

# A Projection Based Method for Sparse Fuzzy System generation<sup>Δ</sup>

CHONG, A.<sup>1</sup>, GEDEON, T.D.<sup>1</sup>, KOCZY, L.T.<sup>1,2</sup>

<sup>1</sup>School of Information Technology  
Murdoch University  
South Street, Murdoch, WA, 6150  
AUSTRALIA

<sup>2</sup>Department of Telecom & Telematics  
Budapest University of Technology and Economics  
HUNGARY

*Abstract:* - A projection based method for sparse fuzzy system generation is proposed. Given a set of training data, clustering is first performed on the output space. Data points from each output cluster are projected back to each input dimension forming one-dimensional clusters. The clusters from different dimension are then merged to form fuzzy rules. Experiments have confirmed the effectiveness of the proposed technique.

*Key-Words:* - rule extraction, clustering, cluster validity, sparse fuzzy system, box-jenkins, fuzzy modeling

## 1 Introduction

Fuzzy Systems suffer from rules explosion. To model a system with  $k$  variables and maximum  $T$  fuzzy terms in each dimension, the number of necessary rules is  $O(T^k)$ , which will be very large if  $k$  is not very small. Because of this, fuzzy systems are limited to handle only very few variables.

To widen the range of problems controllable by fuzzy rule-bases, it is essential to reduce  $T$ ,  $K$ , or both. Hierarchical fuzzy system [1] is proposed to reduce  $K$ . On the other hand, decreasing  $T$  leads to sparse fuzzy systems, i.e. fuzzy rule-bases with “gaps” between the rules. In such fuzzy systems, there often exist observations (inputs) that do not match any of the rule antecedents. Fuzzy rule-interpolation [2] is used to infer the conclusions for such observations.

Among the rule extraction techniques proposed in the literature, Sugeno and Yasukawa’s technique (SY) [3] is one of the earliest works that offer a potential way for sparse rule-base generation. The SY approach clusters only the output data and induces the rules by computing the projections to the input domains of the cylindrical extensions of the fuzzy clusters. This way, the method produces only the necessary number of rules for the input-output sample data. The paper [3] discusses the proposed

technique at the methodological level leaving out some implementation details. The SY technique was further examined in [4, 5] where additional readily implementable techniques were proposed to complete the modeling methodology.

In this paper, a novel rule extraction technique that works on the projection of data and fuzzy clustering is introduced. The technique, known as projection-based fuzzy rule extraction (PB), extends the idea of the SY approach for fuzzy modeling. The goal of the rule extraction technique is to automate the construction of a fuzzy rule base from a set of input-output sample data.

This paper is organized as follows. Fuzzy c-Means clustering and the cluster validity problem is discussed in section 2. The proposed technique is discussed in sections 3 and 4. Section 5 reports the experimental results and discussion. The conclusion is presented in section 6.

## 2 Fuzzy Clustering

Given a set of data, Fuzzy c-Means clustering (FCMC) [6] performs clustering by iteratively searching for a set of fuzzy partitions and the associated cluster centers that represent the structure of the data as best as possible. The FCMC algorithm relies on the user to specify the number of

---

<sup>Δ</sup> Research supported by the National Scientific Research Fund OTKA T034233 and T034212, the Bilateral Hungarian-Portuguese Research Cooperation Fund P-14/01, and the Australian Research Council.

clusters present in the set of data to be clustered. Given the number of clusters  $c$ , FCMC partitions the data  $X = \{x_1, x_2, \dots, x_n\}$  into  $c$  fuzzy partitions by minimizing the within group sum of squared error objective function as follows (eqn 1).

