Proceedings of the World Congress on Engineering and Computer Science 2010 Vol I
WCECS 2010, October 20-22, 2010, San Francisco, USA

Fuzzy Modeling with Genetically Optimized Feature Space Reduction

Mingli Song, Witold Pedrycz, *IEEE, Fellow*

Abstract— **This paper contributes to the ongoing studies on Genetic Algorithms applied to problems of feature selection in fuzzy modeling. The optimization scheme of Genetic Algorithms to reduce the dimensionality of input space is legitimate as the problem itself is of combinatorial nature. Fuzzy clustering realized through Fuzzy C-Means (FCM) is carried out in the reduced input space and the information granules obtained therein are used to form a series of local models of the rule-based fuzzy model. Our ultimate objective is to form a way of an efficient reduction of the input space leading to the enhanced interpretability of the fuzzy models and investigate possibilities of optimization of fuzzy models with respect. The experimental studies highlighting the numerical aspects of the design comprise synthetic data and data sets publicly available at several data sites.**

Index Terms— **interpretability, fuzzy models, feature reduction, genetic algorithm**

I. Introduction

Feature Reduction is indispensable in system modeling which otherwise being carried out for high-dimensional data make the development of the models highly inefficient and may lead to models that are of lower quality in particular when it comes to their generalization capabilities. For instance, in fuzzy rule-based models, the number of features determines the size of the condition part: the larger the number of features, the poorer readability and interpretability of the rules. In this scenario, choosing discriminative features become a key stage in constructing models. Feature selection can make the modeling more efficient and at the same time keep similar model accuracy. This is critical when faced with massive data.

The solution to the feature selection problems is not unique. According to the criteria used to assess the quality of the resulting (reduced) feature space, there are two general categories of methods, namely filters and wrappers. Using filters we consider some criterion that pertains to the statistical characteristics of the selected attributes and evaluate them with this respect. In contrast, when dealing with wrappers, we are concerned with the effectiveness of the features as a vehicle to carry out classification so in essence there is a mechanism which effectively evaluates the performance of the selected features with respect to their discriminatory capabilities.

Admittedly, optimality of a feature subset can only be guaranteed by exhaustive evaluation. However, this is only feasible as long as the number of inputs is small. Alternatively, evolutionary methods can be used to search for a better performance subset of features in the feature space. GAs [1] are one of the effective vehicle and they have been successfully applied to optimize parameters in both the antecedent and consequent part of fuzzy rules. Also, GAs have been combined with other techniques like fuzzy clustering [2], [3], neural networks [4], [5], [6-7], Kalman filters [8], and others. Another population based search method is Particle Swarm Optimization (PSO) [9-11] in which individuals representing possible solutions carry out a collective search by exchanging their individual findings while taking into consideration their own local experience and evaluating their own performance through comparing a performance index value.

The architecture of the fuzzy model dwells upon a collection of information granules, which are formed in the input space. These information granules form a nonlinear transformation of the input space into the c-dimensional unit hypercube where any input **x** results in a vector of activation levels – membership grades of information granules A_1, A_2, …, A_c. The choice of these fuzzy sets forms a suitable cognitive perspective at which the model is being formed. With the information granules we associate a local linear model $f_i(\mathbf{x}, \mathbf{a}_i) = \mathbf{a}_i^T\mathbf{x}$ with $\mathbf{a}_i$ being the vector of the parameters of the i-th local model. The aggregation of these models to form an overall input –output relationship is governed by the well-known relationship.

This paper discusses the use of GAs as a vehicle of searching a subset of features, which contains essential discriminatory information for fuzzy modeling. Fuzzy C-Means is adopted to form a set of information granules in the reduced space and then the associated local linear models are constructed. We envision a general flow of design as portrayed in Figure 1. In contrast to the standard development of fuzzy models, the essential phase present here deals with dimensionality reduction of input (feature) space.

Manuscript received July 14, 2010.

Mingli Song is with the Department of Electrical and Computer Engineering of University of Alberta, Canada. (corresponding author to provide phone: e-mail: sml607@ yahoo.com.cn).

