Validation Data Accuracy as an Additional Objective in Multiobjective Fuzzy Genetics-based Machine Learning

Samir Antonio Fersobe Dilone[1], Naoki Masuyama[1], Yusuke Nojima[1], Hisao Ishibuchi[2]

[1]Department of Computer Science and Intelligent Systems
Osaka Prefecture University
Sakai, Osaka, Japan
[2]Department of Computer Science and Engineering
Southern University of Science and Technology
Shenzhen, China
{samir.fersobe@ci., masuyama@, nojima@}cs.osakafu-u.ac.jp,
hisao@sustech.edu.cn

Abstract. This paper examines the effects of using validation data accuracy as an additional objective in multiobjective fuzzy genetics-based machine learning (MoFGBML). The MoFGBML algorithm uses an evolutionary multiobjective optimization algorithm (EMOA) to maximize the training data accuracy and minimize the classifier complexity in the design of fuzzy rule-based classifiers. During the optimization process, the classifiers generated through MoFGBML may become overfitted to the training data. We show that using validation data accuracy as an additional objective in MoFGBML provides much more non-dominated classifiers while increasing the generalization ability of the obtained classifiers.

Keywords: Evolutionary Multiobjective Optimization, Fuzzy Classifier, Pattern Classification

1 Introduction

In finance and healthcare industries where trust is of utmost importance, understanding how results are obtained is fundamental to their success [1]. Current black-box models do not have the necessary degree of interpretability required for these types of tasks. Furthermore, this lack of understanding prevents us from tuning the models to our preferences. One solution to this problem is to use fuzzy rule-based classifiers. These classifiers use several linguistic fuzzy if-then rules to classify the input patterns. They are easy to interpret and can clearly describe how the model classifies each pattern [2]. Since fuzzy rules in a classifier represent classification boundaries, a large number of rules are often required to obtain a high degree of accuracy. Correspondingly, a fuzzy classifier with a small number of rules tends to have a low degree of accuracy whereas it is easy to interpret. This is known as the interpretability-accuracy trade-off [3].

To deal with the interpretability-accuracy trade-off, a multiobjective fuzzy genetics-based machine learning (MoFGBML) algorithm was proposed [3]. In MoFGBML, fuzzy classifiers are optimized by formulating a multiobjective optimization problem. By using an evolutionary multiobjective optimization algorithm (EMOA), MoFGBML can generate a wide variety of non-dominated fuzzy classifiers along the interpretability-accuracy trade-off surface.

Due to the nature of machine learning, during the search for more accurate classifiers with the training data, a certain degree of overfitting may occur. For the highly accurate classifiers generated through MoFGBML, this bias may cause the expected error rate for the test data to be significantly worse. In this paper, we improve the generalization ability of MoFGBML by using validation data accuracy as an additional objective. In computational experiments, we show that the classifiers generated through this new formulation are more accurate and have higher generalization ability.

This paper is organized as follows: Section 2 briefly explains fuzzy classifiers and the original problem formulation of MoFGBML. Section 3 explains our proposed formulation which uses validation data accuracy as an additional objective in MoFGBML. In Section 4, we explain computational experiments and compare the proposed formulation with the original formulation. Lastly, Section 5 has some concluding remarks as well as future research directions to conduct.

2 Multi-objective Fuzzy Genetics-based Machine Learning

2.1 Fuzzy Rules

We can generate fuzzy if-then rules which are true to a certain degree for a certain number of patterns. These fuzzy if-then rules are represented as follows:

$$\text{Rule } R : \text{If } x_1 \text{ is } A_1 \text{ and ... and } x_n \text{ is } A_n \text{ then Class } C \text{ with } CF, \tag{1}$$

where $\mathbf{x} = (x_1,..., x_n)$ is a pattern composed of n attributes, $\mathbf{A}$ is the antecedent fuzzy set $\mathbf{A} = (A_1,..., A_n)$, C is the corresponding consequent class, and CF is the corresponding rule weight. By combining different rules into a rule set S, we can build a classifier with high classification accuracy and high interpretability.

2.2 Fuzzy Classifiers

There are two main approaches in genetics-based machine learning (GBML): Michigan and Pittsburgh approaches. The Michigan approach uses each rule as an individual and the set of rules as a population. The Pittsburgh approach uses a rule set as an individual and a number of rule sets as a population. Considering the benefits of both approaches, MoFGBML implements a hybrid approach.

In MoFGBML, we follow the Pittsburgh approach and use fuzzy rule sets as individuals which are evolved through genetic operations to generate better rule sets. We also randomly use Michigan-style genetic operations as local search with a prespecified probability (0.5 in this paper). Our goal is to generate as many non-dominated classifiers as possible. A classifier is considered non-dominated when none of its objective functions can be improved without degrading the others. Due to the interpretability-accuracy trade-off, it is impossible to obtain a classifier which is both easy to interpret and highly accurate [3]. For this reason, we formulate the design of fuzzy classifiers as a multi-objective optimization problem (MOP). Consequently, we use an EMOA to generate fuzzy classifiers. In this study, we apply NSGA-II [4] which achieves both diversity and performance through the evolution process.

