)$$

where w_{A_{ij},E_k} denote the weights of atomic linguistic propositions and $\mu_{A_{ij}}(x)$ represent the membership function of the linguistic terms modified by the weights, with Δ and ∇ denoting the interpretation of logical conjunction and disjunction operators respectively.

This method does not require any threshold value and generates a fixed number of rules, with the ruleset cardinality equalling to the number of classes of interest (i.e. one rule will be created for each class). In the process of generating fuzzy rules, linguistic terms that have a weight greater than zero will automatically be promoted to become part of the antecedents of the resulting fuzzy rules. Any linguistic term that has a weight equal to 0 will of course be removed from the fuzzy rule. This will make the rules simpler than the original default rules (9). In running WSBA for classification tasks, the concluding classification will be that of the rule whose overall weight is the highest amongst all.

3.2 Modifying WSBA with Fuzzy Quantifiers

The aim of this proposed technique is to replace crisp weights in WSBA by fuzzy quantifiers. For this, the quantification method originally proposed by Vila et al. is employed here. Several reasons have been taken into account to support the use of Vila et al.'s approach:

a) The use of *degree of orn*ess enables the implementation of continuous quantifiers. Thus, any possible quantifier can be created in principle.
b) The relative quantifier based method proposed by Villa et al. can be adapted into WSBA easily, thanks to the structure of the WSBA general rule. Thus, the simplicity of WSBA can be preserved.
c) Relative subsethood values can be used as the *degree of* orness *(λ_Q)* of the fuzzy quantifiers. Thus, the two seemingly separate approaches are unified.
d) This approach fulfils the desirable monotonicity and duality properties of quantification [1].

Computationally, the induced ruleset can be represented by

$$Y_k = \mathop{\Delta}_{i=1..m} (\mathop{\nabla}_{j=1..n} (Q_{A_{ij},E_k} \nabla \mu_{A_{ij}}(x)), k = 1,2,...,n. \quad (11)$$

where Q_{A_{ij},E_k} are now fuzzy quantifiers as defined in (5).

The crisp weights that were used in WSBA are herein replaced by fuzzy quantifiers. The main difference of FuzzyQSBA compared to WSBA lies in the interpretation of the inference between weights/quantifiers with the linguistic terms. In WSBA the weights for each linguistic term are crisp values and behave as multiplication factor for the linguistic terms. Clearly the use of crisp value will limit the interpretation of $(w_{A_{ij},E_k} \times \mu_{A_{ij}}(x))$. In FuzzyQSBA both the quantifiers and the linguistic terms are fuzzy sets. This offers flexibility as it enables the use of *t-norm* operators to interpret $(Q_{A_{ij},E_k} \nabla \mu_{A_{ij}}(x))$ whilst guarantee that the inference results are fuzzy sets.

The use of fuzzy quantifier in QSBA also enables representation of the ruleset in a more natural way. This can be shown by the following example:

In WSBA, the ruleset is in the form of "IF SL is (SL1 OR 0.09SL2) AND SW is (SW2 OR 0.2SW3) AND PW is (PL1) AND PW is (PW1) THEN the class is Iris-setosa".

In FuzzyQSBA, the ruleset will be in the form of "IF SL is ((*almost all*)SL1 OR (*a little*)SL2) AND SW is ((*almost all*)SW2 OR (*almost a quarter of*)SW3) AND PW is ((*almost all*)PL1) AND PW is ((*almost all*)PW1) THEN the class is Iris-setosa.

Clearly, the use of fuzzy quantifiers make the model more readable, although the computation still needs to be done using real numbers.

Based on the definitions of the fuzzy subsethood value (1), the existential quantifier (6) and the universal quantifier (7), it can be shown that if λ_Q is equal to 0 then the truth-value of quantifier Q will also equal 0. Thus any linguistic terms which have the truth-value of the quantifier equal to 0 will be removed automatically from the fuzzy rule antecedents. The final FuzzyQSBA ruleset will contain the same antecedents as the WSBA ruleset. Figure 1 shows the framework of this approach. It is clear that the main structure of WSBA general rules is preserved [17].

4 Experimental Results

To demonstrate the advantages of the proposed method (FuzzyQSBA), five datasets which have different features are chosen from the UCI machine learning repository [19]. The datasets, which are widely used as benchmarks for classification tasks are the Iris-Plant, Wine Recognition, Wisconsin Breast Cancer, Spambase and Mushroom datasets as summarised in Table 1.