Witold Pedrycz is with the Department of Electrical and Computer Engineering of University of Alberta, Canada (e-mail: pedrycz@ee. ualberta.ca).

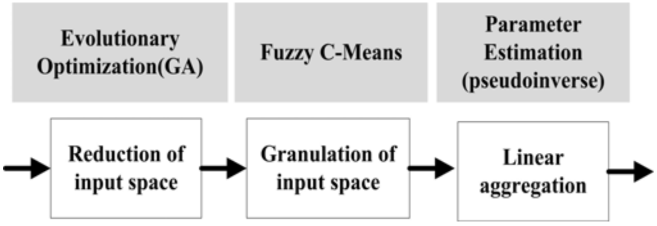

Figure 1. An overall flow of design of the fuzzy model and its association with its underlying architecture and ensuing mechanisms of unsupervised and supervised learning

The structure of this paper is as follows. Next section discusses in detail of the application of genetic algorithms to feature selection problems in fuzzy modeling. In section three, a brief description of fuzzy modeling employed here is provided. Next, in section four, our approach is tested on synthetic and real world data sets.

II. GENETIC ALGORITHMS FOR FEATURE SELECTION

In order to deal with feature searching space's explosion problem when the number of features is large, some feature dimensionality reduction approaches have been proposed. For many practical problems, the original data sets are obtained without professional analysis and this makes the noisy features' existence. Our experiments show that only subset features may have the same or similar performance (accuracy) with using all the features on regression problems.

In this paper, we do the data preprocessing (feature selection) with evolutionary algorithms (EA) and genetic algorithms (GAs), in particular. EA has already been used for structural and parameter optimization in fuzzy modeling. Real-code GA is adopted as the technique to select essential features. In order to decide the number of subset features, we have done experiments from one feature to all features on some benchmark data sets. All the results show the same tendency that there exits a critical drop on output error and after which the error just fluctuates in a very small range. In this way, we can say that this point or its next point (or we can give a criterion to decide the point) is the number we would like. At the same time we also know which features are selected. At this point, the preprocessing has been completed. The subset features can be used as the input in the next step in fuzzy modeling.

Given a certain number of subset of features, the selection procedure is as follows.

1. Randomly generate n population (chromosome) in which each element is a real number between 0 and 1.
2. For each chromosome, there is a corresponding subset of features. Build a fuzzy model by using each subset of features and evaluate the performance of this model with a certain criterion.
3. Keep the subset with the best performance of the associated model and put it into next generation (by replacing the first chromosome in the population after crossover and mutation). Execute crossover and mutation given the crossover rate and mutation rate on all chromosomes.
4. Go to step 2 until meeting stop condition.

Finally we have a subset of features with its performance index under a certain number of features. Do this from one feature space to all feature space. Use a criterion to select a number of subset features, with which we realize fuzzy modeling.

III. RULE-BASED FUZZY MODELS

Rule-based models play a vital role in fuzzy modeling. A multi-input one-output fuzzy rule-based system, in which the experimental data set is given as ($\mathbf{x}_k$, y_k), k = 1, 2, …, N, is a system whose rule base is made up of a set of fuzzy rules of the form [82]

R_i: If X_1 is A_i, X_2 is B_i, …, then y is f_i i = 1, 2, …, c

where X_1, X_2 … are linguistic variables whose values are information granules (here we use fuzzy sets as information granules) A_i, B_i, …, f_i, defined in the corresponding input and output spaces. The local linear models $f(\mathbf{x}, \mathbf{a}_i)$ are constructed in a straightforward way as the optimization problem can be handled analytically. With the squared error treated as the underlying performance index, the optimal coefficients of the models are derived in a standard fashion. Let us rewrite the model in an explicit way by using the activation levels of the individual rules. For the i-th local model we have

$$f_i(\mathbf{x},\mathbf{a}_i) = a_0 + a_1^i x_1 + a_2^i x_2 + ... + a_p^i x_p = \sum_{j=1}^{p} a_j^i x_j + a_0 \tag{1}$$

The aggregation of the local models is realized in the form

$$\hat{y}_k = \sum_{i=1}^{c} A_{ik} f_i(\mathbf{x}_k, \mathbf{a}_i) \tag{2}$$

where we use a shorthand l_{ik} notation to denote $A_{ik} = u_i^m(\mathbf{x}_k) \Big/ \sum_{i=1}^{c} u_i^m(\mathbf{x}_k)$, with $u_i(x_k)$ being the ik-th element of the partition matrix.