2.3 Original Problem Formulation: MOP1

The original multiobjective optimization problem (**MOP1**) uses two objective functions, which take in a rule set S as an input. The first objective function $f_1(S)$ is the error rate against the training data while the second objective function $f_2(S)$ is the number of rules in the classifier. Therefore, we can define **MOP1** as: minimize the error rate for the training data $f_1(S)$, and minimize the number of rules in the classifier $f_2(S)$.

3 Use of Validation Data Accuracy

3.1 New Problem Formulation: MOP2

In general, validation data is used in some machine learning algorithms to monitor the learning process. It is generally used to prevent the classifier from overfitting to the training data. Our implementation introduces validation data by dividing the training data into validation data and subtraining data. We then execute our algorithm with the subtraining data where the validation data accuracy is used as an additional objective in the EMOA (i.e., in NSGA-II).

Using validation data and subtraining data, we obtain two new objective functions which can be used to formulate a new three-objective optimization problem (**MOP2**). These are $f_3(S)$ which is the error rate against the subtraining data and $f_4(S)$ which is the error rate against the validation data. We can define **MOP2** as: minimize the number of rules in the classifier ($f_2(S)$), minimize the error rate for the subtraining data ($f_3(S)$), and minimize the error rate for the validation data ($f_4(S)$).

3.2 Algorithm

We first perform cross-validation which divides the data into training data and test data. After data partitioning is completed our algorithm executes as follows:

Step 1: Divide training data into subtraining data and validation data according to a predetermined validation rate r.

Step 2: Generate an initial population of N rule sets where N is the population size.

Step 3: Generate an offspring population by iterating the following procedure N times.

1. Select a pair of parent rule sets from the current population using binary tournament selection.
2. Randomly select a Pittsburgh-style or Michigan-style approach to generate the offspring.
    (a) If Pittsburgh-style is selected, generate an offspring from the selected pair of parent rule sets by using crossover and mutation operations
    (b) If Michigan-style is selected, apply a single iteration of Michigan-style GBML to one of the parents.

Step 4: Evaluate the offspring population.

Step 5: Merge the offspring population with the current population.

Step 6: Perform the non-dominated sorting and crowding distance calculation in the merged population.

Step 7: Select the best N individuals from the merged population based on the rank and crowding distance to generate the next population.

Step 8: If the stopping condition is satisfied, stop the algorithm and obtain the non-dominated rule sets as the final solutions. Otherwise return to Step 2.

Before dividing the training data, we specify a validation rate r which is the ratio of training data allocated for validation data. In addition, we make sure that the ratio of classes in the subtraining data is proportional to that of classes in the validation data.

4 Computational Experiments

4.1 Experiment Settings

We perform computational experiments with **MOP2** as well as **MOP1** to compare and assess the generalization ability of the classifiers obtained through MoFGBML. We attempt these experiments with seven different validation rates. The experiment settings are as follows.

- Number of Trials: 30 (10-fold cross-validation $\times$ 3)
- Stopping Condition: 10,000 generations
- Population Size: 60
- EMOA: NSGA-II
- Validation Rates: 0.1, 0.2, 0.3, 0.5, 0.7, 0.8, 0.9

We use 20 datasets obtained from the UCI machine learning repository. The datasets are shown in Table 1.

4.2 Experiment Results

Because of the page limitation we only show the results for the 0.5 validation rate. Table 1 shows that the average error rate against the test data from the best classifiers with respect to the training data for both **MOP1** and **MOP2** as well as the average number of rules in the classifiers. Since we ran 10-fold cross-validation three times, we obtained the best classifier with respect to the training data for each trial and averaged the results over 30 trials. We can clearly see that **MOP2** obtains classifiers with a lower error rate for most datasets. On the other hand, **MOP1** has classifiers which have a smaller number of rules than **MOP2**. To verify the statistical significance of the obtained results, we perform the Wilcoxon signed rank test [5] with the significance level 0.05 to test the null hypothesis that the test data error rate between both formulations are statistically equal. In this experiment, the obtained results are treated as statistically significant since the p value of the statistical test shows 0.00554.

Furthermore, we analyze the results by collecting all the classifiers for each specific number of rules and plotting their mean. We plot the number of rules which have more than 15 classifiers considered in their mean. Figure 1 shows the results for the Spambase