$$J_m(U, V) = \sum_{k=1}^n \sum_{i=1}^c (U_{ik})^m \|x_k - v_i\|^2, \quad 1 \leq m \leq \infty \quad (1)$$

where  $J_m(U, V)$  is the sum of squared error for the set of fuzzy clusters represented by the membership matrix  $U$ , and the associated set of cluster centers  $V$ .  $\|\cdot\|$  is some inner product-induced norm. In the formula,  $\|x_k - v_i\|^2$  represents the distance between the data  $x_k$  and the cluster center  $v_i$ . The squared error is used as a performance index that measures the weighted sum of distances between cluster centers and elements in the corresponding fuzzy clusters. The number  $m$  governs the influence of membership grades in the performance index. The partition becomes fuzzier with increasing  $m$  and it is proven that the FCMC algorithm converges for any  $m \in (1, \infty)$  [6]. The necessary conditions for eqn 1 to reach its minimum are

$$U_{ik} = \left( \sum_{j=1}^c \left( \frac{\|x_k - v_i\|}{\|x_k - v_j\|} \right)^{2/(m-1)} \right)^{-1} \quad \forall i, \forall k \quad (2)$$

and

$$v_i = \frac{\sum_{k=1}^n (U_{ik})^m x_k}{\sum_{k=1}^n (U_{ik})^m} \quad (3)$$

In each iteration of the FCMC algorithm, matrix  $U$  is computed using eqn 2 and the associated cluster centers are computed as eqn 3. This is followed by computing the square error in eqn 1. The algorithm stops when either the error is below a certain tolerance value or its improvement over the previous iteration is below a certain threshold.

The FCMC algorithm cannot be used in situations where the number of clusters in a set of data is not known in advance. Since the introduction of FCMC, a reasonable amount of work has been done on finding the optimal number of cluster in a set of data. This is referred to as the cluster validity problem. The optimal number of clusters are determined by means of a criterion, known as the cluster validity index. Fukuyama and Sugeno (FS) proposed the following cluster validity index [7]:

$$S(c) = \sum_{k=1}^n \sum_{i=1}^c (U_{ik})^m (\|x_k - v_i\|^2 - \|v_i - \bar{x}\|^2) \quad 2 < c < n \quad (4)$$

where  $n$  is the number of data points to be clustered;  $c$  is the number of clusters;  $x_k$  is the  $k^{\text{th}}$  data,  $\bar{x}$  is the average of data;  $v_i$  is the  $i^{\text{th}}$  cluster center;  $U_{ik}$  is the membership degree of the  $k^{\text{th}}$  data with respect to the  $i^{\text{th}}$  cluster and  $m$  is the fuzzy exponent. The number of clusters,  $c$ , is determined so that  $S(c)$  reaches a local minimum as  $c$  increases. The terms  $\|x_k - v_i\|$  and  $\|v_i - \bar{x}\|$  represent the variance in each cluster and variance between clusters respectively. Therefore, the optimal number of clusters is found by minimizing the distance between data and the corresponding cluster center and maximizes the distance between data in different clusters.

It is reported in [8] and [9] that the FS index performs unreasonably when dealing with clusters that are close to one another. Typically, an unreasonably large number of clusters is obtained. This is a serious drawback for fuzzy modeling since a large number of clusters can lead to an explosive number of fuzzy rules. To overcome this problem, a hybrid approach for the cluster validity problem was proposed in [8]. The technique has two steps. In the first step, a cluster validity index is used to find a rough estimation of the optimal number of clusters. The number is later refined in the second step by means of the following merging index:

$$P(v) = \sum_{j=1}^n e^{-4 \left| \frac{(v-x_j)/(v_i-v_j)}{2} \right|^2} \quad (5)$$

where  $x_j$  is the  $j^{\text{th}}$  data and  $v$  is a cluster center. For each pair of cluster centers  $v_i$  and  $v_j$ , the index (eqn 5) is calculated for  $v_i$ ,  $v_j$ , and  $v_m$  where  $v_m$  is the middle point  $(v_i + v_j)/2$ . If  $p(v_m)$  is smaller than both  $p(v_i)$  and  $p(v_j)$ , then the centers stay un-merged. Otherwise, they should be merged. The algorithm to find the optimal number of clusters is as follows [8]:

1. Compute a rough estimate  $c$  of the optimal number of clusters using the FS index.
2. Let  $C = \{c_1, c_2, \dots, c_n\}$  be a set of  $n$  cluster centers. Find a pair  $\langle c_p, c_q \rangle \in C$  that should be merged based on eqn 5. There should not be any other cluster center that comes between  $c_p$  and  $c_q$  in the multi-dimensional space.
3. If the pair  $\langle c_p, c_q \rangle$  is successfully found, decrease the number of clusters by one and perform fuzzy clustering on the data. Repeat step 2 until no more clusters can be merged.

### 3 Projection Based Rule Extraction

In this section, our novel technique will be discussed. The algorithm consists of 6 steps.

1. Perform Fuzzy c-Means clustering on the output space. The FS index is used to determine the optimal number of clusters. We remark that our algorithm does not place any restriction on the clustering technique and validity index used.
2. For each fuzzy output cluster  $B_i$  approximated, all the points belonging to the cluster are projected back to each of the input dimensions. For each dimension, fuzzy clustering is again applied to the projection of the points. In this step, the FS index is used in conjunction with the merging index (see section 2).
3. The previous step results in multiple 1D fuzzy clusters in each input dimension. For each fuzzy cluster, a trapezoidal cluster is approximated. We refer the reader to [4] for a simple trapezoidal cluster approximation technique. The partition is converted to a Ruspini partition [10] for the convenience of the latter steps.
4. Each of the  $n$  clusters ( $C_{d1} - C_{dn}$ ) in the input dimension  $d$ , is a projection of the multi-dimensional input cluster to that input dimension. Next the clusters from individual dimensions are combined to form the multi-dimensional input cluster. The merging process is described in section 4.
5. For each of the multi-dimensional clusters identified, a rule can be formed. For example, if a multi-dimensional cluster is formed with  $[C_{11}, C_{23}, C_{34}]$  for the points projected from output cluster  $B_i$ , we obtain the following rule:  
*If  $x_1$  is  $C_{11}$  and  $x_2$  is  $C_{23}$  and  $x_3$  is  $C_{34}$  then  $y$  is  $B_i$*   
 Where  $C_{dn}$  is the  $n^{\text{th}}$  cluster identified at input dimension  $d$ .
6. The completed fuzzy rule-base then goes through a parameter identification process where each trapezoidal cluster in the input and output space is adjusted to improve the overall performance. The parameter identification is described in [3]. An alternative technique is proposed in [4].

### 4 Cluster Merging

The reconstruction of multi-dimensional clusters by combining the 1D clusters identified at each dimension can be problematic. Let  $m$  be the average number of clusters identified at each dimension. The total number of possible combinations is  $m^d$ . Since the number of combination grows exponentially with the increase of dimensions, examining every combination of the 1D clusters is computationally intractable. In this section, we propose a fast merging technique.

The merging process involves the use of a threshold  $t$ . The cluster in the multi-dimensional space is determined to be the region where the number of projected points in the region exceeds  $t$ . A point  $p$  is contained in the cluster  $C_i$  if  $\mu_{C_i}(p) > \mu_{C_j}(p)$  for all  $j \neq i$ .

The 3-step algorithm is presented.

1. Find one of the multi-dimensional clusters  $C$  where the number of points that falls into its projection exceeds the threshold  $t$ .
2. Remove all data points that are contained in the cluster  $C$  approximated.
3. Repeat steps 1 – 2 until no more cluster can be found.

The pseudo-C-code for step 1 of the algorithm is presented as follows.