We optimize the structure of this fuzzy model by minimizing the squared error of the differences between the output of the model and the data.

Note higher order local models (say 2nd order polynomials) can be constructed in the same manner.

IV. EXPERIMENTAL STUDIES

Through the series of experiments reported in this section, we quantify the performance of the model and analyze the resulting structure as well as draw several observations pertaining to the overall design process and discuss an impact of various parameters of the model on its performance.

The structural optimization of the model (selection of a subset of input variables) is realized through the use of the floating-point version of Genetic Algorithm (GA). The values of the crossover and mutation rates are equal to 0.8 and 0.05, respectively. They are in line of the values being encountered in the literature.

Synthetic data

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

A two-dimensional synthetic data set is generated from the nonlinear relationship of the form f($\mathbf{x}_1$, $\mathbf{x}_2$) = 0.8*sin($\mathbf{x}_1$) + 0.2*sin(2*$\mathbf{x}_2$) with the inputs assuming values in-between -4 and 4 ($\mathbf{x}_1$) and 10 and 18 ($\mathbf{x}_2$). The training and testing data comprise 240 and 160 input-output data being uniformly distributed throughout the input space.

As we have only two input variables, there is no need to carry out genetic optimization to select a subset of input features. Fig. 2 shows that there is a clearly visible minimum of Q in a series of m (from 1.5 to 10 with a step of 0.1) by using two features when c is equal to four.

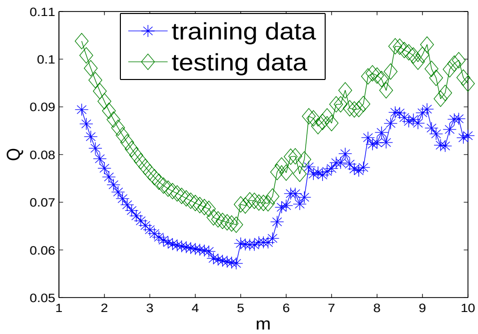

Figure 2. Performance index Q as a function of “m” – results shown for the training and testing data (c = 4)

Machine Learning data- Auto MPG data set

We present the results for one Machine Learning data set by focusing on the performance of the fuzzy models being treated as functions of the granularity of the architecture (c) and the dimensionality (p) of the reduced feature (input) space.

In auto-MPG data set, the automobile’s fuel consumption expressed in miles per gallon is to be the output of the model. The dataset includes 392 input–output pairs (after removing incomplete instances) where the input space involves 6 features. 10 fold cross-validation is adopted to quantify the performance achieved on the training and testing data. The results are visualized in Figure 3 where we show the values of the performance index for the training and testing data (both in terms of the mean value and standard deviation) for selected values of “m”.

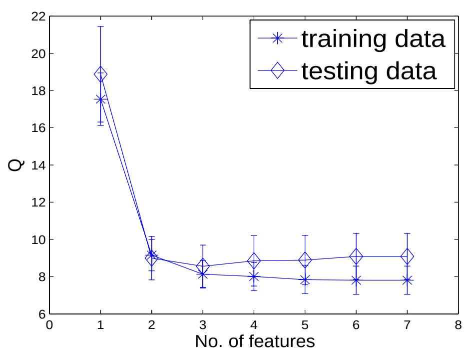

Figure 3. Performance index (mean value and standard deviation) versus the dimensionality of the successively reduced feature space for c = 2, m = 1.5

It becomes apparent that the set of input variables can be significantly reduced by retaining only 2 or 3 inputs and this reduction is possible irrespective of the values of the fuzzification coefficient.