Dataset	Number of Instances	Number of Conditional Attributes	Number of classes
Iris-Plant	150	4	3
Wine Recognition	178	13	3
Wisconsin Breast Cancer	699	9	2
Spambase	4601	58	2
Mushroom	8124	22	2

Table 1. Classification Problem Datasets.

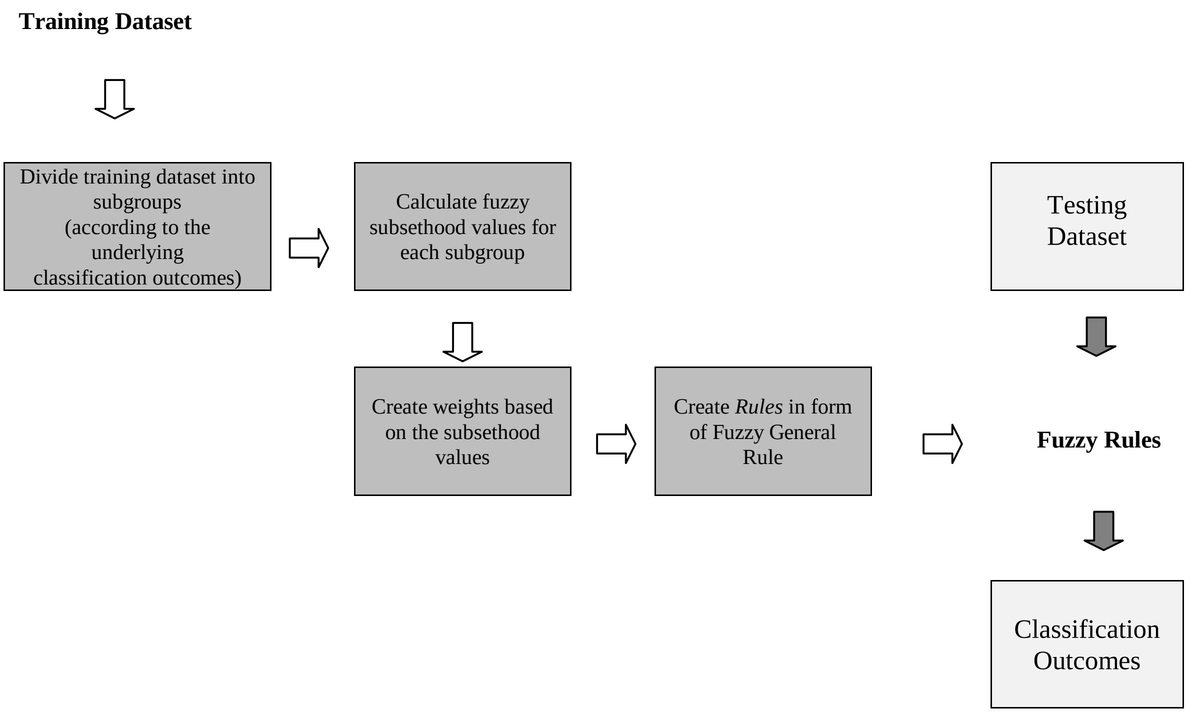

Figure 1. Framework of FuzzyQSBA.

Ten-fold cross-validation is used to evaluate the classification accuracy: Each dataset was divided randomly into ten subsets, with nine subsets used for training and the remaining one for testing. Thus, ten sub-experiments have been carried out for each dataset. Table 2 shows the comparison of the classification accuracy of the models induced by WSBA and FuzzyQSBA, and Figures 2 - 5 show such comparison between SBA and FuzzyQSBA. All the comparisons are made in terms of minimum, maximum and average classification accuracy.

4.1 Comparison between WSBA and FuzzyQSBA

The experimental results on five different datasets show that the classification accuracy for models induced by FuzzyQSBA are as good as that of the models learned by WSBA. It clearly demonstrates that the differences between average classification accuracies produced by the two algorithms are very small. However, as discussed previously, FuzzyQSBA has an advantage over WSBA in terms of the transparency of the associated inference process and of the readability of the induced ruleset.