#### PROCEDURE find\_MD\_cluster

```

Let  $U_i$  be the set of one-dimensional clusters
in dimension  $i$ 
Let mdCluster = [ ]
for  $i = 1$  to  $k$ 
  for each unit  $u \in U_i$ 
    utemp = mdCluster x  $u$ 
    if utemp is dense
      denseunit = utemp
      break
    end if
  end for
end for

```

For the convenience of discussion, we define  $[ ]$  as the zero-dimensional (empty) cluster where  $[ ] \times C_i = C_i$ . The algorithm scans through each of the  $k$  dimensions to find one of the multi-dimensional clusters in the data, giving the complexity  $O(k)$ . Having identified a cluster, all the data points that are contained within the cluster are removed. The

process is repeated until no more clusters can be found. The overall complexity is  $O(ck)$  where  $c$  is the total number of clusters in the data. Since the complexity of the algorithm is linear, it is computational feasible to deal with data with very large numbers of dimensions.

The threshold  $t$  governs the degree of sparseness in the rule-base to be generated. The higher the threshold, the fewer multi-dimensional clusters will be obtained and consequently fewer rules are generated.

## 5 Experimental Results And Discussion

Two experiments have been carried out to validate the effectiveness of the proposed technique. In each experiment, a fuzzy rule base is generated from a set of input-output sample data using the proposed rule extraction technique. The same set of data is then used with the fuzzy rule base generated to produce a set of outputs. This is followed by the evaluation of the accuracy of the fuzzy rule base output. In this study, the mean square error of output is used as a performance index (equation 6).

$$PI = \sum_{i=1}^m (y^i - \hat{y}^i)^2 / m \quad (6)$$

where  $m$  is the number of data,  $y^i$  is the  $i^{\text{th}}$  actual output and  $\hat{y}^i$  is the  $i^{\text{th}}$  model output. The lower the performance index, the more accurate the fuzzy rule base.

The threshold  $t = 0.01$  (i.e. one percent of the data) is used in the merging process (see section 4). The use of the threshold leads to the removal of some multi-dimensional clusters that are less dense. This allows the generated rule-base to have ‘gaps’. Due to the gaps, some of the data points may not be covered by any of the fuzzy rules generated. We remark that this should not be considered a failure or weakness of the algorithm. After all, one of the ultimate goals of our technique is to generate a sparse rule base. For data points that do not fire rules, fuzzy rule interpolation techniques [2] can be used to infer the output. These techniques have not been used here. In the experiments, when there are no rules to fire for a data point, the average of the range of each output variable is used as the default output.

For the first experiment, two results are presented. The first result shows the performance index of the whole data set whereas the second shows the performance index on only those data that can find rules to fire. The latter gives a better idea of the potential of this fuzzy rule base to be used with fuzzy rule interpolation techniques.

### 5.1 Experiment One

The method was used to model a quadratic function:

$$y = (1 + x_1^{-2} + x_2^{-1.5})^2, \quad 1 \leq x_1, x_2 \leq 5 \quad (7)$$

Five hundred input-output data were generated from the quadratic function. The output range is 1.3 - 6.9. Altogether 26 fuzzy rules were generated. During the evaluation, 3.6% of the data points cannot find rules to fire. The overall performance index of the system is 0.025. The performance index of the data that can find rules to fire is 0.018.

### 5.2 Experiment Two

The data given by Box and Jenkins [11], often used as a benchmark by research papers, is used in this experiment. There are 290 data points, where the input variable  $u$  is the methane concentration and the output variable  $y$  is the CO<sub>2</sub> concentration. The data has been used in different ways in the literature. Tables 1 and 2 summarize the performance of the proposed technique as well as other models that use the data as benchmark. The input  $y(t)$  and  $u(t)$  are the output and input at time step  $t$  respectively. In this experiment, all data can find rules to fire.