The results being reported in a tabular format in Table 1 focus on the details of the feature space by showing subsets of input variables (features), which form the reduced feature space; these relationships are reported for selected values of “m”. The notation used here shows combinations of inputs which appear most often (results contained in the first brackets) which is followed by frequency it appeared (the result in the second bracket). For instance, for p = 3 and m = 2, the reduced subset of inputs is (3, 4, 6) and in the 10 fold cross validation it appeared 10 times out of 10 times. We can observe that the subsets of features are stable (viz. they appear quite consistently over all repetitions of the experiments) and the increase of the dimensionality of the input space results in adding new features while retaining the smaller subset that has been already identified. With this regard, the growth of the input space results in the sequence of the inputs.

p = 1: weight (4)
p = 2: weight (4), model year (6)
p = 3: weight (4), model year (6), horsepower (3)
p = 4: weight (4), model year (6), horsepower (3), displacement (2)
p = 5: weight (4), model year (6), horsepower (3), displacement (2), number of cylinders (1)

These subsets of inputs are intuitively appealing and reveal an interesting relationship between fuel consumption and the main characteristics of vehicles. If only a single input is to be considered then the weight comes into the play. For higher dimensionality of input spaces, the year of the model is to be considered and next horsepower and displacement start to appear in the realization of the model.

V. CONCLUSIONS

The interpretability of the fuzzy model relates to the dimensionality of the input space and thus its reduction becomes one of the efficient ways of increasing the transparency of the model. The reduction problem is of combinatorial nature, in which we can resort ourselves to the methods of Evolutionary Computing and swarm optimization. The level of reduction varied from data to data and this could have been anticipated.

In some cases it was shown that the reduced space led to the better performance of the models in terms of the resulting accuracy. We also demonstrated that the optimization of the fuzzification coefficient impacts the quality of the model. Furthermore the fuzzification coefficient relates to the interpretability of the rules considering that the level overlap between information granules translates into the level of interaction between the rules. Further investigations can include nonlinear local models in which case we may envision the ability to use a lower number of rules (local models) by accommodating more sophisticated (nonlinear) local models.

REFERENCES

[1] D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989.W.-K. Chen,

Linear Networks and Systems (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[2] L. O. Hall, I. B. Özyurt, and J. C. Bezdek, Clustering with genetically optimized approach, *IEEE Trans. Evolutionary Computing*, 3, 2, 1999, 103–112.

[3] A. Mukhopadhyay, U. Maulik, and S. Bandyopadhyay, Multiobjective Genetic Algorithm-Based Fuzzy Clustering of Categorical Attributes, *IEEE Trans. Evolutionary Computation*, 13, 5, 2009, 991-1005.

[4] Leng Gang, T. M. McGinnity, and G. Prasad, Design for Self-Organizing Fuzzy Neural Networks Based on Genetic Algorithms, ***IEEE Trans. on Fuzzy Systems***, 14, 6, 2006, 755-766.

[5] W. Pedrycz, M. Reformat, Evolutionary fuzzy modeling, ***IEEE Trans. Fuzzy Systems***, 11, 5, 2003, 652–665.

[6] I. Jagielska, C. Matthews, and T. Whitfort, An investigation into the application of neural networks, fuzzy logic, genetic algorithms, and rough sets to automated knowledge acquisition for classification problems, *Neurocomputing*, 24, 1-3, 1999, 37–54.

[7] M. Russo, FuGeNeSys—A fuzzy genetic neural system for fuzzy modeling, *IEEE Trans. Fuzzy Systems*, 6, 3, 1998, 373–388.

[8] J.M. Mendel, R.I. John, F. Liu, Interval Type-2 Fuzzy Logic Systems Made Simple, *IEEE Trans. on Fuzzy Systems*, 14, 6, 2006, 808-821.

[9] J. Kennedy and R. C. Eberhart, Particle swarm optimization, *Proc. IEEE Int. Conf. Neural Network*, Perth, Australia, 4, 1995, 1942–1948.

[10] R. C. Eberhart and J. Kennedy, A new optimizer using particle swarm theory, *Proc. 6th Int. Symp. Micromachine Human Sci*., Nagoya, Japan,, 1995,39–43.

[11] J. Kennedy, R. C. Eberhart, and Y. H. Shi, Swarm Intelligence, San Mateo, CA: Morgan Kaufmann, 2001.

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)