Dataset	WSBA			FuzzyQSBA		
	Min	Ave	Max	Min	Ave	Max
Iris Plant	86.7	96.0	100	86.7	96.0	100
Wine Recog.	83.3	97.2	100	77.8	96.7	100
Breast Cancer	88.4	92.8	97.1	88.4	92.2	97.1
Mush.	87.7	89.3	91.1	87.7	89.4	91.1
Spam.	85.7	86.7	88.9	85.7	86.9	88.9

Table 2. Comparison of Classification Accuracy between WSBA and FuzzyQSBA.

4.2 Comparison between SBA and FuzzyQSBA

On Iris-Plant dataset, which involves a small number of instances and conditional attributes, results show that the average classification accuracy of FuzzyQSBA is better than the average classification accuracy of SBA, although both FuzzyQSBA and SBA can reach the maximum classification accuracy of 100%. However, for the Wine Recognition dataset, which also has a small number of instances and conditional attributes, results show that FuzzyQSBA has outperformed SBA in terms of maximum, average and minimum classification accuracy, consistently. For the experiment on the Breast Cancer dataset, which has a medium number of instances but still involves small number of conditional attributes, results show that SBA manage to obtain a higher classification accuracy over certain threshold values.

Experiments on Spambase and Mushroom datasets, both of which involve a high number of instances and of conditional attributes, show that FuzzyQSBA has again outperformed SBA in all of the classification results. This clearly demonstrate that the FuzzyQSBA is capable of inducing accurate rule models from given training data.

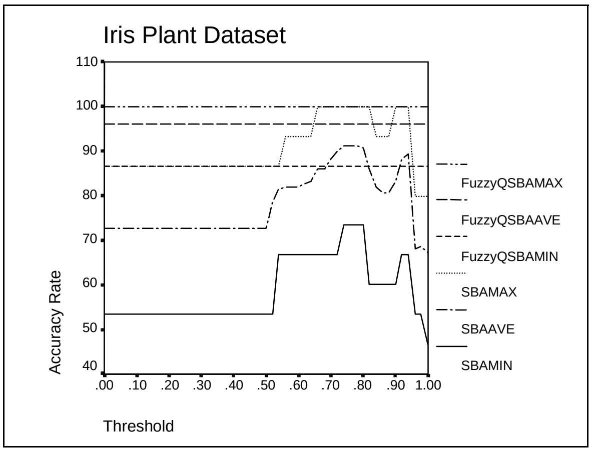

Figure 2. Classification Accuracy of Iris-Plant Dataset.

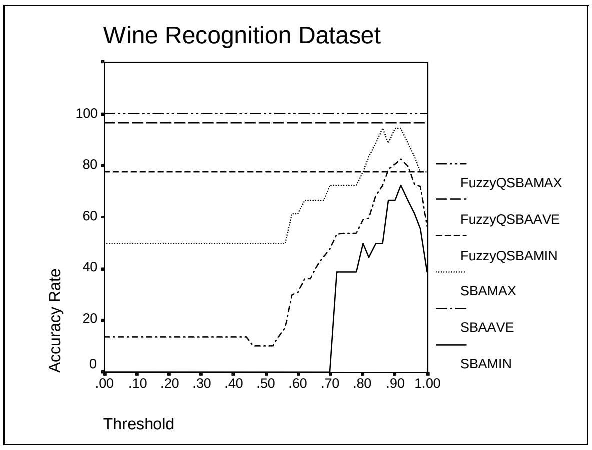

Figure 3. Classification Accuracy of Wine Recognition Dataset.

In particular, it is worth noting that classification accuracy of FuzzyQSBA is obtained without the use of any threshold value, whereas for SBA, certain threshold values are needed to be specified to induce models that have a high classification accuracy. Furthermore, the use of threshold values is very confusing as different training datasets may need different threshold values to obtain a good classification performance. Of course, all such advantages of FuzzyQSBA are achieved on top of its ability in inducing more readily interpretable models.