Method	M	PI
Tong [12]	19	0.469
Xu et al [13]	25	0.328
Pedryc [14]	81	0.320
<b>Proposed Method</b>	<b>41</b>	<b>0.144</b>

**Table 1** Number of rules, M, and performance index, PI, of models using  $y(t-1)$  and  $u(t-4)$  as input variables

Method	M	PI
Nakoula et al. [15]	90	0.175
Evsukoff et al. [16]	36	0.153
<b>Proposed Method</b>	<b>41</b>	<b>0.152</b>

**Table 2** Number of rules, M, and performance index, PI, of models using  $y(t-1)$  and  $u(t-3)$  as input variables

## 6 Conclusion

A novel rule extraction technique for generating sparse fuzzy rule-base has been proposed. Experiments have confirmed the feasibility and effectiveness of the proposed technique. In our next step, the integration of fuzzy interpolation into our technique will be investigated. We believe that the integration of such techniques in the current framework can increase the accuracy of the rule base and further reduce the number of fuzzy rules.

### References:

- [1] Koczy, L.T. *Approximative inference in hierarchical structured rule bases*. in *Fifth IFSA World Congress*. 1993. Seoul: International Fuzzy Systems Association.
- [2] Koczy, L.T. and Hirota, K., *Interpolation in fuzzy rule bases*. 1993.
- [3] Sugeno, M. and Yasukawa, T., *A fuzzy-logic-based approach to qualitative modeling*. IEEE Transactions on Fuzzy Systems, 1993. **1**(1): p. 7-31.
- [4] Tikk, D., Gedeon, T.D., Koczy, L.T., and Biro, G., *Implementation details of problems in Sugeno and Yasukawa's qualitative modelling*. IEEE Transactions on Fuzzy Systems, 2002.
- [5] Wong, K.W., Koczy, L.T., Gedeon, T.D., Chong, A., and Tikk, D. *Improvement of the Clusters Searching Algorithm in Sugeno and Yasukawa's Qualitative Modeling Approach*. in *Proceedings of 7th Fuzzy Days in Dortmund*. 2001.
- [6] Bezdek, J.C., *Pattern Recognition with Fuzzy Objective Function Algorithms*. 1981, New York: Plenum Press.
- [7] Fukuyama, Y. and Sugeno, M. *A new method of choosing the number of clusters for fuzzy c-means method*. in *Proceedings of the 5<sup>th</sup> Fuzzy System Symposium*. 1989.
- [8] Chong, A., Gedeon, T.D., and Koczy, L.T. *A Hybrid Approach for Solving the Cluster Validity Problem*. in *submitted to 14th International Conference on Digital Signal Processing*. 2002. Greece.
- [9] Yang, M.S. and Wu, K.L. *A New Validity Index For Fuzzy Clustering*. in *The 10th IEEE International Conference on Fuzzy Systems*. 2001. Melbourne, Australia.
- [10] Ruspini, E.H., *A new approach to clustering*. Information and Control, 1969. **15**: p. 22-32.
- [11] Box, G.E.P. and Jenkins, G.M., *Time Series Analysis: Forecasting and Control*. 1970, San Francisco: Holden Day.
- [12] Tong, R.M., *The evaluation of fuzzy models derived from experimental data*. Fuzzy Sets and Systems, 1980. **4**: p. 1-12.
- [13] Xu, C.W. and Lu, Y.Z., *Fuzzy model identification and self-learning for dynamic systems*. IEEE Transactions on Systems, Man, and Cybernetics, 1987. **17**(4): p. 683-689.
- [14] Pedrycz, W., *An identification algorithm in fuzzy relational systems*. Fuzzy Sets and Systems, 1984. **13**: p. 153-167.
- [15] Nakoula, Y., Galichet, S., and Foulloy, L., *A learning method for structure and parameter identification of fuzzy linguistic models*, in *Selected Approaches to Fuzzy Model Identification*. 1997, Springer Verlag. p. 281-319.
- [16] Evsukoff, A., Branco, A.C.S., and Galichet, S. *Incremental fuzzy modeling*. in *Proceedings of The 10th IEEE International Conference on Fuzzy Systems*. 2001. Australia.