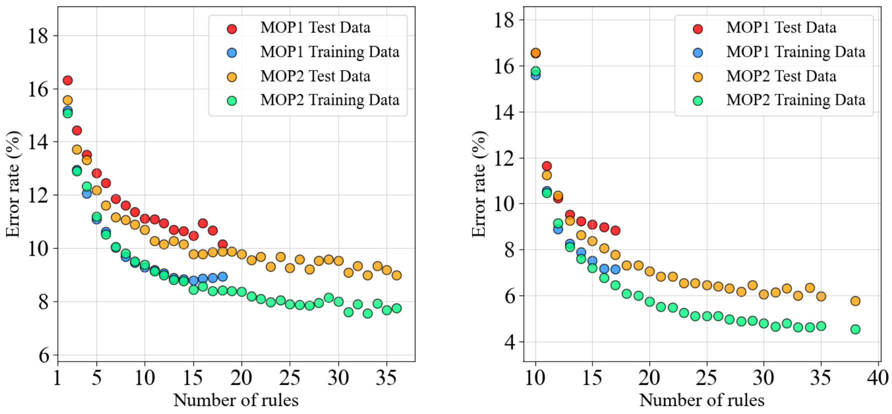
Table 1. Dataset information and experiment results

Datasets	Statistics of dataset			Error rate for test data (%)		Number of rules	
	Pattern	Attribute	Class	MOP1	MOP2	MOP1	MOP2
Iris	150	4	3	5.78	**5.56**	4.43	**4.37**
Wine	178	13	3	5.83	**5.81**	3.63	**3.43**
Sonar	208	60	2	23.40	**21.04**	**10.33**	10.80
Newthyroid	215	5	3	7.27	**5.71**	5.73	**5.67**
Australian	690	14	2	**13.48**	14.25	**11.13**	19.30
Pima	768	8	2	24.13	**23.65**	**16.30**	31.20
Vehicle	846	18	4	29.24	**29.00**	**17.50**	36.00
Yeast	1,484	8	10	41.78	**40.79**	**22.20**	36.50
Segment	2,310	19	7	7.59	**6.38**	**15.47**	22.57
Spambase	4,597	57	2	10.44	**9.00**	**19.40**	33.43
Phoneme	5,404	5	2	15.50	**15.19**	**23.57**	41.23
Page-blocks	5,472	10	5	3.65	**3.46**	**13.33**	16.27
Texture	5,500	40	11	8.70	**6.14**	**17.87**	35.43
Satimage	6,435	36	6	15.19	**15.17**	**19.97**	40.20
Twonorm	7,400	20	2	**3.61**	4.16	**18.87**	39.30
Ring	7,400	20	2	**4.31**	4.72	**24.47**	44.70
Penbased	10,992	16	10	4.65	**4.55**	**42.30**	49.07
Magic	19,020	10	2	15.27	**15.05**	**19.90**	40.93
Letter	20,000	16	26	41.97	**40.43**	**37.97**	43.00
Shuttle	57,999	9	7	**0.47**	0.49	**9.33**	9.87

dataset and the Texture dataset. Figure 1 shows that **MOP2** generates more classifiers than **MOP1**. One of the reasons for this could be that **MOP2** is a three-objective problem which allows us to generate more non-dominated classifiers than **MOP1**. We observe that **MOP2** performs better than **MOP1** for most number of rules. Focusing on the classifiers with 5-15 rules of the Spambase dataset in Figure 1 (a), **MOP2** shows a lower test data error rate than that of **MOP1**. Moreover, it also shows that the training data error rate for **MOP2** to be similar to **MOP1**. Since the difference in the error rates between the training data and the test data is decreased for **MOP2**, the generalization ability of the classifiers is improved. We can notice in Figure 1 (b), for the Texture dataset, both the training data error and test data error rates for **MOP2** outperform those of **MOP1**. This shows not only an improvement in generalization ability but in classification accuracy as well.

5 Conclusion

This paper proposed the use of validation data accuracy as an additional objective in MoFGBML to increase the generalization ability of the obtained classifiers. We demonstrated that this proposal leads to the generation of more accurate classifiers and increases the number of non-dominated classifiers in the population. This may come at the cost of an increased number of rules for the most accurate classifiers in **MOP2**. Nevertheless, the classifiers with the same number of rules obtained through **MOP2**



(a) Spambase, $r = 0.5$

(b) Texture, $r = 0.5$

Fig. 1. 30-trial average of classifiers for each number of rules

are more accurate and less overfitted than the classifiers obtained through **MOP1**. This indicates that the new formulation improves the generalization ability of the classifiers generated through MoFGBML. As a future research topic, we will further investigate the effects of different validation rates to understand how it affects MoFGBML.

Acknowledgement

This research was partially supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant JP19K12159.

References

1. A. Adadi and M. Berrada, “Peeking inside the Black-Box: A survey on explainable artificial intelligence (XAI),” *IEEE Access*, vol. 6, pp. 52138-52160, September 2018.
2. H. Ishibuchi, T. Nakashima, and M. Nii , *Classification and Modeling with Linguistic Information Granules*, Springer-Verlag Berlin, Heidelberg, 2004.
3. H. Ishibuchi and Y. Nojima, “Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning,” *International Journal of Approximate Reasoning*, vol. 44, no. 1, pp. 4-31, January 2007.
4. K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, April 2002.
5. F. Wilcoxon, “Individual comparisons by ranking methods,” *Biometrics Bulletin*, vol. 1 , no. 6, pp. 80-83, December 1945.