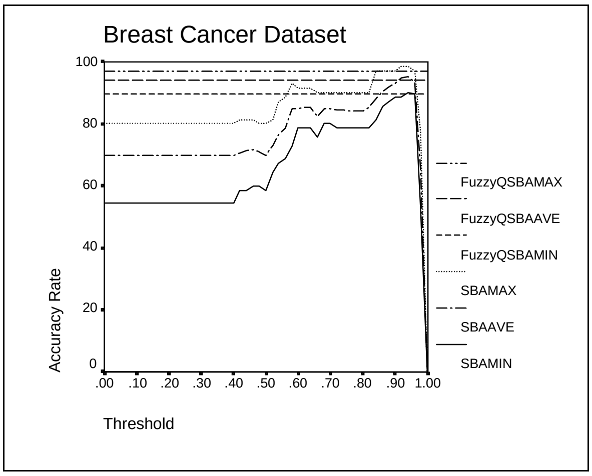

Figure 4. Classification Accuracy of Breast Cancer Dataset.

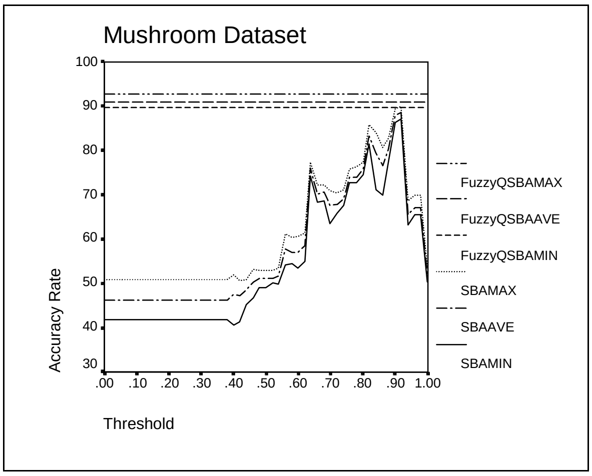

Figure 5. Classification Accuracy of Spambase Dataset.

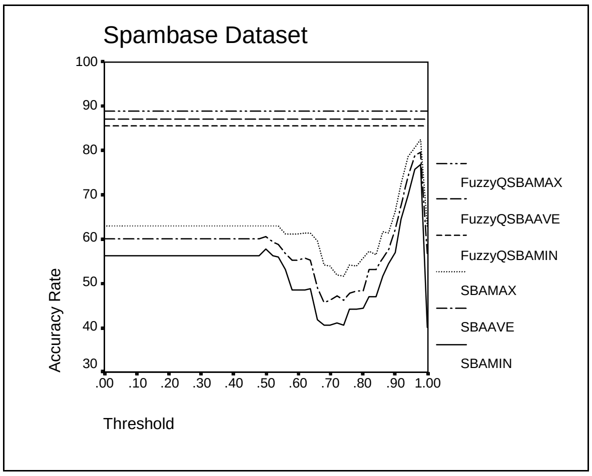

Figure 6. Classification Accuracy of Mushroom Dataset.

5 Conclusion

This paper has presented a new method for generating linguistic rule models from data based on fuzzy subsethood measurements. The work has been applied for classification tasks, offering a number of advantages over existing subsethood-aided approaches. It does not need any threshold values while creating a fixed number of rules with respect to the potential classification outcomes. In addition, the use of fuzzy quantifiers makes the induced models more interpretable and hence the associated inference processes more transparent.

The proposed method also has potentials to be developed further. In particular, it can be expected to perform better generalisation if the fuzzy membership functions employed are optimised. Also, it would be interesting to investigate how the algorithm would perform in coping with scaled-up real-world problems. For this, work is being carried out, which applies FuzzyQSBA for evaluation of student academic performance. It is obvious that in this problem domain, transparency and interpretability are very important as the users are mainly the students, teachers and policy planners who might not have any background in fuzzy rule-based systems at all.

Acknowledgements

This research is partially supported by the MARA University of Technology, Malaysia and The Public Service Department, Government of Malaysia. The authors are grateful to all members of the Approximate and Qualitative Reasoning Group within the School of Informatics, The University of Edinburgh, for their comments and suggestions in producing this paper.

References

[1] Barro, S., Bugarin, A.J., Carinena, P. and Diaz-Hermida, F. (2003). A Framework for Fuzzy Quantification Models Analysis, *IEEE Transactions on Fuzzy Systems*, 11(1)1: 89-99.

[2] Castro J.L. and Zurita J.M. (1997). An Inductive Learning Algorithm in Fuzzy Systems, *Fuzzy Sets And Systems* 89, 193-203.

[3] Chen, S.M., Lee, S.H. and Lee, C.H. (2001). A New Method for Generating Fuzzy Rules from Numerical Data for Handling Classification Problems, *Applied Artificial Intelligence*, 15: 645 - 664.

[4] Delgado, M., Sanchez, D. and Vila, M. A. (2000). Fuzzy Cardinality based Evaluation of Quantified Sentences, *International Journal of Approximate Reasoning*, 23: 23-66

[5] Herera, F. and Herera-Viedma, E. (1995). Aggregation Operators for Weighted Linguistic Weighted Information, Unpublished Technical Report, Department of Computer Science and Artificial Intelligence, Universidad de Granada.

[6] Hong, T.P. and Lee, C.Y. (1996). Induction of Fuzzy Rules and Membership Functions from Training Examples. *Fuzzy Sets and Systems* 84(1), 33-47.

[7] Janikow, C.Z. (1998). Fuzzy Decision Trees: Issues and Methods. *IEEE Transactions on Systems , Man and Cybernatics* 28 (1), 1-14.

[8] Liu, Y. and Kerre, E.E. (1998). An Overview of Fuzzy Quantifiers. (i) Interpretations, *Fuzzy Sets and Systems*, 95 : 1-21.

[9] Liu, Y. and Kerre, E.E. (1998). An Overview of Fuzzy Quantifiers. (ii) Reasoning and Applications, *Fuzzy Sets and Systems*, 95 : 135-146.

[10] Lozowski, A., Cholewo, T.J. and Zurada, J.M. (1996). Crisp Rule Extraction from Perceptron Network Classifiers. *Proceedings of International Conference on Neural Network, Volume of Plenary, Panel and Special Sessions*, Washington DC, 94 - 99.

[11] Luo, X. and Zhang, C. (2002). The Weighting Issue in Fuzzy Logic, *Informatica: An International Journal of Computing and Informatics,* 21 : 255 - 262.

[12] Marin-Blazquez, J.G. and Shen, Q. (2002). From Approximate to Descriptive Fuzzy Classifiers, *IEEE Transactions on Fuzzy Systems*, 10 (4) : 484 -509.

[13] Nauck, D. (2000). Adaptive Rule Weights in Neuro-Fuzzy Systems, *Neural Computing and Applications*, 9 : 60-70.

[14] Nozaki, K., Ishibuchi, H. and Tanaka, H. (1997). A Simple but Powerful Heuristic Method for Generating Fuzzy Rules From Numerical Data. *Fuzzy Sets and Systems* 86(3), 251-270.

[15] Piskounov, A. (2003). Fuzzy Rule-Based Models: Decision-Making vs Control, *Proceedings of the IEEE International Conference on Fuzzy Systems,* St Louis Missouri: 96-101.

[16] Rasmani, K.A. and Shen, Q. (2003). Weighted Linguistic Modelling Based on Fuzzy Subsethood Values, *Proceedings of the IEEE International Conference on Fuzzy Systems,* St Louis Missouri: 714-719.

[17] Rasmani, K.A. and Shen, Q. (2004). Modifying Weighted Fuzzy Subsethood-based Rule Models with Fuzzy Quantifiers, To appear in *Proceedings of the IEEE International Conference on Fuzzy Systems,* Budapest.

[18] Shen, Q. and Chouchoulas, A. (2002). A Fuzzy-Rough Approach for Generating Classification Rules, *Pattern Recognition*, 35(11), 341-354.

[19] UCI Machine Learning Databases. Available online: http:// www.ics.uci.edu/~mlearn/MLRepository.html

[20] Vila. M.A., Cubero, J.C., Medina, J.M. and Pons, O. (1997). Using OWA Operators in Flexible Query Processing, *The Ordered Weighted Averaging Operators: Theory, Methodology and Applications*, London : Kluwer Academic: 258-274.

[21] Wang, X.Z., Wang, Y.D., Xu, X.F., Ling, W.D. and Yeung, D.S. (2001). A New Approach to Fuzzy Rule Generation: Fuzzy Extension Matrix. *Fuzzy Sets and Systems* 123, 291-306.

[22] Yuan, Y. and Shaw, M.J. (1995). Induction of Fuzzy Decision Trees, *Fuzzy Set and Systems*, 69(2) : 125-139.